How Unique Is Your Web Browser?

Peter Eckersley*

Electronic Frontier Foundation
pde@eff.org

Abstract. We investigate the degree to which modern web browsers are subject to "device fingerprinting" via the version and configuration information that they will transmit to websites upon request. We implemented one possible fingerprinting algorithm, and collected these fingerprints from a large sample of browsers that visited our test side, panopticlick.eff.org. We observe that the distribution of our fingerprint contains at least 18.1 bits of entropy, meaning that if we pick a browser at random, at best we expect that only one in 286,777 other browsers will share its fingerprint. Among browsers that support Flash or Java, the situation is worse, with the average browser carrying at least 18.8 bits of identifying information. 94.2% of browsers with Flash or Java were unique in our sample.

By observing returning visitors, we estimate how rapidly browser fingerprints might change over time. In our sample, fingerprints changed quite rapidly, but even a simple heuristic was usually able to guess when a fingerprint was an "upgraded" version of a previously observed browser's fingerprint, with 99.1% of guesses correct and a false positive rate of only 0.86%.

We discuss what privacy threat browser fingerprinting poses in practice, and what countermeasures may be appropriate to prevent it. There is a tradeoff between protection against fingerprintability and certain kinds of debuggability, which in current browsers is weighted heavily against privacy. Paradoxically, anti-fingerprinting privacy technologies can be self-defeating if they are not used by a sufficient number of people; we show that some privacy measures currently fall victim to this paradox, but others do not.

1 Introduction

It has long been known that many kinds of technological devices possess subtle but measurable variations which allow them to be "fingerprinted". Cameras [1,2], typewriters [3], and quartz crystal clocks [4,5] are among the devices that can be

* Thanks to my colleagues at EFF for their help with many aspects of this project, especially Seth Schoen, Tim Jones, Hugh D'Andrade, Chris Controllini, Stu Matthews, Rebecca Jeschke and Cindy Cohn; to Jered Wierzbicki, John Buckman and Igor Serebryany for MySQL advice; and to Andrew Clausen, Arvind Narayanan and Jonathan Mayer for helpful discussions about the data. Thanks to Chris Soghoian for suggesting backoff as a defence to font enumeration.

M.J. Atallah and N. Hopper (Eds.): PETS 2010, LNCS 6205, pp. 1–18, 2010.

entirely or substantially identified by a remote attacker possessing only outputs or communications from the device.

There are several companies that sell products which purport to fingerprint web browsers in some manner [6,7], and there are anecdotal reports that these prints are being used both for analytics and second-layer authentication purposes. But, aside from limited results from one recent experiment [8], there is to our knowledge no information in the public domain to quantify how much of a privacy problem fingerprinting may pose.

In this paper we investigate the real-world effectiveness of browser fingerprinting algorithms. We defined one candidate fingerprinting algorithm, and collected these fingerprints from a sample of 470,161 browsers operated by informed participants who visited the website `https://panopticlick.eff.org`. The details of the algorithm, and our collection methodology, are discussed in Section 3. While our sample of browsers is quite biased, it is likely to be representative of the population of Internet users who pay enough attention to privacy to be aware of the minimal steps, such as limiting cookies or perhaps using proxy servers for sensitive browsing, that are generally agreed to be necessary to avoid having most of one's browsing activities tracked and collated by various parties.

In this sample of privacy-conscious users, 83.6% of the browsers seen had an instantaneously unique fingerprint, and a further 5.3% had an anonymity set of size 2. Among visiting browsers that had either Adobe Flash or a Java Virtual Machine enabled, 94.2% exhibited instantaneously unique fingerprints and a further 4.8% had fingerprints that were seen exactly twice. Only 1.0% of browsers with Flash or Java had anonymity sets larger than two. Overall, we were able to place a lower bound on the fingerprint distribution entropy of 18.1 bits, meaning that if we pick a browser at random, at best only one in 286,777 other browsers will share its fingerprint. Our results are presented in further detail in Section 4.

In our data, fingerprints changed quite rapidly. Among the subset of 8,833 users who accepted cookies and visited `panopticlick.eff.org` several times over a period of more than 24 hours, 37.4% exhibited at least one fingerprint change. This large percentage may in part be attributable to the interactive nature of the site, which immediately reported the uniqueness or otherwise of fingerprints and thereby encouraged users to find ways to alter them, particularly to try to make them less unique. Even if 37.4% is an overestimate, this level of fingerprint instability was at least momentary grounds for privacy optimism.

Unfortunately, we found that a simple algorithm was able to guess and follow many of these fingerprint changes. If asked about all newly appearing fingerprints in the dataset, the algorithm was able to correctly pick a "progenitor" fingerprint in 99.1% of cases, with a false positive rate of only 0.87%. The analysis of changing fingerprints is presented in Section 5.

2 Fingerprints as Threats to Web Privacy

The most common way to track web browsers (by "track" we mean associate the browser's activities at different times and with different websites) is via HTTP cookies, often set by with 3rd party analytics and advertising domains [9].