Of Mice and Men

Mouse movements tracking and browser privacy UI protections

Lukasz Olejnik
INRIA
Rhone-alpes
lukasz.olejnik@inria.fr

Claude Castelluccia
INRIA
Rhone-Alpes
claude.castelluccia@inria.fr

ABSTRACT

We perform a Web privacy measurement of presence of mouse movement tracking scripts in popular Web sites. Such data are potentially of great sensitiveness. We point out the scientific evidence, emphasizing the privacy risks of mouse movement leaks. We highlight the future risks of using this technology in ways detrimental to the users: profiling, tracking, private data inference.

We introduce a transparency enhancing technology (TET), improving browser's user interface and privacy usability. The tool shows to the user when a mouse movement and key press recording takes place on the visited Web site.

The lack of this functionality in the current privacy user interfaces models of Web browsers is a significant drawback in the presence of mouse movement analytics services. Our tool files this gaping hole.

1. INTRODUCTION

Recently, mouse movement tracking companies are gaining popularity. The purpose of these tools is to allow Web developers to discover the Web browsing patterns of their users, by tracking their mouse movements. We perform a measurement aimed at detecting the use of mouse tracking, as we detected they are used in a non-transparent manner, i.e. Web users are not informed about these practices.

Transparency is a crucial factor in maintaining privacy. When certain information related to the use of a service is concealed from the user, it is not clear if his will or intent are honored.

When the new APIs, known under the collective name of *HTML5*, were implemented, it was clear that protections must be in place in some specific cases. A good example is a GPS service, which allows the transmission of the user's location data to the visited site. Browsers can provide this information, but only with an explicit user permission. Google's Chrome can also inform the user whenever sensitive resources such as geolocation, web cam or microphones are in use, significantly contributing to the building of user's awareness.

The introduction of permissions model in browsers was certainly a feature worth praising. However, this has left some legacy sensors unprotected, constituting to serious privacy and transparency problems in the security models of current browsers. Namely, Web sites can track mouse movements and key presses [19]. This in itself already renders the permissions model *inconsistent*. Even more so, no indications informing the user about key presses or mouse movements being tracked during the visiting of a site, exist. This clear *privacy user interface* (UI) imperfection might soon bring consequences to Web users.

Recent body of works has shown unanimously that mouse movement tracking can be detrimental to user's privacy. It is known that mouse movements are directly related to eyes and the direction the user looks (parts of Web site he is interested in) [6]. This fact reveals the information about the content the user is specifically interested in. For example, an e-commerce store may know which products the user is looking at. By analyzing user's mouse movement dynamics, Web sites can also reason about the user's age [7], which in itself is a personal information. It is also clear that now companies are aware of it and mouse movement analyses are employing advanced techniques to take advantage of this data [1].

More importantly, it was shown recently that mouse-movements based biometric systems achieved astounding performance characteristics in terms of high users recognition results [20]. What this means is there *now exist* prototypical profiling approaches which utilize mouse movement data. The consequences for users would be that they might be subject to tracking across sites without any use of cookies or device fingerprinting [8, 1]. Instead, profiles based on mouse movement might be used.

There are already infrastructures for mass acquisition of mouse movement data. Mouse movement analytics are typically 3rd-party scripts recording and retrieving the users' mouse movements. An example company in this domain is Mouse Flow (*mouseflow.com*). We stress that the existence

[1]A good example is Mouse Eye Tracking: `http://met.picnet.com.au/NewUser.mvc/Faq\#q1`

of such 3rd-party entities collecting users' mouse movements across Web sites already creates a non-negligible privacy leak vector, in that user-related information is flowing from their computers to 3rd-party servers for later analysis. Recently, Facebook announced that it is considering to track the mouse movements of its users, as well [12].

In summary, the main contributions of this work are as follows:

1. *Mouse movement leaks.* We crawled the top $1M$ Alexa sites and detected over $1,200$ sites utilizing mouse movement analytics services that record and retrieve the users' mouse movements. We then perform an extensive analysis of mouse tracking scripts and distribution of Web sites utilizing those scripts (section 3).

2. *Inconsistency in browser privacy user interfaces.* We highlight that the permissions model employed by all major browsers is not consistent. For example, permissions are required to use *geolocation*, but not in case of *mouse movements* tracking (section 4.1.1).

3. *Mouse movement indicator.* We present MouseIndicator, a transparency enhancing browser extension which shows to the user when a Web site monitors mouse movements (section 4.2).

We highlight that in 2014, Web browsers still continue not to be transparent to their users. We believe that Web browsers' users must have total control over the technology they use. As of now, the users must accept a reality where any Web site can record mouse movements or key presses without the knowledge, let alone consent of the end-users.

In addition to performing studies of mouse tracking scripts, our work touches crucial aspects of usability, namely, the privacy user interface. The main objective of this paper is to start a discussion in the community about the possible future threats of mouse movement tracking risks. We propose to take this risk seriously and respond *before it happens*, as opposed to the often seen *posteriori* responses. We are well aware of the notorious *CSS history leak* [4, 3], specifically that it required more than 10 years to convince browser vendors to adequately address the issue. We purposefully draw a direct parallel here.

2. RELATED WORK

Jang et al. discusses mouse and keyboard tracking on popular Web sites and introduce browser source code modifications allowing the monitoring of information flows [11]. They mention findings of (1st-party) Web sites which were hijacking mouse movements; we are not aware of any prior work providing a detailed discussion of privacy implications due to these exact mechanisms. The authors also list a number of Web sites including behavioral tracking scripts of Tynt. In our paper we entirely concentrate on 3rd-party scripts that are capable of retrieving mouse movements. Specifically, we discuss the new agents in the Web ecosystem, called *mouse movement analytics*. In addition to that, we are strictly concerned by the user interface and the user in general. We propose to introduce browser solutions enabling the informing the users about the very fact of mouse movement tracking taking place. Our proposed first solution can be used without the modification of browser's internals. Understandably, this is perhaps less robust, but much more simple to deploy, from the users and usability perspective.

Mouse movements monitoring is also mentioned in [18], however, risks, dangers nor defenses are not discussed, although dangers are acknowledged. In our work we put efforts on mouse movement tracking by 3rd-party mouse movement analytics services, exclusively; we position them in a clear privacy threat context. Our focus is entirely on the privacy user interface and we deliver and propose protection measures.

In the recent years there was a considerable progress in the mouse-movement based biometric schemes [9, 17, 14, 16, 2]. The most advanced work and results from the mouse dynamics domain is perhaps the one by Zheng et al. [20], where the authors obtain good metrics in terms of low false accept and reject ratios, by analyzing the curvature aspects of mouse movements. The recent works give credibility to this threat of privacy-invading mouse movement recording.

Zheng et al. collected their dataset of mouse movements on an online forum [20]. More than $1,000$ users participated in the experiment. The data were being collected over a one hour window and this proves the severity of mouse movement profiling is practical, as people usually spend a considerably more time on browsing the Web.

Zheng et al. did not discuss the *true implications* of their study. They positioned the potentially arising risks against keystrokes dynamics analysis, which require the logging of all the user pressed keys. They therefore did not address the privacy risks on the basis that storing key presses would be more invasive: *"we believe that our verification system will not cause any privacy violations"* [20]. Because of the biometric nature of the profile and given the good characteristics of that scheme (equal error rate of 1.3%), it was proved that systems based on mouse movement analysis can reliably be applied to discriminating between different users.

Being fully aware of the consequences to privacy, we propose to restrict mouse movement tracking under the model of browser permissions. Moreover, the very fact of mouse movement tracking should be clearly indicated to the user. We advocate for a solution introduced directly by browser vendors, to ensure wide-spread adoption by users.

3. MOUSE MOVEMENT LEAKS ANALYSIS

Mouse and clicks data provide information on the user's behaviors and actions during his Web browsing. Site owners might be interested in such analytics data in order to improve their Web sites' designs and possibly to learn the users' interests. Mouse movements usage on the Web sites can show the actual usability patterns. Mouse analytics companies exist for these purposes. As 3rd-parties, they provide scripts responsible for the collection and submission of mouse movements data to their servers. These scripts can be included to Web sites by their owners, who later can discover usability insight behind the users of their sites.

Furthermore, there are many Web sites, such as those devoted to games or otherwise interactive sites, that track mouse movements for functional purposes (e.g. to change the cursor icon, or as a mean of detecting if the user is active).

Our analysis focuses entirely on *mouse movement analytics*: scripts which continuously collect mouse movements and send them to 3rd-party servers. We are not interested in simple activity monitors or even collection of mouse movements by first-party servers (visited Web site).

3.1 Mouse tracking study

The use of `window.onmousemove` event handler can monitor all the mouse movements in the entire Web site's screen, with pixel resolutions. Identical data can be obtained by registering a listener to the `mousemove` event, using a standard JavaScript `addEventListener` function. Detailed timing information with milliseconds accuracy can also be included.

With mouse movement analytics gaining traction, soon users might face being tracked transparently on many sites and potentially across them. In general, the tracking of users' mouse movements by these kind of scripts is *never* related to the functional aspects (e.g. widgets or JavaScript libraries) of the sites.

3.1.1 Measurement methodology

In our measurements, we used PhantomJS to visit the $1M$ Alexa[2] most popular sites. We executed between 40 and 80 browser instances at a time, to visit Web sites in parallel. All of the sites were visited in February, 2014.

Web Sites can register event handlers like `window.onmousemove` and *document.onmousemove* to track mouse movements. It is possible to detect if such an event handler is registered, for example by simply just verifying if they are defined. We programmed the tool to verify whether mouse tracking takes place, checking for the event handlers definitions (mouse was tracked in this case on $14,861$ Web sites).

We logged all the HTTP requests executed during the visiting of a Web site. After the Web site has loaded (which was reported by PhantomJS), we waited for 5 seconds for remaining content to load. We then simulated the movement of the mouse using standard PhantomJS function calls. Subsequently, we analyzed HTTP requests after the mouse movements took place. This allowed us to observe and log mouse-movement related HTTP requests, if any occurred. We created lists of these hostnames and the numbers of times requests to these hostnames were performed. We extracted a list of suspicious 3rd-party servers. We then manually analyzed JavaScript code responsible for those requests.

In order to show strict results we had to ensure that not only mouse movements were being collected on the visited sites, but also establish, with high accuracy, that mouse movements are being sent to 3rd-party servers. For this reason, we only analyzed 3rd-parties for which we are sure that mouse movements collection happened. This proved to be challenging because mouse movements recording, by the very nature of user's use of mouse, can result in considerably large data chunks sent to servers. Due to this, it is understandable that this data might not be sent in the clear. Rather, often more effective solutions are in place which decrease the size of submitted data.

[2] `www.alexa.com`

mouseflow.com
collect.mouse3k.com
www.mousestats.com
mousetrace.com

Table 1: List of mouse analytics companies.

We therefore manually analyzed the (obfuscated) source code of these analytics scripts to confirm important facts. Specifically, in each studied case, we established that in the source code there was a functionality of *detection* (monitoring of mouse movements events), *storing* (typically in JavaScript arrays) and *submitting* of the data related to mouse movements and clicks (for example a cross-site request for an image was executed, and the mouse movements data were included as parameters).

In the end, we again searched through all the logs related to visiting the $1M$ sites. In particular, we searched for requests to the identified mouse movement trackers. This allowed us to compile a comprehensive list of Web sites including those scripts. We focused our analysis on those selected mouse analytics scripts.

3.2 Mouse analytics: results

A model example of mouse analytics is Mouse Flow (`mouseflow.com`). Mouse Flow registers a listener for the `mousemove` event upon the loading of the site: this event launches a JavaScript handler function whenever the user moves with his mouse. Each mouse movement is logged in a JavaScript array. The information contain the cursor's position (coordinates) on the screen (X, Y) and the time. After a sufficient number of data describing mouse movements is accumulated, a HTTP request is performed to `X.mouseflow.com/c.gif` (X depends on the server). This HTTP request may contain parameters, such as $d=$ (encoded chunks) or $p=$ (ID of a user for a given session). The nature of mouse analytics, to which we also include similar scripts of *mouse3k.com*, *mousestats.com* in addition to others, is a detailed monitoring of mouse movements. Table 1 enlists the hostnames of 3rd-party sites which collect mouse movements that we analyzed.

We stress that mouse movements are sent to 3rd-party servers and it is impossible to know what happens to this data later on. Because mouse movement characteristics are proved to be of biometric nature, it is not unwise to think that it would be very challenging in practice to anonymize this data, as it directly relates to a particular user. We experimentally verified that *mouseflow.com* accepts very robust data in terms of mouse movement coordinates and time of these events: each of the users' mouse movement is logged, a consequence of registering a listener for the `mousemove` event. The interval between the recorded mouse movements must be small

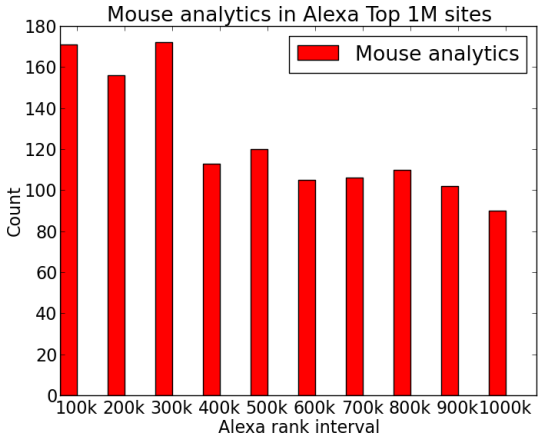

Figure 1: Numbers of mouse analytics scripts on Alexa sites.

enough to enable the creation of *heatmaps* of users' mouse movements: regions where users use their mouse the most. Using this information it would be potentially possible to deploy Zheng et al.'s biometric system [20] and profile Web users.

It is also interesting to note that there exist a *Google Analytics* extension which enable mouse movements tracking [10]. Due to the fact that Google Analytics is used widely on the Web, mouse tracking could easily become an issue, soon.

Next sections are devoted to analysis of mouse analytics proliferation.

3.2.1 Mouse analytics proliferation

In each visited Web site, we took account for scripts including mouse movement analytics JavaScript code. The scripts corresponded to a subset of hostnames in Table 1, notably: *mouseflow.com*, *mouse3k.com* and *mousetrace.com*.

Out of $1M$ Alexa sites, we detected that $1,208$ of them were utilizing these technologies. Most of the sites (92%) were using services of *Mouse Flow*. Although not in all of the cases the scripts appeared to actually actively record and submit users' mouse movements, all of the scripts were capable of doing so. Perhaps Web site developers turn them off on some occasions, depending on their needs.

Severity. Although the number of detected Web sites resorting to mouse tracking by 3rd-party scripts was not substantial, we emphasize that it is nonetheless a privacy leak: behavioral data is being collected. It is difficult to predict how popular mouse movement tracking will become in the future. However, authentication systems may make it popular and prevalent.

Distribution of sites using mouse analytics. Plot on Figure 1 is made for $100k$ intervals (X axis) of Alexa ranks. It can be seen that we found 169 sites with $Rank < 100k$, that included mouse analytics scripts. The numbers of included scripts appear to be slightly decreasing when studying subsequent intervals. For example, only 98 sites with rank corresponding to $800k < interval \leq 900k$, included such scripts.

Category	Mouse(%)	Top(%)	R
Sports	0.83	1.33	0.62
Games	1.16	3.77	0.31
Society/Lifestyle	1.16	1.25	0.93
Restaurants/Food	1.49	0.58	2.57
Vehicles	1.41	1.13	1.25
Job Search/Careers	1.16	1.24	0.94
Education	1.82	1.85	0.98
Search Engines/Portals	1.08	5.39	0.2
Financial Services	2.73	3.5	0.78
Entertainment	1.32	5.04	0.26
Blogs/Web Commu.	1.24	3.4	0.36
News/Media	1.74	9.41	0.18
Health	3.97	0.81	4.9
Travel	6.46	2.27	2.85
Shopping	15.23	7.15	2.13
Computers/Internet	14.49	14.74	0.98
Business/Economy	16.97	6.69	2.54

Table 2: Categories of sites using mouse movement analytics (*Mouse*), general Alexa 10k (*Top*) and the ratio (*R*) between these two percentages. Values are percentages.

3.2.2 Categories analysis

We studied the categories of sites utilizing mouse movement analytics. We categorized all these sites using Trend Micro Site Safety Center and McAfee Threat Center.[3] The categorizations provided by these sites were not consistent, so we used the former to first resolve the category of a site. In case of an unsuccessful categorization, we resorted to McAfee's tool. Afterwards, we unified categorizations. Specifically, we manually changed the McAfee categories to Trend Micro categories. So for example, "*Internet Services*" (McAfee terminology) became "*Computers / Internet*" (Trend Micro).

The results are shown in Table 2 (second column; only subset of categories surpassing 1 per cent point shown). We can see that over 16% of studied sites belong to category *Business / Economy*. From this table, we can conclude that most of the sites including mouse analytics are business, shopping and generally commercially related. In fact, the categories seem to be related with commercial activity on the Web. This is understandable as these companies must have Web sites optimized along the actual users' expectations and their use habits, so products or articles should be conveniently located on them. Using mouse movement services can potentially help, as Web site developers can study the actual patterns of use and the sites' layout can be improved with this insight.

3.2.3 Comparison to Alexa Top 10k

The third column of Table 2 shows the percentages of sites from the first 10k Alexa sites we categorized, for comparison purposes. The fourth column also shows the ratio (R) between the percentage of mouse-tracking Web sites and the

[3] `http://global.sitesafety.trendmicro.com/result.php`, `http://www.mcafee.com/us/threat-center.aspx`

general Alexa Top10k. A large R means that the category is an heavy user of mouse analytics. A low R means the inverse.

It can be seen that the shares of categories of most popular sites are different, with respect to the categories of sites including mouse movement analytics scripts. For example, over 15% of sites from the category *Business / Economy* included mouse movement analytics scripts. But the general share of sites in this category in Alexa 10k top sites is lower (6.69%), the ratio is therefore relatively large (2.54). Similar comment applies to others such as Shopping (ratio 2.13), Health (ratio 4.9) and Travel (ratio 2.85) sites. Sites of the category *Computers / Internet* have a similar share (with ratio 0.98 showing this similarity), but the most interesting difference is in the case of *News / Media* sites. Web sites of the category *News / Media* account for nearly 9.5% of the most popular sites in the Alexa 10k. However, we detected that only about 1.74% of the sites from this category included mouse analytics scripts and the ratio R was exceptionally low (0.18). This may again depend on the usability patterns. One explanation might be that the users have a tendency for a continuous use of their favorite news sites. Consequently, the users are well aware of the layouts and designs of these Web sites and it is easy for them to find the interesting information. On the other hand, business-related Web sites want to put special attention to their layouts. They want to obtain the best possible designs and to make their sites easy to use, so that users can find the interesting content before they leave to another site.

Apparent over-representation of sites from these specific non-infotainment categories (i.e. business sites) in terms of mouse movement analytics, could be a result of a focus on the continuous improvement of those layouts. In case the users find a layout of a visited Web site hard to use, they might not want to stay on this site. Infotainment sites usually provide a large amount of interesting content (from user's point of view), and thus they can make the users more likely to stay on their sites. A dissimilar situation arises in the case of specialized sites and then a focus on the ease of use becomes essential.

In an analogous manner, *Search Engines / Portals* sites are often easy to use and provide interesting content, the low ratio of 0.2 again shows a trend similar as in the case of *News / Media* sites.

Indeed, Web maintainers cannot expect users to stay on their sites for long (i.e. longer than 70 seconds), and this is especially the case of non-entertaining sites, which users tend to "screen" and the first seconds of their visit are critical in their decision as to whether they stay or leave this Web site [13].

3.2.4 User's consent

During our tests we detected Web sites including scripts performing mouse movement hijacks for analysis purposes. We later manually visited several (about 10%) of the analyzed Web sites. None of them informed its users about the practice of mouse movement tracking. We believe that this is also true in general: Web site developers do not feel an obligation to inform their users about the behavioral monitoring taking place.

Figure 2: Standard Chrome informs the user about utilization of sensitive data sources such as geolocation of webcam (red circle).

4. BROWSER UI AND MOUSE API

Browsers already employ the notion of *privacy user interface* (*UI*). Implementations allow the users to execute such basic actions as clearing of *cookies* or *Web browsing history*. Browser permissions are of special note.

4.1 Permission model

Sensitive data sources and new APIs such as geolocation, desktop notifications, camera or microphones require explicit permission of a user [5]. When a Web site intends to use a sensitive API such as geolocation, browsers typically ask the user for granting a permission to perform this task. This access can be granted on a session-basis (just once) or indefinitely.

4.1.1 Inconsistency of browsers models

Chrome is the only browser which informs the user about the use of geolocation service and the feature is currently under consideration in Firefox [15]. Chrome and Firefox also inform the user about the use of sources such as web cam. Figure 2 shows how this functions.

However, we found that the privacy UIs in all of the modern browsers are inconsistent: neither the permissions models do encompass such data sources as pointing or typing devices, nor the information is displayed to the user.

The discrepancy between treating of old-style sensors, such as mouse or keyboard, and the newer ones, such as geolocation and Web camera, obviously comes from the fact that the new additions have been thoroughly designed with privacy and control in mind, *by design*. However, on the same time no focus on the protection of old, but still powerful, browser functionalities have been made. As a result, a mixed model of permissions in the browsers exist, when some of the functionalities require the permission of the user, while others do not. Given the risk vectors we present in this work, it is therefore required to design and connect the user interface (UI) of the browsers with privacy in mind and create a clear and consistent privacy UI.

4.2 Improving transparency of mouse tracking

The Firefox transparency enhancing tool (TET) extension we introduce detects whether events like mouse movement or keypress tracking take place, by monitoring the values of event handlers.

After a Web site is loaded, the plugin checks if mouse or keypress events are monitored in the Document Object Model's `window` and `document` objects. Specifically, the extension utilizes the Firefox extension APIs to achieve this goal. While

Figure 3: The information that mouse movements and key presses are recorded is placed in the URL bar (red circle).

the approach is limited and certainly not ideal, it can serve as a proof of concept of the transparency aiding capabilities.

If the user-visited Web site records the user's mouse movements or keyboard presses, the TET extension clearly displays an information in the browser's URL bar. An example is shown on Figure 3 (red circle is highlighting the included custom icons). The proposed method is similar to Chrome browser's displaying the fact that a GPS location is being read.

4.2.1 Mouse tracker detection

Web sites routinely do not inform their users about the use of mouse analytics scripts. The TET plugin we introduce in this paper is also capable of detecting whether mouse movements are collected by mouse analytics scripts. If the plugin detects the inclusion of a mouse movement tracker, it includes an indicator (a red dot) to the mouse movement icon (Figure 3). The detection is currently done by the verification if a script from *mouseflow.com* is included on the visited Web site (via the `script` tag).

We are confident similar solutions could and should be implemented directly in the browsers, for the benefit of end users. Until then, transparency-aiding tools of the type we deliver, can be used by users. The Firefox extension is available at: [**ANONYMIZED**].

5. CONCLUSION

As a result of Web privacy measurement, we established that mouse movement monitoring scripts is in use on the Web and this data is sent to 3rd-party sites, which greatly complicates the control over this data. We analyzed the scripts of several companies and detected functionality for registration, processing and sending of users' mouse movements. As a result of our measurements, we discovered that about 0.1% of top Alexa sites include such scripts. Although the number is modest, it is challenging to judge how this number might change with future.

Our work was primarily motivated by the growing scientific evidence showing a relationships between mouse movements dynamics and user's age or the direction where he looks, as well as the ever-improving performance of mouse-movement biometric systems able to profile Web users.

As a final contribution, we discuss browser privacy user interfaces and introduce a proof of concept Firefox extension (a transparency enhancing tool) which demonstrates how it could be possible to integrate transparency solutions in the browser privacy UI.

6. REFERENCES

[1] G. Acar, M. Juarez, N. Nikiforakis, C. Diaz, S. Gürses, F. Piessens, and B. Preneel. Fpdetective: Dusting the web for fingerprinters. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*, CCS '13, pages 1129–1140, New York, NY, USA, 2013. ACM.

[2] A. A. E. Ahmed and I. Traore. A new biometric technology based on mouse dynamics. *IEEE Trans. Dependable Secur. Comput.*, 4(3):165–179, July 2007.

[3] Bugzilla. Bug 57351 - css on a:visited can load an image and/or reveal if visitor been to a site. `https://bugzilla.mozilla.org/show_bug.cgi?id=57531`, 2000.

[4] Bugzilla. Bug 147777 - :visited support allows queries into global history. `https://bugzilla.mozilla.org/show_bug.cgi?id=147777`, 2002.

[5] P. Byers, F. Hirsch, and D. Hazaël-Massieux. Permissions for device api access. `http://www.w3.org/TR/api-perms/`, 2010.

[6] M. C. Chen, J. R. Anderson, and M. H. Sohn. What can a mouse cursor tell us more?: correlation of eye/mouse movements on web browsing. In *CHI'01 extended abstracts on Human factors in computing systems*, pages 281–282. ACM, 2001.

[7] Y. Cheong, R. L. Shehab, and C. Ling. Effects of age and psychomotor ability on kinematics of mouse-mediated aiming movement. *Ergonomics*, 56(6):1006–1020, 2013.

[8] P. Eckersley. How unique is your web browser? In *Privacy Enhancing Technologies*, pages 1–18, 2010.

[9] R. A. J. Everitt and P. W. McOwan. Java-based internet biometric authentication system. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1166–1172, Sept. 2003.

[10] S. Genders. Sessioncam's session replay, heatmaps and web analytics. `https://www.google.com/analytics/apps/about?app_id=2256001`, 2013.

[11] D. Jang, R. Jhala, S. Lerner, and H. Shacham. An empirical study of privacy-violating information flows in JavaScript Web applications. In A. Keromytis and V. Shmatikov, editors, *Proceedings of CCS 2010*, pages 270–83. ACM Press, Oct. 2010.

[12] C. Johnston. Facebook may start logging your cursor movements. `http://arstechnica.com/business/2013/10/facebook-may-start-logging-your-cursor-movements/`, 2013.

[13] C. Liu, R. W. White, and S. Dumais. Understanding web browsing behaviors through weibull analysis of dwell time. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, pages 379–386, New York, NY, USA, 2010. ACM.

[14] R. Moskovitch, C. Feher, A. Messerman, N. Kirschnick, T. Mustafic, A. Camtepe, B. Löhlein, U. Heister, S. Möller, L. Rokach, and Y. Elovici. Identity theft, computers and behavioral biometrics. In *Proceedings of the 2009 IEEE international conference on Intelligence and security informatics*, ISI'09, pages 155–160, Piscataway, NJ, USA, 2009. IEEE Press.

[15] Mozilla. Last comment bug 630614 - create persistent indicator for geolocation sharing. `https://bugzilla.mozilla.org/show_bug.cgi?id=630614`,

2011.

[16] M. Pusara and C. E. Brodley. User re-authentication via mouse movements. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, VizSEC/DMSEC '04, pages 1–8, New York, NY, USA, 2004. ACM.

[17] C. Shen, Z. Cai, X. Guan, H. Sha, and J. Du. Feature analysis of mouse dynamics in identity authentication and monitoring. In *Proceedings of the 2009 IEEE international conference on Communications*, ICC'09, pages 673–677, Piscataway, NJ, USA, 2009. IEEE Press.

[18] M. Tran, X. Dong, Z. Liang, and X. Jiang. Tracking the trackers: Fast and scalable dynamic analysis of web content for privacy violations. In *Proceedings of the 10th International Conference on Applied Cryptography and Network Security*, ACNS'12, pages 418–435, Berlin, Heidelberg, 2012. Springer-Verlag.

[19] W3C. Document object model events: Mouse event types. `http://www.w3.org/TR/DOM-Level-2-Events/events.html#Events-eventgroupings-mouseevents`, 2000.

[20] N. Zheng, A. Paloski, and H. Wang. An efficient user verification system via mouse movements. In *Proceedings of the 18th ACM conference on Computer and communications security*, CCS '11, pages 139–150, New York, NY, USA, 2011. ACM.