Measuring the Cookie Setting Behavior of Web Pages Showing Privacy Warnings

Bartosz Struzinski

MInf Project (Part 1) Report

Master of Informatics
School of Informatics
University of Edinburgh

2021

Abstract

With the introduction of GDPR, the domains serving content to EU-based users have been forced to eliminate, or at least reduce, user tracking activities. Following this legislation, a considerable portion of websites has started showing Cookie Privacy Warnings, prompting their visitors to agree to the use of technologies, mostly HTTP cookies, aiming at monitoring user behavior and collecting personal data. The goal of this study is to measure how user interaction with these privacy warnings affects the tracking environment developed around that user. To that end, a privacy research platform, called Cookie Crumble Tracer (CCT), has been created. CCT is a Selenium-based crawler that visits the top 1000 most popular domains according to Tranco ranking. While crawling these websites, CCT interacts with the aforementioned GDPR Cookie Privacy Warnings to either agree (opt-in) or disagree (opt-out) to the server setting tracing cookies. By doing so, CCT is capable of measuring how the user's privacy policy affects the characteristics of cookies saved in the browser and the outgoing HTTP traffic. This information is then used to create network graphs representing the tracking ecosystems for opt-in and opt-out policies. CCT constructs two types of graphs - Publishers-Trackers (PT) and Trackers-Trackers (TT). PT graphs represent the connections between the trackers and the publishers hosting them. TT graphs represent pairs of trackers that perform cookie synchronization - a mechanism trackers use to share user identifying data. CCT finds that around 50% of the analyzed domains contain a GDPR Cookie Privacy Warning. By using graph analysis in form of various centrality metrics, CCT finds that opting out can reduce the amount of data shared between the trackers by 40% and the number of collaborating third-party pairs by 35%. Opting out also decreases the size of the tracking ecosystem, reducing the number of trackers by 38% in PT graphs and 17% in TT graphs. However, it is found that the user's cookie policy seems to have no significant effect on the structure of the online ecosystem, and poses no threat to the dominance of well-known companies, such as Google or Facebook.

Acknowledgements

First and foremost I would like to express my deepest appreciation to my supervisor, Prof. Kami Vaniea, for her continuous support and invaluable advice during this project. Her knowledge and academic experience helped me to formulate my research goals and motivated me at every step of my research.

Second, I would like to thank my Mother, whose relentless support and candid comments have always encouraged me to raise the bar higher. I will never be able to express my gratitude for the countless times she helped me throughout my journey at the university and beyond.

Finally, I cannot express enough thanks to my Father. His hard-working personality, wisdom, and outstanding conscientiousness will always lead by example and drive me to excel at whatever I do. Dad, I can only hope I am making you proud.

Table of Contents

Chapter 1

Introduction

The last several months, heavily influenced by worldwide pandemic, have certainly proved one thing - there is hardly anything that the Internet fails to provide. Whether it is making groceries or attending a university - the spectrum of possibilities available through the World Wide Web has been a blessing to many of us. However, there is one commodity, which - even though in high demand - seems to be notoriously unavailable via that ever-growing network.

Privacy

The dependence on electronic devices, interconnected and communicating online, moved a formidable part of our lives into the digital domain. As the pool of web users grew, the economic and political incentives encouraged some parties to use that environment for profit or political advantage. Nowadays, entire business models have been formed around collecting and sharing user data, often treated as a company's economic asset. This has led to the development of sophisticated tracing technologies, which, for a user with no technical background, are hard to remove or even detect. *HTTP Cookies* is one of the web technologies adopted for user tracking. Even though cookies are critical to providing a good user experience in multiple web services, such as social networks or online retail, they are also the most widely used method of tracing online activity [1], and may be used to collect Personally Identifiable Information (PII) [2].

To address the issue of online privacy, policymakers have been trying to come up with legislation aiming at maintaining user privacy online and increasing the transparency of tracking. Arguably the most famous, and the strictest legislative data protection framework to date, is the European Union's General Data Protection Regulation. The main goal of GDPR is to broaden the scope of personal data and limit user tracking activities. Furthermore, it requires the websites to receive online visitors' informed consent, in form of affirmative action, for any potential tracking, data collection, or data sharing they may do [3]. To comply with the rules outlined by GDPR, internet domains of companies operating within the European Economic Area started displaying Cookie Privacy Warnings asking for user consent to data collection and profiling techniques.

The influence of GDPR on the online advertising and tracking ecosystem has been a subject of many studies, most important of which are detailed in Chapter 2. Previous

work (e.g., [4], [5] [6]) have researched the aftermath of GDPR and its effect on online surveillance ecosystem, showing a reduction in tracking. Other studies, such as [4], have shown that not all domains comply with new legislation - around 50% of domains still set tracing cookies before the user's consent. There has also been research into the evolution of general structure of tracking ecosystem over the past few years (e.g., [3], [7], [8]).

The goal of this study is fundamentally different from all the previous work that has been done - instead of focusing on the large-scale view of the post-GDPR world and collecting data from a large group of users whose browsing pattern is unstandardized, the focus is put on an individual user. How, using the opportunities provided by GDPR, can the individuals affect the tracking ecosystem that gradually builds around them as they surf the web? How can their individual decisions to accept or reject cookies impact the frequency with which trackers share the information about them? To answer these questions, a specialized research platform has been developed, allowing for the detection of user tracking and identification of parties that exchange personally identifiable user data. This tool, whose design and implementation is discussed in Chapter 3, is named *Cookie Crumble Tracer*, or CCT for short. CCT is capable of controlling a Chrome browser to simulate human-like behavior and interact with the aforementioned privacy warnings introduced by GDPR. When interacting with these banners, CCT either agrees (opts in) or refuses (opts out) to accept any non-necessary cookies. It then analyzes the network traffic coming from the user's browser, as well as saved cookies, to detect privacy intruding activities. By repeating this procedure over the set of most popular domains, CCT is capable of measuring the extent to which user's decision to accept or reject cookies influences their tracking ecosystem.

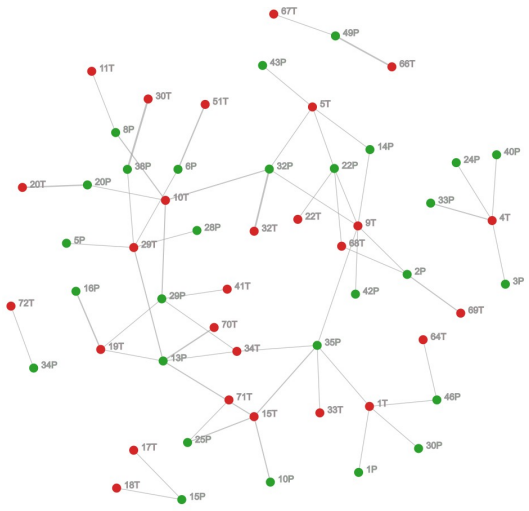

Figure 1.1: Network of Publishers and Trackers when the user opts out and rejects as many cookies as possible

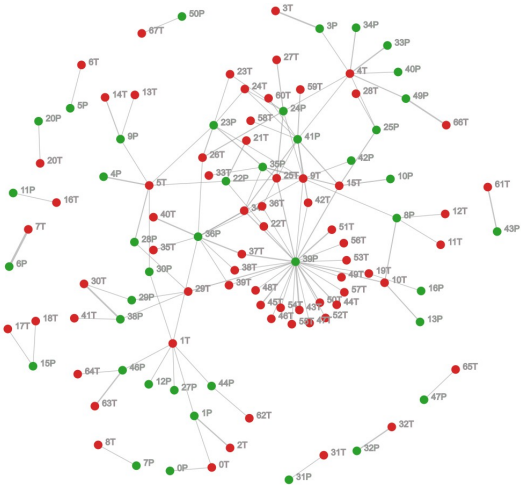

Figure 1.2: Network of Publishers and Trackers when the user opts in and accepts as many cookies as possible

Each of the above figures presents a network model of publishers and trackers, basing

on data collected and analyzed by CCT. Green dots are the publishers - domains a user explicitly visits. Red dots, on the other hand, are the trackers - domains embedded within publishers and setting tracking cookies in the user's browser. The network on the left presents a tracking ecosystem created around a user who consistently opts out from accepting all non-mandatory cookies. The one on the right presents the same network, but for a user opting in and hence accepting all possible cookies. The difference seems to be self-evident - a decision to accept or reject non-essential cookies affects the tracking ecosystem, increasing the number of observed trackers. What about the structure of that tracking ecosystem? In this study we dig deeper, trying to answer even more specific questions. What are the most important nodes within the network? Which trackers exchange the most information? How does the user's interaction with Cookie Privacy Warnings affect the network of trackers and publishers?

To answer these pressing questions, CCT crawls the set of top 1000 most popular domains [1] to locate the GDPR privacy warnings and their elements responsible for saving user's privacy settings. By interacting with these cookie dialogues, CCT influences the number and characteristics of cookies saved on the browser, as well as network traffic, which is then analyzed to detect any data sharing among potential privacy intruding parties. Using this data, CCT creates network graphs representing the tracking ecosystem. In addition to Publishers-Trackers network graphs, examples of which have been presented in Figures 1.1 and 1.2, CCT also creates Trackers-Trackers graphs, which represent the network of domains performing *Cookie Synchronization* - a hard-to-detect mechanism, allowing the trackers to share user identifiable data and perform server-to-server user database merges. To create the Publishers-Trackers network graphs CCT clears the browser from any cache or cookies remaining from visits to the previous websites, thus making the websites visits independent from each other. However, this policy changes when creating the Trackers-Trackers graph. In this case, CCT wants to exploit the cross-domain tracking capabilities of cookies, hence they are not cleared in between websites visits [2]. Having created these network graphs, CCT deploys various graph mining tools and metrics, such as (but not limited to) clustering coefficient, density, and degree centrality, to study the influence of user's cookie policy (either opt-in or opt-out) on the structure of the tracking ecosystem.

CCT findings, presented in Chapter 4, show that around 50% of all analyzed domains contain a GDPR Cookie Privacy Warning and around 90% of the domains set a potentially tracking cookie before any website interaction. The analysis of network models points out that user's decision to opt out can reduce the number of trackers in the ecosystem by 38%, and the number of collaborating third-party pairs by 35%. Opting out also reduces the amount of observed data sharing by 40%. However, the general structure of the tracing ecosystem is not significantly affected by the user's cookie policy, with well-known companies such as Google or Facebook firmly occupying the most important positions in the network.

[1] According to a ranking called Tranco, which is discussed in Section 3.2.2 on page 11

[2] The details of gathering the privacy data used for network graph creation are presented in Section 3.2.7 of Chapter 3, page 19

Chapter 2

Background

The following sections introduce background information and the context in which this study is placed. We start with a brief introduction to the policies aiming at preserving users' privacy on the Internet, carrying on to the technical background describing the use of cookies as a tracking and profiling technology. Finally, to show the uniqueness of this work, we present and discuss related literature from the past.

2.1 GDPR Enforcement and Online Privacy

General Data Protection Regulation [9], or GDPR for short, has been introduced to address the international aspects of the World Wide Web and to standardize data protection laws between its member states. This new legislation came into effect on May 25. 2018, introducing ground-shaking changes to how personal data can be collected, stored, processed, and shared. GDPR served as a follow-up to the ePrivacy Directive (ePD), known as *The Cookie Law*, which made it mandatory to ask for user's consent before storing or accessing any non-necessary data. As a consequence of ePD, domains accessible from within the EU started showing banners asking the visitors for their consent or informing them about the use of cookies. With GDPR further regulating the definition of user's consent, from that point on it is defined as a clear, affirmative, purpose-specific, and informed indication of agreement to the processing of personal data [9]. This new definition has ultimately forced domains operating within the EU to display a new generation of Cookie Privacy Warnings, which now require the user to give consent before interacting with the website, usually by clicking a button or submitting some kind of a form. An example of a GDPR Cookie Privacy Warning is presented in Figure 3.4 (page 18).

2.2 Cookies and Modern Vision of HTTP

HTTP, or Hyper Text Transport Protocol, is an application layer protocol and the networking language behind each browser. Initiated by Tim Berners-Lee at CERN as part of the WWW project, this protocol is used to request web page content from the

server. As HTTP was primarily designed to be fast and efficient, it is a stateless protocol, i.e. each HTTP request has no memory of the previous requests to the same server. This means that sessions could not be initially supported by HTTP, leading to the introduction of session cookies in 1994, which act as a state management mechanism for the normally stateless protocol. Cookies are essentially small text files, up to 4096 bytes, installed on the client's side. They contain **`name=value`** pairs and metadata, for example an expiration date. Cookies are usually set on the first visit to the domain, and can be installed in the browser in two different ways: an API call to the server performed by embedding JavaScript in the website's source code, or through the `Set-Cookie` header of an HTTP response. In both cases, these files are stored by the browser to be sent back on subsequent requests to the server. The decision on whether or not a cookie should be included in the HTTP request is done based on attributes `Expires`, `Max-Age` (or both), which specify the time after which the cookies should be deleted.

2.2.1 Types of Cookies

Cookies have multiple applications, all of which operate thanks to the cookie's core functionality - enabling the server to modify its actions based on the information contained within the cookie file. *Session cookie* may allow a server to authenticate the user and keep him logged in until the browser is closed. By keeping the cookies even when the user exits the browser, cookies can be utilized by the server to serve the client with personalized content, for example, website language - in this case, we deal with *personalization cookies*.

Tracing cookies, which among others allow for collecting and analysis of client's data, are the main focus of privacy-focused research. While cookies used for authentication purposes are only set at the moment of logging in and are kept in short-term volatile memory, tracing cookies are persistent, meaning that once issued, they are stored on non-volatile memory for a long period.

Relevant to our study is the notion of *first-party* and *third-party cookies*. First-party cookies are stored on the client's side by the host domain, which is the domain the user explicitly visits. These cookies are supposed to facilitate the user experience of web surfing, for example by enabling the browser to remember which items a user added to the shopping cart.

Third-party cookies are considered more privacy-intruding. A third-party cookie is one created by a domain different than the one explicitly visited. Third-party cookies embedded across multiple websites may origin from the same domain, for instance, the same ad network, which allows the third-parties to use a single tracing third-party cookie across many domains. By setting the `Referrer` HTTP header to the address of the domain from which the request originated, third-parties can analyze the user's cross-site browsing pattern and learn their browsing history.

An important concept in the area of third-party cookies is the *Same Origin Policy*. This security mechanism allows a document, script, or any other resource loaded from one

origin to interact only with objects and resources from the same origin [1]. It helps to isolate potentially malicious documents and prevent them from accessing other web pages' data through its Document Object Model. [2]

2.2.2 Cookie Synchronization

Cookie Synchronization is a mechanism used to bypass the above Same Origin Policy. Cookie synchronization relies on tracking domains sharing pseudonymous IDs, stored in cookies and associated with a specific user, amongst each other. According to Google's developer guide to cookie synchronization, this mechanism provides a way for domains, which normally cannot read each other's cookies, to share cookie values and improve user targeting. Cookie synchronization is mostly a three-step process.

1. User makes a direct visit to a host domain, call it `host.org`. A script from one third-party, say `foo.org`, is loaded into host's website DOM.
2. The request loading the script is then redirected, or the embedded script belonging to `foo.org` makes a separate request to a partnering third-party, let's call it `dummy.com`. This redirected/new HTTP request contains the unique ID that `foo.org` assigned to the end user. Through the aforementioned `HTTP Referrer` header filed, `dummy.com` knows that the user with that specific ID has visited `host.org`.
3. If `dummy.com` has already stored a cookie on the client's side (because that user could have visited another website with cookie-requesting scripts belonging to that third-party), `dummy.org` is capable of pairing the ID assigned to the client by `foo.org` with its identification number.

In this way, `foo.org` and `dummy.com` can exchange the information they have on a particular user over a separate channel, which neither the website publisher nor the tracked user is aware of. This technique has been criticized from the point of preserving individual's privacy on the Internet, as no user consent is required for the third-parties to share profiling data.

2.3 Related Research

The introduction of GDPR has been the motivation behind numerous studies focused on investigating the evolution of the tracking ecosystem and the impact of this legislation on individual's privacy. Dabrowski et al. [4] assessed the impact of GDPR on browser cookie setting behavior. In their study, cookies were collected from Alexa Top 100,000 websites by accessing these domains both from within and outside the European Economic Area. Furthermore, the changes in cookie setting behavior between the pre and post-GDPR era have been investigated by comparing the results from 2018 with the data set created in 2016. Dabrowski et al. found that only 12.4% domains

[1] Two URLs have the same origin if the protocol, port, and host are the same for both. [10]

[2] In the context of cookies, Same Origin Policy means that if facebook.com sets a cookie, only facebook.com can access and read it.

restricted its cookie storing when accessed from within the EEA, relative to when accessed from the USA. This study concluded that 49.3% of cookie-using websites from the Alexa Top 1000 websites, and 26% of Alexa Top 100,000 ranking, refrain from setting cookies without the user's consent when facing an EU visitor. Moreover, according to the study, due to the legal impact of GDPR, 46.7% of the top 100,000 websites have refrained from installing non-consensual cookies as compared to the study from 2016.

Other studies have focused on measuring websites' compliance with GDPR. Trevisan et al. [11] performed a study shortly after GDPR came into effect and analyzed more than 35,000 websites, concluding that 49% of the visited websites do not comply with the legislation. In this research, a manual study on cookie banners has been performed to analyze the storage of cookies after the user's consent. It has been observed that 28% of studied domains do not provide any cookie notices and out of the 72% remaining, only 7% do not store any cookies before the user decides on the consent. Degeling et al. [5] quantified the privacy policy changes on Top-500 domains of 28 EU countries. Their work reports that 85% of these domains have a privacy policy. It also points out that GDPR has not significantly changed how third-party cookies are utilized. The lack of significant changes in the general state of the web has also been reported by Sorensen et al. [6], who measured the changes in the presence of third-parties due to GDPR, showing that it has a potential effect only on specific types of domains, such as retail or entertainment.

Particularly important to this work are studies which use network graphs, such as the ones visualized in Figures 1.1 and 1.2, to dissect the online tracking ecosystem and analyze its inner mechanisms. In that direction, Solomos et al. [3] constructed bipartite (two modal) graphs connecting third-parties with their hosts, for different time snapshots. Then, based on these "publishers-to-trackers" networks, their study utilized a process known as bipartite graph projection, to extract "tracker-to-tracker" graphs from the original, "publishers-to-trackers" ones, connecting third-parties sharing common hosts, thus revealing potential collaborations. These graphs are then compared with datasets of parties performing cookie synchronization in the past, revealing a 47% to 81% overlap. Just like Solmos et al., Kalavri et al. [7] also creates a bipartite, two modal graphs, based on real user traffic logs, representing relations between embedded third-parties and their hosts. Their analysis focused on communities formed by graph vertices and showed that third-parties are well connected since 94% of them are in the network graph's largest connected component. Urban et al. [12] used emulated users located in 20 countries within European Union to collect behavioral data and create a cookie synchronization graph connecting third-parties that have been observed to use this data sharing mechanism. Their study reports that, although GDPR does not significantly affect the structure of the online tracking ecosystem, it has a significant impact on the number of observed cookie synchronizations, which decreases by 40%.

This study takes inspiration from many of the works described in the paragraph above. Similarly to [3], the collected data is used to create bipartite graphs presenting host domains with embedded third-parties. Unlike their study, however, we do not artificially create another, "trackers-to-trackers" graph, out of them. Instead, an algorithm is crafted to separately detect cookie synchronization and use that data to create the

corresponding network graphs. In that regard we take the approach similar to [12] and [8]. However, there is one aspect that distinguishes this study from all the previous ones, making it unique in this area. The aforementioned pieces of work were longitudinal studies, in which data was collected by using traffic logs of many individuals, who volunteered for these experiments. It has not been documented what the policy towards cookies these users had, and certainly, this policy was not consistent among the participants [3]. For this reason, this study focuses on a single user and tries to measure how a consistent cookie policy of an individual person affects the state of the unique online tracking ecosystem which that person builds around themselves.

2.4 Ethical Considerations

The methodology of data collection in this study does not involve any human subjects, and is based on entirely passive web behavior - it only involves visiting websites available to the general public, interacting with their cookie warning dialogues, and finally investigating the files stored on the researcher's machine. This study's presence on the Web is limited to the most popular domains only and their main pages. Such a limited browsing pattern, considered as standard user behavior, has no possibility of collecting any personally identifiable information and presents no harm to the websites' publishers. What is more, to collect the data, we only visit a set of carefully chosen websites, therefore we do not perform any sort of web crawling or indexing. Hence, in this research, we do not adhere to the specifications of `robots.txt` files.

[3] By "policy towards cookies" we mean the way the users interacted with GDPR Cookie Privacy Warnings. The aforementioned studies did not document which users opted out, and which decided to opt in.

Chapter 3

Design and Implementation of Research Platform

One of the main contributions of this research is the design and implementation of a platform capable of simulating human-like behavior and measuring how interaction with Cookie Privacy Warnings impacts the network of publishers (first-party domains) and trackers (third-party domains collecting information on user's activity). The basis for detecting user tracking activities, such as assigning a pseudonymous ID or exchanging personally identifiable data, were cookies set on the user's browser. Hence the platform has been named **Cookie Crumble Tracer**, or CCT for short. CCT is primarily a web crawler that visits a set of predefined domains and interacts with Cookie Privacy Warnings, if present, to manipulate both the cookies set on the client's browser and the HTTP requests that the browser makes.

This chapter first describes the decision-making process and lays out the requirements for the research platform. The focus then shifts to the implementation, system structure, and the technology stack used while creating CCT. Finally, the platform is evaluated with respect to its initial requirements.

3.1 Cookie Crumble Tracer Design

3.1.1 Simulating Real User-Browser Interaction

To reliably collect relevant data, CCT had to simulate a human-like web browsing experience. This requirement's importance cannot be underestimated. Some browser automating tools used for privacy-related studies (e.g., [4]) use stripped-down browsers, such as PhantomJS [13], hence compromising fidelity for speed [14].

The decision to make such a compromise has an important consequence - the webserver to which a request is made might detect strange, nonhuman browser activity and respond with content modified relative to what a normal user would receive. Englehardt and Narayanan [14] have previously tested the importance of using a complete browser in privacy-related research requiring human-like activity. They com-

pared PhantomJS, the aforementioned headless browser for automating website activity, with OpenWPM - a tool reproducing human browsing experience using a full-fledged browser. According to this study, PhantomJS loads around 30% fewer HTML files and about 50% fewer resources with plain text. Particularly interesting for our study is the fact that many sites don't serve ads to PhantomJS. This hints that using a headless browser not only affects cookies set on the client's browser but greatly limits the number of third-parties embedded on visited websites. This makes headless browsers completely unusable for our study, forcing us to create a research platform with a standard browser at its core.

3.1.2 Fault Tolerance

Cookie Privacy Warnings use a wide range of technologies, thus forcing the behavior of these dialogues to be different across the web. For example, some domains embed their warnings within `HTML iframe` elements, so that their creation can be outsourced to specialized providers. Moreover, the HTML code corresponding to the banners is often loaded asynchronously from the website's main content, which in practice means that a user (and hence CCT) has to wait for a second or two before it shows up.

The inconsistency of solutions present in the domains visited by CCT makes runtime exceptions difficult, if not impossible, to prevent. A robust system has to be capable of catching these runtime exceptions and handling them to ensure consistency of collected data. Fault tolerance means that even if a failure occurs, the entire process of collecting data relevant to this study does not have to be started from scratch, and that edge cases are taken care of. In practice, whenever a failure occurs, the system should be able to resume working from some not too distant checkpoint, using data written to the disk at regular intervals.

3.1.3 Locating Cookie Privacy Warnings - Recall over Precision

Key functionality of CCT is the ability to demonstrate a consistent cookie policy - either opting in or opting out - by interacting with the Cookie Privacy Warnings. However, to accurately define which buttons are responsible for opting in or out, HTML elements corresponding to the privacy warnings have to be first located.

Interacting with the cookie banners influences the cookies set on the client browser and third-party activity. Hence, it is the interaction with these banners which makes the datasets of collected cookies for opt-in and opt-out policies differ from one another. Thus, when it comes to locating the banners, the system design prioritizes recall over precision. With low recall and high precision, CCT would locate and interact with a tiny portion of actual cookie warnings, hence making the collected datasets similar for each cookie policy. With low precision yet high recall, the system would locate and interact with a high portion of cookie warnings, but it would additionally interact with HTML elements which are false positives. However, the latter case does not negatively impact the collected data, as interaction with an HTML element which is a false positive would not affect the cookies set on the user browser. Ideally, the system would have both high recall (so that the datasets for both cookie policies are maximally

different) and high precision (for efficiency, so that no resources are wasted for interaction with elements that are not cookie banners). However, from these two measures, only a compromise in recall would harm the final data, therefore high recall in locating Cookie Privacy Warnings is a priority.

3.2 Cookie Crumble Tracer Implementation

3.2.1 Automating Web Browser

As outlined by Englehardt and Narayanan, there exist numerous platforms for automating browsing activity for privacy-related research [14]. Such systems include FPDetective, which uses a hybrid PhantomJS and Chromium-based automation infrastructure [15] to detect and analyze browser fingerprinting. Another popular privacy measurement platform, OpenWPM, enables a wider spectrum of privacy measurements, for example, cookie respawning. Previously used privacy measurement platforms usually focus on detecting and analyzing specific tracing technologies, without any actual interaction with the visited domains. This deems these tools unsuitable for our study, as interacting with DOM elements corresponding to the Cookie Privacy Warnings on visited websites is at the very core of this work.

As explained in Section 3.1.1, one of the most important requirements for CCT is the ability to simulate a human-like browsing experience. Selenium, a lightweight framework for performing functional tests of web applications, is a perfect match for that requirement. Selenium supports the automation of popular browsers, such as Chrome or Firefox, by using a WebDriver. WebDriver is a protocol with an exposed API, providing a language-independent interface for controlling web browsers, just like a human actor would [16]. Selenium allows for easy interaction with and manipulation of the Document Object Model elements of visited domains through the provided API, making it a perfect tool for this research. Selenium is not a headless browser, making it significantly more difficult for a server to detect automated browsing.

According to the documentation, Selenium supports the automation of all major web browsers, each of which is backed by a dedicated WebDriver implementation. A natural choice was to use the platform with the largest browser market share worldwide. With around 64% [17], Chrome was an obvious choice. Although focusing on using one specific user agent might be seen as a limitation of CCT, using Chrome allows maximizing the user group whose browsing experience is reproduced by CCT. Another important reason for using Chrome as CCT's user agent was Chrome DevTools Protocol - API for instrumenting and interacting with the browser. Chrome DevTools Protocol commands can be executed directly by Selenium WebDriver, thus providing a reliable way of extracting first and third-party cookies from the browser.

3.2.2 Analysed domains

Another important decision concerned the source of websites which CCT will crawl and on which user tracking data will be collected. The most widely used list of popular websites, utilized by studies such as [4], [14], [11] or [5], is the one created by a

commercial publisher Alexa, an Amazon company. An obvious disadvantage of the Alexa Top One Million list is its lack of stability - since 30 January 2018, almost 50% of this list's entries change every single day. What is more, according to the study carried out by Pochat et al. [18], lists such as Alexa Top One Million or Cisco Umbrella can be easily manipulated, which in turn makes them an unreliable source for data on which privacy-related research is based, as the measurements are difficult to repeat. In their study Pochat et al. present a new website popularity ranking called Tranco, a "research-oriented top sites ranking hardened against manipulation" [18]. Tranco revises and improves four other popularity rankings, considered to be the most important ones - Alexa, Cisco Umbrella, Quantcast, and Majestic.

Although Tranco provides a list of the top one million websites ranked by popularity, only the top one thousand domains are considered in this study. It is believed that collecting data from this subset of domains will give results representative of the real tracking ecosystem that a European user present on the Internet is surrounded by. Moreover, according to [3], the cumulative number of trackers observed while crawling top domains is not linear to the number of domains visited, i.e. most distinct trackers will be embedded on a small subset of websites. Limiting the number of crawled domains to 1000 should therefore not significantly decrease the number of observed trackers.

3.2.3 Visiting Domains and DOM Interaction

The technology pool of the modern web is vast, thus making errors difficult to prevent. AJAX, or Asynchronous JavaScript and XML, has proved to be one of these error-incurring technologies. AJAX allows for asynchronous requests and data retrieval from the server. In practice, AJAX decouples data interchange from the presentation layer, allowing web pages to change their content without reloading the entire page. Loading of Cookie Privacy Warnings is often achieved using AJAX, allowing the web publishers to load them asynchronously on their domains [1].

The existence of AJAX means that it cannot be determined whether or not a given domain contains a Cookie Privacy Warning immediately after Selenium finishes loading that website's DOM. If we did so, asynchronously loaded cookie warnings could be excluded from the DOM elements that our platform analyzes because they have not been loaded yet. Hence, the number of domains classified as false negatives would increase which as described in Section 3.1.3 is not acceptable.

Asking Selenium WebDriver to retrieve an element which is not a part of DOM throws `NoSuchElementException` (or `StaleElementReferenceException` if that element is no longer a part of DOM). If not guarded against, such runtime exceptions crash the platform. To safeguard against AJAX and any asynchronously loaded content, we use a simple heuristic approach - waiting, even if Selenium seems to finish loading the domain. For this purpose Selenium provides *"explicit waits"* - code structures one defines to wait for a certain condition to occur before moving on in the code. To achieve that, Selenium provides `WebDriverWait` and `ExpectedCondition`. Exemplary code

[1] This is often a byproduct of outsourcing the creation of these banners to other companies.

structure guarding against asynchronously loaded content is presented in Figure 3.1.

```
try:
    cookie_banner = WebDriverWait(driver, 10, ignored_exceptions=(
        NoSuchElementException, StaleElementReferenceException)).until(
        EC.presence_of_element_located((By.CSS_SELECTOR, domains_to_selectors[domain]))
    )
except TimeoutException as e:
    return clickables
```

Figure 3.1: Code snippet guarding against asynchronously loaded content, taken from the implementation of CCT. Selenium WebDriver waits until an element is located by given CSS selector, with timeout set to 10 seconds.

3.2.4 Ensuring Fault Tolerance with Incremental Code Execution

As mentioned in Section 3.1.2, CCT has to be robust and fault-tolerant. The platform must ensure consistency of behavior and collect data even when facing technologies unaccounted for. To achieve this goal, several mechanisms have been put in place. First one, and the most obvious, is catching runtime exceptions, such as the aforementioned `NoSuchElementException` or `TimeoutException`, using language-specific exception handling blocks. However, a more important measure aimed at ensuring robustness and fault tolerance is the platform's architecture.

CCT makes multiple crawls of analyzed domains to perform different sets of actions. In the very beginning, the platform was being developed as a single script. Its initial design envisioned CCT crawling a list of websites, performing a set of activities on each of them - locating privacy warnings and clickable elements responsible for accepting or rejecting cookies, then interacting with them, collecting the data saved by the server on the client's side, and finally parsing the data. However, as the development progressed, it became apparent that the system is too complex to run as one complete, monolithic process. Developing CCT in such a way would require exceptional, perfect error handling - otherwise, one runtime exception would discard all previous code execution and collected data.

Thus, the system was split into core functionalities, allowing to break the CCT's execution into several rounds. After each one, data collected by the CCT is written to the disk, so that it can be used by the subsequent round. Thus, CCT works in an incremental manner, increasing the robustness of the system. Moreover, it has been deemed necessary to make the data human-readable, so that the intermediate results can be analyzed by an experienced human. JavaScript Object Notation, or JSON, has been used as the format of the data read from and written to the disk.

3.2.5 Round 1: Locating Cookie Privacy Warnings

The purpose of this stage was to locate Cookie Privacy Warnings on each domain that contains one. Locating an element means finding a CSS selector uniquely identifying it within the DOM. CSS selector is a pattern that selects elements within DOM one

wants to style. Here's an example of a CSS selector identifying a privacy warning on `facebook.com`:

$$\#yDmH0d > c - wiz > div.T4LgNb.eejsDc > div > div > div$$

As mentioned in Section 3.1.3, although both high precision and high recall are preferred, high recall is prioritized in locating Cookie Privacy Warnings. Ideally, CCT would find all cookie warnings existing in the set of visited domains without the need for human supervision. However, bearing in mind that such dialogues are implemented in different ways across the web, creating a fully automated tool capable of collecting data with adequate recall seemed to be infeasible for this research. Thus it has been decided to create a scraping tool automatically collecting data, which would later be reviewed and refined by a human actor. Excluding the possibilities of creating and training a machine learning model, this human-computer cooperation has been deemed an optimal solution of extracting cookie warnings with adequate recall and limited human involvement.

The initial idea of collecting cookie warnings envisioned a heuristic approach - searching the DOM for HTML elements containing phrases and wordings often occurring in cookie dialogues. To do that, one could use the findings presented by Molnar [19], who analyzes privacy warnings' inner text to find the most important unigrams and bigrams. However, this study analyzes the phrases that are already within cookie dialogues, hence it does not guarantee that n-grams which it reports as important are unique (within the entire DOM) to a cookie banner. Since phrases such as "policy, "cookies" or "privacy", can be located outside privacy warnings, using such a simple heuristic methodology would result in inaccurate measurements. Such an approach makes it also difficult to determine whether it is the extracted element, or one of its ancestors, that corresponds to the entire banner, not just its sub-part.

Cookie Privacy Warnings can be located, with both higher recall and precision, using a predefined CSS selectors list. Such lists are used to remove unwanted content, including annoying advertisements or (for privacy negligent users) bothersome cookie warnings. *EasyList Cookie List* is one of such lists, containing a set of rules blocking cookies banners, GDPR overlay windows, and other privacy-related notices [20]. *EasyList Cookie List* has been tested with popular ad-blocking tools such as AdBlock, Adblock Plus, and uBlock Origin, hence making it a credible filter list. As it has been created with the vision to be used with these ad-blocking tools, this list contains an extensive set of CSS Selectors. Being a community project, it is constantly updated, making it a good starting point for locating Cookie Privacy Warnings.

Parsing CSS Selectors List

EasyList Cookie List is vast and contains, in addition to CSS Selectors pointing to cookie banners, scripts to prevent JavaScript from loading cookie notices or to block certain requests. This additional content has been included to adapt the list to ad blocking tools. As it is unnecessary for this study, *EasyList* has to be first parsed to obtain two filtering rules we are interested in:

1. General Element Hiding Rules - if an element with this CSS Selector is found on any website, it is likely a cookie banner. Put in a set.

2. Specific Element Hiding Rules - if an element is found on a domain to which this rule corresponds, it is a cookie banner. Put in a hash map with domains as keys and selectors as values.

Locating and Double Checking Cookie Privacy Warnings

To detect the cookie banners, CCT iterates over the Tranco list of the top 1000 websites. For each website, we make a lookup in the hash map to see if a domain-specific selector exists for that domain. If it does, we make a call to Selenium WebDriver API and screenshot the corresponding DOM element. If not, we iterate over the set of general selectors to see if any of them matches any element of the current website's DOM and take a screenshot if it does. Websites with no cookie warning detected, or websites that failed to load, are put in a set, stored as JSON on the disk so that a human actor could manually ensure that these domains have no cookie banners.

At this point, CCT has made a full iteration over the top Tranco websites and extracted those websites which contain a Cookie Privacy Warning matched by a CSS selector from the list. Although CCT prioritizes recall over precision, high precision is still desired for system efficiency. Moreover, precision is important from the view of reporting statistical data, e.g. how prevalent cookie warnings are. For these reasons, CCT includes a support tool which at this point iterates over the set of websites for which a cookie banner has been potentially identified. For each website, the support tool displays the element which has been screenshot using its Graphical User Interface. A human moderator can then either click on the screenshot or signal that the screenshot is invalid. Clicking on a screenshot saves the corresponding CSS selector as the correct identifier of Cookie Privacy Warning on that website while clicking on the "NOT A COOKIE PRIVACY WARNING" button adds the website to the list of domains with potentially no cookie dialogue.

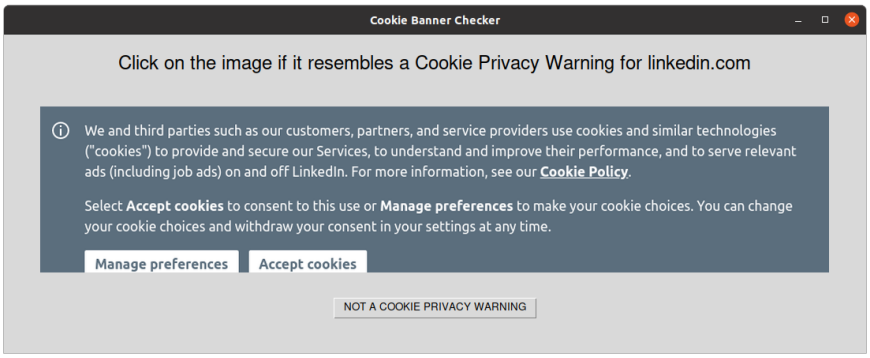

Figure 3.2: GUI of Cookie Privacy Warning Checker

The described support tools make sure that the elements we have retrieved so far are in fact Cookie Privacy Warnings, leaving us with a set of websites for which potentially no warning exists. This list contains websites which:

(a) failed to load

(b) had no DOM element matching any specific or general CSS selectors

(c) had an element matching a CSS selector, but the element turned out not to be a Cookie Privacy Warning

(d) had a Privacy Warning, but the corresponding HTML element was within an `iframe`

The last possibility is due to the nature of HTML `iframe` tag. An `iframe`, or inline frame, represents a nested browsing context, embedding another HTML document within the current one. As a separate browsing context, `iframe` tags have their own DOM. Thus, elements within an `iframe` cannot be accessed by the DOM of the top-most browsing context [2]. This characteristic means that if a website had a Privacy Warning embedded inside an inline frame, Selenium WebDriver would not be able to locate it, even with a correct CSS selector.

Consequently, to ensure maximum recall, human interaction was needed. Otherwise, we would have to come up with a machine learning model capable of detecting HTML elements representing a Cookie Privacy Warning without human supervision, which can be considered as a possible improvement of CCT. Accordingly, another support tool has been created. This tool iterated over the list of websites for which no privacy warning has been identified so far, loading these domains using Selenium. A human moderator would then verify that no banner exists on that domain or use Chrome Developer Tools to extract a CSS selector uniquely identifying the Cookie Privacy Warning within the DOM.

To account for Privacy Warnings within inline frames, the moderator was also asked whether the element was inside such an HTML tag. With a CSS selector of the warning, this supplementary information regarding inline frames allowed switching to the browsing context of specific `iframe` and interacting with the cookie dialogue.

This simple support tool allowed for fast correction of the initial Cookie Privacy Warnings search, as there was no need for the human to manually copy-paste the domain address. Manual correction also allowed for achieving both maximum recall and precision. Figure 3.3 presents the refinement of Cookie Privacy Warning search on one of the websites using the aforementioned support tool.

3.2.6 Round 2: Locating Clickable Elements

At the end of the first phase, a mapping from domains to CSS selectors identifying Cookie Privacy Warnings was written to the disk. This file was the starting point for the subsequent step of CCT aiming at locating clickable elements, within the Cookie Privacy Warnings, responsible for interacting with the banner and accepting cookies in accordance with two policies: opt-in or opt-out.

Opt In: given a choice, the user agrees to the server setting all non-mandatory cookies.

Opt Out: given a choice, the user disagrees with the server setting any non-mandatory cookies, or leaves the website if no option to disagree is initially presented.

[2] From the user perspective, the top most browsing context is the window.

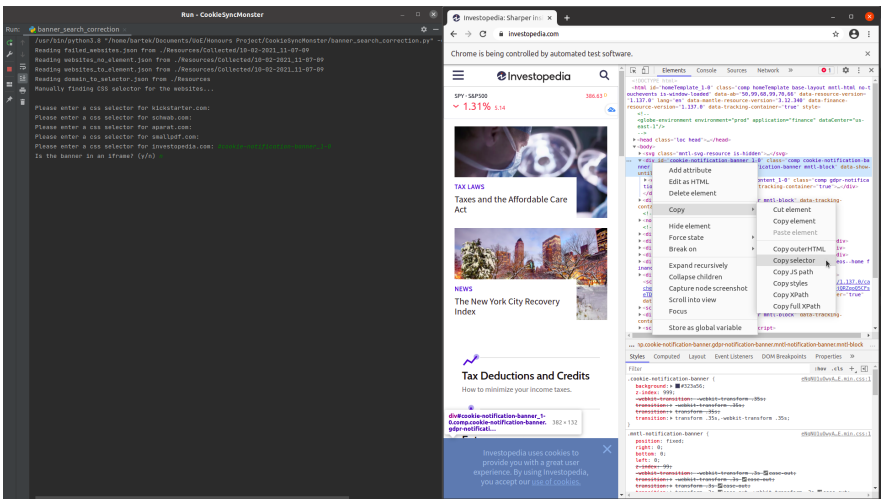

Figure 3.3: Manual correction of Cookie Privacy Warning search

To identify clickable elements on particular domain, CCT would look for any descendant tags of the HTML element of a Cookie Privacy Warning, falling into one of these categories:

- button tag `<button>`
- tag given a button role by including attribute `role="button"`
- anchor tag `<a>`
- input tag submitting a form to the server `<input type="submit">`

Selenium WebDriver API allows for extracting DOM elements in multiple ways, including XPath which uses path expressions to select nodes within an XML, or in the case of DOM, HTML document. XPath proved to be an ideal tool for extracting descendant tags because of XPath Axes. An axis represents a relationship to the current node and is used to identify nodes within some context. One of XPath Axes is called ”descendant” and selects all descendant tags of the current node. XPath expressions for selecting clickable elements within privacy warnings are presented below:

Element	XPath
`<button>`	`descendant::button`
`role="button"`	`descendant::*[@role='button']`
`<a>`	`descendant::a`
`<input type="submit">`	`descendant::input[@type='submit']`

Table 3.1: XPath expressions identifying clickable elements within Privacy Warnings

Once the clickable elements have been located within privacy warnings, CCT had to find the ones responsible for opting in and opting out. To do this, CCT utilizes a heuristic approach basing on the findings of Molnar [19]. In this paper, a comprehensive study of phrases used not only within cookie banners but also specifically in clickable elements responsible for opting in and out has been presented. CCT uses these phrases

to heuristically determine the function of a given clickable. CCT maintains dictionaries (in form of sets) of phrases commonly used in clickable items of cookie warnings, sorted by the functionality they describe. There is one dictionary with phrases for opting in, one for opting out, and one indicating a custom cookie selection. Using these data, a heuristic algorithm checks if a phrase from a dictionary is within the inner text of a clickable element, thus classifying it as opt-in, opt-out, or custom.

To improve this heuristic approach, some modifications have been made further on in CCT's development. To account for domains with non-English content, whenever comparing the text of clickable elements to the phrases in the dictionaries, that text was first translated to English using Google Translate API. That text has been also normalized by making all letters lowercase. Hence a clickable element with the text ”estar de acuerdo” simply becomes ”agree” and is therefore matched by CCT to opt-in policy.

At this point an important clarification has to be made: most of the domains do not enable the user to reject all non-mandatory cookies by providing a single ”reject all” button. Instead, as presented in Figure 3.4, an option to manage cookie preferences is usually provided.

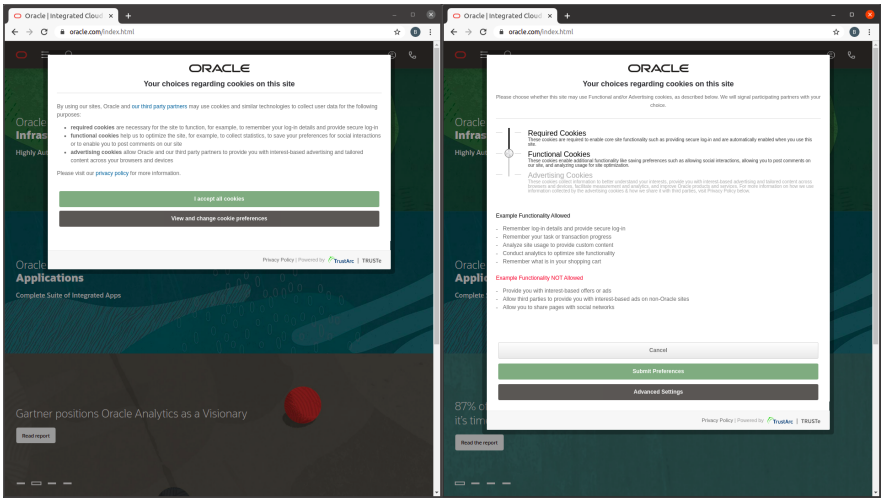

Figure 3.4: Exemplary Manual Management of Cookie Preferences

This makes it incredibly difficult for CCT to interact with a cookie warning in a way consistent with a particular cookie policy. One of the reasons is the banners' front-end. The example above uses a slider to select cookie preferences. Other websites might use a form or any other input method. Moreover, an automated tool aiming at successfully selecting cookie preferences would require a way of understanding the semantics of the text within the form it is interacting with so that it knows what it agrees or disagrees with. To limit the system complexity, CCT uses a simplified approach. When presented with no option to immediately reject all non-mandatory cookies, the opt-out policy states that CCT simply leaves the website, thus imitating a user too lazy to read and select cookie settings. This ”lazy” user behavior, however, is not far from the truth and can be seen as the correct browsing pattern exhibited by a standard user. According to Hu et al. [21], users do not generally make use of the privacy-increasing

choices offered by GDPR-compliant Cookie Privacy Warnings.

3.2.7 Round 3: Collecting Data on the Tracking Environment

CCT's final phase aims at collecting the data exposing the structure of the online tracking ecosystem and how a cookie policy can change it.

Data Collected by CCT

CCT's paramount goal is to conduct a study on the online tracking ecosystem and how it is impacted by the user's cookie policy. For collecting the data, Selenium WebDriver was used to crawl the domains while controlling the browser's interaction with Cookie Privacy Warnings, thus affecting the cookies set on the client's side. While crawling the websites, CCT saved all potentially tracking, id-like cookies, and all HTTP requests made by the user agent when visiting each website. Each crawl visited only the main (starting) page of each domain. CCT was deployed on a personal computer connected by a VPN to the University of Edinburgh network. Response time for each domain was set to 60 seconds. The collected data was saved in JSON format for subsequent utilization in network graph creation.

For the purpose of this study, CCT defines two important entities:

- *Publishers* - the domains explicitly visited by a user, a.k.a. first-parties
- *Trackers* - third-parties embedded within publishers and responsible for setting potential tracking cookies in the browser

To collect the desired information, CCT performs six crawls of the Tranco top one thousand websites. The first two crawls are aimed at identifying trackers embedded within publishers' domains - one crawl for opt-in policy, another one for opt-out. These crawls logged id-like cookies set by third-party domains. CCT extracted the cookies from the browser only after interacting, if possible, with the Cookie Privacy Warning using the clickable elements from Round 2. Before visiting each domain, using Selenium WebDriver API, the browser was cleaned of any cookies or cache, hence making the visits to new domains independent from the preceding ones [3].

The remaining four crawls aimed at identifying pairs of trackers performing cookie synchronization. Collecting data for each policy required two crawls - one for visiting the domains, allowing the cookies to be set on the browser, and another one for capturing HTTP traffic made by the browser [4]. Thus visits to the domains were not isolated like in the first two crawls. The reason for that was simple - cookie synchronization will not happen unless the cookies are set on the client's side. By making a separate crawl to first populate the browser with cookies, CCT exploits cookies' cross-website tracking capabilities and allows id-sharing to all trackers that want to do so. Hence the amount of cookie synchronization observed is maximized.

[3] This action was necessary, as the cookies saved while visiting previous domains would interfere with new measurements. The identification of third-parties embedded in a domain must be done only by using the cookies set on visiting that particular domain.

[4] With two cookie policies, opt-in and opt-out, and two crawls per policy, 4 crawls were needed in total.

Filtering Non-tracing Cookies

CCT's must be able to filter out these cookies which non-tracking cookies. There are many ways in which a cookie can trace a user, for example, several cookies with little information can be combined to track a user. Theoretically, a user could be traced even by a cookie name. This study, however, takes a conservative approach to avoid tracing overestimation and takes into account only the cookies' `value` fields.

To determine if a cookie is an identifier or not, some previous studies, such as one by Sanchez-Rola et al. [22], take an information-theoretical approach of measuring cookies' entropy. [22] takes inspiration from studies focusing on measuring password strength, defined as the amount of information it carries, by calculating the number of guesses it would take an adversary to guess it. To quantify the strength, `zxcvbn` [23] score is used - large `zxcvbn` values indicate difficult to guess strings.

Using the above score as a sole indicator of a tracking cookie might be partially unjustified. According to Englehardt et al. [24], a tracking cookie must be long-lived, so that it can identify a user over a long period. A short-lived cookie indicates a functional cookie, unattractive from the privacy perspective. CCT partially adopts the approach taken in [24] and analyses both `expires` and `value` fields of the cookie.

First, CCT parses the string within the cookie's `value` field, which is often composed of multiple name=value pairs, as shown on the example below:

```
name1=value1$name2=value2$...$name3=value3
```

`$` is a delimiter character matching a regular expression `[^a-zA-Z0-9_=-]`. After parsing cookie data into separate pairs, the cookie was classified as tracing if any of these sub-values had a length larger than 7 characters and `zxcvbn` score $\geq 10^9$, which are the same threshold values used by [22]. Moreover, just like [24], we require a tracing cookie to have an expiration date of at least 3 months away from its creation.

Detecting Cookie Synchronization

In the past, different studies have taken multiple approaches to detect cookie synchronization. Initially, only heuristic-based mechanisms were used, in which cookie synchronization is discovered whenever an id-like cookie delivered by one domain is sent in an HTTP request to a different one. Such approach has been taken by numerous studies, such as [2], [25] or [12].

As reported by Bashir et al. [26], companies such as DoubleClick have recently begun encrypting or their cookies before sharing them to other ad networks, to prevent any potential adversary from learning the cookie by packet sniffing. For this reason, studies such as [27] have started using machine learning, stateless cookie synchronization detection mechanisms. Such an approach envisions using only characteristics of an HTTP connection to detect synchronization and hence does not rely on any previously stored cookies.

Unlike [27], CCT uses a purely heuristic approach. As cookie synchronization is just a request to a third-party, carrying at least one tracking cookie set by a referring website, heuristic detection requires analyzing both HTTP traffic made by the user agent and

the cookies set on the browser. Sometimes, id sharing detection can be straightforward, as we could simply search for parameters named suggestively (e.g., ”uuid”, ”user_id”). However, generalizing this approach would not be reliable, as different domains use potentially unstandardized parameter naming, possibly resulting in increasing the number of false negatives. The following paragraph discusses the algorithm designed for detecting cookie synchronization.

First, non id-like cookies set on the browser are filtered out, leaving us with only potentially tracing ones, which are put in a hash map with cookie value as key and domain (which set the cookie) as value. Then a dataset of HTTP requests is parsed to detect id-like strings contained in:

- GET parameters within the URL
- URL path
- Cookie header
- Request body

Each detected id-like string is then looked up within the aforementioned hash map. If there is a map entry with the same cookie value and the domains are different (receiver of the cookie is different from the domain setting it), cookie synchronization is detected.

Simple string matching, however, cannot be utilized to check whether two domains are the same or not. Such an approach would not account for cases of different subdomains owned by the same company (e.g., store.steampowered.com and steampowered.com). Hence, CCT uses `DNS whois` protocol to obtain information about the domain provider and discriminate between intentional cookie synchronization and legitimate internal information sharing.

3.3 Cookie Crumble Tracer Evaluation

3.3.1 System Stability

As reasoned in Section 3.2.4, CCT has been implemented in such a way so that it performs regular saves to the disk. In case of a crash, it can be rerun from a checkpoint. In the process of gathering required data, CCT has executed each of the described rounds at least once on the set of 1000 most popular domains from the Tranco list. CCT has failed only twice, once during the execution of Round 1 and once during Round 2, due to unstable WiFi connection which caused almost all domains to timeout. Even in these cases CCT has not technically crashed, but simply delivered meaningless, mostly empty results. After switching to a more reliable Ethernet connection, CCT managed to repeat these rounds without any errors. Taking into account all domain visits made by CCT in all three rounds, with only two failures, restarting CCT was necessary on average once every 4000 domain visits.

3.3.2 Loading Web Pages

Across all three rounds, CCT fully analyzed 88.1% (881 out of 1000) Tranco websites, meaning that 11.9% of all websites have failed to load. Round 3, which aimed at collecting data, was responsible for most of the domains not loading properly - 7.9% (79) websites failed to load while analyzing network traffic. This could have been caused by selenium using a proxy server to analyze the traffic, slowing down the network throughput and hence increasing the number of pages that timed out. A possible improvement could be simply increasing the time tolerance of the system. Another 2.7% (27) of all domains failed to load during the first round, when cookie warnings were being collected, with another 13% of all domains failing to load during locating clickable elements in the second round.

Timeout errors were most frequent in the third phase of CCT's operation. Another reason behind the domains failing to load was unresponsive servers. Some entries in the Tranco list are not supposed to be visited by human actors. For example, `http://tiktokcdn.com` is used to deliver media content to geographically close clients. However, given the complex nature of performed crawls and the wide spectrum of servers that top entries of the Tranco lists refer to, the overall performance of CCT is more than satisfying.

3.3.3 Identifying Privacy Warnings and Clickable Elements

As described in Section 3.1.3, CCT prioritises recall and low false-negative rate. As outlined in the sections documenting CCT's implementation, this has been achieved by combining resources used by popular ad-blocking tools and human supervision. To assess whether those requirements have been met, 100 domains have been sampled from the Tranco list's top 1000 entries. Out of these 100 websites, 53 contained a Cookie Privacy Warning. In the first phase of Round 1, by utilizing the EasyList CSS selectors list, CCT identified 24 (45%) of these cookie dialogues. By using GUI of CCT to verify these banners we have confirmed that 100% of these initially retrieved HTML elements are privacy warnings. The remaining 29 privacy warnings were located manually by a human moderator, thus achieving both 100% recall and 100% precision on the given domain sample. The process of collecting data by CCT and verifying it (using support tools provided by the platform) by a human actor allows for achieving such results every single time.

Out of these 53 privacy warnings, 9 were simply notifications of cookie use. Another 27 were banners that had only the option to opt in. 13 banners provided both an option to opt in and another option to customize cookie settings. Finally, there were only 4 domains that provided both an option to opt in, and an option to opt out and reject all non-mandatory cookies. Using the heuristic approach of detecting clickable elements within Cookie Privacy Warnings outlined in Section 3.2.6, CCT has correctly classified 95% (42) of clickable items corresponding to opting in and 100% of elements corresponding to opting out, achieving an aggregated recall of 96%.

Chapter 4

Results and Data Analysis

4.1 General Cookie Characteristics

This section investigates the general characteristics of cookies set in the browser and how the nature of these cookies is affected by the user's cookie policy. In addition to presenting cookie data for opt-in and opt-out policies, we also show the characteristics of cookies set on the client's side before any interaction with a Cookie Privacy Warning.

Policy	Cookie	Mean	Std Dev	Median	Mode	1st Quartile	3rd Quartile	Max
No interaction	All	16.8	19.0	11.0	3.0	11.0	10.0	160.0
	First party	9.2	9.3	6.0	0.0	6.0	6.0	73.0
	Third party	7.6	13.1	3.0	0.0	3.0	3.0	128.0
	ID like	7.6	10.5	4.0	1.0	4.0	4.0	88.0
Opt In	All	17.6	20.7	11.0	4.0	12.0	10.5	238.0
	First party	10.0	9.8	7.0	0.0	7.0	6.0	57.0
	Third party	7.6	15.0	2.0	0.0	3.0	2.0	217.0
	ID like	8.0	11.1	4.0	2.0	4.0	4.0	111.0
Opt Out	All	16.8	19.1	11.0	3.0	11.0	10.0	160.0
	First party	9.3	9.4	6.0	0.0	6.0	6.0	73.0
	Third party	7.5	13.2	3.0	0.0	3.0	3.0	128.0
	ID like	7.6	10.6	4.0	1.0	4.0	4.0	88.0

Table 4.1: General Cookie Data

A detailed summary describing the gathered cookies can be found in Table 4.1. What has to be stressed is the fact that in this table, an ID-like cookie does not have to be set by a third-party. However, this is an exception in this study. Generally speaking, in this work the term "ID-like cookie" is used in the context of third-party cookies.

As one would expect, the largest means for all categories can be observed for opt-in cookie policy. With this policy, visiting each website saved on average 17.6 new cookies in the client's browser. This number includes both first and third-party cookies, which can be used for session management, tracing, or personalization. Further refinement of the opt-in policy results shows that on average 10 first-party cookies and 7.6 third-party cookies were saved on each domain, with 8 cookies identified as potentially tracing.

An interesting statistic that has been extracted by the CCT and presented in Table 4.1

is the maximum number of cookies set on visiting a domain. The largest number of cookies set by a domain is 238 for opt-in policy, 78 more than both the number of cookies set before any interaction and the number of cookies set for the opt-out policy (160). When only third-party cookies are considered, the differences between the maximums are even larger - 217 for opt-in compared to 128 for opt-out and no-interaction. These numerical differences decrease when only ID-like, potentially tracing cookies are considered. The maximum number of such cookies is 111 for opt-in policy, 23 more than for the other two criteria (88 for both).

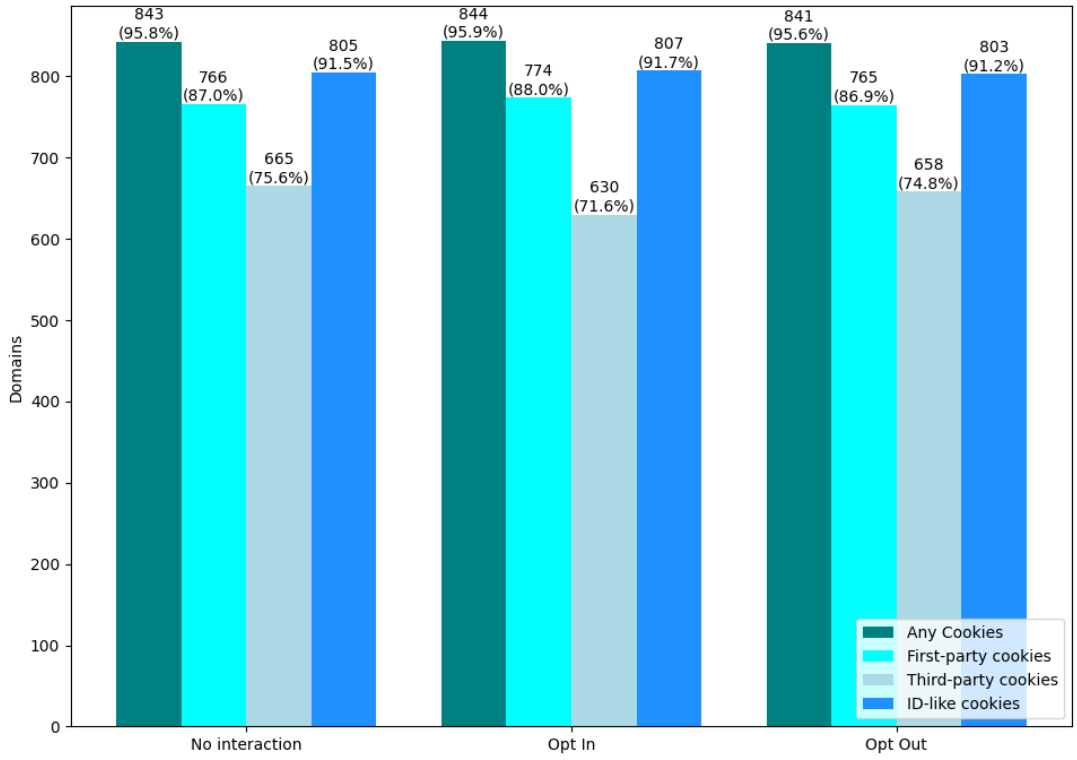

Figure 4.1: Number of unique domains setting particular types of cookies, grouped by the policy of interaction with Cookie Privacy Warning. Percentages represent fractions from all 881 loaded domains. Visiting one domain can result in setting multiple types of cookies. Therefore the number of first-party and third-party cookies are not complementary and do not add up to the ”Any Cookies” number.

Looking at the data presented in Figure 4.1 and Table 4.1, one can note that the extracted statistics are generally quite similar. To depict the distribution of the data together with some of the statistics from Table 4.1 *violin plots* were used. Violin plots show summary statistics similarly to box plots, at the same time combining it with *kernel density estimation plots*. Instead of showing counts of data points falling within order statistics, like in the case of box plots, violin plots utilize Gaussian Kernel Density Estimation to compute an empirical distribution of the sample [28]. Figure 4.2 presents labeled violin plots for cookie data collected by CCT on these websites which contain a GDPR privacy warning. The reason for excluding the websites which do not have such a warning is simple. Cookie Privacy Warnings have been located on around 50% (497) of all websites. This in turns means that 503 domains behave in the same

way irrespective of the cookie policy of the crawl. Including these websites would most likely homogenize the datasets and make the plots very similar to each other. For this reason, Figure 4.2 considers the data from GDPR compliant domains only.

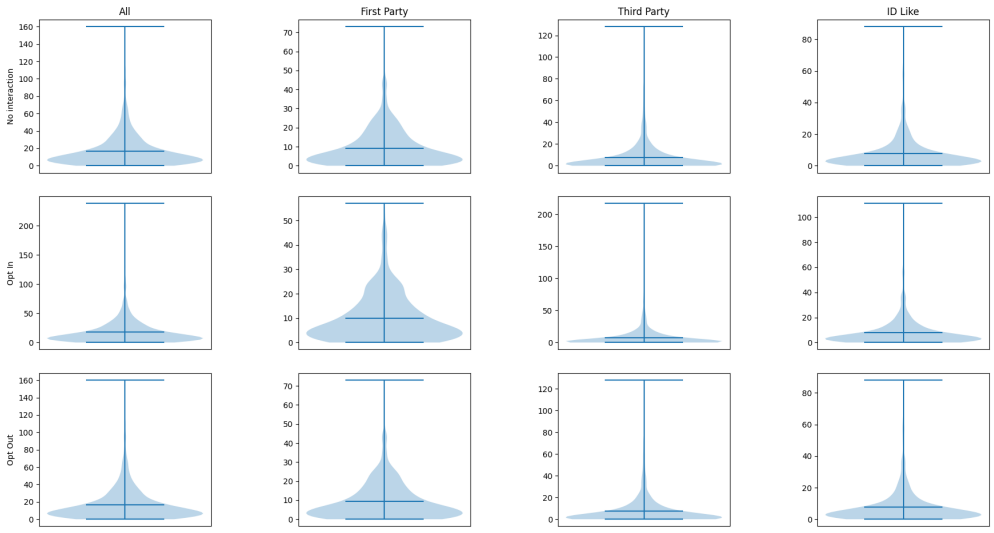

Figure 4.2: Distribution of cookies of specific types over the crawled websites according to the policy of interaction with Cookie Privacy Warning. The distribution is accompanied by horizontal lines specifying the maximum, minimum, and mean.

.

The graphs further confirm what could already be deduced from Table 4.1 - datasets collected for each policy criterion are similar. For this reason, another form of analysis is required. In the next section, the tracing ecosystem will be modeled and analyzed as network graphs.

4.2 Creating Network Graphs

By quantifying the behavior of first and third-party cookies, analysis of numerical results performed in the previous section summarizes the state of the tracking ecosystem that users can develop around themselves. However, it does not investigate the structure of that network, which could shed light on the importance of specific domains within it. By analyzing the inner structure of the tracking environment, one could also answer questions regarding, for example, communities formed by publishers and trackers [1]. Thus, an important part of this study is transforming the data collected by the CCT into network graphs representing the tracking ecosystem. Graph mining tools and metrics are then used to study the properties of these graphs. By doing so, we can dissect the environment of publishers and trackers and observe how mechanisms, such as cookie synchronization, affect the network.

In creating the graphs, this study takes an approach similar to the one by [3], [7] [12]: two network graphs, ”Publishers-Trackers” and ”Trackers-Trackers”, are created. The

[1] Please refer to Section 3.2.7 for the definitions of terms ”publisher” and ”tracker”

following paragraphs describe how the data is transformed into graphs and present the results of their analysis.

Publishers-Trackers (PT) Network Graphs

This graph represents the connection between third parties (trackers) and their hosts (publishers). It is a bipartite, or 2-mode graph, meaning that its vertices can be divided into two disjoint and independent sets, V_P (publishers) and V_T (trackers), such that every edge connects vertices of different modes. In this undirected graph, a publisher node can connect to multiple tracker nodes and vice versa.

PT graph is constructed by creating two sets of mappings, V_P and V_T. V_P represents the set of domains from the Tranco list that CCT visits during the crawl - the publishers. V_T is the set of third parties embedded within the domains included in V_P - the trackers, which set an ID-like cookie in the user's browser. Furthermore, we also introduce E_{PT} - a set of weighted edges connecting nodes of different modes, with weights equal to the number of ID-like cookies set by a tracker hosted by that publisher. Thus, the PT graph can be concisely represented as $PT = (V_P, V_T, E_{PT})$.

To apply graph metrics, network graphs have to be connected in the sense of topological space - there has to be a path between any two nodes of the graph. For this reason, after parsing the collected data into the Publishers-Trackers graph, their largest connected components have been extracted for further analysis. According to an observation made by Solomos et al. [3] the majority of distinct trackers and publishers can be found in the largest connected component of the PT graph. Their study reports that 99% of all publishers and 95% of all trackers can be found in the LCC. However, that study collects data by crawling Alexa top 100 000 websites - significantly more than CCT does in this study. Thus, we would expect the graphs created as part of this study not to achieve that high level of connectivity.

This, indeed, has turned out to be the case, but the differences were not as large as expected. LCC of Publishers-Trackers graph for opt-out policy contains 86.9% (766) of all distinct publishers and 84.3% (404) of all trackers. For opt-in policy, these numbers are 84.7% (746) and 85.1% (660) respectively. What is perhaps even more important, is the fraction of edges of the original PT graphs contained in their largest connected components. These percentages are significantly higher than fractions of publishers and trackers, with 97.1% (3232) and 96.9% (3956) for opt-out and opt-in policies respectively. This observation proves that basing our analysis solely on the largest connected components is justified because it contains almost all of the network's connections and nodes.

The preliminary analysis of PT graphs shows the effect a cookie policy has on the network's size - opting out decreased the number of trackers in the network by 38%, or 39% if only networks' largest components are considered. It also shows another characteristic of the tracking ecosystem - excluding the largest connected component, Publishers-Trackers network graph contains only small, isolated groups of nodes, that include very few publishers hosting very few trackers. This observation is in line with results presented by Solomos et al. [3] who reported identical observations, thus giving credibility to data collected by the CCT.

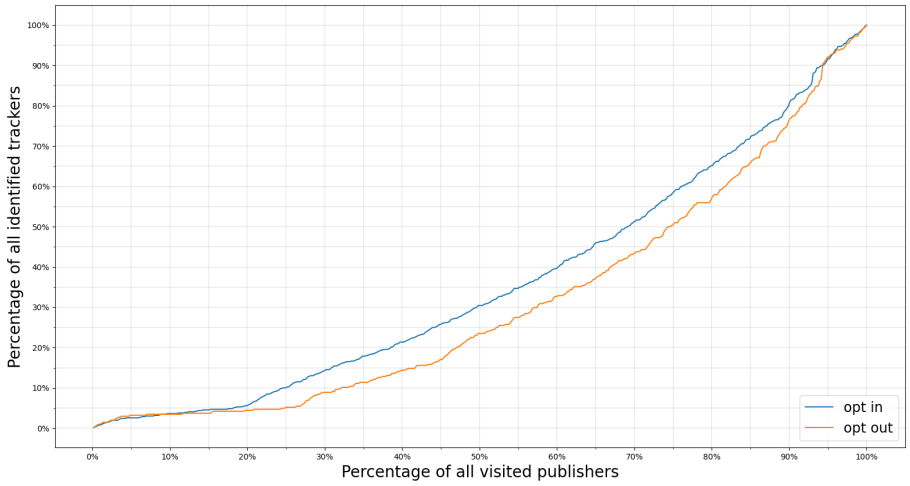

Figure 4.3: The relation between cumulative number of visited publishers, and identified trackers hosted by them, normalized and expressed as percentages

.

In total, while crawling the 1000 most popular domains according to Tranco, CCT has detected 776 distinct trackers when opting in and 479 when opting out. In Section 3.2.2 a hypothesis has been made - we argued that the cumulative number of trackers will not be proportional to the number of publishers a user visits. In practice, it would mean that a minority of publishers will host the majority of identified trackers. By analyzing the vertices of the Publishers-Trackers graph, a plot showing the percentage of observed trackers in relation to the percentage of visited publishers has been created. The horizontal axis represents the fraction of publishers CCT has visited, while the vertical axis presents the percentage of all distinct trackers identified over that fraction. As presented in Figure 5.3, 50% of all trackers are hosted by around 32% of publishers for opt-in policy and 25% for opt-out policy, hence confirming our previous speculation.

Trackers-Trackers (TT) Network Graphs

Unlike PT graphs, Trackers-Trackers Graphs are single modal - they only contain nodes classified as trackers. In a nutshell, TT Graph can be represented as $TT = (V_T, E_{TT})$. V_T is a set of trackers embedded in the publishers, and E_{TT} is a set of undirected, weighted edges connecting the trackers. The weight of an edge is the number of times cookie synchronization has been observed between two particular trackers.

Just like in the case of PT graphs we extract the largest connected component from the TT graphs for further network analysis. TT graphs have a fairly dense structure and are better inter-connected than PT graphs. LCC of Trackers-Trackers network graph contains 98.4% (1195) of all trackers for opt-in and 96.3% (968) for opt-out policy. The fractions of all edges within the LCC are 99.7% (4049) and 99.3% (2618) respectively. One can see the difference a cookie policy makes by looking at the number of vertices and edges within the networks. Opting out decreases the number of trackers in the

graph by 17% (19% if only LCCs are considered). Moreover, opting out results in almost 35% less pairs of trackers sharing data than opting in (2618 vs. 4049). Similar to PT graphs, analysis of remaining connected components showed that each of them contains a single-digit number of non-popular trackers.

The difference in the connectivity of PT and TT graphs sheds light on how collaborative the embedded trackers are, and is somewhat expected. Exchanging user-identifying data allows the trackers to perform server-to-server user data merges, which as shown by Englehardt and Narayanan, takes place at a massive scale [14]. According to [29] and [30], web companies frequently enter mutual agreements for either data exchanges or purchases, so they could improve the quality and increase the number of their user datasets. This, in turn, can be observed by an increased amount of cookie synchronization and, consequently, by a more meshed structure of TT graphs.

While analyzing the Tranco top 1000 domains, CCT has observed 27070 cases of id-sharing when opting in and 16429 cases when opting out, clearly showing the influence of cookie policy on the amount of cookie synchronization. [2]. Similarly to the PT graphs, we expect the relationship between the amount of cookie synchronization observed, and the number of identified collaborating pairs of trackers, to be non-linear, i.e. most of the observed data exchange will be performed by a minority of trackers. To verify this hypothesis, TT graphs' largest connected component has been analyzed to obtain a plot showing the cumulative number of all cookie synchronizations. The horizontal axis of Figure 4.4 presents a fraction of all identified pair of collaborating trackers, while the vertical axis presents the normalized cumulative number of cookie synchronizations (expressed as a percentage) over that fraction.

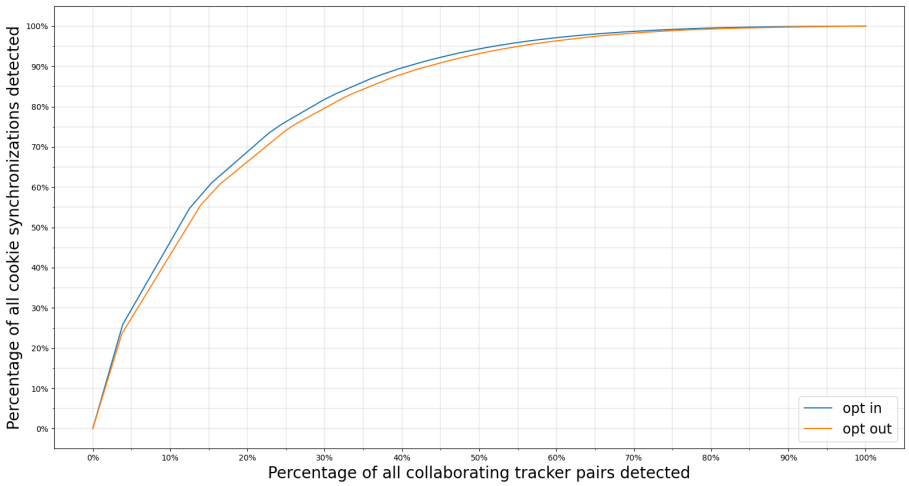

Figure 4.4: The relation between cumulative number of observed pairs of collaborating trackers, and cookie synchronizations performed by them, normalized and expressed as percentages

.

[2]Interestingly, the same reduction of 40% has been previously reported by [12]

As seen in Figure 4.4, the disproportion is even larger for the PT graphs. For opt-in policy, just 11% of all collaborating tracker pairs are responsible for 50% of all data exchange. For opt-out policy, the number is similar, around 12%. This preliminary analysis of the TT graph's largest connected components already shows that some vertices are significantly better connected than others. The following sections utilize graph mining tools and metrics to further study the properties of TT and PT graphs and shed light on the importance of individual vertices within the tracking ecosystem.

4.3 Analysing Publishers-Trackers Graphs

We use graph metrics similar to those utilized by [7], [12]) and [8]. To analyze the inner structure of a Publishers-Trackers graph, we use notions of density, diameter, and radius, as well as global clustering coefficient. To reveal the importance of individual nodes within the network, we compute Degree, Betweenness, and Closeness centralities. Finally, we check how these centrality measures correlate with each other using Pearson Correlation Coefficient.

Policy	Density	Diameter	Radius	Average Clustering Coefficient
Opt In	0.0048	12	6	0.1755
Opt Out	0.0077	11	6	0.2046

Table 4.2: PT graph inner structure's characteristics

Table 4.2 presents how the inner structure of the Publishers-Trackers graph is influenced by the cookie policy. In general, we can observe that these metrics are similar for both of them. Both graphs are sparse, which is indicated by the density and average clustering coefficient. The average clustering coefficient measures the degree to which a graph's nodes tend to cluster together. The low values for both Publisher-Tracker Graphs are a result of its two-modal nature - clustering is limited since edges are present only between vertices of different modes. In conjunction with low density, average clustering coefficients for both graphs indicate their sparsity.

4.3.1 Centrality Metrics

This section computes various centrality metrics to identify the most important vertices within the Publisher-Trackers network. Figure 4.5 plots the cumulative distribution for each metric, computed based on the LCC of Publishers-Trackers graph. [3]

4.3.1.1 Degree Centrality

Degree centrality is defined as the number of ties a node has, or in other words, how many edges are incident upon that node. The first plot in each row of Figure 4.5 presents the cumulative density of normalized degree centrality for both cookie policies. As one could see, in the distribution's upper tail the scores for publishers are an

[3]The presented metrics are normalized, meaning they are divided by the maximum possible value of that metric for given graph.

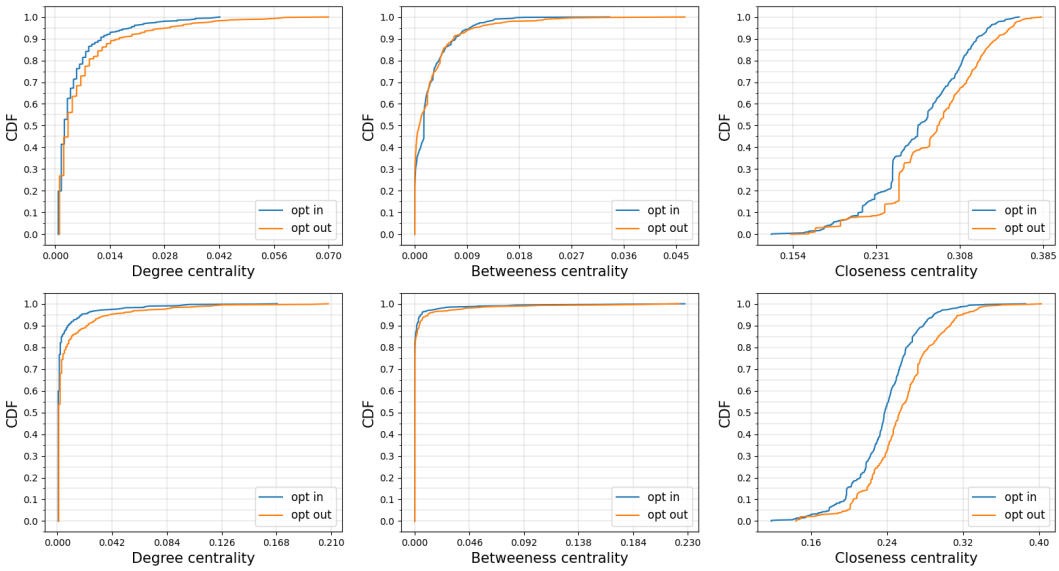

Figure 4.5: Cumulative distribution for each centrality metrics, normalized, as computed for both cookie policies. Top row shows metrics for the Publishers, bottom for Trackers

.

order of magnitude lower than for trackers, suggesting that the most important publishers are far less important than the most important trackers. 50% of publishers have a score ≤ 0.00234 and 0.00328 for opt-in and opt-out policies respectively. Fraction of publisher vertices with a score ≥ 0.01 is 12% (opt-in) and 18% (opt-out), with only 1% of all publishers having a score $\geq 0.035/0.048$.

Although the scores of trackers node at the upper end of the distribution are significantly larger than for the publishers, most of the trackers are at the lower end of the distribution. 90^{th} percentile for tracker nodes is around 0.00858 - for publishers, it is 0.01171. 50% of all trackers have a score ≤ 0.0007 and 0.001 for opt-in and opt-out policies respectively. Top 1% of trackers have a degree centrality $\geq 0.07/0.11$ (opt-in/opt-out), twice the score for publisher's top 1%. The relatively high score for opt-out policy hints at the existence of third-parties with large tracing capabilities, even if a user opts out. What is also interesting is the polarization in trackers degree centrality graph - most of them have a very low score, but there are some for which the score is very large, larger than for any tracker and any publisher. This is due to the presence of domains specialized for user tracking, such as `doubleclick.net` or `facebook.com`, which are often employed by host domains specifically to track their visitors.

As one can see, top positions are occupied by well-known domains, all of which are trackers, showing how important third-parties are in connecting the vertices of the network. Opting out slightly reshuffles the top ten positions, but does not affect the included domains.

4.3.1.2 Betweeness Centrality

For every pair of nodes in a connected graph, there is at least one shortest path between these vertices. The betweenness centrality of a vertex is the number of such shortest

Rank	Opt In			Opt Out		
	Domain	Type	Value	Domain	Type	Value
1	doubleclick.net	Tracker	0.16862	doubleclick.net	Tracker	0.20765
2	facebook.com	Tracker	0.15144	facebook.com	Tracker	0.19454
3	scorecardresearch.com	Tracker	0.10070	scorecardresearch.com	Tracker	0.12678
4	adnxs.com	Tracker	0.09836	adnxs.com	Tracker	0.12131
5	bing.com	Tracker	0.09602	google.com	Tracker	0.11803
6	google.com	Tracker	0.09446	demdex.net	Tracker	0.10710
7	demdex.net	Tracker	0.09133	bing.com	Tracker	0.10273
8	everesttech.net	Tracker	0.07026	everesttech.net	Tracker	0.08743
9	dpm.demdex.net	Tracker	0.06792	linkedin.com	Tracker	0.08743
10	linkedin.com	Tracker	0.06714	dpm.demdex.net	Tracker	0.08415

Table 4.3: Top nodes of Publishers-Trackers graph according to their degree centrality.

paths passing through a particular node [31] and measures the influence of a vertex over the flow of information in a network.

Just like in the case of degree centrality, betweenness scores for the trackers are located mostly around the lower extreme, meaning that trackers can be roughly split into two groups. One contains a small number of well-known, connected domains, central to the graph. The other one consists of the majority of trackers which are neither well-connected nor central to the network. 90^{th} percentile for tracker nodes is 0.00167/0.00387 (opt-in/opt-out) - for publishers nodes it increases to 0.007/0.00623. However, the situation changes for the 99^{th} percentile. The top 1% of trackers have a betweenness score of 0.0538/0.0684, while the top 1% of publishers have a score of $\geq$ 0.0137/0.0222 (opt-in/opt-out). This means that while most of the publishers have scores larger than most of the observed trackers, the most important third-parties are far more central to the tracking environment than all other publishers. This tendency is presented in Table 4.4.

The distributions for trackers and publishers are almost identical for both cookie policies, meaning that the cookie policy does not particularly affect the centrality of individual nodes within the network.

Rank	Opt In			Opt Out		
	Domain	Type	Value	Domain	Type	Value
1	doubleclick.net	Tracker	0.22756	doubleclick.net	Tracker	0.22317
2	facebook.com	Tracker	0.18860	facebook.com	Tracker	0.19388
3	google.com	Tracker	0.15692	google.com	Tracker	0.16595
4	scorecardresearch.com	Tracker	0.10569	scorecardresearch.com	Tracker	0.10473
5	bing.com	Tracker	0.08475	bing.com	Tracker	0.09380
6	demdex.net	Tracker	0.08097	demdex.net	Tracker	0.06842
7	adnxs.com	Tracker	0.07735	adnxs.com	Tracker	0.05694
8	hm.baidu.com	Tracker	0.05381	youtube.com	Tracker	0.05334
9	linkedin.com	Tracker	0.05068	op.gg	Publisher	0.04650
10	youtube.com	Tracker	0.04009	hm.baidu.com	Tracker	0.04265

Table 4.4: Top nodes of Publishers-Trackers graph according to their betweeness centrality

Table 4.4 shows top network vertices for both cookie policies. As one can see, from all the included nodes only one is a publisher. Since betweenness centrality measures the extent to which a vertex lies on paths between other nodes, the publisher's scores are expected to be lower than those of the trackers. Publishers are not connected to each other but are connected through trackers instead. Well-known trackers central to the

network are found at the upper tail of the distribution with scores ≥ 0.02.

Just like in the case of degree centrality, there is little variation in the top 10 rankings between opting in and opting out. This time, however, the policy (slightly) changes the entries within the lower end of the lists.

4.3.1.3 Closeness Centrality

The closeness centrality of a vertex is calculated as the inverse of the sum of the shortest paths' lengths between that node and all other nodes in the graph. The higher this metric is for a vertex, the closer to all other nodes in the network it is. Focusing on the corresponding plots within Figure 4.5, one can see that this distribution is more uniform than for all other metrics and, unlike for the previous one, the range of scores for trackers and publishers is almost the same. The difference, however, is in the curve gradient, which seems to be larger for trackers. This means that most of the trackers have a similar closeness score (0.22-0.28), and only some of them are located at the extremes. The upper tail of the distribution is expected to include well-connected trackers belonging to popular domains, such as Google or Facebook, or other domains which specialize in tracing users for financial profit.

Comparing to the two previous metrics, closeness centrality is more affected by the cookie policy. Opting out increases the closeness of the network vertices, meaning that the nodes are closer to each other than they are when opting in. One of the factors which could influence this centrality metric is the number of trackers in the graph, which is larger for opt-in policy. A larger number of trackers means that in the case of opt-in policy the shortest path between any two nodes must go through extra tracker vertices, decreasing their closeness to other nodes.

Rank	Opt In			Opt Out		
	Domain	Type	Value	Domain	Type	Value
1	doubleclick.net	Tracker	0.38584	doubleclick.net	Tracker	0.40237
2	facebook.com	Tracker	0.36789	facebook.com	Tracker	0.38936
3	newyorker.com	Publisher	0.36289	inquirer.net	Publisher	0.38349
4	inquirer.net	Publisher	0.36024	mirror.co.uk	Publisher	0.3803
5	mediafire.com	Publisher	0.35862	boston.com	Publisher	0.37439
6	tribunnews.com	Publisher	0.35782	colorado.edu	Publisher	0.37439
7	venturebeat.com	Publisher	0.35663	merdeka.com	Publisher	0.36895
8	vimeo.com	Publisher	0.35524	dailymail.co.uk	Publisher	0.36777
9	nfl.com	Publisher	0.35348	tribunnews.com	Publisher	0.36659
10	colorado.edu	Publisher	0.35077	venturebeat.com	Publisher	0.36542

Table 4.5: Top nodes of Publishers-Trackers graph according to their closeness centrality

Unlike in the cases of degree and betweenness centrality, closeness centrality gives importance to the publishers. Although the two most important domains are still `doubleclick.net` and `facebook.com`, the rest of the entries in the ranking are publishers. One could say that relatively high closeness centrality is to be expected from publishers, as by nature the trackers gather around them. A publisher is very likely to be directly connected with a popular tracker, which then connects it directly to a large number of domains. An unpopular tracker, however, cannot - for it to be close to other

nodes, it has to coexists in one host with some other, more popular tracker. This is the only way such an unpopular tracker can be connected to the rest of the network. As we have observed in sections devoted to other centralities, trackers are polarized - a lot of them are significantly less important than just a few well-connected domains like Google. These trackers will connect to the rest of the tracing environment through their host, meaning that the importance of publishers, as measured by closeness centrality, greatly increases.

4.3.2 Correlation of Centrality Metrics

After investigating the inner structure of the bipartite graphs, we focus on evaluating the relationship between these centrality metrics. By calculating Pearson Correlation Coefficient between the Degree Centrality, Betweenness Centrality, and Closeness Centrality for trackers and publishers independently, we will understand if the network vertices tend to have large values in all metrics at the same time, or if there is any dissociation between them.

Policy	Publishers			Trackers		
	DC-BC	DC-CC	BC-CC	DC-BC	DC-CC	BC-CC
Opt In	0.75	0.631	0.538	0.887	0.556	0.383
Opt Out	0.705	0.611	0.469	0.882	0.629	0.418

Table 4.6: Pearson Correlation Coefficient between Degree & Betweeness Centrality (DC-BC), Degree & Closeness Centrality (DC - CC) and Betweeness & Closeness Centrality (BC-CC), for PT graphs corresponding to different cookie policies. All results are statistically significant at $p - value \leq 0.05$

Overall the coefficients show a strong correlation between all metrics with a high confidence level. Particularly high is the correlation between degree and betweenness Centralities for tracking domains, showing that no matter the metric used, these nodes are very important and central to the network. On contrary, publisher nodes have a higher correlation of betweenness and closeness centrality, as well as degree and closeness centrality, thus validating the significance of these publishers, who are connected to many trackers and thus tend to have an important position in the network structure. As one can see, cookie policy does not have a changing effect on the correlations.

4.4 Analysing Trackers-Trackers Graphs

Unlike Publishers-Trackers graphs, Trackers-Trackers network graphs are not bipartite - all nodes are classified as trackers. Nevertheless, the following paragraphs analyze TT graphs in the same manner as PT graphs. We start with simple metrics, presented in Table 4.7. We then carry on to the analysis of degree, betweenness and closeness metrics. Finally, we compute Pearson Correlation Coefficients to see how the metrics correlate.

The values presented in Table 4.7, particularly low average clustering coefficient with low density, show that TT graphs are also sparse. Opting in makes the networks even sparser - this is so because opting in adds new trackers to the network, but not enough

Policy	Density	Diameter	Radius	Average Clustering Coefficient
Opt In	0.0057	9	5	0.1488
Opt Out	0.0055	9	5	0.1076

Table 4.7: PT graph inner structure's characteristics

new connections between vertices to account for the increase in the network's size. TT graphs' diameter and radius are smaller, even though TT graphs have significantly more edges than the Publishers-Trackers graph. This points to tracker nodes forming a more interconnected structure, where vertices are generally slightly closer to each other than nodes of Publishers-Trackers graphs.

4.4.1 Centrality Metrics

Figure 4.6 plots the distributions of normalized degree, betweenness, and closeness centrality for TT graphs corresponding to opt-in and opt-out policies. Basing on this plot, a similar analysis to the one for the Publishers-Trackers graph is performed and presented in the following paragraphs.

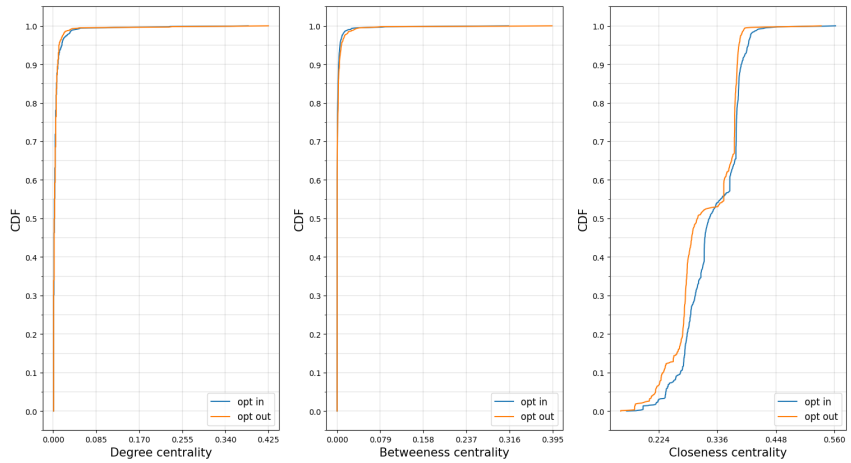

Figure 4.6: Cumulative distribution for each centrality metric, normalized, as computed for both cookie policies

4.4.1.1 Degree Centrality

Unlike in the case of PT graphs, the distributions of degree centralities are almost identical when opting in and opting out. 95% of trackers have scored lower than 0.018 and 0.011 for opt-in and opt-out policies respectively. Top 1% of trackers has normalized degree centrality larger than 0.042 when opting in and 0.033 when opting out. Thus, opting in seems to increase the degree centrality of very few, most important trackers.

When ranking domains according to their degree centrality (Table 4.8), Google's dominance cannot be underestimated, with 4 out of the top 10 domains belonging to that company. Although the values change when opting out, the top 4 entries stay the same. Opting out increases the importance of `facebook.com` and Amazon's `alexametrics.com`.

Rank	Opt In		Opt Out	
	Domain	Value	Domain	Value
1	google-analytics.com	0.38526	google-analytics.com	0.42503
2	doubleclick.net	0.34925	doubleclick.net	0.35471
3	google.com	0.23116	google.com	0.23475
4	google.pl	0.22781	google.pl	0.22958
5	yandex.ru	0.16164	facebook.com	0.12823
6	facebook.com	0.1206	alexametrics.com	0.05171
7	lijit.com	0.10469	ad.gt	0.0486
8	free.fr	0.0536	omtrdc.net	0.03723
9	adsrvr.org	0.05276	chartbeat.net	0.03619
10	techradar.com	0.05109	ups.com	0.03413

Table 4.8: Top nodes of Trackers-Trackers graph according to their normalized degree centrality

4.4.1.2 Betweenness Centrality

Figure 4.6 indicates very similar distributions for opting in and opting out. 95% of nodes have betweenness centrality smaller than 0.00531 when opting in and 0.00776 when opting out. Conversely to the degree centrality, top domains have larger scores for opt-out policy. The difference is also observed in the maximum values - 0.3141 for opt-in policy and 0.39373 for opt-out. Both scores belong to `google-analytics.com`, showing that even opting out does not pose a threat to Google's dominance in the tracing environment. A possible reason for that is the scale of Google's operation - while opting out decreases the tracking activity of less powerful domains, Google's tracking capabilities are large enough to capitalize on other domains' limited abilities, hence explaining its increase in degree centrality scores for opt-out policy.

Rank	Opt In		Opt Out	
	Domain	Value	Domain	Value
1	google-analytics.com	0.31410	google-analytics.com	0.39373
2	doubleclick.net	0.24771	doubleclick.net	0.29365
3	yandex.ru	0.17504	google.com	0.08079
4	lijit.com	0.08808	google.pl	0.07403
5	google.com	0.08226	facebook.com	0.05853
6	google.pl	0.06064	ad.gt	0.04107
7	facebook.com	0.03806	ups.com	0.03751
8	1rx.io	0.03069	goodreads.com	0.03453
9	amazon.com	0.02656	amazon.com	0.03311
10	demdex.net	0.02589	ibm.com	0.03214

Table 4.9: Top nodes of Trackers-Trackers graph according to their normalized degree centrality

Table 4.9 again confirms that top trackers performing cookie synchronization have larger betweenness scores when opting out. Opting out seems to increase the importance of well-known domains, which move up in the ranking for opt-out policy.

4.4.1.3 Closeness centrlity

The distribution of closeness centrality is perhaps the most interesting one. The opt-out curve seems to have a temporary plateau around value 0.336, showing that very few nodes have closeness scores similar to that value. Around 35% of all trackers have a score in the range $0.259 - 0.298$ for opt-out policy and $0.274 - 0.321$ for opt-in policy. Around the value of 0.340, both distributions are similar, with around 40% of all nodes having closeness centrality close to that score.

Rank	Opt In		Opt Out	
	Domain	Value	Domain	Value
1	google-analytics.com	0.56215	google-analytics.com	0.53455
2	doubleclick.net	0.54003	doubleclick.net	0.50948
3	google.com	0.49896	google.com	0.47402
4	yandex.ru	0.45142	google.pl	0.42487
5	google.pl	0.44536	ups.com	0.40025
6	lijit.com	0.43817	newrelic.com	0.39055
7	nfl.com	0.42888	hootsuite.com	0.38820
8	rubinproject.com	0.42612	mgid.com	0.38804
9	pubmatic.com	0.42446	pixnet.com	0.38726
10	demdex.net	0.41909	imgur.com	0.38695

Table 4.10: Top nodes of TT graphs according to their normalized degree centrality

Basing on Table 4.10 and the closeness centrality distribution in Figure 4.6, which for opt-in policy is shifted to the right, trackers tend to be closer to each other when the user opts in. This might be caused by the fact that opting-in enables the trackers to establish a new connection with each other, hence increasing their closeness.

4.4.2 Correlation of Centrality Metrics

To understand the relationship between the centrality metrics for TT graphs and get an idea of how a cookie policy can affect it, we perform correlation analysis, using Pearson Correlation Coefficients, just like in the case of PT graphs. Table 4.11 presents the computed results.

Policy	DC-BC	DC-CC	BC-CC
Opt In	0.925	0.347	0.271
Opt Out	0.935	0.281	0.235

Table 4.11: Pearson Correlation Coefficient between Degree & Betweeness (DC-BC), Degree & Closeness (DC - CC) and Betweeness & Closeness (BC-CC), for TT graphs for different cookie policies. All results are statistically significant at $p - value \leq 0.05$

We find a positive association between the distribution of all metrics. Particularly high is the correlation between degree and betweenness centralities, showing that well-connected trackers are crucial to the information flow in the network. This correlation is the only one that is increased by opting out. The lowest scores are observed for the relationship between betweenness and closeness centrality, hinting at a lack of correlation between these two metrics.

Chapter 5

Future Work

The measurements of privacy-related data, along with related studies, constitute a methodology of exploring the influence of an individual's cookie policy on the tracking ecosystem. As it has been shown, measuring the state of online privacy and identifying tracking activity of the Web is a non-trivial process, forcing us to compromise and make certain simplifications on the way to creating the privacy measurement platform. To paint a more detailed and comprehensive picture of the tracking ecosystem, the following limitations of the methodology presented in this paper should be addressed in future work.

Deeper Crawls. CCT has been designed to visit only the main page of each domain. Such behavior was desired to produce consistent, easily replicable visits that are lightweight in terms of computing power and network usage. However, this is an obvious limitation, as more tracking could have been detected if CCT navigated to the websites' subdomains or interacted with their content. A possible improvement envisions CCT locating menu on the main page of each domain and visiting each subdomain linked in that menu. This approach, however, would significantly increase network usage and computing power, thus increasing the time required by CCT to collect its data. Moreover, although it is desirable to go beyond the main page, such browsing behavior requires careful planning of CCT's online activity. Aspects such as visit duration or interaction with the website's content should be taken into account, as they might potentially affect the tracing activity of the embedded third parties.

Custom Cookie Consent. As mentioned on page 18, CCT has been implemented simplistically - whenever an option to reject all non-mandatory cookies is absent, CCT leaves the domain. Hence, the platform does not interact with custom consent forms. As introducing such a feature would significantly increase the system complexity, it has been deemed as a functionality outside the scope of this study. However, CCT is already capable of detecting buttons within privacy warnings that display custom consent forms. To make CCT capable of filling them in, two improvements have to be introduced. First of all, CCT must be able to locate buttons within consent forms responsible for expressing and submitting user's preferences. This might be done using a heuristic approach similar to the one used for identifying clickable items. After displaying custom consent banner, CCT would look for HTML `form` tag. Using XPath

expressions, it would then look for any descendant `input` tags, using their `id` attribute, CCT would then match individual form inputs with text in the `label` tags. The second improvement is to use NLP/IR model, such as bag-of-words or TF-IDF, to semantically analyze the text corresponding to form inputs, so that CCT knows what it agrees to. Using these two improvements, CCT would be able to film in a standard HTML form, which could be finally submitted by clicking `input` tag with `type="submit"` attribute within the identified form.

Detecting Cookie Synchronization. As mentioned in Section 3.2.7, companies such as DoubleClick have recently begun encrypting or cryptographically hashing their cookies. An improvement that should be therefore introduced into CCT is a cookie-less mechanism of detecting cookie synchronization. To achieve this, one could follow the approach used by [27] and use cookie synchronizations detected by a heuristic algorithm as a ground truth dataset, on which a machine learning algorithm is trained. The first step taken in that direction would be to create a dataset used for training the classifier. This could be done in two ways. The first method envisions adding a feature to CCT that would allow the platform to crawl significantly more than 1000 websites from the Tranco and use its heuristic algorithm to detect these HTTP requests which are used for cookie sharing. Beyond these confirmed events, the dataset would also contain these requests which were initially selected by the CCT as potential id-sharing events, but eventually were rejected as they did not match the cookie already observed by CCT. These requests' headers would be then stored as features in the dataset, with binary labels indicating if a request is a cookie synchronization or not. Another approach is to create a browser plugin using the same heuristic cookie synchronization detection algorithm as the previous solution. This plugin would not be used as part of CCT, and hence would not use Selenium to crawl a predefined set of domains. Instead, it would be distributed among volunteers who would agree to have their HTTP traffic analyzed for potential cookie synchronization. This approach would not analyze the content the volunteers receive, but rather the metadata of the requests made by their browsers. Whichever, method of creating cookie synchronization is used, according to [27] such data with binary labels would detect cookie synchronization with high accuracy, even if the shared IDs are obfuscated. A combination of the heuristic approach used by the CCT and the ML approach would surely increase the recall of cookie synchronization detection, thus creating a more realistic model of the online tracking ecosystem.

Chapter 6

Conclusions

The first goal - designing a methodology of collecting privacy and tracking related data and implement it as a privacy research platform - has been successfully met. Cookie Crumble Tracer, a privacy measurement tool capable of automated data collection with limited human supervision, has been designed and implemented. It has proved to be an effective way of solving the main objectives of this study - visiting a set of predefined domains, locating specific Document Object Elements corresponding to Cookie Privacy Warnings with adequate recall, and interacting with them to modify the cookies set in the browser. To collect its data and measure the user's impact on creating the online tracking ecosystem, CCT visited 1000 most popular domains according to Tranco list, finding 497 websites containing a cookie privacy warning. What is more, CCT has proved to be successful at detecting cookie synchronization - a privacy intruding mechanism used to leak potentially sensitive data.

The data collected by interacting with Cookie Privacy Warnings has been used to create graph models representing the ecosystem of publishers and trackers that users build around themselves. These graphs capture different aspects of the online tracking ecosystem. The analysis of PT graphs show that half of all distinct trackers is hosted by only 32% of publishers when opting in and 25% when opting out. Similar disproportion has been reported by examining the TT graphs - just 11% of all trackers is responsible for 50% of all data sharing.

By deploying various graph metrics we have been able to detect domains, mostly trackers, of high embeddedness with respect to connectivity with other nodes. By observing how cookie policies change the rankings of most influential network vertices we have learned one thing - the structure of the online tracking ecosystem is not significantly affected by either opt-in or opt-out policy. The policy affects the size of the network - opting out can reduce the number of observed trackers in the Publisher-Trackers network by around 38% and around 17% in the Trackers-Trackers network. Moreover, opting out can reduce the number of collaborating tracker pairs by 35% and the amount of data shared between third parties by up to 40%. However, the structure of the network and the centrality of individual nodes remain very similar for both cookie policies. Rankings of top domains within the networks, as ranked by different metrics, have been observed to be almost immutable, with top positions firmly occupied by

companies such as Google, Facebook, or Amazon. Opting out has been observed to increase the centrality of large, well-known companies, relative to the other, less important nodes in the network. This points to the fact that GDPR enforcement has little influence on these companies, both in terms of their importance in the online tracking ecosystem, or coverage across websites. Hence GDPR poses no threat to these companies' dominance in the tracking environment developed around individual users.

By building CCT we have provided a tool that can measure how an individual user affects the online tracking ecosystem, which gradually builds around them as more and more id-like cookies are set in the browser. In this study, we have used websites from the Tranco list, but this list could be substituted for any other set of domains. Any person wishing to study their online privacy can use domains extracted, for example, from their browser history. In this way, CCT enables measurements on how much control over their privacy a person with a specific browsing pattern has.

In conclusion, Cookie Crumble Tracer, together with collected data and its subsequent analysis, can be seen as a contribution to the area of online privacy research. A limited number of studies have been conducted into the structure of the online tracking ecosystem, with none of them focusing on directly interacting with GDPR Cookie Privacy Warnings. In previous work, data was collected through user studies, meaning that the cookie policy adopted by these users was not treated as a factor potentially influencing the online tracking ecosystem. Our study contributes with a novel approach that successfully simulates human-like behavior, thus shedding light on the state of the online tracking ecosystem at a more personal level.

Bibliography

[1] F. Roesner, T. Kohno, and D. Wetherall, "Detecting and defending against third-party tracking on the web," in *9th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 12)*, pp. 155–168, 2012.

[2] 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*, pp. 674–689, 2014.

[3] K. Solomos, P. Ilia, S. Ioannidis, and N. Kourtellis, "Clash of the trackers: Measuring the evolution of the online tracking ecosystem," *arXiv preprint arXiv:1907.12860*, 2019.

[4] A. Dabrowski, G. Merzdovnik, J. Ullrich, G. Sendera, and E. Weippl, "Measuring cookies and web privacy in a post-gdpr world," in *International Conference on Passive and Active Network Measurement*, pp. 258–270, Springer, 2019.

[5] M. Degeling, C. Utz, C. Lentzsch, H. Hosseini, F. Schaub, and T. Holz, "We value your privacy... now take some cookies: Measuring the gdpr's impact on web privacy," *arXiv preprint arXiv:1808.05096*, 2018.

[6] J. Sørensen and S. Kosta, "Before and after gdpr: The changes in third party presence at public and private european websites," in *The World Wide Web Conference*, pp. 1590–1600, 2019.

[7] V. Kalavri, J. Blackburn, M. Varvello, and K. Papagiannaki, "Like a pack of wolves: Community structure of web trackers," in *International Conference on Passive and Active Network Measurement*, pp. 42–54, Springer, 2016.

[8] M. A. Bashir and C. Wilson, "Diffusion of user tracking data in the online advertising ecosystem," *Proceedings on Privacy Enhancing Technologies*, vol. 2018, no. 4, pp. 85–103, 2018.

[9] C. o. t. E. U. European Parliament, "Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec," *https://eur-lex.europa.eu/eli/reg/2016/679/oj*, 2016.

[10] Mozilla, "Same-origin policy," *https://developer.mozilla.org/en-US/docs/Web/Security/Sa*.

[11] M. Trevisan, S. Traverso, E. Bassi, and M. Mellia, "4 years of eu cookie law: Results and lessons learned," *Proceedings on Privacy Enhancing Technologies*, vol. 2019, no. 2, pp. 126–145, 2019.

[12] T. Urban, D. Tatang, M. Degeling, T. Holz, and N. Pohlmann, "The unwanted sharing economy: An analysis of cookie syncing and user transparency under gdpr," *arXiv preprint arXiv:1811.08660*, 2018.

[13] "Phantomjs - scriptable headless browser," *https://phantomjs.org/j*, Feb 2020.

[14] 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*, pp. 1388–1401, 2016.

[15] G. Acar, M. Juarez, N. Nikiforakis, C. Diaz, S. Gürses, F. Piessens, and B. Preneel, "Fpdetective: dusting the web for fingerprinters," in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pp. 1129–1140, 2013.

[16] "The selenium browser automation project," *https://www.selenium.dev/documentation/en/*, April 2021.

[17] "Browser market share worldwide," *https://gs.statcounter.com/browser-market-share*, Feb 2021.

[18] V. L. Pochat, T. Van Goethem, S. Tajalizadehkhoob, M. Korczyński, and W. Joosen, "Tranco: A research-oriented top sites ranking hardened against manipulation," *arXiv preprint arXiv:1806.01156*, 2018.

[19] B. Molnar, "Measuring the cookie-setting behaviour of web pages showing privacy warnings,"

[20] Fanboy, MonztA, Famlam, and Khrin, "Easylist," *https://easylist.to/*, Jun 2020.

[21] X. Hu and N. Sastry, "Characterising third party cookie usage in the eu after gdpr," in *Proceedings of the 10th ACM Conference on Web Science*, pp. 137–141, 2019.

[22] I. Sanchez-Rola, M. Dell'Amico, P. Kotzias, D. Balzarotti, L. Bilge, P.-A. Vervier, and I. Santos, "Can i opt out yet? gdpr and the global illusion of cookie control," in *Proceedings of the 2019 ACM Asia conference on computer and communications security*, pp. 340–351, 2019.

[23] D. L. Wheeler, "zxcvbn: Low-budget password strength estimation," in *25th {USENIX} Security Symposium ({USENIX} Security 16)*, pp. 157–173, 2016.

[24] 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*, pp. 289–299, 2015.

[25] P. Papadopoulos, N. Kourtellis, and E. P. Markatos, "Exclusive: How the (synced) cookie monster breached my encrypted vpn session," in *Proceedings of the 11th European Workshop on Systems Security*, pp. 1–6, 2018.

[26] M. A. Bashir, S. Arshad, W. Robertson, and C. Wilson, "Tracing information flows between ad exchanges using retargeted ads," in *25th {USENIX} Security Symposium ({USENIX} Security 16)*, pp. 481–496, 2016.

[27] P. Papadopoulos, N. Kourtellis, and E. Markatos, "Cookie synchronization: Everything you always wanted to know but were afraid to ask," in *The World Wide Web Conference*, pp. 1432–1442, 2019.

[28] "Violin plot basics - matplotlib 3.3.4 documentation," *https://matplotlib.org/stable/gallery/statistics/violinplot.html*, Jan 2021.

[29] G. C. Ousey, P. Wilcox, and S. Brummel, "Déjà vu all over again: Investigating temporal continuity of adolescent victimization," *Journal of Quantitative Criminology*, vol. 24, no. 3, pp. 307–335, 2008.

[30] R. R. K. Opsahl and R. Reitman, "The disconcerting details: How facebook teams up with data brokers to show you targeted ads," *Electronic Frontier Foundation*, vol. 22, 2013.

[31] Wikipedia contributors, "Betweenness centrality — Wikipedia, the free encyclopedia," 2021. [Online; accessed 14-March-2021].