# The browser: the key to your privacy on the Web

Walter Rudametkin, Olivier Zendra

Currently, and in the Next Computing Paradigm (NCP) as well, one of the major ways of interacting with the Web is through the browser. But tracking users through their browser is commonplace, and often underestimated, with the user not being aware of them. The browser is thus a keystone to user privacy in the NCP.

# The browser: the key to your privacy on the Web

**by Walter Rudametkin and Olivier Zendra**

## Key insights

- User tracking through the browser is extremely widespread. Users are tracked almost all the time, on most websites.
    - first-party and third-party cookies are used almost all the time
    - browser fingerprinting is used less often but is powerful and complementary to cookies
- IP address tracking, combined with fingerprinting, is a very effective for replacing third-party cookies
- Tools exist to check that user preferences for 3rd-party cookies are respected, but they are not easily available to end-users.
- User preferences for controlling fingerprinting are non-existent, whereas browser fingerprinting falls into the same GDPR data category as cookies and other trackers
- Browser fingerprinting has security uses that websites actively use
- Vendors are adding new functionalities to browsers at a frenetic pace, not enough effort is being put into minimizing the privacy risks

## Key recommendations

- Promote research into technologies and tools that break the uniqueness of people when browsing, hence preserve their privacy.
- Promote 3rd-party cookies interdiction by the browsers.
- Promote research into technologies and tools to identify, display and allow users to limit fingerprinting techniques.
- Enforce GDPR wrt. user tracking, ensuring that the collected data is strictly necessary to the usages accepted by the users. This should include all tracking technologies: cookies, IP tracking, fingerprinting…

## Browser tracking, for good or for bad, it's there

Modern Internet technologies are progressing at an amazing rate and redefining the limits of the Internet. Some research in the field of cybersecurity focuses on identifying new security and privacy threats on the Web, with extensive expertise in an identification technique that can be exploited for security and tracking purposes, called browser fingerprinting. In this article, we consider two facets. The first attempts to improve authentication and explores the use of browser fingerprinting to improve the security of websites and user accounts by verifying elements of their device's configuration. The second explores how novel features being quickly added to browser's can be exploited to enhance fingerprinting and track users without their knowledge, and also tries to understand how different tracking technologies combine into persistent, omnipresent tracking mechanisms. This is particularly important as cookie-based tracking, which is very prevalent and based on third-party cookies, is inevitably going to be being deprecated [1], sometime in the future (Chrome has pushed the deadline to late 2024 [17]), while browser fingerprinting is potentially "an invasive workaround to replace cookies" [2], and Google, who controls development on the Chrome browser and derivatives [3], exerts pressure to control the future of tracking [4].

The ability to track users on the Internet and their online habits is indeed lucrative to content producers and advertising companies, such as major Big Tech players, as well as being very intrusive to user privacy. Furthermore, users have grown accustomed to "free" services that provide ever more extensive and impressive functions, everything from video services, social media platforms, to video games, office platforms, and much more is now all Web-based. In essence, the Web browser is our window to the Internet and the NCP, our window to the world. Yet, these services are anything but free; aggressive tracking and profiling is pushed for monetization. Studies have shown that user tracking continues to increase on popular websites [7], [8]. State-of-the-art tracking techniques assign a unique identifier, which is stored in the browser—either as a cookie or some other storage mechanism (e.g., local storage, Etags). To protect users, private browsing modes and extensions automatically delete cookies and clear storages at the end of a session, decreasing the efficiency of the standard cookie-based tracking techniques.

But to compensate for this process of deleting cookies and blocking stateful tracking, a new identification technique that leaves no traces on the user's device is being used. It is called Browser fingerprinting [11], [12]. *Browser fingerprinting* is the process of identifying devices by accessing a collection of relatively stable attributes through Web browsers. We call the identifiers browser *fingerprints*. Fingerprints are stateless; no information is stored on the client's device. Browser fingerprinting exploits the diversity of modern web configurations, technologies, protocols and APIs (Application Programming Interfaces) to uniquely identify devices. And contrary to tracking cookies that are stored on the device and can be erased, fingerprints are stored on servers the user has no control over. Encryption does little to limit fingerprinting because it is performed by the website you visit; it is not a sniffing nor man-in-the-middle attack.

Extensive studies of browser fingerprinting, including those from the *Am I Unique* platform [27] for the past 10 years, have identified three main properties of browser fingerprinting that make it both a risk to privacy, but also useful for security. The first property, **uniqueness**, is the power to uniquely identify a device. Fingerprint uniqueness, although not a perfect identifier, has statistical qualities that allow uniquely identifying a high percentage of both desktop computers and mobile devices [12]. The second property, **linkability**, is the capacity to re-identify, or *link*, fingerprints from the same device over time. This is arguably the main risk to privacy and enables *fingerprint tracking*. Some devices are highly trackable, while other devices' fingerprints are too similar to be tracked [9]. The third and final property is **consistency**, which refers to the capacity to verify the attributes in a fingerprint. Through redundancies, correlations or dependencies, many attributes are verifiable, making them difficult to spoof convincingly. Most countermeasures to browser fingerprinting are identifiable through inconsistencies [10], a useful property for security applications, but also increases the privacy risks for users.

These are real concerns and the use of browser fingerprinting for security could be a net positive to companies, to universities and to society. However, abuses may arise. Finding a good balance between security, usability and the risks to privacy is a major and fundamental challenge to get a better, safer, more privacy friendly internet. This is important to avoid social cooling, whose concept is simple: *if you feel you're being watched, you change your behavior*, which is, in essence, a form of restricting

your freedom. Among the things that change, are what you say and how you act. Through what is known as surveillance capitalism, many people are discovering that everything we do is being monitored, dissected and monetized.

This article addresses two main aspects that are both complementary yet opposing by nature: browser identification for security purposes, and browser identification for tracking purposes. The first aspect will focus on ***i)*** **browser fingerprinting for Web authentication**. The main objective is to enhance and augment multi-factor authentication through advanced browser fingerprinting. This requires identifying APIs that can be used to create hard-to-forge browser fingerprints and authentication algorithms that resist different attack models. The second aspect will focus on ***ii)*** **stateless tracking techniques that** create privacy risks in the browser. Since browser fingerprinting is rarely used by itself, we'll explain how it complements other tracking techniques like IP addresses and cookie-based tracking.

## Browser fingerprinting for authentication

Advanced browser fingerprinting can be used as a configurable authentication mechanism. It even has the potential to be the only authentication mechanism when used in very low-security, public websites on the Internet. It can be used to block bots [6] and other fraudulent users. It also has the potential to be an additional security factor in Multi-Factor Authentication (MFA) schemes. Besides strengthening a session's initial authentication, it can be used for continuous session authentication [13] to protect against session hijacking attacks [14]. In many contexts fingerprinting is fully transparent to users, and unlike security cards, code generating keys, apps, SMS verification codes, users do not have to do anything to improve their security. In more restricted contexts, administrators can even enforce different policies, such as enrolling fingerprints from devices that connect from trusted IP addresses (e.g., in an internal enterprise network), and then verifying these fingerprints when the same users connect from untrusted IP addresses. Plugging the browser fingerprinting authentication process into existing authentication systems raises issues, such as fingerprint forgeability, usability and effectiveness. These could be addressed with attributes that focus on identifying hard-to-forge hardware characteristics of the device and by designing dynamic challenge-response tests to limit replay attacks.

But building browser fingerprints that are beneficial to authentication raises issues. While modern browsers are strengthening the protection of their users' privacy by deploying defenses against potential privacy leaks, the evolution of Web programming technologies pushes towards personalization that inevitably leads to the disclosure of sensitive parameters. This continuous tension between technological evolution and browser engagement stresses browser fingerprinting techniques to exploit a moving target set of fingerprintable attributes. Moreover, websites that wish to use fingerprinting to improve the security of their users and services, e.g. as a second factor authentication, must decide what attributes to collect, yet, since fingerprints are calculated on the client's device, unforgeable attributes that resist against motivated attackers that can intercept client requests and replay their fingerprints are needed.

Furthermore, online fraudsters, malicious users and crawlers represent a growing share of Internet traffic. State-of-the-art protections rely on analyzing the traffic to detect suspicious patterns and fallback to CAPTCHAs [18] to block unwanted activity [19][20][21][22]. However, these techniques offer only partial protection at the cost of expensive monitoring. They tend to be slow, ineffective to distributed crawlers, and miss much of the actual fraudster traffic.

Websites are also strengthening their authentication processes to ensure only valid users can authenticate a session. To increase security, Multi-Factor Authentication (MFA) combines factors (e.g, credentials, SMS, smartphone apps) to prevent credentials from being used maliciously. However, these additional factors add cost (e.g., purchasing USB keys for all users) and impact the user experience (by forcing interactions with multiple devices to authenticate), which limits their adoption. Browser fingerprinting can mitigate many of these issues because a fingerprint can be connected before accessing content or an account, and can immediately detect fraudsters, in one step. Furthermore, users don't have to bear any costs and, in the case of legitimate users, this is transparent as long as the false positives are low.

## Browser tracking: you're just a little different but that's enough for them to know who you are

Data collection from users of the Am I Unique browser extensions show that minor fabrication differences in GPUs can be used to uniquely identify devices despite being otherwise identical hardware, similar to research on clock differences [15]. Users who customize their computers or browsers, or their different tools, have fingerprints that are more unique than those that don't. In some cases, users are confronted with a paradox: do I modify my configuration to protect my privacy from, for example, cookies? Should I move to a privacy friendly operating system and browser? Usually this would be the obvious choice and will work for some techniques, such as blocking your cookies, but when done poorly, this can lead to differences in the browser fingerprint making you more identifiable. Furthermore, moving from popular technological choices to less popular but more privacy protecting has the same effect of increasing your browser fingerprint uniqueness. Interestingly, even in the case of ad blockers, which increase their privacy in general, could in fact make you more identifiable through fingerprintable customizations, despite the overall privacy advantages of the tool.

Most uses of browser fingerprinting for tracking purposes occur in conjunction with other techniques, such as cookie-based tracking, invisible pixels, IP address tracking, the collection of personal identifiable information (PII), and many others. This leads to a dire situation for privacy on the Web. Browser fingerprinting is a far from perfect tracking technique, but combined with cookie identifiers, IP addresses, ETags, invisible pixels, PII, and many others, it's much worse for privacy. It's indeed difficult to properly isolate tracking techniques, such as browser fingerprinting, especially when the identifiers are collected and stored server-side. IPv4 address retention is a serious problem [16], and combining this with browser fingerprinting is likely to be very effective for tracking. For example, to avoid tracking, if users erase all their cookies, they would simultaneously need to clear all caches, change their browser fingerprint, change their IP address (e.g., restart their router and obtain a new DHCP address from their ISP or use a VPN), not provide any PII, to have any strong assurance that they are not potentially and immediately reidentified. Furthermore, with third party cookies being deprecated, these techniques risk being further exploited and novel tracking techniques, such as server-side tracking, developed and perfected. This is not a theoretical approach, Meta already provides a new server-side tracking product called the Conversions API [23] where websites can collect IP addresses, user agents, and personal identifiers and send them to Meta. Moreover, the "best practices" call for combining the Conversions API with their client-side tracker [24], the Meta Pixel [25], that now supports both first and third party cookies. You read this right, first and third party cookies, fingerprinting, IP addresses and PII are all combined to track users across any domain that uses Meta's advertising products. And Google has its own server-side product that is more extensible and configurable than Meta's called server-side tagging [26].

This begs the question, how do people effectively defend against a barrage of privacy invasive attacks? The simple answer is that there are no simple answers.

Browsers and their features are rapidly developed, with little interest or caution to privacy issues, opening the door to fingerprinting and other side-channel attacks. Browsers are very configurable and different configurations can introduce many issues. Furthermore, browsers are extensible through extensions, and these extensions can introduce privacy bugs of their own that make users identifiable (e.g., in FP-Scanner [10] we showed privacy focused extensions to often be counter-productive despite their promises). This field of research is of even more importance now that a few major actors on the Web – all of Big Tech are outside of the EU – have concentrated control over entire swaths of the Web, and have begun to push to a more closed, more tracked, more controlled and privacy destructive future. EU researchers and regulators need to step up and attack these issues from all angles, legal, technical, economical and societal.

## Conclusion

The browser is one of the most important user interfaces to the NCP, that is used daily by billions of people without any second thoughts. However, browser tracking techniques can both be used for ease of use, or cause tremendous privacy issues. It is thus crucial that research and regulation address the issues of tracking, provide the users with legal rights and the technical means to verify them, including easy to use tools to detect, control, verify and limit undesirable tracking in all its forms.

## References


[1] Building a more private web: A path towards making third party cookies obsolete. Chromium Blog. 2020-01-14 (accessed 2023-11-28). https://blog.chromium.org/2020/01/building-more-private-web-path-towards.html

[2] Building a more private web. Justin Schuh. 2019-08-22 (accessed 2023-11-28). https://blog.google/products/chrome/building-a-more-private-web/

[3] Browser Market Share Worldwide. statcounter GlobalStats. (accessed 2023-11-28). https://gs.statcounter.com/browser-market-share

[4] Google's FLoC Is a Terrible Idea. EFF. Bennett Cyphers. 202-03-03 (accessed 2023-11-28). https://www.eff.org/deeplinks/2021/03/googles-floc-terrible-idea

[5] Big Tech. Wikipedia. (accessed 2023-11-28). https://en.wikipedia.org/wiki/Big_Tech

[6] A. Vastel, W. Rudametkin, R. Rouvoy, X. Blanc. FP-Crawlers: Studying the Resilience of Browser Fingerprinting to Block Crawlers. MAD-Web'20 - NDSS Workshop on Measurements, Attacks, and Defenses for the Web. https://hal.archives-ouvertes.fr/hal-02441653

[7] G. Acar, M. Juarez, N. Nikiforakis, C. Diaz, S. Gurses, F. Piessens, and B. Preneel, "FPDetective: Dusting the web for fingerprinters," Conf. on Computer and Communications Security (CCS '13). https://www.esat.kuleuven.be/cosic/publications/article-2334.pdf

[8] S. Englehardt and A. Narayanan, "Online tracking: A 1-million-site measurement and analysis,". CCS '16. http://randomwalker.info/publications/OpenWPM_1_million_site_tracking_measurement.pdf

[9] A. Vastel, P. Laperdrix, W. Rudametkin, R. Rouvoy. FP-STALKER: Tracking Browser Fingerprint Evolutions. IEEE Symp. on Sec. and Privacy (IEEE S&P'18). https://hal.inria.fr/hal-01652021

[10] A. Vastel, P. Laperdrix, W. Rudametkin, R. Rouvoy. FP-Scanner: The Privacy Implications of Browser Fingerprint Inconsistencies. USENIX Security 2018. https://hal.inria.fr/hal-01820197

[11] P. Eckersley. How unique is your web browser? International Conference on Privacy Enhancing Technologies (PETS'10). https://panopticlick.eff.org/static/browser-uniqueness.pdf

[12] P. Laperdrix, W. Rudametkin, B. Baudry. Beauty and the Beast: Diverting modern browsers to build unique browser fingerprints. IEEE Symposium on Security and Privacy (S&P'2016). https://hal.inria.fr/hal-01285470v2

[13] D. Preuveneers, W. Joosen. SmartAuth: dynamic context fingerprinting for continuous user authentication. Symposium on Applied Computing SAC'15. https://dl.acm.org/citation.cfm?id=2695908

[14] A. Durey, P Laperdrix, W Rudametkin, R. Rouvoy. "FP-Redemption: Studying Browser Fingerprinting Adoption for the Sake of Web Security." DIMVA'21. https://hal.inria.fr/hal-03269174

[15] I. Sanchez-Rola, I. Santos, D. Balzarotti. Clock Around the Clock: Time-Based browser fingerprinting. Conference on Computer and Communications Security (CCS '18). https://dl.acm.org/citation.cfm?id=3243796

[16] V. Mishra, P. Laperdrix, A. Vastel, W. Rudametkin, R. Rouvoy, M. Lopatka.. Don't count me out: On the relevance of IP addresses in the tracking ecosystem. The Web Conference 2020 (WWW'20). https://hal.inria.fr/hal-02435622

[17] The Privacy Sandbox Timeline for the Web. 2023-11 (accessed 2023-11-30). https://privacysandbox.com/intl/en_us/open-web/#the-privacy-sandbox-timeline

[18] L. Von Ahn, M. Blum, N. J. Hopper, and J. Langford, "Captcha: Using hard ai problems for security," in International Conference on the Theory and Applications of Cryptographic Techniques. Springer, 2003, pp. 294–311

[19] G. Wang, T. Konolige, C. Wilson, X. Wang, H. Zheng, and B. Y. Zhao, "You are how you click: Clickstream analysis for sybil detection." in USENIX Security Symposium, vol. 9, 2013, pp. 1–008.

[20] A. Stassopoulou and M. D. Dikaiakos, "Web robot detection: A probabilistic reasoning approach," Computer Networks, vol. 53, no. 3, pp. 265–278, 2009

[21] D. Stevanovic, A. An, and N. Vlajic, "Feature evaluation for web crawler detection with data mining techniques," Expert Systems with Applications, vol. 39, no. 10, pp. 8707–8717, 2012.

[22] A. Balla, A. Stassopoulou, and M. D. Dikaiakos, "Real-time web crawler detection," in Telecommunications (ICT), 2011 18th International Conference on. IEEE, 2011, pp. 428–432.

[23] Conversion API. (accessed 2023-12-04). https://developers.facebook.com/docs/marketing-api/conversions-api/

[24] Best practices for Conversions API. (accessed 2023-12-04). https://www.facebook.com/business/help/308855623839366?id=818859032317965

[25] About cookie settings for the Meta pixel. (accessed 2023-12-04). https://www.facebook.com/business/help/471978536642445?id=1205376682832142

[26] An introduction to server-side tagging. (accessed 2023-12-04). https://developers.google.com/tag-platform/tag-manager/server-side/intro

[27] Am I Unique ? (accessed 2023-12-04). https://amiunique.org



**Walter Rudametkin** is a Professor in Computer Science at the University of Rennes, France.

**Olivier Zendra** is a Tenured Computer Science Researcher at Inria, Rennes, France.