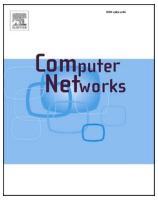
Computer Networks 205 (2022) 108783

Contents lists available at ScienceDirect

Computer Networks

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



# A dataset of labelled device Wi-Fi probe requests for MAC address de-randomization

Lucia Pintor [a,*], Luigi Atzori [a]

[a] University of Cagliari

ARTICLE INFO

*Keywords:*
Wi-Fi
Probe requests
MAC randomisation

ABSTRACT

Probe requests are management frames emitted by devices that perform active scanning to connect to Access Points nearby. These messages can be captured and analysed to implement device counting algorithms. However, using random MAC addresses to protect users' privacy challenges these algorithms, which must then perform address de-randomization (i.e., cluster the frames with the same source device by analysing valuable features). Datasets of labelled probe requests are needed to develop efficient de-randomization algorithms. Our dataset contains 20 min duration captures collected both in isolated and in "noisy" environments. Twenty-two different devices produced data in six different modes, including settings based on display status, Wi-Fi connection, and power saving. For each mode, we considered three channels contemporaneously for a total of 315 non-empty files. A Raspberry Pi captured the messages through a sniffing algorithm specifically designed to generate this dataset. We then filtered the data by deleting the messages from known sources and using power thresholds that exploit the burst structure of the probe requests. To the best of our knowledge, there are no other available datasets with labelled probe requests. This kind of dataset allows a more accurate analysis of the behaviour of individual devices in different modes and the training and test of algorithms for counting the number of devices through probe requests in the presence of random MAC addresses.

## Specifications Table

| | |
|---|---|
| **Subject** | Engineering |
| **Specific subject area** | The dataset consists of Wi-Fi packets of probe requests type, which have been captured in parallel in three non-overlapping channels (1, 6, and 11) |
| **Type of data** | Labelled Wi-Fi probe requests saved as PCAP files. The labels identify the source of the probe requests.<br>CSV tables |
| **How data were acquired** | Data was acquired via a Raspberry Pi 3 (Model *B+*) with three additional Wi-Fi interfaces supporting monitor mode. The software programs used are our Open Source scripts in Python, available on Github. |
| **Data format** | Filtered |
| **Parameters for data collection** | There are two types of data in the dataset: data collected in an anechoic chamber and data collected in a "noisy" environment. The first type was published after removing packets with Raspberry embedded interface MAC address; the second type has been filtered to simulate the anechoic chamber environment. |
| **Description of data collection** | After setting up the environment (e.g., removing all Wi-Fi devices except the Raspberry and the device to be analysed), we started the capture script and waited for its completion. This script starts the sniffing contemporaneously in three different interfaces, each set to a different channel, and saves the output on PCAP files. After that, we removed the packets emitted by the Raspberry (which uses its factory MAC address). In capture sessions outside the anechoic chamber, we performed additional filtering with our specific filtering algorithm. |
| **Data source location** | Institution: University of Cagliari<br>City/Town/Region: Cagliari, Sardinia<br>Country: Italy |
| **Data accessibility** | Our dataset is stored in the following public repository:<br>Repository name: Mendeley Data<br>Data identification number: 10.17632/j64btzdsdy.1<br>Direct URL to data: https://data.mendeley.com/datasets/j64btzdsdy/1 |
| **Related research article** | Author's name: M. Nitti, F. Pinna, L. Pintor, V. Pilloni, and B. Barabino<br>Title: iABACUS: A Wi-Fi-Based Automatic Bus Passenger Counting System<br>Journal: Energies (2020)<br>DOI: 10.3390/en13061446 |

(*continued on next column*)

(*continued*)

(*continued on next page*)

* Corresponding author.
*E-mail addresses:* lucia.pintor@unica.it (L. Pintor), l.atzori@unica.it (L. Atzori).

https://doi.org/10.1016/j.comnet.2022.108783
Received 30 August 2021; Received in revised form 25 November 2021; Accepted 6 January 2022
Available online 11 January 2022

(*continued*)

| | |
|---|---|
| | Author's name: M. Uras, R. Cossu, and L. Atzori<br>Title: PmA: a solution for people mobility monitoring and analysis based on WiFi probes<br>Book Title: *2019 4th International Conference on Smart and Sustainable Technologies (SpliTech)*<br>Pages: 1–6<br>Publisher: IEEE<br>DOI: 10.23919/SpliTech.2019.8783040<br>Author's name: M. Uras, R. Cossu, E. Ferrara, O. Bagdasar, A. Liotta, and L. Atzori<br>Title: WiFi Probes sniffing: an Artificial Intelligence based approach for MAC addresses de-randomization<br>Book Title: 2020 IEEE 25th International Workshop on Computer Aided modelling and Design of Communication Links and Networks (CAMAD)<br>Pages: 1–6<br>Publisher: IEEE<br>DOI: 10.1109/CAMAD50429.2020.9209257 |
| **Related project(s)** | Project name: Monifive<br>Funding body: Italian Ministry for the Economic Development (MISE), under the framework "Asse II del programma di supporto tecnologie emergenti (FSC 2014–2020)".<br>Type of grant: National grant<br>Project duration: 6 years<br>Project website (if available): http://moni5g.it/ |

### Value of the Data

- Our dataset can be helpful to analyse the sending patterns of probe requests from devices of various vendors and models.
- Our dataset is public. Anybody can access it and analyse it for both research and education purposes.
- The labelled traces in the dataset can be merged to simulate the presence of multiple devices together. These can be used to train and test machine-learning-based algorithms aimed at clustering packets generated by a single device to achieve MAC address de-randomization.
- Grouping probe requests by source device allows counting the number of devices in an area and thus estimating the number of people.
- We make the software program and procedures available so that other researchers can extend the dataset and make it even more powerful.

### Data

The data files contain messages transmitted in Wi-Fi channels in PCAP format (grouped by source device inside separate folders) and CSV tables. Various captures were made for each device with different setting combinations classified as A, S, PA, PS, WA, and WS. These modes can be subdivided into active-screen modes (A, PA, and WA) and inactive-screen modes (S, PS, and WS). The device kept the screen switched on during the whole capture in the active-screen modes by playing a video. On the contrary, the device kept the screen on standby in inactive-screen modes. Furthermore, power-saving modes (PA and PS) refer to captures in which the device kept the power saving setting active (all other captures are with this setting disabled). Finally, captures in WA and WS modes were made with the device keeping the Wi-Fi interface switched off, whereas, in all other modes, devices were keeping the Wi-Fi interface active without connecting it to any Access Point.

Table 1 summarises the settings of each mode with the three aspects considered. For example, the mode WS implies that the examined device has its screen on standby, its Wi-Fi interface is switched off, and the power saving setting is disabled.

For each device, six captures were collected (A, S, PA, PS, WA, WS). However, they were empty after filtering in some cases, especially when the Wi-Fi interface was turned off. These empty captures are not present in the dataset. The name of each PCAP file of the dataset contains information about the ID of the considered device, the timestamp of the capture, the selected channel, and the device setting.

**Table 1**
Device modes ("X" means that the relevant mode is "on").

| Mode | Active screen on | Wi-fi on | Power saving on |
|---|---|---|---|
| A | X | X | |
| S | | X | |
| PA | X | X | X |
| PS | | X | X |
| WA | X | | |
| WS | | | |

In the following example, we consider the device with ID A: the capture took place on 21st May 2021 at 11:57, considering channel 1, S mode (i.e. screen off, Wi-Fi on, and power-saving mode disabled) and power threshold −40 dBm.

A-ts-2021-May-21-h11-m57-s24-modeS-ch-1-th-40.pcap

The "files_list.csv" table shows the following information for each file to avoid any ambiguity:

1) Filename, 1) Device id (unique identifier), 3) Timestamp, 4) Device mode, 5) channel, and 6) power threshold for the filtering.

Details about each device are provided in the "devices.csv" table (partially shown in Table 2) :

1) "Device ID" is an acronym that identifies the device in the dataset.
2) "Device OS" is the Operative System that is running in the device.
3) "Device OS version" is the Operative System version installed in the device during the data collection.
4) "Device vendor" is the company that produced the device.
5) "Device model" is the model of the device.
6) "Anechoic room" is a boolean value that refers to the environment of data capture for that device; it defines whether it took place in an anechoic chamber (YES) or not (NO).

"Random MAC" is a boolean value that refers to the device's characteristic to use a random MAC address (YES) or the factory MAC address (NO).

"Factory MAC" is the factory MAC address (this field is empty if the MAC address is unavailable).

"Power Saving modes support" is a boolean value that defines whether the device supports power saving mode (YES) or not (NO).

The dataset contains Wi-Fi probe request captures in PCAP format divided according to the analysed device, considering three non-overlapping channels (1, 6, 11). Files were collected into an isolated environment (anechoic chamber of the Department of Engineering of the University of Cagliari) and in a "noisy" environment (area without specific shielding).

Each PCAP file contains data acquired from a single device, described in table devices.csv, including manufacturer, model, and operating system. The main feature of the dataset is precisely the subdivision by device, which allows a more accurate analysis of the behaviour of individual devices in different modes. Moreover, it is possible to label the data to train Machine Learning algorithms or to verify the correct functioning of algorithms that have as their objective the counting of devices through probe requests in the presence of random MAC addresses.

Experimental Design, Materials, and Methods

## 1. Introduction

Probe requests are packets broadcasted in plain text by Wi-Fi mobile devices to discover 802.11 Access Points (APs) in their proximity [1]. These unencrypted messages contain information about their sources (i.e., MAC address and supported data rate and supported connection to an AP). The operation of capturing data on a Wi-Fi channel is called sniffing

**Table 2**
List of the smartphones that were used to produce the dataset (extract of devices.csv file).

| Device ID | Device OS | Device OS version | Device vendor | Device model | Anechoic room | Random MAC |
|---|---|---|---|---|---|---|
| A | Android | 11 | Samsung | Galaxy M31 | YES | YES |
| B | Android | 6.00.01 | Xiaomi | Redmi 4 | YES | YES |
| C | Android | 4.02.02 | Samsung | Galaxy S4 | YES | NO |
| D | Android | 6.0 | Huawei | ALE-L21 | YES | YES |
| E | Android | 10 | Xiaomi | Mi A2 Lite | YES | YES |
| G | Android | 10 | Huawei | CLT-L09 (P20) | YES | NO |
| H | Android | 7.0 | Samsung | Galaxy S6 edge+ (SM-G928F) | NO | NO |
| I | Android | 8.00.00 | Samsung | Galaxy S7 | NO | YES |
| J | Android | 8.01.00 | Xiaomi | Redmi 5 Plus | NO | YES |
| K | Android | 10 | Samsung | Galaxy J6 | NO | YES |
| L | Android | 11 | Google | Pixel 3A | NO | YES |
| M | Ios | 14.05.01 | Apple | XS max | YES | YES |
| N | Ios | 12.05.02 | Apple | iPhone 6 | YES | YES |
| O | Android | Oxygen 11 | One Plus | Nord | NO | YES |
| Q | Android | 9 | Huawei | VTR-L09 (P10) | NO | NO |
| R | Android | 9 | Huawei | STF-L09 (honour 9) | NO | YES |
| S | Android | 10 | Xiaomi | Redmi Note 7 | NO | YES |
| T | Android | 11 | Xiaomi | Redmi Note 9S | NO | YES |
| U | Ios | 14.6 | Apple | iPhone XR | NO | YES |
| V | Android | 11 | Google | Pixel 3A | NO | YES |
| W | iOS | 14.05.01 | Apple | iPhone 12 | NO | YES |
| X | iOS | 14.6 | Apple | iPhone 7 | NO | YES |

and requires a Wi-Fi antenna that supports monitor mode and specific software to capture packets. This procedure has already been used to detect the presence of personal devices (by observing the unique MAC addresses in the captured traces) and consequently estimate the number of people in a given area [2]. However, major mobile device manufacturers [3] have developed algorithms that randomise the MAC address in probe requests to protect their customers' privacy since 2012.

The need for hiding factory MACs has become necessary because, even if this address does not contain any personal information, it might be linked to personal information through data cross-checking. Once the connection between the device owner's identity and the MAC is known, factory MAC addresses might become a way to track people [4]. Most modern mobile OSs contrast these privacy breaches by avoiding the transmission of unnecessary information (i.e. SSID fields are often empty), sending probe requests less frequently, and using random MAC addresses.

MAC address randomisation algorithms might be different depending on the Operating System installed in the device: a new MAC address might be assigned every time the screen is turned off, or at regular intervals, or every time the user interacts with it. MAC address randomisation challenges the counting algorithms that use MAC addresses as device identifiers. This functionality led to an evolution of these kinds of algorithms that now perform additional steps to group the messages that might have been produced from the same source by analysing valuable features of the sniffed messages [5,6,7]. This clustering does not compromise users' privacy because the factory MAC cannot be reconstructed. Moreover, once the tracked device moves away from the sniffer, it cannot be linked again to probe request streams collected previously.

Finally, most of the de-randomisation algorithms were validated by comparing the number of probe request clusters counted by each algorithm and the ground truth of the number of people in the observed area. This comparison has some flaws because the number of people might differ from the number of devices. Moreover, even if this comparison demonstrates good results, individual probe requests might not be clustered correctly. A more accurate analysis can be performed with labelled data, and we can verify how many probe requests are correctly assigned to each device cluster.

The present dataset includes captures of individual devices in isolated or pseudo-isolated environments in order to study the probe request pattern with different device operating modes. MAC randomisation is performed by 18 out of 22 devices of our dataset. Similar datasets might be produced with the guidelines provided in this paper.

## 2. Experimental design

This dataset was designed to provide detailed information about the device which had emitted each probe request to allow the analysis of the behaviour of each model separately. The need for this kind of dataset depends on the fact that, even though many probe request datasets are accessible and open-source [8,9,10], none of them considers devices separately. Distinguishing a device from another in an unknown environment might be unfeasible because there are no unique identifiers: IP addresses are not defined in probe requests, and MAC addresses might be randomised. Additionally, not knowing the environment and the position of each device hinders the usage of power thresholds to discriminating devices. These aspects also highlight the need for datasets with labelled Wi-Fi probe requests that are still missing in the literature.

The collection of probe requests from multiple devices that implement the randomisation of MAC addresses is complex because, even using power threshold filters, we cannot discriminate against more than one device at a time. Power level fluctuates depending on the distances between the emitting channel frequency of the source and the detecting channel frequency of the sniffer. Moreover, unpredictable noise and the structure of the environment might affect this measure. Due to this fact, considering different thresholds for more than one device might lead to errors.

In order to get labelled probe requests, a simple solution is to collect them separately for each device and then simulate the presence of multiple devices at the same time by modifying the timestamp of each probe request. For example, we can merge the capture of device A in mode PA and the capture of device B in mode S; since those captures are collected in different timings, we can define an offset equal to the difference of seconds between the first probe request of the first capture and the first probe request of the last capture. This offset might be increased with a random quantity and applied to all the timestamps of the second capture.

This method allows for the creation of synthetic captures of labelled probe requests with more than one device. Having data similar to the one we can collect in real-time (PCAP format) can be helpful for testing and training algorithms that directly count the number of people in an area. However, our dataset is not appropriate for algorithms based on power threshold (i.e. counting the number of devices or calculating the

relative locations of each source) because the original captures are collected in the same environmental conditions (same distance from the sniffer), and particular power thresholds have been used to filter data.

Furthermore, to ensure that each file contains probe requests from a single device, we verified that the anechoic chamber shielded any Wi-Fi communications before starting the data collection. To achieve this, we placed the sniffer inside it. We performed a background capture with the door closed: the only captured packets were emitted by the embedded interface of the sniffer, easily identifiable because it was using its factory MAC address. We repeated the same experiment keeping the door of the anechoic chamber open. We collected probe requests from unknown devices and management packets of the Wi-Fi Access Points of the University.

## 3. Materials

The sniffer comprises a Raspberry Pi 3 (Model $B+$) and three additional Wi-Fi interfaces supporting the monitor mode. The Raspberry has an embedded Wi-Fi interface that does not support the monitor mode. This fact made it necessary to search for extra antennas that can be powered through the embedded USB ports of the Raspberry, which is powered by a micro-USB cable.

In order to inspect three channels contemporaneously, we have chosen three low-consumption antennas that can be directly plugged into the Raspberry Pi. The three extra antennas are the same model, Realtek RTL8188CU, and support 802.11n connections in various modes, including the monitor mode. The minimum power of the signal that can be detected is $-110$ dBm.

Other devices involved in the data collection are the smartphones that produced the probe requests. A complete list of them is provided in Table 2. Most of these devices use Android OS (17) and the others iOS (5).

## 4. Method

The first phase of data collection took place in an isolated environment, the anechoic chamber of the Engineering Department of the University of Cagliari. We placed only our sniffer and a smartphone inside the anechoic chamber for each capture. Each device was sniffed alone, simultaneously on three channels for 20 min for each of the modes we described in the "Data" section.

After analysing the data collected in the anechoic chamber, we made additional captures in other environments. The analysis of this data allowed us to design a filtering algorithm based on power threshold for environments with a specific setting. To correctly use our filtering procedure, any undesired Wi-Fi interface within two meters from the sniffer must be removed, and the smartphone to analyse must be placed near the sniffer (within 20 cm). The radius of the free space around the sniffer was defined through experiments in which various captures had been made with sources (smartphones and other Wi-Fi devices) at different distances. At distances greater than 2 m, no power peak of signal equal to or over $-60$ dBm was detected, which is distant enough from the $-40$ dBm threshold used for filtering.

Our sniffing algorithm, running in the Raspberry, configured the monitor mode in all interfaces, set them to a specific channel, and started the data acquisition. The Python algorithm we used to record the captures is available in a public Github repository, "WiFi-Sniffer". This algorithm configures the sniffing interfaces, starts the sniffing in each interface, and saves collected data in different files for each channel.

The sniffing script is composed of two parts:

a) Configure interfaces (configure_interfaces.py)
b) Start sniffing sub-processes (start_sniffing.py)

Further details are provided in the code documentation.[1]

All packets that are not probe requests and all packets originating from the sniffer were removed. Our filtering algorithm performed additional steps in case of capture in a non-isolated environment to simulate anechoic chamber capture conditions via particular power thresholds that exploit the sending pattern of the probe requests. The dataset comprises captures of 22 different devices, of which eight were collected in the anechoic chamber and 14 in a noisy environment.

## 5. Filtering

The first step of the filtering is the removal of packets originating from a list of known interfaces (e.g., those of the Raspberry). Access Point MAC addresses are added to this list next. AP MAC addresses are identified by observing the management packets that characterise the behaviour of these devices. MAC addresses from Access Points are easily identifiable because they do not change during time and because APs also send other types of management messages (probe response and beacon messages) with the same MAC they use for probe requests. Later this list is used to discard all the packets that use one of these addresses as the source.

The second step of the filtering uses a particular power threshold that takes advantage of the burst structure. From the experimental tests in the anechoic chamber, we have verified that iOS devices transmit packets with almost the same power level in all channels. In contrast, Android devices have more variable power values. Android devices send a series of packets in short time intervals (which we call bursts) followed by pauses of a few minutes in which nothing is transmitted. During the burst, all packets maintain the same MAC address. Also, in the case of some Android devices, if the screen remains active, the MAC does not change even in different bursts.

Our filtering algorithm "SnifferFiltering" is composed of four parts:

a) Group file names (get_data_from_filename.py)
b) Convert data in Python structures (merge.py)
c) Power threshold filtering (analysis_ds.py)
d) Statistics and chart generation (statistics.py)

Further details are provided in the code documentation.[2]

## 6. Conclusions

Our dataset might be used to achieve different purposes related to probe request analysis. Some use cases of this dataset might be: i) assessment of the performance of de-randomization algorithms; ii) training of new machine learning-based algorithms to improve clustering performance; iii) analysis of other elements such as the Information Elements, Sequence numbers or burst structures.

## 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.

*Acknowledgements*

This work has been partially funded by the Italian Ministry for the Economic Development (MISE), under the framework "Asse II del programma di supporto tecnologie emergenti (FSC 2014–2020)", project Monifive.

[1] https://github.com/luciapintor/WiFi-Sniffer
[2] https://github.com/luciapintor/SnifferFiltering

*Luigi Atzori*

Luigi Atzori (PhD, 2000) is a professor of Telecommunications at the University of Cagliari, where he leads the activities of the Net4U laboratory (Network for Humans) with around 20 affiliated researchers. Since 2018, he has been the coordinator of the master degree course in Internet Technology Engineering at the University of Cagliari. His research interests fall in the area of the Internet of Things (IoT), with particular reference to the design of effective algorithms for the realisation of social networks amongst connected devices to create the Social IoT paradigm. His interests also fall in the area of Quality of Experience (QoE), with particular application to the management of services and resources in new generation networks for multimedia communications. Lately, he also applied the study of QoE to IoT services. He regularly serves in the conference organising committee of the sector and as an associate and guest editor of several international journals (IEEE IoT journal, Ad Hoc Networks, IEEE Open Journal of the Communications Society, IEEE Communications Magazine, etc.).

*Lucia Pintor*

Lucia Pintor is a PhD student in the Department of Electrical and Electronic Engineering at the University of Cagliari. She has been a researcher in Smart Mobility at the University of Cagliari since 2018. Her current research is concerned with Mobility as a Service (MaaS) and Smart Cities. She worked on monitoring people flows by analysing Wi-Fi messages and on-demand transport services for low-demand areas integrated with traditional public transport.

## References


[1] E. Fenske, D. Brown, J. Martin, T. Mayberry, P. Ryan, and E. Rye, "Three Years Later: a Study of MAC Address Randomization In Mobile Devices And When It Succeeds", in *Proceedings on Privacy Enhancing Technologies*, 2021. 164-181. 10.2478/popets-2021-0042.

[2] A. Di Luzio, A. Mei, and J. Stefa, "Mind your probes: de-anonymization of large crowds through smartphone WiFi probe requests", *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, 2016, pp. 1–9, doi: 10.1109/INFOCOM.2016.7524459.

[3] M. Vanhoef, C. Matte, M. Cunche, L.S. Cardoso, and F. Piessens, "Why MAC Address Randomization is not Enough: an Analysis of Wi-Fi Network Discovery Mechanisms", in *Proceedings of the 11th ACM On Asia Conference On Computer and Communications Security* (*ASIA CCS '16*), 2016, Association for Computing Machinery, New York, NY, USA, pp.413–424, doi: 10.1145/2897845.2897883.

[4] M. Cunche, Mohamed Ali Kaafar and R. Boreli, "I know who you will meet this evening! Linking wireless devices using Wi-Fi probe requests", 2012, IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2012, pp. 1-9, doi: 10.1109/WoWMoM.2012.6263700.

[5] L. Oliveira, D. Schneider, J. De Souza, W. Shen, "Mobile Device Detection Through WiFi Probe Request Analysis", IEEE Access 7 (2019) 98579–98588, https://doi.org/10.1109/ACCESS.2019.2925406.

[6] M. Uras, R. Cossu, E. Ferrara, O. Bagdasar, A. Liotta and L. Atzori, "WiFi Probes sniffing: an Artificial Intelligence based approach for MAC addresses de-randomization", *2020 IEEE 25th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, 2020, pp. 1-6, doi: 10.1109/CAMAD50429.2020.9209257.

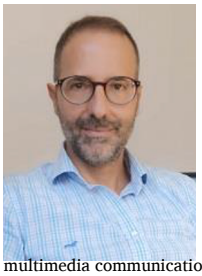
[7] Mario Vega-Barbas, Manuel Álvarez-Campana, Diego Rivera, Mario Sanz, Julio Berrocal, "AFOROS: a Low-Cost Wi-Fi-Based Monitoring System for Estimating Occupancy of Public Spaces", Sensors 21 (11) (2021) 3863, https://doi.org/10.3390/s21113863.

[8] Enrico Ferrara, Marco Uras, and Raimondo Cossu, "Probe requests of 24 devices in a semianechoic chamber", [Data set]. Zenodo. 2020. https://doi.org/10.5281/zenodo.3928500.

[9] Marco V. Barbera, Alessandro Epasto, Alessandro Mei, Sokol Kosta, Vasile C. Perta, Julinda Stefa, CRAWDAD dataset sapienza/probe-requests (v. 2013-09-10), downloaded from https://crawdad.org/sapienza/probe_requests/20130910, https://doi.org/10.15783/C76C7Z, Sep 2013.

[10] Pieter Robyns, Bram Bonné, Peter Quax, and Wim Lamotte, "Non-cooperative 802.11 MAC layer fingerprinting and tracking of mobile devices" [Data set], in *Security and Communication Networks*, Zenodo, 2017. https://doi.org/10.5281/zenodo.545970.


Lucia Pintor is a PhD student in the Department of Electrical and Electronic Engineering at the University of Cagliari. She has been a researcher in Smart Mobility at the University of Cagliari since 2018. Her current research is concerned with Mobility as a Service (MaaS) and Smart Cities. She worked on monitoring people flows by analysing Wi-Fi messages and on on-demand transport services for low-demand areas integrated with traditional public transport.



Luigi Atzori (PhD, 2000) is a professor of Telecommunications at the University of Cagliari, where he leads the activities of the Net4U laboratory (Network for Humans) with around 20 affiliated researchers. Since 2018, he has been the coordinator of the master degree course in Internet Technology Engineering at the University of Cagliari. His research interests fall in the area of the Internet of Things (IoT), with particular reference to the design of effective algorithms for the realisation of social networks among connected devices to create the Social IoT paradigm. His interests also fall in the area of Quality of Experience (QoE), with particular application to the management of services and resources in new generation networks for multimedia communications. Lately, he also applied the study of QoE to IoT services. He regularly serves in the conference organising committee of the sector and as an associate and guest editor of several international journals (IEEE IoT journal, Ad Hoc Networks, IEEE Open Journal of the Communications Society, IEEE Communications Magazine, etc.).