Client-side and Server-side Tracking on Meta: Effectiveness and Accuracy

Asmaa El fraihi*
LIX, CNRS, Inria, École Polytechnique, IP Paris
Palaiseau, France

Nardjes Amieur*
LIX, CNRS, Inria, École Polytechnique, IP Paris
Palaiseau, France

Walter Rudametkin
Univ. Rennes, Inria, CNRS, IRISA, IUF
Rennes, France

Oana Goga
LIX, CNRS, Inria, École Polytechnique, IP Paris
Palaiseau, France

ABSTRACT

Growing concern over digital privacy has led to the widespread use of tracking restriction tools, such as ad blockers, Virtual Private Networks (VPN), and privacy-focused web browsers. All major browser vendors have also deprecated, or plan to deprecate, third-party cookies to reduce tracking. Despite these efforts, advertising companies continuously innovate to overcome these restrictions. Recently, advertising platforms, like Meta, have been promoting *server-side tracking* solutions to bypass traditional *browser-based tracking* restrictions.

This paper explores how server-side tracking technologies can link website visitors with their user accounts on Meta products. The goal is to assess the *effectiveness* and *accuracy* of employing this technology, as well as *the effect of tracking restrictions* on online tracking. Our methodology involves a series of experiments where we integrate Meta's client-side tracker (the Meta Pixel) and server-side technology (the Conversions API) on different web pages. We then drive traffic to these pages and evaluate the success rate of linking website visitors to their profiles on Meta products.

Our findings show that Meta's server-side technology can match between 34% and 51% of website visitors to user profiles on Meta products using basic information like the visitor's IP address, user agent, and location data. This is comparable to Pixel-based user matching in optimal conditions (i.e., in the absence of tracking restrictions), which links between 42% and 61% of user profiles. Nevertheless, we see a considerable difference in accuracy: while the Pixel-based tracking achieves 100% accuracy, less than 65% of the profiles matched by server-side tracking are accurate.

KEYWORDS

Server-side tracking, third-party cookies, client-side tracking, Meta Pixel, Conversions API, cross-website tracking, retargeting, browser and device fingerprinting.

1 INTRODUCTION

In recent years, public concern about online privacy has been growing as consumers become aware of extensive online data collection practices and how advertising companies handle their data. This has resulted in the implementation of numerous online privacy protections, which have heavily impacted the digital advertising industry. For example, Apple has implemented Intelligent Tracking Prevention (ITP) in WebKit [40, 41], which blocks third-party cookies and isolates each website's storage to limit cross-website tracking. Furthermore, recent versions of web browsers, such as Firefox and Safari, block third-party cookies by default, while Google Chrome, after multiple delays, plans to phase out third-party cookies by late 2024 [6]. Ad blocking extensions also see extensive use, boasting a worldwide market penetration rate of 42.7% [5]. Furthermore, recent statistics indicate a growing use of Virtual Private Networks (VPNs), with an estimated 31% of users worldwide currently relying on them. These restrictions have decreased the efficiency of browser-based tracking that relies on third-party cookies to match Internet users across domains (a.k.a. cross-website tracking).

The *Meta Pixel*, a JavaScript code that can be installed on advertisers' websites to track user activity [15], plays a crucial role within Meta Ads, one of the biggest advertising platforms today. The script can track users across websites, collecting data about the actions they perform while browsing, such as button clicks and content views. This allows Meta to capture massive amounts of behavioral data on users, which is then used to optimize ad delivery, measure ad effectiveness, and allows advertisers to target potential prospects, returning customers, as well as people who have shown an interest in businesses similar to theirs.

Since privacy-preserving technologies and various tracking restrictions impact the Meta Pixel, Meta has been encouraging advertisers to adopt the *Conversions API* [9], their server-side tracking technology, which allows advertisers to send tracking data directly from their servers to Meta's advertising tools, instead of relying on connections from the user's browser. The tracking data collected from each website visitor is matched to a unique user, who can later be targeted with advertisements. The Pixel achieves this through third-party cookies containing unique identifiers linked directly to users. In contrast, the Conversions API relies on personal identifiers, such as emails, phone numbers, or full names, which advertisers collect from users visiting their websites, or on *fingerprinting* information, such as the visitor's IP address and user agent, readily available to the advertiser for each website visit.

[*] Asmaa El fraihi and Nardjes Amieur contributed equally to this work.

Proceedings on Privacy Enhancing Technologies 2024(3), 431–445
© 2024 Copyright held by the owner/author(s).
https://doi.org/10.56553/popets-2024-0086

Our paper aims to measure and compare the **effectiveness** and **accuracy** of **client and server-side tracking technologies** as implemented by Meta while also assessing the impact of tracking restrictions on them. First, we investigate *the extent to which website visitors tracked using the Conversions API can be matched to user profiles on Meta products* using only fingerprinting data such as IP addresses, device information, and IP address-based geolocation data. We compare this with the Pixel's effectiveness in optimal conditions, characterized by the absence of tracking restrictions and enabled third-party cookie use. Next, we assess *how tracking restrictions can impact the effectiveness of both the Pixel and the Conversions API*. Finally, since our implementation of server-side tracking technology is based on user fingerprinting and could, hence, lead to false matches, we test *the accuracy of matching website visitors to their user profiles on Meta products*.

Our proposed measurement methodology relies on using targeted ad campaigns to quantify the number of website visitors linked to user profiles via tracking technologies (Section 3). We conduct a series of experiments by creating websites that integrate Meta trackers and directing authentic user traffic to these sites through crowdsourcing platforms. For each website visitor, we collect various tracking data and send it to Meta. Using this tracking data, we create targets for ad campaigns on Meta and investigate how many website visitors are linked to user profiles by assessing the reach of our ad campaigns. To our knowledge, our proposed measurement methodology is novel and may be of independent interest to the community.

We highlight the following key findings:

Effectiveness: We conducted seven experiments, ensuring that participating users used browsers with third-party cookies enabled and did not employ any tracking restrictions (Section 4). Using statistics from Meta about our ad campaigns, we measured the *tracking effectiveness* of each experiment, i.e., the fraction of users who visited our website and whom we subsequently reached with our ads. Our results demonstrate that Meta's server-side tracking tools can match website visitors to social media profiles using only IP addresses, location information, and user agents. The Conversions API matched between 34% to 51% of website visitors to user profiles, depending on specific configurations, such as Desktop vs. Mobile. In comparison, the Meta Pixel matched between 42% to 61% of our website visitors under identical experiment conditions. Therefore, server-side tracking can achieve an effectiveness comparable to client-side tracking in ideal conditions.

Tracking restrictions: We conducted experiments to assess the effectiveness of client and server-side tracking technologies in the presence of four types of tracking restrictions: privacy-preserving browser protections, filter lists, VPNs, and user agent spoofing (Section 5). Our findings reveal that the Meta Pixel can track users even without third-party cookies. This implies that, alongside cookies, the Pixel relies on browser and device information to associate website visitors with user accounts. This is possible as requests to Meta servers are allowed but simply do not contain cookies.

When filter lists are installed in the browser, as expected, the Pixel is blocked, preventing third- and first-party cookies used for tracking from being set. Consequently, client-side tracking becomes infeasible. However, tracking requests sent from the website servers via the Conversions API go undetected. Hence, the use of filter lists has no impact on server-side tracking.

Accuracy: Server-side tracking relies on device fingerprinting and may result in false matches, particularly when the matching is based on IP addresses and user agents. We face uncertainty regarding whether Meta delivers our ads to users who visited our website or to unrelated users who share the same IP address and user agent since Meta preserves the anonymity of users exposed to our ads.

To *measure accuracy*, we leverage a feature provided by Meta that enables advertisers to check the overlap between users reached by two ad campaigns (Section 6). Instead of recruiting users on a crowdsourcing platform, we initiated an ad campaign on Facebook for user recruitment. This approach allows us to examine the overlap between the *recruiting* Facebook ad campaign and the subsequent *tracking* Facebook ad campaign. In a scenario of perfect user matching, the tracking Facebook ad campaign should exhibit a 100% overlap with the recruiting Facebook ad campaign. Our results indicate that while the Pixel can achieve 100% matching accuracy via third-party cookies, server-side tracking, relying on IP addresses, user agents, and IP address-based geolocation, only achieves an accuracy range of 60% to 65%. This implies that the *Conversions API produces false matches for over one-third of our website visitors*. Therefore, advertisers employing server-side tracking with only fingerprinting data face the risk of misspending one-third of their retargeting ad budget.

First-party cookies: We investigate the role of first-party cookies in matching website visitors to users on Meta. Our experiments show that including the "Facebook Browser ID" (*fbp*) in the event sent by the server-side tracker does not improve tracking effectiveness. As for the "Facebook Click ID" (*fbc*), our results indicate that, while the difference is minimal, incorporating its value enhances tracking accuracy and effectiveness. Additionally, our experiments show that the *fbp* and *fbc* cookies can link data from different tracking events, facilitating the connection of various user fingerprints for future matching.

The end of third-party cookies was expected to enhance user privacy. However, as revealed in this paper, IP addresses and device fingerprinting have emerged as effective advertising identifiers. Remarkably, relying solely on device data allows linking many Internet users to unique profiles. While most tracking restrictions impede the effectiveness of client-side tracking, the shift to server-side tracking presents a potential avenue for advertisers to circumvent these limitations. Furthermore, unlike the Pixel, detecting whether a website uses Meta's server-side tracking and understanding the nature of the data being collected is challenging, if not impossible.

We hope that our proposed measurement methodology can be adapted to further scrutinize client-side and server-side tracking technologies. Additionally, it is crucial to raise awareness about the limitations of privacy tools and underscore that various data types, including IP addresses and browser details, are used for tracking. Finally, further research is needed to develop methods for identifying and limiting server-side tracking.

2 BACKGROUND

This section reviews the mechanisms behind online tracking employed by Meta, how they are implemented, and discusses common tracking restrictions available to users.

2.1 Tracking Mechanisms

The online advertising landscape consists of four main players: the *advertising platform* (e.g., Meta), *advertisers*, *users*, and *publishers*. Advertising platforms gather diverse data points on their users, such as viewed content and visited websites, to infer their interests and effectively deliver tailored ads [34]. While online platforms such as Meta can easily track users' activity across their products (e.g., Instagram, Facebook), there is a lot of value in tracking users outside of their products as well, as it helps to better understand user behaviors and interests. This is referred to as third-party tracking and is central to ad optimization algorithms employed by Meta to achieve maximum user engagement [34].

Advertisers implement tracking to get insights on their customers and re-target them with ads, while publishers implement tracking to allow the advertising platforms to know which users are checking their websites to deliver the most effective/tailored ads. This improves the likelihood of engagement and conversions, directly impacting the publisher's revenues. In this paper, we focus on measuring tracking mechanisms proposed by Meta to advertisers that have a product website. The advertiser's website isn't a publisher in our setup because it doesn't show ads.

2.1.1 *Client-side tracking.*

Also known as browser-based tracking, this type of online tracking relies on the user's browser to send behavioral data to advertising platforms. To identify users, client-side trackers store unique identifiers on the user's browser in the form of third and first-party cookies, which allows for re-identifying the device and aggregating the collected data across websites, linking it to a unique profile (Figure 1). When an internet user visits an advertiser's website, the tracking script is retrieved from the advertising platform and loaded into the user's browser. This script retrieves first- and third-party cookies installed on the browser and reports user activity to the advertising platform.

The Meta Pixel: Meta's solution for client-side tracking is known as the Meta Pixel, a piece of JavaScript code that advertisers can add to their websites. The script loads a small library of functions that advertisers can use to report user actions on their website. These actions are called *events*. Meta allows advertisers to report standard events (e.g., button clicks, purchases), and custom events with specific parameters. When reporting events, the Pixel sends Meta's third- and first-party cookies and details about the user's device such as IP address, geolocation information, brand and model, and operating system. Each reported event includes this data, regardless of whether the user has an account on Meta products, and it is not dependent on their logged-in status [12].

2.1.2 *Server-side tracking.*

a tracking technique that involves hosting the tracking implementation on the website's servers. When a user interacts with the website, Website interactions are sent to the advertiser's servers. The server processes the request, acquiring data such as the device and browser fingerprint (IP address, geolocation data, user agent, first-party cookies, screen width, etc.), alongside the user's personally identifiable information if it is available to the advertiser. On the server, a tracking logic is implemented to transfer all this data to the advertising platform (Figure 2).

The Conversions API: Meta's server-side tracking technology, known as the Conversions API (CAPI), establishes a direct connection between the website's servers and Meta technologies. Operating independently of browser cookies, CAPI presents a robust tracking solution. Advertisers can transmit a range of events, spanning default to custom events. Additionally, advertisers can enhance tracking data by including additional customer details that are not typically available in the browser, such as previous offline purchases. To link tracking data generated by website visitors to the users, advertisers can send personally identifiable information (PII), such as full names and email addresses, geolocation data, and even IP addresses. Furthermore, if advertisers choose to implement both the Meta Pixel and CAPI on their website they can send the first-party cookies installed by the Pixel for each website visitor, via the server-side connections [19]. Meta offers various methods for implementing the Conversions API, from direct code integration using the Facebook Business SDK to no-coding solutions like Conversions API Gateway, Commerce Platforms, or tools such as Google Tag Manager [16].

2.1.3 *Client vs. Server-side Tracking.*

Client-side tracking achieves high accuracy by matching website visitors to their user profiles through third-party cookies (e.g., *c_user* cookie for the Meta Pixel). However, it faces limitations from tracking restrictions, like filter lists (e.g., ad block) and browsers that disable third-party cookies. On the contrary, server-side tracking is immune to such restrictions, offering advertisers more control over shared data. For instance, advertisers can send only the user's email address, redacting their IP address and browser details. However, this tracking may have limited user data compared to client-side tracking, especially for unauthenticated users on the advertiser's website. This limitation could affect the effectiveness and accuracy of matching website visitors to their user profiles on the advertising platform through device and browser fingerprinting.

2.2 Tracking Restrictions

2.2.1 *Browser privacy.*

Most efforts to protect user privacy on the browser focus on limiting the use of third-party cookies. By 2024, most commonly used browsers would disable third-party cookies by default [23, 31, 39, 41]. Some browsers are also working to limit fingerprinting, a technique used to track users based on unique device and browser details. Safari, for example, has introduced measures to reduce the effectiveness of fingerprinting by limiting browser-related information websites can access [3]. Another feature on browsers is the "Do Not Track" (DNT) signal. Although not universally adopted, the DNT signal was designed to inform websites that a user does not want to be tracked. However, its effectiveness depends on whether websites respect the signal [24].

2.2.2 *Filter lists.*

Filter lists are a set of rules that are used to automatically remove unwanted content from websites during load. Filter lists identify content to block based on various criteria such

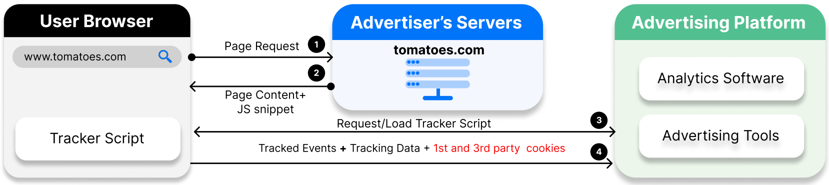


Figure 1: Client-side tracking.

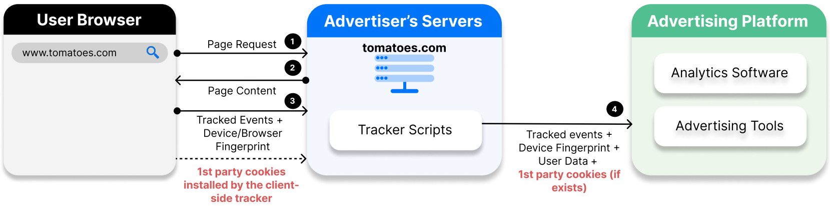


Figure 2: Server-side tracking.

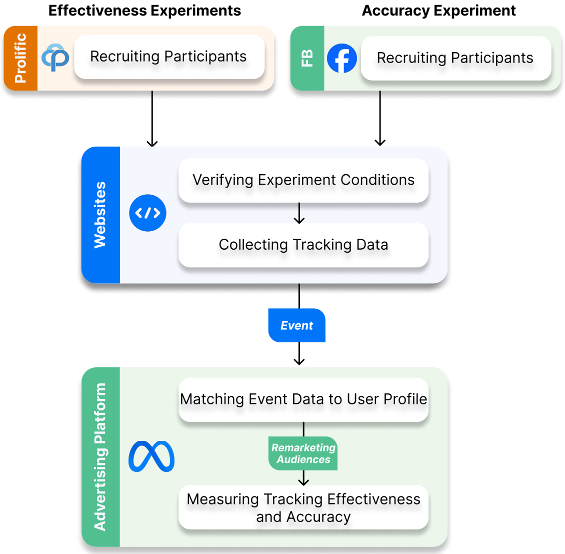


Figure 3: General workflow of experiments.

as the URL it is loaded from, characteristics of the HTML (labels, CSS selectors), or both. Originally, filter lists were designed to eliminate annoying website disruptions, such as ads and pop-ups. Now, they can also enforce restrictions on third-party trackers, including those from Meta, to block requests to advertising platforms. Adblocking extensions, such as AdBlock [7], or browser built-in filters like Brave Shields [8], make use of filter lists. Filter lists are also used by private search engines such as Qwant and DuckDuckGo.

This tracking restriction mechanism, unlike browser privacy settings that only block third-party cookies, may prevent client-side trackers from loading onto the webpage, thereby preventing the installation of any tracking cookies both first- and third-party.

2.2.3 VPN and proxy. Virtual private networks (VPNs) and proxy servers are two common methods for masking an internet user's IP address. VPNs work by routing the user's data traffic through an encrypted virtual tunnel, which disguises their IP address and location information. Proxy servers act as an intermediary between the user's device and the Internet, which can also obscure the user's IP address. VPNs and proxy servers can be effective in limiting IP address-based and location-based tracking, but they are not foolproof. Even if users employ VPN connections while browsing online, the trackers implemented by advertisers can still identify them through other identifiers, such as third-party cookies and PII.

2.2.4 User agent switchers. User agent switchers are browser add-ons that enable users to modify their user agent string while accessing websites (e.g., User-Agent Switcher for Chrome). The user agent string is information the browser transmits to websites through HTTP headers. It generally contains details about the user's browser, device, and operating system. Altering the user agent string can enhance user privacy by concealing the browser fingerprint [1].

3 METHODOLOGY

Our experiments aim to assess the number of website visitors that we can track using the Pixel and the Conversions API. We attempt to match visitors to user profiles on Meta products and to measure the accuracy of this matching. To achieve this, we create websites that integrate both the Meta Pixel and the Conversions API. We drive user traffic to these websites through a recruitment process on the crowdsourcing platform Prolific [21] and through Facebook ads. Subsequently, we use Meta's marketing and advertising tools to assess the effectiveness and accuracy of the trackers in re-identifying users who visited our websites. Figure 3 illustrates the general workflow of our experiments. For clarity, this section presents the main building blocks and logic of our experiments; we provide more detail on the precise parameters of each experiment in Sections 4, 5, and 6.

3.1 User Recruiting

Using synthetic traffic with automated browsers and fake user accounts for our experiments is inadequate as it could be detected and yield biased results. Instead, we direct real users with existing Facebook accounts to our websites. We use Prolific to drive traffic for the *effectiveness experiments*. Prolific allows the definition of selection criteria for participants. We recruited users from the United States who self-reported as maintaining an active Facebook account (active accounts are defined as being accessed at least once a month). We also applied device prescreening criteria, as some of our experiments require only desktop devices, while others require only mobile devices. Once selected, participants are invited to fill out a form on our website about their daily social media use, which takes approximately 1 minute (Table 8, Appendix).

To drive traffic for the *accuracy experiments*, we run a Facebook advertising campaign targeted at a broad audience. We explain why we need to recruit users through Facebook in Sections 3.4.2.

3.2 Website: Verifications

For each experiment, we select participants with specific browser characteristics. Since self-reporting can be unreliable, we implemented a set of automatic checks to ensure that users met the experiment's conditions.

(1) *The browser*: We check the type of browser (e.g., Chrome, Safari) the participant uses to complete the study by parsing the user agent string. This verification ensures that we gather data from participants accessing our websites using the specified browser for the designated experiment.

(2) *Filter lists use*: To detect the presence of filter lists, we added HTML elements commonly blocked by ad blockers, with names containing words like "ad" or "popup". We also tested loading scripts with names that contain "ad" or "prebid". While these methods can be effective in detecting the use of certain ad block extensions (e.g., AdBlock, Adblock Plus), they may overlook other widely used ones (e.g., Privacy Badger). Hence, we initiated a JavaScript call to a bait URL used by Facebook trackers (https://connect.facebook.net/en_US/fbevents.js) sourced from established filter lists like EasyList [10]. This method had the lowest incidence of false negatives among the approaches we tested. A blocked call indicates the presence of filter lists. If participants were found to be using filter lists during the study, they were asked to disable them and reload the page. This does not impact the tracking script, as cookies are already installed if the user is logged in to Facebook. Therefore, the script would retrieve the cookies' value upon page reload.

(3) *VPN use*: Participants are only selected after we determine that their traffic is not routed through a VPN. To identify VPN or proxy use among participants, we use a third-party service. We tested ipinfo.io [13], IPQualityScore [14], VPNAPI.io [25] and IP2Location [36] across various extensions and apps (e.g., Opera in-browser VPN, VeePN Chrome extension, SuperVPN app on mobile). The services examine IP addresses and determine if they are associated with a recognized VPN server. We opted for VPNAPI.io, as it had the lowest incidence of false negatives in our tests.

(4) *Device*: We added automatic device verification checks (i.e., for mobile and desktop) by parsing the user agent. We initiate tracking only when the participant's device meets the experiment criteria.

3.3 Website: Tracker Implementation

To configure tracking on our websites, we registered as an advertiser on Meta products, which gives us access to advertising and analytics tools. This involved creating a Facebook Page and setting up a business account that allows accessing Meta's Event Manager.

To integrate the Meta Pixel on a website, we include its JavaScript snippet in the HTML header tag [18]. Similarly, we used direct code integration with Facebook Business's Software Development Kit (SDK) for Python [11] to integrate the Conversions API. We send Pixel events, labeled as *Page View*, which contain the user's IP address and device information (user agent, device type, operating system, brand and model, screen resolution, preferred language, and referrer), along with first and third-party cookies. For the Conversions API, we send events labeled as *View Content* with the visitor's IP address, user agent, and IP address-based location data obtained from ipinfo.io [13] (country, city, and postal/zip code). Some data may vary depending on the experiment.

Each time a user visits one of our websites, two *events* are sent, one corresponding to the Pixel and the other to the Conversions API. We collect events only after participants have given consent to our privacy notice, following verification of experiment conditions and form submission. Therefore, the total number of users we recruit on Prolific for each experiment exceeds the number of users for whom we send events and report in the paper.

Finally, in the Event Manager, we created event "Datasets" that serve as endpoints for events sent from our websites. We have distinct *event endpoints* for each tracker implementation (client-side or server-side) and each experiment setup (e.g., Desktop, Mobile), ensuring isolation between events from the same website and between events from different experiments. Each event sent from our website contains the ID of the corresponding event endpoint.

3.4 Advertising Platform: Measures of Tracking Effectiveness and Accuracy

To estimate tracking effectiveness and accuracy, we exploit various features and statistics provided by Meta's Business Manager.

3.4.1 Estimating Tracking Effectiveness.

The *effectiveness* of a tracker is gauged by the number of website visitors matched to user profiles on Meta products through events sent via the tracker. Meta provides the number of website events received on each event endpoint (Fig. 5, Appendix). However, this metric does not indicate the platform's ability to link a website visit to a user–it is not possible to distinguish visits from users who have an account from those who do not nor to de-duplicate multiple visits from the same user.

Nevertheless, Meta enables the creation of *remarketing audiences* for each event endpoint through a feature called "custom audiences" [28]. For this, Meta's tools attempt to associate events with unique users on the platform. The size of the resulting *remarketing audiences* would constitute the *true effectiveness* of tracking. Unfortunately, Meta only offers imprecise size estimations like "Below 1000", for the resulting audiences (Fig. 6, Appendix).

To manage our lack of access to the metric, we look at a different but closely related measurement, which is *reach effectiveness*. The idea is to use the advertising platform to create an *ad campaign* that targets people in a remarketing audience. We instruct the platform to send an ad to users we tracked on our website corresponding to a particular event endpoint. Subsequently, Meta provides us with the precise number of users our ad campaign has reached, which we take as a proxy for measuring effectiveness (Fig. 7, Appendix).

This measurement represents a lower bound of true effectiveness as our ads might not reach some users for various reasons, such as competition for the ad space or users not being online during our ad campaigns. We measure the reach effectiveness for every event endpoint separately through distinct ad campaigns.

To ensure a fair comparison between tracker configurations, our ad campaigns have identical setups. To maximize the number of users we reach corresponding to each event endpoint, we instruct the ad manager to display ads to the maximum possible number of people within our targeted audience using the "Awareness" objective on Meta ads. No specific targeting criteria are selected, and the ads are shown across all Meta products and available placements. The ads have the same budget (€5 per day) and run for 20 days for each experiment, with reach stabilizing around the first ten days. The ads include only an image and text and advertise an extension for research purposes (Fig. 4 is a screenshot of the ad).

3.4.2 Estimating Tracking Accuracy. The *accuracy* of a tracker is determined by the number of website visitors that were *correctly* matched to their user profiles on Meta products, using the events we send from our website via the tracker. In the case of server-side tracking, relying on user fingerprinting may result in false matches, particularly when the matching is solely based on IP addresses and user agents. When assessing the effectiveness, we face uncertainty regarding whether Meta delivers our ads to users who visited our website or to unrelated users who share the same IP address and user agent. Due to privacy considerations, Meta does not provide the list of users who receive our ads, which prevents us from comparing the participants we directed to our website with those who were successfully matched.

To measure accuracy, we use a feature in Meta's Ad Manager that lets advertisers check the *overlap between users reached by two ad campaigns*. Instead of recruiting users on Prolific, we initiated an ad campaign on Facebook. This establishes a control audience—users that click on our *recruiting ad* on Facebook and then participate in our study. Following our usual approach, we integrate a tracker on our website and create a target audience for a second ad, the *tracking ad*. Accuracy is then measured by examining the overlap between the *recruiting ad* and the subsequent *tracking ad* campaigns. In a scenario of perfect user matching, there should be 100% overlap between the ad campaigns as all users who visited and were tracked by our website had done so by clicking on our Facebook ad.

In total, we spent €1520 to run 86 ad campaigns between March and November 2023 to assess tracking effectiveness and accuracy. This amount encompasses exploratory experiments, including tests of different tracker implementations.

3.5 Limitations

Meta provides imprecise estimations for the number of events matched to user profiles. Instead, we rely on the ad campaign's reach as a proxy to assess the effectiveness of tracking tools. The reach estimates the number of users successfully matched to their social media profiles. However, not all visitors may be reached, as our ad campaigns compete with other advertisers for placement and can only reach users who have logged into their accounts during the duration of the ad campaign. Nevertheless, our primary focus is to compare server-side and client-side tracking. The campaigns target the same population in identical circumstances, ensuring a fair comparison.

3.6 Ethics Statement

This research investigates online tracking mechanisms, focusing on Meta's advertising platform. We gathered data from recruited participants and transmitted it to Meta through their trackers. Throughout the process, we have been committed to upholding the standards of ethical research to ensure the protection of the participant's rights and privacy:
(1) *Beneficence:* This research comprehensively examines tracking technologies and their implications on user privacy, providing insights to enhance privacy measures and support regulatory efforts.
(2) *Informed Consent:* Prior to participating in this study, all participants were explicitly informed about the research objectives, data collection, storage, and deletion process. This includes participants recruited via Prolific for the effectiveness experiments, and on Facebook for the accuracy experiment. Participants were also informed that their data is transmitted to Meta and that they will potentially be targeted with ads on any of their Meta accounts. We have implemented a clear GDPR-compliant consent banner, which was approved by our institution's Data Protection Delegate (DPD). Due to inaccuracies in Meta's tracking technologies, some individuals who did not voluntarily participate in our experiment may receive the ads we run (Fig. 4). However, these individuals only view the ads and are not being tracked without their consent.
(3) *Minimal risk:* We adhered to the guidelines concerning data handling and storage to safeguard the confidentiality and anonymity of the participants' data. We only stored non-sensitive data, like form answers, user agent strings, and the use of filter lists and Virtual Private Networks (VPNs), for analytical purposes. These data records were securely stored in our laboratory infrastructure for a maximum of 12 months after the study concluded. We do not collect other personally identifiable information.
(4) *Ethical Approval:* This study received legal approval from our institution's DPD. We also have the approval of our Institutional Review Board to audit Meta's advertising system through ads.

4 EFFECTIVENESS OF TRACKING TECHNOLOGIES

This section compares the reach effectiveness of Meta's client-side and server-side tracking technologies. Due to differences between Desktop and Mobile environments, we conducted separate experiments for each. To ensure an unbiased comparison and to evaluate tracker effectiveness under optimal conditions, we select participants who have not enabled tracking restrictions, including cookie blockers, filter lists, and VPNs. For the Conversion API (CAPI), we use minimal data (IP address, user agent, IP address-based geolocation), setting a lower bound for its effectiveness as advertisers can provide additional details (e.g., names, email addresses). The same set of users is used to evaluate client and server-side trackers, and we execute the ad campaigns simultaneously.

4.1 Desktop Traffic Experiment

4.1.1 Instantiation. We present the parameters we use to instantiate the experimental setup described in Section 3.

Table 1: Effectiveness of Meta client-side and server-side tracking on desktop and mobile. The *fbp* column shows if CAPI events contain the *fbp* cookie value. The Users column represents the number of users for which we trigger events. Percentages for Pixel and CAPI represent retargeted users among the total participants. Overlap indicates users matched with both client and server-side trackers.

Experiment	#	*fbp*	Users	Pixel	CAPI	Overlap
Desktop: Chrome	1	No	400	46%	51%	43%
	2	No	1000	43%	44%	48%
	3	Yes	500	44%	45%	73%
	4	Yes	500	42%	45%	74%
Mobile: Chrome	1	No	300	61%	34%	50%
	2	No	250	N/A	51%	N/A
	3	No	175	N/A	50%	N/A

- **N/A** indicates that the tracker was not implemented on the website.
- For each experiment, we had multiple attempts with the same configuration, '#' refers to the attempt number.

User recruiting: Aside from the prescreening specified in Section 3.1 (users from the U.S. that self-report to maintain an active Facebook account), we select Prolific users taking the surveys on desktops. We recruited 2400 users across four experiments performed between April and October 2023.

Website verifications: We conduct automatic verifications (refer to Section 3.2) to check if: (1) users are accessing the website on a desktop device; (2) are using Google Chrome–which is a browser that supports third-party cookies; (3) are not using VPNs; and (4) are not using filter lists, such as ad blockers or privacy extensions. Users who do not validate these verifications are discarded from the experiment and not tracked further.[1]

Website tracking implementation: We deploy a website that integrates both the Pixel and the CAPI, connecting each tracker to a distinct event endpoint. Before participants access the website they answer a consent form and a few questions about their social media use (see Table 8, Appendix). Upon clicking the submit button, two events are generated: a *Page View* event with the Pixel and a *View Content* event with CAPI. We use the default Pixel setup, which sends the IP address, device, and browser details in HTTP headers, along with its first and third-party cookies [18]. The Pixel's first-party cookie is called *fbp*(Facebook Browser ID) and is unique for each website visit and has a lifespan of 90 days. When implementing the CAPI, we can optionally send the first-party cookie. Since the *fbp* value is unique to each website visit [17], Meta could potentially use its value to link visitors with profiles on their products. Therefore, by including the *fbp* cookie with CAPI events, the overall tracking effectiveness might be improved. Hence, we deploy two versions of the experiment. One in which CAPI sends the IP address, user agent, and IP address-based geolocation for each participant, and another that includes the *fbp* cookie.

4.1.2 Results. The reach effectiveness results for both the client and the server-side trackers are presented in Table 1 (the Pixel and CAPI columns). The *fbp* column shows if the CAPI events contain the *fbp* cookie value. We performed two experiments for each setup to check for consistency. The User column represents the number of users for which we triggered events on our website, which is lower than the number of Prolific users initially recruited. We expected the Pixel to perform better than CAPI as it cannot rely on cross-domain unique identifiers and only performs device fingerprinting based on IP addresses, which are often shared and/or dynamic, and user agents, which have various degrees of uniqueness. In addition, even if we ensured optimal conditions for the Pixel, we expected the reach effectiveness to be lower than 100%. First, for Pixel to link website visits to Facebook profiles, users need to be logged into Facebook on the browser they are taking the survey on. This might not be the case for all users, as some only log in from their phone, while others might take the survey on a professional computer on which they do not log into Facebook. Second, our ad might not reach users who are not active on Facebook during our ad campaign. To help interpret the results, our survey asked users "Do you log in to your Facebook account on a desktop device?" and "Are you logged in to your Facebook account on your current browser?"

Without first-party cookies: The CAPI showed higher reach effectiveness (41% to 51%) in comparison to the reach effectiveness of events recorded from the Pixel (43% to 46%); however, this improvement is not statistically significant.[2] The results on the Pixel are consistent with the percentage of users that report they log in to Facebook accounts from their desktop devices (52-58% on average across all the attempts). Since we implement the CAPI and the Pixel on the same website, we can leverage the overlap functionality on Meta's Ad Manager to compare how many users are matched with both CAPI and Pixel events. The overlap is (43% to 48%).

With first-party cookies: Similarly to the first setup, there are no statistically significant differences between the CAPI's reach effectiveness (45%) and that of the Pixel (42% to 44%). Additionally, our findings show that the *fbp* cookie does not improve the overall tracking effectiveness. This suggests that the *fbp* does not contain additional data that can be used to uniquely identify the visitor. However, we observe a significantly higher overlap (73% to 74%) between matched users with both Pixel and CAPI audiences. We further explore the role of *fbp* cookie in Section 7.1).

Takeaways: Under optimal conditions for the Pixel (i.e., with third-party cookies enabled and no tracking restrictions), server-side tracking implemented by Meta's Conversions API can achieve tracking effectiveness similar to that of client-side tracking through Meta's Pixel. This suggests that server-side tracking could outperform client-side tracking in the real world, as many users and browsers are adopting tracking protections that impact the Pixel. The overlap analysis indicates that about half of the users are identified by both tracking methods. Although the *fbp* cookie is a unique identifier, it did not improve the general tracking effectiveness. This suggests that Meta may not utilize its value to associate website visitors with their user profiles.

[1] These users are nevertheless paid on Prolific for their participation.

[2] We used two-proportion statistical significance test, specifically Z-test, to calculate the difference between the reach effectiveness results, with a confidence level of 95%.

4.2 Mobile Traffic Experiment

4.2.1 Instantiation. We maintain experimental conditions similar to those of the desktop traffic experiment.

User recruiting: We used device prescreening criteria to select users on mobile devices. We recruited 725 users across three experiments performed between April and October 2023.

Website verifications: We verify that participants access our websites on a mobile device and ensure that users are not using a VPN or employing filter lists that may affect Pixel behavior.

Website tracker implementation: We create a website where we integrate the Pixel and the CAPI. Participants answer the consent form and then the survey. We send the *Page View* event with all available data (IP address, device and browser information from HTTP headers and first and third-party cookies) and a *View Content* event via CAPI containing only IP address, user agent, and location data. We do not send the *fbp* cookie in the CAPI events (see Section 7.1 for a detailed exploration of the *fbp*).

4.2.2 Results. Our findings are presented in Table 1. We hypothesize that the Meta Pixel may exhibit suboptimal performance on mobile devices compared to desktops. To link users to their social media profiles, the Pixel needs users to be logged in to Facebook on their mobile browser. In this case, when the user visits a website that integrates a Meta Pixel, the script can load the cookies and include them in the event data. However, as users often log into their accounts through the Facebook app rather than their mobile browsers, events collected via the Pixel on mobile browsers are less likely to include Facebook's third-party cookies.

The Pixel has a reach effectiveness of 61%, while the CAPI achieves 34%. Since the first attempt yielded unexpectedly low results for CAPI compared to Desktop, we replicated the exact experimental setup to reproduce the outcome. In two other attempts, the reach effectiveness of CAPI exceeded 50%, bringing it closer to the results of the Desktop experiment.

Contrary to our expectations, Meta's Pixel achieved a significantly higher reach effectiveness on Mobile (61%) than Desktop (42%–46%). We have two hypotheses about this: (1) contrary to expected, users are logging in to Facebook in their mobile browsers (in addition or instead of using the Facebook app); (2) When third-party cookies are unavailable in the Meta Pixel events, the platform uses fingerprinting data collected by the Pixel from HTTP headers (such as IP address, OS, and screen resolution) to match website visitors to their accounts (we further test this hypothesis in Section 5.1). When users are logged in on their mobile devices, this data is already associated with their accounts. The higher effectiveness could be attributed to the number of logged-in users, which was lower in the Desktop experiment, with only 52-58% of participants reporting they accessed their Facebook accounts on their desktops, while 72%-92% of participants in the Mobile experiment reported they accessed their accounts on their mobile devices.

Takeaways: Server-side tracking effectiveness is similar on mobile and desktop devices. However, client-side traffic effectiveness is higher on mobile devices than on desktops. We expected it to be lower as it is less likely for Facebook's third-party cookies to be installed on mobile browsers as users generally login with the Facebook app.

5 IMPACT OF TRACKING RESTRICTIONS

This section explores the impact of tracking restrictions, including privacy-preserving browsers, filter lists, VPNs, and user agent spoofing tools, on the effectiveness of both the Pixel and the CAPI.

5.1 Safari Experiment

To investigate how trackers are affected by specific browser privacy enhancements, we focus on Safari. While there are multiple privacy-focused browsers on the market, such as Firefox, Brave, and DuckDuckGo, it is challenging to find and recruit a large enough number of participants who use these browsers. Safari serves as an interesting case study because it blocks third-party cookies by default, significantly impacting the effectiveness of client-side trackers. Additionally, Safari claims to hide the user's IP address from trackers through relays (this feature is enabled by default) [3, 27]. Moreover, for premium iCloud+ users, Safari offers a feature (iCloud+ Private Relay), which, in part, conceals the IP address not only from trackers but also from websites [26] (this feature is disabled by default). iCloud Private Relay differs from a VPN in that it only encrypts DNS requests made through the Safari browser. And is limited to masking the IP address without fully changing the location.

5.1.1 Instantiation. We ensured similar conditions to the desktop traffic experiment, with the exception that users be on Safari.

User recruiting: We used device prescreening criteria to select Prolific users on macOS devices. We recruited 1350 users across four campaigns between March and October 2023.

Website verifications: We verify that we only select participants accessing our website with Safari. We restrict submissions to participants without a VPN connection and without any type of filter list to ensure that the results are solely linked to Safari's privacy enhancements and not influenced by other factors.

Website tracker implementation: Similarly to the desktop experiment, we implemented two setups: one omitting first-party cookies (*fbp*) and the other including them. Recruited participants are redirected to a website where we integrated the Pixel and the CAPI. Once they submit the questionnaire on the website, we send a Page View event via the Pixel and a View Content event via the CAPI. The data and implementation are identical to the experiment in Section 4.1.

5.1.2 Results. Table 2 presents the reach effectiveness.

Without the first-party cookie: We enlisted 850 participants who met our criteria across three experiments. Table 2 shows that the reach effectiveness for both the Pixel and the CAPI varies each time we instantiate the setup (from 0% to 51% reach effectiveness for the Pixel, and from 0% to 19% reach effectiveness for the CAPI). We were unable to reproduce consistent results on Safari. When comparing the proportion of website visitors who reported logging into their Facebook accounts (53-61%), we found no statistically significant differences between each attempt, nor between these experiments and the desktop traffic experiments in Section 4.1. Additionally, as the third-party service we employed to detect VPN usage (VPNAPI.io) also identifies the use of private relays, we attempted to gather statistics on relay usage but found no evidence among any of our participants. This suggests that the CAPI receives the correct IP address for each participant (considering possible

Table 2: Reach effectiveness of Meta client and server-side tracking technologies in the presence of tracking restrictions.

Experiment	#	*fbp*	Users	Pixel	CAPI	Overlap
Desktop: Safari	1	No	250	0%	0%	0%
	2	No	300	51%	0%	0%
	3	No	300	11%	19%	24%
	4	Yes	500	41%	55%	60%
Desktop: VPN	1	N/A	500	N/A	27%	N/A

- **N/A** indicates that the tracker was not implemented on the website.
- For each experiment, we had multiple attempts with the same configuration, '#' refers to the attempt number.

inaccuracies in the service we employed). However, the pixel receives relay IP addresses. The first experiment (with the lowest reach effectiveness) was conducted in March 2023, while attempts #2 and #3 were performed in April/May and October/November 2023, respectively.

Nevertheless, since we were able to target users using the Pixel events from Safari, albeit with varying effectiveness, it implies that third-party cookies are important but not essential to the functioning of the Pixel, as we hypothesized in Section 4.2.

With the first-party cookie: We sent events from 500 website visitors that met our experiment criteria. The Pixel achieved a 41% reach effectiveness, while the CAPI achieved a 55% reach effectiveness. The effectiveness is similar to the Desktop experiment results (Section 4.1) where participants used Google Chrome. Therefore, the additional privacy protections provided by Safari do not seem to reduce user tracking in terms of effectiveness when both the Pixel and CAPI trackers are implemented and the first-party cookies are present. We further explore the role of *fbp* in Section 7.1.

Takeaways: Despite some discrepancies in our experimental results, there are two important observations: (1) the Pixel can match users in the absence of third-party cookies (probably relying on fingerprinting from the HTTP headers); (2) privacy-protections offered by Safari reduce but do not stop server-side or client-side tracking, especially when first-party cookies are used.

5.2 Filter Lists Experiment

We examine how filter lists affect the Meta Pixel and CAPI. This experiment did not involve a recruitment process. We hypothesize that for the Pixel, using any popular filter list (e.g., through an ad blocker, or a privacy-preserving browser) will block it from loading and prevent the installation of first and third-party cookies on the website. As for CAPI, since filter lists operate in the browser, they do not impact requests we make from the server to Meta (request 4 in Fig.2). Therefore, we test how filter lists impact the requests we make from the browser to the server, where we implement CAPI (request 3 Fig.2).

5.2.1 Instantiation. We created a website where we integrated the Meta Pixel and the CAPI. Integrating the Meta Pixel's third-party script is straightforward. For CAPI events, we set up two scenarios to test filter lists. The first scenario uses endpoints under the same domain, that is, the first-party domain. The second scenario uses endpoints under a different domain, that is, a third-party domain. In each scenario we set up six endpoints to simulate data being transmitted from the browser to the server and sent six requests from the browser to the server, one to each endpoint, simulating a website collecting data through CAPI. In both the first and third-party domains the six endpoints are 'basic/', 'analytics/', 'tracking/', 'conversion/', 'get_item/', and 'user/'.

By monitoring the website's network activity, we investigated instances where Pixel calls were blocked, as well as monitoring the cookies that were set or transmitted for each successful call. As for CAPI, we monitored occurrences of blocked calls made in each scenario.

5.2.2 Results. The Meta Pixel:

Extensions: As indicated in Section 2.1.1, the Pixel loads a library of functions in the browser, which can later be used to send tracking requests to Meta's marketing tools. However, the call used to load this library is blocked by both EasyList and EasyPrivacy, the most commonly used filter lists in ad blockers. The Pixel is also prevented from loading with private search extensions, such as DuckDuckGo and Qwant VIPrivacy (with the Protection feature enabled). Therefore, most ad block extensions directly block Pixel scripts from loading.

Browsers' built-in filters: Privacy-preserving browsers with default filters, such as Brave and DuckDuckGo, block the Pixel from loading. However, Safari does not block the Pixel by default, and we successfully sent Page View events. As for Firefox with default privacy features (Standard), the Pixel is loaded and tracking requests are executed correctly. In both Safari and Firefox, third-party cookies are unavailable, while first-party cookies are set and transmitted with Page View events correctly.

The Conversions API:

First-party requests scenario: All our attempts to send data from the browser to our website's server, where the CAPI is implemented, were unaffected by filters in ad blockers and private search extensions (e.g., AdBlock, Adblock Plus, Privacy Badger, uBlock Origin, Ghostery, DuckDuckGo, Qwant), as well as built-in browser filters on Brave and DuckDuckGo. This remained true even with requests routed through 'analytics/', 'conversion/' or 'tracking/', regardless of the method used (GET and POST).

Third-party requests scenario: None of the previously mentioned ad blockers and private search extensions blocked requests to the third-party endpoints. The same held for built-in browser filters on Brave and DuckDuckGo. One possible explanation for why filters do not block the requests, even when routed through conspicuous endpoints like 'tracking/', is that some websites can break if they do so. The calls remained unaffected regardless of the method used (GET or POST), and the endpoint name.

Takeaways: Filter lists can prevent client-side tracking by blocking known trackers from loading. Server-side tracking requests involve two steps. First, the browser sends a request to the server, which then sends a request to Meta. Filter lists only operate client-side and, therefore, cannot regulate outgoing requests from the website's server. However, our tests show that tracking requests sent from the browser to the server are currently not blocked by filter lists.

5.3 VPN Experiment

The purpose of this experiment is to test the effectiveness of the CAPI when receiving traffic from VPN connections. Ideally, we would direct traffic from participants who use VPNs to our website or request participants to turn on their VPNs when accessing it. However, since Prolific user guidelines prohibit the use of VPNs and proxies [4], we cannot do so. To address this, we leveraged a database of IP addresses linked to VPN servers sourced from ipinfo.io [20]. When we drive traffic from Prolific, instead of sending the participants' authentic IP addresses, we replace them with IP addresses that are known to be linked to VPN servers. We hypothesize that the user-matching system could flag IP addresses linked to VPN servers, and IP addresses provided by VPN services are less likely to be linked to a unique user. Hence, we expect the reach effectiveness to be low or close to zero.

5.3.1 Instantiation. User recruiting: We employ device prescreening criteria to select Prolific users only on desktops. We recruited 500 users, and the experiment was performed in November 2023.

Website verifications: We implement website verifications to only select users who access the website through a desktop device, using Google Chrome as their browser.

Website tracker implementation: We implement a single website that integrates the CAPI. For each participant, instead of sending the authentic IP address, we send an IP address linked to a known VPN server, the user agent, and IP address-based geolocation data derived from the VPN's IP address. It is possible to change the IP address sent through the CAPI because we control the server and the calls it produces. We deliberately opted out of integrating client-side tracking for this experiment to ensure we are not sending the user's authentic IP through other means. We directed traffic to our website from 500 participants whose authentic IP addresses were replaced with VPN-linked IP addresses.

5.3.2 Results. Results are shown in Table 2. The server-side tracker had a 27% reach effectiveness, less than experiments with authentic IP addresses, which had effectiveness between 44%-51%, but far from 0%. One hypothesis was that the matching tools may downplay the significance of the IP address when they identify VPN usage, giving more weight to the user agent. However, only 7% of our participants have unique user agents, which cannot explain a reach effectiveness of 27%. Alternatively, this might happen if the platform associates our website visitors with other users who accessed Facebook.com or similar sites implementing Meta trackers while browsing through the same VPN-linked address. These results made us push to find methods to assess the accuracy of the user matching in Section 6.

Takeaways: Our experiments show that even if a VPN is used, retargeting through server-side tracking is possible. Nevertheless, the experiment raises questions about the accuracy of retargeting.

5.4 User Agent Spoofing Experiment

This experiment aims to evaluate the effectiveness of server-side tracking when quasi-non-existing user agents (UA) are used, to understand the extent to which user agents play a role in server-side tracking. Participants are assigned real user agent strings dating before 2010, with new (randomly generated) strings not present in their authentic UA attached. This ensures that the attributed strings do not correspond to real browsers, and Meta does not have these user agents in its matching database.

Table 3: Reach effectiveness of server-side tracking technologies with spoofed User Agents. The overlap indicates users matched using spoofed user agents that were also matched using authentic user agents.

Experiment	Users	Authentic	Spoofed	Overlap
User Agent	250	44%	44%	82%

5.4.1 Instantiation. User recruiting: We used device prescreening criteria to select only users on desktop devices. We directed traffic from 250 users on Prolific.

Website verifications: We verify that we only select participants on desktops and that the participants access our website from a Google Chrome browser.

Website tracking implementation: We create a single website that integrates the CAPI. We send View Content events to two distinct tracking endpoints. Both endpoints receive the IP address of website visitors, their IP address-based geolocation data, and a user agent string. In the first endpoint, we send events containing the real user agents, while the second receives one of the spoofed user agents. To compare the tracking effectiveness of the two endpoints, we make sure to create an isolation between events. We do this by deliberately modifying the "source" of each event to create the appearance that they originate from two different websites.

5.4.2 Results. Our findings are shown in Table 3. We can see an equal number of users matched when using authentic user agents (44%), and a spoofed user agent (44%). This reach effectiveness is consistent with desktop experiments in 4.1. The high overlap indicates that 86% of users matched using spoofed user agents are also matched using authentic user agents. This suggests that IP addresses are sufficient to link website visitors to unique users, using the CAPI and that the user agent plays a smaller role than expected. It is also probable that IP addresses are considered the sole user identifier in our event data, and other information has lesser weight in the user matching process.

Takeaways: Using spoofed user agents does not impact tracking effectiveness; hence, IP addresses seem to be sufficient identifiers for CAPI to match users.

6 ACCURACY OF TRACKING TECHNOLOGIES

Previous experiments illustrate the effectiveness of Meta's CAPI and the Meta Pixel. Notably, the CAPI demonstrates the ability to successfully link a substantial portion of website visitors to user accounts on Meta products. However, given that server-side events are linked to user profiles based on IP address, user agent, and geolocation data, alone, it is possible that the matching process was inaccurate in some cases. For instance, users can switch between multiple IP addresses (e.g., home, work). Furthermore, multiple Internet users can share the same IPv4 address simultaneously (e.g., through Network Address Translation). We conduct an experiment specifically designed to assess the accuracy of user matching with IP address, user agent, and geolocation data that we send through the CAPI, and compare it to the Pixel's.

Table 4: User matching accuracy with Pixel and CAPI tracking. The overlap is the proportion of users matched by the CAPI that were also matched by the Pixel.

Implementation	Events	Reach	Accuracy	Overlap
Pixel (isolated)	812	63%	100%	58%
CAPI (isolated)	958	16%	60%	
Pixel	1722	61%	81%	52%
CAPI	1833	20%	65%	

6.1 Instantiation

User recruiting: As described in Section 3.4.2, we direct traffic to our website through an advertising campaign on Facebook instead of Prolific. Our campaign ran from October 25th to November 12th, 2023, and was promoting an extension developed by our team for research purposes CheckMyNews[3]. Visually, the website we advertised is designed to showcase the extension. When the user visits our website, a consent banner with a privacy notice pops up. This banner explains the purpose of the website and that by consenting, the user will possibly be targeted by another ad campaign.

We used a broad target audience for our advertising campaign; we did not set any country, demographic, or interest, and we only excluded users who access their accounts on Safari and iOS devices. In total, our recruiting campaign reached 524k unique users and received 11k link clicks. A total of 2,791 users consented and took part in this experiment.

Website verifications: We trigger events only for users that: (1) do not use filter lists; (2) do not have active VPN connections; and (3) do not come from browsers that disable third-party cookies (Firefox, Safari, Brave, Opera).

Website tracking implementation: In the effectiveness experiments, we opted for isolated tracking endpoints, which means that the events we send from the Pixel are sent to one endpoint, and the events from CAPI to another (see Section 3.3). This was to ensure the platform was not linking event data coming from the same website from users that have the same IP address and user agent, which would compromise the validity of our experiments. In this experiment, we test this hypothesis.

In the first setup, as for the effectiveness experiments, the Pixel and CAPI events are linked to separate endpoints (with distinct dataset IDs that are sent to the event). In the second setup, both the Pixel and the CAPI events are linked to the same endpoint. For each website visitor, we send two distinct event types: a Page View event through the Pixel, which contains first- and third-party cookies and browser and device information from HTTP headers, and a View Content event through the CAPI, containing IP address, user agent, and location data. Events are triggered simultaneously after verifying experiment conditions and obtaining user consent. For the separate endpoints setup, to ensure perfect isolation, we also altered the source URL sent with the events to make it appear that events are sent from different websites.

6.2 Results

Our results are presented in Table 4. The table presents the number of events received from the two trackers, their reach effectiveness, and the accuracy of the matching. As described in Section 3.4.2, the accuracy is measured by the overlap between users reached with the tracking ad campaign and the users reached with the recruiting ad campaign. In a scenario of perfect user matching, the tracking ad campaign should exhibit a 100% overlap with the recruiting ad campaign (as all users who visited and were tracked on our website were users who saw our Facebook ad).

There is a discrepancy in the number of events received between the server and the client-side trackers. We observed that most users in the accuracy experiments have IP addresses corresponding to developing countries (contrary to the effectiveness experiments where we recruited users in the U.S.). As expected, the server-side calls are consistently recorded but it's possible that slow or unstable connections in these countries could impede the Pixel from loading or from sending events properly.

Separated Endpoints: 63% of website visitors tracked with the Pixel, and 16% of website visitors tracked with CAPI are matched to user accounts. The CAPI reach effectiveness is much lower than what we previously observed (e.g., Table 1). This discrepancy might be again due to the fact that in this experiment, the majority of users come from developing countries (contrary to the effectiveness experiments where we recruited users in the U.S.), which might affect how unique their ‹IP address + user agent› fingerprints are. In particular, these countries have much fewer IPv4 addresses [35].

In terms of accuracy, as illustrated in the Overlap column, 100% of users matched with the Pixel had previously visited our website. On the contrary, only 60% of users matched with the CAPI had previously visited our website. Hence, server-side tracking, as implemented through CAPI, leads to false matches. Given these results, advertisers should not expect exclusive targeting of users who have visited their websites if they only collect IP addresses, user agents, and geolocation data, since 40% of the users that we matched had never actually been to our website.

Common Endpoint: The tracking effectiveness for Pixel and CAPI is similar in the single endpoint and isolated endpoint setups. However, the accuracy of the Pixel decreases from 100%, in the isolated setup, to 81% in the single endpoint setup. This suggests sharing the endpoint can impact how both Pixel-based and CAPI-based user matching is performed, and that the platform may aggregate events occurring simultaneously, originating from the same website and sharing the same IP address. This is also problematic for advertisers since the standard implementation uses both the Pixel and the CAPI with a single endpoint.

Takeaways: Our results indicate that while the Pixel can achieve 100% matching accuracy via third-party cookies, server-side tracking relying on IP addresses, user agents, and IP address-based geolocation only achieves an accuracy range of 60% to 65%. This implies that the Conversions API, as currently implemented, produces false matches for more than a third of website visitors. Advertisers employing server-side tracking with only fingerprinting data face the risk of misspending one-third of their retargeting ad budget.

[3] CheckMyNews is an extension that captures news-related content on Facebook and presents statistics on it.

7 THE ROLE OF FIRST PARTY COOKIES

First-party cookies are pieces of information stored in the user's browser under the domain they are visiting. Unlike third-party cookies, these identifiers are not accessible to other websites; only the one being visited can access them. However, as more restrictions are introduced to third-party tracking, first-party cookies are increasingly being utilized as a means to track users' activity online [32]. For instance, the Meta Pixel allows the installation of first-party cookies under the advertiser's website domain on the visitor's browser. These cookies are known as the Facebook browser ID or *fbp* and the Facebook click ID or *fbc*.

In their privacy terms, Meta claims to use first-party cookies to track the user's activity on the website and report user actions following ad impressions; however, they do not specify the role of first-party cookies in matching website visitors to user accounts on their products. Prior research has proposed how website visitors, in theory, can be matched to their Facebook profiles through the *fbc* value [29]; however, they did not show that Meta is doing this in practice. We propose a set of experiments to test the role of the *fbp* and *fbc* cookies in linking website visitors to user profiles.

7.1 Facebook Browser ID (fbp)

The Facebook browser ID (*fbp*) is a first-party cookie installed by the Meta Pixel under the domain of the visited website. Referred to as *fbp*, this value is as a unique identifier created by the Meta Pixel SDK. The *fbp* has a lifespan of 90 days as long as the user does not close the browser [17]. From the previous experiments with desktop traffic (Section 4.1), we have seen that the use of the *fbp* did not improve the general tracking effectiveness of server-side events, indicating that while this cookie's value is unique, it is not necessarily used to match visits to unique users. However, the overlap between the CAPI and Pixel events that shared the *fbp* value was much higher than events that did not share it. This suggests that the use of this identifier may, however, link event data. To test this hypothesis, we implement the following experiment.

7.1.1 Instantiation. User recruiting: Aside from the details specified in Section 3.1, we used device prescreening criteria to select Prolific users only on desktops. We recruited 500 users in April 2023.

Website verifications: Identical to the Desktop experiment in 4.1.

Website tracking implementation: We deploy a single website and integrate three trackers: two server-side trackers implemented with the CAPI that link to distinct tracking event endpoints and a client-side tracker implemented with the Pixel, whose sole purpose is to create and install the first-party cookies in the browser.

We send two server events for each participant: the first contains the visitor's authentic IP address, their user agent, their IP address-based geolocation data, and the *fbp* cookie's value installed by the Pixel. The second event contains a randomly generated public IP address (spoofed address) different from the visitors' authentic IP address, the visitor's user agent, the IP address-based geolocation data from the spoofed address, and the authentic *fbp* cookie's value. We verify that the spoofed IP address is geolocated to the United States. Finally, we made sure that the CAPI events seemed to come from different websites by altering the source URLs of the events we sent.

Table 5: Analysis of the use of the *fbp* cookie. The reach effectiveness for events sent to Endpoint 1 (authentic IP addresses) and Endpoint 2 (spoofed IP addresses). The overlap shows the proportion of users matched through both.

Cookie	Users	Endpoint 1	Endpoint 2	Overlap
fbp	500	45%	28%	86%

The purpose of this setup is to test if the *fbp* is used to aggregate event data coming from the authentic ‹IP address + user agent› and the event corresponding to the spoofed ‹IP address + user agent›. Without the *fbp*, the advertising platform cannot link the event with the spoofed ‹IP address + user agent› to the user who visited our website.

7.1.2 Results: Our findings are presented in Table 5. Endpoint 1 shows the tracking effectiveness of CAPI with authentic ‹IP address + user agent›, while Endpoint 2 shows the effectiveness of CAPI with spoofed ‹IP address + user agent›. Meta indicates that there is an 86% overlap between the audience generated with the spoofed IP address and the audience generated with the authentic IP address. The overlap should be close to zero if only the ‹IP address + user agent› information in the event is used. Hence, *fbp* is used to link the events corresponding to the spoofed and the authentic IP address, suggesting that *fbp* is used to link together the different ‹IP address + user agent› fingerprints of a user that can later be used for matching new website visits.

Takeaways: Prior experiments in Section 4.1 show that the *fbp* cookie did not improve the general tracking effectiveness of server-side events, indicating that the cookie appears not to have additional data that can directly be used to match a website visit to actual user profiles on Meta. Here, we show that *fbp* can be used to link together the data from different events such as linking the different ‹IP address + user agent› fingerprints of a user, that can later be used for matching new website visits.

7.2 Facebook Click ID (fbc)

When a user clicks on an ad on Facebook, Instagram, or Messenger, a query parameter is sometimes appended to the link. If the advertiser implements a tracking Pixel on their website, when the user lands on it, the Pixel creates a first-party cookie, called the *fbc* cookie, with the query parameter value. When tracking events containing the *fbc* value, Meta's reporting tools accurately attribute the actions such as purchases to events originating from their platform.

The *fbc* cookie is not stated to be unique in the Meta developer documentation [17], but one hypothesis could be that since the URL parameter used to create the *fbc* (called the FBCLID) is generated on the platform, it can be used as a unique user identifier. Our objective is to investigate whether the *fbc* results in a higher user matching rate. Such an outcome would suggest that the *fbc* establishes a direct link between website visitors and their user profiles.

Since this experiment requires website visitors to be directed from their Facebook accounts to our website via ad campaigns on Meta, we set it up alongside the accuracy experiment in Section 6.

7.2.1 Instantiation. Website traffic: Same as Section 6.

Website verifications: Same as Section 6.

Table 6: Analysis of the use of the *fbc* cookie. The Overlap indicates users matched with the CAPI endpoints and the Pixel endpoint.

	Users	Reach	Accuracy	Overlap w/ Pixel
Pixel	1722	61 %	82%	
CAPI w/ *fbc*	1833	23%	70%	75%
CAPI w/o *fbc*	1833	20%	65%	52%

Website tracking implementation: On the same website where we conducted the accuracy experiment, we implemented an additional CAPI tracker linked to a new event endpoint. We then sent events to Meta, containing each visitor's IP address, user agent, IP address-based geolocation data, and the value of the *fbc* cookie.[4] This allows us to compare the tracking effectiveness and accuracy of server-side tracking with and without sending the *fbc* cookie.

7.2.2 *Results.*

Our results are presented in Table 6. The table demonstrates that *fbc* contributes to improving tracking effectiveness (from 20% to 23%) and accuracy (from 65% to 70%). Additionally, we observe a higher overlap between users matched with the Pixel and the CAPI with *fbc* (75%) compared to users matched with CAPI without *fbc* (52%). There are two possible explanations for these results: first, the *fbc* cookie may contain additional data that can uniquely identify visitors. The second explanation is that the cookie is used to link events matched by the Pixel to events lacking sufficient data for matching by the CAPI, which would explain the high overlap. This linkage occurs regardless of whether or not *fbc* is used for matching users to profiles.

Takeaways: The *fbc* value seems to improve the effectiveness and accuracy of matching website visitors to social media profiles. Similarly to the *fbp*, the *fbc* cookie is used to establish connections between the data gathered via different sets of events.

8 RELATED WORKS

Prior studies on tracking systems, predominantly focused on client-side technologies, have examined the prevalence of third-party tracking online, mostly via network crawling [30, 37, 38]. For instance, research shows the high adoption rate of the Meta Pixel in the top 10K websites listed on Alexa, with an average of 23.4% websites using the tracker [29, 33]. Researchers have also questioned the use of first-party cookies set by third-party tracking scripts as a way to circumvent tracking restrictions by browsers. For instance, Chen el al. [32] show what is called first-party "Cookie Leakage" where agreements and partnerships between different companies are established to map user IDs across multiple systems, enabling data exchange across different third-party platforms. Bekos et al. [29] delved into the possibility of harvesting internet users' browsing history via the *fbc* and *fbp* cookies created by the Meta Pixel.

Additionally, research on fingerprinting and tracking [42] has shown that user agent strings combined with IP addresses can identify hosts with a precision[5] of 80% from a dataset including hundreds of millions of users across the global IP address space, and that the entropy[6] of the user agent combined with the IP address is higher than the entropy of more complex browser fingerprints (e.g., that combine screen resolution, timezone, system, fonts, and user agents) that exclude the IP address. These results confirm our findings regarding the ability to track and identify users using their IP address and user agent.

Unlike previous studies that focused on examining the mechanisms of online tracking, the data that is collected, and how it is shared between third parties, the focal point of our research is to measure how *effectively* can this data be used to match users to profiles on advertising platforms, a point overlooked by most research in the field. Additionally, while previous research heavily focuses on client-side trackers, our research studies server-side tracking, a technology currently being pushed by many advertising platforms, including Meta and Google, as all major browsers move to disable third-party cookies by default [2] and partition their caches [22].

9 CONCLUSION

Motivated by the shift towards server-side tracking, as client-side trackers face numerous restrictions imposed by browsers, our study compares the effectiveness and accuracy of Meta's client and server-side tracking technologies through a series of innovative experiments. Our results show that, in the absence of tracking restrictions, server-side tracking is comparable to client-side tracking on Meta. However, the current implementation of Meta's server-side tracking suffers, to a large extent, from false matches when provided with minimal data (IP address, user agent, and IP address-based location). Given that server-side connections are not visible to browsers, discerning which third parties are receiving data from the website through server-side tracking is challenging. In comparison, it is significantly easier to verify a website's compliance with privacy regulations like the GDPR when client-side trackers are used.

In an era marked by heightened concerns about online privacy and evolving regulations, our study sheds light on the challenges and opportunities inherent in user tracking via server-side tracking methods and emphasizes the critical need for transparent and compliant practices in the evolving landscape of online data privacy.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their helpful comments. This research was supported in part by the French National Research Agency (ANR) ANR-21-CE23-0031-02, ANR-22-CE38-0017, ANR-21-CE39-0019 and ANR-22-PECY-0002 grants, and by the EU 101041223 and 101021377 grants.

[4] Note that the value of the *fbc* cookie is sent by default with Pixel events.

[5] The percentage of fingerprints that correspond to one host (one hardware ID).

[6] Entropy measures randomness and uncertainty. High entropy means data is spread out, while low entropy means data is concentrated.

REFERENCES

[1] 2017. User-Agent Switcher – Get this Extension for Firefox. https://addons.mozilla.org/en-US/firefox/addon/uaswitcher/

[2] 2019. Effect of disabling third-party cookies on publisher revenue. https://services.google.com/fh/files/misc/disabling_third-party_cookies_publisher_revenue.pdf

[3] 2019. Safari Privacy Overview. https://www.apple.com/safari/docs/Safari_White_Paper_Nov_2019.pdf

[4] 2021. Why participants get banned on Prolific. https://www.prolific.com/blog/why-participants-get-banned

[5] 2022. Adblocking penetration rate in selected countries/territories worldwide as of 3rd quarter 2022. https://www.statista.com/statistics/351862/adblocking-usage/

[6] 2022. How We're Protecting Your Online Privacy - The Privacy Sandbox. https://privacysandbox.com/open-web/

[7] 2023. AdBlock: Introduction to Filter Lists. https://helpcenter.getadblock.com/hc/en-us/articles/9738523403027-Introduction-to-Filter-Lists

[8] 2023. Brave Shields. https://brave.com/shields/

[9] 2023. *Conversions API - Documentation.* https://developers.facebook.com/docs/marketing-api/conversions-api/

[10] 2023. The EasyList filter lists. https://easylist.to/

[11] 2023. Get Started with the Meta Business SDK. https://developers.facebook.com/docs/business-sdk/getting-started/

[12] 2023. Information processed by Meta for people who don't use Meta Products. https://www.facebook.com/help/637205020878504/?helpref=related_articles

[13] 2023. ipinfo: IP address data provider. https://ipinfo.io/

[14] 2023. IPQUALITYSCORE: Fraud Detection & Bot Detection Solutions. https://www.ipqualityscore.com

[15] 2023. Meta Business Help Centre: About Meta pixel. https://en-gb.facebook.com/business/help/742478679120153

[16] 2023. Meta Business Help Centre: Compare Conversions API setup options. https://www.facebook.com/business/help/433493041367251?id=818859032317965

[17] 2023. Meta for developers: fbp and fbc Parameters. https://developers.facebook.com/docs/marketing-api/conversions-api/parameters/fbp-and-fbc/

[18] 2023. Meta for developers: Meta Pixel. https://developers.facebook.com/docs/meta-pixel/

[19] 2023. Parameters - Conversions API - Documentation. https://developers.facebook.com/docs/marketing-api/conversions-api/parameters

[20] 2023. Privacy detection database. https://ipinfo.io/products/anonymous-ip-database

[21] 2023. Prolific: a crowdsourcing platform. https://www.prolific.com/

[22] 2023. State Partitioning in Firefox. https://developer.mozilla.org/en-US/docs/Web/Privacy/State_Partitioning

[23] 2023. Third-party cookies and Firefox tracking protection. https://support.mozilla.org/en-US/kb/third-party-cookies-firefox-tracking-protection

[24] 2023. Turn "Do Not Track" on or off. https://support.google.com/chrome/answer/2790761?hl=en&co=GENIE.Platform%3DDesktop

[25] 2023. VPN & Proxy Detection API. https://vpnapi.io/

[26] 2024. About iCloud Private Relay. https://support.apple.com/en-us/102602

[27] 2024. Apple Legal - Legal - Safari & Privacy - Apple. https://www.apple.com/legal/privacy/data/en/safari/

[28] 2024. Create a Custom Audience from website events. https://en-gb.facebook.com/business/help/666509013483225

[29] Paschalis Bekos, Panagiotis Papadopoulos, Evangelos P. Markatos, and Nicolas Kourtellis. 2023. The hitchhiker's guide to Facebook web tracking with invisible pixels and click IDs. In *Proceedings of the ACM Web Conference 2023.* ACM, New York, NY, USA, 2132–2143.

[30] Reuben Binns, Ulrik Lyngs, Max Van Kleek, Jun Zhao, Timothy Libert, and Nigel Shadbolt. 2018. Third Party Tracking in the Mobile Ecosystem. In *Proceedings of the 10th ACM Conference on Web Science* (Amsterdam, Netherlands) *(WebSci '18)*. Association for Computing Machinery, New York, NY, USA, 23–31. https://doi.org/10.1145/3201064.3201089

[31] Anthony Chavez. 2022. Expanding testing for the Privacy Sandbox for the Web. https://blog.google/products/chrome/update-testing-privacy-sandbox-web/

[32] Quan Chen, Panagiotis Ilia, Michalis Polychronakis, and Alexandros Kapravelos. 2021. Cookie Swap Party: Abusing First-Party Cookies for Web Tracking. In *Proceedings of the Web Conference 2021* (Ljubljana, Slovenia) *(WWW '21)*. Association for Computing Machinery, New York, NY, USA, 2117–2129. https://doi.org/10.1145/3442381.3449837

[33] Quan Chen, Panagiotis Ilia, Michalis Polychronakis, and Alexandros Kapravelos. 2021. Cookie swap party: Abusing first-party cookies for web tracking. In *Proceedings of the Web Conference 2021.* ACM, New York, NY, USA, 2117–2129.

[34] Salim Chouaki, Islem Bouzenia, Oana Goga, and Beatrice Roussillon. 2022. Exploring the Online Micro-Targeting Practices of Small, Medium, and Large Businesses. *Proc. ACM Hum.-Comput. Interact.* 6, CSCW2, Article 378 (nov 2022), 23 pages. https://doi.org/10.1145/3555103

[35] IP2Location. 2023. Internet IP Address 2023 Report. https://www.ip2location.com/reports/internet-ip-address-2023-report

[36] IP2Location. 2023. IP Geolocation API. https://www.ip2location.io/

[37] Jonathan R. Mayer and John C. Mitchell. 2012. Third-Party Web Tracking: Policy and Technology. In *2012 IEEE Symposium on Security and Privacy.* 413–427. https://doi.org/10.1109/SP.2012.47

[38] Franziska Roesner, Tadayoshi Kohno, and David Wetherall. 2012. Detecting and defending against third-party tracking on the web. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation.* USENIX Association, USA, 12.

[39] Brave Software. 2020. OK Google, don't delay real browser privacy until 2022. https://brave.com/ok-google/

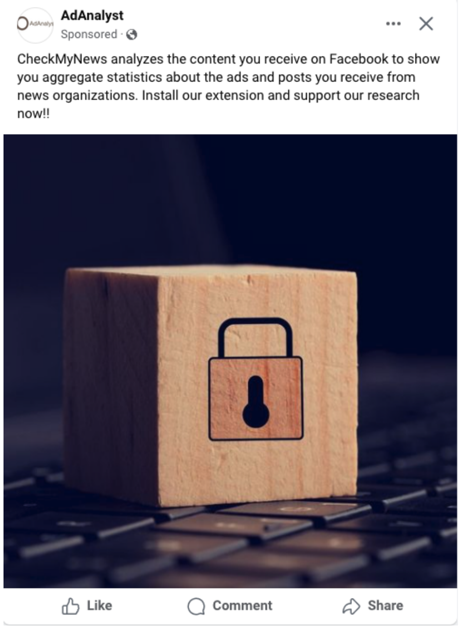
[40] John Wilander. 2017. Intelligent Tracking Prevention. https://webkit.org/blog/7675/intelligent-tracking-prevention/

[41] John Wilander. 2020. WebKit: full Third-Party Cookie Blocking and More. https://webkit.org/blog/10218/full-third-party-cookie-blocking-and-more/

[42] Ting-Fang Yen, Yinglian Xie, Fang Yu, Roger Peng Yu, and Martín Abadi. 2012. Host Fingerprinting and Tracking on the Web: Privacy and Security Implications. In *Network and Distributed System Security Symposium.* https://api.semanticscholar.org/CorpusID:12389538

A APPENDICES

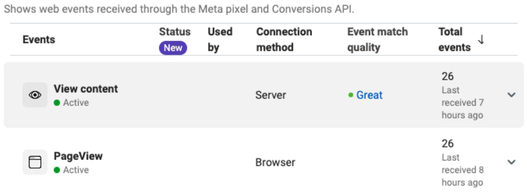
Figure 4: Screenshot of the content of the ads we used to measure the effectiveness of Meta tracking tools, as they appear on the Facebook Feed of users.



AdAnalyst
Sponsored ·

CheckMyNews analyzes the content you receive on Facebook to show you aggregate statistics about the ads and posts you receive from news organizations. Install our extension and support our research now!!

Figure 5: Screenshot of events count from Meta's Event Manager tool: example for events sent to the same event endpoint.



Shows web events received through the Meta pixel and Conversions API.

Events	Status New	Used by	Connection method	Event match quality	Total events ↓
View content ● Active			Server	● Great	26 Last received 7 hours ago
PageView ● Active			Browser		26 Last received 8 hours ago

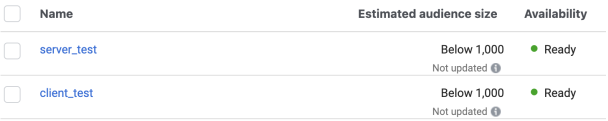
Table 7: Examples of cookies installed by Meta's client-side trackers used for advertising.

Cookies	Domain	Content
c_user	Facebook	The Facebook account ID of the user currently connected on the browser. Lifespan of 1 year.
fr	Facebook	Stores Facebook account details. Used for ad delivery and improving ads relevancy. Lifespan of 90 days.
sb	Facebook	Used to store browser details. Lifespan of 373 days.
_fbp	First-party	Facebook Browser ID. A **unique ID** saved under the website domain when the user first visits the website. Used for advertising and site analytics. Lifespan of 90 days.
_fbc	First-party	Facebook Click ID. When users click on ads on Facebook, the link includes a "fbclid" query parameter. When users land on the target website, the Meta Pixel automatically this query parameter to the _fbc cookie. It is used to report actions, such as purchases, generated through Facebook ads. Lifespan of 90 days.
usida	Facebook	A session cookie that collects a combination of the user's browser and unique identifiers. Used for tailored advertising.
oo	Facebook	Used for Facebook advertisement and behavioral targeting. Lifespan of 5 years.

Table 8: Survey questions presented to participants on experiments' websites.

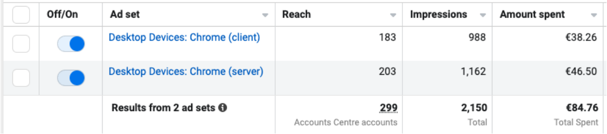
Question	Possible answers
Do you have a Facebook account?	Yes No
Do you have an Instagram account?	Yes No
How often do you log in to your Facebook account?	daily basis (at least once a day). Weekly basis (at least once a week). Monthly basis (at least once a month). Never
How often do you log in to your Instagram account?	Daily basis (at least once a day). Weekly basis (at least once a week). Monthly basis (at least once a month). Never
Do you log in to your Facebook account on a desktop device?	Yes No
Are you logged in to your Facebook account on your current browser ?	Yes No

Figure 6: Screenshot of the estimations for custom audiences' size as shown in the Audiences tool on Meta's Business Manager.



□	Name	Estimated audience size	Availability
□	server_test	Below 1,000 Not updated	● Ready
□	client_test	Below 1,000 Not updated	● Ready

Figure 7: Screenshot for ad campaigns' reach (results for Section 4.1) as shown in the Ad Manager tool on Meta's Business Manager.



□	Off/On	Ad set	Reach	Impressions	Amount spent
□	●	Desktop Devices: Chrome (client)	183	988	€38.26
□	●	Desktop Devices: Chrome (server)	203	1,162	€46.50
		Results from 2 ad sets	299 Accounts Centre accounts	2,150 Total	€84.76 Total Spent