POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Wireless Internet Project

Characterizing MAC Randomization in Modern Smartphones

Authors: **Matteo Bevacqua**

Manuela Merlo

Professor: **Alessandro Enrico Cesare Redondi**

Academic Year: **2022-2023**

Contents

1 | Introduction

1.1. Abstract

In recent years, privacy and security in wireless networks have become increasingly important concerns. To solve this problem, modern smartphones use a technique called MAC randomization to improve user privacy. MAC randomization involves using a temporary, artificial MAC address when sending probe requests in Wi-Fi networks instead of the original one.

The goal of this project is to create a dataset that characterizes the MAC randomization behavior of mobile devices. This characterization will be achieved by capturing probe requests issued under various conditions, following the methodology presented in the article ***"A dataset of labelled device Wi-Fi probe requests for MAC address de-randomization"*** [1] by L. Pintor and L. Aztori, published in Computer Networks, Vol. 205, March 2022.

The dataset will be used to examine the frequency of MAC address changes, identifying patterns or correlations and evaluating the effectiveness of randomization measures employed by smartphones. Visualizations and statistical measures will be used to improve understanding of the data.

1.2. What is active scanning

Wireless networks have become an indispensable part of modern connectivity, giving us the freedom to access the Internet and communicate wirelessly. The efficient management and use of these wireless networks relies on scanning techniques to discover and identify available Wi-Fi networks.

Active scanning involves a proactive approach in which a Wi-Fi device initiates a search for available wireless networks. During active scanning, the device sends probe requests, which are broadcast frames containing the device's unique identifier (MAC address) and a request for network information. These probe requests are used to query nearby wireless access points, prompting them to respond with probe responses that contain details about the network such as the network's Service Set Identifier (SSID), supported Wi-Fi standards, encryption methods and other configuration parameters.

The main advantage of active scanning lies in the precision and real-time information it provides. By actively soliciting responses from access points, the Wi-Fi device can gather accurate and up-to-date data about nearby networks, facilitating more informed decisions for network selection.

1.3. Privacy concerns

The use of probe requests in wireless networking can raise privacy concerns, particularly in public or densely populated areas. Probe requests, though essential for active scanning, can reveal sensitive information about Wi-Fi devices and their owners.
Below are the some privacy issues associated with the use of probe requests:

Device Tracking and Profiling: when a Wi-Fi device sends out probe requests, it includes its unique MAC address as part of the frame, making it possible to identify and track individual devices. In public spaces where multiple access points are present, these MAC addresses can be captured and logged by nearby Wi-Fi devices, creating a trail of the device's movement and presence over time.

SSID Leakage: in some cases, probe requests may include the SSID of the user's preferred or home network. While this helps the device connect quickly to known networks, it also means that the SSID information is broadcasted to any nearby Wi-Fi access points. Malicious actors can capture this information, potentially linking specific devices to particular networks and gaining insights into users' habits and behaviors.

Rogue Access Points and Wi-Fi Eavesdropping: rogue access points with the same SSID can be create as legitimate networks. Devices, by default, may automatically connect to networks with matching names, putting users at risk of connecting to malicious hotspots.

Privacy Inference: even when probe requests do not contain explicit personal information, patterns of device connections and behavior over time can be used to infer sensitive details about users.

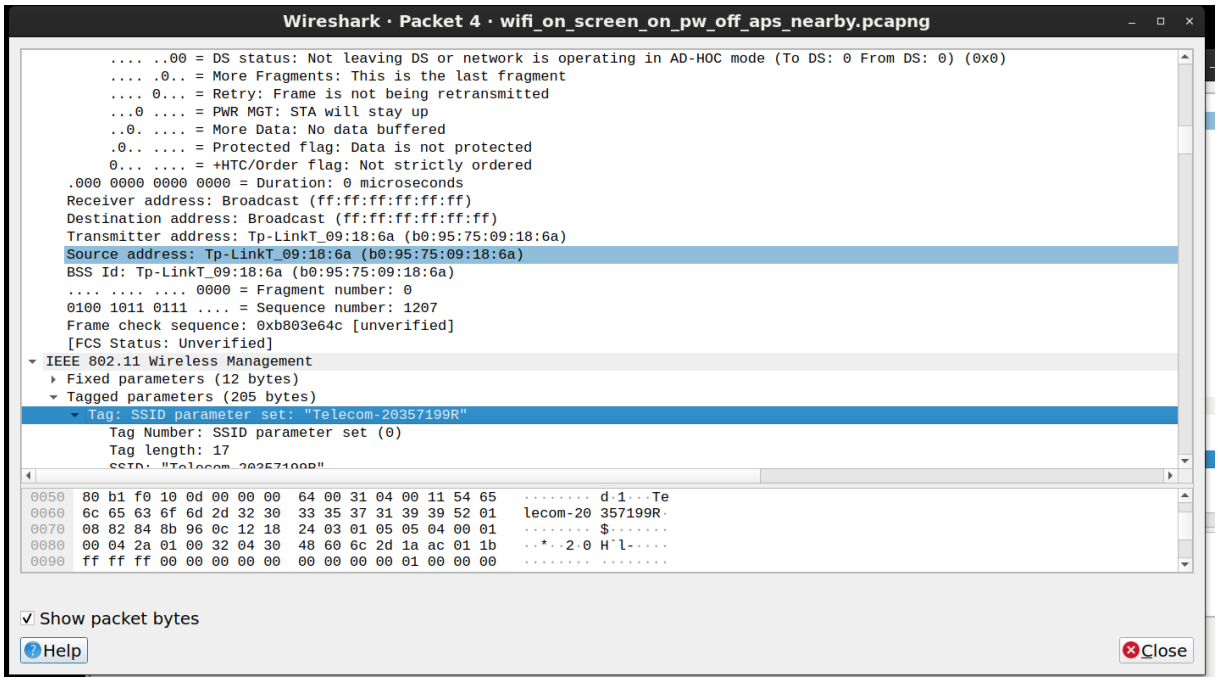


Figure 1.1: *MAC address and SSID fields in a Probe Request*

1.4. MAC randomisation

MAC address randomization is a privacy feature implemented in Wi-Fi devices to enhance user privacy and security. It works by periodically changing the device's Medium Access Control address. This helps prevent long-term tracking and profiling of users based on their MAC addresses, which change frequently, making it difficult to link a specific device to its owner.

1.4.1. How MAC Randomization Works

When a Wi-Fi device enables MAC address randomization, it generates a new, randomized MAC address at regular intervals or when connecting to different networks. This new MAC address is used for sending out probe requests and other Wi-Fi communication,

instead of the device's actual hardware MAC address.

The local bit is a single bit within the MAC address that determines whether the address is universally administered (0) or locally administered (1). In the context of MAC address randomization, the local bit is set to 1 in the generated random MAC addresses, indicating that the address is locally administered and not globally unique.

The randomized MAC address is changed periodically to further enhance privacy. The rotation interval can vary depending on the device's settings or the Wi-Fi network's configuration.

It's important to note that MAC randomization primarily addresses privacy concerns related to passive tracking and eavesdropping on probe requests. It does not protect against more sophisticated active attacks or prevent tracking based on other identifiers or network-related information.

This technique has gained traction and is widely supported by modern Wi-Fi devices, operating systems, and software. Major operating systems such as Android, iOS, macOS, and Windows have integrated MAC randomization capabilities into their Wi-Fi stack to provide enhanced privacy for users.

2 | Dataset

2.1. The environment

To build *"A dataset of labelled device Wi-Fi probe requests for MAC address de-randomization"* [1] researchers aimed to eliminate external signals that could impact the accuracy of the captured probe requests so the first phase of data collection took place in an isolated environment known as the **anechoic chamber**, a specially designed room that is engineered to minimize reflections of sound, electromagnetic waves, or other types of waves located within the Engineering Department of the University of Cagliari.

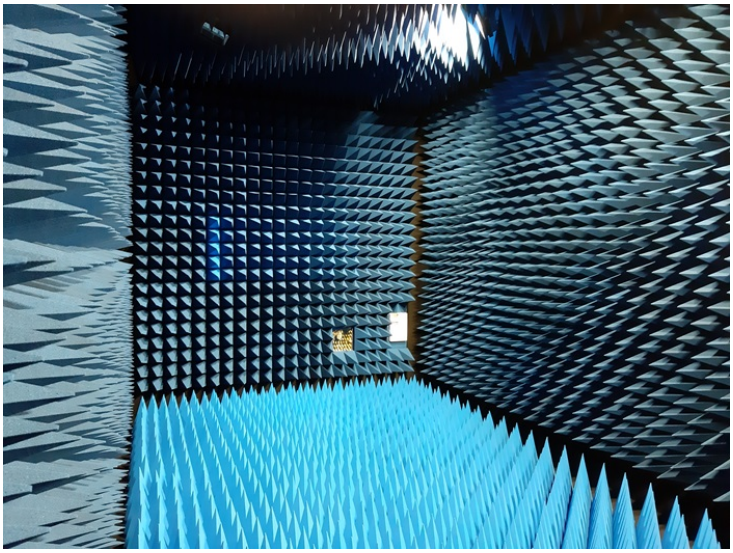


Figure 2.1: *Example of anechoic chamber*

During this phase, only the researchers' sniffer device and a smartphone were placed inside the anechoic chamber for each capture.

Since an anechoic chamber was not available, we took measures to mitigate external interference during data acquisition. To this end, we chose a garage that provided sufficient spatial separation from other Wi-Fi devices in the surrounding area.

2.2. Requirements

To proceed with the construction of the dataset, it was essential to have:

1. **Computer**: a device with a wireless network interface capable of operating in monitor mode.
2. **Wireshark** installed: a powerful open-source network analysis tool that provides the capability to capture, analyze, and dissect network packets, allowing for detailed examination of the data exchanged over a network.
3. **Mobile devices**: capable of generating probe request, captured under the following conditions:

Mode	Active Screen On	Wi-Fi on	Power saving on	APs Nearby
A	X	X		
S		X		
PA	X	X	X	
PS		X	X	
WA	X			
WS				
XA	X	X		X

The XA mode represents a unique dataset collected while access points were within range of the device, allowing it to receive their beacons. The combination of nearby APs, active WiFi, and active screen resulted in a limited number of probe requests being generated. Other types of datasets with other combinations while within range of Access Points were not reported because most battery-powered devices tend to switch to passive scanning mode when in close proximity to APs, leading to empty datasets.

To construct our dataset, we used the following mobile device:

Device ID	OS	OS Version	Vendor	Model
iPhone 8	iOS	16.0	Apple Inc.	8

Table 2.1: Mobile device's information

2.3. Capturing Probe Requests

To capture the probe requests of the chosen device, we followed the following step:

1. **Sniffing on the network interface**: this step has been automated using the scripts present in the .zip folder in the project, that interfacing with the AirPort suite of the sniffer device are able to select a specific channel and launch the capture.

2.4. Data Validation

The data validation process aimed to ensure the reliability and accuracy of the acquired Wi-Fi probe requests since, in the absence of an anechoic chamber, there was a possibility of external interference affecting the captured data. To mitigate this concern, we implemented the following validation criteria:

1. **Cross-referencing with known device information**: we cross-referenced the acquired probe requests with known device information, including MAC addresses and signal characteristics, to verify their accuracy. By comparing this information we could identify any inconsistencies or anomalies within the dataset.
2. **Signal Strength analysis**: to detect and remove probe requests from external devices that may have contaminated the capture, we utilized the received signal strength indicator (RSSI) of each packet. Probe requests originating from within the controlled experimental environment were expected to exhibit low signal strength values. By analyzing the RSSI we could exclude packets that deviated significantly from the expected range.

2.5. Pre-processing

During the data capture process using Wireshark, we were able to obtain output files in the ***.pcapng*** format. This is the native file format of Wireshark and represents network capture in "Packet Capture" format. Files with this extension contain detailed information about the captured network packets, including network, transport and application layer data.

2.5.1. Use of tshark for data preparation

During the pre-processing we have used **tshark** to:

1. **Convert the files**: we have converted the capture files, in the .pcapng format to

.csv files. This conversion made the data more easily manipulated and accessible using Python.

2. **Filter Probe Requests**: during the conversion, we have applied specific filters to select only the probe request packets. This process made it possible eliminating packets that were not relevant for further analysis.

3. **Exclusion of the contaminating device**: additional filters could be applied to exclude packets from an external device that had contaminated the capture.

```
to_csv = "tshark -r \"%s\" -R \"wlan.fc.type_subtype == 0x04 && wlan_radio.
  signal_dbm >= -40\" -2 -T fields -e frame.number -e frame.time -e wlan.sa -
  e wlan.da -e wlan_radio.signal_dbm -e wlan.ssid -E header=y -E separator=,
  -E quote=d -E occurrence=f > %s"
```

2.6. Data set Description

The constructed dataset includes a comprehensive collection of Wi-Fi probe requests issued by a device under different conditions during a specific time frame (20 minutes). It includes useful information for Mac address randomization analysis such as:

1. *frame number*;
2. *frame time*;
3. *wlan_sa*: source MAC address;
4. *wlan_da*: destination MAC address;
5. *wlan_ssid*: SSID of the network
6. *wlan_radio.signal_dbm*: signal strength.

frame.number	frame.time	wlan.sa	wlan.da	wlan_radio.signal_dbm
1	May 7, 2023 10:28:12.513169000 CEST	e2:38:0a:a1:4e:2d	ff:ff:ff:ff:ff:ff	-21

Figure 2.2: *Data set description*

Each file contained on average around 40 probe requests. It should be noted that the device with Wi-Fi turned off generated 0 probe requests in all condition, as such no .pcapng files are included under such conditions.

2.7. Data Analysis

To explore the dataset, we followed a series of steps to extract and analyze information from the captured probe request data. The Python code was implemented with the aid of numpy and pandas libraries.

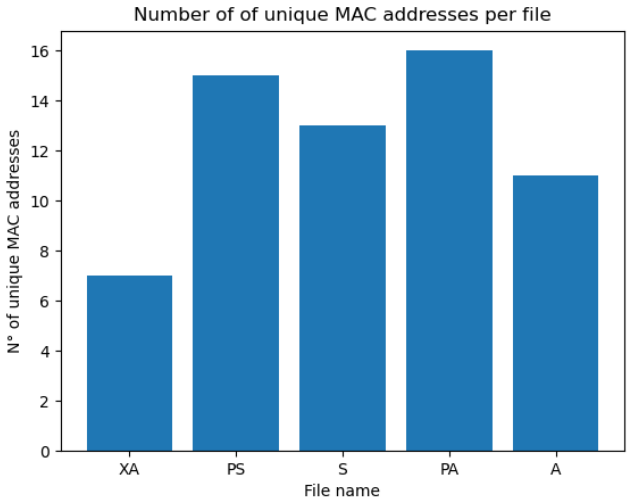
To get an overview and to be able to compare the behavior of Mac address randomization in different configurations, we have computed the following parameters for each file:

1. **Unique MAC addresses**: we identified the number of unique MAC addresses in the dataset by extracting the column "wlan.sa" from the DataFrame.

```
for file_name in captures:
    df = pd.read_csv(file_name)
    unique_macs = pd.unique(df['wlan.sa'])
    macs_per_file.append(len(unique_macs))
```

Device	A	S	PA	PS	WA	WS	XA
A	11	13	16	15	0	0	7

Table 2.2: Unique MAC addresses

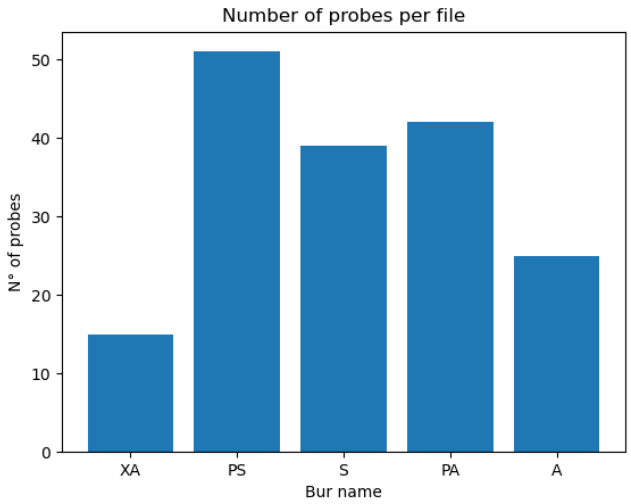


2. **Number of probe requests**:

```
for file_name in captures:
    df = pd.read_csv(file_name)
    number_of_probes_per_file.append(len(df.index))
```

Device	A	S	PA	PS	WA	WS	XA
A	25	43	43	51	0	0	15

Table 2.3: Number of probe requests



3. **Frequency of probe requests**: we calculated the average frequency of probe requests by calculating the time elapsed between the first and last request in each data file. This information provided insight into the rate [req/min] at which devices were issuing probe requests.

Device	A	S	PA	PS	WA	WS	XA
A	1.2350	1.8881	2.1644	2.8483	0	0	0.9605

Table 2.4: Frequency of probe requests

4. **In-burst probe request analysis**: we have analized:
   (a) the number of probe requests per burst [Figure 2.3];
   (b) the average number of requests in a burst per file [Table 2.5];
   (c) the average signal strength of the probe requests per burst [Figure 2.4];
   (d) the average delay between consecutive probe requests in a burst [Figure 2.5];

Device	A	S	PA	PS	WA	WS	XA
A	2.14	3.18	2.77	3.35	0	0	2.6

Table 2.5: Average number of probe requests per burst

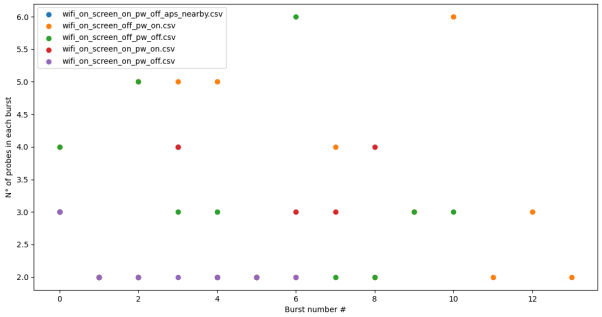


Figure 2.3: Number of probe requests per burst

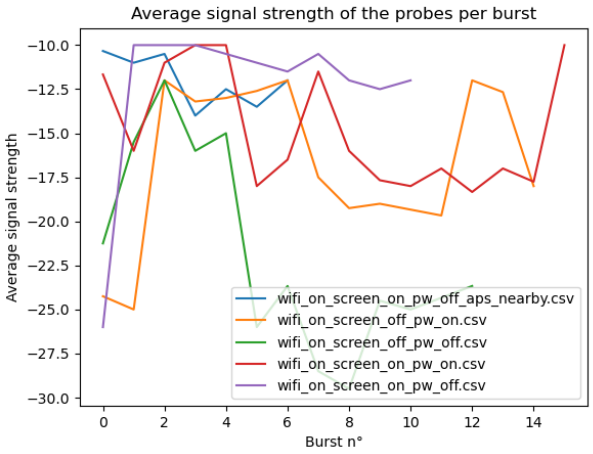


Figure 2.4: Average signal strength of probe requests per burst

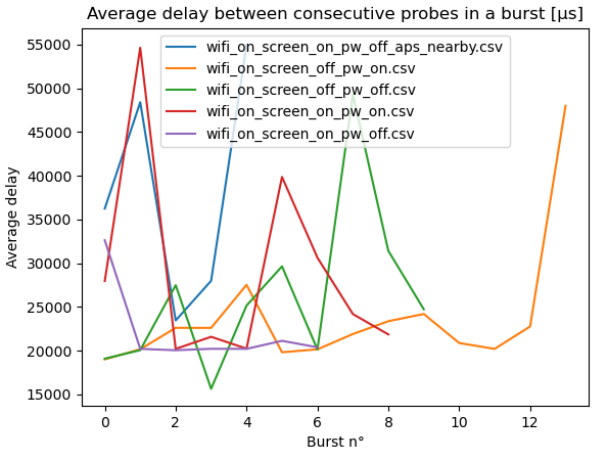
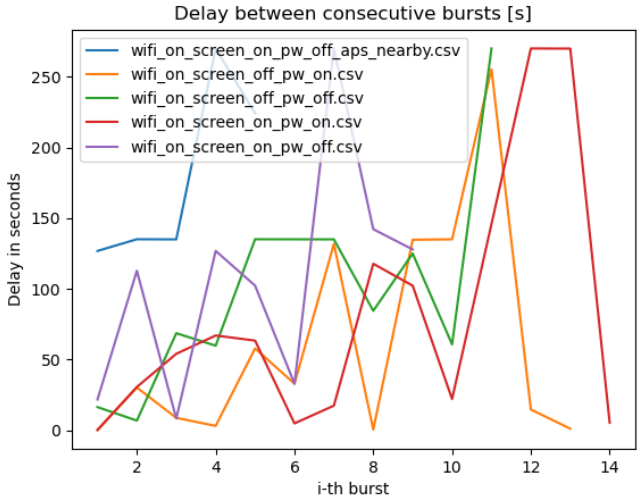


Figure 2.5: Average delay between consecutive probe requests in a burst

5. **Interburst interval**: we calculated the average delay between consecutive bursts by analyzing the timestamps of each burst. This allowed us to understand the temporal patterns and intervals between bursts of consecutive probe requests.

Device	A	S	PA	PS	WA	WS	XA
A	85 s	84 s	73 s	53 s	0	0	127 s

Table 2.6: Interburst interval

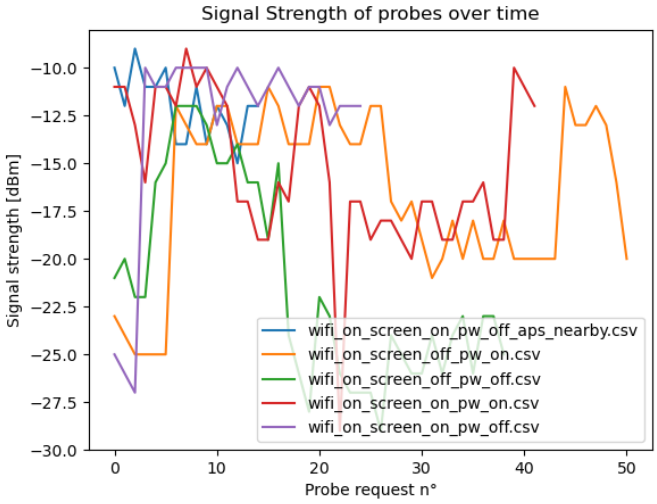


6. **Average signal strength**: we calculated the average power with which probe requests were received.

```
sum_rssi = 0
for index,row in df.iterrows():
    sum_rssi += row['wlan_radio.signal_dbm']
sum_rssi/=len(df)
log("Average signal strenght : %d" % sum_rssi)
```

Device	A	S	PA	PS	WA	WS	XA
A	-12	-22	-15	-16	0	0	-12

Table 2.7: Average signal strength over time



7. **Presence of Real MAC**: for each file, we checked whether the actual MAC of the device being used was present.

```
real_mac = "--:--:--:--:--:--"  #left darkened for privacy reasons
mac_contained = real_mac in unique_macs
log("Real MAC is contained in the probes : " + str(mac_contained))
```

Device	A	S	PA	PS	WA	WS	XA
A	False	False	False	False	Ø	Ø	False

Table 2.8: Presence of real MAC

8. **Local bit**: since MAC randomization is based on the locally administered MAC bit in the MAC address, we checked whether all MAC addresses had the locally administered bit that differentiates a random MAC address from a real MAC address set to 1. The result of the analysis reported that **100%** of the addresses had the local bit set to 1.

```
for mac in unique_macs:
        if(int(mac[0:2],16) & 0b10) : random_bit_set += 1
```

Note: To ensure focus on the analysis results, we chose to present only essential and concise parts of the source codes. Excessively long code has been omitted to avoid distractions from the main context of the report. The full version of the python code used in the report is available in the .zip folder.

2.8. Comparison of Results Across Different Modes

The data analysis revealed interesting variations in probe request behaviors across different modes, raising important considerations regarding the effectiveness of MAC address de-randomization. The trend we would have expected is as follows:

1. **Number of Unique MAC Addresses**: the variations in the number of unique MAC addresses among different modes should be attributed to the unique device settings and configurations for each mode. Modes that allow more active screen time and Wi-Fi usage should generate a larger number of unique MAC addresses as devices interact more with surrounding networks.

2. **Number of Probe Requests**: modes that involve more active Wi-Fi usage should generate a higher number of probe requests since the device continually seeks connections with available Wi-Fi networks.

3. **Average Frequency of Probe Requests**: modes with more active screen time and Wi-Fi usage should show higher probe request frequencies, reflecting the device's attempt to maintain network connections.

4. **Average Number of Probe Requests per Burst**: modes with more active screen time and Wi-Fi usage should exhibit a higher number of probe requests per burst to reflect the device's efforts to scan and connect to nearby Wi-Fi networks.

5. **Average Interburst Interval**: modes with more active Wi-Fi usage display shorter interburst intervals, indicating frequent and continuous Wi-Fi scanning by the device.

6. **Presence of Real MAC Address**: the absence of the real MAC address in all modes would be consistent with successful MAC address randomization.

Overall, the obtained results do not align with the expected behaviors of MAC address de-randomization in various modes. The variations observed can be attributed to the different settings and interactions of devices in each mode. Moreover, additional uncertainty is added by the software/firmware layer of the device to which we have limited to no access due to Apple's policies on the matter.

In particular, the mode that we would have expected to send fewer probe requests at shorter time intervals (PS) turns out to be the most efficient one.

Given the mismatch between the expectations and the actual outcome of the data analysis, we ran the captures twice [1] in order to make sure of the soundness of the conclusions and confirm this behavior.

The only behavior that we would expect that is demonstrated by the data is the locally administered bit being set to 1 in all captured MAC addresses.

2.9. Conclusion

Based on the data analysis, it can be concluded that MAC address randomization appears to be effective as all MAC addresses in the dataset had their local bit set to 1. In addition to that, all the probe requests didn't include the SSID of known networks of the device and as such avoiding possible tracking of the device based on the SSID of known networks. The analysis also revealed insights into the frequency and timing of probe requests under different conditions.
In conclusion, this dataset serves as a valuable resource for studying MAC address randomization and its effectiveness in different scenarios. Further research can build upon these findings to develop more robust strategies for enhancing user privacy and security in wireless networks.

[1] Only one of the two datasets is provided for the sake of brevity.

List of Figures

Bibliography

[1] L. Pintor and L. Aztori. A dataset of labelled device wi-fi probe requests for mac address de-randomization. *Computer Networks*, page 5, 2023.