POLITECNICO DI TORINO
Repository ISTITUZIONALE

The Rise and Fall of Google's Privacy Sandbox

Original
The Rise and Fall of Google's Privacy Sandbox / Grib, R.Y., Verna, A., Jha, N., Trevisan, M., Mellia, M.. - ELETTRONICO. - (In corso di stampa). (ACM Conference on Computer and Communications Security The Hague (NL) November 15-19, 2026).

Availability:
This version is available at: 11583/3012614 since: 2026-07-02T09:51:23Z

Publisher:
Association for Computing Machinery

Published
DOI:

Terms of use:

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

Publisher copyright
ACM postprint/Author's Accepted Manuscript

(Article begins on next page)

15 July 2026

This work has been accepted for publication at ACM CCS 2026. Citation and DOI information will be updated as soon as they are available.

The Rise and Fall of Google's Privacy Sandbox

Rachid Youssef Grib
Politecnico di Torino
Torino, Italy
rachidyoussef.grib@studenti.polito.it

Alberto Verna
Politecnico di Torino
Torino, Italy
alberto.verna@polito.it

Nikhil Jha
Politecnico di Torino
Torino, Italy
nikhil.jha@polito.it

Martino Trevisan
University of Trieste
Trieste, Italy
martino.trevisan@dia.units.it

Marco Mellia
Politecnico di Torino
Torino, Italy
marco.mellia@polito.it

Abstract

On October 17th, 2025, Google announced the retirement of most Privacy Sandbox APIs, concluding nearly five years of experimentation with its alternative to privacy-invasive data collection on the Web. Designed to balance privacy with advertising functionality and cross-site tracking, the initiative faced repeated redesigns and limited ecosystem support.

In this work, we present the first longitudinal, consent-aware measurement of the Privacy Sandbox's deployment across the Web. Using a custom call listener and weekly crawls of the top-10,000 websites, we monitor the usage of all major APIs in the months preceding their retirement. Adoption had already stagnated well before Google's announcement: most APIs were used by only a handful of actors, whose activity declined steadily throughout our study. Even the APIs that Google plans to maintain show no sign of growth. The sole exception is Cookies Having Independent Partitioned State (CHIPS). Overall, the demise of the Privacy Sandbox leaves unresolved the challenge of enabling privacy-preserving interest-based advertising.

Keywords

Web privacy, Web measurement, Privacy Sandbox

ACM Reference Format:
Rachid Youssef Grib, Alberto Verna, Nikhil Jha, Martino Trevisan, and Marco Mellia. 2018. The Rise and Fall of Google's Privacy Sandbox. In *Proceedings of ACM Conference on Computer and Communications Security (ACM CCS '26).* ACM, New York, NY, USA, 14 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Online tracking underpins today's digital economy by enabling extensive profiling of users based on their behaviour, interests, and browsing patterns. These profiles fuel the targeted advertising ecosystem and sustain much of the content and services available on the Web [49, 61]. However, the traditional mechanisms that support this ecosystem, most notably third-party cookies and browser fingerprinting [12, 55], enable pervasive cross-site tracking, raising long-standing privacy concerns and attracting increasing regulatory scrutiny.

In response, browser vendors have begun deploying countermeasures, including blocking third-party cookies by default [52, 62] and proposing privacy-preserving replacements [4, 7, 13, 14, 27, 51]. These efforts expose a structural tension: while privacy regulation and user expectations increasingly constrain unrestricted tracking, targeted advertising remains a central economic driver of the Web.

Google's Privacy Sandbox represents the most ambitious attempt to resolve this tension through a browser-enforced redesign of advertising and tracking primitives. Rather than exposing stable identifiers or unrestricted browser state (e.g., via persistent third-party cookies), the Privacy Sandbox proposes a collection of APIs that move sensitive processing (e.g., user profile creation, ad selection, and reporting) from remote servers to the user's browser. The browser acts as a trusted enforcement point, locally executing privacy-sensitive logic and only releasing limited, aggregated or noisy signals to third parties on a controlled basis.

Since its first announcement in 2019, the Privacy Sandbox has undergone repeated redesigns, public testing phases, and changes in deployment strategy [9]. APIs reached general availability between 2023 and 2024 [10], originally intended to coincide with Chrome's phase-out of third-party cookies [16]. However, in mid-2024, Google announced that third-party cookies would remain enabled by default under a user-choice model [40]. On 17 October 2025, Google declared that most Privacy Sandbox APIs would be retired in the following months [11], with deprecation in Chrome M144 (January 2026) and removal in M150 (July 2026) [41].

This sequence of reversals raises a central question: *how was Privacy Sandbox actually used in the Web ecosystem, and how did their adoption evolve during their rise and subsequent downfall?* Prior studies have examined individual APIs—most notably Topics and Protected Audience [2, 3, 5, 6, 8, 42, 43, 45, 46, 48, 54, 57–60]. However, no existing study provides a unified, ecosystem-wide, and longitudinal view of Privacy Sandbox usage. Moreover, prior measurements do not account for the role of consent banners and management platforms (CMPs), which critically regulate third-party components execution and may strongly impact when and how these APIs are invoked.

In this paper, we present the first large-scale, longitudinal, and consent-aware measurement of Privacy Sandbox deployment across

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ACM CCS '26,
© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/XXXXXXX.XXXXXXX

the Web. Using a custom Chrome-based crawler equipped with a comprehensive API call interceptor, we monitor the usage of the ten Privacy Sandbox APIs that reached general availability. Our methodology explicitly distinguishes between pre-consent, post-consent, and explicit consent-denial execution, allowing us to characterise both intended usage and consent-inconsistent behaviour [59, 60].

We conduct weekly crawls of the top-10,000 websites over six months, capturing the evolution of API adoption over time and the third-party ecosystem responsible for their deployment. Our analysis reveals that the contraction of the Privacy Sandbox ecosystem began well before its formal retirement. With the notable exception of partitioned cookies (CHIPS—Cookies Having Independent Partitioned State), adoption remained highly concentrated among a small number of actors and steadily declined throughout our measurement period. Even the APIs that Google plans to maintain exhibit no signs of growth or ecosystem momentum.

Taken together, our findings depict that Google's proposal of a browser-mediated privacy architecture has failed to achieve sustained adoption at Internet scale. More importantly, the demise of the Privacy Sandbox leaves unsolved the broader challenge of enabling privacy-preserving, interest-based advertising on the Web and highlights the difficulty of deploying technically complex privacy mechanisms that require substantial changes to entrenched advertising workflows: third-party cookies (and tracking) remain the major technology to fuel the online ad ecosystem.

This paper makes the following contributions.

- We provide the first longitudinal, consent-aware measurement of Privacy Sandbox API usage across the Web.
- We characterise the adoption dynamics of both retired and maintained APIs, showing that ecosystem disengagement preceded Google's deprecation decision.
- We demonstrate that partitioned cookies (CHIPS) are the only component to achieve widespread deployment, with 50% of sites using them. Yet, 75% of websites still use classical persistent unpartitioned third-party cookies.
- We document questionable API invocations even when no consent is provided, shedding light on the complex interaction between consent enforcement and actual technical implementation.

The rest of the paper is organised as follows: Section 2 introduces the Privacy Sandbox initiative and the APIs it includes, while Section 3 provides an overview of related work. Section 4 details the data collection methodology while Section 5 introduces the main results. Next we drill down our investigation: we witness the adoption of CHIPS (or partitioned cookies) in Section 6, while in Section 7 and Section 8 we present the results for retired and maintained APIs, respectively. Section 9 discusses questionable usage of the Privacy Sandbox APIs from the consent point of view. Finally, Section 10 concludes the paper.

2 Privacy Sandbox primer

The Privacy Sandbox is Google's umbrella initiative to redesign key Web functionalities related to personalised advertising, cross-site tracking, and identity without relying on third-party cookies or other stable cross-site identifiers. Instead of allowing unrestricted access to browser state to third parties, the initiative introduces a set of browser-mediated APIs that constrain data access and relocate privacy-sensitive computation, such as profiling, ad selection, and measurement, into the user's browser. A core design principle of the Privacy Sandbox is that the browser acts as the primary enforcement point for privacy. Sensitive operations are executed locally, and only limited, aggregated, or noise-added outputs are released to remote endpoints. In principle, this architecture aims to preserve advertising functionality while reducing the ability of third parties to perform cross-site tracking.

In this work, we focus on the ten APIs that reached general availability in 2023–2024, summarised in Table 1 and grouped according to the functional domains defined by Google.[1]

2.1 Advertising and measurement APIs

Attribution Reporting API [17, 18]. It enables privacy-preserving measurement of *ad conversions* by linking user interactions (e.g., ad clicks or views) with subsequent actions on an advertiser's site without exposing cross-site identifiers. The browser generates event-level or aggregated reports locally and releases only limited, noise-added signals to reporting endpoints.

Topics API [38, 39]. Provides a privacy-preserving mechanism for *interest-based advertising*, allowing advertisers to obtain the interests of a user without learning about the specific websites visited. It represents the evolution of the discontinued FLoC proposal [24]. The browser locally derives coarse-grained *topic identifiers* based on recent browsing history. During ad selection, the browser discloses some of the topics to the advertiser, occasionally introducing random noise to limit traceability.

Protected Audience API [32, 33]. Formerly *FLEDGE*, it enables privacy-preserving ad *re-targeting*, i.e., the ability of displaying ads about websites previously visited. When a user visits a website, advertisers can assign them to an *interest group*, representing the site or a product. During ad rendering, the browser runs an auction among eligible interest groups and fetches the winning ad. Auction reports and related metadata are generated by the browser and released under controlled channels.

Private Aggregation API [28, 29]. It enables the generation of aggregated statistics based on client-side data, without revealing individual user information. It is designed to be used in combination with other APIs (Protected Audience and Shared Storage) to compute metrics such as ad reach and auction winning rate while preserving user anonymity.

2.2 Storage, identity, and partitioning APIs

Shared Storage API [36, 37]. This API introduces a new form of shared browser storage that can be accessed across multiple domains. To prevent data leakage, the stored data can only be accessed from a secure JavaScript environment known as a *worklet*. This design enables privacy-preserving use cases, such as frequency capping or A/B testing, without exposing raw user-level data to external scripts.

[1] https://privacysandbox.com/intl/en_us/open-web/#the-privacy-sandbox-timeline, accessed on July 1, 2026.

Table 1: Summary of Privacy Sandbox APIs, grouped by domain. The start quarter refers to the opening period of the first public testing phase.

API		Scope	Start	Depr.	Restricted	Ref.	Rel. work
Attribution Reporting		Ad measurement	Q1 '21	✓	✓	[17, 18]	[15]
Topics	Show relevant ads	Ad personalisation	Q2 '22	✓	✓	[38, 39]	[3, 5, 6, 42, 43, 57, 59, 60]
Protected Audience	Show relevant ads	Ad retargeting	Q2 '22	✓	✓	[32, 33]	[2, 8, 45, 46, 48, 54, 58]
Private Aggregation	Cross-site privacy boundaries	Privacy-preserving data aggregation	Q1 '22	✓	✓	[28, 29]	[15]
Shared Storage	Cross-site privacy boundaries	Cross-site data storage and aggregation	Q2 '22	✓	✓	[36, 37]	[53]
Related Website Sets	Cross-site privacy boundaries	Cross-site identity and storage grouping	Q2 '21	✓	-	[34, 35]	[50]
CHIPS	Cross-site privacy boundaries	Opt into partitioned cookies	Q1 '22	-	-	[19, 20]	[63]
Fenced Frames	Cross-site privacy boundaries	Privacy-preserving Iframe rendering	Q2 '22	-	-	[25, 26]	-
FedCM	Cross-site privacy boundaries	Identity and single sign-on	Q2 '22	-	-	[22, 23]	-
Private State Tokens		Spam and fraud prevention	Q1 '21	-	-	[30, 31]	[2]

Related Website Sets API [34, 35]. Formerly *First-Party Sets*, it provides a way for organisations to declare multiple domains as part of a single group of services. Browsers apply looser privacy boundaries within a set—for example, enabling shared cookies or identity workflows, while still isolating unrelated third parties. The mechanism targets large multi-domain services. (e.g., shop.acme.com and support.acme.co.uk).

CHIPS (Cookies Having Independent Partitioned State) [19, 20]. Provides a standardised way to mark cookies as *partitioned* by the first-party site. Each third-party embedded in a website receives an isolated cookie jar, preventing cross-site reuse of identifiers while preserving legitimate embedded functionality (e.g., login widgets, shopping carts, media players).

Fenced Frames API [25, 26]. Introduces a `<fencedframe>` HTML element, which embeds content similarly to an `iframe` but prevents data sharing between the top-level site and the embedded content (e.g., an ad banner). This allows ads or other third-party content to be rendered securely and privately, without exposing user data or permitting bidirectional communication.

2.3 Identity and security APIs

Federated Credential Management API (FedCM) [22, 23]. Allows users to log into sites using their federated identity (e.g., "Sign in with...") without relying on third-party cookies or navigational redirects, normally used by protocols such as OAuth 2.0. By mediating authentication exchanges directly within the browser, FedCM reduces cross-site tracking risks while preserving usability for both users and identity providers.

Private State Tokens API [30, 31]. Formerly *Trust Tokens*, it allows websites to issue cryptographically signed tokens that *attest to a user's legitimacy* (e.g., "human vs. bot" CAPTCHA) without exposing identity or linking user's activity across sites. This minimises the need for passive user tracking, such as that used by other CAPTCHA services.

2.4 Deployment characteristics

A distinctive feature of the Privacy Sandbox ecosystem is that several APIs require explicit developer enrolment before use (marked as "Restricted" in Table 1. Third parties must complete an attestation process with Google to obtain the necessary permissions, which the browser verifies at runtime. This mechanism limits API access to a vetted subset of actors and directly affects observable adoption in the wild. These restricted APIs are related to the processing of privacy-sensitive data.

As of early 2026, 265 organisations have completed the enrolment process for some restricted API, up from approximately 220 reported in March 2025. However, enrolment does not imply sustained or production-level deployment. In the remainder of this paper, we show that actual usage remains highly concentrated among a small subset of these actors and that many enrolled entities never invoke the APIs at scale.

Following Google's October 2025 announcement [11], only four of the ten APIs described above—CHIPS, FedCM, Private State Tokens, and Fenced Frames—are expected to remain supported, while the others are scheduled for removal in July 2026.

3 Related work

Research on Google's Privacy Sandbox spans three partially overlapping directions: (i) analyses of the privacy and security properties of individual APIs, (ii) empirical measurements of their deployment and adoption in the wild, and (iii) studies of adjacent or competing privacy-preserving Web technologies. We review these lines of work below and position our contribution accordingly, with Table 1 that summarises the main works grouped by API.

Privacy and security analyses of individual APIs. The majority of prior work has focused on the *Topics API* and its predecessor FLoC. Several studies have shown that, despite coarse-grained categorisation and built-in randomisation, Topics disclosures can be aggregated over time to probabilistically re-identify users or link their activity across sites [3, 6, 42, 43, 57]. Earlier work on FLoC similarly demonstrated that cohort identifiers could be exploited

for cross-site inference [5]. Together, these results highlight the intrinsic tension between interest-based advertising and resistance to cross-site tracking, even in the absence of explicit identifiers.

The *Protected Audience API* (formerly FLEDGE) has received comparable scrutiny. Multiple works demonstrate that covert channels and protocol-level artefacts can be exploited to leak user information or construct stable identifiers, undermining the intended privacy guarantees [2, 8, 48, 58]. These findings question whether browser-mediated ad auctions meaningfully limit tracking in adversarial settings.

Other Privacy Sandbox components have attracted more limited attention. Ghazi *et al.* formally analyse the differential privacy guarantees of the *Attribution Reporting* and *Private Aggregation APIs*, showing that their reporting mechanisms satisfy differential privacy even under interactive use [15]. In contrast, Nisenoff *et al.* demonstrate that the *Shared Storage API* can be abused as a covert channel, enabling cross-site information leakage despite its restricted execution model [53]. McQuistin *et al.* examine the assumptions underlying *Related Website Sets*, showing that users frequently fail to correctly identify site affiliation, potentially enabling overly permissive cross-domain data sharing [50]. Finally, early implementation-focused work on *CHIPS* reported limited adoption among tracking domains during initial deployment phases [63].

Measurement studies of Privacy Sandbox deployment. Several empirical studies have measured the adoption of individual Privacy Sandbox APIs during early deployment. Large-scale crawls report that the Topics API appeared on approximately 30-40% of websites during testing phases, but with fragmented and inconsistent usage patterns, often indicative of experimentation or A/B testing [45, 59, 60]. These studies also highlight frequent invocation before user consent or through unauthorised scripts, commonly attributed to misconfigured consent management logic [60].

Measurements of the Protected Audience API consistently report much lower adoption. Early work observed usage on roughly 1% of websites [54], with later measurements showing peak penetration around 25% during intensive testing periods [45]. Importantly, adoption was found to be heavily concentrated on Google-controlled infrastructure, with the vast majority of auctions initiated by Google-owned domains [8].

To date, no deployment-focused studies provide a unified view across *all* Privacy Sandbox APIs, nor do existing measurements analyse adoption trends over extended time periods or explicitly model the role of consent management platforms in regulating API invocation.

Positioning of this work. Our work differs from prior literature in three fundamental respects. First, rather than analysing individual APIs in isolation, we provide a *comprehensive, cross-API* measurement of all ten Privacy Sandbox components that reached general availability. Second, we conduct a *longitudinal analysis* spanning more than six months, capturing both the expansion and contraction phases of deployment preceding Google's deprecation announcement. Third, our measurements are explicitly *consent-aware*, distinguishing pre-consent, post-consent, and post-denial execution contexts, which prior studies largely overlook despite their central role in regulating third-party behaviour.

By combining these dimensions, our work complements existing privacy and security analyses with ecosystem-level evidence about deployability, concentration of adoption, and the practical limits of browser-enforced privacy architectures in the contemporary Web advertising ecosystem.

4 Data collection and datasets

To characterise the deployment of Privacy Sandbox APIs in the wild, we rely on large-scale, longitudinal active crawling. We design and implement a custom measurement framework that instruments Google Chrome to detect invocations of Privacy Sandbox APIs at runtime, while explicitly modelling user consent states that govern third-party execution on modern websites.

Our measurement campaign targets the top-10,000 websites according to the Tranco ranking and runs weekly over a six-month period. For each site and each crawl iteration, we emulate realistic consent interactions and distinguish between execution occurring before consent, after explicit consent, and after explicit consent denial. This design allows us to capture both intended post-consent usage and execution that is inconsistent with the user's declared choice.

At a technical level, our crawler combines browser-level instrumentation via the Chrome DevTools Protocol with JavaScript-level interception to observe API invocations across the main document and embedded frames. In addition to Privacy Sandbox calls, we record third-party requests and cookie-setting behaviour, enabling joint analysis of API adoption and cookie partitioning.

4.1 API interceptor

The goal of our instrumentation is to detect runtime invocations of Privacy Sandbox APIs as they occur in the browser, together with their execution context and caller origin, while preserving the original semantics of page execution. We aim to capture API usage across the main document and embedded frames, independently of whether calls originate from first-party or third-party scripts.

We build on and extend Priv-Accept [60] to record the Privacy Sandbox API usage using two complementary approaches. In both cases, we use the Selenium library to communicate with the browser through the Chrome DevTools Protocol (CDP) [21], which allows an external program to issue fine-grained commands and listen to specific events. When available, we rely on native Chrome DevTools Protocol (CDP) events, as they provide direct, browser-level visibility into API activity and are not affected by JavaScript obfuscation or redefinition. With this method, we can track 3 Privacy Sandbox APIs: Attribution Reporting, FedCM and Fenced Frames (e.g., `Storage.attributionReportingSourceRegistered`[2]) and record all the exposed parameters.

For APIs that do not emit CDP events, we instrument their JavaScript entry points to observe invocation at runtime, a technique commonly used in Web measurement to capture dynamic API usage. To this end, we use the CDP method `Page.addScriptToEvaluateOnNewDocument` to inject a custom script into the page [56]. The script automatically wraps selected functions (e.g., `document.browsingTopics()` for the Topics API).

[2]https://chromedevtools.github.io/devtools-protocol/tot/Storage/#event-attributionReportingSourceRegistered, accessed on July 1, 2026

Each wrapper logs the function name, arguments, return value, and script origin, and forwards this information to the interceptor using CDP bindings (`Runtime.addBinding`).[3] The original API is finally executed to avoid altering the behaviour of the webpage. We inject the script into the main page and every iframe as soon as they are discovered. Iframe discovery is facilitated by CDP's "auto-attach" feature,[4] with the `waitForDebuggerOnStart` option enabled. This option forces the browser to halt all activity (i.e., page loading and script execution) whenever a new iframe is found and transfer the control to the CDP until it completes its task.[5] This allows us to proceed with the script injection before any iframe element loads, ensuring that we do not miss any invocations due to race conditions. All wrappers transparently forward arguments to the original API implementation and return its original return value, ensuring that page behaviour is not altered by our instrumentation.

Thanks to this twofold strategy, we can collect, for each API of the Privacy Sandbox, the name of the function being called, the URL of the caller, henceforth *Calling Party* (CP)[6], the domain of the frame where the call occurred, eventual parameters passed to the function and its return value, independently of the context where the call happens.

4.2 Managing consent banners: *Before-Accept*, *After-Accept*, *After-Deny* visits

Since Privacy Sandbox APIs are predominantly invoked by third-party scripts whose execution is often gated by consent management platforms (CMPs), any measurement that ignores the user's consent risks substantially mischaracterising both the prevalence and timing of API usage [44, 60]. For this reason, our crawler models three distinct user-consent states and performs separate visits in each state to capture the Privacy Sandbox behaviour under realistic CMP-controlled loading logic.

***Before-Accept*.** For every website, the crawler performs an initial visit with a fresh browser profile. The crawler does not interact with the page and only records all Privacy Sandbox API invocations. During this phase, no consent is provided: any API invocation here reflects code executed before user agreement, either because CMPs fail to block components correctly or because the website does not implement blocking at all. The Before-Accept state approximates the experience of a first-time user who has not yet interacted with the consent banner, and API invocations reflect code executed before any explicit user choice.

After-Accept After completing the first visit, we attempt to locate and click an "Accept button" by matching it against a curated list of multilingual keywords.[7] After accepting consent, we clear the browser cache and reload the page to ensure that all consent-dependent scripts are executed under the new state, yielding a clean and comparable post-consent measurement. This yields a second measurement where all components released after consent, including additional ads, profiling scripts, and consent-dependent Privacy Sandbox calls, can execute. If no Accept button is found, we skip the *After-Accept* visit. We treat the After-Accept state as the reference scenario for legitimate Privacy Sandbox deployment, when the users granted consent to API usage.

Before-Deny* and *After-Deny. To complement the process, we extend the crawler to emulate users explicitly denying consent. Replicating the same strategy, we collect a list of keywords to search for a "Deny button" within a privacy banner. We manually build the list by visiting the top-100 websites according to the SimilarWeb [1] list in Italy, Germany, France, Spain and the UK. The final list contains 166 keywords across the five languages.

During this analysis, we verified that many banners do not expose a straightforward "Deny Button", despite regulatory guidance (e.g., France's data protection authority CNIL, which mandates that rejecting cookies must be as simple as accepting them): some hide the control behind "Options" submenus, while others present a "Deny and Subscribe" paywalled alternative. Unlike simpler crawlers, our implementation supports multi-step denial flows in which the "deny" option is nested behind configuration or "options" menus, increasing coverage of realistic CMP designs. However, it cannot bypass paywalls.

The resulting *After-Deny* dataset captures Privacy Sandbox activity when the user has explicitly refused personal-data processing. Any invocation observed here is labelled as *questionable* behaviour, in line with [60], regardless of their legal interpretation.

In the remainder of the paper, we collectively refer to *Before-Accept* and *Before-Deny* as *Before-Click*, and to any API invocation observed in *After-Deny* as consent-inconsistent usage. In summary, we have:

- Pre-consent execution (*Before-Accept*/*Before-Deny*, collectively *Before-Click*);
- Legitimate consent-dependent usage (*After-Accept*);
- Consent-inconsistent usage (*After-Deny*).

4.3 Data collection

To provide a complete picture of Privacy Sandbox usage over time, we run the crawler to visit the top-10,000 websites according to the Tranco list [47] that we refresh weekly to reflect current site popularity.[8] All crawls are executed using Google Chrome version 138.0.7204.4, with all Privacy Sandbox features explicitly enabled. We fix the browser version throughout the measurement period to avoid confounding effects due to browser updates.

For each website, we store:

- the URL of each first- and third-party object downloaded by the browser to render the website;
- the set of cookies created during the visit;
- the calls made to each Privacy Sandbox API.

For each cookie, we collect its name, value and other relevant metadata, such as the domain of its creator and whether it is partitioned.

We repeat the crawl weekly, starting every Monday at 00:00 (GMT+1) from a European server. For each website, we load the landing page (following redirects when present) up to four times

[3] https://chromedevtools.github.io/devtools-protocol/tot/Runtime/#method-addBinding, accessed on July 1, 2026.
[4] https://chromedevtools.github.io/devtools-protocol/tot/Target/#method-setAutoAttach, accessed on July 1, 2026.
[5] https://chromedevtools.github.io/devtools-protocol/tot/Runtime/#method-runIfWaitingForDebugger, accessed on July 1, 2026.
[6] We define the calling party (CP) as the eTLD+1 of the script origin responsible for invoking the API; for inline scripts, we associate the call with the document origin.
[7] We include keywords in English, French, German, Italian and Spanish, following the approach of Priv-Accept [44].
[8] Tranco provides only a single global ranking, but it is the most established ranking service, while SimilarWeb offers a per-nation list but only for the top-100 websites.

Table 2: Summary of the number of successful visits for the different paths followed by the crawler.

Visit	Action	Visits min-max	3rd parties
Before	*Accept*	7,625–8,262	7,324–7,973
	Deny	7,423–8,448	7,041–7,989
	Click	7,703–8,473	7,354–8,025
After	Accept	3,209–3,772	4,471–5,132
	Deny	2,082–2,199	2,524–2,906

to capture the *before* and *after* consent states, allowing up to 60 s per visit. To mitigate transient network failures and rate limiting, failed visits are retried up to three times before being discarded for that snapshot. The campaign runs from 23 June 2025 to 5 January 2026, covering more than six months. Using 20 crawler instances in parallel, each full cycle completes in roughly 36 hours.

As with any automated crawl, websites or embedded third parties may detect non-human behaviour and alter page execution, for example, by presenting CAPTCHA or verification interstitials or by suppressing third-party script execution (cloaking). Likewise, the crawler may occasionally fail to trigger the "Accept/Deny" flow correctly. These effects introduce randomness and potential bias into our measurements.

We mitigate their impact by repeating measurements weekly and analysing trends across large populations rather than individual websites. Furthermore, we compute all statistics over the set of websites successfully visited for the corresponding visit type and week. This mitigates bias due to population changes, such as the fact that the *After-Accept* and *After-Deny* datasets are smaller than the *Before-Click* dataset.

Finally, our primary findings rely on persistent adoption patterns and strong concentration effects among major third parties, which are inherently less sensitive to sporadic cloaking or transient crawl failures.

4.4 Dataset overview

We present a summary of first and third parties contacted in the different scenarios of the crawling activity in Table 2. For completeness, we report the *Before-Accept* and *Before-Deny* visits separately, and their merging in the *Before-Click* dataset. Here, we also merge the set of contacted third parties, which is the union of the contacted third party sets in *Before-Accept* and *Before-Deny*.

During each week, we successfully visited between 7,400 and 8,400 websites in any *Before-* visit. On average, 93% of websites overlap between the *Before-Accept* and *Before-Deny* visits across all dates. The high overlap between *Before-Accept* and *Before-Deny* visits indicates that failures are largely stochastic and do not systematically affect specific subsets of websites, e.g., random connection failures and website anti-bot measures like CAPTCHA. Interestingly, in the *Before-Click* visit, each website includes fewer than one third-party request on average.

Moving to the *After-Accept* visits, our crawler succeeds in accepting the website policy in about 44% of *Before-Accept* visits, a percentage in line with [44]. The majority of failures are due to the website not showing a consent banner at all. During the *After-Accept* visits, we encounter about 4,400-5,100 total third parties, with the average number of third parties per website that grows to 1.4-1.5. This reflects the fact that websites enable additional third-party-powered features, e.g., ads, after users provide their consent.

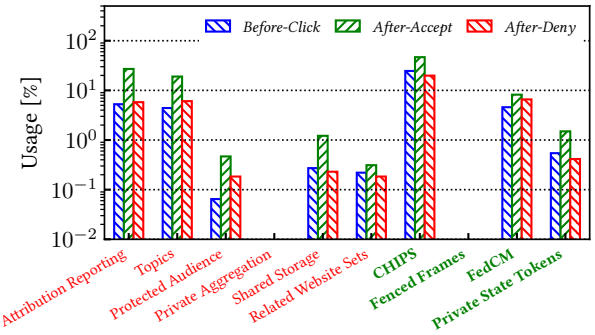


Figure 1: Percentage of websites where each API was called during the 5 January 2026 snapshot. In red, the retired APIs. In bold green, the still-supported ones.

For the *After-Deny*, the crawler succeeds in denying consent in about 2,100 cases (≈28% of the successful *Before-Deny* cases). In addition to the cases where no banner is present, this lower percentage is due to the additional complexity of denying cookies, which often requires multiple interactions or access via paywalled mechanisms.

To better characterise the limitations of automated denial, we manually inspected a random sample of 50 websites where the crawler failed to deny consent. In 23 of them, there is no banner, 7 are in unsupported languages, 6 use multiple toggles or checkboxes instead of a single deny button, 5 are not active, 2 have no deny option, and 1 offers a paywalled option. In the remaining cases, the crawler failed to match the keywords (3 sites), or was unable to detect the deny button despite its presence (3 sites). In short, failures are primarily due to the absence of a consent banner, unsupported languages, multi-toggle interfaces, hidden deny options, or paywalled alternatives.

In the following sections, we leverage these datasets to characterise Privacy Sandbox usage at scale, first providing a cross-API snapshot and then analysing longitudinal adoption trends. To support reproducibility, the code and datasets used in this study are released as described in Appendix C.

5 Privacy Sandbox usage at large

We begin by providing a cross-API snapshot of Privacy Sandbox usage using the most recent measurement week, in order to characterise overall adoption levels and the impact of consent state on observed API invocation. Figure 1 reports, for each API, the share of websites where at least one invocation is recorded, distinguished by consent state. Recall that we compute percentages over the set of successfully visited websites for that specific visit type to avoid bias.

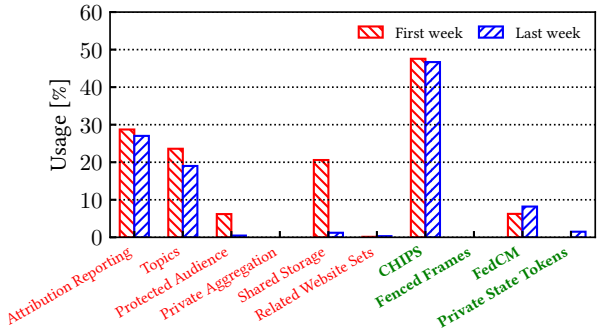


Figure 2: Percentage of websites where each API call was encountered in the first and last week of measurements, *After-Accept*. In red, the retired APIs. In bold green, the still-supported ones.

Figure 1 reveals a clear stratification among Privacy Sandbox components, with a small number of APIs exhibiting non-negligible deployment and the majority remaining marginal. Only CHIPS, Topics, and Attribution Reporting API exceed the 10% threshold for *After-Accept* visits. CHIPS (active on ≈ 50% of websites) stands apart from other components, as it enables stateful embedded functionality rather than user profiling, and its usage is therefore not necessarily contingent on user consent. For several restricted APIs, we observe invocations in *Before-Click* and, more notably, *After-Deny* visits, indicating execution that is inconsistent with the declared user choice. As shown in prior work, such behaviour often stems from CMP misconfiguration or third-party integration errors rather than deliberate misuse [60]. We will investigate this later.

5.1 *After-Accept* usage

We focus on the *After-Accept* state as it represents the intended deployment scenario for Privacy Sandbox APIs, where execution is explicitly enabled by user consent. Here, we observe the uneven adoption landscape (green bars in Figure 1). After CHIPS, Topics API (≈17%) and Attribution Reporting API (≈21%) show notable penetration. All remaining components show marginal use: FedCM reaches barely 6%, while Protected Audience, Private State Tokens, and Related Website Sets usage remain below 1%.[9] Private Aggregation and Fenced Frames are practically absent. In the following, where not explicitly indicated, the figures show API penetration in the *After-Accept*.

Overall, these figures indicate limited ecosystem uptake of most Privacy Sandbox APIs, even two years after they reached general availability. Given its markedly higher prevalence, we analyse CHIPS in detail in Sec. 6 and compare its deployment to that of traditional third-party cookies.

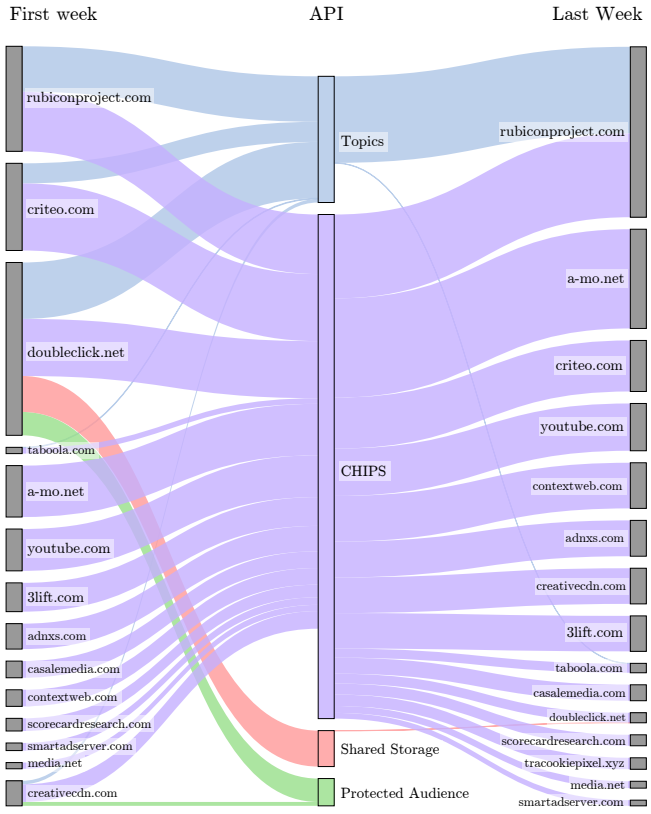


Figure 3: Flow graph of third-party callers during the latest campaign.

5.2 Effect of API retirement

Figure 2 compares API presence during the first and last measurement week (23 June → 5 January) in the *After-Accept*. The overarching trend is a contraction: for retired APIs, Shared Storage and Protected Audience collapse almost entirely, falling from ≈20% and ≈8% to near zero. Topics and Attribution Reporting decline more gradually (-1.5 to -4.5 percentage points), and their usage remains non-negligible at 19–27%.

Among maintained APIs (in green), CHIPS remains the most pervasive, maintaining a stable adoption rate slightly below 50% of the websites. FedCM and Private State Tokens remain low and flat across the entire period. Fenced Frames are ignored altogether.

Crucially, Google's announcement that these APIs would be maintained did not coincide with any visible increase in their adoption, suggesting a broader retreat from Privacy Sandbox technologies rather than a reallocation of adoption efforts.

5.3 Adopter Third Parties

Out of more than 7,000 third parties observed in our data, 265 are attested entities. Only 50 of them called at least one Privacy Sandbox API during the first week; callers drop to 41 by January 2026 (see Section 7 and 8 for details), testifying how the set of adopters is extremely concentrated.

[9]The Protected Audience figure is in line with Google's statement in [41].

Figure 3 shows the top 15 calling domains among the 41 still active in January. We compare their API usage with that seen during the first week of our measurements. The thickness of the link between the CP and the API represents the number of websites where a third party calls an API. The landscape is dominated by major advertising intermediaries such as doubleclick.net, criteo.com, and rubiconproject.com. All rely on CHIPS, often exclusively. Topics API remains their second most common choice, although only rubiconproject.com uses it consistently. It is interesting to note that, although the overall CHIPS adoption rate remains stable as reported in Figure 2, the usage among the most pervasive third parties increases.

Shared Storage and Protected Audience disappear rapidly: doubleclick.com—the last relevant user of Shared Storage—withdraws it towards the end of the measurement period, causing the API's usage to collapse to almost 0 across all websites.

Interestingly, Attribution Reporting is under-represented in this analysis because many events expose only the execution context and not the initiating third party, limiting caller attribution. This is an artefact of the API's reporting model that does not allow us to trace back those invocations to the triggering third-party platform.

Curiously, doublelick.net—the Google-owned digital advertising platform—was the top adopter in the first week. In the last week, it shows the greatest usage reduction, abandoning all APIs but CHIPS. creativecdn.com, the only other adopter of the Protected Audience, abandons them altogether. Conversely, criteo.com and rubiconproject.com maintain significant usage of the Topics API.

Taken together, these results depict an ecosystem in which only a small subset of Privacy Sandbox components achieved measurable deployment, and even those exhibit declining or stagnant adoption. In the remainder of the paper, we examine these trends in greater detail, first analysing CHIPS and cookie partitioning (Sec. 6), and then focusing on the longitudinal evolution of retired and maintained APIs (Sec. 7 and 8).

6 Cookies Having Independent Partitioned State (CHIPS)

In this section, we focus on the adoption of CHIPS, the most widely deployed Privacy Sandbox component, and we compare it to traditional, unpartitioned third-party cookies. Although Google presents CHIPS as a Privacy Sandbox API, it reduces to setting the `Partitioned` attribute in the HTTP Set-Cookie header (or in the equivalent JavaScript API). Cookies marked as `Partitioned` are stored in an isolated environment per top-level site, so that a third party cannot re-access the same cookie value across different first parties, thereby preventing cross-site tracking and the possibility of building browsing profiles. In the remainder of this paper, we refer to cookies with the Partitioned attribute as partitioned cookies, and to all others as traditional cookies.

6.1 CHIPS v. traditional cookies

We first dig deeper into the comparison between partitioned and traditional cookies, comparing the rate of websites where third parties install cookies. To make a greater sense of this comparison, we also consider a similar crawling procedure on Mozilla Firefox, which enforces cookie partitioning by default since 2021.[10] We apply the same methodology described in Section 4, including the Firefox browser and maintaining the accept-or-deny-cookies stages. In total, with the Firefox browser, we were able to click on an "Accept"-or-equivalent button in 3,409 websites, and a "Deny"-or-equivalent in 2,239 websites, figures that are comparable with the Google Chrome scenario. For this analysis, we consider both session and persistent cookies.

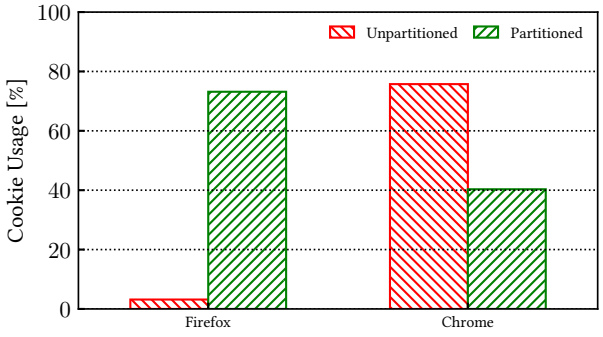


Figure 4: Percentage of websites where a third party installs a cookie (both session and persistent cookies).

Figure 4 sums up the comparison, showing clear differences: using Chrome, we observe a third party installing a partitioned cookie in around 40% of websites, while in 70% of cases they still install unpartitioned cookies. With Firefox, third parties are forced to use partitioned cookies only, reducing the business opportunities of tracking. Note indeed that the percentage of websites where we observe a traditional cookie in Chrome is the same as where we observe partitioned cookies in Firefox.

Finally, some unpartitioned cookies are observed in Firefox. A manual inspection reveals that these cases arise from limitations in our cookie classification: we classify cookies as third-party when their effective TLD differs from that of the visited site. In some cases, TLDs slightly differ (e.g., cookies with domain zoom.com set on zoom.us), even though the two domains belong to the same service.[11] Correctly identifying these cases would require manual and domain-to-service reconciliation. We therefore acknowledge the presence of this bias in our dataset (≈2.5% of Firefox websites contain seemingly third-party cookies), which may similarly affect Google Chrome.

6.2 CHIPS v. other Privacy Sandbox APIs

We then move to comparing the usage of partitioned cookies with that of the other Privacy Sandbox APIs. We do so in Figure 5, listing the 15 third parties using a Privacy Sandbox technology in the largest share of websites—among those present on at least 50 websites. Among highly prevalent third parties, CHIPS is the only Privacy Sandbox technology in use. Only rubiconproject.com uses

[10] https://blog.mozilla.org/security/2021/02/23/total-cookie-protection/, accessed on July 1, 2026

[11] We use the *tldextract* library https://github.com/joeguo/tldextract and the DuckDuckGo map https://github.com/duckduckgo/tracker-radar/blob/main/build-data/generated/domain_map.json to extract the TLD.

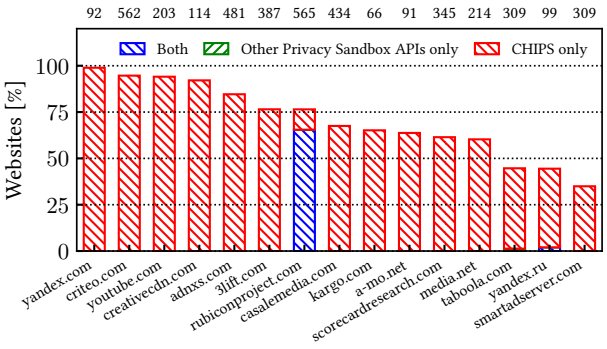


Figure 5: Percentage of websites where a third party calls a Privacy Sandbox API other than CHIPS, in *After-Accept*. The top axes contains the amount of websites where each third party is active.

other Privacy Sandbox APIs (Topics API in particular, as shown in Figure 3, combining them with CHIPS. yandex.com installs partitioned cookies on all the websites where it is present. criteo.com, youtube.com, and creativecdn.com show very high percentages. Interestingly, Google's owned doubleclick.com, not in the figure, uses partitioned cookies only on ≈12% of the websites and traditional cookies on ≈55%, highlighting differences between promoted mechanisms and observed deployment patterns.

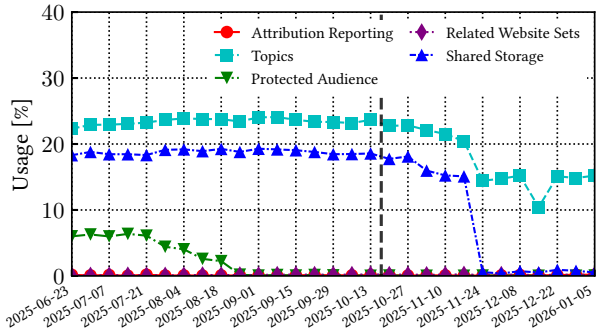
7 Retired API trend

We now analyse the longitudinal adoption trends of the six Privacy Sandbox APIs that Google announced would be retired, focusing on their deployment by third-party callers over 6 months. In Figure 6, we show the per-week measurement timeline. Again, we consider only third-party callers and *After-Accept*. We do not include the Private Aggregation API, given its virtual absence from our dataset.

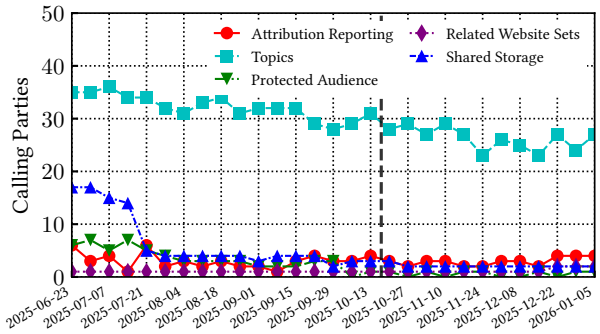
Start from the percentage of websites where these APIs are invoked, in Figure 6a. Several observations emerge: First, the Protected Audience API (green curve) shows the most abrupt decline: its usage drops from around 8% of websites in late June to nearly zero by early August, well before the API retirement announcement. This contraction aligns with popular adopters, most notably creativecdn.com, discontinuing their calls during July. After August, the few remaining invocations exclusively initiate auctions, with no evidence of new interest-group assignments. The rapid decline suggests that early adopters discontinued testing or failed to transition from experimentation to sustained deployment.

Second, Topics API and Shared Storage exhibit relatively stable presence until Google's retirement announcement, after which both APIs show a gradual decline that culminates in a sharp collapse in 24 November 2025 when Shared Storage falls from roughly 19% of websites to almost zero. Also doubleclick.com dismissed its usage—see below.

Figure 6b complements this picture by reporting the number of third-party callers for each retired API. Across all retired APIs, the number of active callers never exceeds a few dozen and steadily decreases over time, underscoring the highly concentrated nature of



(a) Percentage of websites where a call to a given API is present.



(b) Number of API callers.

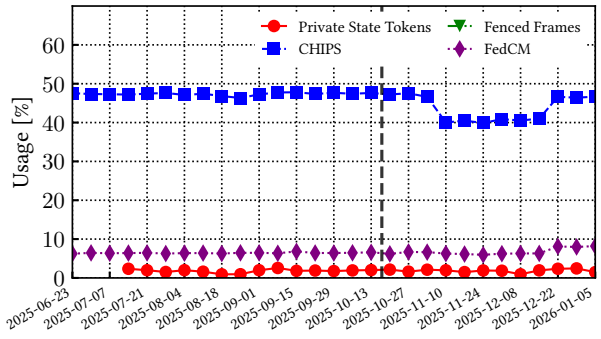
Figure 6: Timeline of retired APIs adoption by third parties. The vertical dashed line highlights the retirement announcement.

Privacy Sandbox experimentation. The Topics API shows a slow decline that started before the retirement announcement. For Shared Storage, the original 20 callers drop to 3 within a month in July. Yet, the apparent stability in website-level presence despite a sharp reduction in callers is driven by the single large actor doubleclick.com deploying the API across many sites, masking ecosystem-wide disengagement.

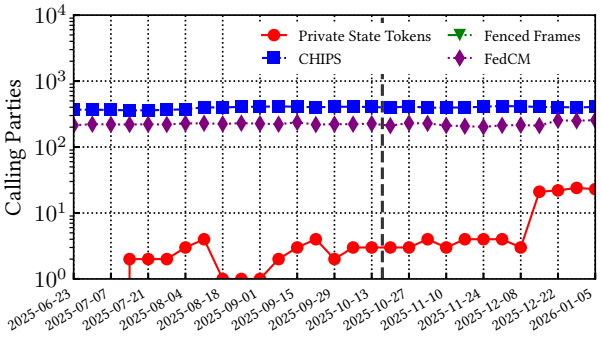
Overall, our measurements show that the decline of retired Privacy Sandbox APIs was not triggered by their deprecation announcement, but rather reflects a sustained lack of ecosystem uptake and the withdrawal of early adopters. The small number of callers, coupled with declining usage well before the announcement, suggests that these APIs failed to transition from experimental deployments to stable production use.

8 Maintained API trend

Google opted to maintain four components of the Privacy Sandbox: CHIPS, Private State Tokens, Fenced Frames, and Federated Credential Management. To assess their adoption, we track their usage across our longitudinal measurements, as summarised in Figure 7, *After-Accept* case. Here we consider all callers, and not only third



(a) Percentage of websites where a call to a given API is present.



(b) Number of API callers.

Figure 7: Timeline of supported API adoption. The vertical dashed line highlights the date of the retirement announcement.

parties. Indeed, these APIs do not require the user to complete an attestation process and can be freely used by any service provider.

Figure 7a shows that, except CHIPS, none of the maintained APIs exceeds the 10% adoption rate, with no meaningful changes over time. Fenced Frames never appear in our measurements. Their absence from our measurements likely reflects the lack of immediate pressure to replace iframes, rather than measurement limitations.

Figure 7b summarises the number of callers for each API. Private State Tokens API is invoked by only a handful of players (from 0 to 5 in each snapshot, its variability due to measurement variance), whereas FedCM shows a broader and more diverse caller base. Unlike other APIs, FedCM is primarily invoked by first-party sites, explaining the broader but relatively stable caller base. CHIPS is invoked by a substantially larger and more diverse set of callers (about 350), including many first-party services, consistent with its role as a compatibility-oriented mechanism rather than an advertising-specific API.

Notably, we do not observe any measurable increase in adoption following Google's announcement that these APIs would remain supported, suggesting that the market has not strongly reacted to Google's strategic signalling.

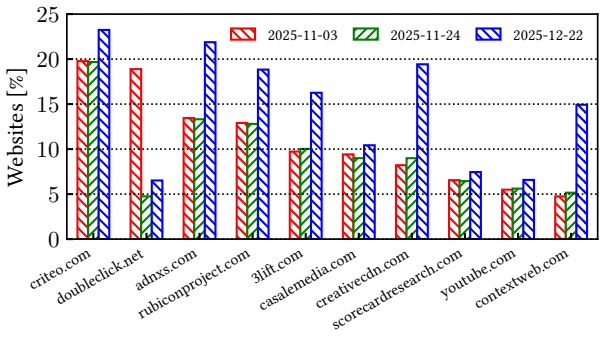


Figure 8: Breakdown of CHIPS usage by third-party caller between 3 November, 24 November, and December 22, 2025.

We take a closer look at the temporary decrease in the percentage of websites using CHIPS observed between 10 November and 22 December 2025 (Figure 7a).

Figure 8 compares the share of websites where a third party deploys CHIPS across three representative snapshots: before the decrease (3 October 2025), during the decrease (24 November 2025), and after the recovery (22 December 2025). We observe that the decrease and subsequent increase have different underlying dynamics. The initial drop is driven by doubleclick.com, which substantially reduces the number of websites on which it installs partitioned cookies (only green bar to decrease), while the behaviour of other major third parties remains largely stable. In contrast, the subsequent increase involves a coordinated rise across multiple third-party callers (all blue bars grow). This pattern suggests a shared implementation change, such as an update in a widely used cookie management library, affecting several actors simultaneously. A more detailed investigation of these mechanisms is left for future work.

9 Questionable usage

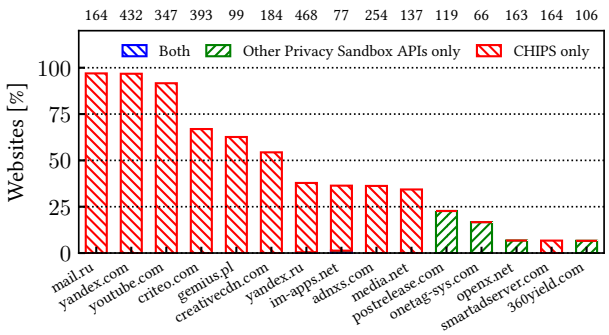
We conclude our results by analysing the use of Privacy Sandbox technologies in situations where the user has not yet accepted nor declined the privacy banner (*Before-Click*), or has explicitly denied consent (*After-Deny*). We refer to both as *questionable* in the sense of being inconsistent with the intended CMP-mediated gating model, without asserting legal non-compliance.

As previously shown in Figure 1, we witness a non-negligible fraction of questionable usage.

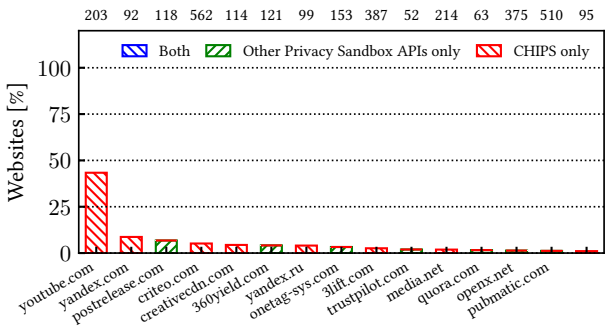
9.1 Pervasive third parties

Figure 9 reports the third parties responsible for questionable usage. It shows, for each domain, the percentage of websites where it either sets CHIPS cookies only, invokes another privacy-sensitive API, or uses both (similarly to Figure 5). All percentages in Figure 9 are conditional on the third party being present on the site.

Comparing *Before-Click* and *After-Deny* visits, Figure 9 shows that the majority of questionable events involve CHIPS only (red pattern). mail.ru, youtube.com, and yandex.com set CHIPS in nearly all websites where they appear, while other actors exhibit more partial or sporadic behaviour. Since CHIPS are a form of cookie

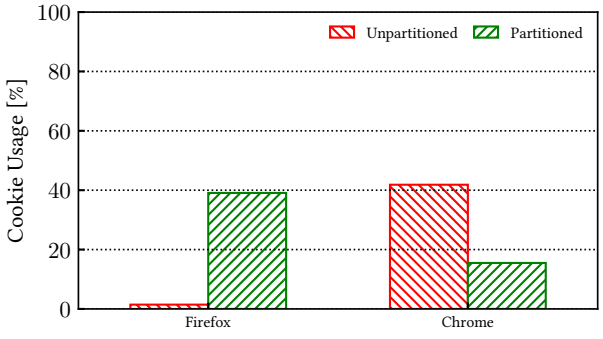


(a) *Before-Click*

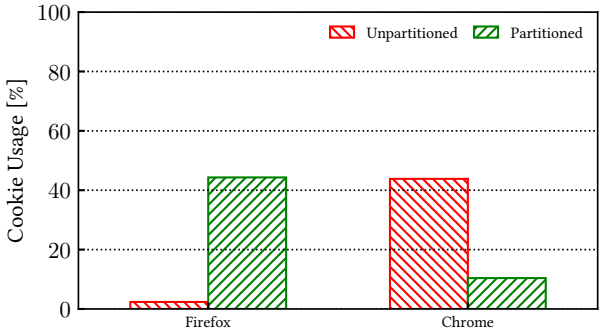


(b) *After-Deny*

Figure 9: Comparison between CHIPS and other Privacy Sandbox API usage per top third party during *Before-Click* and *After-Deny* visits.



(a) *Before-Click*



(b) *After-Deny*

Figure 10: Comparison of persistent third-party cookie usage percentage during *Before-Click* and *After-Deny* visits.

and may include benign, non-privacy-violating cases (e.g., stateful embedded services), we cannot infer malicious intent solely from their presence, and we limit ourselves to documenting this usage.

However, the share of websites showing questionable usage drops sharply after users explicitly deny consent (see Figure 9b). This indicates that some API-related activity is indeed controlled by consent mechanisms, and that a meaningful part of their usage observed before consent corresponds to components that are deactivated after user consent denial. For example, mail.ru sets CHIPS cookies on nearly all the 164 websites where it appears before consent, but largely disappears after consent is denied. Similarly, youtube.com sets CHIPS cookies on more than 340 websites before consent, while doing so on fewer than 100 websites after consent denial.

A few third parties use other Privacy Sandbox technologies on a fraction of websites where they are present, namely postrelease.com, 360yield.com, and onetag-sys.com in both *Before-Click* and *After-Deny*. Again, this fraction decreases in *After-Deny*, hinting at an undesirable but limited behaviour.

Overall, these questionable patterns indicate an immature and inconsistent deployment of Privacy Sandbox technologies rather than deliberate attempts to bypass consent or engage in privacy-violating behaviour.

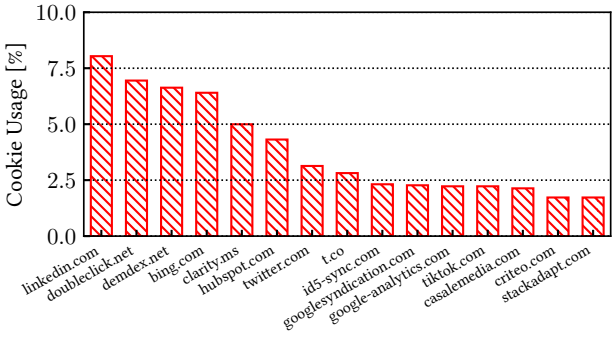


Figure 11: Breakdown of third parties that set an unpartitioned cookie during *After-Deny* on the Chrome browser.

9.2 Questionable CHIPS and traditional cookie usage

Finally, we extend the analysis of Section 6.1, observing how the usage of unpartitioned and partitioned cookies changes when considering *Before-Click* and *After-Deny* scenarios. Figure 10 shows the percentage of websites where we observe a third party installing a cookie, both in Firefox and Chrome. Contrary to Figure 4, here we focus on persistent cookies only, which could possibly be used for tracking. For completeness, we report the cookie duration Cumulative Distribution Function (CDF) in Appendix A.

Compared to *After-Accept*, cookie-setting activity decreases, yet we still observe third-party cookie installation on over 40% of sites in both *Before-Click* and *After-Deny*. Firefox forces the usage of partitioned cookies, while Chrome allows for unpartitioned and persistent cookies. Differently from Figure 9, we notice no major difference comparing *Before-Click* and *After-Deny*.

Looking at which are the third-parties setting questionable cookies in *After-Deny*, in Figure 11, we observe that most of the biggest players in the field are responsible for this behaviour, with linkedin.com, doubleclick.com, and demdex.net leading the way. Observing the same graph (not shown for brevity) for the *Before-Click* scenario shows similar results, hinting at constant behaviour.

This indicates that third-party state is not consistently gated by the consent flow, motivating deeper auditing of CMP integration and third-party loading logic as seen in [60]. Investigating possible legal implications is outside the scope of this paper and left for future work.

10 Conclusions

In this paper, we analysed the final months of Google's Privacy Sandbox initiative, measuring how these APIs were deployed across the Web during the period preceding their retirement. Our longitudinal results show that the deprecation announcement merely formalised an ongoing trend: most APIs saw limited adoption and were already being abandoned by the ecosystem, with only a small set of actors ever testing them.

Only CHIPS, i.e., partitioned cookies, achieved broad and sustained usage. However, in Google Chrome, their adoption has not come at the expense of classical, unpartitioned third-party cookies, which continue to be by far the most popular—as well as the most privacy-invasive—means to manage stateful services.

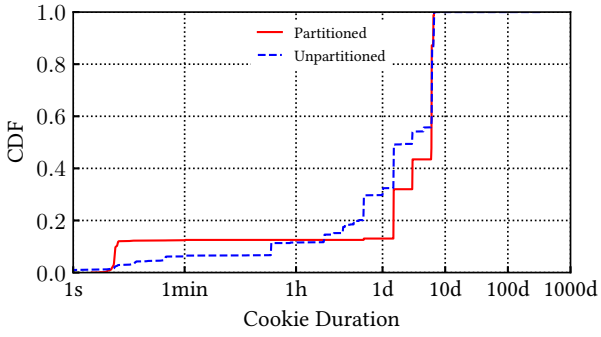
All other APIs are approaching irrelevance. Protected Audience and Shared Storage usage declined to near-zero months before the API retirement announcement, and Topics API is following a similar trajectory. For the APIs that Google will maintain, we observe no momentum: Private State Tokens, Fenced Frames, and FedCM appear on only a small fraction of sites, with no sign of growth.

A framework originally presented as a compromise between privacy and advertising needs ultimately failed to gain traction, and Google finally abandoned it. With the Privacy Sandbox fading without leaving a viable alternative, the challenge of enabling interest-based yet privacy-preserving advertising remains open, calling for renewed efforts to design privacy-preserving advertising mechanisms that better align deployability, incentives, and enforcement.

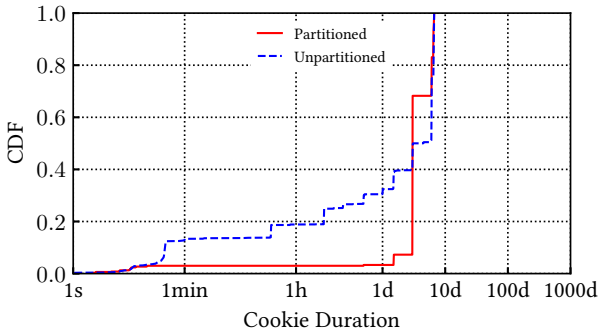
References

[1] 2024. SimilarWeb. https://similarweb.com/, accessed on July 1, 2026.
[2] Mir Masood Ali, Binoy Chitale, Mohammad Ghasemisharif, Chris Kanich, Nick Nikiforakis, and Jason Polakis. 2023. z. In *Network and Distributed System Security (NDSS) Symposium*.
[3] Mário S Alvim, Natasha Fernandes, Annabelle McIver, and Gabriel H Nunes. 2024. The privacy-utility trade-off in the topics API. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*. 1106–1120.
[4] Apple. 2024. AdAttirbutionKit. https://developer.apple.com/documentation/AdAttributionKit, accessed on July 1, 2026.
[5] Alex Berke and Dan Calacci. 2022. Privacy limitations of interest-based advertising on the web: A post-mortem empirical analysis of google's floc. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. 337–349.
[6] Yohan Beugin and Patrick McDaniel. 2024. Interest-disclosing Mechanisms for Advertising are Privacy-Exposing (not Preserving). *Proceedings on Privacy Enhancing Technologies* 1 (2024), 41–57.
[7] Brave. 2021. Ephemeral third-party site storage. https://brave.com/privacy-updates/7-ephemeral-storage/, accessed on July 1, 2026.
[8] Giuseppe Calderonio, Mir Masood Ali, and Jason Polakis. 2024. Fledging Will Continue Until Privacy Improves: Empirical Analysis of Google's {Privacy-Preserving} Targeted Advertising. In *33rd USENIX Security Symposium (USENIX Security 24)*. 4121–4138.
[9] Anthony Chavez. 2022. Expanding testing for the Privacy Sandbox for the Web. https://blog.google/products/chrome/update-testing-privacy-sandbox-web/, accessed on July 1, 2026.
[10] Anthony Chavez. 2023. Privacy Sandbox for the Web reaches general availability. https://privacysandbox.com/news/privacy-sandbox-for-the-web-reaches-general-availability/, accessed on July 1, 2026.
[11] Anthony Chavez. 2025. Privacy Sandbox for the Web reaches general availability. https://privacysandbox.com/news/update-on-plans-for-privacy-sandbox-technologies/, accessed on July 1, 2026.
[12] Steven Englehardt and Arvind Narayanan. 2016. Online Tracking: A 1-million-site Measurement and Analysis. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (Vienna, Austria) *(CCS '16)*. Association for Computing Machinery, New York, NY, USA, 1388–1401. https://doi.org/10.1145/2976749.2978313
[13] Firefox. 2024. Privacy-Preserving Attribution. https://support.mozilla.org/en-US/kb/privacy-preserving-attribution, accessed on July 1, 2026.
[14] Firefox. 2025. Introducing Total Cookie Protection in Standard Mode. https://support.mozilla.org/en-US/kb/introducing-total-cookie-protection-standard-mode/, accessed on July 1, 2026.
[15] Badih Ghazi, Charlie Harrison, Arpana Hosabettu, Pritish Kamath, Alexander Knop, Ravi Kumar, Ethan Leeman, Pasin Manurangsi, Mariana Raykova, Vikas Sahu, et al. 2025. On the Differential Privacy and Interactivity of Privacy Sandbox Reports. *Proceedings on Privacy Enhancing Technologies* (2025).
[16] Google. 2024. Preparing for the end of third-party cookies. https://blog.google/products/chrome/update-testing-privacy-sandbox-web/, accessed on July 1, 2026.
[17] Google. 2025. Attribution Reporting API. https://privacysandbox.google.com/private-advertising/attribution-reporting, accessed on July 1, 2026.
[18] Google. 2025. Attribution Reporting API Explainer. https://github.com/WICG/attribution-reporting-api, accessed on July 1, 2026.
[19] Google. 2025. CHIPS API. https://privacysandbox.google.com/cookies/chips, accessed on July 1, 2026.
[20] Google. 2025. CHIPS API Explainer. https://github.com/privacycg/CHIPS, accessed on July 1, 2026.
[21] Google. 2025. Chrome Devtools Protocol. https://chromedevtools.github.io/devtools-protocol/, accessed on July 1, 2026.
[22] Google. 2025. Federated Credential Management API. https://developer.chrome.com/docs/identity/fedcm/overview, accessed on July 1, 2026.
[23] Google. 2025. Federated Credential Management API Explainer. https://github.com/w3c-fedid/FedCM, accessed on July 1, 2026.
[24] Google. 2025. Federated Learning of Cohorts. https://github.com/WICG/floc, accessed on July 1, 2026.
[25] Google. 2025. Fenced Frames API. https://privacysandbox.google.com/private-advertising/fenced-frame, accessed on July 1, 2026.
[26] Google. 2025. Fenced Frames API Explainer. https://github.com/WICG/fenced-frame, accessed on July 1, 2026.
[27] Google. 2025. The Privacy Sandbox. https://privacysandbox.com/, accessed on July 1, 2026.
[28] Google. 2025. Private Aggregation API. https://privacysandbox.google.com/private-advertising/private-aggregation, accessed on July 1, 2026.
[29] Google. 2025. Private Aggregation API Explainer. https://github.com/patcg-individual-drafts/private-aggregation-api, accessed on July 1, 2026.
[30] Google. 2025. Private State Tokens API. https://github.com/WICG/trust-token-api, accessed on July 1, 2026.

[31] Google. 2025. Private State Tokens API Explainer. https://github.com/WICG/trust-token-api, accessed on July 1, 2026.
[32] Google. 2025. Protected Audience API. https://privacysandbox.google.com/private-advertising/protected-audience, accessed on July 1, 2026.
[33] Google. 2025. Protected Audience API Explainer. https://github.com/WICG/turtledove/blob/main/FLEDGE.md, accessed on July 1, 2026.
[34] Google. 2025. Related Website Sets API. https://privacysandbox.google.com/cookies/related-website-sets, accessed on July 1, 2026.
[35] Google. 2025. Related Website Sets API Explainer. https://github.com/WICG/first-party-sets, accessed on July 1, 2026.
[36] Google. 2025. Shared Storage API. https://privacysandbox.google.com/private-advertising/shared-storage, accessed on July 1, 2026.
[37] Google. 2025. Shared Storage API Explainer. https://github.com/WICG/shared-storage, accessed on July 1, 2026.
[38] Google. 2025. Topics API. https://privacysandbox.google.com/private-advertising/topics, accessed on July 1, 2026.
[39] Google. 2025. Topics API Explainer. https://github.com/patcg-individual-drafts/topics, accessed on July 1, 2026.
[40] Google. 2025. Update on the plan for phase-out of third-party cookies on Chrome. https://privacysandbox.com/news/update-on-the-plan-for-phase-out-of-third-party-cookies-on-chrome/, accessed on July 1, 2026.
[41] Paul Jensen. 2025. Intent to Deprecate and Remove: Protected Audience. https://groups.google.com/a/chromium.org/g/blink-dev/c/k_nubsMb97g?pli=1, accessed on July 1, 2026.
[42] Nikhil Jha, Martino Trevisan, Emilio Leonardi, and Marco Mellia. 2023. On the robustness of topics API to a Re-identification attack. *arXiv preprint arXiv:2306.05094* (2023).
[43] Nikhil Jha, Martino Trevisan, Emilio Leonardi, and Marco Mellia. 2024. Re-identification attacks against the topics API. *ACM Transactions on the Web* 18, 3 (2024), 1–24.
[44] Nikhil Jha, Martino Trevisan, Luca Vassio, and Marco Mellia. 2022. The internet with privacy policies: Measuring the web upon consent. *ACM Transactions on the Web (TWEB)* 16, 3 (2022), 1–24.
[45] Garrett Johnson. 2024. Unearthing Privacy-Enhancing Ad Technologies (PEAT): The Adoption of Google's Privacy Sandbox. *Available at SSRN* (2024). https://doi.org/10.2139/ssrn.4983927
[46] Shunto Kobayashi, Garrett Johnson, and Zhengrong Gu. 2024. Privacy-enhanced versus traditional retargeting: Ad effectiveness in an industry-wide field experiment. *Available at SSRN* (2024).
[47] Victor Le Pochat, Tom Van Goethem, Samaneh Tajalizadehkhoob, Maciej Korczyński, and Wouter Joosen. 2019. Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation. In *Proceedings of the 26th Annual Network and Distributed System Security Symposium (NDSS 2019)*. https://doi.org/10.14722/ndss.2019.23386
[48] Minjun Long and David Evans. 2024. Evaluating Google's Protected Audience Protocol. *arXiv preprint arXiv:2405.08102* (2024).
[49] Jonathan R Mayer and John C Mitchell. 2012. Third-party web tracking: Policy and technology. In *2012 IEEE symposium on security and privacy*. IEEE, 413–427.
[50] Stephen McQuistin, Peter Snyder, Hamed Haddadi, and Gareth Tyson. 2024. A First Look at Related Website Sets. In *Proceedings of the 2024 ACM on Internet Measurement Conference*. 107–113.
[51] Microsoft. 2024. New Privacy-Preserving Ads API coming to Microsoft Edge. https://blogs.windows.com/msedgedev/2024/03/05/new-privacy-preserving-ads-api/, accessed on July 1, 2026.
[52] Chris Mills. 2023. Saying goodbye to third-party cookies in 2024. https://developer.mozilla.org/en-US/blog/goodbye-third-party-cookies/, accessed on July 1, 2026.
[53] Alexandra Nisenoff, Deian Stefan, and Nicolas Christin. 2025. Exploiting the Shared Storage API. (2025).
[54] Michiel Philipse, Güne Acar, and Christine Utz. 2024. *Post-Third-Party Cookies: Analyzing Google's Protected Audience API.* Master's thesis. Radboud University, Nijmegen, NL. https://www.cs.ru.nl/masters-theses/2024/M_Philipse___Post-Third-Party_Cookies_Analyzing_Google's_Protected_Audience_API..pdf
[55] Franziska Roesner, Tadayoshi Kohno, and David Wetherall. 2012. Detecting and Defending Against Third-Party Tracking on the Web. In *9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12)*. USENIX Association, San Jose, CA, 155–168. https://www.usenix.org/conference/nsdi12/technical-sessions/presentation/roesner
[56] Asuman Senol and Gunes Acar. 2023. Unveiling the Impact of User-Agent Reduction and Client Hints: A Measurement Study. In *Proceedings of the 22nd Workshop on Privacy in the Electronic Society*. 91–106.
[57] Martin Thomson. 2023. *A Privacy Analysis of Google's Topics Proposal.* Technical Report. Mozilla.
[58] Martin Thomson. 2024. *Protected Audience Privacy Analysis.* Technical Report. Mozilla.
[59] Alberto Verna, Nikhil Jha, Martino Trevisan, and Marco Mellia. 2024. A First View of Topics API Usage in the Wild. In *Proceedings of the 20th International Conference on emerging Networking EXperiments and Technologies*. 48–54.



(a) *After-Accept*



(b) *After-Deny*

Figure 12: Distribution of persistent cookie duration in *After-Accept* and *After-Deny*.

[60] Alberto Verna, Nikhil Jha, Martino Trevisan, and Marco Mellia. 2025. Understanding Topics API in the Wild: Dubious Usage and Stale Adoption. *IEEE Transactions on Privacy* (2025).
[61] Miranda Wei, Madison Stamos, Sophie Veys, Nathan Reitinger, Justin Goodman, Margot Herman, Dorota Filipczuk, Ben Weinshel, Michelle L Mazurek, and Blase Ur. 2020. What Twitter knows: Characterizing ad targeting practices, user perceptions, and ad explanations through users' own Twitter data. In *29th USENIX Security Symposium (USENIX Security 20)*. 145–162.
[62] John Wilander. 2020. Full Third-Party Cookie Blocking and More. https://webkit.org/blog/10218/full-third-party-cookie-blocking-and-more/, accessed on July 1, 2026.
[63] Maximilian Zöllner, Anja Feldmann, and Ha Dao. 2025. A First Look at Cookies Having Independent Partitioned State. In *International Conference on Passive and Active Network Measurement*. Springer, 182–196.

A Cookie duration

Figure 12 compares the cookie duration in *After-Accept* and *After-Deny*, for both partitioned and unpartitioned cookies. Most cookies have a duration of several days, even in the *After-Deny* visit. This strengthens their questionable usage.

B Ethical considerations

This work did not present severe ethical aspects to consider. We ran a crawling campaign to some of the most visited websites in the world, issuing up to six web page visits once a week. We deem

this a negligible volume of traffic, which we expect did not cause any problem to websites of such size.

C Open Science

This work is based on the results of an extensive, specialised Web crawling campaign and subsequent data analysis. To enable independent verification and facilitate reproduction of our results, we release the code and data we produced in the following anonymous Git repository: https://anonymous.4open.science/r/priv-accept-ps-anon-4087. The repository is anonymised and does not contain identifying information about the authors. To minimise privacy risks, we release filtered and processed datasets sufficient to reproduce all figures and results in the paper.

The repository is organised into the following components:

- The `crawler/` folder contains the Python-based crawler used to visit individual websites and collect measurement data in JSON format;
- The `post-process/` folder contains the code used to filter and pre-process the raw Web logs produced by the crawler;
- The `extract-allowed/` folder contains scripts that extract the set of domains that have enrolled to use certain Privacy Sandbox components;
- The `open-data/` folder contains the filtered data, plus the code that analyses it to produce the plots with the same procedure we employed in the paper.

Each component has its own `README.md` file containing additional information.

To support reproducibility of the measurement workflow, we provide a set of scripts that automate the setup and execution of the crawling and analysis pipeline:

- `install-dependencies.sh`, which installs the required software dependencies;
- `build.sh`, which builds the repository's components as Docker containers;
- `analyze-ps-complete.sh`, which executes the complete measurement campaign over a configurable set of N websites and produces the final analysis outputs as zipped CSV files.

These scripts were verified to work on a fresh installation of Ubuntu 22.04 LTS.

D AI Assistance Disclosure

The authors used generative AI-based tools to revise the text, improve flow and correct any typos, grammatical errors, and awkward phrasing. The authors are fully responsible for the content of the paper, including the study design, data collection, analysis, interpretation of results, and all conclusions.