What Your Wearable Devices Revealed About You and Possibilities of Non-Cooperative 802.11 Presence Detection During Your Last IPIN Visit

Tomáš Bravenec*†, Joaquín Torres-Sospedra‡, Michael Gould* and Tomas Fryza†
*Institute of New Imaging Technologies, Universitat Jaume I, Castellón, Spain
†Department of Radio Electronics, Brno University of Technology, Brno, Czech Republic
‡Algoritmi Research Centre, University of Minho, Guimarães, Portugal
bravenec@uji.es – jtorres@algoritmi.uminho.pt – gould@uji.es – fryza@vut.cz

***Abstract*—The focus on privacy-related measures regarding wireless networks grew in last couple of years. This is especially important with technologies like Wi-Fi or Bluetooth, which are all around us and our smartphones use them not just for connection to the internet or other devices, but for localization purposes as well. In this paper, we analyze and evaluate probe request frames of 802.11 wireless protocol captured during the 11th international conference on Indoor Positioning and Indoor Navigation (IPIN) 2021. We explore the temporal occupancy of the conference space during four days of the conference as well as non-cooperatively track the presence of devices in the proximity of the session rooms using 802.11 management frames, with and without using MAC address randomization. We carried out this analysis without trying to identify/reveal the identity of the users or in any way reverse the MAC address randomization. As a result of the analysis, we detected that there are still many devices not adopting MAC randomization, because either it is not implemented, or users disabled it. In addition, many devices can be easily tracked despite employing MAC randomization.**

***Keywords*—MAC randomization, temporal analysis, privacy, probe requests**

I. Introduction

During the last decade, the development of new wireless technologies continued and with it the field of indoor positioning and indoor navigation. Positioning and navigation indoors are more difficult than outdoors, where Global Navigation Satellite Systems (GNSSs) are widely adopted. That is due to the inability of outdoor positioning system signals to penetrate the walls of buildings. In addition, the heterogeneity of indoor spaces makes positioning even more challenging. This means the systems for indoor positioning and navigation require different technologies. Since the technologies suitable for these applications are already deployed (Wi-Fi) or the beacons are easy to deploy (Bluetooth). With these, the issues of privacy come to the surface.

The authors gratefully acknowledge funding from European Union's Horizon 2020 Research and Innovation programme under the Marie Skłodowska Curie grant agreement No. 813278 (A-WEAR: A network for dynamic wearable applications with privacy constraints, http://www.a-wear.eu/) and No. 101023072 (ORIENTATE: Low-cost Reliable Indoor Positioning in Smart Factories, http://orientate.dsi.uminho.pt/). This work does not represent the opinion of the European Union, and the European Union is not responsible for any use that might be made of its content.

User privacy with wearable devices is a big topic, as it is difficult to find a balance between enough privacy and functionality. That is due to many services requiring user data to provide useful and helpful information. That is especially true in positioning applications, in which the position is not calculated on the device. That is due to the fact that the server needs some identifier to send the location data back as well as the signal strengths or channel state information and other information useful for localization. This information is known only to the server and is therefore transmitted using encrypted communications.

In this work, we focus only on the Wi-Fi communication protocol and its unencrypted management frames. These frames are not only used for managing the connection to an access point, but also for detection of nearby access points using probe request frames. The detected networks are then used either for connection to known networks saved in the preferred network list of the device or for approximate localization without global positioning systems. A side effect of using these scans for nearby networks is the ability of adversaries to collect these unencrypted frames.

The main paper contribution is a look at presence detection and user tracking at an isolated conference using not encrypted management frames of Wi-Fi protocol. We show the simplicity of tracking devices not employing MAC address randomization, as well as some devices that do. On the contrary, we also present the appearance of devices with well-implemented MAC address randomization.

In this paper we first provide an overview of related works in Section II, followed by Section III with description of current implementation of MAC address randomization depending on platforms. In Section IV in which we first provide information on the ethics of the capture of user data, which we follow with the description of the conference space. Furthermore, we describe the creation of the IPIN 2021 Probe Request dataset. Section V presents the results of the analysis of the created dataset, with several points of view on the gathered data. In the end, in Section VI we discuss the results, conclude the paper, and provide the lines for future work.

II. Related Work

The analysis of privacy weaknesses in management frames of 802.11 protocol is not new and has been explored for user tracking in the past. Out of the management frames, probe requests are most vulnerable to tracking.

In 2012 the authors of [1] exploited the availability of unique identifiers (MAC address) in the probe request for urban mobility tracking. Before the introduction of privacy measures considering Wi-Fi probe requests, specifically MAC address randomization in 2014 by Apple in the operating system iOS 8 [2], the Sapienza Probe Request Dataset was published [3], [4]. Several researchers already proved the vulnerability of probe requests before the implementation of MAC address randomization in [5], [6].

Since the introduction of MAC address randomization, researchers focused on exploring ways to bypass the newly introduced privacy measures and revealing the globally unique MAC address of each device using locally assigned MAC address [7]. One year after the introduction of MAC address randomization, researchers worked on reverse-engineering the MAC address randomization Apple had used [8]. Researchers also captured probe requests of Wi-Fi users in Italy during two political events and focused on figuring out the origin of the participants [9]. Their results were very closely matching the official voting reports. Other forms of analysis used temporal differences between subsequently transmitted probe requests to distinguish different devices [10]. The authors of [11] compiled a very comprehensive study of privacy-related measures in probe request frames of 802.11. This study did explore when exactly the MAC address randomization is not enough and when exactly it fails to protect the user privacy. In 2020 the authors of [12] successfully explored ways of protecting user privacy by encrypting probe requests, but the encryption of probe request frames was not adopted by the industry yet. As a follow-up to [11], another deep study of MAC randomization was published [13], which provides a deep look into the progress of protecting user privacy over the years.

Industry introduced the MAC address randomization in 2014 [2], but its implementation is still not perfect as sensitive information (enabling user's tracking) is still leaked. To explore the vulnerability of the implementations of MAC randomization we decided to use an environment of a scientific conference. The contribution of this paper is in the analysis of the current state of privacy-related measures in Wi-Fi probe requests around an isolated scientific event and the possibilities of non-consensual presence detection.

III. Current MAC Randomization Implementation

Implementation of randomizing MAC addresses is varying depending on the manufacturers and software developers. This fragmentation in implementation exists due to the lack of a commonly followed standard for MAC randomization. Few years after the MAC randomization was introduced, the specification of a standard amendment 802.11aq-2018 [14] was specified by IEEE SA Standards Board in 2018, but the implementation itself differs between manufacturers.

A. Identification of Randomized MAC Address

Even with the fragmentation in implementations, all implementations follow the setup of the two least significant bits in the first byte of MAC address as shown in Fig. 1. In case the 2nd least significant bit of first byte B1 is set, the MAC address was assigned locally by the network controller of the device. The least significant bit of the first byte B0 distinguishes between individual devices and device groups. Since randomized MAC address will always have bit B1 set to 1 and individual devices have bit B0 cleared to 0, recognition of randomized MAC address is simple. Due to the fixed values the two least significant bits can have, the 2nd digit of randomized MAC address in hexadecimal format has only four options: 2 (0010), 6 (0110), A (1010) or E (1110).

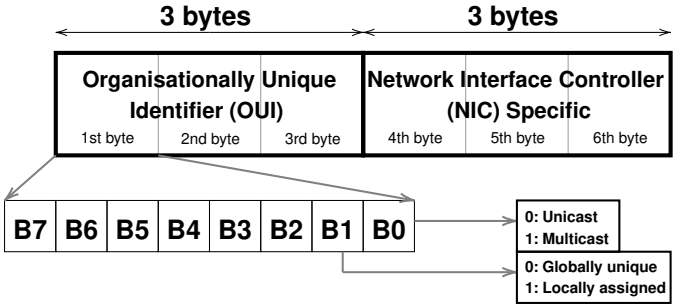

Fig. 1. Structure of MAC address with the functional bits

B. Google Android

The MAC address randomization was supported by Android since version 6, although the implementation of randomized MAC addresses for probing was first included with Android 8 [15]. Android 9 first implemented an option to connect to a Wi-Fi network using a randomized address, even though it was disabled by default and only available through developer options. Starting with Android 10, MAC randomization is enabled by default [15] for every new network and randomizes the MAC address for every SSID. The randomized address does not change as the generated address depends on the network profile (SSID, security and so on), non-persistent randomization that might change the MAC address for every connection to a network can also be enabled in the developer options [16].

In the first versions of Android supporting MAC address randomization, the system randomized only the last 3 bytes of the address, while using a fixed prefix for the first 3 bytes. Android devices using this older MAC randomization implementation had their randomized MAC address starting with DA:A1:19 prefix [11].

There is one catch to this though, even though the latest version is Android 12 at the time of writing (May 2022), barely any device is actually running it. Even though the majority of devices are running Android 10 or higher, there are still a lot of devices running older versions of Android, which do not use MAC randomization or any other privacy measures when it comes to probe requests.

C. Microsoft Windows

Similar to Google's Android, the operating system developed by Microsoft supports MAC address randomization per SSID. The MAC randomization is present in Microsoft's operating system since version 10 [17] and is turned on by default. The default option uses the same randomized MAC address for each SSID, and the MAC address stays randomized for the actual connection to the access point. This approach is the same as in Android and helps keep MAC address-based authentication working. On the other hand, there is an option to change the MAC address daily, but this needs to be enabled manually as this approach could cause issues with connection to some networks using MAC address authentication. The implementation of MAC address randomization in the newest version of the system developed by Microsoft stayed the same as in Windows 10, with MAC addresses staying the same for one SSID.

D. Apple iOS

Apple first introduced MAC address randomization with iOS 8 in 2014 [2]. Then later on iOS 10 added a tag in the information element of the probe request, which allowed for simple identification of iOS devices. The last change to the implementation of MAC address randomization happened in iOS 14. Since iOS 14 the devices use randomized MAC address per SSID just like Microsoft Windows and Android devices running the latest versions of their own systems. The implementation of MAC address randomization on iOS 15 [18] changed a little. The modifications were related to the changes in the locally assigned address of the device for one network. This happens on multiple occasions:

- on forgetting the network and reconnecting again,
- if the device did not connect to the network for 6 weeks,
- on-device content reset or network settings reset.

This all makes the implementation of MAC address randomization on Apple devices very robust, while not breaking existing systems with MAC address authentication.

E. Other devices

From other common operating systems, the implementation of MAC addresses again differs on the developer or the manufacturer of the device. Linux supports MAC address randomization since 2014, specifically kernel 3.18. [19]. For Apple devices, the support changes from device to device. The support for the non-iOS devices produced by Apple varies based on the generation of the product [20].

IV. Dataset Creation

Since the investigation into probe requests related to the IPIN conference required capturing a new dataset during the conference, we wrote a firmware for a Wi-Fi enabled microcontroller and used it to collect the probe requests transmitted in the 2.4 GHz frequency band. The probe request sniffer was active starting on 29 November, 08:22 until the end of the closing ceremony on 2 December, 13:02. During this time, we captured a total of 390 810 probe requests.

A. Probe Request Sniffer

The device we used for collecting probe requests is an ESP32 microcontroller with custom firmware available from Gitlab repository [21].

Since probe requests do not contain time information, the ESP32 first connects to a predefined Wi-Fi to download current time information. After getting the current time, the SD card is connected, and the wireless interface is switched to monitoring mode.

While in monitoring mode, every collected frame is checked for its type. In case of capturing a probe request, the frame is stored in a file on the SD card. All other frames are discarded and therefore, not recorded. The file is periodically saved and a new one gets to be created to prevent data corruption in case of power loss.

The sniffing of probe requests continues until the *Stop* button is pressed, which raises an interrupt which stops the data collection and safely disconnects the SD card. The firmware of the ESP32 is simplified in Algorithm 1.

Algorithm 1 Probe Request Sniffer

```
Initialize MCU peripherals and GPIO
Connect to Wi-Fi and download current time
Initialize and mount SD card
Check for existing files and open new pcap file
Reinitialize Wi-Fi in monitoring mode
Start probe sniffing task - run callback Received Packet
Start saving task - periodically run callback Save PCAP

callback Received Packet
    if packet.type = ProbeRequest then
        Write packet to file
    end if
end callback

callback Save PCAP
    Close current pcap file
    Open new pcap file
end callback

isr On Button Press
    Stop probe sniffing task
    Close pcap file and unmount SD card
end isr
```

The sniffer is detectable by other wireless devices only during the time acquisition period as it involves bi-directional communications. After downloading the current time, the Wi-Fi interface of the device is set to "monitoring only". The sniffer is collecting 802.11 probe request frames without being detectable as the wireless interface is not transmitting any packets. i.e., it is only passively receiving packets from the radio environment around it.

B. Ethics and Sensitive Information

One thing to mention about the captured data is the fact, that right after capture by the ESP32-based probe request sniffer, it does contain the user information. That is to reduce the computational complexity of the probe request sniffer and produce almost the same packet capture files as network analysis tools like Wireshark. The captured data are then exactly the same as transmitted by devices and contain sensitive information. Be it globally unique or randomized MAC addresses, leaking SSIDs from devices preferred networks list, up to the device manufacturers and device names.

Additionally, since the captured probe requests are from in-person and a quite isolated event, with minimal presence of people not participating in the IPIN conference, it is necessary to say, that first of all we anonymized the captured data in a way we could not get any personal information during the analysis itself. The anonymization was done by hashing fields containing personal information with SHA512. Using hashing algorithms on the user information ensures the anonymity of the users while preserving the option for analysis. The analysis was focused on the vulnerability of the management frames of the current implementation in the 802.11 protocol, not on linking private information to specific users.

Even though the captured data does contain real globally unique MAC addresses and many randomized ones, thanks to the anonymization we did before the analysis itself, it is not possible to link the specific individuals to a MAC address or identify anyone directly. The anonymized version of the dataset is publicly available from Zenodo repozitory [22].

C. Space Description

The conference took place in Lloret de Mar, Spain in Evenia Olympic Congress Centre from 29 November to 2 December. The only people present around the hotel lobby and near the session rooms from the beginning to the end of the conference were attendants of the conference, conference organizers, hotel employees and cleaning staff.

The entire conference space was around the lobby, with hotel rooms and hotel restaurants being far enough to not pose interference and capture probe requests from sources we did not care about, similarly the location of the conference was in a single-floor section of the hotel complex, which guaranteed that all of the collected probe requests were from the area of the actual conference. The probe request sniffer was placed under the stage in Session Room 2 as presented in Fig. 2. The entire conference space was in the radio range of the sniffer.

D. Data Description

During the four days of the conference, we captured 390 810 probe requests. The capture started 38 minutes before the first tutorial session, on Monday 29th November, 2021 at 08:22. The last probe request was then captured on Thursday 2nd December, 2021 at 13:02, just few minutes after the closing ceremony concluded. Unfortunately, we were unable to keep the ESP32 working past the closing ceremony, due to the preparation of the session rooms for the next event.

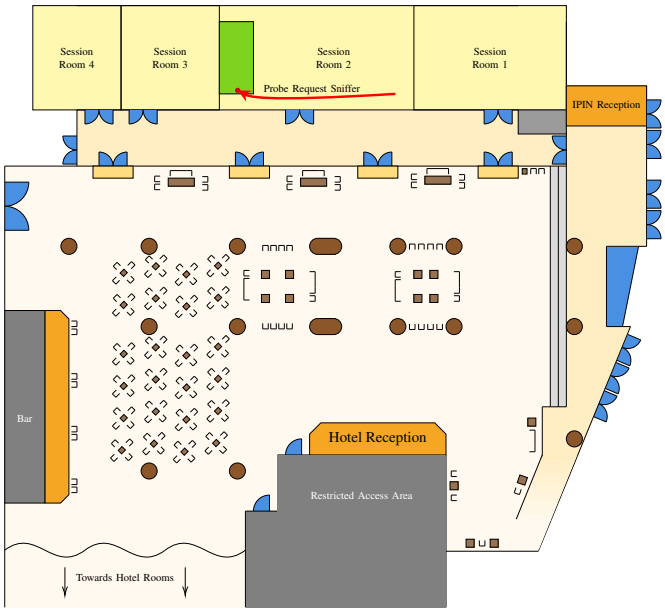

Fig. 2. Floorplan of the Evenia Olympic Congress Centre (Lloret de Mar, Spain)

The probe requests captured during the conference contained many of the additional fields carrying extra information about the capabilities of the origin device. Be it information about supported data rates, capabilities related to certain Wi-Fi standards (HT and VHT Capabilities), vendor-specific elements or Wi-Fi Protected Setup fields with UUID-E. All of the fields used for creating the fingerprint of each device are in Table I.

TABLE I
PROBE REQUEST FIELDS USED TO CREATE DEVICE FINGERPRINT AND FREQUENCY OF OCCURRENCE IN DATA COLLECTED IN OUR LAB

Information Element	Included in Probes	[%]
Supported rates	390 211	99.85
Extended Supported rates	385 606	98.67
HT Capabilities	359 391	91.96
VHT Capabilities	51 031	13.06
Extended Capabilities	312 181	79.88
Vendor Specific elements	228 970	58.59
1 Vendor Specific element	84 215	21.55
2 Vendor Specific elements	67 663	17.31
3 Vendor Specific elements	55 524	14.21
4 Vendor Specific elements	21 462	5.49
5+ Vendor Specific elements	106	0.03
WPS - UUID-E	3733	0.96
WEP Protected	599	0.15
Total Collected Probe Requests	390 810	

During the conference, we also captured more unusual types of probe requests, which did not contain any information other than 22 B long encrypted data sequence, which was identical in all 599 occurrences. These encrypted probe requests also in 44/599 cases contained randomized MAC address.

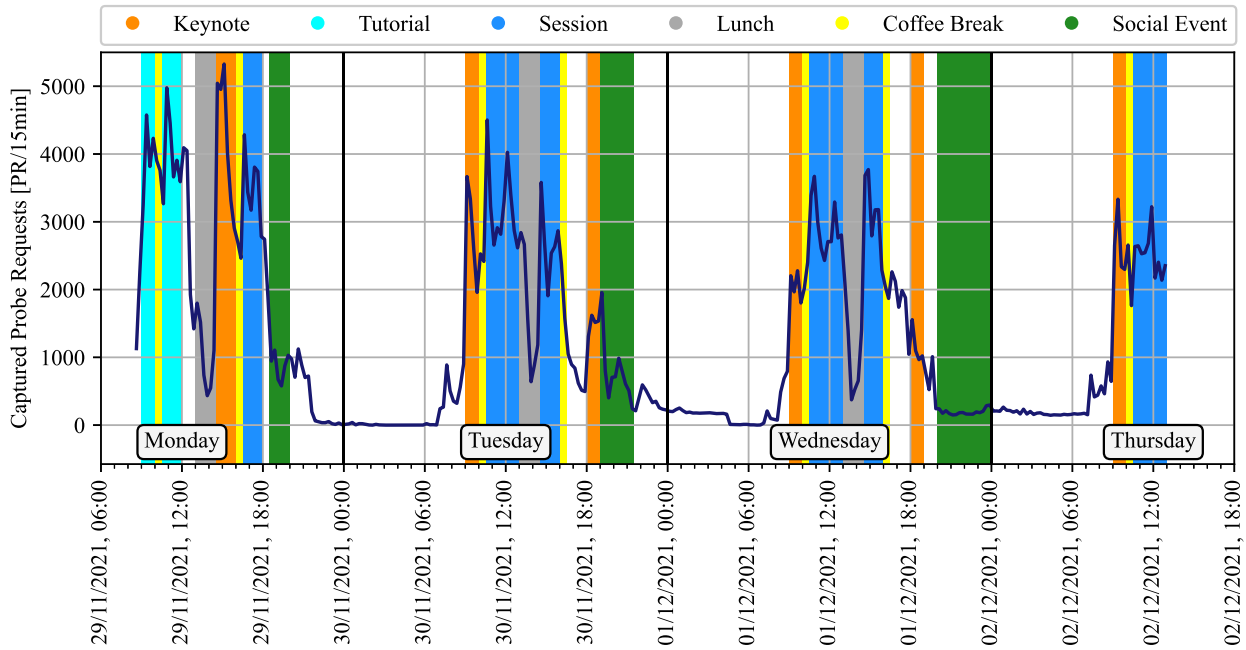

Fig. 3. Density of captured Probe Requests correlated with the program of the conference (amount of probe requests grouped in 15-minute clusters)

V. Analysis & Results

To analyze the gathered probe requests, we looked at the data from several perspectives. Using the gathered probe requests for crowd presence detection. Then we analyze the impact of MAC randomization on our ability to track individual users, both with MAC randomization on and off.

A. Presence detection

We counted the amount of probe requests received every couple of minutes, to see how big of a crowd could we detect from the density of probe requests. We chose a 15-minute interval as a compromise between readability and resolution of the final plot. From Fig. 3 we can see that during every keynote, tutorial or session the presence of users was much higher compared to the breaks in between the sessions. This can be caused by people turning off computers for the duration of coffee breaks. Quite a lot of people also left the range of the sniffer to go into the hotel restaurants for lunch. During the nighttime we can also see, if someone stayed in the lobby, be it for work or socializing. During the night from Monday to Tuesday, the lobby stayed empty with someone going through it but not staying long. The next two nights, the lobby was not empty during the night.

From the data plotted in Fig. 3 we can also see which keynote or session group (IPIN 2021 had 4 parallel session tracks) was more interesting to the participants of the conference. Unfortunately, due to the deployment of only one Probe Request Sniffer, we could not implement an indoor localization method based on RSSI to determine which session room the participants of the conference occupied at any time.

The Tuesday social event (Networking in the Kitchens) took place mostly out of the range of the probe sniffer in one of the hotel's restaurants. After the event, some of the participants stayed for further socializing, which can be seen on the small local peak right after the event ended. Another drop in received probes happened on Wednesday during the gala dinner, which took place in neighboring village and the presence in the conference space was minimal.

One of the noticeable trends is also the drop in the amount of captured probe requests during coffee breaks. This indicates people leaving the area either to get some fresh air outside of the hotel lobby, use the restroom or go to their hotel rooms. Since the amount of probe requests increases after each coffee break again, it is safe to assume that the conference attendants were coming back after each of the coffee breaks was over.

B. Analysis of user presence with global MAC address

It is no surprise that devices that transmit their real MAC address are very easy to track. Since we are able to identify probe requests using their globally unique identifier, their identification is very simple. At the IPIN 2021 conference, 28.62 % of identified scan instances (58 393 of 204 038 scan instances) used their globally unique MAC address. Since these devices used their real MAC address, we clustered the identified instances together and distinguished 229 individual devices without MAC address randomization through the duration of the conference. This data can be seen in Fig. 4. We then explored the presence of these devices in the proximity of our probe request sniffer. This presence in time proved to us how easy it is to track devices that do not employ any privacy-related measures related to the probe requests. The temporal presence of 10 devices using their real MAC address in the conference space is in Fig. 5 as an example of the simplicity of tracking these devices.

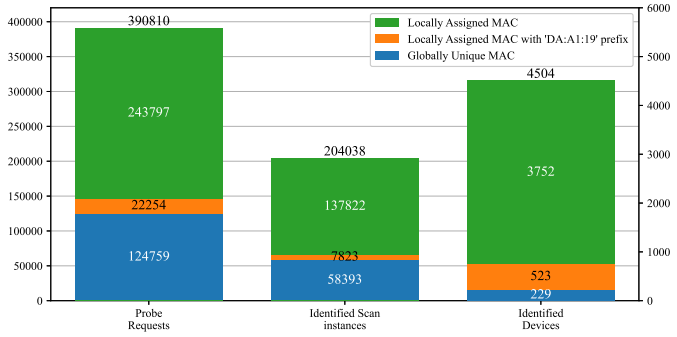

Fig. 4. Randomized MAC addresses in probe requests, identified scan instances and distinguished devices

C. Analysis of user presence with local MAC address

On the contrary, tracking devices which do not transmit unique identifiers is much more challenging. Out of the captured probe requests, 68.08 % (266 051) were using locally assigned MAC addresses. Since the devices do not change the MAC address they transmit during the Wi-Fi search burst, we were able to identify 204 038 different scan instances using just the MAC addresses available from the captured probe requests.

Out of all the transmitted probe requests with randomized MAC address, 22 254 used *DA:1A:19* as the first 3 bytes of their MAC address. We identified 7823 individual scan instances using this prefix. After we matched these instances together using fingerprinting of information elements, the similarity between transmitted Preferred Network Lists and the recurrence of the same randomized MAC addresses, we identified 523 devices using the *DA:1A:19* MAC address prefix. These data are presented in Fig. 4 with the comparison to the number of devices with a fully randomized MAC address and with a globally unique one. 50 of these devices then showed up more than 10 × (we chose 10 as a threshold as we found out that devices with more than 10 appearances are easy to track over time).

After identifying individual scan instances, we used the same approach consisting of fingerprinting, the similarity of preferred network lists and reappearance of MAC addresses to match together all other devices as well. With this approach we initially distinguished 4274 devices using locally assigned MAC address out of which 3752 randomized 46 out of the 48 bits in a MAC address. In this initial number of devices, 3544 appeared less than 10 times. On the other hand, 296 devices with fully randomized MAC address showed up more than 10 × which made them easily identifiable despite them using randomized MAC address, as can be seen from 10 example devices in Fig. 6.

D. Single Occurrence of Devices in Time Domain

From Fig. 4 we can see that the number of identified devices is still really high for just 3 full days in a conference space. Especially since the event space was primarily occupied by the attendants of the conference and hotel staff. The sniffer was also in the range of the sidewalk next to the entrance of the hotel. It is quite possible many of the single occurrences were just from pedestrians walking in the proximity of the sniffer. Another reason for this is a good implementation of MAC address randomization, the transmission of reduced information elements in the probe requests and omitting the transfer of SSIDs from the saved preferred network list. Representation of unmatched devices is shown in Fig. 7 with 10 examples.

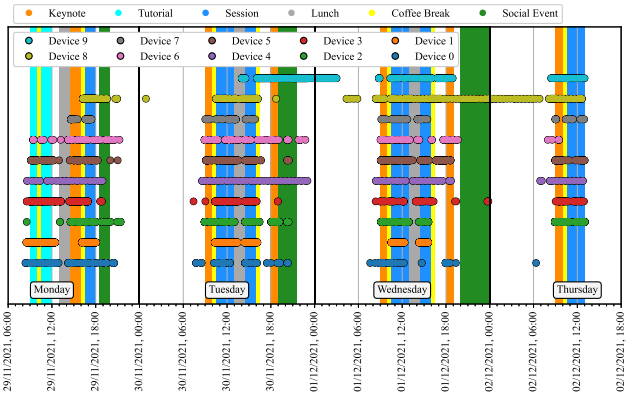

Fig. 5. Repeated occurrences of devices identified by the usage of globally unique MAC address

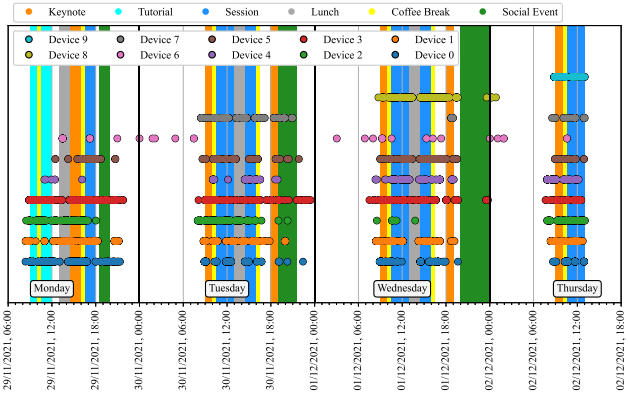

Fig. 6. Recurrent identification of the same devices despite using locally assigned MAC address

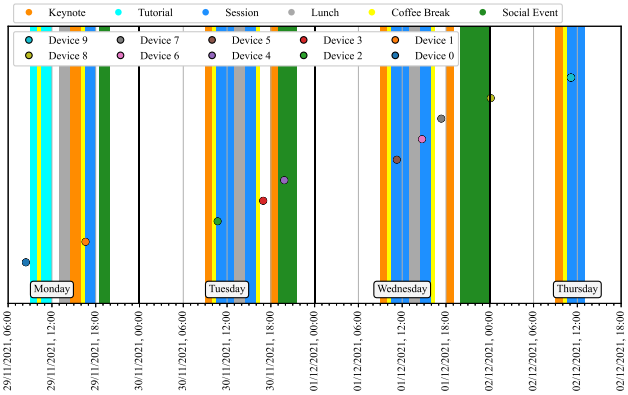

Fig. 7. Detected devices without identified recurrences in time

VI. Conclusions and Future Work

In this paper, we explored the possibilities of passive presence detection and tracking at 4 days-long scientific conference. We did this by exploiting unencrypted management packets of the 802.11 communication protocol. Specifically, we focused on probe request frames.

We used the ESP32 microcontroller with custom firmware for sniffing Wi-Fi probe requests. After the data collection, we analyzed the data and matched together devices that had the same MAC address, be it a globally unique one or a locally assigned one. We also used information elements fingerprinting and similarity of proffered network lists to further identify different scan instances as one device. Not surprisingly, the devices without a randomized MAC address were easily tracked. As we expected, devices employing MAC address randomization were more difficult to track, but even then, we identified 296 devices reappearing in time.

Even though the manufacturers are employing privacy-related measures like MAC address randomization since 2014, many devices are still easily tracked. That is not to say that MAC address randomization is not working, as we have seen with many appearances of only 1 or 2 scan instances. For these devices, the implementation of locally assigning MAC address either works well or we were also capturing probe requests from the pedestrians using the sidewalk next to the hotel lobby. In any case, we expect the situation to get better in time with older devices (with worse implementations of MAC address randomization) being replaced by newer devices after reaching their end of life.

To continue this research work, we are going to further explore privacy-related measures in wireless networks. We plan to study and search for user information leaks in Wi-Fi and other wireless technologies. Another point of interest is to extend this work with passive and non-cooperative indoor localization of users. We also plan to publish the dataset used in this work as part of a new publicly available Wi-Fi probe request dataset.

References

[1] A. B. M. Musa and J. Eriksson, "Tracking unmodified smartphones using wi-fi monitors," in *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, ser. SenSys '12. New York, NY, USA: Association for Computing Machinery, 2012, p. 281–294. [Online]. Available: https://doi.org/10.1145/2426656.2426685

[2] L. Hutchinson, "iOS 8 to Stymie Trackers and Marketers with MAC Address Randomization," Jun 2014. [Online]. Available: https://arstechnica.com/gadgets/2014/06/ios8-to-stymie-trackers-and-marketers-with-mac-address-randomization/

[3] M. V. Barbera, A. Epasto, A. Mei, V. C. Perta, and J. Stefa, "Signals from the crowd: Uncovering social relationships through smartphone probes," in *Proceedings of the 2013 Conference on Internet Measurement Conference*, ser. IMC '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 265–276. [Online]. Available: https://doi.org/10.1145/2504730.2504742

[4] M. V. Barbera, A. Epasto, A. Mei, S. Kosta, V. C. Perta, and J. Stefa, "CRAWDAD dataset sapienza/probe-requests (v. 2013-09-10)," Downloaded from https://crawdad.org/sapienza/probe-requests/20130910, Sep. 2013.

[5] M. Cunche, M.-A. Kaafar, and R. Boreli, "Linking wireless devices using information contained in Wi-Fi probe requests," *Pervasive and Mobile Computing*, vol. 11, pp. 56–69, 2014.

[6] N. Cheng, X. O. Wang, W. Cheng, P. Mohapatra, and A. Seneviratne, "Characterizing privacy leakage of public wifi networks for users on travel," in *2013 Proceedings IEEE INFOCOM*, 2013, pp. 2769–2777.

[7] J. Martin, E. Rye, and R. Beverly, "Decomposition of mac address structure for granular device inference," in *Proceedings of the 32nd Annual Conference on Computer Security Applications*, ser. ACSAC '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 78–88. [Online]. Available: https://doi.org/10.1145/2991079.2991098

[8] J. Freudiger, "How talkative is your mobile device? an experimental study of wi-fi probe requests," in *Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, ser. WiSec '15. New York, NY, USA: Association for Computing Machinery, 2015. [Online]. Available: https://doi.org/10.1145/2766498.2766517

[9] A. Di Luzio, A. Mei, and J. Stefa, "Mind your probes: De-anonymization of large crowds through smartphone WiFi probe requests," in *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*. IEEE, 2016.

[10] C. Matte, M. Cunche, F. Rousseau, and M. Vanhoef, "Defeating MAC Address Randomization Through Timing Attacks," in *Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, ser. WiSec '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 15–20. [Online]. Available: https://doi.org/10.1145/2939918.2939930

[11] J. Martin, T. Mayberry, C. Donahue, L. Foppe, L. Brown, C. Riggins, E. C. Rye, and D. Brown, "A study of mac address randomization in mobile devices and when it fails," *Proceedings on Privacy Enhancing Technologies*, vol. 2017, no. 4, pp. 365–383, 2017. [Online]. Available: https://doi.org/10.1515/popets-2017-0054

[12] X. Gu, W. Wu, X. Gu, Z. Ling, M. Yang, and A. Song, "Probe request based device identification attack and defense," *Sensors*, vol. 20, no. 16, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/16/4620

[13] 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," *Proceedings on Privacy Enhancing Technologies*, vol. 2021, no. 3, pp. 164–181, 2021. [Online]. Available: https://doi.org/10.2478/popets-2021-0042

[14] "IEEE 802.11aq-2018 - IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Preassociation Discovery." [Online]. Available: https://standards.ieee.org/standard/802_11aq-2018.html

[15] "Implementing mac randomization." [Online]. Available: https://source.android.com/devices/tech/connect/wifi-mac-randomization

[16] "Mac randomization behavior." [Online]. Available: https://source.android.com/devices/tech/connect/wifi-mac-randomization-behavior

[17] C. Huitema, "MAC address randomization in Windows 10," Dec 2015. [Online]. Available: https://huitema.wordpress.com/2015/12/31/mac-address-randomization-in-windows-10/

[18] Apple, "Use private wi-fi addresses on iphone, ipad, ipod touch, and apple watch," Nov 2021. [Online]. Available: https://support.apple.com/en-us/HT211227

[19] E. Grumbach, "iwlwifi: mvm: support random MAC address for scanning," Nov 2014, Commit effd05ac479b. [Online]. Available: https://github.com/torvalds/linux/commit/effd05a

[20] Apple, "Apple platform security: Wi-fi privacy." [Online]. Available: https://support.apple.com/en-gb/guide/security/secb9cb3140c/web

[21] T. Bravenec, "ESP32 Probe Sniffer," Nov 2021. [Online]. Available: https://gitlab.com/tbravenec/esp32-probe-sniffer

[22] T. Bravenec, J. Torres-Sospedra, M. Gould, and T. Frýza, "Supplementary Materials for "What Your Wearable Devices Revealed About You and Possibilities of Non-Cooperative 802.11 Presence Detection During Your Last IPIN Visit"," Jul. 2022. [Online]. Available: https://doi.org/10.5281/zenodo.6798302