I know your MAC address: targeted tracking of individual using Wi-Fi

Mathieu Cunche

I know your MAC Address: *Targeted tracking of individual using Wi-Fi*

Mathieu Cunche

Université de Lyon, INRIA,
INSA-Lyon, CITI-INRIA, F-69621, Villeurbanne, France
mathieu.cunche@inria.fr

Abstract. This work is about wireless communications technologies embedded in portable devices, namely Wi-Fi, Bluetooth and GSM. Focusing on Wi-Fi, we study the privacy issues and potential missuses that can affect the owners of wireless-enabled portable devices. Wi-Fi enable-devices periodically broadcast in plain-text their unique identifier along with other sensitive information. As a consequence, their owners are vulnerable to a range of privacy breach such as the tracking of their movement and inference of private information [11, 8]. As serious as those information leakage can be, linking a device with an individual and its real world identity is not a straightforward task. Focusing on this problem, we present a set of attacks that allow an attacker to link a Wi-Fi device to its owner identity. We present two methods that, given an individual of interest, allow identifying the MAC address of its Wi-Fi enabled portable device. Those methods do not require a physical access to the device and can be performed remotely, reducing the risks of being noticed. Finally we present scenarios in which the knowledge of an individual MAC address could be used for mischief.

1 Introduction

Wi-Fi technology is the main solution for medium range communications, and is embedded in more and more *smart* objects. In particular most smart-phones possess a Wi-Fi network interface that allows getting a cheapest and fastest access to the Internet than GSM technologies.

Wi-Fi is today featuring robust encryption/authentication mechanisms that ensure that the data is securely transmitted of the wireless channel. However, recent works have shown that those Wi-Fi enabled devices are a threat to their owners' privacy. For instance, the name of the previously accessed network can be found unencrypted in some management frames and can be used to infer various private information [11] on the owners such as social links [8].

In addition, if the payload of some Wi-Fi frames can be encrypted, the header is always transmitted in clear. This means, that the MAC address of the devices, a unique identifier, can be collected and used to uniquely identify the device's owner. Emission of Wi-Fi frames is not limited to the time when the device is connected to a Wi-Fi network. In fact, due to an active service discovery

enabled on most devices, Wi-Fi interfaces are periodically broadcasting frames containing their MAC address. As a result, a device with a Wi-Fi interface turned-on, act as an actual wireless beacon by periodically advertising in clear a unique identifier. This is also true for other technologies such as GSM and Bluetooth that periodically send in clear a unique identifier (MAC address for Bluetooth and TMSI for the GSM). Therefore, part of the methods presented in this work could be extended to these other RF technologies.

Thanks to those wireless beacons that we are carrying in our pockets, Radio-Frequency tracking (RF-tracking) is now possible. A number of researchers and hackers have started to demonstrate such system to gather mobility data-sets or to increase privacy awareness on this topic [12, 13]. Beyond those scientific works and demonstration, wide scale commercial application of RF-tracking are already up and running. For example, RF-tracking is used in traffic monitoring application, for which it provides traffic information such as point-to-point travel time and traffic intensity [2]. It is also used to monitor people activities within retail stores and shopping centres [3]. Indeed deployed in these locations, RF-tracking systems collect information about customers' flows and their shopping habits.

If RF-tracking allows tracking of individuals based on a unique identifier, the link to a real identity is not directly available. In this work we are considering the problem of finding the link between a MAC address broadcasted by a device, and the identity of its owner. More particularly, given a person of interest, the *target*, we are willing to obtain the MAC address of its Wi-Fi enabled portable device.

To achieve this goal we propose to use a combination of wireless technology hacking along with physical and social actions. Furthermore, in order to ensure the practicality of our proposal, we aim at designing methods having the following properties:

- *Accuracy*: the obtained MAC address must belong to the target with a high probability;
- *Stealthiness*: the attack must remain unnoticed by the target.

The contributions of this work are the following. We present two methods achieving the aforementioned goals, i.e. establishing, with a high probability and stealthily, the match between an individual's identity and a wireless devices unique MAC address. The first method named *Wi-Fi AP Replay attack* impersonates the networks to which the target has been previously connected to, in order to identify its MAC address. The second attack called *Stalker attack* consists in monitoring the Wi-Fi channel while following the target in a public space, and to latter identify its MAC address by analysing the captured trace. In addition we present a set of scenarios in which the knowledge of an individual's MAC address could be useful to breach privacy or to do mischief.

The paper is organized as follows. Section 2 presents an introduction to the Wi-Fi technology and associated hacking tools. A preliminary investigation on information transmitted in plain text by Wi-Fi devices is done in Section 3. Then two methods to identify the MAC address of an individual are presented

in Section 4 and 5 present two attacks. Section 6 presents a number of malicious applications and Section 8 concludes.

This article is an extended and revised version of the workpresented at the 2nd International Symposium on Research in Grey-Hat Hacking (GreHack'13) [7].

2 Background and problem statement

2.1 Wi-Fi technology

A typical Wi-Fi network in *infrastructure mode* is composed of an access point (AP) to which a set of stations (device equipped with a Wi-Fi interface) are connected. Wi-Fi technology is based on the 802.11 protocols family. Wi-Fi packets are called frames and can be divided in 2 categories: data frames on one side and management and control frames on the other side. Data frames are in charge of carrying the actual data traffic while management and control frames serve various purposes such as association, authentication and service discovery. If the payload of data packets can be encrypted when security mechanisms are enabled (WEP, WPA, etc.), frame header are always in clear, leaving all the corresponding information available to eavesdroppers.

Figure 1 presents the structure of a 802.11 frame: a 30 bytes long header and a payload followed by a 4 bytes checksum. Within the header the Address fields contains MAC addresses: Address 1 designates the source and Address 2 the destination.

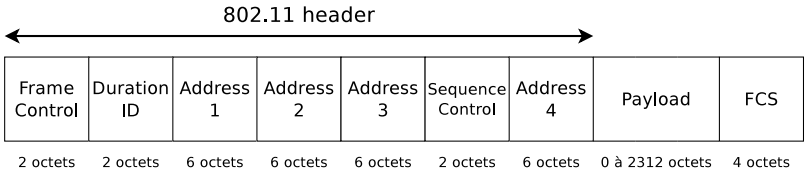


Fig. 1: Structure of a 802.11 frame.

MAC address Amongst the multiple information contained in frame headers, there are MAC addresses of the emitting device and the destination. A MAC address is a 48 bits number used to uniquely identify a network interface. MAC addresses are attributed to interface vendors by block of 2^{24}. As a result the 24 leftmost bits of a MAC address can be used to identify the interface's manufacturer.

In any frame, the source address field of the header contain the MAC address of the emitting interface. As noted before, 802.11 headers are never encrypted, therefore the source MAC address is available in plaintext in all the frames emitted by a device. This would be of limited importance if Wi-Fi devices where emitting frames only when connected to a network, but in fact, because of service discovery mechanisms, they transmit frames even when they are not connected.

Configured Network List Most operating systems are storing a list of wireless networks to which the device have been connected to. This list is called the *Configured Network List* (CNL) and contains information such as the network's SSID and its security features. On Windows operating systems, the CNL is stored in the registry at the following location:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\
    CurrentVersion\NetworkList\Profiles
```

While on GNU/Linux operating systems the CNL is stored in the Network Manager at the following location:

```
org.freedesktop.NetworkManagerSettings.Connection
```

Service discovery The Wi-Fi technology features a service discovery mechanism, which allows stations to discover the access points in range. Two variants of service discovery are co-existing. In the first one, called *passive service discovery*, APs are periodically advertising their presence by broadcasting *beacon* frames containing various information (SSID, security features), while stations passively listen to those beacons to discover APs in range. In the second, called the *active service discovery*, the station plays an active role by periodically probing the neighbourhood with *probe request* frames to which AP respond with *probe response* frames.

A probe request frame includes an SSID field to designate the wireless network sought by the device. A Wi-Fi device probes for network to which it has been previously connected by circling through the CNL. By doing so, devices are actually broadcasting in plaintext their connection history. In response to obvious privacy issues [11, 8], a new convention have been adopted: stations can send probe requests with an empty SSID field; and in return AP must respond to them with a *probe response* even if the SSID field do not match their own SSID. The resulting probe requests are called *broadcast probe requests*. As a consequence, devices are not revealing their connection history anymore, but are still periodically broadcasting their MAC address in clear.

Figure 2 presents the header of a 802.11 probe request containing, in plaintext, the source MAC address (`00:23:14:a7:e0:dc`) and the destination MAC address (`ff:ff:ff:ff:ff:ff`).

```
IEEE 802.11 Probe Request, Flags: ........
    Type/Subtype: Probe Request (0x04)
    Frame Control: 0x0040 (Normal)
        Version: 0
        Type: Management frame (0)
        Subtype: 4
        Flags: 0x0
            .... ..00 = DS status: Not leaving DS or network is operating
                 in AD-HOC mode (To DS: 0 From DS: 0) (0x00)
            .... .0.. = More Fragments: This is the last fragment
            .... 0... = Retry: Frame is not being retransmitted
            ...0 .... = PWR MGT: STA will stay up
            ..0. .... = More Data: No data buffered
            .0.. .... = Protected flag: Data is not protected
            0... .... = Order flag: Not strictly ordered
    Duration: 0
    Destination address: Broadcast (ff:ff:ff:ff:ff:ff)
    Source address: IntelCor_a7:e0:dc (00:23:14:a7:e0:dc)
    BSS Id: Broadcast (ff:ff:ff:ff:ff:ff)
    Fragment number: 0
    Sequence number: 0
```

Fig. 2: Example of a 802.11 probe request's header captured by `tshark`.

2.2 Wi-Fi tracking

As noted before, the MAC address of a wireless device constitutes an excellent unique identifier to track its owner. In fact MAC address of wireless devices are collected and stored by several systems.

A first example is the network infrastructure that is often storing information on the device that are connecting to it. For instance logs of wireless routers include the MAC address of all the devices that have been connected. Those logs contains events related to management aspect of the wireless network (association, authentication, disconnection, etc.) and each event associates a mac address with a timestamp.

The second example is the Radio-Frequency tracking systems [13] that are specifically designed to track the movement of individuals thanks to the wireless devices that they are wearing. Those systems are based on a set of sensors collecting wireless signals that triangulate and track over time the movement of individuals. Those systems are deployed in areas such as shopping centres, museum, roads where they provide valuable information on mobility patterns and shopping habits[1].

We can also add tracking systems deployed by criminals, spies, stalkers or any curious person. Indeed as the next section will show, collecting radio signal emitted by personal wireless devices does not require advanced skills nor

[1] Examples of commercial RF-tracking systems: Navizon ITS (`http://www.navizon.com/product-navizon-indoor-triangulation-system`), Euclyd-Analytics (`http://euclidanalytics.com/`)

expensive tools. Therefore, tracking systems can be easily deployed using cheap hardware and open-source software such as Snoopy [9].

2.3 Wi-Fi hacking tools

In wireless communications protocols, the security is of prime importance. As a consequence a number of penetration tools have been developed to test the security of wireless networks. Focusing on the Wi-Fi technology, an auditing software suite called *aircrack-ng* [1] is freely available and extensively documented. Along with this software, drivers supporting raw traffic monitoring have been developed. Thus a compatible wireless interface combined with the associated monitoring mode driver and *aircrack-ng* suite is sufficient to capture Wi-Fi traffic and perform a wide range of attacks.

In this work we used the *aircrak-ng* suite along with the network protocol analyser *tshark* [3] (the command line version of *wireshark*). Using appropriate filters we have design a tool capturing the MAC address of the Wi-Fi devices along with other private information such as SSIDs probed by those devices. Figure 3 presents a screenshot of an anonymized[2] capture displaying for each device in range the device manufacturer, the device mac address, the received signal strength along with the number and the list of associated SSIDs.

```
total number of devices : 32
Apple, Inc.     | 40:a6:d9:ee:__:__ | -28 dB | 1 | ''
SAMSUNG ELECTRO| 20:64:32:c1:__:__ | -45 dB | 1 | ''
Murata Manufact| 00:37:6d:ea:__:__ | -88 dB | 1 | ''
Apple, Inc      | 00:26:b0:7d:__:__ | -43 dB | 1 | ''
RIM             | a0:6c:ec:2a:__:__ | -67 dB | 3 | 'SSID_1','SSID_2'
Apple Inc       | 70:56:81:bb:__:__ | -58 dB | 1 | ''
Agere Systems   | 00:02:2d:bf:__:__ | -49 dB | 1 | 'SSID_4'
Apple, Inc      | f8:1e:df:d9:__:__ | -50 dB | 1 | 'SSID_5'
Murata Manufact| 00:37:6d:42:__:__ | -89 dB | 1 | ''
Intel Corporate| 00:24:d7:59:__:__ | -57 dB | 1 | 'SSID_6'
LG Electronics  | 10:68:3f:4e:__:__ | -58 dB | 1 | ''
Apple, Inc.     | 24:ab:81:8d:__:__ | -82 dB | 1 | ''
Apple, Inc.     | 58:55:ca:f3:__:__ | -91 dB | 1 | ''
Intel Corporate| 00:21:6a:7f:__:__ | -76 dB | 1 | ''
...
```

Fig. 3: Example of the information obtained from captured Wi-Fi probe requests.

It is important to note that, although using software belonging to a suite designed for security auditing; passively collecting this information does not require breaking any encryption or security mechanism. In fact this traffic is transmitted in plaintext; in particular the MAC address is part of the 802.11

[2] The tail of the mac address and SSIDs have been replaced by '_'.

header that is never encrypted. In addition, a significant part of this traffic (probe requests) specify a broadcast destination address (`ff:ff:ff:ff:ff:ff`) and is therefore by definition sent to all device in range.

2.4 Problem statement

As we have just seen, wireless devices, can reveal a lot of information about their owners. They can be used to track individuals' whereabouts and movements as well as other, potentially private, information. Emergence of wireless monitoring tools has made the collection of those signals an easy task. At the same time systems exploiting this information to track individuals for optimisation or profiling purposes have emerged.

In those systems it is assumed that each device is associated with an individual, and the unique identifier of the device (the MAC address) is used to identify the corresponding individual. However, the real identity of the device's owner is never stored in the system as this information is not directly available. In fact, the MAC address act as a *pseudonym* for the tracked individual. As a consequence, one could think that the impact on privacy is limited since, as private as the collected information can be, it is never matched with the real identity of the individual.

We focus on the problem of finding the link between real identities and the MAC address broadcasted by a wireless device. If available, this information could be combined with the data stored by physical tracking systems to gain knowledge on private information. Or it could simply be used to physically track an individual, by monitoring the radio signal emitted by its device that is literally acting as a *portable beacon*.

3 Preliminary investigation

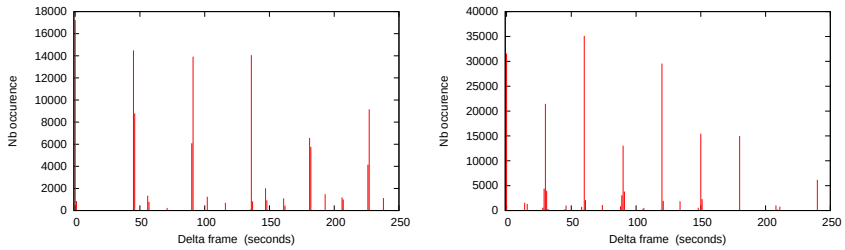
As illustrated before, the MAC address of a device constitute an ideal personal unique identifier. In this section we focus on the feasibility of collecting those unique identifiers by studying the frequency at which they are emitted as well as the maximum range at which they can be captured. Those characteristics are of prime importance as they determine the maximum distance between the attacker and the probability of success of our attacks.

3.1 Frequency of the unique identifier transmission

The emission frequency has been studied by monitoring a Wi-Fi channel and computing the arrival time delta, i.e. the difference between the arrival times of all the received probe request frames. The result for two device type (an Apple and a Samsung smartphone[3]) is presented in on Figure 4. For both devices,

[3] The information about the device manufacturer has been obtained through the OUI list that link MAC address prefixes to vendor names (`http://standards.ieee.org/develop/regauth/oui/oui.txt`).

peaks can be observed at a regular interval. For the Apple device the peaks can be observed around 0, 45, 90 and all other multiple of 45 seconds while for the Samsung device, peaks are appears at 0, 30, 60 and all other multiple of 30 seconds. This indicates that those devices regularly broadcast probe request frames and that the typical period is 45 seconds for the Apple device and 30 for the Samsung device. Those results cannot be extended to all Wi-Fi devices, but they give an idea on the period at which probe requests are emitted by a typical Wi-Fi device.



(a) Apple device (b) Samsung device

Fig. 4: Delta arrival time of probe requests frames

3.2 Transmission range

The range of a typical Wi-Fi transmission depends on the characteristics of the emitter and the receiver. For a part of the attacks that will be presented in this work a partial reception of the Wi-Fi frame is sufficient. We have measured in an outdoor environment that the frames emitted by two common smartphones, the *Samsung Galaxy SII* and the *Apple iPhone 4S*, can be received by the embedded Wi-Fi interface[4] of our DELL latitude E4310 at a distance of more than 100 meters for the *Samsung* device and more than 30 meters for the *Apple* device. This ensures that frames emitted by a Wi-Fi device can be collected at a quite long range.

4 Beacon replay attack

The idea of this attack is to identify a device through one or several *personally identifying wireless networks* (PIWN) to which it has been connected. Indeed, a set of PIWN can form a unique identifier than can be linked to an individual. Examples of PIWN are personal home network, work network. By guessing the

[4] Listed as `Corporation Centrino Ultimate-N 6300` by our GNU/Linux operating system.

networks to which a device has been connected, we hope to trigger a reaction from the device that will reveal its link with the targeted individual.

We propose to identify the association between a device and a network by leveraging the service discovery mechanism of the 802.11 protocol. More particularly by impersonating some Wi-Fi network, we can trigger a reaction from the device that has been associated to this network [17]. Impersonating a Wi-Fi network can be done by replaying its beacon frames. Indeed, when receiving a beacon frame identifying a known network (network in the Configured Network List of the device), the device will try to connect and, as a consequence, reveal its MAC address.

Some Wi-Fi devices are publicly revealing their wireless network association through a specific active service discovery mechanism. In this mode, devices are sending probe requests containing the SSID of the wireless network in their Configured Network List (see Section 2). For these devices, impersonation of a wireless network by replaying beacons may not be required as passive monitoring of the probe request frames could be enough to identify the association between a device and a wireless network.

The efficiency of our method relies on the ability of getting enough information through the PIWN. Gaining knowledge of enough PIWN to uniquely identify an individual can be a challenging task. We propose to combine this information with spatial information about the user. Let's consider an individual I, H its home address, Let N_H be the protected wireless network visible at address H. Then by replaying the beacons corresponding to N_H in a location different from H such as A's workplace, then only A's device will respond to those beacons. We note that it is preferable to focus on a protected wireless network because they are usually associated to a much smaller group of users than open networks.

As shown by Golle and Partridge in [10] the `Home`/`Work` location pair can be a very strong pseudo-identifier, i.e. the probability that two persons working at a given place also live at the same place is very small. This specificity can be used to mount an efficient attack, assuming that the `Home` and `Work` location of the target are known. Figure 5 present an illustration of the beacon replay attack when using the home and work location of the target.

Implementation This attack requires collecting information about a number of wireless networks in the first phase and in a second phase to impersonate those networks and analyse the potential reaction of Wi-Fi devices. To this aim we have developed two tools that we have made available to the community [5]:

- The Wi-Fi network fingerprinter
- The Wi-Fi AP replayer

[5] Wi-Fi Stalking tools are available at `http://mathieu.cunche.free.fr/?page_id=438`

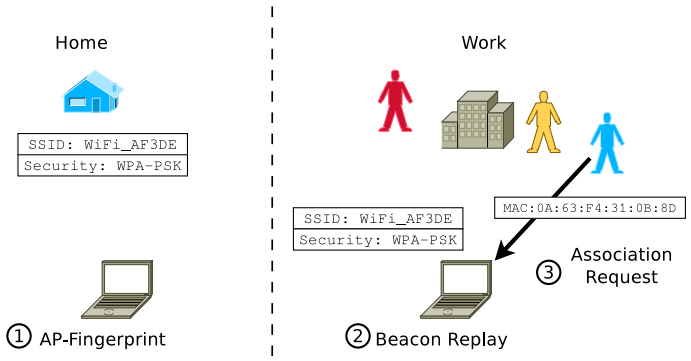


Fig. 5: Principle of the beacon replay attack using the home and work locations.

Wi-Fi fingerprinter The Wi-Fi network fingerprinter collects information about visible wireless networks and save it to a file for later use. The information collected consists in the network SSID and a simplified version of the network's security features (*open* or *secured*). This tool monitors the Wi-Fi channel for a fixed amount of time and, using *tshark*, selects only the beacon frames received from the surrounding access points. For each detected AP, it extracts the SSID and the security features and save the result to a file that will be later used by the Wi-Fi AP replayer.

Assuming that a monitoring Wi-Fi interface `mon0` have been created as follows:

```
$ sudo airmon-ng start wlan0
wlan0            Unknown          iwlwifi - [phy0]
                   (monitor mode enabled on mon0)
```

The WiFi AP fingerprinter should be used at a location familiar to the target, such as its home, using the following command:

```
$ ./WiFi_AP_fingerprinter.sh APfingerprint.txt mon0
Capturing on mon0
result saved to APfingerprint.txt :
-------------------------
FBI_Surveillance-Van_02;0
allo;0
Freeboite_RoXoR;1
Frit_WiFi;0
HuitBox_1234;1
-------------------------
```

In this case 5 networks have been detected, three of them being open (security = 0) and two being secured (security = 1).

Wi-Fi AP replayer The second step of the attack is to impersonate the wireless networks that have been fingerprinted in the previous step. This is done using the Wi-Fi AP replayer tool that takes as input a file containing the characteristics of several Wi-Fi networks and replay them, before analysing the response of surrounding Wi-Fi devices. In a first time the replay is performed using *airbase-ng*, a tool from the *aircrack-ng* suite, whose purpose is to create Wi-Fi access points in order to attack Wi-Fi clients. In our case we do not use the attack features and are only motivated by triggering a reaction from surrounding Wi-Fi clients. For each Wi-Fi network in the configuration file, a Wi-Fi AP is created for a fixed amount of time and the traffic received by this AP is stored in a capture file.

Then in a second time the each capture file is analysed to extract the MAC address of the Wi-Fi clients that have attempted to connect to the corresponding fake AP. This information is summarized for all the APs and displayed for the user under the form of a list of client MAC address and corresponding SSIDs. This list should contain the MAC address of the targeted device.

The WiFi AP replayer script should be run in range of the target using the following command:

```
$ ./WiFi_AP_replayer.rb test1.txt mon0
Creating fake AP : "eduroam" (privacy=1)
Creating fake AP : "FBI_Surveillance-Van_02" (privacy=0)
Creating fake AP : "allo" (privacy=0)
Analyzing results ...
Displaying results ...
c8:bc:c8:__:__:__ "Freeboite_RoXoR"
```

Here the results indicates that one device have reacted to the the network "Freeboite_RoXoR". Therefore we can infer that this device belongs to the target.

Concerning the stealthiness aspect, the two phases of the attack must be considered. Obtaining the wireless AP fingerprint of the home location, can be done at any time of the day (it is fair to assume that the wireless AP are running 24/day) and only require to walk pass the house or the building where the person is living. Alternatively, one can rely on online database of Wi-Fi access points such as WiGle [2] to get the characteristics of Wi-Fi networks at a given location.

In the second phase, the rogue AP must be in range of the targeted devices, which mean within a couple of meters. During the second phase of the attack,

one can reduce the distance between the target and the fake AP in order to use the signal strength to narrow done the device. However this improvement in term of accuracy can reduce the stealthiness of the attack.

5 *Stalker* attack

In order to identify the device associated to a given individual, the ideal solution is to be isolated with this person. This means making sure that the distance between the target and the monitorer is small compared to the distance between the monitorer and other individuals. In a real world scenario, this configuration can be hard to achieve and can raise suspicion of the target, compromising the *stealthiness* requirement.

Another approach is to use a *set intersection attack* that consists in considering several distinct groups of individuals and by studying their intersection. The intersection of the individual should match with the intersection of the collected device identifiers. In the particular case where this intersection is reduced to one element, the device identifier of the *target* can be directly deduced.

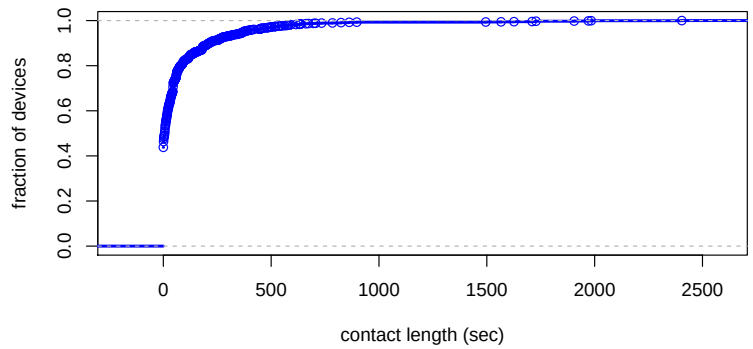
In practice clearly isolating and identifying a group of individual can be troublesome. In the following we consider a *continuous* alternative to this *discrete* method. Instead of considering fixed group of individual, we focus on a stream of individual. In most social places, the group of individuals within the monitorer covering area is a set of individual that is continuously changing. The idea is to make sure that the targeted individual stays in the monitorer covering area, while the rest of the set of the monitored individual is changing. The targeted individual will be uniquely identifiable once the set of monitored people have been totally renewed at the exception of *target*.

A simple way to maintain an individual in the monitored area, while the rest of the set changes through time, is simply to *stalk* the target by following him. Assuming that the target is walking in the street, the method consists in following this person at a reasonable distance (close enough to stay in reception range and not too close to avoid suspicion) with a monitoring device. The target will stay in the monitored area while the bystander will only stay in the monitored area for a short period of time.

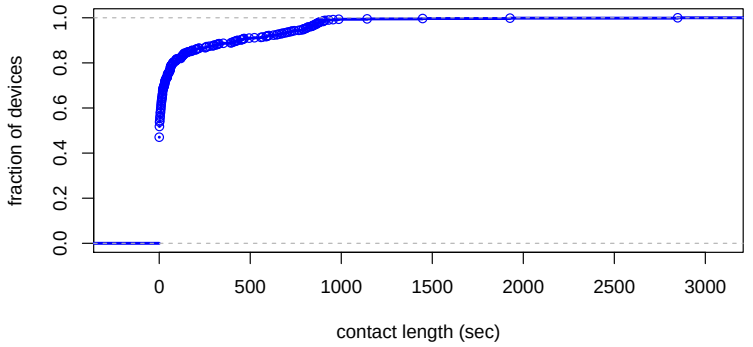
5.1 Empirical evaluation of Wi-Fi contact length

In order to demonstrate the previous assertion and to estimate the time the target should be monitored to be uniquely identifiable, we have performed a set of experiments in the wild. During this experiment, a monitorer equipped with a monitoring equipment have randomly moved across a large city during a period of two hours. The first capture contains contacts with 1644 devices while the second contain 460 devices.

Figures 6 presents the results for two capture traces as a cumulative distribution of the contact length. Overall, a large fraction of the contacts are short: for the capture 1, more than 80% of the contacts are below 500 seconds. In some



(a) Trace 1



(b) Trace 2

Fig. 6: Cumulative distribution of contact length in two captures.

rare cases the contact is rather long (up to 3000 seconds $\simeq$ 50 minutes). We can observe a slight difference in the shape of the curves between. In particular, the second capture exhibit a floor around 90% corresponding to a contact length of around 1000 seconds. This can be explained by the fact that the second capture included a train travel of around 15 minutes, meaning that the persons travelling in that train have stayed in range for at least 15 minutes.

The result of this experiments shows that during mobility within an urban area, radio contacts are short in most cases, while in some rare cases they can be as long as several tenth of minutes. In other words, a long and continuous contact is maintained with only a very small set of individuals.

5.2 Attack implementation

This method is in two steps. First during the stalking part, the monitorer follows the target while monitoring the wireless channel using the `Wi-Fi monitor`, and then the collected traces are analysed with `Stalking analyser` in order to identify the target's MAC address.

Overall the method works as follows:

1. Visually identify target

2. Monitor wireless communications and log device identifier (in the case of Wi-Fi the interface MAC address)
3. Follow the target in the street for N minutes, while keeping in transmission range
4. Search the log for a MAC@ that have been seen all along the N minutes

Wi-Fi monitor The Wi-Fi monitor captures the wireless frames and extracts their source MAC address. This tool uses the monitoring features of the aircrack-ng suite and collects the source MAC address using the `tshark` command line software.

The targeted individual should be followed, after having started the Wi-Fi monitor using the following command:

```
$ ./WiFi_monitor capture_file.txt mon0
Storing capture in  capture_file.txt
Capturing on mon0
```

Once the stalking done for a long enough period of time (see section 5.1) the capture file can be analysed as follows:

```
$ ./Analyze_capture capture_file.txt
Analyzing capture file: capture_file.txt
capture_file.txt
-----------------------------------------
 MAC addr            : Contact Length (sec)
[20:64:32:__:__:__] : 1023.129089
[1c:4b:d6:__:__:__] : 13.435345
[f8:1e:df:__:__:__] : 0.12231
...
[24:ab:81:__:__:__] : 0.0
```

The output of the command is a list of devices that have been in range for the longest period of time. Then we can conclude that the device that has stayed in range for the longest period of time (in the ideal case during the whole capture) belongs to the target. The devices that have the longest contact length are likely to belong to the target.

6 Applications

Knowing the MAC address of an individual can be used to collect sensitive information from systems storing MAC address along with other information. This is of the course the case of Radio-Frequency tracking systems (see section 2.2) or Wi-Fi routers that stores in their log time stamped connection events along with the MAC address of the device. The knowledge of the MAC address can also be used to launch targeted attack over a Wi-Fi, for instance by exploiting Wi-Fi drivers vulnerabilities [6]. In the following we present a number of scenario in which the knowledge of an individual MAC address can be used to various purposes including pranking, stalking or more dramatic ones.

Wi-Fi Booby Trap The knowledge of the MAC address of a given individual can be used to trigger an action when this person enters a given area. As previously presented in this work, by monitoring the Wi-Fi channel and by examining the source MAC address of the captured frames, it is possible to detect when a device comes in range. Furthermore by considering the signal strength of the received signals, it is possible to estimate the distance between the receiver and the device or even to estimate its position by triangulation if several receivers are deployed [5].

Using this information, one can think of multiple applications involving an action triggered when a targeted individual enters a location. This can be for example for pranking or for more harmful purpose like proximity weapons.

Tracking High-Profile individuals Wi-Fi tracking is the perfect tool for following high-profiles individuals (HPI), and knowing the MAC address of such person can be a great asset for *paparazzi* and journalists. We can envision the deployment of Wi-Fi sensors, like in the snoopy system [9], inside an area of interest, in order to acquire knowledge on the whereabouts of the targeted HPI. For instance we can know in advance by which exit the HPI will go when living a building.

Obtaining the mac address of a HPI can be a challenging task. One could use repeated encountered along with a *set-intersection* approach to narrow down the Wi-Fi identifier. HPI are often surrounded by a flock of people (manager, assistants, bodyguards), and by using the previous approach, we may end with multiple MAC address that may not belong to the HPI. However, this information could still be useful since, as we just said, those people are following the HPI and their position can have the same value as this of the HPI.

Who have you met today? GPS tracking either by planting a device on a vehicle or by installing an application on a phone, is a common method for tracking the movements of an individual. A similar approach could be considered with Wi-Fi technology. Instead of logging the movement of a person, it will monitor the interactions of a person with other identified individuals. Indeed planting a Wi-Fi monitoring device on a vehicle is totally feasible, and we are starting to see smartphones supporting Wi-Fi monitoring [6], i.e. the mode required to collect frames transmitted by surrounding devices.

7 Related works

Information leakage in the 802.11 technologies has been reviewed in several works. The nature of sensitive information leaked by 802.11 networks has been presented in [11]. In [15], et. al. go a step further by showing how SSIDs can be used to infer the locations where the device owner has travelled. In [8] the authors demonstrate how leaked SSIDs can be used to infer social links between the owners of Wi-Fi enabled devices.

The idea of replaying beacons in order to get a response from Wi-Fi devices has been initially proposed in [4]. By impersonating an access point, the attacker force the station to generates traffic that could be later used to recover the WEP key of the corresponding network.

Wi-Fi based tracking of individuals has recently received attention from the academic and the hacker community. Snoopy [9] and CreepyDOL [14] are two Wi-Fi tracking systems based on cheap and commercially available devices like the Raspberry-Pi.

Finally, Shue et. al. presented in [16] how wireless networks can be used to accurately find the physical address behind an IP address. The idea is to send the traffic to a host connected to the wireless network and to recognize the signature of this traffic by monitoring the wireless communications.

[6] Monitor mode for Broadcom WiFi Chipsets `http://bcmon.blogspot.fr/`

8 Conclusion

Wireless device can reveal a lot of information about their owner (tracking, connection history, etc.). However, the link between an individual and the device unique identifier, the MAC address, is rarely available. In this work we have presented two methods to find the MAC address belonging to a given individual link. To this aim, we have designed a set of tools, based on Wi-Fi monitoring techniques that we've made available for the community. The result of this work shows that any individual equipped with a Wi-Fi enable device, such as a smartphone, can be easily tracked in its daily life and that by putting enough effort it is possible to identify the association between a person and its device's MAC address. An interesting future work would be to consider the reverse problem: given a device identified by its MAC address, find the owner of this device.

References

1. Aircrack-ng, a set of tools for auditing wireless networks. http://www.aircrack-ng.org/.
2. WiGLE: Wireless Geographic Logging Engine. http://wigle.net/.
3. The wireshark network analyzer. http://www.wireshark.org/.
4. M. S. Ahmad and V. Ramachandran. Cafe latte with a free topping of cracked wep retrieving wep keys from road warriors. In *TOORCON9*, 2007.
5. P. Bahl and V.N. Padmanabhan. RADAR: an in-building RF-based user location and tracking system. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 775–784 vol.2, 2000.
6. Laurent Butti and Julien Tinnès. Discovering and exploiting 802.11 wireless driver vulnerabilities. *Journal in Computer Virology*, 4(1):25–37, 2008.
7. Mathieu Cunche. I know your MAC Address: Targeted tracking of individual using Wi-Fi. In *International Symposium on Research in Grey-Hat Hacking - GreHack*, Grenoble, France, November 2013.
8. Mathieu Cunche, Mohamed-Ali Kaafar, and Roksana Boreli. Linking wireless devices using information contained in Wi-Fi probe requests. *Pervasive and Mobile Computing*, (0):–, 2013.
9. Daniel Cuthbert and Glenn Wilkinson. Snoopy: Distributed tracking and profiling framework. In *44Con 2012*, 2012.
10. Philippe Golle and Kurt Partridge. On the anonymity of home/work location pairs. In *Proceedings of the 7th International Conference on Pervasive Computing*, Pervasive '09, pages 390–397, Berlin, Heidelberg, 2009. Springer-Verlag.
11. Ben Greenstein, Ramakrishna Gummadi, Jeffrey Pang, Mike Y. Chen, Tadayoshi Kohno, Srinivasan Seshan, and David Wetherall. Can Ferris Bueller still have his day off? protecting privacy in the wireless era. In *Proceedings of the 11th USENIX workshop on Hot topics in operating systems*, pages 10:1–10:6, Berkeley, CA, USA, 2007. USENIX Association.
12. Nathaniel Husted and Steven Myers. Mobile location tracking in metro areas: malnets and others. In *Proceedings of the 17th ACM conference on Computer and communications security*, CCS '10, pages 85–96, New York, NY, USA, 2010. ACM.

13. A. B. M. Musa and Jakob Eriksson. Tracking unmodified smartphones using Wi-Fi monitors. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, SenSys '12, pages 281–294, New York, NY, USA, 2012. ACM.
14. Brendan OConnor. CreepyDOL: Cheap, Distributed Stalking. In *BlackHat*, 2013.
15. Ian Rose and Matt Welsh. Mapping the urban wireless landscape with Argos. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, SenSys '10, pages 323–336, New York, NY, USA, 2010. ACM.
16. Craig A. Shue, Nathanael Paul, and Curtis R. Taylor. From an IP address to a street address: Using wireless signals to locate a target. In *7th USENIX Workshop on Offensive Technologies (WOOT '13)*, 2013.
17. Nils Ole Tippenhauer, Kasper Bonne Rasmussen, Christina Pöpper, and Srdjan Čapkun. Attacks on public wlan-based positioning systems. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*, MobiSys '09, pages 29–40, New York, NY, USA, 2009. ACM.