Finger Printing on the web

Preethi Josephina Mudialba, Siddharth Nair, Jun Ma
Carnegie Mellon University
Pittsburgh, USA
Email: {pjosephi, siddharn, junm2} @andrew.cmu.edu

***Abstract*—Nowadays, companies such as Google [1] and Facebook [2] are present on most websites either through advertising services or through social plugins. They follow users around the web collecting information to create unique user profiles. This serious threat to privacy is a major cause for concern, as this reveals that there are no restrictions on the type of information that can be collected, the duration it can be stored, the people who can access it and the purpose of use or misuse. In order to determine if there indeed is any statistically significant evidence that fingerprinting occurs on certain web services, we integrated 2 software namely, FPBlock [6], a prototype browser extension that blocks web-based device fingerprinting, and AdFisher [10], an automated tool that sending out thousands of automated Web browser instances, in such a manner so that the data collected by FP-Block could be fed to AdFisher for analysis.**

Keywords—Fingerprinting; AdFisher; FPBlock; Privacy; Online-tracking

I. Introduction

It wasn't too long ago that sites tracked users by deploying cookies, but now with the overabundance of cookie blocking technology companies have had to significantly modify their tracking strategy in order to continue targeted advertising. One of these new tracking technologies is termed fingerprinting.

Fingerprinting, is a passive tracking technique and therefore more robust against manipulation by the user. Brower characteristics which may seem trivial such as screen dimensions, installed fonts, browser plugins, browser version number etc, can be combined to create a unique ID, which can be used to track a user, in other words a browser fingerprint is created and used to uniquely identify a user. In an experiment conducted by Eckersley in 2010, 94.2 % of the participants who used either Flash or JAVA could be uniquely fingerprinted and thus tracked without the use of browser or flash cookies [3]. Thus, more common the browser configuration, the harder it is to fingerprint a user. The Electronic Frontier Foundation (EFF), conducted a study to determine the browser characteristic which are most revealing of a user's identity and the extent to which a fingerprint can be unique [4].

Numerous companies deploy fingerprinting to understand user preferences and to carry out targeted advertising. Moreover, by deploying browser fingerprinting instead of cookies advertising companies can circumvent the legal limitations imposed on the use of cookies both in the United States and the European Union. Furthermore, fingerprinting also allows companies to track users across multiple devices, such as smartphones, tablets, laptops etc. Antifraud companies are portraying fingerprinting as a means of protection against malicious online attackers, where such companies store the device fingerprint along with the users browsing history and reputation score [3]. A study conducted by researchers at Princeton and KU Leuven in Belgium in May 2014 exposed the list of the top 5,000 popular websites that use fingerprinting scripts on their sites. 95% of these sites, including the WhiteHouse.gov gained fingerprinting access from a JavaScript widget created by AddThis.com [5]. As such newer user tracking techniques are shifting their focus from authentication techniques which are based on what the user has to authentication techniques based on what an user is, as its harder to hide what you are than what you have [6]. Using existing anti-fingerprint tools, it is possible to determine the attributes making up the fingerprint vector [6]. However, in this project we intend to investigate the depth to which fingerprinting occurs, analyze the type of ads served to the users, determine the effectiveness of an anti-fingerprint tool when it runs on a browser and identify if there exists any relation between the ads displayed to a user while the anti-fingerprint tool is running.

II. Related Work

Currently numerous tracking mechanisms exists, which are very hard to detect and hence very difficult to control, block or avoid. One such mechanism is canvas fingerprinting, an advanced type of fingerprinting which exploits the difference in the images rendered by different computers for the same. The difference may be the result of differences in font rasterization such as antialiasing, hinting or sub-pixel smoothing, differences in system fonts, API implementations or even the physical display and the diversity of outcomes can be maximized by drawing as many different letters as possible on the canvas. The different outputs are equivalent to different fingerprints and are usually achieved within a fraction of a second without the user's knowledge. Besides, even simple sentences in a widely distributed system font can produce significant variation [5]. Mowery and Shacham's paper on Canvas printing, discusses the possible methods by which the occurrence of finger printing can be prevented, such as addition of noise to the pixel data or production of the same pixel for every single system. However, none of the methods described are faultless. The best solution would be to ask the user for permission for every canvas read attempt and this is exactly what the Tor browser implements to successfully protect its users from canvas fingerprinting. For every canvas function that can be used to read image data, the Tor browser returns an empty image and a pop-up dialog box requesting permission to access the canvas [7]. With respect to fingerprinting in general, there are numerous protective measures which can be taken to prevent its occurrence. For instance, the Ghostery browser extension allows users to view the list of trackers present on each web site and to take action to block the trackers in an easy and efficient manner. Furthermore, Ghostery allows users to block or unblock trackers selectively and offers the option to whitelist (allows all trackers for a particular website) or pause

978-1-5386-1104-3/17/$31.00 ©2017 IEEE

tracking whenever required [8]. Other such similar tools are PrivacyBAdger, AdBlock Plus etc

Duck Duck go is another option for users seeking to enhance their online privacy. Known as "The search engine that doesnt track you" [9] DuckDuckGo neither collects nor shares personal user information and thus in turn avoids does personalization of search results. Similarly, the Firefox based browser known as the Tor Browser, uses the Tor network a dedicated network used to route the browser traffic, to makes a strong case against most kind of fingerprinting [9].

III. Experimental Setup

A. Ad Fisher

AdFisher is an automated tool that works by sending out hundreds or thousands of automated Web browsers on carefully chosen trails across the Web thereby allowing ad-targeting networks to infer certain interests or activities. The software then records the ads that are displayed when an automated browser is sent to a news website that uses Googles ad network. Any changes to the ad settings page are also recorded [10].

B. FP Block

FPBlock is a prototype browser extension that detects and prevents the occurrence of web-based device fingerprinting. FPBlock detects the presence of fingerprinting scripts by dynamically analyzing the JavaScript codes embedded in websites; it then prevents the scripts from leaking an users fingerprint to third-party servers [11]. FP Block also blocks all third-party cookies [6]. So, a website attempting to fingerprint using third-party cookies is also blocked. Currently FP-Block just runs on version 45 of Mozilla Firefox.

C. Integration: Ad Fisher + FP Block

We first compiled the source-code of FP Block into an extension file that could be installed on Mozilla Firefox, analyzed the code of Ad Fisher and modified the existing Ad Fisher script to allow us to control the instances in which FP Block would launch as a part of the browser with the help of Selenium.

We then modified the script test.substance.py [13], a simplified version of the original experiment found in the example folder on AdFisher's Github page, so that in the Control and Experimental groups, two units of the browser instance without FP Block and two units with FP Block would be launched. We collected ads from both the groups and analyzed them. The results indicated a **significant p-value (0.0068)**, showing that the ads served were indeed different for the same websites.

Fig. 1. Test Results 1

Websites Collected Ads From	Site Category Visited	Test Accuracy	p-value
BBC	Cars	1.0	0.0009
	Substance Abuse	1.0	0.0006
	Drugs & Medication	1.0	0.0004
	Computers	0.95	0.0006
USAToday	Cars	0.4375	0.9163
	Substance Abuse	0.875	0.0066
	Drugs & Medication	0.9375	0.0017
	Computers	0.875	0.0074

Fig. 2. Ad Fisher – test.substance.py

```
lk@lk-VirtualBox: ~/info-flow-experiments/AdFisher/examples
Instance 2 exiting!
Experiment Complete
Reading log.....................Reading complete
Treatments:  ['control (null)', 'experimental (substance abuse)']
Creating feature vectors-->>Complete
---Time for getting feature vectors:  0:00:00.583641
Number of blocks in log:  20
Number of agents in a block:  4
Size of feature vector:  100
Total count of features:  1395
[treatments] [blocks] [features] [unique-features] ::  [0 1] [40 40] [669, 726] [ 54.  70.]

Split at block  16
Training Set size:  16 blocks
Testing Set size:  4 blocks
Max score:  0.975
Selected Classifier:
LogisticRegression(C=0.5, class_weight=None, dual=False, fit_intercept=True,
          intercept_scaling=1, max_iter=100, multi_class='ovr', n_jobs=1,
          penalty='l2', random_state=None, solver='liblinear', tol=0.0001,
          verbose=0, warm_start=False)
---Time for selecting classifier:  0:00:00.617908
CVscore:  0.975
Test accuracy:  0.875
Running permutation test

Tobs:  14
---Time for running permutation test:  0:00:00.641624

Confidence Interval of p-value: (0.0048682235095779394, 0.009215681379837112)
p-value:  0.0068
lk@lk-VirtualBox:~/info-flow-experiments/AdFisher/examples$
```

IV. Discussion

We chose 4 categories namely cars, substance abuse, drugs & medication and computers as they were listed as the most common search queries and for each category we chose the 5 most popular websites from Alexa [12]. The experimental design was set up, so that 2 instances were running with FP Block and 2 instances without FP Block. Due to limited resources, only 20 blocks were used to carry out the testing as the tests very computationally intensive and took a while to complete. For all categories for which Ads were collected from BBC, we obtained significant p-values with high test accuracy, indicating that there indeed is a difference in the Ads displayed in the browser instances running with FP Block and without it. We obtained similar results when we collected Ads from USAToday for all cases except for that of Cars. To ensure accuracy of results we tested Cars multiple times, however we still obtained the same result. This may be an anomalous case, as the results obtained for cars on BBC did not indicate such a high p-value. However, there is also a possibility that, USAToday by default displays car Ads which are unrelated to a user's search preferences.

From the results obtained we could conclude that in general fingerprinting indeed does occur however, it is su-cessfully blocked while running FP Block.

V. Conclusion & Future Work

It's no secret that almost all major web companies finger-print their users, either to improve user experience or to gain competitive advantage by analyzing user behavior. This type of Cross-domain tracking is generally considered a threat to user privacy [10] The aim of the project was to combine 2 software namely FPBlock define and AdFisher, to determine if there is statistically significant evidence that fingerprinting occurs on major web services. By combining 2 software namely FPBlock and AdFisher we were able to carry out a series of tests which helped confirm that fingerprinting indeed does occurs on web services. In the future, we intend to use Ghostery, to block individual trackers and to carry out the tests in between websites which have common trackers.

Acknowledgment

We would like to take this opportunity to express our profound gratitude and deep regards to Prof Datta and Loh Lay Kuan who advised, helped and patiently guided us throughout the duration of this project.

References

[1] Simonite, T. (2017). It's Google's Web, and we're just surfing it. Their tracking code is on almost a million websites. [online] MIT Technology Review. Available at: https://www.technologyreview.com/s/601488/largest-study-of-online-tracking-proves-google-really-is-watching-us-all/ [Accessed 15 Aug. 2017].

[2] Facebook.com. About Facebook Adverts. [online] Available at: https://www.facebook.com/ads/about/ [Accessed 15 Aug. 2017].

[3] Acar, G., Juarez, M., Nikiforakis, N., Diaz, C., Gürses, S., Piessens, F. and Preneel, B. (2013). FPDetective: Dusting the Web for Fingerprinters. [online] Available at: https://www.esat.kuleuven.be/cosic/publications/article-2334.pdf [Accessed 15 Aug. 2017].

[4] Langdon, B. (2013). Browser Fingerprinting. Available at: https://brett.is/writing/about/browser-fingerprinting/ [Accessed 15 Sept. 2017].

[5] Acar, G., Eubank, C., Englehardt, S., Juarez, M., Narayanan, A. and Diaz, C. (2014). The Web Never Forgets. [online] Available at: https://securehomes.esat.kuleuven.be/~gacar/persistent/the_web_never_forgets.pdf [Accessed 15 Aug. 2017].

[6] Torres, C., Jonker, H. and Mauw, S. (2015). FP-Block: Usable Web Privacy by Controlling Browser Fingerprinting. Computer Security -- ESORICS 2015, pp.3-19.

[7] Mowery, K. and Shacham, H. (2012). Pixel Perfect: Fingerprinting Canvas in HTML5.

[8] Ghostery. Ghostery Makes the Web Cleaner, Faster and Safer!. [online] Available at: https://www.ghostery.com [Accessed 15 Aug. 2017].

[9] Amiunique.org. (n.d.). Am I unique?. [online] Available at: https://amiunique.org/tools [Accessed 15 Aug. 2017].

[10] Simonite, T. (2017). Study Suggests Google's Ad-Targeting System May Discriminate. [online] MIT Technology Review. Available at: https://www.technologyreview.com/s/539021/ probing-the-dark-side-of-googles-ad-targeting-system/ [Accessed 15 Aug. 2017].

[11] Kim, D. (2014). Poster: Detection and Prevention of Web-based Device Fingerprinting. IEEE Symposium on Security and Privacy. [online] Available at: http://www.cs.utexas.edu/dkim/papers/webfingerprint-poster_sp14.pdf.

[12] Alexa.com. (n.d.). Alexa Top 500 Global Sites. [online] Available at: http://www.alexa.com/topsites [Accessed 15 Aug. 2017].

[13] GitHub. (n.d.). tadatitam/info-flow-experiments. [online] Available at: https://github.com/tadatitam/info-flow-experiments/blob/master/AdFisher/examples/test.substance.py [Accessed 15 Aug. 2017].