Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Data Article

TrackSign-labeled web tracking dataset

Ismael Castell-Uroz [*], Pere Barlet-Ros

Universitat Politècnica de Catalunya (UPC), Spain

ARTICLE INFO

Keywords:
Internet
Security
Privacy
Web tracking
Fingerprinting
TrackSign
URL

ABSTRACT

Recent studies [8] show that more than 95% of the websites available on the Internet contain at least one of the so-called web tracking systems. These systems are specialized in identifying their users by means of a plethora of different methods. Some of them (e.g., cookies) are very well known by most Internet users. However, the percentage of websites including more "obscure" and privacy-threatening systems, such as fingerprinting methods identifying a user's computer, is constantly increasing. Detecting those methods on today's Internet is very difficult, as almost any website modifies its content dynamically and minimizes its code in order to speed up loading times. This minimization and dynamicity render the website code unreadable by humans. Thus, the research community is constantly looking for new ways to discover unknown web tracking systems running under the hood.

In this paper, we present a new dataset containing tracking information for more than 76 million URLs and 45 million online resources, extracted from 1.5 million popular websites. The tracking labeling process was done using a state-of-the-art discovery web tracking algorithm called TrackSign [8]. The dataset also contains information about online security and the relation between the domains, the loaded URLs, and the online resource behind each URL. This information can be useful for different kinds of experiments, such as locating privacy-threatening resources, identifying security threats, or determining characteristics of the URL network graph.

1. Specifications table

Subject	Computer Networks and Communications
Specific subject area	Internet subjacent website information
Type of data	Relational database model (MySQL Workbench) and CSVs containing information for each of the database tables.
How data were acquired	The data were collected using ORM [1], a self-developed public open source tool to inspect many aspects of the Internet.
Data format	Raw
Parameters for data collection	The website selection is based in the combination of the Alexa and Majestic most popular websites list (1.5 million of websites). A timeout of 60 seconds was used to discard very slow or non-responsive websites.
Description of data collection	To collect the data, we used our own custom open source tool called ORM. ORM opens a website and inspects the browser's internal network communications to find each URL loaded in the background. Then, it identifies the URLs and online resources accessed, and stores many important information of each of them as well as the relation between them and the domain that load them. Finally, the TrackSign [8] web tracking detection algorithm is executed over the collected data to discover new web tracking not detected by current methods such as the common adblockers.
Data source location	Institution: Universitat Politècnica de Catalunya (UPC) City/Town/Region: Barcelona Country: Spain
Data accessibility	Direct URL to data: https://www.cba.upc.edu/downloads/category/29-web-tracking-datasets?download=1085:tracksign-labelled-web-tracking-dataset
Related research article	**Author's name:** Ismael Castell-Uroz, Josep Solé-Pareta, Pere Barlet-Ros **Title:** TrackSign: Guided Web Tracking Discovery **Conference:** IEEE INFOCOM 2021-IEEE Conference on Computer Communications, 1-10 **DOI:** 10.1109/INFOCOM42981.2021.9488842 **Author's name:** Ismael Castell-Uroz, Rubén Sanz-García, Josep Solé-Pareta, Pere Barlet-Ros **Title:** Demystifying Content-blockers: Measuring their Impact on Performance and Quality of Experience **Journal:** IEEE Transactions on Network and Service Management (TNSM), **DOI:** 10.1109/TNSM.2022.3179267 **Author's name:** Ismael Castell-Uroz, Theo Poissonnier,

(*continued on next page*)

[*] Corresponding author.
E-mail address: Ismael.castell@upc.edu (I. Castell-Uroz).

https://doi.org/10.1016/j.comnet.2023.109687
Received 19 June 2022; Received in revised form 28 December 2022; Accepted 5 March 2023
Available online 12 March 2023

(*continued*)

	Pierre Manneback, Pere Barlet-Ros **Title:** URL-based Web Tracking Detection Using Deep Learning **Conference:** 2020 16th International Conference on Network and Service Management (CNSM), 1-5 **DOI:** 10.23919/CNSM50824.2020.9269065 **Author's name:** Ismael Castell-Uroz, Josep Solé-Pareta, Pere Barlet-Ros **Title:** Network measurements for web tracking analysis and detection: A tutorial **Journal:** IEEE Instrumentation & Measurement Magazine 23 (9), 50-57 **DOI:** 10.1109/MIM.2020.9289071
Related project(s)	Spanish I+D+i project TRAINER-A (ref.~PID2020-118011GB-C21), funded by MCIN/ AEI/10.13039/ 501100011033

2. Value of the Data

Why are these data useful?

Today's Internet is a complex mesh of interconnected hosts. Websites where all the content is self-contained are practically extinct. Every website includes some third-party resources, and most of them modify their content dynamically to meet their users' expectations. This makes detecting web tracking as well as conducting other Internet experiments very difficult, as one cannot explore the website code looking for static links in the traditional way anymore. You must ensure that all the content is explored, even if modified in real-time. On top of that, many JavaScript resources are obfuscated or minified, techniques used to minimize the loading times that render the code unreadable by humans. In this dataset, we present the complete information of all the HTTP requests, HTTP responses, certificates, URLs, resources, and geolocation information obtained for each of the more than 1.5 million popular websites. Moreover, all the URLs as well as online resources present in the database have been labeled according to whether they perform web tracking or not, using a state-of-the-art technique called TrackSign [8]. This information should facilitate and speed up any research experiment looking for new web tracking detection methods, as well as experiments for many different online aspects, such as studying the characteristics of the URL graph formed by all the included domains, identifying characteristics of important actors like social networks or CDNs, finding security threads in the form of non-secured requests, or locating the most privacy-threatening resources.
Who can benefit from these data?
Unfortunately, there are only a few previous datasets containing extensive information about online data:

1. The "Common Crawl" [9] dataset is periodically updated and contains information at the URL level from billions of websites. Its main advantage is the massive volume of data. However, the main drawbacks are its lack of complete relation information (i.e., which website loaded each URL), and that the link information is only collected from HTML code, despite the fact that nowadays most websites use JavaScript to dynamically modify their DOM and load online resources. Moreover, the dataset does not contain information about tracking. In contrast, our dataset is presented as a relational database with information about the domain that explored each URL and its inner resource. Moreover, each online resource or URL is already labeled as tracking or not. On top of that, our methodology allows us to obtain complete information even from dynamically modified content using JavaScript API calls.
2. In the "Tracking the Trackers" [10] dataset, the authors explored the "Common Crawl" dataset, looking for URIs within the HTML and JavaScript files to relate them and create a "semi-interconnected" version of it. Moreover, they added information about common tracking domains to explore some of their characteristics. However, the URL relations are obtained directly from the code, which skips most of the dynamically modified content. Moreover, the only tracking information included is obtained from comparing the domain and cookies from each URL with a list of about 300 already-known web tracking domains, lacking information about most of the current web tracking methods directly embedded within the website code (e.g., fingerprinting). In contrast, our dataset contains tracking information from two different web tracking detection methods, one using a list of URL patterns including several thousands of domains and rules, and the other exploring the actual code to find hidden web tracking systems embedded in the website.
3. Z. Yang and C. Yue [11] created a public web tracking dataset for their publication entitled "A Comparative Measurement Study of Web Tracking on Mobile and Desktop Environments". However, the tracking information contained in the dataset is extracted from the cookies or by looking for some specific JavaScript APIs commonly used for tracking purposes. Moreover, the population is relatively small, with only 8000 websites explored. In contrast, our dataset contains information about 1.5 million domains and more than 76 million URLs. Furthermore, our web tracking detection method does not depend on specific API calls, working in a generic way by finding code patterns common to most web tracking systems.

Besides these three datasets there are some others online datasets, but to the best of our knowledge, all of them contain information about a very limited population (a few dozen or at most hundreds of websites) or are not publicly available. Thus, research groups, online privacy-preserving entities, or even country policy regulators can use our dataset to explore a vast amount of information and perform useful studies. For instance, the relational information contained in the dataset has already been used to find the most pervasive web trackers on the Internet as well as the most popular web tracking systems embedded in third-party websites [8]. Nevertheless, it can also be used for complementary studies such as, for instance, collecting the web tracking code pieces for the subset of files inside the dataset detected as tracking, to study patterns within them that can expose new JavaScript APIs or new ways of using the already-known APIs being used for tracking purposes.
Moreover, the data contained in the dataset can also be used for other research problems not directly related to web tracking, such as, for instance, to explore the security status of the current Internet, using the security information about the SSL status and certificates loaded by each URL. This information can be used to relate subsets of websites using the same non-secure CAs, self-signed certificates, or even expired certificates. On the other hand, thanks to the relational information and the deduplication techniques applied to the files explored in the dataset, researchers and companies can study the most popular web services, as well as the most used online resources by looking at the number of URLs that load the same files. This data can be useful, for instance, for location optimization, looking at the geolocation information obtained for each file to explore unoptimized resources and copy them to CDN nodes closer to the final user. Furthermore, the relation information can be used to generate network graphs relating each domain/url/resource, and use machine learning techniques such as Grapchical Neural Networks (GNN) to find patterns that can characterize the most important nodes to be further explored.
How can these data be used for further insights and development of experiments?
As the dataset contains RAW information in the form of a relational database, researchers and developers can deploy it in their own environments and explore different aspects of the data directly by

executing requests through the RDBMS. The extracted data can then be used in data-mining algorithms specific to the experiment.

What is the additional value of these data?

First and foremost, the dataset includes a huge amount of information collected from a population of 1.5 million websites (4.5 million online resources and 75 million URLs) over a period of six months. Moreover, the data has been pre-labeled using two different web tracking detection algorithms. The first one, currently the most common web tracking detection method, is based on URL pattern lists. This labeling was done using uBlock Origin, one of the best adblockers currently available. On the other hand, the TrackSign discovery algorithm was executed on top of the obtained results, detecting millions of new URLs containing tracking that were not detected by pattern lists. Moreover, note that these data were obtained with a self-developed and open-source tool called ORM [1]. This allows other people to complement the data with a list of specific websites or even extend it by modifying ORM's code.

3. Data

The dataset is formed by a MySQL Workbench DB model file containing the database structure where the data is stored and a set of CSV files (one for each of the tables) containing the information. Fig. 1 presents the database model.

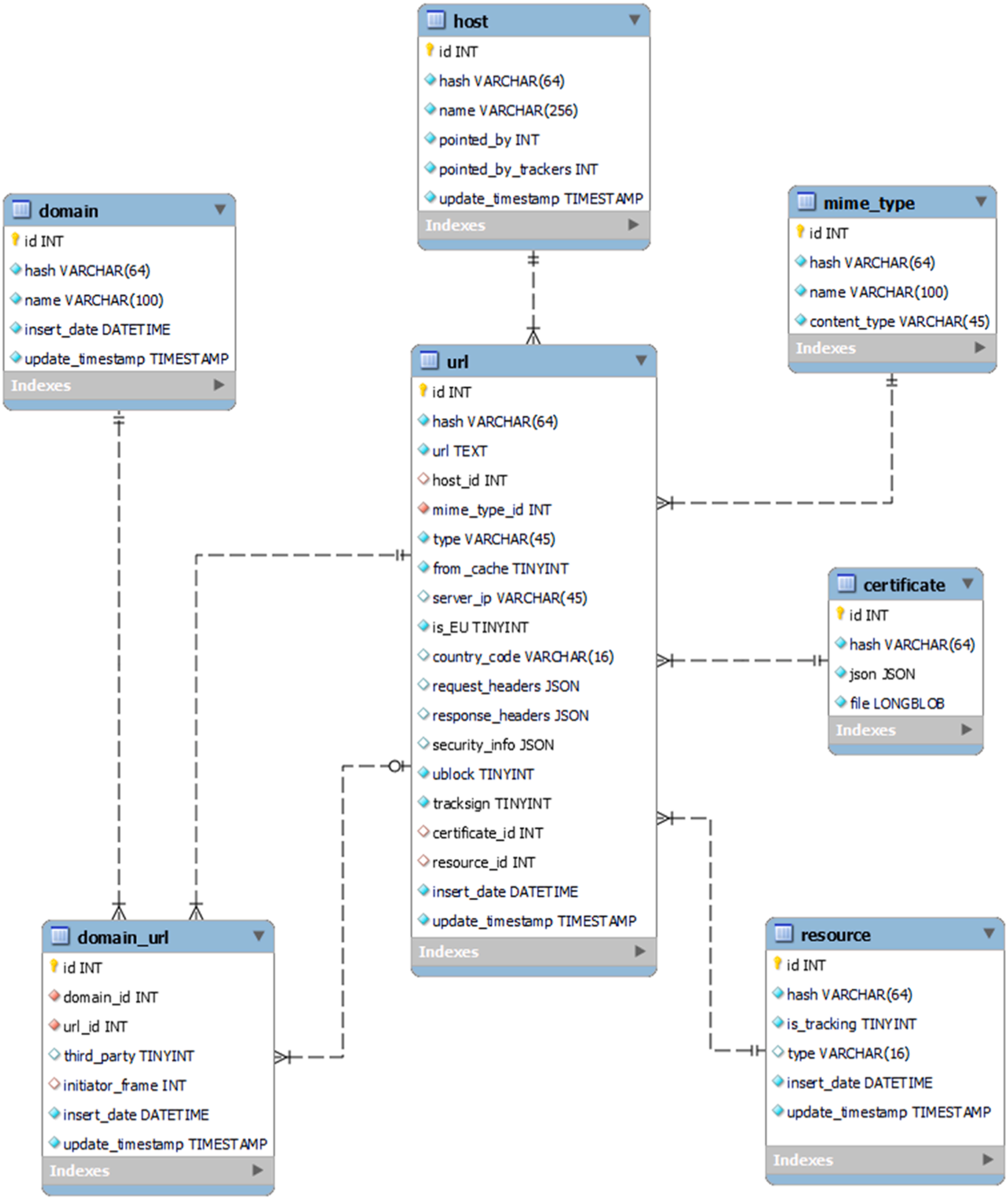

Fig. 1. Database diagram.

Each CSV file contains the data for the DB table named in the file. The following is a summary of the meaning of all the values in the database tables.

- domain: Main table containing a list of the domains included in the database.

Column	Type	Description	Example
id	INTEGER	Main identifier (primary key)	1
hash	VARCHAR (64)	Hash identifier of the domain (SHA256 (name))	"d4c9d9...1e664f"
name	VARCHAR (100)	Level 2 domain name	"google.com"
insert_date	DATETIME	Date of insertion in the database	2021-07-22 07:04:23
update_timestamp	TIMESTAMP	Timestamp of the last update	2022-03-04 06:44:31

- url: Table containing the URL list included in the database.

Column	Type	Description	Example
id	INTEGER	Main identifier (primary key)	4
hash	VARCHAR (64)	Hash identifier of the URL (SHA256 (url))	"ccb344...7fda45"
url	TEXT	URL string	"http://baidu.com"
host_id	INTEGER	Identifier of the domain hosting the resource pointed by the URL (foreign key: host)	47
mime_type_id	INTEGER	Identifier of the MIME type of the resource pointed by the URL (foreign key: mime_type)	1
type	VARCHAR (45)	Type of the resource as given by the URL headers	main_frame
from_cache	TINYINT	Boolean. 1 if the resource was already present in cache, 0 otherwise	0
server_ip	VARCHAR (45) or NULL	IP address of the servers hosting the resource if available	"220.181.38.148"
is_EU	TINYINT	Boolean. 1 if the country is within the EU, 0 otherwise	0
country_code	VARCHAR (16)	The ISO country code (2 or 3 letters)	"CN"
request_headers	JSON	The complete request headers in JSON format	{Host: baidu.com, User-Agent: Mozilla/5.0}
response_headers	JSON	The complete response headers in JSON format	{Server: Apache, Content-length: 81}
security_info	JSON	Security info as obtained by the browser in JSON format	{state: insecure}
ublock	TINYINT	Boolean. 1 if the URL was blocked by uBlock Origin, 0 otherwise	0
tracksign	TINYINT	Boolean. 1 if the URL was found to contain tracking by TrackSign algorithm, 0 otherwise	0
certificate_id	INTEGER or NULL	Identifier of the certificate obtained from accessing the URL (foreign key: certificate)	NULL
resource_id	INTEGER	Identifier of the resource pointed by the URL (foreign key: resource)	1
insert_date	DATETIME	Date of insertion in the database	2021-02-17 14:27:46
update_timestamp	TIMESTAMP	Timestamp of the last update	2022-03-21 14:26:40

- domain_url: Table containing the relation between domains and its loaded URLs.

Column	Type	Description	Example
id	INTEGER	Main Identifier (primary key)	1
domain_id	INTEGER	Identifier of the domain (foreign key)	4
url_id	INTEGER	Identifier of the URL (foreign key)	4
third_party	TINYINT	Boolean. 1 if the URL is third-party, 0 otherwise	0
initiator_frame	INTEGER or NULL	Identifier of the URL parent<->child relation (which URL loaded the current URL) or NULL if no parents	NULL
insert_date	DATETIME	Date of insertion in the database	2021-02-17 14:27:46
update_timestamp	TIMESTAMP	Timestamp of the last update	2022-03-21 14:26:40

- Host: Table containing a list of the hosts accessed by any obtained URL.

Column	Type	Description	Example
id	INTEGER	Main identifier (primary key)	1
hash	VARCHAR (64)	Hash identifier of the host (SHA256(name))	311933...31b874
name	TEXT	Host name	facebook.com
pointed_by	INTEGER	Number of domains loading at least one URL hosted by this host	226717
pointed_by_trackers	INTEGER	Number of domains loading at least one URL classified as web tracking hosted by this host	193217
update_timestamp	TIMESTAMP	Timestamp of the last update	2022-03-30 18:42:26

- mime_type: Table containing all the MIME types collected in the database.

Column	Type	Description	Example
id	INTEGER	Main identifier (primary key)	1
hash	VARCHAR (64)	Hash identifier of the MIME type (SHA256(name))	Bb4770...0fc25a
name	TEXT	MIME type name	text/html
content_type	VARCHAR (45)	String identifying the group containing this MIME type (e.g. Image, Media, Script)	Frame

- certificate: Table containing all the certificates collected in the database.

Column	Type	Description	Example
id	INTEGER	Main identifier (primary key)	1
hash	VARCHAR (64)	Hash identifier of the certificate (SHA256 (json))	8baff9...b7ee72
json	JSON	JSON containing the certificate information	{“subject”: {“commonName”: *. wikipedia.org”}, ...}
file	LONGBLOB	BLOB containing the gziped file of the certificate	0x789C7D...C8CA62

- resource: Table containing the resources loaded by the URLs.

Column	Type	Description	Example
id	INTEGER	Main Identifier (primary key)	1
hash	VARCHAR (64)	Hash identifier of the resource (SHA256 (file))	1d2d89...1431fb
is_tracking	TINYINT	Boolean. 1 if the resource does web tracking, 0 otherwise	0
type	VARCHAR (16)	Resource’s type	frame
insert_date	DATETIME	Date of insertion in the database	2021-02-17 14:27:46
update_timestamp	TIMESTAMP	Timestamp of the last update	2021-05-11 10:13:29

4. Experimental design, materials, and methods

The dataset was collected between June and December of 2021 using ORM [1] a tool developed by the Universitat Politècnica de Catalunya to perform online experiments. ORM uses a combination of Selenium [2], a tool for automation of website experiments, and Firefox [3] to open all the websites explored. Firefox runs a customized version of uBlock Origin [4], an adblocker plugin, that allows us to intercept all the URLs being accessed by the website and at the same time label them as web tracking/advertisements or not if they would have been blocked by uBlock Origin. Then, ORM opens once more each of the found URLs, collecting all kinds of information such as the HTTP headers (request and response), cookies, and URL characteristics (e.g. mime-type, encoding, security status). In the case of being transmitted over HTTPS, it also downloads and stores the corresponding certificates. Finally, it downloads the resource to identify it by means of its hash value. Wherever possible, ORM automatically deduplicates the information to optimize the space and identifies the same URLs, hosts, resources, etc. using a hash value as an identifier.

A subset of this dataset has been used for several publications. In [5], the data was used to extract a picture of the current status of web tracking on the Internet at the time and was included in the form of a tutorial about web tracking. In [6], the tracking information was used to feed a deep learning algorithm to automatically classify URLs as tracking or not based solely on the URL itself. In [7], the relation between domains, URLs, and files loaded by the URL was used to compute the effective page size of each domain once all the files loaded dynamically were acquired. In [8], the data was used as the initial ground truth for the research of a new web tracking discovery algorithm that we called TrackSign. The obtained label information has been included in the present dataset to allow other researchers to compare it to it or use it to further develop new web tracking detection algorithms.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

I have shared the link to my data at the Attach File step

Acknowledgments

This publication is part of the Spanish I+D+i project TRAINER-A (ref.~PID2020-118011GB-C21), funded by MCIN/ AEI/10.13039/ 501100011033. This work is also supported by the Catalan Institution for Research and Advanced Studies (ICREA Academia).

References

[1] Online Resource Mapper (ORM). https://github.com/CBA-UPC/ORM, 2022 (accessed 18 April 2022).

[2] Selenium. https://www.selenium.dev/, 2022 (accessed 18 April 2022).

[3] Firefox. https://www.mozilla.org/en-US/firefox/, 2022 (accessed 18 April 2022).

[4] uBlock Origin. https://ublockorigin.com/, 2022 (accessed 18 April 2022).

[5] I. Castell-Uroz, J. Solé-Pareta, P. Barlet-Ros, *Network measurements for web tracking analysis and detection: A tutorial*, IEEE Instrumentation & Measurement Magazine 23 (9), 50-57. 10.1109/MIM.2020.9289071.

[6] I. Castell-Uroz, T. Poissonnier, P. Manneback, P. Barlet-Ros, URL-based Web Tracking Detection Using Deep Learning, in: 2020 16th International Conference on Network and Service Management (CNSM), 2020, pp. 1–5, https://doi.org/10.23919/CNSM50824.2020.9269065.

[7] I. Castell-Uroz, R. Sanz-García, J. Solé-Pareta, P. Barlet-Ros, Demystifying Content-blockers: Measuring their Impact on Performance and Quality of Experience, IEEE Transactions on Network and Service Management (TNSM) (2022), https://doi.org/10.1109/TNSM.2022.3179267.

[8] I. Castell-Uroz, J. Solé-Pareta, P. Barlet-Ros, TrackSign: Guided Web Tracking Discovery, in: IEEE INFOCOM 2021-IEEE Conference on Computer Communications, 2021, pp. 1–10, https://doi.org/10.1109/INFOCOM42981.2021.9488842.

[9] Common Crawl. https://commoncrawl.org/, 2022 (accessed 22 December 2022).

[10] S. Schelter, J. Kunegis, Tracking the Trackers: A Large-Scale Analysis of Embedded Web Trackers, in: 10th International AAAI Conference on Web and Social Media, 2016.

[11] Z. Thang, C. Yue, A Comparative Measurement Study of Web Tracking on Mobile and Desktop Environments, in: 20th Pribvacy Enhancing Technologies Symposium, PETS, 2020.

Ismael Castell-Uroz (ismael.castell@upc.edu) is a Ph.D. student at the Computer Architecture Department of the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain, 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, anomaly detection, internet privacy and web tracking.

Pere Barlet-Ros (pere.barlet@upc.edu) is a Full Professor with the Computer Architecture Department of the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain, and Scientific Director at the Barcelona Neural Networking Center (BNN-UPC). He received the M.Sc. and Ph.D. degrees in Computer Science from UPC, in 2003 and 2008, respectively. From 2013 to 2018, he was Co-founder and Chairman of the machine learning startup Talaia Networks. His research has been integrated in several open-source and commercial products, including Talaia Polygraph, Auvik TrafficInsights, Intel CoMo and SMARTxAC. In 2015, he was awarded as the best entrepreneur of the UPC School of Informatics (FIB). More recently he has been awarded by the ICREA Academia Programme (2023).