FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Personal Privacy in Wireless Emissions

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Wirtschaftsinformatik

eingereicht von

Anton Stütz, BSc.
Matrikelnummer 1129424

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Dr. Edgar Weippl
Mitwirkung: Dr. Martin Schmiedecker
Dipl.-Ing. Wilfried Mayer

Wien, 31. Jänner 2019

Anton Stütz

Edgar Weippl

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Personal Privacy in Wireless Emissions

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Business Informatics

by

Anton Stütz, BSc.
Registration Number 1129424

to the Faculty of Informatics

at the TU Wien

Advisor: Dr. Edgar Weippl
Assistance: Dr. Martin Schmiedecker
Dipl.-Ing. Wilfried Mayer

Vienna, 31st January, 2019

Anton Stütz

Edgar Weippl

Erklärung zur Verfassung der Arbeit

Anton Stütz, BSc.
Waldgasse 6b; 8680 Muerzzuschlag

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 31. Jänner 2019

Anton Stütz

Danksagung

Zuerst bedanke ich mich bei meinem Institut, meinem Professor und meinen Betreuern für die Ermöglichung der Arbeit und der langanhaltenden Unterstützung.

Im folgenden bedanke ich mich bei den beiden Partnerfirmen die mir die Durchführung dieser Masterarbeit und der zugehörigen Studie überhaupt ermöglicht haben. Außerdem bedanke ich mich auch noch sehr herzlich bei den Mitarbeitern die mich tatkräftig bei dem Aufstellen der Geräte unterstützt haben.

Darüber hinaus bedanke ich mich auch noch bei meiner Freundin, meinen Freunden und Familienmitgliedern für deren moralische Unterstützung, auf die ich immer zählen konnte.

Acknowledgements

I first want to thank the institute, my professor and my advisors for making this master thesis possible, their long-lasting support and also for their continuous input and suggested improvements.

I would like to thank the partner companies, which made it possible to conduct two data collections in a realistic environment in Vienna. Furthermore, I want to thank the employees who helped me during the deployment of the devices.

I also want to thank my girlfriend, friends and family for their continuous support.

Kurzfassung

Seit einigen Jahren sind Smartphones für viele Menschen aus dem alltäglichen Leben nicht mehr wegzudenken. Eine Studie von Eurostat zeigt, dass 74% der ÖsterreicherInnen zwischen 16 und 74 das Internet unterwegs am Handy benutzen. Jedoch wissen nur wenige welche Auswirkungen ihr Smartphone auf ihre Privatsphäre hat. Wenn ein Handy die WLAN-Funktion aktiviert hat und nicht mit einem Access Point verbunden ist, kann es trotzdem zu Datenübertragungen kommen. Diese Datenübertragungen oder Anfragen werden verwendet um in der Nähe liegende Access Points ausfindig zu machen. Sollte sich ein bekannter Access Point in der Nähe befinden, wird sich das Handy dadurch schnell und automatisch verbinden. In dieser Masterarbeit werden genau diese Datenübertragungen untersucht. Deshalb wurde eine umfangreiche reale Datensammlung an WLAN-Daten in einer Studie erstellt. Diese Datensammlung wurde im Anschluss ausgewertet und führte zu dem Schluss, dass es mit nicht randomisierten oder mit schlecht randomisierten MAC-Adressen sehr einfach ist Geräte über einen längeren Zeitraum zu verfolgen. Auch randomisierte MAC-Adressen in Kombination mit direkten Anfragen ermöglichen es Zusammenhänge zwischen verschiedenen MAC-Adressen herzustellen.

Abstract

In todays world, smartphones are a constant and almost indispensable companion in day-to-day life. A study from Eurostat shows that 74% of Austrian individuals aged 16 to 74 use Internet over mobile phones. Therefore, it is important to know why and how modern devices can effect the personal privacy through wireless emissions. Especially if the mobile phone is not connected to an Access Point and Wi-Fi is turned on. There can be wireless emissions because of a regular occurring network discovery to detect nearby Access Points. In this master thesis an extensive data collection of Wi-Fi frames is created and analyzed. The evaluation of the collected data led to the result that capturing of Probe Requests which are used for network discovery can be misused to provide a feasible and cheap tracking solution. Even if MAC randomization is used, devices could be tracked for multiple days. Also a combination of randomized MAC addresses which use directed Probe Requests was recorded. Directed Probe Requests can render MAC randomization completely useless.

Contents

CHAPTER 1

Introduction

Personal privacy is a valuable good. In the European Union new data protection laws give personal privacy a higher value. With the new General Data Protection Regulation (GDPR), which took effect on the 25th of May 2018, the fines for data privacy violations can be as high as 4% of the world wide turnover of a company [euG].

According to a study from Eurostat [eur] 74% of Austrian individuals aged 16 to 74 use Internet on mobile phones over mobile phone networks or Wi-Fi. Therefore, many modern phones are equipped with a Wi-Fi interface. This Wi-Fi interface makes it possible to connect to the Internet with fewer or no charges compared to mobile phone networks.

If a Wi-Fi interface is in an active state it regularly tries to connect to a network of their preferred network list. This can happen actively or passively. If the first option is used the device sends out so called Probe Requests and asks if known networks are nearby or even asks if specific networks are within reach. The requests which ask for a specific network are of special interest in this master thesis. Because they can not only reveal the current location of a device, but with a unique Service Set Identifier (SSID) it can also be possible to reveal the location of the work or home address. If the devices use a passive network discovery or passive scanning it only listens if an Access Point (AP) makes itself known.

According to Cunche et al. [CKB14] active network discovery has a lower discovery delay. The use of passive scanning can lead to high delays before a nearby AP is discovered. Therefore, many Wi-Fi interfaces use active scanning to maximize the connection time for an AP, especially if the device is moving across coverage areas, as it is likely with mobile phones.

However, this mentioned active way of discovering if a network is nearby can lead to wireless emissions which can reveal, among others, the devices globally unique MAC address and in special cases also the SSID it wants to connect to.

In this master thesis the focus lies on personal privacy in wireless emissions in the IEEE 802.11 Wi-Fi standard. Especially on the Probe Requests which are sent out from Wi-Fi devices, if the device is not connected to an AP. Considering that the data collection took place on the street and in a supermarket most client devices won't be connected to an AP.

Therefore, a data collection which lasted 30 and 14 days was realized to be able to analyze the frames sent out from client devices like smartphones. During this data collection over three million different MAC addresses have been captured.

The process of capturing Wi-Fi frames was completely passive. The capturing devices used, did not send out any frames to deceive any nearby devices to send out specific frames like in Vanhoef et al. [VMC+16] where fake hotspots were used. Consequently, the capturing devices could not be detected by analyzing the Radio Frequency (RF) of the 2.4Ghz spectrum.

Furthermore, the program which was used to capture the necessary data was configured to only capture management frames of the IEEE 802.11 standard, which have the type 0. Control frames and data frames were not captured, mainly because this would have unnecessarily increased the size of the log files. Moreover, data frames are usually encrypted and give little insight.

To minimize the on site setup the device's capturing program was automatically started after booting the system and a new log file was created. The on site setup was optimized to plug in the power cord only. Also if someone by accident or on purpose removed the device from the power supply for a short time, the device recovered automatically without user interaction.

To proof that there are wireless emissions that are privacy relevant, the data collection was then used to answer the following research questions:

- What is the entropy of MAC addresses collected in the field studies?
- What part of the collected data in the field studies can be used to identify the re-occurrence of users over multiple days?
- What part of the data, collected in the field studies, could be used to draw conclusions if a MAC address is randomized or not?

The master thesis is organized as follows. In the next chapter the necessary background knowledge is presented to create a better understanding for readers who are not familiar with this specific topic. In the following chapter the related work and recent research in this field is presented. Afterwards the chapter Methodology describes the initial setup and process of the research study. Followed by the description of analyzing and evaluating the collected dataset. The chapter Results is used to show the outcome of the evaluation. Followed by the discussion of the presented results. The final chapter is the conclusion.

CHAPTER 2

Background

In this chapter the necessary background knowledge is outlined for the subsequent chapters. A fundamental knowledge about the general networking standard and the wireless communication standard IEEE 802.11 is of importance. Therefore, in this chapter some basics about networking are explained. Furthermore, the differences between networking over ethernet and IEEE 802.11 is elaborated.

The IEEE 802.11 is a networking standard for implementation of Wireless Local Area Networks. In 802.11 one medium access control and several physical layer specifications are defined for wireless connectivity [80216]. So far the most widely used frequencies for Wi-Fi are 2.4 Gigahertz (Ghz) and 5 Ghz. 2.4 Ghz is an often used frequency which leads to interferences in crowded places, also microwaves cause interferences in this spectrum. To minimize interferences there are multiple channels available for consumer products in the 2.4 Ghz and 5 Ghz range. In Figure 2.1 the available channels on the 2.4Ghz spectrum can be seen, which can be used without a special license. The distance between the center frequency of every channel is 5 Mhz and the channel bandwidth is approximately 20 or 40Mhz. This leaves only four or two none overlapping channels [rtr].

In the following some necessary nomenclature is defined which will be used in the next chapters:

- A device or station is able to receive and transmit signals according to the IEEE 802.11 Wi-Fi standard [Gas]. For example a smartphone with a Wi-Fi interface.
- An access point is a device, which functions as bridge between the wireless medium and a distribution system. A distribution system can be a router connected over Ethernet. In general not only one but many Wi-Fi client devices can be connected to an access point. It is also possible that multiple access points are connected over the distribution system and form an areal bigger wireless network [Gas].

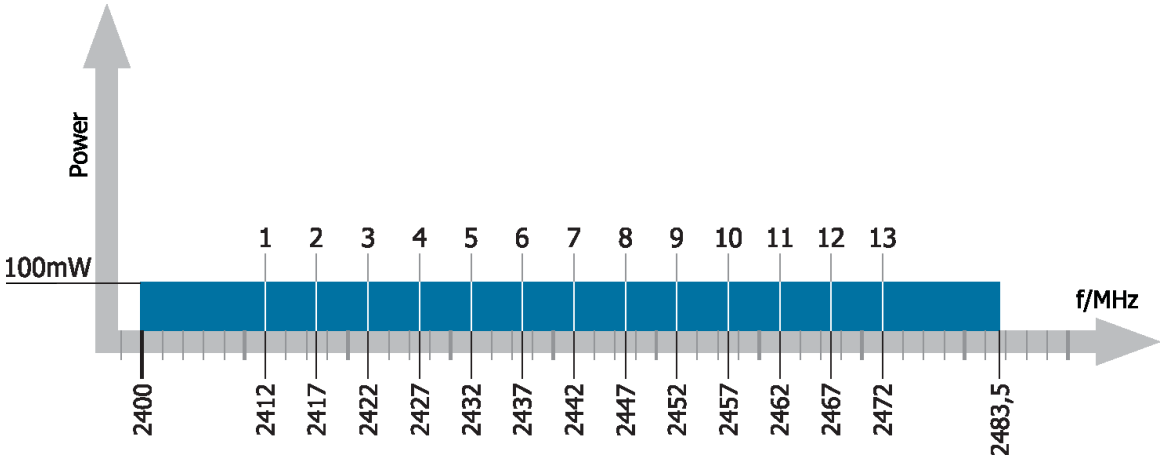

Figure 2.1: Available channels in Austria on the 2.4Ghz spectrum [rtr].

- A SSID also referred to as a network name is the name with which an access point advertises the availability of a specific network [Gas].
- In the evaluation of the dataset a device is considered a client device if at least one Probe Request was captured from it.
- A byte or octet consists of 8 bits, a nibble of 4 bits [HH83]. Therefore, a MAC address is divided by colons into octets and one hex number can be represented in one nibble.

Of fundamental knowledge is the conceptual ISO/OSI model [iso] which divides the communication of a network into seven different layers illustrated in Figure 2.1. Each layer has a header and a payload. The layer above is encapsulated in the payload of the layer underneath until reaching the physical layer from which the data is sent to the next endpoint.

When a device sends a package over the network, it first starts to create the data for layer 7 and then it wraps this layer into the payload of layer 6 and so forth, till it reaches the first or physical layer. Then the packages get send to the next endpoint. On the next endpoint this process starts again in reverse order. First the data link layer gets extracted, then the networking layer and so on. If this endpoint also happens to be the destination, the device will extract all layers and notifies the receiving application. But usually there are numerous endpoints or nodes between the sender and receiver. In general there are two types of nodes which are fundamental for networking: [iso]

- Routers
- Switches

Layer	Protocol data unit(PDU)
7. Application	Data
6. Presentation	Data
5. Session	Data
4. Transport	Segment(TCP) or Datagram(UDP)
3. Network	Packet
2. Data Link	Frame
1. Pyhsical	Bit

Table 2.1: ISO/OSI reference model [iso].

A switch considers only data up to the data link layer. It looks at all the bits it receives and assembles them to frames. A frame contains a sender and receiver MAC address. These MAC addresses are usually from adjacent systems. All other layers are just considered as payload. A router goes one layer further and also looks at the networking layer and is therefore IP aware and can route between different IP networks [Cea12].

Layer 7 or application layer interacts directly with an executable program and is therefore closest to the end user. It's also responsible for everything not defined in lower layers [HS01].

In general the presentation layer deals with the conversion of data structures into strings, e.g., serializing a Java class into a xml document. Furthermore, it is responsible for compression and encryption although often performed in other layers, e.g., IPSec. Examples for presentation layer protocols are American Standard Code for Information Interchange (ASCII) or Transport Layer Security (TLS) [lin].

The functionality of the session layer on the Internet is usually taken care of by the TCP protocol which is assigned to the third layer. In general the session layer is responsible for connection management and to establish and terminate logical sessions. The third down to the first layer are the most important layers for networking. Most of the network devices used only care for the first three layers, e.g., routers [Cea12].

The physical layer is responsible to transfer bits and bytes over a medium. This can be a wireless medium or a coaxial cable [Gas].

The data link layer links two directly connected nodes. It is responsible for error detection and correction which could occur in the physical layer during transmission. Examples for protocols which operate on this layer is Ethernet [IEE16] and the later further discussed 802.11 Wi-Fi protocol [IEE14].

With the network layer there is also the concept of ip addresses and routing introduced. In contrast to the data link layers MAC addresses, the ip source and destination address do not change in general. An exception is IPv4 NAT where the source address is changed.

In the IEEE 802.11 standard the data link layer from the ISO/OSI reference model is

further divided into two sublayers, the Logical Link Control (LLC) sublayer and MAC sublayer. Special algorithms take care of automatic channel switching and coordination for sending and receiving data between several clients and access points [Gas].

A client device can be in an associated or an unassociated state. Associated means a client device is connected to an access point. If a device is in an unassociated state it seeks to connect to an access point. In general there are two methods to find out which access point is in proximity:

- Passive scanning
- Active scanning

If a client uses passive scanning it listens on different Wi-Fi channels for Beacon frames from known access points. Another method is active scanning. This method brings Probe Requests into play. A client sends out Probe Requests on a regular basis. There are two types of Probe Requests: Broadcast and Directed Probe Requests (DPR). A DPR consists of the SSID the client wishes to connect to. If an access point receives a DPR and matches the SSID it will send a Probe Response. If an access point receives a broadcast Probe Request it will send a Probe Response also if the client does not know the access point. A broadcast Probe Request (PR) is better for privacy but it leads to more overall traffic, since every access point in range will answer.

Usually these two methods are combined, meaning that an access point sends out Beacons and a client device sends out Probe Requests. An exception is if the broadcasting of the SSID(Beacons) is disabled for the access point. In this case no Beacons are sent out. This option is often referred in the access point configuration as hide SSID. If the SSID is hidden by the access point and therefore the access point does not advertise himself it is only possible to detect such an AP with active scanning.

The general frame format used in IEEE 802.11 is shown in Figure 2.2. The main difference to a ethernet frame is that the wireless data link layer has four address fields, but not all four are used for every frame type.

A IEEE 802.11 frame starts with a two byte frame control field which has according to Gast [Gas] the following components:

- Protocol version: Two bits are used to represent the protocol version. The version is incremented when a new 802.11 standard renders incompatible with older versions.
- Type and subtype: The type value is especially interesting because it is used to decide if a frame is a management frame, control frame or data frame. Since management frames had been captured only, this field is used to filter out all other frames. All possible type and subtype values are shown in the Table 2.2

- ToDS and FromDS bit: These bits determine if a frame origins from a Distribution System or is destined to a distribution system. Management and control frames have both bits set to '0'. Data frames in infrastructure networks have the ToDS bit set if transmitted from a wireles station.
- More fragments bit: This bit is set if a higher level packet is fragmented in this layer and this is not the final frame.
- Retry bit: Is set to '1' if this frame gets retransmitted.
- Power management bit: This bit is set to '1' if the device will go into power saving mode after sending this frame. Access points are not allowed to go into power saving mode.
- More data bit: This bit is used for stations which are in power saving mode, to inform them, that there is more data available.
- WEP bit: If Wireless Equivalent Privacy (WEP) is used, this bit is enabled. WEP is highly insecure and broken [LxDRpW10]. Therefore, this bit should not be used.
- Order bit: If it is set to '1', strict ordering is used.

The second field in the 802.11 frame(2.2) is the Duration or ID field. It is 16 bits long and handles low level transmissions. To understand this field two important aspects need to be understood, Network Allocation Vector (NAV) and contention-free period. This field has different meanings depending on the 15th and 14th bit, when the 15th bit is '0' it contains the NAV. Otherwise if the 15th bit is '1' and the 14th bit is '0' it means that the medium is in the contention-free period. The NAV is a virtual carrier sensing mechanism, if the NAV field has a non-zero number other stations are asked to defer access to the medium for the number of microseconds which are transmitted in the NAV field. This mechanism makes it possible to prevent collisions in the medium. There is also a physical carrier-sensing function which is based on energy thresholds and other physical measurements. Every station receiving a frame checks the duration field and updates their own NAV counter accordingly. The NAV is counted down to zero, only then it is allowed to send a frame. The contention-free period is announced with a Beacon. In the contention-free period the AP decides who should access the medium, but this is rarely implemented by APs.

Following the second field there are the address fields. In total there are four address fields in an 802.11 frame. This is also a difference between Ethernet and 802.11 frames. The address fields are numbered because the meaning depends on the frame type.

According to Gast [Gas], there are five use cases for the address fields:

- Destination address: Is equal to the destination address in Ethernet and is used for the final recipient, which is the station which will give the frame to higher layers for processing. This is also discussed in the ISO/OSI reference model.

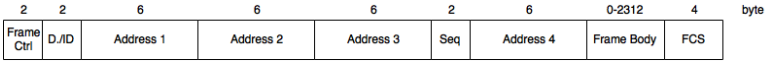

Figure 2.2: General frame format [Gas].

- Source address: This address is used to specify where a frame or transmission is coming from.
- Receiver address: This address determines which wireless station should process the frame. If the frame is destined to a wireless station then the receiver and destination address are the same. If not the destination address could be a router which is connected via ethernet with the AP
- Transmitter address: Specifies which device has been used to transmit the frame onto the wireless medium. This address is only used in wireless bridging.
- Basic Service Set Identifier (BSSID): For identifying different Wireless LANs (WLANs), a station has a Basic Service Set (BSS) assigned. For infrastructure WLANs this is the MAC address of the AP.

Following the first three address fields is the sequence control field. It consists of 4 bit fragment number and an 12 bit sequence number which totals to 2 bytes. This field is used for defragmentation and to detect duplicate frames. The sequence number is always incremented by one for every frame sent, unless the frames are fragmented, then the sequence number stays the same for all fragments, but the fragment number is increased. If the frame is not fragmented or if it is the first fragment then the fragment number is 0. Retransmissions keep the sequence number. On the sequence number counter a modulo 4096 is applied to not exceed the maximum value of 4096.

The frame body consists of the payload from higher layers. The maximum size of the payload in this layer is 2034 bytes.

The last field in the 802.11 frame is the frame check sequence. This checksum takes all header fields and body fields into account. It is calculated when the frame reaches the wireless interfaces which sends the frame onto the wireless medium. The receiving wireless interface checks the integrity of the frame and if the checksum holds the probability is high that there was no transmission error. Unlike Ethernet the wireless receiver has to send a positive acknowledgment if the checksum of the frame is correct. Otherwise the sender would re-transmit the frame after the acknowledgment timeout.

The IEEE 802.11 divides the different frames into types and subtypes. Table 2.2 shows an overview of the frame types and its corresponding subtypes.

A MAC address is 48 bits long and is usually divided into six octets each octet can be shown as a hex value. FF:FF:FF:FF:FF:FF would be a valid MAC address and in this case also used as broadcast address [80214]. There are two different representations distinguished:

Subtype values	**Subtype names**
Management Frames(type=00)	
0000	Association Request
0001	Association Response
0010	Reassociation Request
0011	Reassociation Response
0100	Probe Request
0101	Probe Response
0110	
0111	
1000	Beacon
1001	Announcement Traffic Indication Message
1010	Disassociation
1011	Authentication
1100	Deauthentication
1101	
1110	
1111	
Control Frames(type=01)	
1010	Power Save (PS)-Poll
1011	RTS
1100	CTS
1101	Acknowledgment(ACK)
1110	Contention-Free (CF)-End
1111	CF-End+CF-Ack
Data frames(type=10)	
0000	Data
0001	Data+CF-Ack
0010	Data-CF-Poll
0011	Data+CF-Ack+CF-Poll

Table 2.2: IEEE 802.11 frame types and subtypes [Gas].

Binary representation	Hex	Is local
00**0**0	0	false
00**0**1	1	false
00**1**0	2	true
00**1**1	3	true
01**0**0	4	false
01**0**1	5	false
01**1**0	6	true
01**1**1	7	true
10**0**0	8	false
10**0**1	9	false
10**1**0	A	true
10**1**1	B	true
11**0**0	C	false
11**0**1	D	false
11**1**0	E	true
11**1**1	F	true

Table 2.3: Universal/Local address bit mapping.

- Hexadecimal representation
- bit reversed representation

These two representations are illustrated in Figure 2.3. As the name suggests, the bit reversed representation reverses the bits in one octet. This is especially important if the universal/local bit is evaluated since the position changes. The first part of the Figure shows the MAC address represented in reverse order. The Table below shows it in hexadecimal representation. In the IEEE 802 standard the bit-reversed representation is only of historic interest [80216]. The MAC address is in general assigned from the manufacturer of the hardware device and in general never changes over its lifespan. To eliminate the risk that different manufacturers assign the same MAC address the IEEE Standards Association functions as a registration authority [ieec]. It is distinguished between OUIs and CIDs. An OUI is a 24, 28 or 36 bit unique identifier used to generate a universally unique MAC address. OUIs always have the universal/local bit set to zero [80214]. A CID is also 24 bit long and used if the MAC address does not need to be universally unique [ieea].

In this master thesis the hexadecimal representation is used to represent a MAC address. Therefore, to calculate if the MAC address has set the local bit the second hexadecimal character of the MAC address string was used and checked against Table 2.3

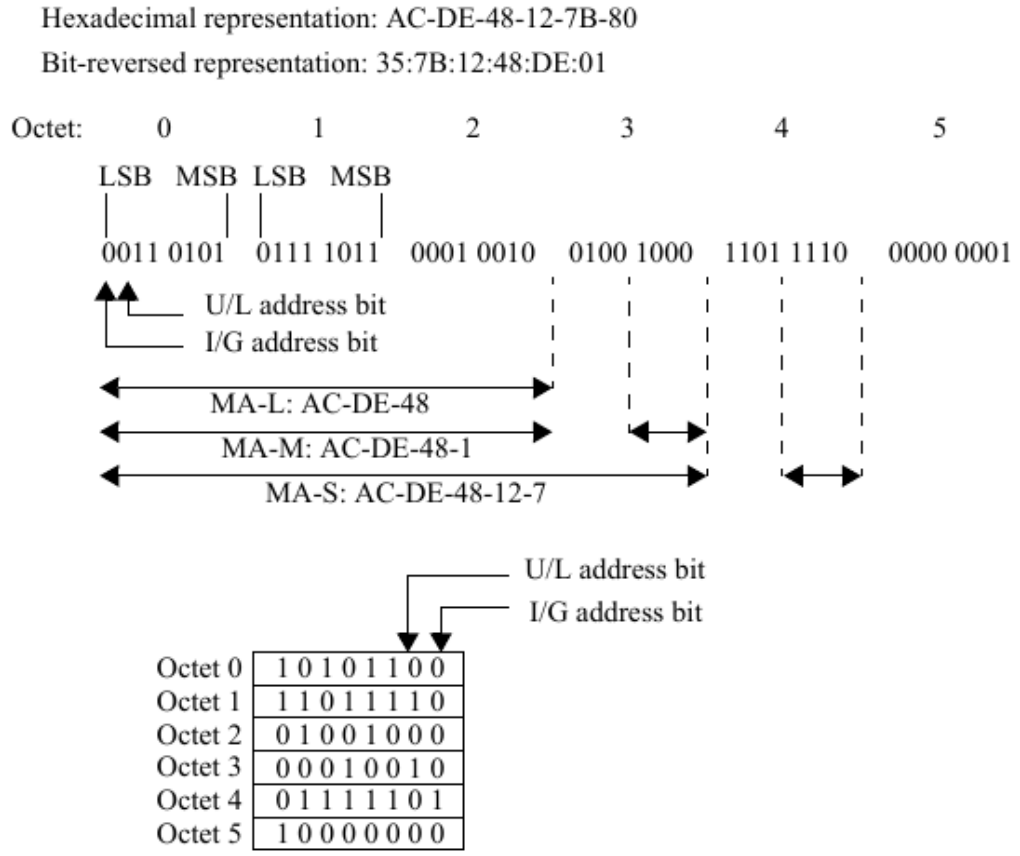

Figure 2.3: MAC address representation [80214].

A station is a Wi-Fi enabled device which can either be an AP or a client device e.g., a smartphone or laptop.

The BSS identifies a set of stations which form together a wireless network. This wireless network uses the same networking standards to communicate with one another. Each BSS is identified with the BSSID which is in general the MAC address of the access point. If two or more BSS are connected through a distribution network it forms an Extended Service Set (ESS). The distribution system can be any network like a ethernet connection [Che13].

Basically speaking there are two different kinds of wireless networks: Infrastructure based WLANs and Ad-Hoc WLANs. Infrastructure based WLANs use a predefined base station e.g., an AP which functions as a manager node and all other stations only open connections between the AP and their self to transfer network packets, illustrated in Figure 2.4. Most wireless networks use infrastructure based WLANs. Ad-Hoc WLANs do not need a predefined station. The stations in an Ad-Hoc network communicate directly

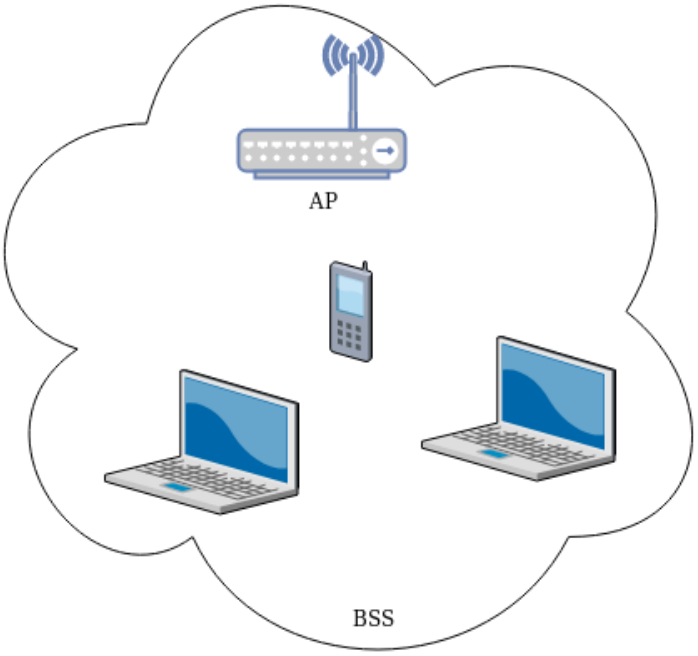

Figure 2.4: Infrastructure based WLANs [Che13].

with each other also called Peer to Peer (P2P) networking [Che13].

CHAPTER 3

Related Work

In this chapter multiple research papers are outlined and brought into context of this master thesis.

According to Freudiger [Fre15] the privacy threat of Wi-Fi Probe Requests gets quantified with an experimental study on specific smartphones. According to their study, an average mobile device sends out 55 Probe Requests in an hour. In the data collections conducted in this master thesis an average of 503 and 369 Probe Requests per minute have been measured in the Billboard Advertising and Supermarket dataset respectively. The total amount in both datasets of captured PR was more than 29 million.

In Vanhoef et al. [VMC+16] several techniques are presented which allow tracking of unassociated devices. Furthermore, a tracking algorithm is proposed which does not rely on MAC addresses to uniquely identify a device. Demonstrated on a real world dataset, the algorithm tracks about 50% of the devices for at least 20 minutes. To improve the algorithm also scrambler seeds are analyzed. In this master thesis a tracking algorithm is proposed which is solely based on directed Probe Requests and the fact that some network names are very distinguishable. This method can assign about 19.5% of MAC addresses to at least one other MAC address recorded in the data collection if special criteria are met.

An approach to circumvent MAC randomization discussed in Vanhoef et al. [VMC+16] is a fake hotspot with common public SSID names, which many client devices have in their preferred network list. A client device has therefore a specific probability to send an association request with his real device MAC address. In this thesis the data collection was generated by completely passive devices that were just listening on specific frequencies. Meaning that no device was acting as an AP or deceiving passing devices by sending out special frames in any way. This method used is therefore not detectable. A fake hotspot on the other hand can be detected by an expert user.

Another proposal in Vanhoef et al. [VMC+16] to circumvent MAC randomization is to use Hotspot 2.0 features or in more detail an ANQP request, which can reveal the real MAC address. In this thesis it is shown that a high number of randomized MAC addresses are re-used. Some devices had a local bit MAC address and could be seen during the whole data collection period.

According to Pang et al. [PGG+07] implicit identifiers of 802.11 devices can identify devices with high accuracy. It is estimated that adversaries can track 64% of the devices with an accuracy of 90%. [PGG+07]

A layer 1 attack is introduced in Bloessl et al. [BSDE15] which uses the scrambler seed in the physical layer to re-identify a device. In this special case it disables the location privacy of vehicular networks. In essence this vehicular networks should enable cars to exchange messages. Therefore, so called Roadside Unitss (RSUs) are deployed which act as access points. Whoever runs these access points could locate a car, because it sends out regular Beacon requests. For this reason changing identifiers have been introduced. According to Bloessl et al. [BSDE15], this attack works on bit level and not with signal characteristics which makes it more robust. It builds up on the fact that a simplistic pseudo random number generator is used, which makes sequences of the scrambler state predictable. Since this is a layer 1 attack it renders all privacy protection mechanisms on upper layers useless.

In another paper Cunche et al. [CKB14] concentrate on finding social links between the owner of devices. Therefore, they used a dataset collected in Sydney, Australia and composed of more than 8000 devices. Furthermore, they use the directed Probe Requests to find the preferred networks of the owner and infer relations based on the probed SSIDs. In this master thesis a two datasets with 2.5 million MAC addresses and 700 000 MAC addresses was used to analyze and evaluate the privacy concerns mentioned earlier in the research questions. It is also shown to that MAC randomization can be circumvented if directed Probe Requests are used.

The paper from Pang et al. [PGSW07] states that "Local service discovery exposes sensitive information about identity, location, and relationships. We present empirical evidence that suggests the release of this information in wireless environments poses a real danger to our privacy." They then discuss an architecture that enhances the privacy of existing service discovery methods. The data collections realized in this master thesis also show that many devices can be tracked over multiple days just with the unique identifier of the network interface.

Gruteser and Grunwald [GG05] suggest disposable interface identifiers in their paper about "Enhancing Location Privacy in Wireless LAN Through Disposable Interface Identifiers". This early paper about location privacy and disposable MAC addresses shows its fruits in the data collection made in this master thesis. Hence, over 70% of recorded MAC addresses have the local bit set, which means that these MAC addresses are disposable. Since the 70% is calculated out of MAC addresses which used Probe Requests. So only client devices are considered in this calculation.

Since internet privacy is a serious concern nowadays according to Bernardos et al. [BZO15]. In this paper the main problem of the unique Layer 2 MAC address is discussed. Therefore, some devices were configured with a special script to randomize their MAC address during meetings at the IETF and an IEEE meeting during this trial a few hundred MAC addresses were seen which participated. During the trial an average "lifetime" of a device was around 4 minutes and 46 seconds. It is also noted that MAC randomization is influenced by the context. For example if L2 address access filtering is used by the AP MAC randomization would make connecting to the network impossible. This master thesis concentrates on a real world set up where no self developed scripts come into play on the captured devices. Since the data collection lasted for multiple weeks a more accurate statement can be provided concerning the current implementation of MAC randomization in the majority of devices recorded.

In a paper from Desmond et al. [DYPL08] a device fingerprinting technique is proposed which can differentiate between unique devices based on timing analysis of Probe Requests. Furthermore, it is shown that for different Network Interface Card (NIC) drivers the time interval for periodic Probe Requests is distinct. They achieved an accuracy of 70% to 80% by solely using timing analysis of Probe Request frames to distinguish between unique devices in their environment.

The paper of Musa and Eriksson [ME12] describes a method to passively track Wi-Fi devices based on Wi-Fi frame capturing. A method was developed to detect trajectories of Wi-Fi detections. An also interesting finding was that a large amount of MAC addresses had no listed Organizationally unique identifier (OUI) prefix and no local bit set. In contrast also a majority of MAC addresses had no listed OUI in this master thesis, but the majority of those had the local bit set and are therefore assumed to be randomized MAC addresses or MAC addresses from virtual environments. Moreover, addresses with the local bit should have been listed in the Company ID (CID), but as it turned out this was not the case for the majority of local MAC addresses. The only company which widely used CID listed MAC addresses in the data collection was Google Inc.

CHAPTER 4

Methodology

After an extensive literature review and data collection, a dataset with captured IEEE 802.11 management frames has to be analyzed to answer the following research questions:

- What is the entropy of MAC addresses collected in the field studies?
- What part of the collected data in the field studies can be used to identify the re-occurrence of users over multiple days?
- What part of the data, collected in the field studies, could be used to draw conclusions if a MAC address is randomized or not?

Since there was no extensive dataset available for IEEE 802.11 management frames it was necessary to create a dataset which could be analyzed in a later step. For this reason it was necessary to develop sniffing devices which were able to reliable capture IEEE 802.11 frames.

The main aspects for the sniffing devices were, that they could be easily fitted into small spaces for example cash desks and that there were no sophisticated deployment procedures necessary for the person on site. It was also important that after a power failure the device automatically started again and continued sniffing without any user interaction.

For this reason the following aspects have been identified:

- small form factor
- little weight
- standard power supply socket

- noiseless
- plug and play
- cheap components
- monitor mode enabled Wi-Fi dongle
- functionable without internet connection
- little energy consumption
- power loss resistant
- correct system time

Once the devices had been deployed there was no user interaction necessary till the data collection was finished and the devices got removed from site.

As a result of those requirements a Raspberry Pi[1] 3 board was chosen with a micro usb port for power supply, which has a maximum energy consumption of about $2.5A * 5V = 12.5W$. Also the form factor is small enough to fit into small spaces, e.g., under a supermarket checkout desk. Furthermore, no active coolers guaranteed a noiseless operation.

The Raspberry Pi 3 has an internal Wi-Fi chip, but sadly with the kernel used, this chip could not be turned into monitor mode, which is necessary to capture all frames, even those which are not destined for the capturing device. For this reason a external Wi-Fi dongle which met the criteria to be set in monitor mode was used.

The challenge with power loss resistance was solved with a Real Time Clock (RTC). If no RTC is used, the Raspberry Pi gets the system time from an ntp server. Since no internet connection was available during the research period, on a power loss the system time would have been reset and could not be recovered. For the recording of captured data frames, time was essential.

The operating system and all data is stored on an SD card which can be cloned by standard bash tools. This feature was very important for the scalability of these data collections. With this concept the software part of a new sniffing device was generated with standard linux bash commands within half an hour and nearly no user interaction.

All components together had a total cost of about 66EUR per sniffing device at the beginning of the data collection. This sum is divided into the following components:

- Raspberry Pi 3 Platine for 35EUR
- Wi-Fi Dongle 2.4Ghz for 10EUR

[1] https://www.raspberrypi.org/

- Micro Usb Power Supply 2.5Ampere (A) for 10EUR
- RTC for 1EUR
- Micro SD-Card for 10EUR

The RTCs are usually very cheap but they had very poor quality. Some RTCs had broken soldering joints and had to be repaired.

It was important that for the Wi-Fi dongle there were linux drivers available which could set the Wi-Fi dongle into monitor mode. Otherwise it would not be possible to receive frames which are not addressed to this device. The external Wi-Fi dongle was a TP-Link TL-WN722N with a chipset version of v1.10. The chipset version was very important because with version 2.0 there were no drivers available which were able to set the networking interface into monitor mode. This Wi-Fi dongle supported the IEEE802.11n, IEEE 802.11g and IEEE 802.11b standards and the following modulation technologies: DBPSK, DQPSK, CCK, OFDM, 16-QAM and 64-QAM [tpl].

To speed up the setup of the devices one device was set up and an image was created of the final configuration which could be easily transferred to another SD card with the same size. For image creation the bash command in Listing 4.1 was used:

```
dd if=raspiTShark.img of=/dev/mmcblk0 bs=4M
```

Listing 4.1: Image copy command.

The script in Listing 4.2 was used to capture management frames only.

```
#!/bin/sh

FILE=log.cap

if [ -f $FILE ];
then
 mv $FILE ${FILE}_$(date +%Y%m%d%H%M)
fi

cd /home/pi

logger "tshark started"
until /usr/bin/tshark -w $FILE -i wlan_extern -f "type mgt"; do
   logger "tshark crashed with exit code $?. Respawning.."
   sleep 10
done

logger "tshark ended"
```

Listing 4.2: Management frames capture script.

The script stores all data into log.cap, but to avoid overriding existing data it renames the file, if it exists, and adds a timestamp to it. It has also a build in fail safe, if tshark should crash at some point it will restart automatically in ten seconds. To capture the actual Wi-Fi frames tshark is used. In order to avoid capturing all Wi-Fi frames the parameter "f" which means filter is used and set to management frames. Hence, this script is not switching between channels another script was started at startup which set the Wi-Fi interface into monitor mode and switched the channel every 0.1 seconds. This script is presented in Listing 4.3.

```
#!/bin/bash
IFACE=wlan_extern

/sbin/ip link set $IFACE down
/sbin/iw dev $IFACE set type monitor
/sbin/ip link set $IFACE up

while true ; do
  for CHAN in "1 2 3 4 5 6 7 8 9 10 11" ; do
    echo "Switching channel $CHAN"
    /sbin/iw dev $IFACE set channel $CHAN
    sleep 0.1
  done
done
```

Listing 4.3: Channel switching script.

Sadly a mistake was made within this script and it only switched between channels 1 to 11, but 13 channels would have been available in Austria [rtr]. However, when looking at the data collection more than enough data was captured.

Both scripts were started after boot with a cronjob.

Added to the hardware costs there is an initial setup cost which needs to be considered for creating the first device. For all other devices the SD-card of the first device can be copied and used with only minimal changes. E.g., it is not known which wireless chip is the external Wi-Fi dongle and which one is the internal management Wi-Fi chip. But maybe this could also be automated by considering on which USB ports the devices are connected. Since it couldn't be automated in time, a udev rule was created in /etc/udev/rule.d/70-persistent-net.rules where the MAC addresses of each interface had to be assigned. An example file is presented in Listing 4.4.

```
SUBSYSTEM=="net", ACTION=="add", ATTR{address}=="12:23:45:67:89:ab",
    NAME="wlan_intern"
SUBSYSTEM=="net", ACTION=="add", ATTR{address}=="de:ad:be:ef:12:34",
    NAME="wlan_extern"
```

Listing 4.4: Udev rule for network interface naming.

Considering that Wi-Fi works only in ranges fewer than one hundred meters [SVS+06] there were only devices captured which were in proximity of the sniffing device. A Wi-Fi dongle with an external antenna was used.

In total twelve devices were prepared for deployment and used for the following timely independent data collections:

- Supermarket data collection
- Billboard Advertising data collection

It is important to note, that these data collections are not entirely independent, because of their proximity to each other. All data collections took place in Vienna/Austria. This means that there is a higher possibility that a device's MAC address was captured in multiple datasets.

The first data collection with a supermarket company lasted for approximately two weeks from end of April to mid-May. The devices were placed under the point of sale cash desks. So whenever a customer left the shop his or her device would have been captured. For very big branches with many cash desks not every customer device could be captured because the range of the sniffing devices was rather limited.

The second data collection took place on the street and some devices were placed on very popular shopping streets in Vienna. Obviously the devices could not be placed on the open street because they are not waterproof and neither are they protected from theft or vandalism. Therefore, a partnership with a billboard advertising company was made, which allowed the placement of capturing devices within their boxes for electrical installations on the street. An example of a deployed device can be seen in Figure 4.1.

To check if the devices are functional during the data collection it was necessary to connect over wireless. Because it was not easily possible to gather fast physical access without bringing lots of equipment like monitor and keyboard. It would have also interfered with the person working at the point of sale. Therefore, the internal wireless chip was used for management access. The internal wireless card was programmed to connect to a specified SSID. Therefore, the /etc/network/interfaces file, which is shown in Listing 4.5, needed to be changed so that the internal network interfaces connects to the specified network in the wpa_supplicant.conf

```
...

iface wlan_intern inet manual
   wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
```

Listing 4.5: /etc/network/interfaces file.

The wpa_supplicant.conf consists only of the network name and password, as shown in Listing 4.6.

Figure 4.1: Deployed Raspberry Pi, ready for data collection.

```
network={
   ssid="myAndroidHotspot"
   psk=deafbeef

}
```

Listing 4.6: WPA Supplicant configuration.

The specified network could then easily be created with the Android Hotspot function. This made it possible to use an Android device and its hotspot function to connect via ssh to the sniffing device, check the health status and to copy the log files. This setup made it necessary to clean the log files from the Probe Requests of the internal Wi-Fi chip, since it would have distorted the results. The cleaning was easily done over the unique SSID which the management network used.

The second data collection was realized with an advertising company where the Raspberry Pis were placed near billboards mostly directly in a box on the street. This data collection lasted for four weeks in the summer of 2017. Some devices were placed in very popular shopping streets in the inner districts of Vienna.

After the data collection the results were checked against errors and plausibility. Two sniffing stations had to be removed from the dataset, because the data was not stable. Furthermore, the collected dataset was evaluated to answer the specified research questions, which are further discussed in chapter 6. The two data collections have been

frame.time	**wlan.sa**	**wlan.da**	**wlan.seq**
Mar 30, 2017 10:00:00.2	11:11:11:11:11:11	ff:ff:ff:ff:ff:ff	800
wlan.fc.type	**wlan.fc.subtype**	**wlan__mgt.ssid**	**frame.cap__len**
0	8	UPC123	200

Table 4.1: Sample of one frame in CSV format.

evaluated independently of each other. Hence, every result was calculated for each dataset and then compared against each other. In most cases the results of the datasets were very similar.

To circumvent possible problems the Raspberry Pi was restarted every day so there was a new pcap file generated for every day.

The Wi-Fi frames had been captured with tshark [tsh] and were therefore stored in the native format, so called pcap files.

This pcap files could only be read by special programs, therefore they needed to be converted in a format which was generally readable. Since the data should be made available through a Structured Query Language (SQL) database, the pcap files were converted to Comma-separated values (CSV) files.

For conversion a python script was written, which was called for every station with the parameters in Listing 4.7.

```
python3 convertCapToCsv.py --inputDir
    /mt/logs/Supermarket/location1/ \
--outputDir /mt/logCsv/Supermarket --location supermarket_location1
    --start 1
```

Listing 4.7: Cap- to csv-file conversion call.

In Listing 4.8 an excerpt from the python script convertCapToCsv.py is presented, which called tshark for every pcap file.

```
tshark -r input.pcap -T fields -e frame.time -e wlan.sa ... >
    output.csv
```

Listing 4.8: Tshark parameters for csv conversion.

These csv files could then be read into a SQL database. An example of the csv format is shown in Table 4.1, which shows a Beacon from an AP. For a detailed listing of all possible field names refer to Reference [wira] and Reference [wirb].

For the SQL database the following Tables have been used: 'frames', 'dates', 'locations', 'ssids' and other tables which were created during evaluation of the dataset. They are shown in Table 4.2, 4.3, 4.5 and 4.4. Every table used an artificial Primary Key (PK),

Column name	Datatype
frameId	bigint, PK
locationName	varchar(30)
logDate	timestamp
sourceMac	varchar(17), index
destMac	varchar(17)
length	int
subtype	int
sn	int
ssidName	varchar(256)

Table 4.2: Structure of 'frames' table.

Column name	Datatype
Date	timestamp, PK
minute	int
hour	int
dayOfMonth	int
weekday	int
month	int
year	int

Table 4.3: Structure of 'dates' table.

Column name	Datatype
ssidId	bigint, PK
name	varchar(256)
longitude	varchar(50)
latitude	varchar(50)

Table 4.4: Structure of 'ssids' table.

Column name	Datatype
name	varchar(50), PK

Table 4.5: Structure of 'location' table.

except the 'location' table. In the frames table there was also an index created for the source MAC address because this field was very often used in the evaluation process. The 'dates' table exists also for performance reasons, for example if a fast selection of all MAC addresses, which have been seen on a Sunday, was necessary. Other than the tables 'frames', 'dates' and 'locations', Table 4.4 was not read from a csv file, but needed to be resolved against a geo-location service like Wigle[2]. Since the request limit per day was very low and could, after negotiations with the service, only be increased to a few hundred requests, it was not possible to resolve every SSID captured during the data collection.

The dataset contained over half a million different network names too much to analyze manually. Therefore, an algorithm was developed to find high value SSIDs within the datasets of the research study. This algorithm was also necessary because of the before mentioned limitation, there was no extensive dataset publicly available which could be used to map a high amount of network names to geographic locations. This will also be explained in more detail in Section 6.4. The algorithm calculated the Shannon Entropy [Sto15] of the network names. The Shannon entropy was then used to order the network names from the highest entropy to the lowest entropy. A high entropy means that this SSID is considered interesting and should be queried in the location service. A low entropy means only query if the limit of the geo-location service allows it. To measure the correctness of the algorithm a wireless network mapping service was queried.

[2]https://wigle.net

If the query gave zero or one result locations back the SSID is considered as high value.

This algorithm favors network names which are longer since the formular for the Shannon Entropy is:

$H(x) = \sum_1^n P(x_i) \log_b P(x_i)$ [Sto15]

were $P(x_i)$ is the probability of the letter i in the network name and n is the length of the network name.

Since this algorithm did not favor SSIDs from a big ISP in Vienna namely UPC, but the network names seem fairly unique and are approximately in the format: "UPC[0-9]{6,8}" Therefore, a subset of 1,000 network names were tested against the geo-location service and the average returned number of results is 1.54 as mentioned in 5.2. The maximum returned locations per SSID was six.

This led to another method to find out valuable SSIDs which are assigned from big companies, a so called grouping of SSIDs. The results of the process are shown in 5.12. This algorithm shows that SSIDs starting with "upc" are by far the most common used network names in this subset.

For requesting SSID locations from wigle.net a special endpoint[3] was used with a GET Parameter named 'ssid', which was set to the appropriate network name. To verify the identity of the sender, BASIC Authorization was used. The result of the request was a JSON array, each element consisted of the latitude and longitude where an AP was recorded with that name. If the array had only one element, the network name could be unambiguously mapped to a location.

For plotting the diagrams in Section 5, a program common in statistics namely R was used. For example, Figure 5.7 was generated with the Listing 4.9:

```
library(RMySQL)
mydb = dbConnect(MySQL(), user='master', password='***',
    dbname='MT_EVALUATION', host='127.0.0.1', port=3308)
rs = dbSendQuery(mydb, paste("select count(*) as amount,
    concat(month,'/',dayOfMonth,'/',year) as time from (select
    sourceMac, dayOfMonth, month, year from frames join MI_dates on
    logdate=date where locationName like 'BillboardAdvertising%' and
    logdate>='2017-06-21 00:00:00' and logdate<='2017-07-20 23:59:59'
    group by sourceMac, dayOfMonth, month, year) macs group by
    dayOfMonth, month, year order by year, month, dayOfMonth;",
    sep=""))
logs = fetch(rs, n=-1)
logs$time <- as.Date(logs$time, "%m/%d/%Y")

rs4 = dbSendQuery(mydb, paste("select count(*) as amount,
    concat(month,'/',dayOfMonth,'/',year) as time from (select
    sourceMac, dayOfMonth, month, year from frames join MI_dates on
```

[3] https://api.wigle.net/api/v2/network/search

```
    logdate=date where locationName like 'BillboardAdvertising%' and
    logdate>='2017-06-21 00:00:00' and logdate<='2017-07-20 23:59:59'
    and subtype=4 group by sourceMac, dayOfMonth, month, year) macs
    group by dayOfMonth, month, year order by year, month,
    dayOfMonth;", sep=""))
logs4 = fetch(rs4, n=-1)
logs4$time <- as.Date(logs4$time, "%m/%d/%Y")

rs5 = dbSendQuery(mydb, paste("select count(*) as amount,
    concat(month,'/',dayOfMonth,'/',year) as time from (select
    sourceMac, dayOfMonth, month, year from frames join MI_dates on
    logdate=date where locationName like 'BillboardAdvertising%' and
    logdate>='2017-06-21 00:00:00' and logdate<='2017-07-20 23:59:59'
    and subtype=5 group by sourceMac, dayOfMonth, month, year) macs
    group by dayOfMonth, month, year order by year, month,
    dayOfMonth;", sep=""))
logs5 = fetch(rs5, n=-1)
logs5$time <- as.Date(logs5$time, "%m/%d/%Y")

rs8 = dbSendQuery(mydb, paste("select count(*) as amount,
    concat(month,'/',dayOfMonth,'/',year) as time from (select
    sourceMac, dayOfMonth, month, year from frames join MI_dates on
    logdate=date where locationName like 'BillboardAdvertising%' and
    logdate>='2017-06-21 00:00:00' and logdate<='2017-07-20 23:59:59'
    and subtype=8 group by sourceMac, dayOfMonth, month, year) macs
    group by dayOfMonth, month, year order by year, month,
    dayOfMonth;", sep=""))
logs8 = fetch(rs8, n=-1)
logs8$time <- as.Date(logs8$time, "%m/%d/%Y")

write.csv(file=paste("daily_uniqueMacs_all.csv"), x=logs, row.names
    = FALSE)
write.csv(file=paste("daily_uniqueMacs_all_Preq.csv"), x=logs4,
    row.names = FALSE)
write.csv(file=paste("daily_uniqueMacs_all_Pres.csv"), x=logs5,
    row.names = FALSE)
write.csv(file=paste("daily_uniqueMacs_all_Beacon.csv"), x=logs8,
    row.names = FALSE)

png(filename="daily_uniqueMacs_all.png",width=1920,height=1080,res=100)
par(cex.axis=1.5,cex.lab=1.5, cex.main=1.5, cex=1.5)
plot(logs$time, logs$amount, type="b",
xlab="Time", ylab="Mac-Addresses", xaxt = "n", pch=16, cex=1,
    ylim=c(min(logs8$amount),max(logs$amount)))
legend("topright",
    lty=c("solid","dotted","dotted","dotted"),col=c("black", "black",
    "blue","red"), legend=c("All", "PReq.","PRes.","Beacon"))
axis(1, logs$time, format(logs$time, "%a,%d.%m."), cex.axis = 1.5)
points(logs4$time, logs4$amount, lty="dotted", type="b", pch=16)
```

```
points(logs5$time, logs5$amount, lty="dotted",col="blue", type="b",
    pch=16)
points(logs8$time, logs8$amount, lty="dotted",col="red", type="b",
    pch=16)
dev.off()
```

Listing 4.9: Example of Diagram creation in R.

In this script the "RMySQL" package is used to extract the data from the SQL server and load it into a R data frame. In total there are four select statements which select the amount of MAC addresses and the day. The first statement selects all MAC addresses and the following statements select MAC addresses which sent out a frame with one specific subtype. After fetching all data from the SQL database it was saved into a CSV file, which made a re-draw of the diagram faster and easier because the SQL server would not be needed. Afterwards the "plot" function of R was used to render the figure and with the "points" function further lines were drawn into the figure.

Besides from R-Scripts there were also many standalone SQL statements used. For example to gather the number of frames captured in the Billboard Advertising dataset, the SQL statement in Listing 4.10 was used.

```
select count(*) as amount from frames where locationName like
    'BillboardAdvertising%' and logdate>='2017-06-21 00:00:00' and
    logdate<='2017-07-20 23:59:59';
```

Listing 4.10: SQL-Statement used to find out the number of captured frames.

For more complex queries which could not be directly expressed in SQL, Java programs have been written. Java programs were also used for creating tables during evaluation to store aggregated results.

In the next chapter there are two concepts used. The first concept are online seconds and the second concept are visits. Obviously there needs to be some definition of those concepts because during the data collection there are no durations collected but only specific points in time. The so called log time of a frame. Hence, to calculate the online time of a specific MAC address these log times were sorted after time and the distance between every log entry or frame was calculated. If the distance between two frames was less than five minutes, the time in between was considered as online time. If the log entries had a time distance of more than five minutes then it was considered as a new visit and the calculation of the online time started from zero. Moreover, if the frames were seen within five minutes it counted as one visit. Five minutes were chosen because it was assumed to be a reasonable time someone would stand in a queue at a supermarket or someone would stand on a street corner waiting.

Another interesting measure are the Probe Requests a device sends out per minute. For the calculation of the Probe Requests per minute every MAC address was considered to be a device. To be able to calculate the Probe Requests per minute only MAC addresses

were considered which sent at least two frames within exactly one minute. Hence, the minimal Probe Requests per minute would be two. This criteria was necessary to filter out outliers, which may have been seen only very brief. The calculation of the average was then done with

$$avg = \frac{1}{n} \sum_{i=1}^{n} f_i/(t1_i - t2_i)/60,$$

where f = is the amount of Probe Requests

$t1$ = is the time of the first frame

$t2$ = is the time of the last frame

The time distance between two Probe Requests must not exceed one minute. Otherwise i is incremented and f is set to zero and $t1$ is set to the current Probe Request. Hence, the sum in the formula above is necessary, because a MAC address can be seen not only once, but multiple times. As mentioned above $t1 - t2/60$ had to be at least 1 to be considered within the calculations.

CHAPTER 5

Results

In this chapter the results from two different data collections are presented in detail. The first data collection was conducted with a major supermarket company in Vienna and lasted approximately two weeks. The second collection was carried out with a major advertising company in Vienna for 30 days. A short summary of the results for every dataset is at the end of every section. The two data collections have very similar results, therefore only the bigger Billboard Advertising dataset is explained in every detail. The Supermarket dataset is described at first with some interesting differences.

5.1 Supermarket Dataset

In this data collection a total amount of 24 million frames have been captured between the 29th of April 2017 00:00:00 and the 14th of May 2017 23:59:59. This means on average there have been 1.5 million frames per day. These frames were sent from approximately 700,000 unique source MAC addresses. Due to MAC randomization this number can be higher than the actual unique devices. From the 700,000 MAC addresses 489,748 client devices had the local bit set. Of these addresses 37,555 MAC addresses set the local bit and used directed Probe Requests. Over 1.9 million frames were captured from client MAC addresses with the local bit set. This data is also summarized in Table 5.1.

The data collection took place at nine different supermarket store locations within Vienna. At the following approximate locations sniffing hardware was placed:

- Train station (T.S.) 1 and 2
- Inner city (I.C.) 1, 2, 3, 4 and 5
- Karlsplatz 1 and 2

Description	Amount
Collected frames	24,397,024
Collected frames w/ LB	1,948,350
Avg. frames per day	1,524,814
Unique MAC addresses	697,588
Client MAC addresses	684,747
Client MAC addresses w/ LB	489,748
Client MAC addresses w/ DPRs	105,023
Client MAC addresses w/ LB and DPRs	37,555
SSIDs captured	89,187

Table 5.1: Supermarket dataset, overview of collected data.

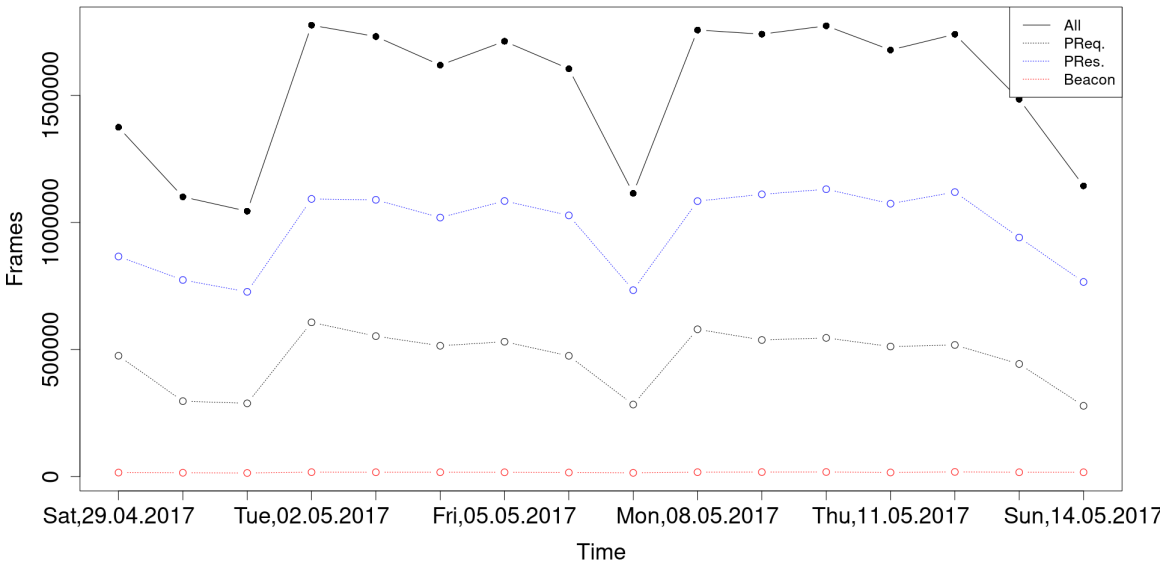

Figure 5.1: Supermarket dataset, captured frames per day.

In Figure 5.1 the captured frames are shown per day. Since this data collection took place in Austria within supermarkets, it is important to note that most supermarkets, except two at a train station were closed on Sundays. Also 1st of May is a national holiday in Austria. Therefore, the low amount of frames captured on Sundays and on the 1st May is not surprising. The figure also distinguishes between three different types of frame subtypes, namely Probe Requests, Probe Responses and Beacons. Probe Requests are in this dataset the most important type, because it reveals a client device nearby. The solid black line consists of all management frame subtypes combined.

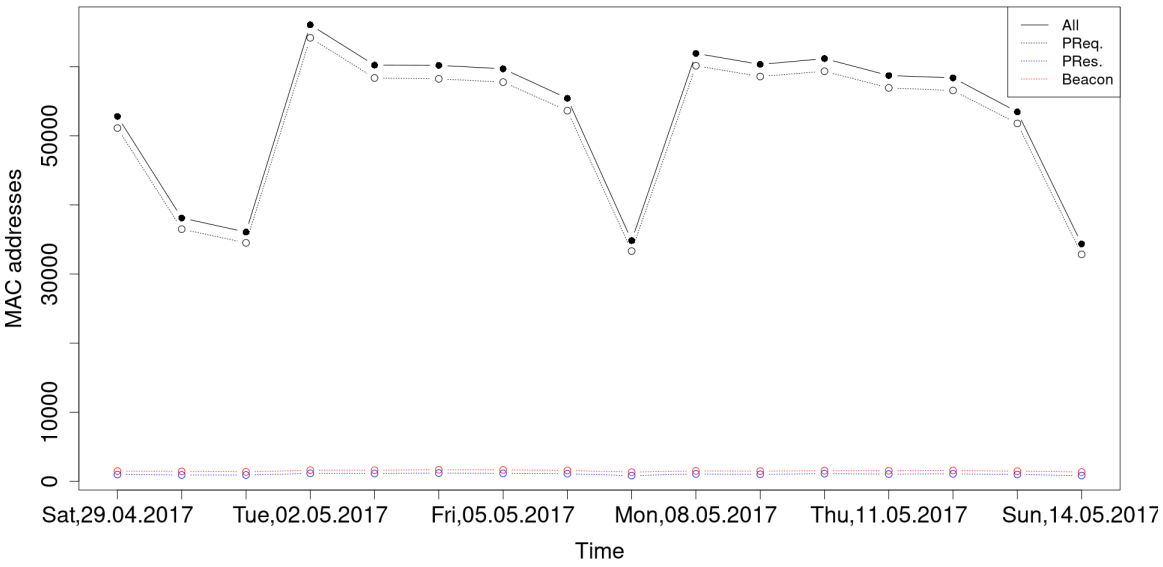

Figure 5.2: Supermarket dataset, captured unique MAC addresses per day.

In Figure 5.2 the different MAC addresses which were captured are shown for every day. It also shows a significant decrease on Sundays and the 1st of May.

In comparison to Figure 5.1 the amount of Probe Responses seems very minimal in Figure 5.2 this is because of the reason that few devices sent out many Probe Responses. This leads to the conclusion that most devices which have been captured are client devices. Since this data collection took place in locations where the use of a big device e.g., a laptop is very inconvenient, it can be assumed that most captured client devices are very small. This would lead to the conclusion that many devices are smartphones.

The two Figures 5.3 and 5.4 show the hourly distribution of unique MAC addresses on Sundays. It is very important to look at the scale of the y-axis. In 5.3 the peak is around 6pm with over 150 unique MAC addresses captured. In Figure 5.4 the peak is over 2500 MAC addresses high. This is due to the fact that the supermarket in Figure 5.4 is opened on Sundays.

The Figure 5.5 shows the captured frames per hour over the whole research period. All captured frames which were captured when the hour was x are summed up and the respective point is drawn where the x-axis shows the point x. E.g., frames which occurred at 05:00, 05:30 or 05:59 are all summed up at the data point 5.

In Figure 5.6 the captured seconds per station is shown. The red bar considers only MAC addresses with the local bit set. For more details on how a captured second is calculated refer to Chapter 4. The two stations "Train station 1" and "Train station 2" are the

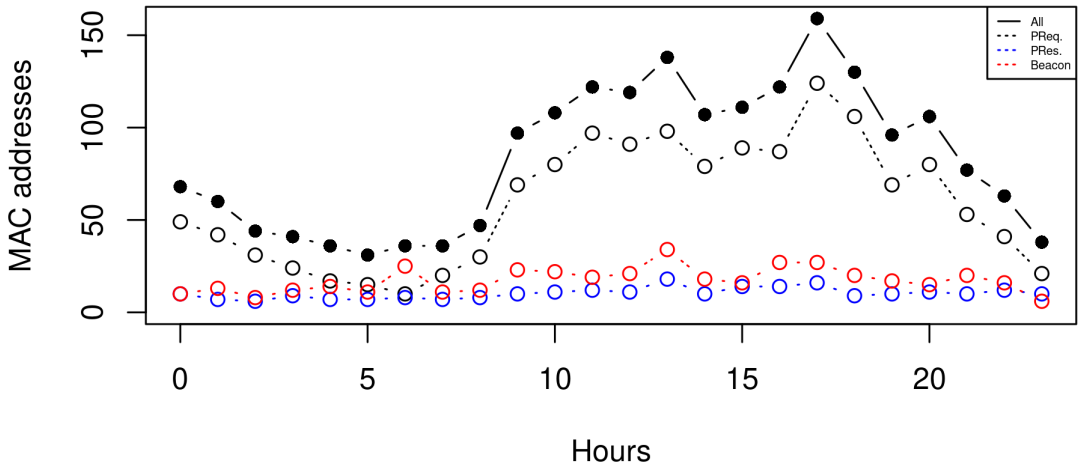

Figure 5.3: Supermarket dataset, captured unique MAC addresses on Sundays at Karlsplatz 1.

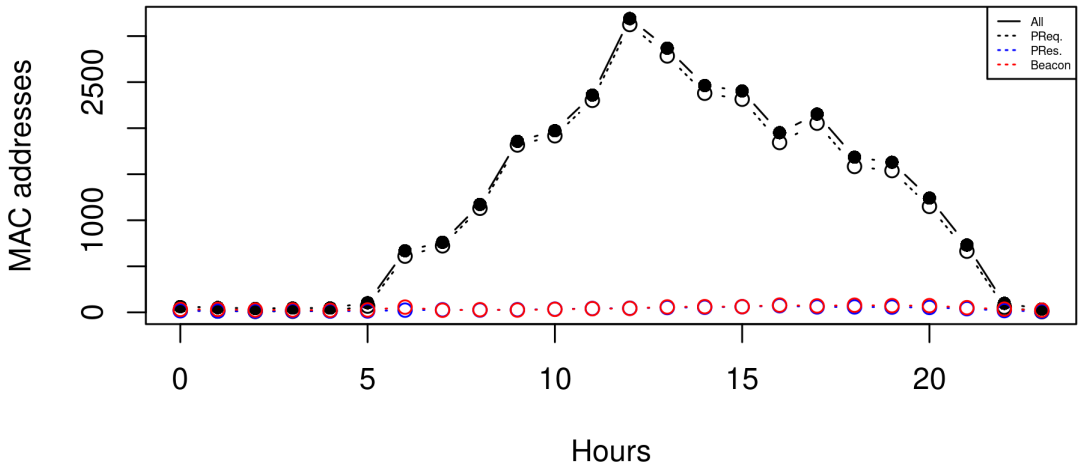

Figure 5.4: Supermarket dataset, captured unique MAC addresses on Sundays at Bahnhof 2.

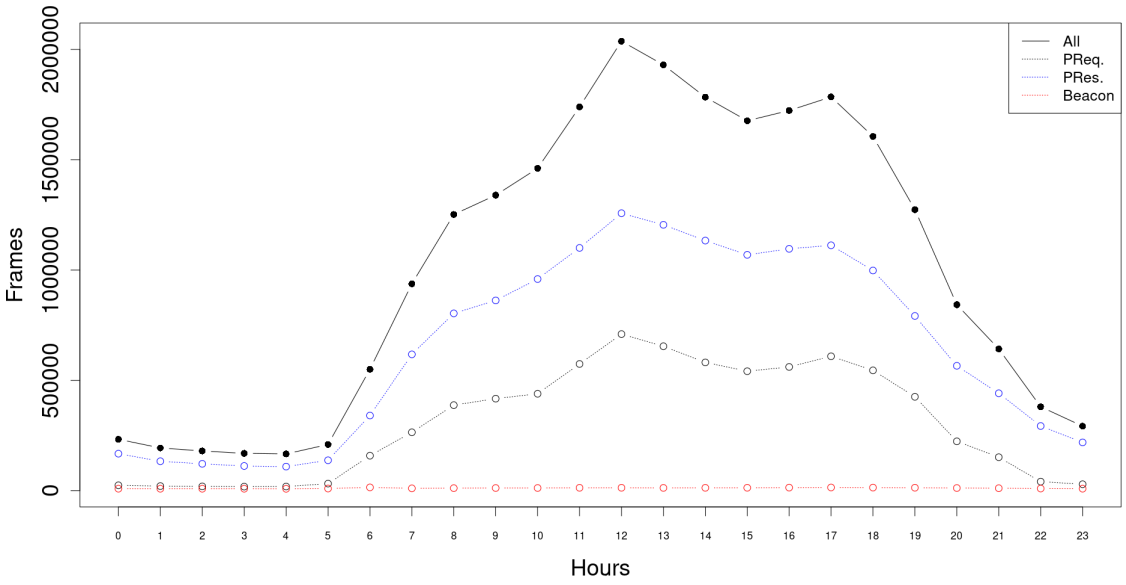

Figure 5.5: Supermarket dataset, captured unique MAC addresses per hour.

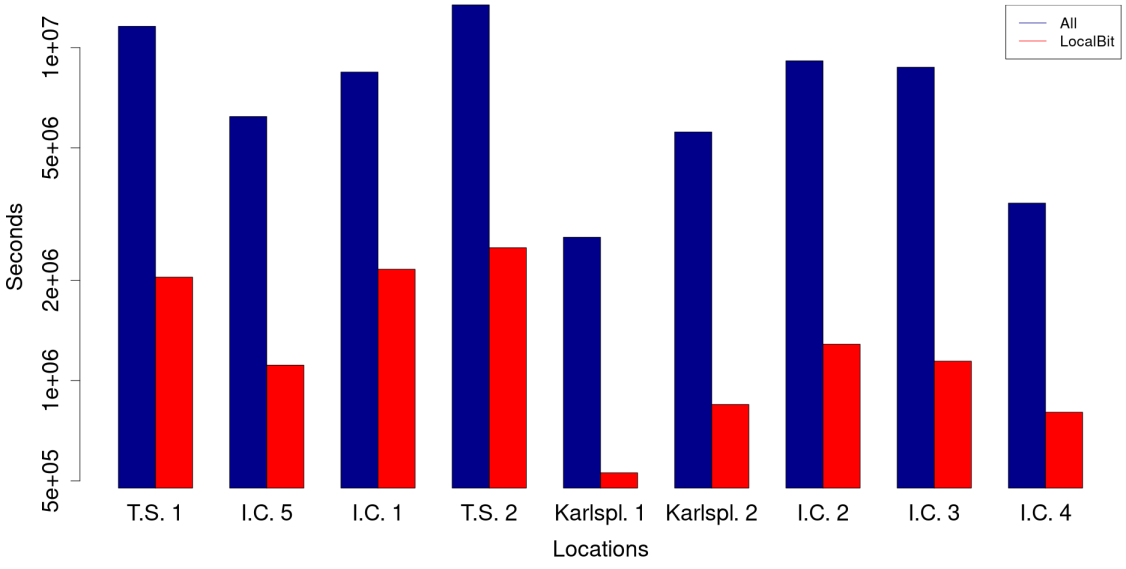

Figure 5.6: Supermarket dataset, captured seconds per location.

Location Name	Amount of MAC addresses
Train station 1	116,954
Train station 2	156,862
Karlsplatz 1	79,005
Karlsplatz 2	39,760
Inner city 1	92,915
Inner city 2	65,778
Inner city 3	92,919
Inner city 4	39,777
Inner city 5	48,458

Table 5.2: Supermarket dataset, MAC addresses per station.

busiest stations regarding the captured seconds. This means many devices tend to stay there for at least a few minutes. The station "Train station 2" has the most captured seconds with around 13,433,998. These results correlate with the results in Table 5.2 where "Train station 1" and "Train station 2" are also leading.

In Table 5.2 the unique MAC addresses per station are listed. It has to be considered that the summed up amount of all stations is higher as the amount stated in the summary Table 5.4. This is because of the reason, that some MAC addresses have been captured on multiple locations and are counted for every station once.

In Table 5.3 the top ten vendors of all captured devices are listed, according to the vendor list in Reference [oui]. This table does not consider that one company could register different address spaces under slightly different names. The biggest part are the not assigned MAC addresses which all have the local bit set. For this reason also the CID [cid] was queried against the MAC addresses and 69,240 of the not assigned MAC addresses belong to Google, Inc.

5.1.1 Summary

In the Table 5.4 a brief overview of the captured data is presented, which shows that nearly all MAC addresses sent out Probe Requests. Out of these, 70% had the local bit set. Table 5.5 shows this in more detail. Also the percentage values of the occurrence of management frame subtypes is shown.

In Table 5.6 a subset of MAC addresses is selected and a special algorithm found relations between MAC addresses according to there SSID probing behavior. The selection criteria is explained in 5.2

Vendor Name	**Captured MAC addresses**
not assigned	439,756
Samsung Electronics Co.,Ltd	44,782
Apple, Inc.	29,331
SAMSUNG ELECTRO-MECHANICS(THAILAND)	22,232
HUAWEI TECHNOLOGIES CO.,LTD	14,450
Motorola Mobility LLC, a Lenovo Company	10,342
Sony Mobile Communications AB	8,883
LG Electronics (Mobile Communications)	8,263
Murata Manufacturing Co., Ltd.	7,082
HTC Corporation	3,703

Table 5.3: Supermarket dataset, the ten most popular vendors [oui].

	# of frames	**# of MAC addresses**	**MACs with local bit**
Probe Requests	7,440,340	684,747	489,748
Probe Responses	15,639,562	8,548	1,665
Beacons	268,994	10,910	2,134
Other subtypes	1,048,128	15,432	522
Combined	**24,397,024**	**697,588**	**492,361**

Table 5.4: Summary of Supermarket data collection.

	# of frames	**# of MAC addresses**	**MACs with local bit**
Probe Requests	30.50%	98.16%	70.20%
Probe Responses	64.10%	1.23%	0.24%
Beacons	1.10%	1.56%	0.31%
Other subtypes	4.3%	2.2%	0.07%

Table 5.5: Summary of Supermarket data collection with percentage.

	# of MAC addresses	**Related to other MAC addresses**	**% of total**
Total	6,246	1,220	19.53%
Only LB	1,434	979	68.27%

Table 5.6: Supermarket dataset, "UPC" subset of directed Probe Requests.

Description	Amount
Collected frames	67,920,963
Collected Probe Requests	21,729,816
Avg. frames per day	2,264,032
Unique MAC addresses	2,555,186
Client MAC addresses	2,499,990
Client MAC addresses w/ DPRs	405,810
SSIDs captured	306,884

Table 5.7: Billboard Advertising dataset, overview of collected data.

5.2 Billboard Advertising Company Dataset

The Billboard Advertising data collection lasted for 30 days from 21st of June 00:00:00 till the 20th of July 23:59:59.

In this data collection there were over 67 million frames captured from over 2.5 million different MAC addresses. Out of the 67 million frames over 21 million were Probe Requests sent out from client devices seeking for an AP. The Probe Requests can be generally divided into to categories broadcast PRs and directed PRs. From a data science perspective the more interesting category are directed PRs. Directed PRs have an approximate share of 35.76% of the total Probe Requests captured. The directed Probe Requests were captured from over 405,000 MAC addresses which is a 16.23% share of the total 2.5 million MAC addresses. In the calculation of this share there are only MAC addresses considered which sent out at least one Probe Request over the whole duration of the data collection. Additionally, more than 306,884 SSIDs were logged from directed Probe Requests and Beacons. This data is also shown in Table 5.7.

The data collection was conducted at the following approximate locations in Vienna/Austria:

- I.C. 1,2,3 and 4
- Uni Wien
- Karlsplatz
- Mariahilferstr. 1, 2 and 3

Within two stations technical difficulties occurred, and they did not log the Wi-Fi data correctly. Therefore, this two stations are not considered in the evaluation of the dataset.

The following Figure 5.7 shows, how many unique MAC addresses were captured per day during the data collection. It distinguishes between three different types of management frame subtypes and shows with the black solid line all MAC addresses regardless of the

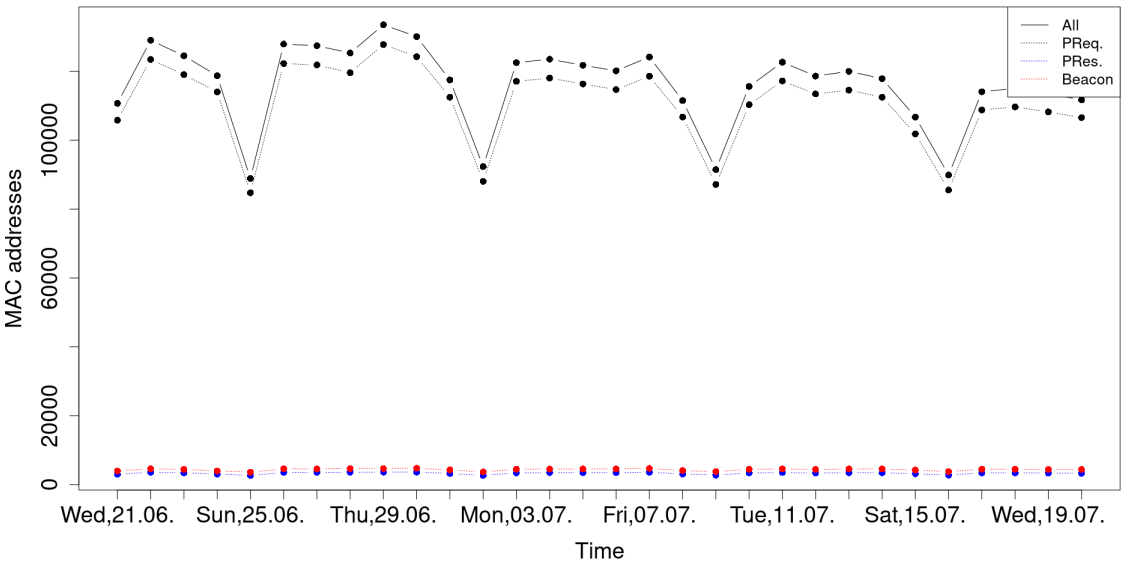

Figure 5.7: Billboard Advertising dataset, unique MAC addresses per day.

subtype. The black dotted line includes only MAC addresses which have sent out at least one Probe Request at the specified day. Analogous are the blue and red lines for Probe Responses and Beacons.

The Figure 5.7 shows clearly that Beacons and Probe Responses are sent out from very few MAC addresses, but Probe Requests are used from many different MAC addresses. Since Probe Requests are sent out from client devices, this concludes that mostly client devices and only a few access points were captured in this dataset. Because access points are often stationary and the sniffing devices were also stationary the fluctuation of AP MAC addresses was low. For the client devices, which often change the location, the fluctuation was high according to this dataset.

In Figure 5.8 the captured frames are broken down into days and the Probe Requests, Probe Responses and Beacons are shown separately. Figure 5.8 in comparison to Figure 5.7 shows a much higher share of Probe Responses. Which means that few devices send many Probe Responses. In contrast are the Beacons which are very low.

Figure 5.9 shows the unique MAC addresses per hour i.e. every frame that was captured at the same hour is added up and every MAC address is only counted once for a specific hour. It is also very clear that nearly all MAC addresses use Probe Requests and only a minority of MAC addresses sent out Probe Responses or Beacons.

In the barplot 5.10 all subtypes of management frames which were captured during data collection are shown. The y-axis has a logarithmic scale. The subtypes 4 and 5 are very

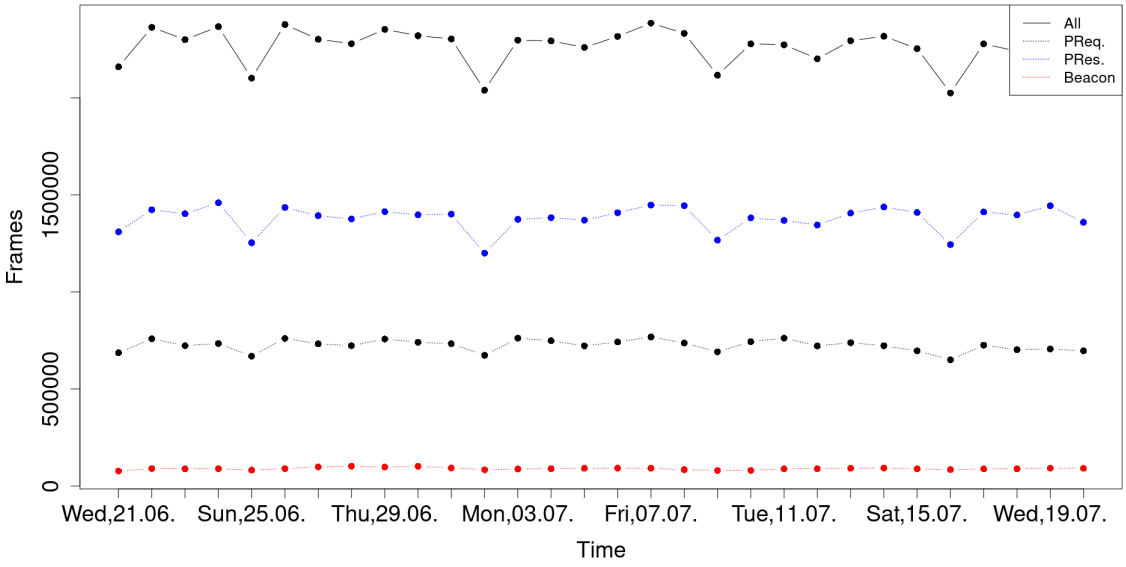

Figure 5.8: Billboard Advertising dataset, frames per day.

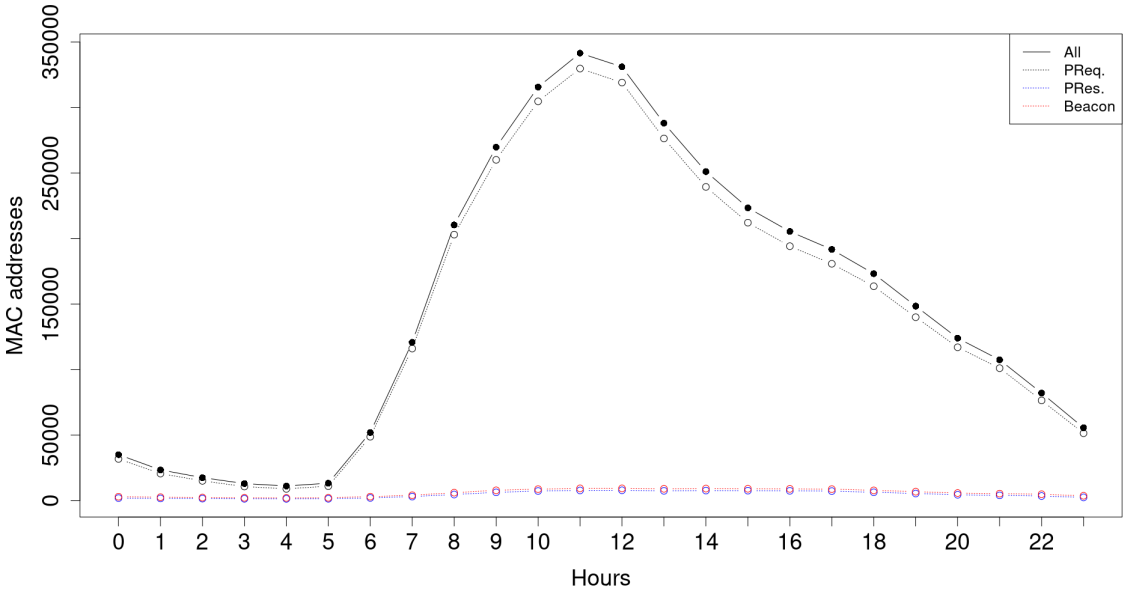

Figure 5.9: Billboard Advertising dataset, unique MAC addresses per hour.

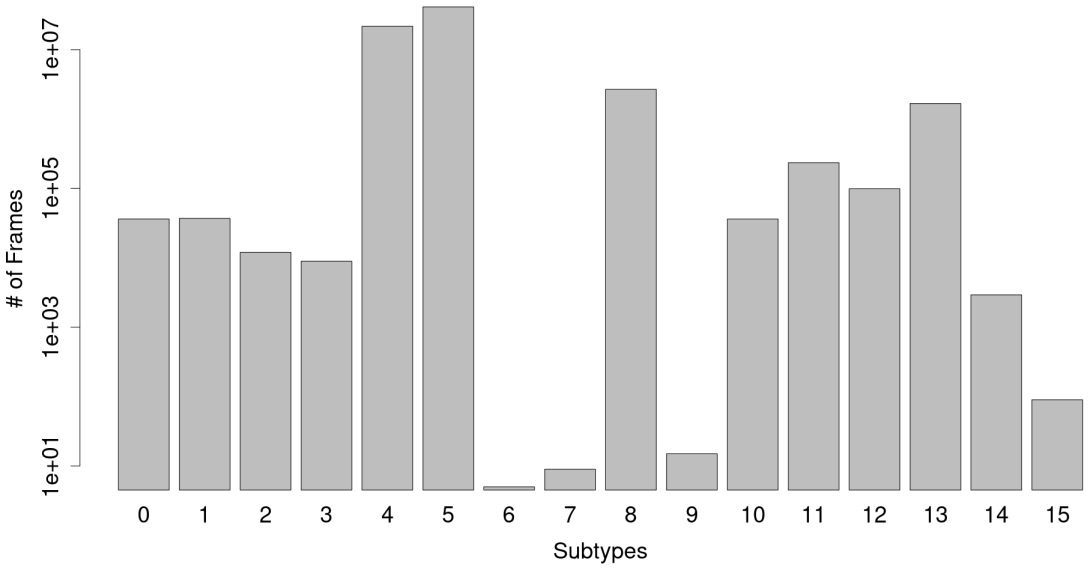

Figure 5.10: Billboard Advertising dataset, different subtypes captured.

common. Subtype 4 is a Probe Request and subtype 5 is a Probe Response. More than 21 million Probe Requests and more than 41 million Probe Responses were captured in this dataset. The next most common subtype is number 8 also called Beacon with approximately 2.6 million captured frames.

Figure 5.11 shows the daily summary of MAC addresses which have the local bit set in their address and also the daily summary of all MAC addresses which sent out at least one Probe Request. If the local bit of an address is set this means that this address is not globally assigned and unique. This leads to the assumption that addresses with the local bit set are used for MAC randomization [VMC+16]. 72.31% of all MAC addresses which used Probe Requests also have the local bit set. In this figure it may not look like 72.31% but this is because of the fact, that every MAC address can occur on multiple days. Therefore, the sum of MAC addresses presented are 3,330,796 and 1,954,956 with LB. This also shows that MAC addresses without the local bit re-occur more often since the total amount of captured MAC addresses which used Probe Requests is 2,499,990 and 1,807,973 with LB.

From the total 2.5 million MAC addresses 134,675 are especially interesting, because they have the local bit set and used directed Probe Requests. From these 134,675 MAC addresses 423,464 frames were captured. In total 1.8 million MAC addresses had the local bit set. About 10% of MAC addresses which have the local bit set, make also use of directed Probe Requests as shown in Figure 5.12. It has to be noted that the ordinate is logarithmic. Also shown are directed Probe Requests which have at least once probed

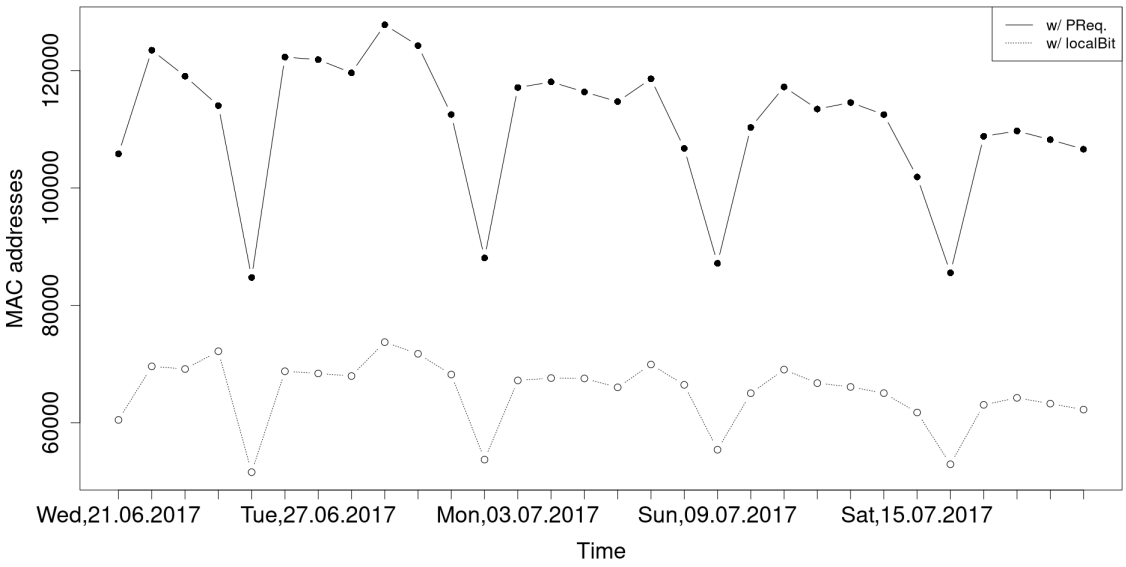

Figure 5.11: Billboard Advertising dataset, daily summary about unique MACs with local bit set.

an SSID starting with "UPC" and have the local bit set.

Figure 5.13 shows how many unique MAC addresses have been captured at how many stations. E.g., if a MAC address has been seen at three different stations it will be added to the third bar, but not to the second. The y-axis has a logarithmic scale. More than 2.1 million MAC addresses have been seen on one station only. Followed by over 180,000 MAC addresses on two different stations and even on all 9 stations 719 MAC addresses were captured. The blue bar is the total of captured MAC addresses and in the red bar there are only MAC addresses considered which set the local bit. Similar data is shown in Figure 5.14 but here the abscissa shows the number of days and not the number of stations. Here it is also clear that MAC addresses with the local bit are seen for multiple days. These figures show in each bar the exact number of seen days or stations, e.g., if a device is shown in the bar number four it is only shown there.

In Table 5.12 some common network names are listed which mostly start with company names of ISPs but also one car company is listed. The asterisk sign in the SSID column means there can be zero or more characters of any kind to be listed in that category. It is clearly visible, that network names which start with "upc" are very common. To check the uniqueness of these SSIDs a sample of 1,000 SSIDs were queried on wigle.net. For every network name the amount of returned locations was recorded. This rendered the results shown in 5.8. Also another major ISP in Vienna has even better results shown in Table 5.9

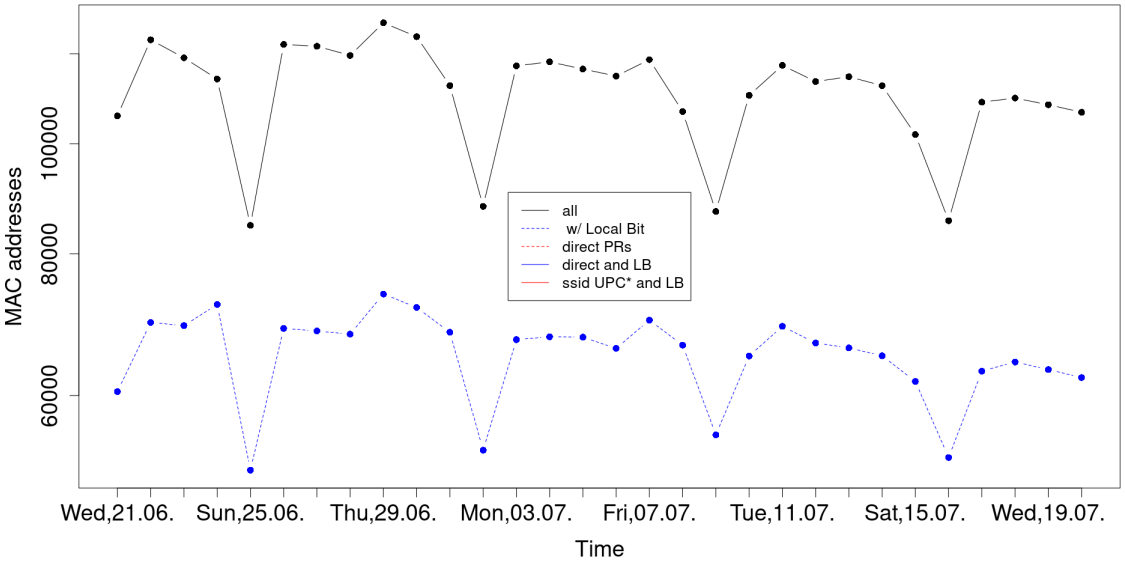

Figure 5.12: Billboard Advertising dataset, daily summary about unique MACs which used Probe Requests.

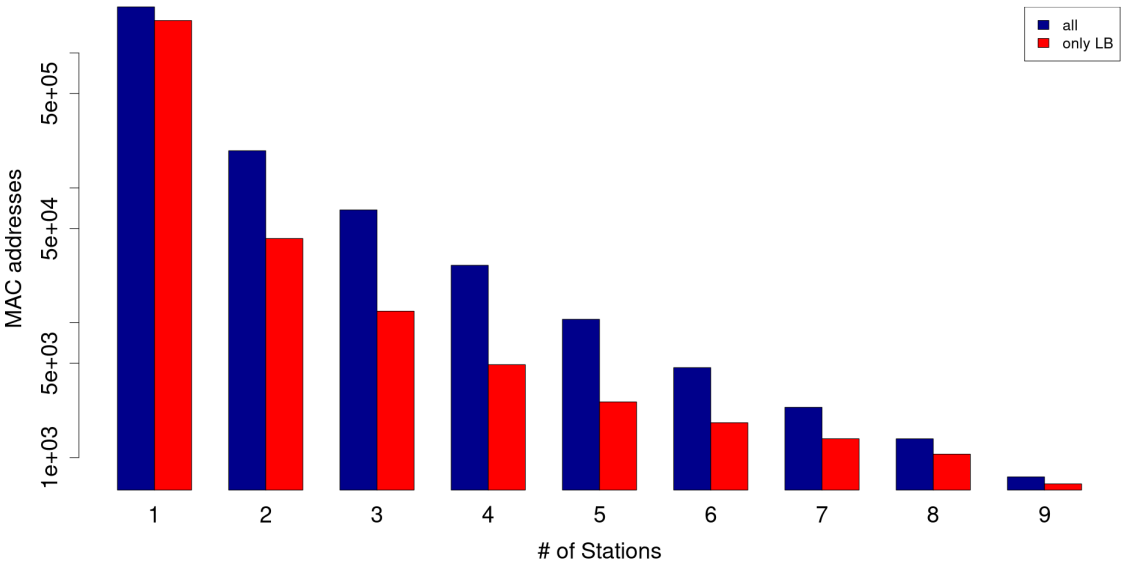

Figure 5.13: Billboard Advertising dataset, unique MAC addresses seen on different stations.

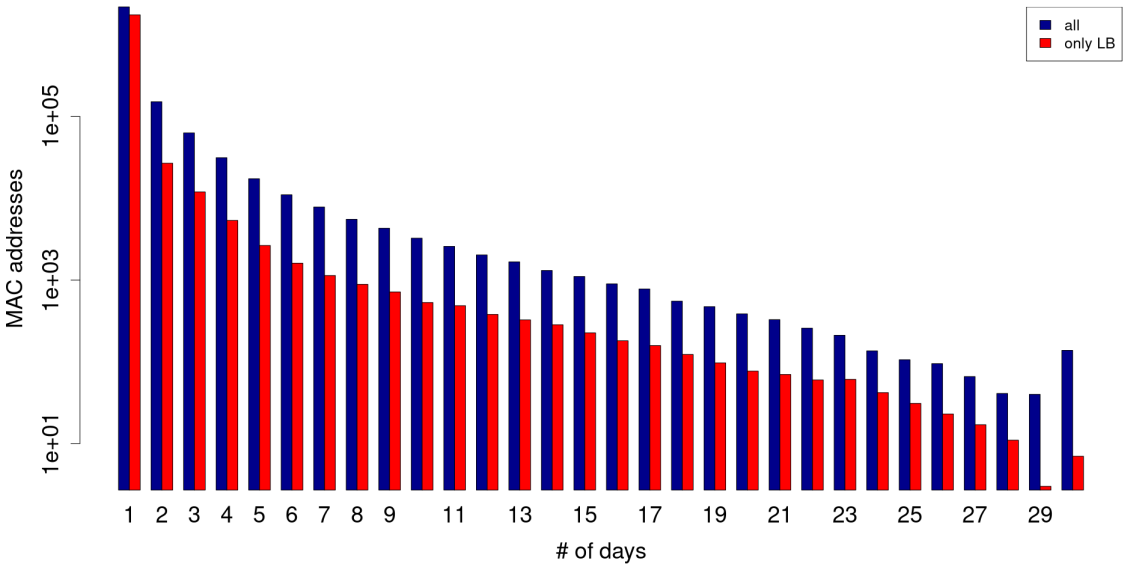

Figure 5.14: Billboard Advertising dataset, unique MAC addresses seen on different days.

Median	Average	Maximum
1	1.54	6

Table 5.8: Billboard Advertising dataset, number of returned results of "upc*" sample.

Median	Average	Maximum
1	1.10	2

Table 5.9: Billboard Advertising dataset, number of returned results of "a1*" sample.

In contrast network names which match the the following pattern: "*free*" were queried against wigle.net. The results are shown in Table 5.10, not surprisingly the queried network names are not unique according to wigle.net. The average is more than 18. The maximum number is hundred, but due to technical limitations. The service returns at most hundred results.

Median	Average	Maximum
5	18.76	100

Table 5.10: Billboard Advertising dataset, number of returned results of "*free*" sample.

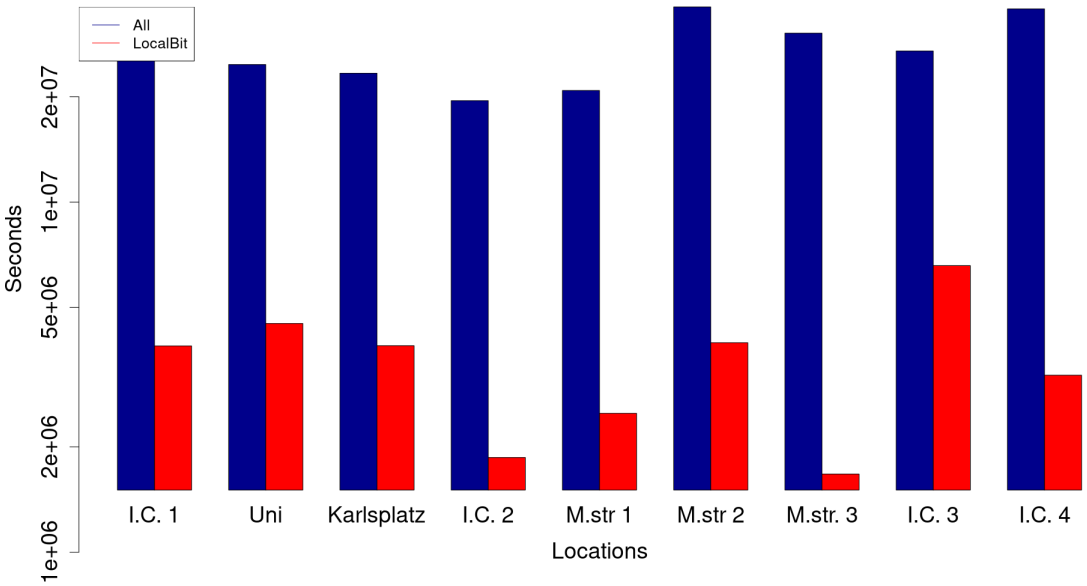

Figure 5.15: Billboard Advertising dataset, captured seconds per location.

The following method was used to find out popular strings with which an SSID starts: All network names of this data collection were grouped by a substring of the network name which started at the first character. The length was initially one character and was continuously expanded until no meaningful grouping(groups smaller than 1,000) was encountered.

In Figure 5.15 the captured seconds per location are shown. A captured second is defined as the absolute time difference between two frames with the same source MAC address. Also the time difference is not allowed to exceed 5 minutes. If only one frame is captured from a source MAC address within a five minute range, this MAC address is not considered in the captured seconds. The blue bar shows all MAC addresses which are considered client MAC addresses. The red bar consists only of client MAC addresses which set the local bit in the MAC address. Many devices set the local bit if they use MAC address randomization [VMC+16]. This is interesting, since these bars are not zero. This means that randomized MAC addresses are used multiple times.

In Table 5.11 unique MAC addresses per station are summed up. If the MAC addresses of all stations are accumulated this leads to a higher amount as in the summary Table 5.16. This is because of the reason that the MAC addresses are only unique per station and can be double counted on different stations i.e. some devices have been seen on more than one station.

In Table 5.13 vendors with the most captured distinct MAC addresses are shown. It is not taken into account that one company could have registered multiple address blocks

Location Name	Amount of MAC addresses
I.C. 1	312,955
Uni Wien	449,063
Karlsplatz	422,375
I.C. 2	186,294
Mariahilferstr. 1	281,317
Mariahilferstr 2	391,639
Mariahilferstr. 3	253,189
I.C. 3	466,047
I.C. 4	238,300

Table 5.11: Billboard Advertising dataset, MAC addresses per station.

SSID	Amount
iphone*	1,661
upc*	17,265
a1*	7,856
wlan*	5,288
wifi*	3,662
tmobile*	1,621
tp-link*	3,357
3webcube*	1,821
audi*	1,702
pbs*	3,115
Android*	1,719
huawei*	3,502
hotel*	3,821
glocalNet*	2,340
free	6456

Table 5.12: Billboard Advertising dataset, categorized networks by name.

Vendor Name	Captured MAC addresses
not assigned	1,807,961
Samsung Electronics Co.,Ltd	169,233
Apple, Inc.	135,347
SAMSUNG ELECTRO-MECHANICS(THAILAND)	85,062
HUAWEI TECHNOLOGIES CO.,LTD	59,898
Motorola Mobility LLC, a Lenovo Company	38,798
Sony Mobile Communications AB	31,740
LG Electronics (Mobile Communications)	30,842
Murata Manufacturing Co., Ltd.	25,729
HTC Corporation	12,980

Table 5.13: Billboard Advertising dataset, the ten most popular vendors [oui].

Vendor Name	Captured MAC addresses
Google, Inc.	230,592
IEEE 802.1 Working Group	2
Cirrus Data Solutions, Inc	2
Microsoft Corporation	2

Table 5.14: Billboard advertising company, the most popular CIDs [oui].

	# of MAC addresses	Related to other MAC addresses	% of total
Total	17,726	2,676	15.10%
Only LB	3,842	2,080	54.14%

Table 5.15: Billboard advertising company, subset of directed Probe Requests.

under different subsidiaries. All of the 1.8 million "not assigned" MAC addresses have the local bit set. Therefore, another query against the CID [cid] was conducted shown in 5.14. According to this dataset Google, Inc. is the only company which has a CID registered and uses it. It is assumed that this CIDs are mainly used for randomized MAC addresses. Interestingly enough Apple[1] as a major mobile operation system provider is not represented in this table. Apple does not even have a registered CID according to Reference [cid]. Samsung, another major manufacturer, has registered a CID.

In Table 5.15 a subset of all MAC addresses were selected which probed at least one SSID starting with "upc" or "UPC", was at least six characters long and was not equal to the string "UPC Wi-Free". These criteria led to the selection of 17,726 MAC addresses from which 15.10% could be related to at least one other MAC address. For MAC

[1] https://www.apple.com

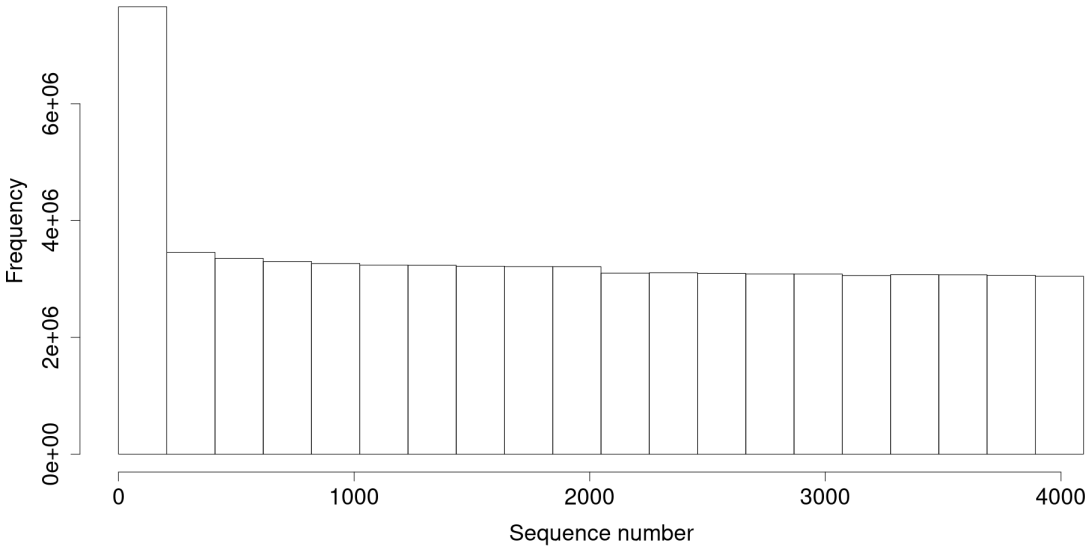

Figure 5.16: Billboard Advertising dataset, distribution of sequence numbers.

addresses, which had the local bit set a related MAC address could be found for 54.14%. The relation was solely built because of their unique probing pattern of their preferred network list. Network names starting with "UPC" were chosen, because they occur quite often in the dataset and according to Table 5.8 the average number of mapped geographic locations is 1.54 which is very low and near the ideal of 1.

It can be seen in Figure 5.16 that the distribution of MAC addresses is not equally distributed among the possible values of 0 to 4095, but the values from 0 to 200 are more likely to occur. In Figure 5.17 there is a detailed view of the sequence numbers up to 200. There it can be seen that the sequence numbers from zero to ten are the most likeliest sequence numbers that occurred during this data collection.

The Figure 5.18 shows the distribution of how long a specific MAC address was seen during the whole data collection. The highest and most common value is around 100 seconds. This means that those MAC addresses have been seen in total for 100 seconds. In contrast there is Figure 5.19 which shows how long an individual visit lasted.

The re-occurrence of individual MAC addresses is shown in 5.20. Here the number of visits versus the numbers of different days seen during the data collection is compared.

5.2.1 Summary

In Table 5.16 a summary about how many data was captured is presented. It gives a brief overview about the size of the data collection. As easily discovered, the sum of the

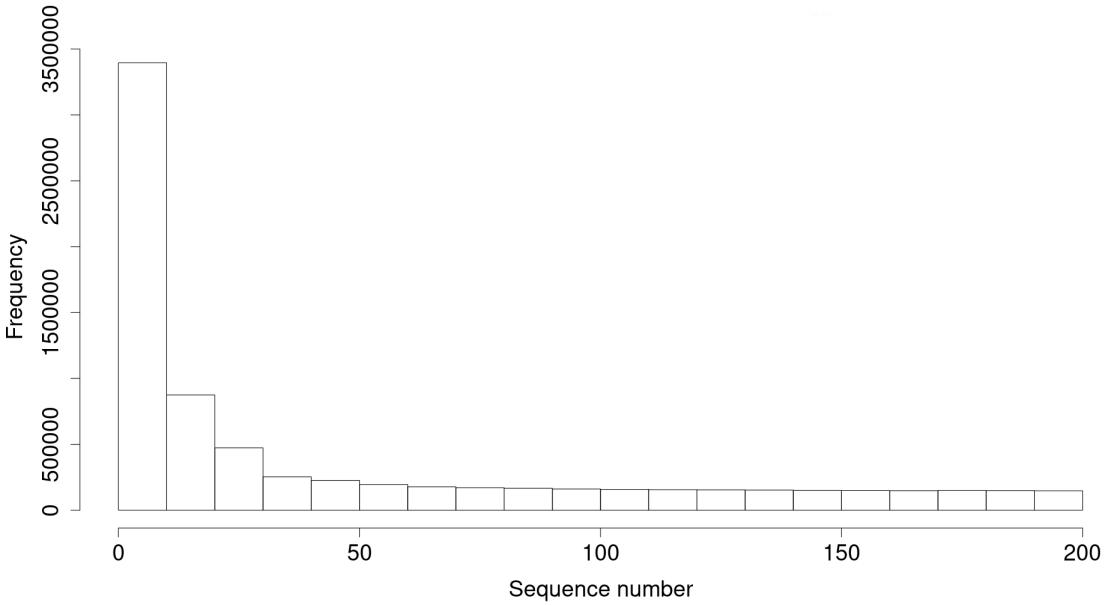

Figure 5.17: Billboard Advertising dataset, distribution of sequence numbers only from 0-200.

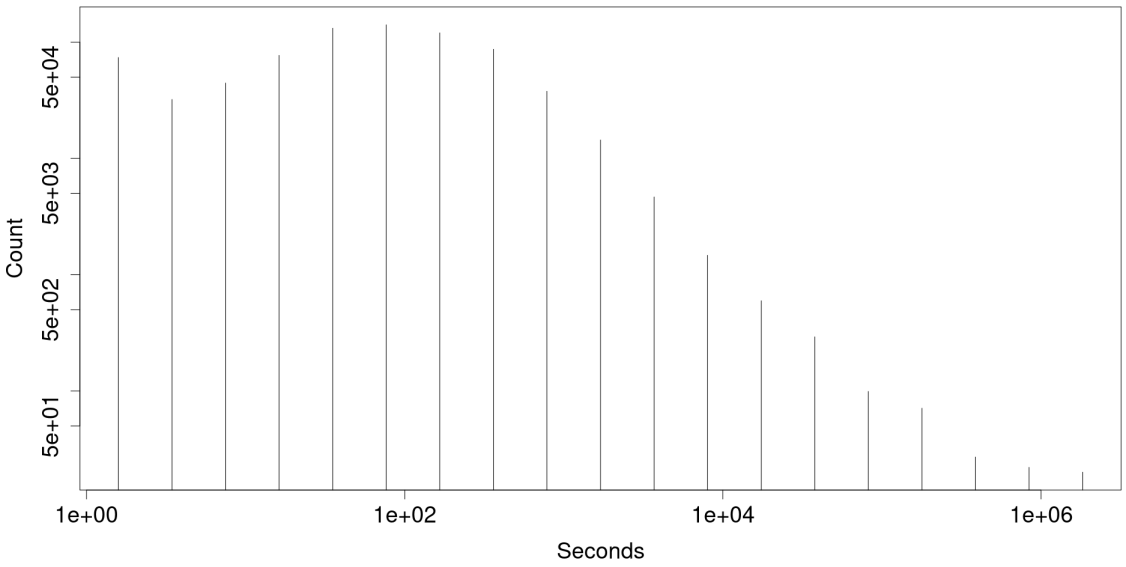

Figure 5.18: Billboard Advertising dataset, distribution of summed up seconds per MAC address.

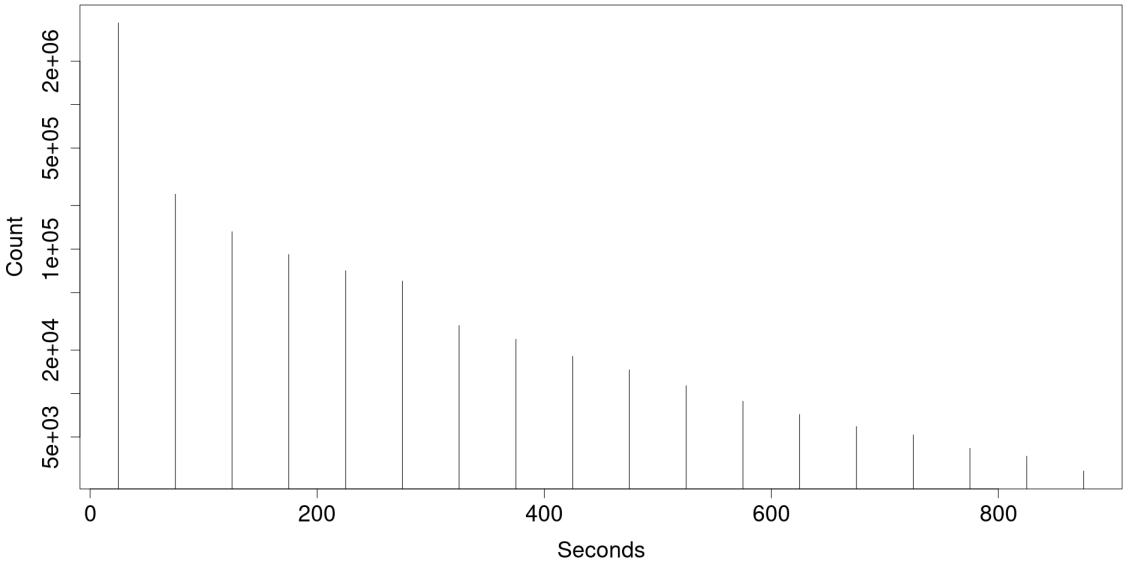

Figure 5.19: Billboard Advertising dataset, distribution of visit duration.

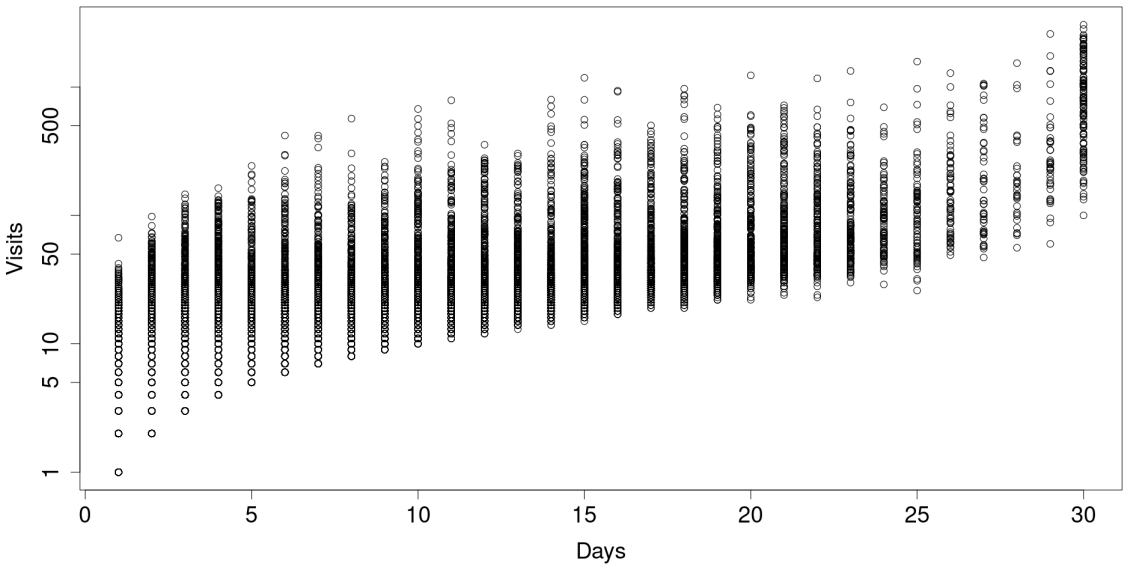

Figure 5.20: Billboard Advertising dataset, re-occurrence of individual MAC addresses.

	# of frames	# of MAC addresses	MACs with local bit
Probe Requests	21,729,816	2,499,990	1,807,973
Probe Responses	41,362,417	37,749	10,163
Beacons	2,682,664	45,346	11,927
Other subtypes	2,146,066	74,814	5,820
Combined	**67,920,963**	**2,555,186**	**1,825,410**

Table 5.16: Summary of Billboard Advertising data collection.

	# of frames	# of MAC addresses	MACs with local bit
Probe Requests	31.99%	97.83%	70.76%
Probe Responses	60.90%	1.48%	0.40%
Beacons	3.95%	1.77%	0.46%
Other subtypes	3.16%	3%	0.32%

Table 5.17: Summary of Billboard Advertising data collection with percentage.

first three rows from the second column is higher than the specified total in the last row. This is because of the fact that there can be double counts in the different rows. E.g., Beacons and Probe Responses are sent out from the same AP.

In Table 5.17 the same data is presented as in Table 5.16 but with percentage values instead of absolute values. In this table it is nicely shown that there are very little devices which send out lots of Probe Responses. However, the majority of the devices are responsible for sending out Probe Requests. Since client devices send out Probe Requests, it can be argued that approximately 97% of the captured MAC addresses in this dataset are from client devices.

An interesting measure are the Probe Requests per minute. The distribution of the average Probe Requests per minute can be seen in Figure 5.21. The overall average is 9.8 Probe Requests per minute and the maximum probes per minute are 629 from a MAC address which is registered to "Samsung Electronics Co.,Ltd". In this calculation only MAC addresses which have at least two Probe Requests in a minute are considered. In total there are 77,483 MAC addresses, which meet this criteria.

5.3 Comparison Between Supermarket and Billboard Advertising Dataset

In Table 5.18 the two datasets are compared and it can be seen that the Billboard Advertising dataset lasted approximately the double of the time as the Supermarket dataset. Also interesting are the average frames per minute captured in the datasets. Here it is shown that the Billboard Advertising dataset has a slightly higher average. Since

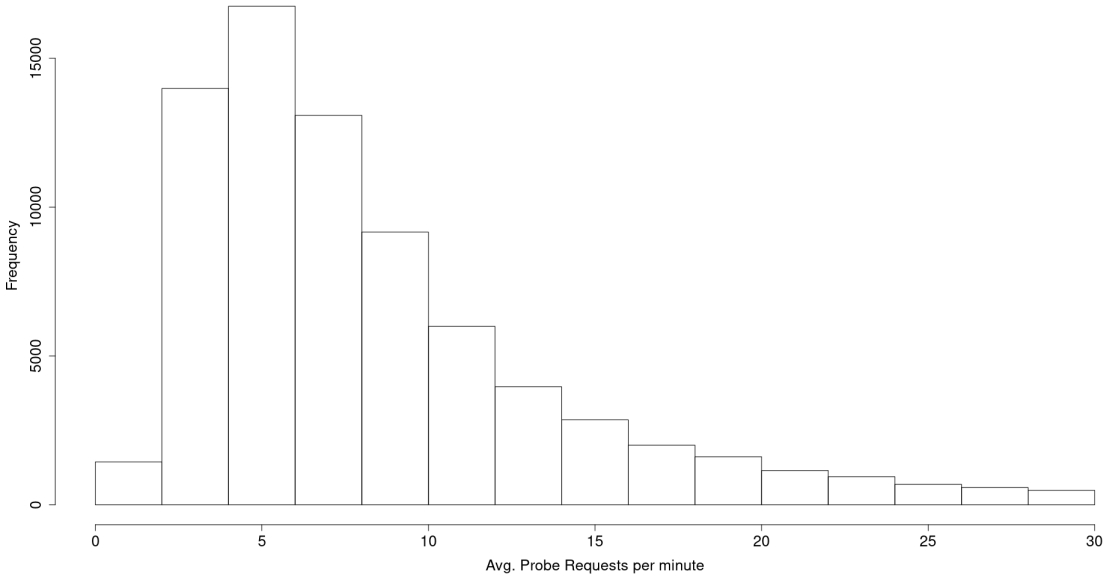

Figure 5.21: Billboard Advertising dataset, average Probe Requests per minute.

	Billboard Advertising dataset	**Supermarket dataset**
Duration	30 days	16 days
# of frames	67.9 million	24.4 million
Avg frames/minute	1572	1210
# of Probe Requests	21.7 million	7.5 million
Avg PR/minute	503	369
# of MAC addresses	2.5 million	~700 000

Table 5.18: Comparison of Billboard Advertising dataset and Supermarket dataset.

the supermarkets are usually closed on Sundays in Vienna and the Billboard Advertising dataset also captured pedestrians on Sundays it is clear why the average is lower.

In Table 5.19 the size of all distinct MAC addresses are shown in Megabyte and also how well they compress in comparison to just random files. Therefore, all MAC addresses were put together into a long string, all duplications and repeating characters like colons were removed. It can be seen that the MAC addresses compress around 38.5% better than just purely randomly generated strings of the same size and character set.

The average amount of Probe Requests per MAC address is shown in Table 5.20. If only MAC addresses with the local bit are considered, then the average of Probe Requests is much lower.

	Original size	**MACs compressed**	**Random compressed**	
Advertising	30MB	9.8MB	Avg 16.4MB	59.68%
Supermarket	8.4MB	2.8MB	Avg 4.5MB	63.16%

Table 5.19: Summary of entropy calculation.

	Billboard Advertising dataset	**Supermarket dataset**
all	8.69	10.86
only LB	2.75	3.98

Table 5.20: Average PRs per recorded MAC address.

	Billboard Advertising dataset	**Supermarket dataset**
total MAC addresses	77,483	31,245
avg PRs	9.8	12.18
only LB	6.09	6.51
maximum	629	835

Table 5.21: Average Probe Requests per minute.

Comparing the average Probe Requests per minute shown in Table 5.21 illustrates that the average is more than 2 requests lower in the Billboard Advertising dataset. The average for both datasets is therefore 10.99 requests per minute. MAC addresses which sent more than 30 requests per minute represent 5.7% and 8.4% of the total subset for the Billboard Advertising dataset and the supermarket dataset respectively.

CHAPTER 6

Discussion

In this chapter the research questions are discussed, taking into account the results from the data collections.

The following research questions shall be examined in this chapter:

- What is the entropy of MAC addresses collected in the field studies?
- What part of the collected data in the field studies can be used to identify the re-occurrence of users over multiple days?
- What part of the data, collected in the field studies, could be used to draw conclusions if a MAC address is randomized or not?

The questions will be evaluated in the following subsequent sections.

6.1 Entropy of MAC Addresses

A MAC address is 48bit long and divided into 6 octets usually separated with a colon. A detailed description can be found in Section 2.

The total number of possible states for a 48 bit long MAC address is 2^{48}. Therefore, the entropy for a MAC address is $S = \log_2(2^{48}) = 48bit$ [Sha48].

This is the reason why MAC addresses are perfect for identifying a device uniquely, because with 48 bit of entropy it is nearly impossible to encounter a device, which has the same MAC address. Also the major purpose of a MAC address is the ability to uniquely identify a device on the data-link or second layer in the ISO/OSI reference model. To circumvent collisions the IEEE Registration Authority [ieeb] regulates the MAC address range. Any individual can reserve a MAC address range.

All MAC addresses from the whole data collection have therefore a memory consumption of $48bit * 2.5million$ for the Billboard Advertising dataset and $48bit * 697,588$ for the Supermarket dataset which equals 122,648,928bit and 33,484,224bit of memory respectively.

Since this is a rather trivial calculation, data compression is used as a measure for the entropy to get a better picture about the internal structure of MAC addresses. According to Nelson and Gailly [NG96]: "Entropy fits with data compression in its determination of how many bits of information are actually present in a message". Another interesting measure is the Kolmogorov Complexity [Kol98], which is defined as the smallest program that can describe a string. In Galatolo et al. [GHR10] a relation between Kolmogorov Complexity and entropy is established. Moreover, in Faloutsos and Megalooikonomou [FM07] it is stated that it is undecidable what the smallest program really is, because of the halting problem, but compression functions as an upper bound for Kolmogorov Complexity.

Hence, entropy and data compression is closely related, a data compression approach was used to find out more about the structure and entropy of the MAC address data. Therefore, all MAC addresses of the data collection were put together into one big string. In this string all duplicates and colons were removed, which are commonly used to display a MAC address. So only a string with the character set 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f was left.

To proof that there is more structure in MAC addresses than just in randomly generated characters, there were one hundred random strings generated with the same character set and size.

Now there are 101 strings, 100 of which are randomly generated. All of these strings were compressed with zlib compression library from Google [zop].

This was done for each data collection and lead to the following results shown in Table 5.19. Since the results suggests that MAC addresses compress better than just purely random strings, it can be calculated that MAC addresses only need around 61.42% of the size than random files, if the average of both datasets is taken. This leads to the assumption that for a large data collection which stores a massive amount of MAC addresses it is not necessary to use 48bit to calculate the memory consumption but rather only $48bit * 61.42\% = 30bit$ can be taken for a single MAC address on average.

Now it is possible to calculate the memory consumption of the collected MAC addresses of the dataset, since it can be assumed that one MAC address needs 30bit. This leads to the following calculations for the Billboard Advertising dataset $30bit * 2.5million = 76,655,580bit$ and for the Supermarket dataset $30bit * 697588 = 20,927,640bit$.

As mentioned before the IEEE Registration Authority [ieeb] assigns MAC address ranges to an interested company or individual. Therefore, this company has to pay a yearly fee and is publicly listed in the OUI list [oui]. Usually this registered range fixes the first 24bit or first 3 bytes of a MAC address. The last 3 bytes can be arbitrarily assigned

to network devices, but shall only be assigned once. So in general every MAC address starting with the same 24bit belong to the same company. This makes it easy to find out if a captured MAC address belongs to an iPhone or Android device, if the MAC address is not randomized. The amount of randomized MAC addresses in the datasets are both above 70%, which means that within these MAC addresses the local bit is set and they are not assigned to an OUI. This also concludes that MAC randomization is widely used since it is already implemented by major operating systems like Linux (including Android [and]), iOS [ios] and Windows [VMC+16]. The local bit is the seventh bit in the first byte of a MAC address. Therefore, the OUI is not applicable anymore because the first 24bit are changed. Furthermore, the IEEE Registration Authority [ieeb] does not assign OUIs with the local bit set.

It can be seen in Table 5.13, that more than 1.8 million MAC addresses out of 2.5 million could not be assigned by using the current OUI [oui] list of the IEEE Registration Authority. Out of these 1.8 million unassigned MAC addresses, only a small amount of around 230,000 could be assigned to a CID, which is shown in Table 5.14. Hence, CIDs are only assigned for MAC address ranges where the local bit is set. This leads to the conclusion that only Android devices use a MAC address with a registered CID if they randomize their MAC addresses.

6.2 Identify Re-Occurrence

In general every Probe Request sent from a Wi-Fi device contains a MAC address which uniquely identifies the transmitter. This address can therefore be used to identify re-occurrences of devices over the lifetime of devices, which usually spans over several years. This is also shown in Figure 5.20, where the re-occurrence of MAC addresses is shown with the number of times a MAC address has been seen, partitioned into the number of different days it was seen. Since this is a very effective method to circumvent privacy, it has been suggested to use MAC randomization [GG05]. In theory MAC randomization should disable the possibility of traceability through MAC addresses. But e.g., Vanhoef et al. [VMC+16] or Freudiger [Fre15] suggest otherwise. Also in the above mentioned datasets it was noticed, that MAC addresses which have the local bit set occur more than once. In general many MAC randomization mechanisms use MAC addresses with the local bit set. This leads to the conclusion that by coincidence some MAC addresses are the same which is very unlikely or in a more likely scenario MAC address randomization does not change the randomized MAC address for every frame. The latter is more plausible. Also Figure 5.13 supports the fact that potentially randomized MAC addresses are used multiple times and also for longer periods. Some addresses are even seen on all nine research stations. It can be concluded that MAC randomization techniques possibly do not change the randomized MAC address on a regular basis like at least every few minutes. It could be argued that some devices may used the path between the stations as a regular commute. Therefore, MAC addresses could be seen on multiple stations but within an hour. To disprove this argument, Figure 5.14 shows that some local bit MAC addresses are seen over the whole research period. It is also shown that there are

more MAC addresses with the local bit set which are seen on 30 days than on 29 days. It is assumed that this is due to the fact that some devices were always near the sniffing stations.

If a randomized MAC address is not changed frequently this could make it possible to find a pattern when a randomized MAC address occurs, e.g., every workday at noon for five minutes. If these patterns are accurate, it could re-identify a changed MAC address by means of the generated pattern.

For directed Probe Request frames the network name can be used to uniquely identify a device. If a device sends out directed Probe Requests it sends out a Probe Request for every network that is in the preferred network list. This combination of Probe Requests can then be used to identify devices which also request these network names. It can then be concluded that this devices could be the same. Also if they use MAC randomization and therefore have a different MAC address. Of course this technique works better if lots of different and unique network names are requested. Unique network names in this regard are MAC addresses which can be mapped to one geographic location with a geo-location mapping service.

If the client device uses directed Probe Requests it is especially effective to track this device even if the device MAC address is randomized.

Directed Probe Requests can also be used to categorize pedestrians or customers. For example if a device probes "IKEA WiFi" it can be concluded that the user is a customer of IKEA and therefore buys IKEA products. This knowledge can be used by an advertising company. It can now show an IKEA ad, if the device passes a billboard. The possibilities are of course endless, a competition can also show an ad, why it is not good to buy IKEA products.

6.3 MAC Address Randomization

In the paper [VMC+16] it is mentioned that MAC addresses with the local bit set are randomized MAC addresses or special assigned MAC addresses, like in virtual environments. As shown in Figure 5.15 or Figure 5.6 which considers only MAC addresses which sent at least one frame with the subtype 4, it can be concluded that the MAC addresses within this figure are randomized if they have the local bit set. The red bar only considers MAC addresses with the local bit set. In both datasets many addresses which are assumed randomized appear multiple times.

In the evaluation of the dataset it was detected that some SSIDs start with "iPhone of" followed often by the full name of a possible friend, relative or other acquaintance. This can potentially be very privacy invasive, depending if this network name is gathered from a Probe Request or a Beacon frame it has the potential to identify a person uniquely over the SSID. If the network name was revealed in a directed Probe Request it could reveal a close relationship with the person named in the SSID. Since a network name starting with "iPhone of" or similar is usually a network name of a hotspot from an iPhone device.

As the research data suggest many names used are plausibly or well known names in Austria or Germany, e.g., Albrecht, Daniel or Max to just name a few.

From Section 6.2 it is known that SSIDs starting with "UPC" are considered high value because of their uniqueness. Therefore, a subset of MAC addresses was selected which directly probed SSIDs according to the following criteria:

- starting with the string "UPC" or "upc"
- at least 6 chars long
- not equal to "UPC Wi-Free"

The MAC addresses selected were then grouped by their SSIDs and so called SSID sets were generated. Those SSID sets were than used to find other MAC addresses which probed the same SSIDs. If one SSID set was used from multiple MAC addresses then it was considered that those MAC addresses are related and the frames were sent from the same device. The results from both datasets are presented in Table 5.15 and Table 5.6. These results show for MAC addresses, which have the local bit set, that over 54% and 68% for the Billboard Advertising and Supermarket dataset can be related to another MAC address. This is a very interesting finding, because MAC addresses with the local bit set can be assumed to be randomized. This finding shows also, that it is feasible to make relations between randomized MAC addresses if directed Probe Requests are used. It can also be concluded that these MAC addresses use MAC randomization with a very high likelihood. Otherwise, there would not be a unique SSID set from two different MAC addresses.

Every Probe Request contains a sequence number. This sequence number is incremented on every request and can therefore be used to re-identify a device even if the MAC address has changed, as mentioned in the paper [VMC+16]. Since the sequence number is 12bit long and can therefore store $2^{12} = 4096$ different numbers. This is not enough to identify millions of devices over a longer period, but for a small time period it could be possible to draw conclusions if two MAC addresses belong to the same device. To test this hypothesis a program was developed which tested the dataset for any meaningful connections between MAC addresses. There are two main parameters which controlled the outcome of the algorithm:

- Maximum distance between sequence numbers
- Maximum timespan between the frames

Furthermore, the frames had to occur at the same sniffing station to have a relation.

After analyzing the data thoroughly it was concluded that the sequence numbers on their own do not have enough entropy to build relations between different MAC addresses.

Since the average frames per minute captured in the dataset was ~1500 and ~1200 the likelihood that a match is just a coincidence is too high. The sequence number similarity check was also performed on the UPC-subset which was used for the more reliable approach with directed Probe Requests, but there were only 6 matches out of 106 which met the timespan criteria of one minute. The maximum increment of the sequence number was from zero to five.

It can also be seen in the Figures 5.16 and 5.17 that the sequence numbers from zero to ten are the most likeliest. Therefore, it can be assumed that the sequence number is not incremented after every request till the highest value is reached, but it is reset after a few requests.

6.4 Limitations

The conducted data collections were limited in time and equipment. For the data collection multiple Raspberry Pi boards with an external Wi-Fi dongle were used. With these components it was possible to capture frames in the surrounding environment from the 2.4 Ghz range. The also common 5Ghz frequency was not considered in the data collection, because the Wi-Fi dongles were not able to capture on this frequency.

It also did not include a big scale distribution of sniffing devices which could have led to much more precise data.

Since only layer 2 management frames have been captured during the data collection. It was not possible to analyze the attack vectors described in Bloessl et al. [BSDE15] and Vanhoef et al. [VMC+16], which use a predictable layer 1 scrambler seed.

The recorded data is not analyzed in real time. The data was recorded and analyzed after the data collection was finished. If the Raspberry Pi had been equipped with a Sim Card, real time access would have been possible. The captured amount of data per station and per day was around 50 MB, which could have been easily handled with current mobile network connections. Alternatively the Raspberry Pi could have used the Wi-Fi of the supermarket to get an up-link.

Big companies like Google have a database of Wi-Fi networks around the world with a mapping to the location. [goo] This huge databases can link nearly every SSID to a geographic location. At best to only one location, if the SSID is a unique string. The problem is the access to these databases is restricted and limited to a few requests per day. Google does not even provide a public API to translate SSID names to GPS locations. Another similar database wigle.net only allows a few hundred requests a day, which was also not sufficient to resolve every SSID captured in these datasets.

6.5 Future Work

In the data collections around ten devices were used each. The Supermarket data collection took place in ten supermarkets. Therefore, there was only one sniffing station

per supermarket at the point of sale. This means that a device was not in range of the capturing station during the whole visit. Of course, it would enhance the results to place multiple stations into one supermarket to be able to give a more precise location of the captured devices.

Furthermore, if the data collection lasts for a year or longer and many capturing devices are used within one supermarket or other store, it would be possible to extract even more precise tracking data. Maybe it would even be possible to just detect a device because of its movement pattern.

Another challenge would be to look into the physical layer. Hence, not only capturing the complete frames but also the header of the physical layer. For example to be able to examine the preamble.

Unlimited access to a database with current SSID mappings to geographic locations would be interesting, to be able to automate and improve the process of SSID assessment, which would have made it possible to make more conclusions within the dataset. Due to the fact that a SSID which only has one geographic location is only used by few people and has therefore a higher value for algorithms which are trying to find relations between MAC addresses.

During the evaluation only MAC addresses, which made directed Probe Requests to SSIDs which start with "UPC" were checked if they can be related to another mac address according to their probing pattern. This algorithm was not used for the whole dataset because some threshold needs to be developed for the number of SSIDs within a SSID set. I.e. there needs to be a threshold on how many different SSIDs need to be probed to conclude that these MAC addresses are related. This threshold is not relevant for MAC addresses which probed "UPC" SSIDs, because these SSIDs are considered to be unique and occur therefore very rarely from different devices.

As already briefly mentioned in the Section 6.2, an advertising company can categorize directed Probe Requests according to well known Wi-Fi names like "IKEA WiFi" to find out general interests about a group of people. This knowledge can than be used to display targeted ads depending on which device is currently passing a billboard ad.

6.6 Ethical Considerations

During the data collection no information was collected which can directly identify a person. Furthermore, if by accident personal information was gathered it was removed immediately on discovery. The collected dataset consists only of public information from websites and the publicly available data on the 2.4GHz spectrum which is generally used by Wi-Fi enabled devices. All used devices were only listening on the specified frequency. The purpose of the data collection was for pure scientific research and will not be used for any commercial projects. Moreover, the collected data will not be given away to third parties without anonymizing the dataset, especially the MAC addresses. Also the

duration of the data collections was not long enough to generate any valuable information, which can be clearly assigned to a real person.

CHAPTER 7

Conclusion

In this master thesis an extensive data collection in the inner districts of Vienna was realized with two independent partner companies. The collected data was mostly captured from client devices which used Probe Requests to discover a nearby AP.

For data collection a series of Raspberry Pis was set up and configured to allow a fail safe logging of all Wi-Fi management frames. An easy on site setup was important for resistance against short power losses. Therefore, a RTC was used to keep time even if no internet connection and power was available.

The collected data was then converted and stored in an SQL database to be able to run analyzes over the large amount of captured frames.

According to [VMC+16] MAC randomization is implemented in every major operating system like Linux, iOS and Windows. This is also seen in the datasets since over 70% of MAC addresses have the local bit set. But the CID is not widely used for randomized MAC addresses. Only Google Inc. used a substantial amount of registered CID in the dataset.

Furthermore, the entropy was calculated for the MAC address in the datasets. Since MAC addresses compress quite well, it was calculated that only 30bit are needed to store a MAC address within a large pool of MAC addresses.

One aspect which was clear from the beginning concerned the identifier which is used within IEEE 802.11 frames. This identifier also called MAC address made it very easy to track the occurrence of a device over multiple days.

During the evaluation of the dataset it was revealed that the majority of MAC addresses are randomized, but often MAC addresses are re-used and not re-randomized after every request. Therefore, randomized MAC addresses which are re-used over a longer time period can also make a device trackable.

Also remarkable are devices which use MAC randomization and use directed Probe Requests. This can render the MAC randomization completely useless. In the worst case directed Probe Requests can even reveal were the owner of a device lives. It was shown that specific Probe Requests can lead to a unique geographic location.

List of Figures

List of Tables

Glossary

IEEE The Institute of Electrical and Electronics Engineers is an association of professional engineers around the world with focus on educational and technical advancement. 1, 3

Java Is a object oriented programming language. https://www.oracle.com/java/. 27

R A command line program commonly used in statistics. https://www.r-project.org/. 25, 26

Acronyms

A Ampere. 19

AP Access Point. 1, 2, 8, 11, 13, 15, 25, 59

ASCII American Standard Code for Information Interchange. 5

BSS Basic Service Set. 8, 11

BSSID Basic Service Set Identifier. 8, 11

CID Company ID. 15, 55, 59

CSV Comma-separated values. 23, 26

DPR Directed Probe Requests. 6

ESS Extended Service Set. 11

GDPR General Data Protection Regulation. 1

Ghz Gigahertz. 3

LLC Logical Link Control. 6

NIC Network Interface Card. 15

OUI Organizationally unique identifier. 15, 54, 55

P2P Peer to Peer. 12

PR Probe Request. 6, 13, 50

RF Radio Frequency. 2

RSU Roadside Units. 14

RTC Real Time Clock. 18, 19, 59

SQL Structured Query Language. 23, 59

SSID Service Set Identifier. 1, 4, 22

TLS Transport Layer Security. 5

WLAN Wireless LAN. 8

Bibliography

[80214] IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture. *IEEE Std 802-2014 (Revision to IEEE Std 802-2001)*, pages 1–74, June 2014.

[80216] IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)*, pages 1–3534, Dec 2016.

[and] Android 6 Changelog. `https://developer.android.com/about/versions/marshmallow/android-6.0-changes`. Accessed: 2018-12-09.

[BSDE15] B. Bloessl, C. Sommer, F. Dressier, and D. Eckhoff. The scrambler attack: A robust physical layer attack on location privacy in vehicular networks. In *2015 International Conference on Computing, Networking and Communications (ICNC)*, pages 395–400, Feb 2015.

[BZO15] Carlos J Bernardos, Juan Carlos Zúñiga, and Piers O'Hanlon. Wi-Fi internet connectivity and privacy: hiding your tracks on the wireless internet. In *Standards for Communications and Networking (CSCN), 2015 IEEE Conference on*, pages 193–198. IEEE, 2015.

[Cea12] Patrick Ciccarelli and Christina Faulkner et al. *Introduction to Networking Basics*. Wiley, 2 edition, 2012.

[Che13] Lei Chen. *Wireless network security*. Springer, 2013.

[cid] CID list from IEEE. `https://standards.ieee.org/develop/regauth/cid/cid.csv`. Accessed: 2018-03-04.

[CKB14] Mathieu Cunche, Mohamed-Ali Kaafar, and Roksana Boreli. Linking wireless devices using information contained in Wi-Fi probe requests. *Pervasive and Mobile Computing*, 11:56–69, 2014.

[DYPL08] Loh Chin Choong Desmond, Cho Chia Yuan, Tan Chung Pheng, and Ri Seng Lee. Identifying unique devices through wireless fingerprinting. In *Proceedings of the first ACM conference on Wireless network security*, pages 46–55. ACM, 2008.

[euG] GDPR - General conditions for imposing administrative fines. `http://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32016R0679#d1e6226-1-1`. Accessed: 2018-03-06.

[eur] Individuals using mobile devices to access the internet on the move. `https://ec.europa.eu/eurostat/tgm/graph.do?tab=graph&plugin=1&language=en&pcode=tin00083`. Accessed: 2018-12-07.

[FM07] Christos Faloutsos and Vasileios Megalooikonomou. On data mining, compression, and kolmogorov complexity. *Data mining and knowledge discovery*, 15(1):3–20, 2007.

[Fre15] Julien Freudiger. How Talkative is Your Mobile Device?: An Experimental Study of Wi-Fi Probe Requests. In *Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, WiSec '15, pages 8:1–8:6, New York, NY, USA, 2015. ACM.

[Gas] Matthew Gast. *802.11 wireless networks: The definitive guide*.

[GG05] Marco Gruteser and Dirk Grunwald. Enhancing location privacy in wireless LAN through disposable interface identifiers: a quantitative analysis. *Mobile Networks and Applications*, 10(3):315–325, 2005.

[GHR10] Stefano Galatolo, Mathieu Hoyrup, and Cristóbal Rojas. Effective symbolic dynamics, random points, statistical behavior, complexity and entropy. *Information and Computation*, 208(1):23–41, 2010.

[goo] Google's Wi-Fi Database May Know Your Router's Physical Location. `https://www.huffingtonpost.com/2011/04/25/android-map-reveals-router-location_n_853214.html`. Accessed: 2018-03-05.

[HH83] Douglas V Hall and Douglas V Hall. *Microprocessors and digital systems*. McGraw-Hill, 1983.

[HS01] Gurdeep S Hura and Mukesh Singhal. *Data and computer communications: networking and internetworking*. CRC Press, 2001.

[ieea] CID - IEEE Standards Association. `http://standards.ieee.org/develop/regauth/cid/index.html`. Accessed: 2018-01-21.

[ieeb] IEEE Registration Authority. https://standards.ieee.org/products-services/regauth/. Accessed: 2018-11-30.

[ieec] IEEE Standards Association. http://standards.ieee.org/develop/regauth/index.html. Accessed: 2018-01-21.

[IEE14] IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture. *IEEE Std 802-2014 (Revision to IEEE Std 802-2001)*, pages 1–74, June 2014.

[IEE16] IEEE Standard for Ethernet. *IEEE Std 802.3-2015 (Revision of IEEE Std 802.3-2012)*, pages 1–4017, March 2016.

[ios] iOS8 MAC Randomization - Analyzed! https://blog.mojonetworks.com/ios8-mac-randomization-analyzed/. Accessed: 2018-12-09.

[iso] ISO/OSI Model. https://www.iso.org/standard/20269.html. Accessed: 2018-01-21.

[Kol98] Andrei N Kolmogorov. On tables of random numbers. *Theoretical Computer Science*, 207(2):387–395, 1998.

[lin] Presentation Layer Definition. http://www.linfo.org/presentation_layer.html. Accessed: 2018-03-08.

[LxDRpW10] W. Liu, H. x. Duan, P. Ren, and J. p. Wu. Weakness analysis and attack test for WLAN. In *The 2010 International Conference on Green Circuits and Systems*, pages 387–391, June 2010.

[ME12] ABM Musa and Jakob Eriksson. Tracking unmodified smartphones using wi-fi monitors. In *Proceedings of the 10th ACM conference on embedded network sensor systems*, pages 281–294. ACM, 2012.

[NG96] Mark Nelson and Jean-Loup Gailly. *The data compression book*, volume 199. M & t Books New York, 1996.

[oui] OUI list from IEEE. https://standards.ieee.org/develop/regauth/oui/oui.csv. Accessed: 2017-08-12.

[PGG+07] Jeffrey Pang, Ben Greenstein, Ramakrishna Gummadi, Srinivasan Seshan, and David Wetherall. 802.11 User Fingerprinting. In *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking*, MobiCom '07, pages 99–110, New York, NY, USA, 2007. ACM.

[PGSW07] Jeffrey Pang, Ben Greenstein, Srinivasan Seshan, and David Wetherall. Tryst: The Case for Confidential Service Discovery. In *HotNets*, volume 2, page 1, 2007.

[rtr] RTR - 2400 MHz spectrum (WLAN). https://www.rtr.at/en/tk/Spektrum2400MHz. Accessed: 2018-12-10.

[Sha48] Claude E Shannon. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27:623–656, 1948.

[Sto15] James V Stone. *Information theory: A tutorial introduction*. Sebtel Press, 2015.

[SVS+06] M. Solarski, P. Vidales, O. Schneider, P. Zerfos, and J. P. Singh. An Experimental Evaluation of Urban Networking using IEEE 802.11 Technology. In *2006 1st Workshop on Operator-Assisted (Wireless Mesh) Community Networks*, pages 1–10, Sept 2006.

[tpl] TL-WN722N 150Mpbs High Gain Wireless USB Adapter Specifications. https://www.tp-link.com/us/products/details/cat-5520_TL-WN722N.html#specifications. Accessed: 2018-12-10.

[tsh] The Wireshark Network Analyzer. https://www.wireshark.org/docs/man-pages/tshark.html. Accessed: 2018-12-07.

[VMC+16] Mathy Vanhoef, Célestin Matte, Mathieu Cunche, Leonardo S. Cardoso, and Frank Piessens. Why MAC Address Randomization is Not Enough: An Analysis of Wi-Fi Network Discovery Mechanisms. In *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*, ASIA CCS '16, pages 413–424, New York, NY, USA, 2016. ACM.

[wira] Display Filter Reference: IEEE 802.11 wireless LAN. https://www.wireshark.org/docs/dfref/w/wlan.html. Accessed: 2018-12-07.

[wirb] Display Filter Reference: IEEE 802.11 wireless LAN management frame. https://www.wireshark.org/docs/dfref/w/wlan_mgt.html. Accessed: 2018-12-07.

[zop] Zopfli Compression Algorithm. https://github.com/google/zopfli. Accessed: 2018-11-14.