Quantifying the Information Leak in IEEE 802.11 Network Discovery

Otto Waltari, Jussi Kangasharju

Quantifying the Information Leak in IEEE 802.11 Network Discovery

Otto Waltari and Jussi Kangasharju

Department of Computer Science
University of Helsinki, Finland

Abstract. Wi-Fi is often the easiest and most affordable way to get a device connected. When a device connects to any Wi-Fi network its identifier (SSID) is stored in the device. These SSIDs are sometimes intentionally exposed to the outside world during periodic network discovery routines. In this paper we quantify the information leak that is present in the current network discovery protocol. Our collected data shows how common it is for a device to leak information and what can be derived from the names of networks a user has connected to in the past. We introduce a way to measure the uniqueness of an entity, which is based on the set of leaked SSID names. We apply previously proposed methods of MAC address randomization reversal on our data and evaluate entity uniqueness. We show how unique SSID names backfire against attempts to obfuscate user devices. Finally we evaluate an existing alternative network discovery scheme that does not leak information.

1 Introduction

One of the most essential properties of smart phones and other mobile devices is the ability to stay connected to the outside world. Cellular data can provide connectivity in most areas where people spend their time. However, in many countries cellular data can be quite expensive, which often motivates users to utilize free Wi-Fi where ever it is provided by some local entity, e.g. shopping mall, airport, cafe or hotel. This is especially true when travelling and data roaming has an extra cost. Terms for using this kind of a public or free Wi-Fi is often displayed to the user upon connection, and it is up to the user whether he trusts the Wi-Fi provider.

What the user often does not know is that the name of each connected network is stored on the device in a *preferred networks list* (PNL). Due to design features in the IEEE 802.11 wireless standard [1] these network names are sometimes exposed to the outside world during so-called *active network discoveries*. Privacy preserving ways of network discovery have been proposed [6, 4], but our collected data set for this paper shows that still 30-40% of collected probe requests contain SSID names. Other studies [3, 2, 13] also show that probing is still widely used. Another concern in wireless networking is traceability of users. Since Wi-Fi is a wireless medium eavesdropping is trivial with any portable networked device. It can even be done without the subject device never knowing

that frames it transmitted were recorded by a third party. This was exploited by Pang et al. [9] over a decade ago. A popular countermeasure against tracking is MAC address randomization [5, 11], which has already made its way to recent mobile operating systems (Android 6.0 and iOS 8). However, several studies [12, 7, 8] have shown that MAC address randomization can be reversed.

In this paper we present a method to quantify the potentially private information that is leaking through exposed SSID names. We show that the network SSID names themselves have an impact on how unique a client device may become. We present a classification of SSID names based on how unique they are, and then introduce a measure to quantify the uniqueness of an entity based on its PNL. We show the uniqueness distribution of entities in a data set we collected. We apply de-randomization methods on our data set that reverses the effects of MAC address randomization and quantify the information value again and compare it to the results from the raw data. We also show how SSID leakage can be stopped with passive network discovery and measure the performance impact it has compared to active network discovery.

The rest of this paper is structured as follows. In Section 2 we explain the reasons why SSID names are exposed to the outside. Section 3 explains how we collected our data set, how we reverse the effects of MAC address randomization, and classify different types of SSID names. We also introduce a metric called *uniqueness*, which indicates how unique an entity is based in its PNL. In Section 4 we discuss our findings. In Section 5 we evaluate an alternative way of network discovery and evaluate it. In Section 6 we discuss related work, and finally conclude the paper in Section 7.

2 Background

The IEEE 802.11 wireless standard[1] specifies a network discovery protocol and a set of management frames designed for the purpose. A user device, or *station* (STA), that is looking for networks to connect to is periodically broadcasting probe request frames. A network access point (AP) may respond to the client with a probe response. If the user device decides to connect to an AP that responded the devices proceed to an authentication and association phase.

Probe requests can be either broadcast or directed. A broadcast probe has the broadcast address (`ff:ff:ff:ff:ff:ff`) defined as the *destination address* (DA), which also means that it can be received by anyone. Similarly, a directed probe has its destination set to the broadcast address, but in contrast to broadcast probes it has a *service set identifier* (SSID) configured in a designated header field. Addressing probe requests based on the access point MAC address would not work since one SSID can be offered by several access points. When a STA succesfully associates with a surrounding network its name, i.e. its *service set identifier* (SSID), is stored on the device in a *preferred networks list* (PNL). This is a mechanism to keep track of networks that the device has connected to in the past. IEEE 802.11 specifies that one *extended service set* (ESS) may consist

of more than two APs. Distinct APs in one ESS share the same SSID so that a client can authenticate with any one of them.

In most cases APs transmit periodic beacon frames in order to advertise their own SSID. However, it is not necessary for an AP to transmit beacon frames. One may also wait quietly for incoming probe requests that carry a network SSID that matches their own. This scenario would imply that the transmitter of that probe request has knowledge of that network from before. These kinds of WLAN networks are commonly known as hidden networks. They were designed to be more safe and secure, but studies[10] have shown that the absence of beacons from an access point did not increase safety against attackers, and was merely a false impression of security.

Despite the fact that STAs can find out about surrounding networks by listening to beacons sent by APs, and that hidden APs are a bad idea and should not be configured to be that way, modern mobile devices still transmit probes that contain SSID names of previously associated networks. Occasionally some devices expose large portions, or even all entries from their PNL. This is never explicitly told to the user and on most devices it happens automatically in the background as long as Wi-Fi is enabled on the device.

Not only does SSID names and locations reveal sensitive information, but a leaked PNL can also ruin the attempt of MAC address randomization. If a sufficiently unique PNL is received from two seemingly different devices, we can with high confidence map them back to the same user. Not necessarily the same device, since some mobile operating systems can sync their PNL over the cloud to multiple devices owned by the same user. Despite that, the user behind the PNL is still the same.

3 Methodology

In this section we present the data set used for our findings later presented in this paper. We also explain reverse randomization that we apply on our data set, and classify different types of SSIDs present in our data.

3.1 Collecting the data set

The data set used in this paper consists of six distinct capture files collected at different events and locations. For collecting the data we used a setup similar to the one described in our previous work [13]. For the sake of portability we only monitored channels 1, 6 and 11. These are the non-overlapping 2.4 GHz Wi-Fi channels that are recommended to be used when establishing new access points. Hardware we used for collecting the data was a Raspberry Pi 2 with three Wi-Fi adapters dedicated for the channels to be monitored. Data was stored locally on the device and it was powered from a USB power brick for wireless operation.

According to the findings in our previous paper [13] we do not have to monitor all available channels to increase our chance to capture a probe from a client device. Since devices scan for networks by transmitting bursts of probes in a

Table 1. Data set described in numbers

#	Data set	Probe count	Directed probes	Unique MACs	Total entities	Leaked PNLs	MAC address randomizers
1	EuroSys 2017	101.1 k	42.2 k (41.8%)	3558	2077	55.1%	608 (29.3%)
2	Pop concert	129.4 k	42.7 k (33.0%)	5225	2280	28.8%	543 (23.8%)
3	Workers day	96.9 k	33.3 k (34.4%)	10363	5541	25.3%	1376 (24.8%)
4	Movie	108.6 k	31.1 k (28.7%)	5869	2540	29.9%	678 (26.7%)
5	Shopping mall	98.4 k	32.4 k (33.0%)	7787	5567	30.8%	1030 (18.5%)
6	University campus	205.5 k	88.5 k (43.0%)	6824	2606	39.1%	652 (25.0%)

sweeping fashion through all available channels, we can safely assume that allocating 100% reception time for three evenly spaced and non-overlapping channels will capture the ongoing network discovery.

Table 1 has a listing of the data sets along with some essential statistics of the data. Set #1 was collected during EuroSys 2017 conference in Serbia. Set #2 was collected during a pop concert in Helsinki, with a predominantly teenage audience. Set #3 is collected during workers day evening celebration in a large outdoor park area in downtown Helsinki. Set #4 was collected in a movie theater during the title *Alien: Covenant*. Set #5 was collected while walking around at one of the busiest shopping malls in Helsinki. Set #6 was collected during two seminars at the Department of Computer Science at Helsinki University. All of the data sets are roughly the same size, except set #6 collected at our university campus. It is twice as large likely because there were more active Wi-Fi users (staff, researchers, students) present within range than in the other data sets.

3.2 Reverse randomization

A recent feature on mobile devices is to use fake MAC addresses when performing network discovery [5, 11], i.e. *MAC address randomization*. Purpose of this convention is to prevent tracking based on a device's static MAC address. This is achieved by intentionally changing the local MAC address and making the client device, i.e. entity look like several different entities. This would make it harder for an external party to keep track of the entity since its MAC address keeps on constantly changing. While a MAC address would be the obvious choice for a primary identification handle for tracking, randomization of the address has not been able to obfuscate entities completely. Studies have shown that MAC address randomization can be reversed [12, 7] and that it is not that effective.

For this paper we implemented de-randomization methods described by Vanhoef et al. [12] and Martin et al. [7]. These methods are; *i)* sequence control (SC) continuity, *ii)* information element (IE) fingerprinting, *iii)* locally/globally administered OUIs, and *iv)* PNL matching. We applied these methods to derandomize our data set and compared characteristics of the set before and after. Table 1 shows how much the data set was reduced and how many devices present in the set use MAC address randomization.

3.3 SSID classification

The SSID of a network is decided by the party responsible for the network. In companies and other organizations there may be a policy that defines naming conventions, but for personal purposes there are no rules or guidelines for naming. The only restriction is that it may not be longer than 32 characters. While the SSID is only an identifier for a *service set* (BSS/ESS), the choice of its textual content can reveal more than just the presence of that particular network. Based on our data set we divide SSIDs into five categories.

Globally scattered SSIDs are often used by fast food restaurants, coffee shops and other similar type big brand companies that have several locations around the world. A free Wi-Fi provided by such entities often has the name of the franchise, but does not indicate a particular site or location. From the business point of view it makes sense, since once a user connects to their network at one location, his device remembers the network in the next location and the service is available instantly. Such SSIDs are for example "Starbucks" and "McDonald's". The information value in these kinds of SSIDs is relatively low since they cannot be pinned to a location. However, they count as elements in the PNL vector.

Public location SSIDs are often present at sites like airports, hotels or shopping malls. These kinds of SSID names can be mapped back to a place somewhere. The name can be a subtle hint, or even explicitly tell what the location is. Services like Wigle[1] can assist in giving a location for the network in case it is not explicit. An examples of such an SSID would be "Helsinki Airport Free Wi-Fi". If an SSID like this is exposed from a user via his PNL, it does not reveal too much sensitive information about the user since travelling and using available Wi-Fi is quite normal.

Private location SSID is one that has been set up by a household for private use. These are often *obvious choices*, such as "Home Wi-Fi" for a home network, which has a relatively low information value. Another common practice is to include a name in the SSID, e.g. "Smith family Wi-Fi", in which case the information value is higher since it has a descriptor making it more unique. These are not unique on a global scale, but can be used to mark out user involvement with certain non-public locations.

A **unique SSID** is such that there likely is not another network by the same name. Our collected data shows that imagination plays a major role in ending up with a unique SSID. Another common way to unknowingly end up with a unique SSID is to leave it as it is by default on several ISP provided access points. The convention in this case is that the SSID has a suffix that matches with the three right-most octets of its own MAC address. This half of a MAC address is by design supposed to be unique, which gives high chances that SSIDs like "Telenet-12-3A-BC" are globally unique.

Portable access point SSIDs are by nature mobile and no guarantees can be be made about their location. The typical most common instance of such

[1] https://wigle.net/

network names are "AndroidAP" and "Alice's iPhone". These are the default hot-spot names when sharing a cellular data connection over Wi-Fi to other devices. No conclusions can be drawn about the location of such SSIDs. However, generic ad-hoc hot-spot SSIDs do count as elements in a user's PNL vector.

3.4 Entity uniqueness

In order to quantify how unique entities extracted from our data set are we introduce a metric called *uniqueness*. The uniqueness value of an entity indicates how likely there is not another entity that has PNL entries, i.e. known SSIDs in common. Uniqueness values are normalized to be between 0 and 1. A high uniqueness value means that the entity stands more out of the crowd and is less likely to have common PNL entries with other entities. A low uniqueness value indicates that the entity blends more into the crowd. For entities that do not transmit a single SSID uniqueness is defined to be 0.

Let entity e have a PNL with k distinct SSID names (1) and rank of n be the number of entites that have network n in their PNL (2):

$$PNL_e = \{n_1, n_2, ..., n_k\} \tag{1}$$

$$rank_{n_i} = |n_i| \tag{2}$$

First we calculate a significance value S for each SSID in e's PNL:

$$S_i = min\left\{\frac{|n_i|^{1+\frac{1}{k}}}{T}, 1\right\},$$

where T is the total number of distinct SSIDs in the dataset. The lower the significance value is, the more it contributes to the uniqueness of an entity.

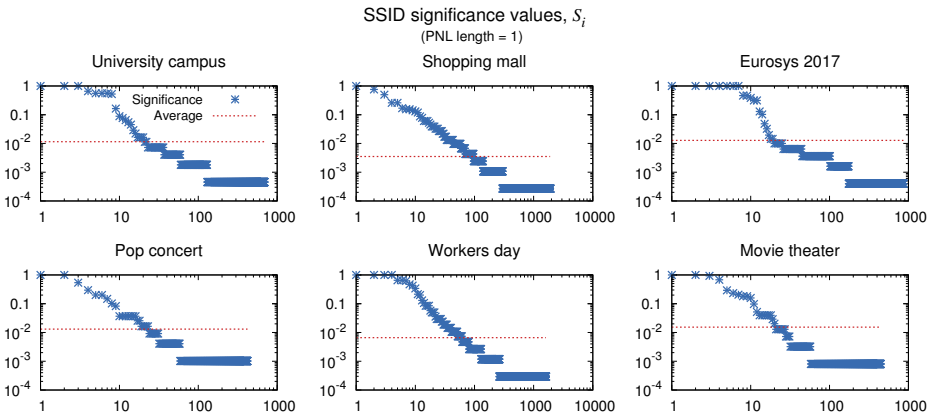

Fig. 1. Distribution of SSIDs significance values with an assumed PNL length of one. Y-axis represents significance values of distinct SSID names presented along the X-axis. Dashed line shows the average of all values. The lower the value, the more it contributes to the uniqueness of an entity.

Figure 1 shows the distribution of all SSID significance values with a PNL length of 1. The figure also shows the average of every S_i in each data set, which is the same as the expected value for any given SSID. If the significance value is equal to or higher than T it is not considered to be contributing to the uniqueness of the entity. This is common in the case where an entity has a PNL length of 1, and that single SSID is a popular one. A popular SSID has a high rank by nature, and k equals 1, which yields a high significance value. On the contrary, a single unique SSID yields a small significance value regardless of the PNL length.

We calculate the uniqueness value for a given entity e with the following formula:

$$uniqueness_e = 1 - \Big(S_1 \cdot S_2 \cdot ... \cdot S_k\Big).$$

In section 4 we use uniqueness as a metric to compare how much more unique entities become after we remove the effect of MAC address randomization from our data set.

4 Findings

For the results presented in this paper we used our data set described in section 3.1. We analyzed each part of the data set separately in order to maintain spatial context in each set. Our primary measure for quantifying the information leak in 802.11 network discovery is *uniqueness* introduced in section 3.4. In order to measure the uniqueness of an entity, we calculate the significance value for each SSID in that entity's PNL. Figure 1 shows the significance values of all SSIDs in the data sets. From the plots we can observe that the vast majority of dots have a small value. This means that most SSIDs have been heard from only a few different devices. The dots with higher values and closer to the top represent SSIDs that were heard from many devices. Considering the log-log scale, these are only about one percent of all SSIDs.

The CDFs in Figure 2 illustrate the uniqueness distribution of entities before and after de-randomizing the data set. The solid line is a cumulative distribution function plot of uniqueness values from the raw data. Next to it, the dashed line represents the same data, but after it has been de-randomized. After de-randomizing the number of entities in that data set reduces. This is because several MAC addresses can be mapped back to one single entity. Table 1 shows how many devices presented in the data employ MAC address randomization. The amount is based on entities in the de-randomized set that are linked to two or more distinct MAC addresses. Based on our results, roughly one fourth of devices use MAC address randomization.

From Table 1 we can also see that roughly 30-40% of all collected probes are directed probes, which means that they carry an SSID in the frame header. This number is about 10 percent points lower than what Barbera et al. [2] presented in their measurement results back in 2013. Directed probes are the reason how and why PNLs are exposed to the outside world. That being said, about every third

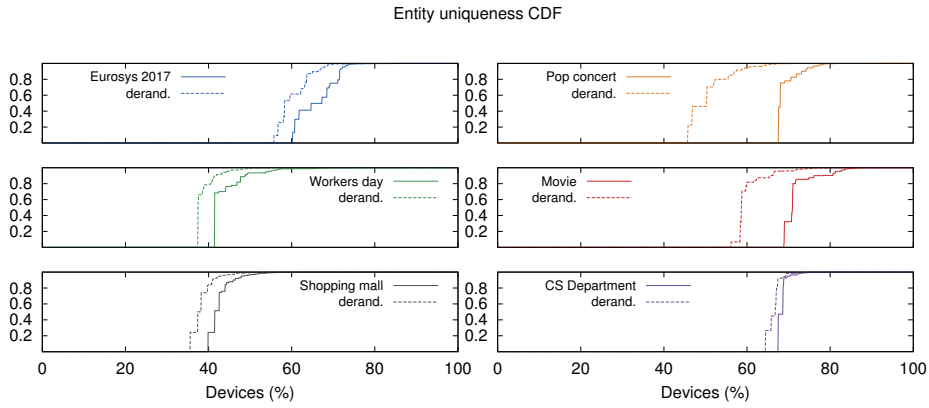

Fig. 2. Distribution of all entities' uniqueness values. Y-axis represents uniqueness values. Solid line represents the raw data set, and dashed line the same set with effects of MAC address randomization removed.

frame captured should contain a piece of information from someones previous network associations. Further textual analysis based on the SSID classification presented in section 3.3 would reveal the significance of a single leaked SSID name.

5 Fixing the leak

Active network discovery in 802.11 works through a protocol where the client device sends out probe requests in hope for nearby access points to receive them. These probe requests can be either broadcast or directed. A broadcast probe is not addressed to any access point in particular, but may – as the name strongly indicates – be received by any access point withing range. A directed probe on the other hand is addressed to a particular *service set* (ESS/BSS). Due to the design of service sets, discovery of networks should be done based on the *service set identifier* (SSID). As the client does not initially know any access point providing access to the service set, it prepares the frame with a broadcast *destination address* (DA) and specifies the SSID in a designated field inside the probe frame header. Because the frame has a broadcast link layer destination, it is by design receivable by anyone. Since management frames, including probe requests, are unencrypted anyone can read the content. This is how PNLs leak to the external parties.

Passive network discovery is an alternative way of finding surrounding wireless networks. In this case it is required that the access point advertises itself by transmitting beacons regularly. Beacons are broadcast so that any potential client can receive the beacons. The beaconing interval can in many cases be user configured, but our measurement shows that most access points send beacons with a 100 to 120 millisecond interval, which on a wide range of access points

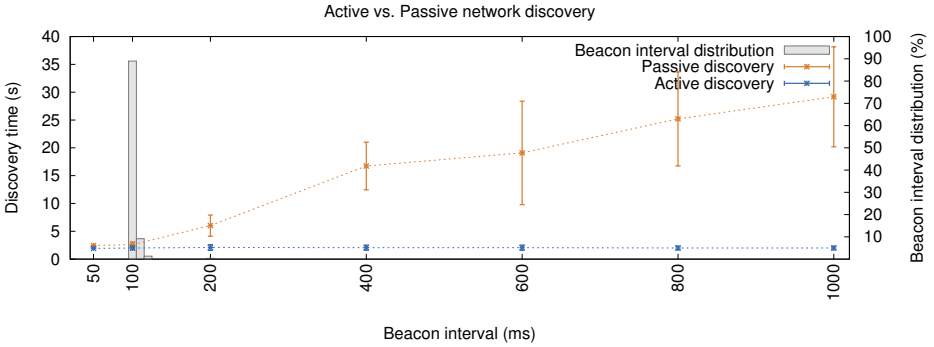

Fig. 3. Comparison of discovery times (left side Y-axis) with both active and passive network discovery. Beacon interval distribution (right side Y-axis) represents a sample of roughly 600 access points.

seems to be the default setting. Once a client receives a beacon from an access point it is willing to associate with, it already knows the MAC address of the AP since it was the set as the *source address* (SA) in the beacon header. Passive network discovery by design never broadcasts SSIDs from clients, and thus is safe from potential privacy violating information leakage.

A downside with passive network discovery is that it is not as fast as active scanning. Since access points can be configured on any channel, clients have to listen to all potential channels for a predetermined time and wait for beacons. If a client does not wait long enough it could miss a beacon. Another drawback of passive network discovery is that it does not work with hidden access points. However, hiding a network does not add any security or privacy, and is thus considered to be a bad practice, or even a IEEE 802.11 protocol violation [10].

For this paper we measured association times with both an active a passive configuration. We used two laptops, where one was configured as an access point and the other one as a connecting client. Both hosts ran Debian Linux and used `wpa_supplicant`[2] and `hostapd`[3] on the AP side, which are the most widely used IEEE 802.11 software backend components. Passive scanning is a built in feature on recent versions of `wpa_supplicant`, and thus does not require anything more than a simple configuration parameter.

We measured the association times with both active and passive network discovery, and also with different beaconing intervals configured, namely 50, 100, 200, 400, 600, 800, and 1000 milliseconds. Figure 3 shows the difference between association times between active and passive network discovery. The figure also shows the distribution of beacon interval times recorded around our university campus area and a nearby shopping mall and residental area. The capture file contains beacons from over 600 different access points.

[2] https://w1.fi/wpa_supplicant/
[3] https://w1.fi/hostapd/

Our measurements show that passive network discovery takes only 0.6 seconds longer in 98% of the cases compared to active scanning. Figure 3 shows that the difference between active and passive scanning is neglible at beacon interval values that the vast majority of access points are using. Since network discovery is an infrequent procedure and may happen in the background, we argue that this penalty has no impact on user experience.

6 Related work

Probing characteristics have been studied since potential tracking and privacy concerns emerged withing the wireless community. Barbera et al. [2] did large scale measurements in public locations about probing routines in mobile devices and looked for social-network properties in the data.

Freudiger et al. [3] studied how much probes modern cellphones transmit. Their work shows that the rate at which probes are transmitted highly depends on the brand and model of the device. They conclude that a frightening amount of frames with potentially sensitive information can be collected efficiently with different antenna and wireless interface configurations. In our recent work [13] we explained and evaluated an multi channel scanning device that has full temporal coverage each channel.

Since probing makes it possible to track users, MAC address randomization has been proposed as a solution to preserve privacy. Gruteser et al. [5] proposed the use of *disposable interface identifiers*, i.e. random MAC addresses to obfuscate entities. Singelée et al. [11] proposes a more cryptographic approach in random identifiers for WPAN networks, but can be applied in the same manner to IEEE 802.11 networks aswell.

Randomization has been adopted by major operating systems (Android v. 6.0, iOS v. 8, Windows v. 10 and Linux kernel v. 3.18), and its implementation differs slightly between the platforms. A study by Vanhoef et al. [12] analyzes different implementations of MAC address randomization. Their major contribution is about reversing the effect of randomization through fingerprinting different parts of frames. Work by Martin et al. [7] claims 100% success ratio in reversing randomization by looking at low level control frame handling. They exploit an existing design flaw in current wireless chipsets and present a breakdown of MAC address randomization techniques different platforms use. Matte et al. [8] presents an alternative approach to reverse randomization by looking at the timing between transmitted frames. Their approach claims a 75% success ratio by only looking at the timing of received frames.

Privacy issues with the current IEEE 802.11 network discovery protocol has been addressed earlier. Lindqvist et al. [6] proposed a privacy preserving access point discovery protocol already back in 2009. Their solution builds on top of the existing discovery protocol. It is a key exchange protocol where the nonce-based keys are piggybacked inside probe request and response frames. The protocol requires support from both parties, and to our knowledge has not been deployed outside their lab.

7 Conclusion

The network discovery protocol specified by the IEEE 802.11 standard has by design a feature that can potentially leak sensitive information. Directed probe requests carry names of SSIDs that the device has previously been connected to. Despite the fact that they are directed, they are transmitted with a broadcast destination so that any access point can receive them. A privacy threat emerges when an eavesdropper successfully collects the whole *preferred networks list* (PNL). In this paper we presented a way to quantify the information leak that is present in the current network discovery protocol. We introduced a metric called *uniqueness* in section 3.4. It is calculated based on the PNL leaked from a mobile user. We collected a data set, consisting is six separate subsets, which shows that roughly 30-40% of collected probes carry an SSID name. Also around 30% of seen entities broadcasted their PNL in the air. We calculated the uniqueness value for all entities found in our data set. Detailed information of the data set could be seen in Table 1.

MAC address randomization is a technique intended to reduce the traceability of devices. A device employing it uses disposable MAC addresses as the sender address in probe request frames. Several studies have shown that it is not as effective as expected and due to e.g. design flaws the effect of randomization can be reversed in various ways. In this paper we implemented our own version of MAC address de-randomization based on techniques presented in by others [12, 7]. We de-randomized our data set and calculated the uniqueness values for the data set again, and compared the uniqueness distribution to our earlier results.

In order to prevent the information leak through PNLs, broadcast probe requests with SSIDs should not be transmitted. Passive network discovery works without actively sending probes. It works by listening to beacons sent periodically by access points. Passive network discovery is slower than active, but our evaluation in section 5 indicates that in the majority of cases the penalty is not significant.

References

1. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Std 802.11-2012 (Revision of IEEE Std 802.11-2007) pp. 1–2793 (March 2012)
2. Barbera, M.V., Epasto, A., Mei, A., Perta, V.C., Stefa, J.: Signals from the crowd: Uncovering social relationships through smartphone probes. In: Proceedings of the 2013 Conference on Internet Measurement Conference. pp. 265–276. IMC '13, ACM, New York, NY, USA (2013). https://doi.org/10.1145/2504730.2504742, http://doi.acm.org/10.1145/2504730.2504742
3. Freudiger, J.: 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. pp. 8:1–8:6. WiSec '15, ACM, New York, NY, USA (2015). https://doi.org/10.1145/2766498.2766517, http://doi.acm.org/10.1145/2766498.2766517

4. Greenstein, B., McCoy, D., Pang, J., Kohno, T., Seshan, S., Wetherall, D.: Improving wireless privacy with an identifier-free link layer protocol. In: Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services. pp. 40–53. MobiSys '08, ACM, New York, NY, USA (2008). https://doi.org/10.1145/1378600.1378607, http://doi.acm.org/10.1145/1378600.1378607
5. Gruteser, M., Grunwald, D.: Enhancing location privacy in wireless lan through disposable interface identifiers: A quantitative analysis. In: Proceedings of the 1st ACM International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots. pp. 46–55. WMASH '03, ACM, New York, NY, USA (2003). https://doi.org/10.1145/941326.941334, http://doi.acm.org/10.1145/941326.941334
6. Lindqvist, J., Aura, T., Danezis, G., Koponen, T., Myllyniemi, A., Mäki, J., Roe, M.: Privacy-preserving 802.11 access-point discovery. In: Proceedings of the Second ACM Conference on Wireless Network Security. pp. 123–130. WiSec '09, ACM, New York, NY, USA (2009). https://doi.org/10.1145/1514274.1514293, http://doi.acm.org/10.1145/1514274.1514293
7. Martin, J., Mayberry, T., Donahue, C., Foppe, L., Brown, L., Riggins, C., Rye, E.C., Brown, D.: A study of mac address randomization in mobile devices and when it fails. arXiv preprint arXiv:1703.02874 (2017)
8. Matte, C., Cunche, M., Rousseau, F., Vanhoef, M.: Defeating mac address randomization through timing attacks. In: Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks. pp. 15–20. WiSec '16, ACM, New York, NY, USA (2016). https://doi.org/10.1145/2939918.2939930, http://doi.acm.org/10.1145/2939918.2939930
9. Pang, J., Greenstein, B., Gummadi, R., Seshan, S., Wetherall, D.: 802.11 user fingerprinting. In: Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking. pp. 99–110. MobiCom '07, ACM, New York, NY, USA (2007). https://doi.org/10.1145/1287853.1287866, http://doi.acm.org/10.1145/1287853.1287866
10. Riley, S.: Myth vs. reality: Wireless ssids (October 2007)
11. Singelée, D., Preneel, B.: Location privacy in wireless personal area networks. In: Proceedings of the 5th ACM Workshop on Wireless Security. pp. 11–18. WiSe '06, ACM, New York, NY, USA (2006). https://doi.org/10.1145/1161289.1161292, http://doi.acm.org/10.1145/1161289.1161292
12. Vanhoef, M., Matte, C., Cunche, M., Cardoso, L.S., Piessens, F.: 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. pp. 413–424. ASIA CCS '16, ACM, New York, NY, USA (2016). https://doi.org/10.1145/2897845.2897883, http://doi.acm.org/10.1145/2897845.2897883
13. Waltari, O., Kangasharju, J.: The wireless shark: Identifying wifi devices based on probe fingerprints. In: Proceedings of the First Workshop on Mobile Data. pp. 1–6. MobiData '16, ACM, New York, NY, USA (2016). https://doi.org/10.1145/2935755.2935757, http://doi.acm.org/10.1145/2935755.2935757