Network Measurements for Web Tracking Analysis and Detection: A Tutorial

Ismael Castell-Uroz
Universitat Politècnica de Catalunya
Barcelona, Spain
icastell@ac.upc.edu

Josep Solé-Pareta
Universitat Politècnica de Catalunya
Barcelona, Spain
pareta@ac.upc.edu

Pere Barlet-Ros
Universitat Politècnica de Catalunya
Barcelona, Spain
pbarlet@ac.upc.edu

***Abstract*—Digital society has developed to a point where it is nearly impossible for a user to know what it is happening in the background when using the Internet. To understand it, it is necessary to perform network measurements not only at the network layer (e.g., IP, ICMP), but also at the application layer (e.g., HTTP). For example, opening a single website can trigger a cascade of requests to different servers and services to obtain the resources embedded inside it. This process is becoming so complex that, to explore only one website, the number of communications can explode easily from tens to hundreds depending on the website. Inside those communications, there is an ever-increasing portion dedicated to web tracking, a controversial practice from the security and privacy perspective.**

In this article, we present a tutorial on web tracking and how network measurements are needed to detect and analyze it. We also classify and review the scientific literature on this specific topic and discuss the open issues and challenges the measurement community has to address to detect web tracking more efficiently. Furthermore, we present Online Resource Mapper (ORM), a new large-scale web measurement framework specifically designed to address these open issues and facilitate the detection of web tracking. As an additional contribution, we made public a large-scale dataset collected with ORM that contains information about all of the resources being loaded by the most popular 100,000 websites on the Internet.

***Index Terms*—web tracking, content-filtering, measurements, targeted advertisement, dataset**

I. An overview of Web Tracking

Web tracking is a technology that comprises multiple methods used intentionally to follow and identify individuals when surfing the Internet. In the beginning, web tracking was designed to identify users within the web services given by a company on their own domains. The most famous tracking method is the use of the so-called "Cookies," small files saved in the computer by the Internet browser that contain an identifier of the current domain and browsing session. Cookies are sent automatically by the browser every time a website of the same domain is accessed. Web tracking is fundamental for online vendors, enabling them to present a shopping cart to the user, a place to put the items intended for purchase. This kind of behavior would be impossible without previously identifying the user.

Recently, web tracking methods crossed the intra-domain barrier with the appearance of Third-party trackers. Third-party trackers can track the user in domains not owned by or related to them. This is done by offering useful services to other companies in the form of embedded resources that can help to improve the number of users and, consequently, the relevance of their website. An embedded resource is an external resource like documents, images or scripts not owned by the company proprietary of the website. A typical and very popular example is the Facebook "Like" button. Many online news, marketplaces and content hosting websites include them (or similar elements from other social networks). However, the fact that Facebook (and the rest of the embedded services) can track the user in that website, despite not having an account or never clicking the "Like" button, is mostly unknown by the common user.

To track users through embedded resources, third-party tracking methods present an increasing complexity and level of detail in the collected data. Among the plethora of new web tracking methods, fingerprinting is the most complex, complete and intrusive of all of them. Fingerprinting tracks not only the user actions, but also the properties of the computers being used to access the web services and other relevant characteristics. Collecting information such as the OS version, browser version, installed fonts, screen dimensions or the city where the user is (through the network IP address) permits companies to combine all of them in a way that precisely identifies a user among all of the rest, even when using anonymous tools, such as private browsing mode [1].

II. Security and privacy implications of web tracking

Services such as the commented Facebook or other well-known actors like Google, Amazon or Twitter are the main examples of the commented third-party trackers. They usually collect personal data to build the most exhaustive profile possible of their users for their own profit. This is emphasized by the fact that they are at the same time first-party and third-party trackers. As a result, they not only collect data from multiple different places, but they also have confidential personal data obtained during the account creation process, such as real names, place of residence, telephone number or school graduation details. All of that data is combined into a profile that will allow them to personalize their websites and advertisements based on the users' likes and opinions.

Besides the usual third-party trackers, the presence of data brokers (i.e., companies focused on collecting data from

 DOI 10.1109/MIM.2020.9289071

millions of users to resell it to other companies) is mostly unknown to regular users but they are becoming more frequent on the Internet and present multiple security and privacy concerns. Usually, the power reached by those companies, due to the amount of data they collect, and how this power can alter everyday life, is not noticed. A practical example would be a political situation where a party has access to the personal information of a widely used social network, where political opinions and personal perspectives of millions of people are easily reachable. All of that information can be used to catalog people into different groups, depending on their ideological positions, and create specific online marketing political campaigns to influence users doubting their vote, and potentially favoring the election of a political party. This perverse use of the data already happened in the past [2] and had consequences not only for the people being tracked, but also for all the people living in the same country.

Besides, having all of this information gathered silently by an unknown company in a remote computer, where the user has neither knowledge nor access at all, represents a privacy problem by itself. Security breaches have been present in the Internet since its foundation, and even big companies that dedicate a lot of resources to secure their data have suffered from security flaws that allowed information theft. On the other hand, there is no guarantee that the collected information (and the conclusions drawn from it) are correct, as the user cannot access nor review it. However, this information is already being used for multiple purposes that affect our lives, such as financial assessment, determination of insurance coverage, price discrimination or background scanning.

Bujlow et al. presents in [1] a survey that includes a comprehensive description of the existing web tracking methods as well as references to the literature explaining the facts contained in this section. In this paper, we focus instead on the methods that have been proposed to analyze and detect web tracking using network measurements.

III. Measurements for the detection and analysis of web tracking

Web tracking is a transparent and pervasive data collection method that operates in the background, mainly at the application layer. As such, the only possibility to effectively detect and analyze it is by using network measurements to observe the information transmitted by the browser in the wild, either at the network or the application layer. Among the different network measurement techniques that can be applied, we can differentiate three main categories: passive, active and client-side measurements.

- Passive measurements: These kinds of measurements base their methodology on collecting all of the traffic seen inside a network (e.g., IP packets). The captured data is explored, looking for patterns and characteristics that could be useful for the experiment. The limited visibility of the traffic content at the application level and the privacy concerns introduced by capturing all the personal data being transmitted over the network are factors to consider.
- Active measurements: This methodology is based on performing measurements acting as the user. In web tracking experiments, this is done by executing a longitudinal study over a selected website population, opening the website as a user would do. This method allows a high number of very precise measurements to be obtained without the intervention of real users. However, this type of measurement presents some ethical considerations. By simulating a user, false visits are generated to the examined websites, a fact that if done in excess can affect their functionality.
- Client-side measurements: Client-side measurements are based on performing the measurements directly on the clients and sending back the results to a centralized place to explore them. In web tracking experiments, it is done mainly by using browser plugins installable by the user. This approximation avoids the privacy concerns introduced in passive measurements, as the user is previously informed of the purpose of the experiment and the overall process. It also gives precise measurement results that are only dependent on the client settings and specifications. The main drawback is a potential lack of support by the user, usually representing a big challenge to get enough participation to obtain a representative number of measurements.

Each presented methodology has its pros and cons and has been used in the past for web tracking measurements, as discussed in the next section.

IV. Network measurements for web tracking analysis and detection

Web tracking presents multiple security and privacy concerns, and as such, it is a hot topic in the network measurement community at the time of this writing. Therefore, there have been many relevant publications related to web tracking in different ways. In Table 1, we present a summary, divided by categories, of the most important related work in the field of network measurements. We mainly focus on research that used network measurements as a methodology to detect or analyze web tracking on the Internet.1

A. Detection/classification

T. Li et al. present a third-party tracker detection method using machine learning techniques to inspect cookies that achieves high accuracy [3]. The work also includes a measurement-based study of the third-party ecosystem within the top 10.000 most popular websites as per the Amazon Alexa's list [15]. They found third-party tracking to be very frequent, with almost 50% of websites having some third-party tracker embedded, and 25% including a tracker from Google.

In [4], Metwalley et al. use passive network measurements in a network with about 10,000 users to study the penetration and intrusiveness of web tracking systems in the Internet. They

Category	Reference and title	Type
Detection/Classification	[3] TrackAdvisor: Taking Back Browsing Privacy from Third-Party Trackers	Active
	[4] Using Passive Measurements to Demystify Online Trackers	Passive
	[5] Detecting and Defending Against Third-Party Tracking on the Web	Active
	[6] Tracking the Trackers	Client-side
	[7] Online Tracking: A 1-million-site Measurement and Analysis	Active
	[8] Towards accurate detection of obfuscated web tracking	Client-side
Regionality/Location	[9] Tracing Cross Border Web Tracking	Client-side
Targeted Advertising/ Price Discrimination	[10] Annoyed Users: Ads and Ad-Block Usage in the Wild	Passive
	[11] The AdWars: Retrospective Measurement and Analysis of Anti-Adblock Filter Lists	Active
	[12] Crowd-assisted Search for Price Discrimination in e-commerce: First Results	Client-side
Privacy/Security	[13] Measuring Privacy Loss and the Impact of Privacy Protection in Web Browsing	Active
	[14] The Chain of Implicit Trust: An Analysis of the Web Third-party Resources Loading	Active

TABLE I
MOST RELEVANT RELATED WORK

found embedded trackers in more than 70% of the websites, and some of them collected data from 98% of the users.

Roesner et al. present a new Firefox addon called ShareMeNot, the first tool to block third-party tracking social buttons by maintaining their functionality [5]. It also makes an extensive study of the state of third-party tracking in a population of 1,000 websites, finding Google as the top tracker with one third of occurrences.

Yu et al. present a new collaborative way of detecting trackers by computing the number of users that reach the same URL resources on a website [6]. The idea behind it is that if a specific URL is only used by one or a few users between all the population loading the same website, that resource probably includes a tracking identifier. They also observed a high percentage of tracking penetration, being more than 95% for the studied German population.

In [7], Englehardt et al. design a new web tracking measurement system called OpenWPM that uses a combination of a browser plugin, an HTTP proxy and the raw files created by the browser (e.g., cookies, temp files) to detect web tracking from active web measurements. The research also includes the study of the state of web tracking from one million websites in 2016.

Lastly, in [8], Le et al. study the presence of obfuscation within web tracking, and specifically in the canvas fingerprinting method. They found obfuscation to be present, although still not very widely used.

B. Regionality/Location

In [9], Iordanou et al. confront the problem of defining the geolocation borders within web tracking traffic. They measure the relation between the location of the user and the location of the company performing web tracking by means of a browser plugin to find if the current regulations (e.g., GDPR) allow to investigate how the collected data was used. They found that most tracking flows are well confined within the GDPR jurisdiction, but also that the most sensitive data (e.g., health, sexual orientation, political opinions) is being tracked at some extent inside as well as outside of the GDPR effective area.

C. Targeted advertisement/Price discrimination

Pujol et al. characterize the advertisement traffic using passive measurements and Adblock Plus, the most famous advertisement content-blocker plugin (also called adblocker) [10]. Starting from there, they explore the usage of adblockers in the wild, taking measurements from a national ISP. They found that about 22% of users browse the Internet with a content-blocker plugin enabled and extract some possible implications of this finding.

To be able to counteract those implications, companies started using some anti-adblocker systems, mechanisms that detect when a user is using an adblocker and correspondingly block the service for that specific user. In [11], Iqbal et al. explore the prevalence of such systems and the performance of anti-adblocker filter lists created by the community to avoid being blocked while using an adblocker.

Another way of improving sales profit is not directly with targeted advertisements but with price discrimination. The underlying idea is that, depending on the user profile collected by the tracking mechanisms (living location, wealth, relatives, friends and other related characteristics), the price that people are willing to pay for the same object is different. In [12], Mikians et al. perform a study on how to detect the price and search discrimination on the Internet using a collaborative approach by means of a browser plugin.

D. Privacy/Security

Krishnamurthy et al. was one of the first works that tried to measure the impact of web tracking in privacy and explore some measures to regain it [13]. The research also includes a study about the quality and usability modifications introduced by the tools used to improve privacy.

On the other hand, Ikram et al. focus their research on exploring the concept of implicit chain of trust within third-parties [14]. The idea is that when a website imports a resource from a third-party, the imported resource can also embed resources from other different third-parties that the original website does not necessarily know about. They found that although most of the websites' chain of trust is relatively small, a high percentage of them end up loading resources from a suspicious website or, in other words, a website that has been categorized as a possible thread to privacy or security.

E. Protection Measures

Currently, the only effective way of protecting the user against web tracking is by means of content-blockers, a type

of plugin installable in the browser that examines all of the URLs being accessed by the browser. Those URLs are then compared to a black list or a database containing the known web tracking websites, and if the comparison is positive, the resource is blocked. There are other methods like Javascript or Flash blockers to improve the security even more, but they usually break the website layout, degrading the usability or even making the website inaccessible which is a price too high to pay for the common user. Thus, content-blockers have become the de-facto solution for the privacy problem. Examples of popular content blockers include AdBlock Plus, Ghostery and uBlock Origin.

V. Challenges/Open issues

Using content-blockers as the main countermeasure is a defensive approach which limits the security to the already known websites. In an environment where new web tracking methods constantly emerge, and creating a new website or domain is done in a matter of minutes, it is not the ideal approach. Even so, there are some important challenges and open issues to be able to block web tracking in a more proactive way.

The first and foremost challenge we must deal with is the constant evolution of web tracking methods, using more and more complex and intrusive techniques specifically designed to overcome the privacy protection mechanisms available within the browser. Usually, there is very little, or no information at all, about those new web tracking mechanisms. To find them requires experts on the topic who are able to inspect the code and the new functions included in successive updates of the website programming frameworks, looking for parameters and characteristics that could be used to track the users. Once found, to check for empirical verification, an experiment should be executed. This process is very hard and time-consuming, which impacts on the ability of researchers to detect new methods.

Another obstacle is how to deal with minification and obfuscation. The first is a technique that tries to minimize the resource loading times eliminating white spaces, break lines and shortening the variable names of the resource code. The second one is similar but additionally changes the code structure and renames the variables by non-sense strings to make the code almost unreadable by humans. These two techniques render traditional methods based on code inspection unfeasible in a large population of websites, making it even harder to solve the first challenge of finding new web tracking methods.

To improve the state-of-the-art on these two aspects, first we need to take network measurements about the resources loaded and shared by first and third-party trackers, to be able to detect patterns and common characteristics between the web tracking code. In addition, we need to take those measurements on a large scale to be able to use other techniques, like machine-learning or data-mining algorithms, to automatically explore the information without requiring human intervention. This presents the third challenge, the lack of public datasets with detailed information and measurements about the resources loaded and parsed by websites at a large scale, that would permit researchers to detect new web tracking methods in a more automated way than it is currently done.

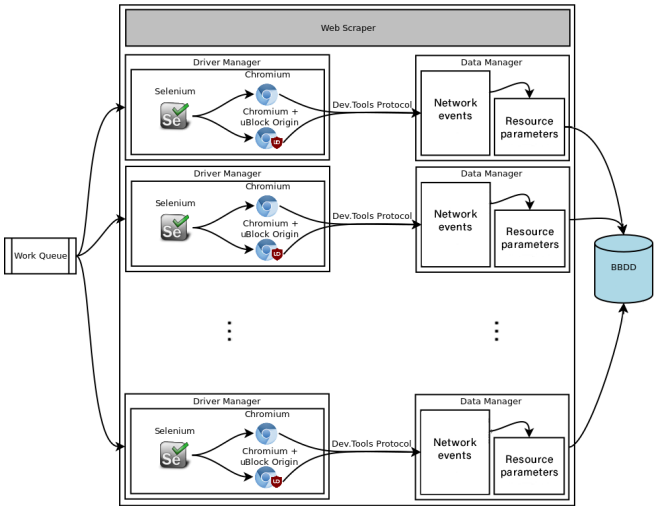

Fig. 1. ORM Structure

VI. ORM: A large scale web measurement framework

In this paper, we introduce Online Resource Mapper (ORM), an open source large-scale web measurement framework specifically designed to address the challenges described in the previous section. ORM is an active measurement framework and as such has been developed to be able to crawl information directly from websites about every resource accessed, either loaded by the main website or embedded within a third-party tracker resource. Moreover, the system automatically tries to unminify/unobfuscate the code of the resources found to ease their study, if necessary.

ORM uses a combination of Selenium, Chromium and the Chrome DevTools protocol to get detailed information about all of the resources loaded by the inspected websites. Selenium is a tool for automatized web experiments that acts as a wrapper of real browsers, like Mozilla Firefox, Google Chrome or, in our case, Chromium, the open source alternative version of Chrome. On the other hand, the Chrome DevTools protocol is a protocol developed by Google, and present in both Google Chrome and Chromium browsers, that allows to inspect all of the parameters, functions and information being processed internally by the browser.

This combination was selected to allow us to collect all of the resources and information included in the website, even if the code is obfuscated, whether the resource resides in the main site or in the third-party loaded resources. This way of taking the complete dataset of the entire explored population allows us to employ data-mining techniques to find common patterns and characteristics not usually visible. In comparison, most of the current active measurement research focuses only on the already known tracking methods, avoiding obfuscation

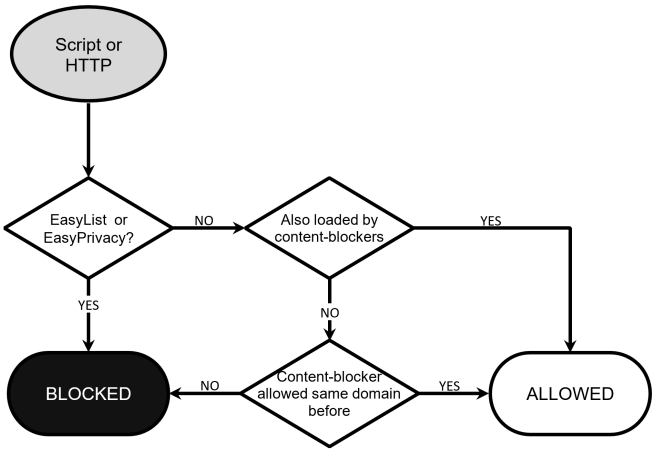

Fig. 2. Tracking decision diagram

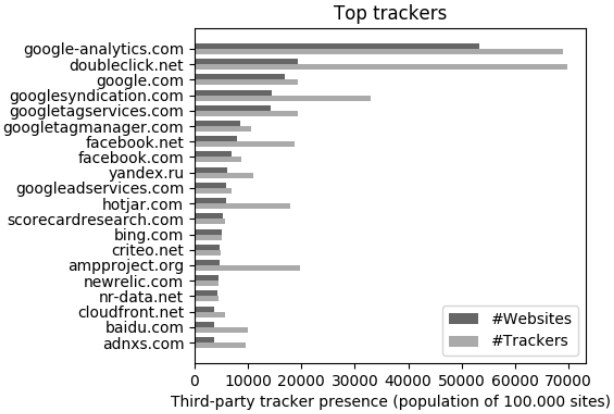

Fig. 3. Top trackers by number of websites an tracking resources

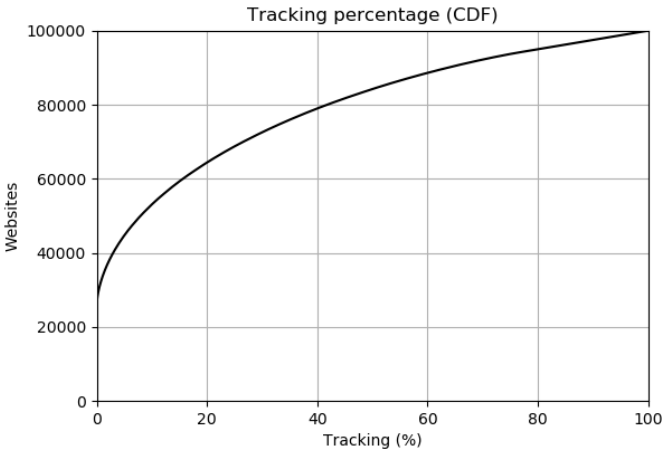

Fig. 4. Tracking percentage

and preventing them to be used to find new web tracking algorithms.

Fig. 1 shows a diagram of the overall system architecture and its different modules. Each Selenium instance maintains two browsers, one of them with a custom plugin loaded. The Driver Manager is the module in charge of loading one or more instances of Selenium with the needed plugins to perform the experiments, opening the websites and recovering in case Selenium becomes unresponsive. The module interacts with the Data Manager module, in charge of exploring the network events to look for resources to translate to real URLs that the Driver Manager can download. Lastly, all of the data is saved in a common database storing the collected information.

VII. Preliminary results

To show the capabilities of ORM, in this section we present the results of a proof-of-concept experiment that analyzes the current state of web tracking on the Internet. For this purpose, we used ORM to collect a large dataset with information of the top 100,000 most popular sites according to the Alexa's list [15]. ORM took seven days to collect the dataset. The infrastructure used was two *Ubuntu 16.04 LTS* servers in parallel with a combination of 30 cores, 60 threads and a total of 64 GB of RAM. The resulting dataset occupies more than 250 GB of data, and contains approximately 20 million resources, including the pages that loaded each of them. The information can be used for other multiple purposes, such as studying the geographical interaction between the different domains or third-party trackers or manually inspecting the code of the top most used resources. Moreover, we instructed ORM to download, unminify and store inside the database the script and document files, which are commonly used to execute web tracking methods. The system also compares each resource URL with *EasyList* and *EasyPrivacy*, the two most famous and used content-blocking lists, to check if it is an already known tracking/advertisement resource. The resulting dataset has been made publicly available at [16].

All of the information was taken for four different browser configurations, one with the vanilla browser (no plugins, no settings) and the other three loading each of them a different content-blocker plugin (AdBlock Plus, Ghostery and uBlock Origin). This configuration allowed us to generate a partially-labeled dataset with the already known web tracking methods by the existing lists and content blockers, which can be useful for training machine-learning algorithms. The labeling process is shown in Fig. 2. Intuitively, if a resource present in the vanilla browser is detected by EasyList/EasyPrivacy lists, or blocked by the content-blockers, that resource can be considered as a tracking system, either for profile creation or for targeted advertisement.

Fig. 3 shows the top web tracking domains found using this initial classification and the data collected by ORM. The results match with most of the previous studies, showing Google as the main tracker company, followed by Facebook, Yandex or Bing. Interestingly, comparing our results to some of the previous works (e.g., [3]), there is a clear increase in the domains being tracked by Google in any of its forms (e.g., analytics, ad services, tag services), reaching almost 55% of the entire population. It is worth noting that Doubleclick.com,

a subsidiary of Google specialized in targeted advertisements, is present in about 20,000 of the total websites, but that it has been blocked more than 70,000 times. This proves that loading multiples instances of the same third-party resource inside the same website is a common practice, probably to get more revenues from targeted advertising.

Fig. 4 presents the cumulative distribution function of the total tracking percentage traffic within the collected population (i.e., the fraction of traffic of the website devoted to tracking). There are approximately 28,000 websites (out of 100,000) that do not include any web tracking methods. On the contrary, almost 72,000 include at least one or more web tracking resources, with about a 10% including more than 90% of resources dedicated to track the user or targeted advertisement.

During the experiments we also observed that within the top 100 websites (including www.google.com), third-party tracking resources are usually not present or very rare. This is because most of them perform tracking as first-party trackers, getting data directly introduced by the user. For instance, Google does not need to look for the searches done by its users, as they introduce by themselves the search terms in Google websites, reaching their servers directly. Thus, the percentage of websites that lack tracking methods should be understood as an upper bound, depending on whether the included websites act also as a first party trackers or not, a fact that would make them remain undiscovered.

VIII. Conclusions and future work

In this article, we presented a tutorial on web tracking, its privacy and security implications and how network measurements can be used to detect and analyze it. It also includes a taxonomy of the most relevant publications on the topic categorized by type of measurement. We also present the three challenges we have to face to improve web tracking detection and analysis: the constantly evolving ecosystem, the obfuscation and dynamicity of the environment and the lack of public datasets. We introduced ORM, an open-source measurement framework specifically designed to address those challenges, and a first public dataset with information of 100,000 websites and about 20 million resources.

As a future work, we plan to develop an application programming interface to allow automated access to the dataset information to other research groups that may have interest in the included data. Moreover, we are currently exploring the data by means of data-mining algorithms to find characteristics that permit us to detect new web tracking methods in an automated and unsupervised way.

IX. Acknowledgments

This work was supported by the Spanish MINECO under contract TEC2017-90034-C2-1-R (ALLIANCE).

References

[1] T. Bujlow, V. Carela-Español, J. Solé-Pareta, and P. Barlet-Ros, "A Survey on Web Tracking: Mechanisms, Implications, and Defenses," *Proceedings of the IEEE*, vol. 105, pp. 1476–1510, Aug. 2017.

[2] The Guardian, "The Cambridge Analytica Files," *https://www.theguardian.com/news/series/cambridge-analytica-files*, 2015.

[3] T.-C. Li, H. Hang, M. Faloutsos, and P. Efstathopoulos, "TrackAdvisor: Taking Back Browsing Privacy from Third-Party Trackers," in *Passive and Active Measurement* (J. Mirkovic and Y. Liu, eds.), Lecture Notes in Computer Science, (Cham), pp. 277–289, Springer International Publishing, 2015.

[4] H. Metwalley, S. Traverso, and M. Mellia, "Using Passive Measurements to Demystify Online Trackers," *Computer*, vol. 49, pp. 50–55, Mar. 2016.

[5] F. Roesner, T. Kohno, and D. Wetherall, "Detecting and Defending Against Third-Party Tracking on the Web," in *Proceedings of Detecting and Defending Against Third-Party Tracking on the Web*, pp. 155–168, 2012.

[6] Z. Yu, S. Macbeth, K. Modi, and J. M. Pujol, "Tracking the Trackers," in *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, (Montréal, Québec, Canada), pp. 121–132, International World Wide Web Conferences Steering Committee, Apr. 2016.

[7] S. Englehardt and A. Narayanan, "Online Tracking: A 1-million-site Measurement and Analysis," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, (Vienna, Austria), pp. 1388–1401, Association for Computing Machinery, Oct. 2016.

[8] H. Le, F. Fallace, and P. Barlet-Ros, "Towards accurate detection of obfuscated web tracking," in *2017 IEEE International Workshop on Measurement and Networking (M N)*, pp. 1–6, Sept. 2017.

[9] C. Iordanou, G. Smaragdakis, I. Poese, and N. Laoutaris, "Tracing Cross Border Web Tracking," in *Proceedings of the Internet Measurement Conference 2018*, IMC '18, (Boston, MA, USA), pp. 329–342, Association for Computing Machinery, Oct. 2018.

[10] E. Pujol, O. Hohlfeld, and A. Feldmann, "Annoyed Users: Ads and Ad-Block Usage in the Wild," in *Proceedings of the 2015 Internet Measurement Conference*, IMC '15, (Tokyo, Japan), pp. 93–106, Association for Computing Machinery, Oct. 2015.

[11] U. Iqbal, Z. Shafiq, and Z. Qian, "The ad wars: retrospective measurement and analysis of anti-adblock filter lists," in *Proceedings of the 2017 Internet Measurement Conference*, IMC '17, (London, United Kingdom), pp. 171–183, Association for Computing Machinery, Nov. 2017.

[12] J. Mikians, L. Gyarmati, V. Erramilli, and N. Laoutaris, "Crowd-assisted search for price discrimination in e-commerce: first results," in *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies*, CoNEXT '13, (Santa Barbara, California, USA), pp. 1–6, Association for Computing Machinery, Dec. 2013.

[13] B. Krishnamurthy, D. Malandrino, and C. E. Wills, "Measuring privacy loss and the impact of privacy protection in web browsing," in *Proceedings of the 3rd symposium on Usable privacy and security*, SOUPS '07, (Pittsburgh, Pennsylvania, USA), pp. 52–63, Association for Computing Machinery, July 2007.

[14] M. Ikram, R. Masood, G. Tyson, M. A. Kaafar, N. Loizon, and R. Ensafi, "The Chain of Implicit Trust: An Analysis of the Web Third-party Resources Loading," in *The World Wide Web Conference*, WWW '19, (San Francisco, CA, USA), pp. 2851–2857, Association for Computing Machinery, May 2019.

[15] K. Cooper, "Alexa: Most popular website list," *http://www.alexa.com*.

[16] Ismael Castell-Uroz, "Online Resource Mapper (ORM)," *https://github.com/ismael-castell/ORM*, 2020.

X. About the authors

Ismael Castell-Uroz is a Ph.D. student at the Computer Architecture Department of the Universitat Politècnica de Catalunya (UPC), where he received the B.Sc. degree in Computer Science in 2008 and the M.Sc degree in Computer Architecture, Networks and Systems in 2010. He has several years of experience in network and system administration and currently holds a Projects Scholarship at UPC. His expertise and research interest are in computer networks, especially in the field of network monitoring, web tracking and anomaly detection.

Josep Solé-Pareta received the M.Sc. degree in telecom engineering and the Ph.D. degree in computer science from the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain, in 1984 and 1991, respectively. In 1984, he joined the Computer Architecture Department, UPC, where currently, he is Full Professor. He did Postdoctoral studies (summer 1993 and 1994) at Georgia Institute of Technology, Atlanta, GA, USA. He is cofounder of the UPC-CCABA (http://www.ccaba.upc.edu). His publications include several book chapters and more than 100 papers in relevant research journals (¿25) and refereed international conferences. His current research interests are in nanonetworking communications, traffic monitoring and analysis and high-speed and optical networking, with emphasis on traffic engineering, traffic characterization, MAC protocols, and QoS provisioning. He has participated in many European projects dealing with computer networking topics.

Pere Barlet-Ros received the M.Sc. and Ph.D. degrees in Computer Science from the Universitat Politècnica de Catalunya (UPC) in 2003 and 2008, respectively. He is currently an associate professor with the Computer Architecture Department of the UPC and scientific director at the Barcelona Neural Networking Center (BNN-UPC). From 2013 to 2018, he was co-founder and chairman of the machine learning startup Talaia Networks. His research has focused on the development of novel machine learning technologies for network management and optimization, traffic classification and network security, which have been integrated in several open-source and commercial products, including Talaia, Auvik TrafficInsights, Intel CoMo and SMARTxAC. In 2014, he received the 2nd VALORTEC prize for the best business plan awarded by the Catalan Government (ACCIO) and in 2015 the Fiber Entrepreneurs award as the best entrepreneur of the Barcelona School of Informatics (FIB).