arXiv:2005.09353v2 [cs.CR] 10 Jun 2020

"Guess Who ?" Large-Scale Data-Centric Study of the Adequacy of Browser Fingerprints for Web Authentication

Nampoina Andriamilanto, Tristan Allard, and Gaëtan Le Guelvouit

Abstract Browser fingerprinting consists in collecting attributes from a web browser to build a browser fingerprint. In this work, we assess the adequacy of browser fingerprints as an authentication factor, on a dataset of $4,145,408$ fingerprints composed of 216 attributes. It was collected throughout 6 months from a population of general browsers. We identify, formalize, and assess the properties for browser fingerprints to be usable and practical as an authentication factor. We notably evaluate their distinctiveness, their stability through time, their collection time, and their size in memory. We show that considering a large surface of 216 fingerprinting attributes leads to an 81.8% unicity rate on a population of $1,989,365$ browsers. Moreover, browser fingerprints are known to evolve, but we observe that between consecutive fingerprints, more than 90% of attributes remains unchanged after nearly 6 months. Fingerprints are also affordable. On average, they weight a dozen of kilobytes, and are collected in a few seconds. We conclude that browser fingerprints are a promising additional web authentication factor.

1 Introduction

Web authentication widely relies on identifier-password pairs. Passwords are easy to use, but suffer from severe security flaws. Indeed, users use common passwords, paving the way to brute-force or guessing attacks [1]. They also use similar pass-

Nampoina Andriamilanto
Institute of Research and Technology b<>com, Rennes, France e-mail: nampoina.andriamilanto@b-com.com

Tristan Allard
Univ Rennes, CNRS, IRISA, Rennes, France e-mail: tristan.allard@irisa.fr

Gaëtan Le Guelvouit
Institute of Research and Technology b<>com, Rennes, France e-mail: gaetan.leguelvouit@b-com.com

words across websites [15], which increases the impact of attacks. Phishing attacks are also a major threat to passwords. Over the course of a year, Thomas et al. [13] achieved to retrieve 12.4 million credentials stolen by phishing kits. These flaws gave rise to multi-factor authentication [2], such that each additional authentication factor provides an *additional security barrier*. However, this usually comes at the cost of *usability* (i.e., users have to remember, possess, or do something).

In the meantime, *browser fingerprinting* gains attention. The Panopticlick study [3] highlights the possibility to build a *browser fingerprint* by collecting attributes from a web browser. In addition to being widely used for web tracking purposes [4] (raising legal and ethical issues), browser fingerprints are used as an authentication factor *in real-life*. Browser fingerprints are indeed good a *candidate* as an authentication factor thanks to their distinctive power, their frictionless deployment (e.g., no additional software), and their usability (no secret to remember, no additional object to possess, and no supplementary action to carry out). As a result, companies like MicroFocus[1] or SecureAuth[2] include browser fingerprints within their authentication mechanisms (see Figure 1 for an example of such mechanism).

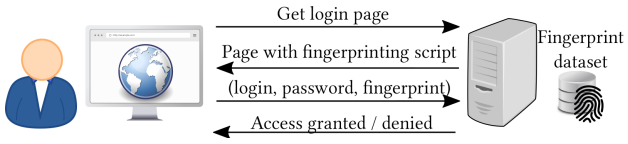

Fig. 1 Simplified browser fingerprinting web authentication mechanism.

Related works. To the best of our knowledge, no large-scale study rigorously evaluates the adequacy of browser fingerprinting as an authentication factor. Most works about their use for authentication concentrate on the design of authentication mechanism [14, 10, 6, 11], and empirical studies on browser fingerprints focus on their efficacy as a web tracking tool [3, 7, 5]. Such a mismatch between the understanding of browser fingerprints for authentication – currently poor – and their ongoing adoption in real-life is a serious harm to the security of web users. The lack of documentation from the existing tools (e.g., about the used attributes, the distinctiveness, and the stability of the resulting fingerprints) only adds up to the current state of ignorance. All this whereas security-by-obscurity contradicts the most fundamental security principles.

Our contributions. We conduct the first *large-scale data-centric empirical study of fundamental properties* of browser fingerprints when used as an additional authentication factor. We base our findings on an in-depth analysis of a real-life fingerprint dataset collected over 6 months, that contains 4,145,408 fingerprints composed of 216 attributes. We formalize, and assess on our dataset, the properties nec-

[1] https://www.microfocus.com/media/white-paper/device-fingerprinting-for-low-friction-authentication-wp.pdf

[2] https://docs.secureauth.com/pages/viewpage.action?pageId=33063454

essary for paving the way to elaborate browser fingerprinting authentication mechanisms. The selected properties are usually used to evaluate biometric characteristics for authentication [8]. We stress that we do not make any assumption on the inner working of the authentication mechanism, and consequently on the adversarial strategy. Our properties aim at characterizing the adequacy and practicability of browser fingerprints, independently of their use within future authentication mechanisms. In particular, we measure the size of browser anonymity sets through time, and show that 81.8% of our fingerprints are unique. Moreover, we measure the proportion of identical attributes between two observations of the fingerprint of a browser, and show that 90% of attributes remains unchanged after nearly 6 months. Finally, we measure the collection time and the size of fingerprints. We show that on average, they weight a dozen of kilobytes, and are collected in a few seconds.

The rest of the paper is organized as follows. Section 2 presents and formalizes the properties evaluated in our analysis. Section 3 describes the dataset analyzed in this study. Section 4 presents our experimental results. Finally, Section 5 synthesizes the results and concludes.

2 Authentication Factor Properties

The "Handbook of Fingerprint Recognition" [8] summarizes properties that a biometric characteristic requires to be *usable*[3] as an authentication factor, and additional properties required for a biometric authentication scheme to be *practical*. We make the connection between fingerprints used to recognize persons, and browser fingerprints used to recognize browsers. So we evaluate browser fingerprints according to these properties, to assert their adequacy for web authentication. In this section, we list these properties, formalize how to measure some properties, and explain why the others are not addressed in this study.

The four properties needed for an anatomical or a behavioral characteristic to be usable as a biometric authentication factor are described below.

- *Universality*: the characteristic should be present in everyone.
- *Distinctiveness*: two distinct persons should have different characteristics.
- *Permanence*: the same person should have the same characteristic over time. We rather use the term *stability*.
- *Collectibility*: the characteristic should be collectible and measurable.

The three properties that a biometric authentication scheme requires to be practical are the following.

- *Performance*: the scheme should consume few resources, and be robust against environmental changes.
- *Acceptability*: the users should accept to use the scheme in their daily lives.

[3] Here, *usable* refers to the adequacy of the characteristic to be used for authentication, rather than the ease of use by users.

- *Circumvention*: it should be difficult for an attacker to deceive the scheme.

The properties that we study are the *distinctiveness*, the *stability*, and the *performance*. We consider that the *universality* and the *collectibility* are satisfied, as the HTTP headers that are automatically sent by browsers constitute a fingerprint. However, we stress that a loss of distinctiveness occurs when no JavaScript attribute is available. About the *circumvention*, we refer the reader to Laperdrix et al. [6] that analyzed the security of an authentication mechanism based on browser fingerprints. We let the evaluation of the *acceptability* as future works, but we stress that such mechanisms are already used in a rudimentary form[4].

2.1 Distinctiveness

To satisfy the *distinctiveness* property, browser fingerprints should enable two different browsers to be distinguishable. The distinctiveness depends on the used attributes, and on the fingerprinted browser population. The two extreme cases are every browser sharing the same fingerprint, which makes them indistinguishable from each other, and no two browsers sharing the same fingerprint, making every browser distinguishable.

Our dataset entries are composed of a fingerprint, the source browser, and the time of collection in the form of a Unix timestamp in milliseconds. We denote B the domain of the unique identifiers (UIDs), and T the timestamp domain. The fingerprint dataset is denoted D and is formalized as:

$$D = \{(f,b,t) \mid f \in F, b \in B, t \in T\} \tag{1}$$

We use the size of browser anonymity sets to quantify the distinctiveness, as browsers belonging to the same anonymity set are indistinguishable. We denote $S(f,D)$ a function returning the set of browsers that provide the fingerprint f in the dataset D. It is formalized as:

$$S(f,D) = \{b \in B \mid \forall (g,b,t) \in D, f = g\} \tag{2}$$

We denote $A(\varepsilon,D)$ a function providing the set of fingerprints having an anonymity set size of ε (i.e., being shared by ε browsers) in the dataset D. It is formalized as:

$$A(\varepsilon,D) = \{f \in F \mid \text{card}(S(f,D)) = \varepsilon\} \tag{3}$$

We measure the anonymity set sizes on the fingerprints currently in use by each browser, and not on their whole history. It is performed by simulating datasets composed of the last fingerprint seen for each browser at a given time. Let $E_\tau(D)$ be the simulated dataset originating from D that represents the state of the fingerprints after τ days. With t_τ the last timestamp of this day, we have:

[4] `https://support.google.com/accounts/answer/1144110`

$$E_\tau(D) = \{(f_i, b_j, t_k) \in D \mid \forall (f_p, b_q, t_r) \in D, t_r \leq t_k \leq t_\tau\} \tag{4}$$

2.2 Stability

Browser fingerprints have the particularity of evolving through time, due to changes in the web environment like a software update or a user configuration. We measure the *stability* by the mean similarity between two consecutive fingerprints coming from a browser, given the elapsed time between them. The two extreme cases are every browser holding the same fingerprint through its life, and the fingerprint of a browser changing completely at each observation.

We denote $C(\Delta, D)$ a function providing the set of consecutive fingerprints in D that are separated by a time difference comprised in the Δ time range. It is formalized as:

$$\begin{aligned} C(\Delta, D) = \{(f_i, f_p) \mid \forall ((f_i, b_j, t_k), (f_p, b_q, t_r)) \in D^2, \\ b_j = b_q, t_k < t_r, (t_r - t_k) \in \Delta\} \end{aligned} \tag{5}$$

We consider the Kronecker delta $\delta(x, y)$, being 1 if x equals y, and 0 otherwise. We denote $f[\omega]$ the value taken by the attribute ω for the fingerprint f. Let $\mathrm{sim}(f, g)$ be a simple similarity function between fingerprints, formalized as:

$$\mathrm{sim}(f, g) = \frac{1}{n} \sum_{\omega=1}^{n} \delta(f[\omega], g[\omega]) \tag{6}$$

We define the function $\mathrm{meansim}(\Delta, D)$ providing the mean similarity of the consecutive fingerprints, for a given time range Δ and a dataset D, as:

$$\mathrm{meansim}(\Delta, D) = \frac{\sum_{(f,g) \in C(\Delta, D)} \mathrm{sim}(f, g)}{\mathrm{card}(C(\Delta, D))} \tag{7}$$

2.3 Performance

To evaluate the performance of browser fingerprints used in an authentication context, we consider three aspects. The first two are the consumption of time and memory resources. The third is the loss of efficacy (i.e., distinctiveness and stability) among device types.

The collection time of fingerprints only depend on JavaScript attributes, as HTTP headers are transmitted anyway. So we measure the *collection time* of our fingerprints composed of 200 JavaScript attributes.

The size of fingerprints depends on their storage format. For example, a canvas [9] image can be encoded as a base64 string or as a hash. We stress that compressing the complete fingerprint to a single hash is unpractical due to the evolution

of fingerprints. The size of attributes is not specified, hence we measure the *size* of the fingerprints of our dataset.

Previous works showed that *mobile and desktop devices* present differences in the properties of their browser fingerprints [12, 7, 5]. Mobile browsers usually have less distinctive fingerprints. Following these findings, we assess that the distinctiveness and the stability of the fingerprints of these two groups are similar.

3 Fingerprint Dataset

To study the properties of browser fingerprints on a real-world browser population, we launched a fingerprint collection experiment. It was performed in collaboration with the authors of [5], and an industrial partner that controls one of the top 15 French websites according to Alexa[5]. The authors of [5] held the 17 attributes of their previous work [7] and focused on web tracking, whereas we held 216 attributes and focused on web authentication.

	PTC [3]	AIU [7]	HITC [5]	**This study**
Collection period	3 weeks	3-4 months*	6 months	**6 months**
Number of attributes	8	17	17	**216**
Number of browsers	-	-	-	**1,989,365**
Number of fingerprints	470,161	118,934	2,067,942	**4,145,408**
Number of distinct fingerprints	409,296	142,023[6]	-	**3,578,196**
Proportion of desktop fingerprints	-	0.890*	0.879	**0.805**
Proportion of mobile fingerprints	-	0.110*	0.121	**0.134**
Unicity of global fingerprints	0.836	0.894	0.336	**0.818**
Unicity of mobile fingerprints	-	0.810	0.185	**0.399**
Unicity of desktop fingerprints	-	0.900	0.357	**0.884**

Table 1 Dataset comparison between Panopticlick, AmIUnique, Hiding in the Crowd, and this study. **-** denotes missing information. ***** denotes deduced information. The attributes only comprises original ones, and fingerprints are counted after data preprocessing.

3.1 Fingerprint Collection

We compiled 200 JavaScript properties and 16 HTTP header fields, and designed a fingerprinting probe that collects these attributes. We integrated the probe to two general audience web pages of our industrial partner, which subjects are political

[5] `https://www.alexa.com/topsites/countries/FR`

[6] This number is provided in Figure 11 as the distinct fingerprints, but also corresponds to the raw fingerprints. Every fingerprint would be unique if the number of distinct and collected fingerprints are equal, hence we are not confident in this number, but it is the one provided by the authors.

news and weather. The probe collected fingerprints from December 7, 2016, to June 7, 2017. Only the visitors that consented to cookies were fingerprinted, in compliance with the European directives 2002/58/CE and 2009/136/CE in effect at the time. To differentiate browsers, we assigned them a unique identifier (UID) as a 6-months cookie. Similarly to [3, 7], we coped with cookie deletion by storing a one-way hash of the IP address, computed by a secure cryptographic hash function.

Previous datasets were collected through dedicated websites, and are biased towards privacy-aware and technically-skilled persons [3, 7]. Our population is more general audience oriented, but the website audience is mainly French-speaking users. This leads to a bias towards this population. The timezone is set to -1 for 98.48% of browsers, 98.59% of them have daylight saving time enabled, and `fr` is present in 98.15% of the `Accept-Language` HTTP header value.

3.2 Dataset Filtering and Preprocessing

Given the experimental aspect of fingerprints and the scale of our collection, the raw dataset contained erroneous or irrelevant samples. We remove $70,460$ entries entries that have a wrong format (e.g., empty or truncated data), that are duplicated, or that come from a robot.

Cookies are an unreliable identification method, hence we perform a resynchronization similar to [3]. We consider the entries that have the same (fingerprint, IP address hash) pair to come from the same browser, and assign them the same UID. Similarly to [3], we do not synchronize the interleaved UIDs, being the pairs that have UID values b_1, b_2, then b_1 again. We replace $181,676$ UIDs with $116,708$ replacement UIDs using this method.

To avoid counting multiple entries of identical fingerprints coming from the same browser, the usual way is to ignore them during collection [3, 7]. Our probe collects fingerprint on each visit, and to stay consistent with common methodologies we deduplicate the fingerprints afterward. For each browser, we hold the first entry having a given fingerprint, and ignore the following entries if they have this fingerprint. For example, if a browser b has the entries $\{(f_1,b,t_1),(f_2,b,t_2),(f_2,b,t_3),(f_1,b,t_4)\}$, we only hold the entries $\{(f_1,b,t_1),(f_2,b,t_2),(f_1,b,t_4)\}$. The deduplication constitutes the biggest cut in our dataset, with $2,420,217$ entries filtered out.

We extract 46 additional attributes from 9 original attributes, which are of two types. The first type consists in extracted attributes composed of parts of original attributes, like the screen resolution that is split into the values of width and height. The second type consists of information sourced from an original attribute, like the number of plugins extracted from the list of plugins.

3.3 Work Dataset

The work dataset obtained after the preprocessing step contains 5,714,738 entries (comprising identical fingerprints for a given browser if interleaved), with 4,145,408 fingerprints (no identical fingerprint counted for the same browser), composed of 3,578,196 distinct fingerprints. The fingerprints are composed of 216 original attributes and 46 extracted ones, for a total of 262 attributes. They come from 1,989,365 browsers, 27.53% of which have multiple fingerprints. Table 1 presents a comparison between the dataset of Panopticlick [3], AmIUnique [7], Hiding in the Crowd [5], and this study.

4 Empirical Evaluation of Browser Fingerprints Properties

4.1 Distinctiveness

Figure 2 presents the size of the anonymity sets (AS) alongside the frequency of browser arrival for the daily-partitioned datasets. We call unicity rate the proportion of fingerprints that belong to an AS of size one. Our fingerprints have a stable unicity rate of approximately 81.3% on the long run, and at least 94.7% of fingerprints are shared by 8 browsers or less. However, the fingerprints of the mobile group are more uniform than that of the desktop group, with a unicity rate of approximately 42% on the long run.

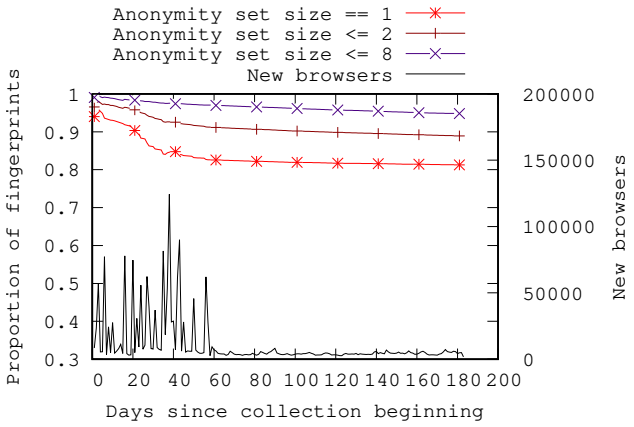

Fig. 2 Anonymity set sizes and frequency of browser arrivals through partitioned datasets obtained after each day.

New browsers are encountered continually, but starting from the 60th day, the arrival frequency stabilizes around 5,000 new browsers per day. Before this stabi-

lization, we have a variable arrival frequency with some major spikes. They seem to correspond to events having happened in France that lead to more visits. For example, the spike on the 38th day corresponds to a live political debate on TV, and the spike on the 43rd day correlates with the announcement of a cold snap.

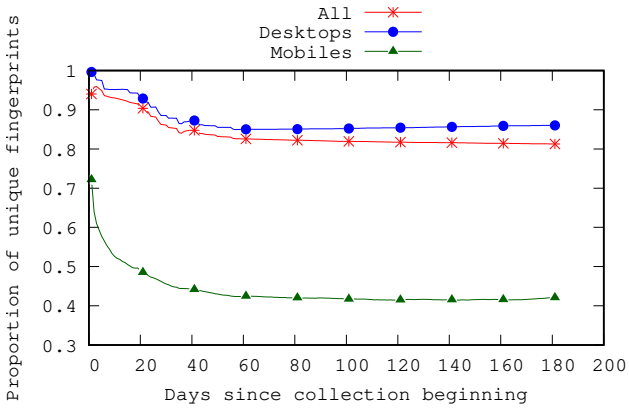

Fig. 3 Proportion of unique fingerprints for overall, mobile, and desktop groups, through partitioned datasets obtained after each day.

Figure 3 presents the proportion of unique fingerprints through partitioned datasets for overall, mobile, and desktop groups. The proportion of unique fingerprints is stable for the desktop browsers, with a slight increase of 1.04 points from the 60th day to the 183th, from 84.99% to 86.03%. The unicity rate of the fingerprints of the mobile group is lower than that of the desktop group, and has a little decrease of 0.29 points on the same period, from 42.42% to 42.13%.

4.2 Stability

Two fingerprints of a given browser can be linked as only a small portion of the attributes is expected to change, even after several months. Figure 4 displays the mean similarity between consecutive fingerprints in function of the time difference. The ranges Δ are expressed in days, so that day d on the x-axis represents the fingerprints separated by $\Delta = [d; d+1[$ days. We ignore the comparisons of time ranges having less than 10 pairs, or with a time difference higher than the limit of our experiment (182 days), which account for less than 0.03% of each category. Our stability results are a lower bound, as consecutive fingerprints are necessarily different (i.e., their similarity is strictly lower than 1).

We have a total of $3,725,373$ compared pairs for the overall group, $2,912,860$ pairs for the desktop group, and $594,591$ pairs for the mobile group. A fingerprint is expected to have at least 90% of its attributes having an identical value after 170 days.

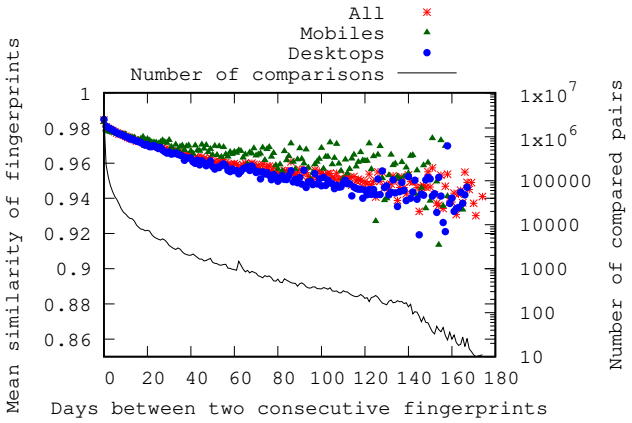

Fig. 4 Mean similarity between consecutive fingerprints in function of the time difference, with the number of compared pairs.

Few attributes among those included in our script are highly unstable. Getting rid of these attributes could reduce the distinctiveness of fingerprints, but would improve their stability.

The fingerprints of the mobile group are generally more stable than these of the desktop group, as suggests their respective similarity curve. However, it seems to be the case only for high time differences, as in the range of $[1;2[$ days difference, the fingerprints of the mobile group are slightly less stable than these of the desktop group with a mean similarity of 97.87% against 98.11%.

4.3 Performance

4.3.1 Time Resource Consumption

Our script takes several seconds to collect the attributes composing the fingerprints. Figure 5 displays the cumulative distribution of the collection time of fingerprints in seconds, with the outliers removed. We measure it by the time difference between the starting of the script and the fingerprint sending. Some values take from several hours to days, that can come from a web page put in background or accessed after a long time. We limit our population to the fingerprints that take less than 30 seconds to collect, and consider higher values as outliers. Outliers account for less than 1% of each group.

We present the collection time of fingerprints in seconds for the (5^{th} percentile, median, 95^{th} percentile). Our script collects most fingerprints within a few seconds, with values (0.66, 2.92, 10.42). A difference occurs between the desktop browsers (0.61, 2.64, 10.45) and the mobile browsers (2.06, 4.44, 10.16). The median collection time is less than the estimated median time taken by web pages to

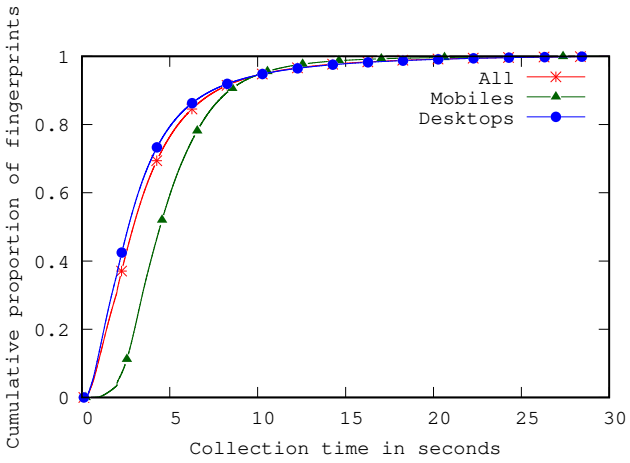

Fig. 5 Cumulative distribution of the collection time of fingerprints in seconds.

load completely[7], being at 6.6 seconds for the desktop browsers and 19.5 seconds for the mobile browsers, at the date of February 1, 2020.

4.3.2 Memory Resource Consumption

Our script consumes a dozen of kilobytes per fingerprint, a size easily handled by the current storage and bandwidth capacities. Figure 6 displays the cumulative distribution of fingerprint size in bytes, with the outliers removed, and canvases stored as sha256 hashes. The mean fingerprint size is $\mu = 7,692$ bytes, and the standard deviation is $\sigma = 2,294$. We remove 1 fingerprint from a desktop browser considered an outlier because of its size being greater than $\mu + 15 \cdot \sigma$.

Half of our fingerprints take less than 7,550 bytes, and 99% less than 14 kilobytes. It is negligible given the current storage and bandwidth capacities. We observe a difference between the fingerprints of mobile and desktop browsers, with 95% of fingerprints weighing respectively less than 8,020 bytes and 12,082 bytes. This is due to heavy attributes being lighter on mobiles, like the plugins or mime types lists that are most of the time empty.

5 Synthesis of Results and Conclusion

In this study, we evaluate the properties offered by browser fingerprints as an additional web authentication factor, through the analysis of a large-scale real-life fingerprint dataset. We show that browser fingerprints offer a satisfying *distinc-*

[7] `https://httparchive.org/reports/loading-speed#ol`

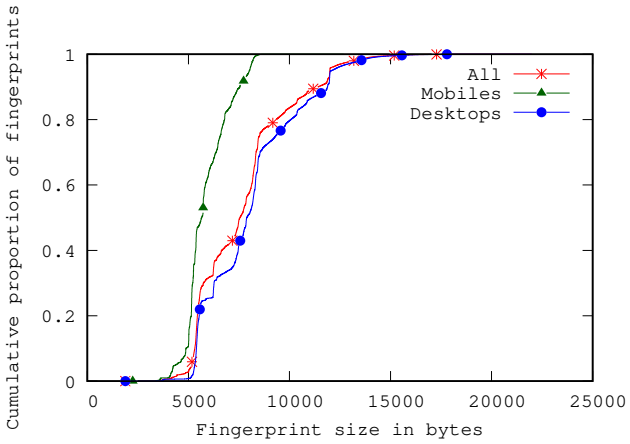

Fig. 6 Cumulative distribution of the size of fingerprints in bytes.

tiveness, as 81.8% of our fingerprints are only shared by one browser. Moreover, fingerprints are *stable*. At least 90% of the attributes are expected to stay identical between two observations of the fingerprint of a browser, even if they are separated by nearly 6 months. We validate that fingerprints offer a high *performance*, as they only weight a dozen of kilobytes and take a few seconds to collect. We conclude that browser fingerprints provide satisfying properties for an additional web authentication factor, and can strengthen password-based systems without a major loss of usability.

Acknowledgements We want to thank Benot Baudry and David Gross-Amblard for their valuable comments, together with Alexandre Garel for his work on the experiment. This is a preprint of a contribution published in *Innovative Mobile and Internet Services in Ubiquitous Computing*, edited by Leonard Barolli, Aneta Poniszewska-Maranda, and Hyunhee Park, and published by Springer International Publishing. The final authenticated version is available online at: `https://doi.org/10.1007/978-3-030-50399-4_16`.

References

1. Bonneau, J.: The science of guessing: analyzing an anonymized corpus of 70 million passwords. In: Symposium on Security and Privacy (2012)
2. Bonneau, J., Herley, C., van Oorschot, P.C., Stajano, F.: Passwords and the Evolution of Imperfect Authentication. Communications of the ACM (2015)
3. Eckersley, P.: How unique is your web browser? In: Privacy Enhancing Technologies (2010)
4. Englehardt, S., Narayanan, A.: Online Tracking: A 1-million-site Measurement and Analysis. In: Conference on Computer and Communications Security (2016)
5. Gómez-Boix, A., Laperdrix, P., Baudry, B.: Hiding in the Crowd: an Analysis of the Effectiveness of Browser Fingerprinting at Large Scale. In: The Web Conference (2018)

6. Laperdrix, P., Avoine, G., Baudry, B., Nikiforakis, N.: Morellian analysis for browsers: Making web authentication stronger with canvas fingerprinting. In: Conference on Detection of Intrusions and Malware & Vulnerability Assessment (2019)
7. Laperdrix, P., Rudametkin, W., Baudry, B.: Beauty and the Beast: Diverting modern web browsers to build unique browser fingerprints. In: Symposium on Security and Privacy (2016)
8. Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S.: Handbook of Fingerprint Recognition. Springer-Verlag (2003)
9. Mowery, K., Shacham, H.: Pixel perfect: Fingerprinting canvas in html5. In: W2SP (2012)
10. Preuveneers, D., Joosen, W.: SmartAuth: Dynamic Context Fingerprinting for Continuous User Authentication. In: Annual Symposium on Applied Computing (2015)
11. Rochet, F., Efthymiadis, K., Koeune, F., Pereira, O.: SWAT: Seamless web authentication technology. In: The World Wide Web Conference (2019)
12. Spooren, J., Preuveneers, D., Joosen, W.: Mobile Device Fingerprinting Considered Harmful for Risk-based Authentication. In: European Workshop on System Security (2015)
13. Thomas, K., Li, F., Zand, A., Barrett, J., Ranieri, J., Invernizzi, L., Markov, Y., Comanescu, O., Eranti, V., Moscicki, A., et al.: Data breaches, phishing, or malware?: Understanding the risks of stolen credentials. In: Conference on Computer and Communications Security (2017)
14. Unger, T., Mulazzani, M., Frhwirt, D., Huber, M., Schrittwieser, S., Weippl, E.: SHPF: Enhancing HTTP(S) Session Security with Browser Fingerprinting. In: Conference on Availability, Reliability and Security (2013)
15. Wang, C., Jan, S.T., Hu, H., Bossart, D., Wang, G.: The next domino to fall: Empirical analysis of user passwords across online services. In: Conference on Data and Application Security and Privacy