FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Assignment of master's thesis

Title:	Device for Wi-Fi Security Testing
Student:	Bc. Petr Heřmánek
Supervisor:	Ing. Josef Kokeš
Study program:	Informatics
Branch / specialization:	Computer Systems and Networks
Department:	Department of Computer Systems
Validity:	until the end of summer semester 2021/2022

Instructions

1. Research the current state-of-the-art of Wi-Fi security area - the KR00K vulnerability, attacks such as Evil Twin or KRACK, defenses, and mitigations.

2. Design a device capable of performing Wi-Fi security testing. Focus on portability and use of easily available components, e.g., Raspberry Pi, open source software, etc.

3. Construct a prototype of such a device using technology of your choice.
4. Develop a demonstration of the Evil Twin attack using your device.
5. Document and discuss your results.

Electronically approved by prof. Ing. Pavel Tvrdík, CSc. on 9 February 2021 in Prague.

Czech Technical University in Prague
Faculty of Information Technology
Department of Computer Systems

Master's thesis

Device for Wi-Fi Security Testing

Bc. Petr Heřmánek

Supervisor: Ing. Josef Kokeš

14th May 2021

Acknowledgements

I would like to thank my supervisor Ing. Josef Kokeš and reviewer Ing. Tomáš Čejka, Ph.D for their academic help and support. Equal amount of gratitude goes to my colleagues at Avast Software, specifically to my team leader Ing. Petr Schreiber, and research counselor Martin Hron. Lastly, this project would not be possible without the continuous support of my family and friends during this time of global crisis.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended. In accordance with Article 46(6) of the Act, I hereby grant a nonexclusive authorization (license) to utilize this thesis, including any and all computer programs incorporated therein or attached thereto and all corresponding documentation (hereinafter collectively referred to as the "Work"), to any and all persons that wish to utilize the Work. Such persons are entitled to use the Work in any way (including for-profit purposes) that does not detract from its value. This authorization is not limited in terms of time, location and quantity. However, all persons that makes use of the above license shall be obliged to grant a license at least in the same scope as defined above with respect to each and every work that is created (wholly or in part) based on the Work, by modifying the Work, by combining the Work with another work, by including the Work in a collection of works or by adapting the Work (including translation), and at the same time make available the source code of such work at least in a way and scope that are comparable to the way and scope in which the source code of the Work is made available.

In Prague on 14th May 2021 ....

Czech Technical University in Prague
Faculty of Information Technology

Citation of this thesis

Heřmánek, Petr. *Device for Wi-Fi Security Testing.* Master's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2021.

Abstrakt

Narušení bezpečnosti bezdrátových sítí představuje závažné ohrožení soukromí, integrity a autentičnosti komunikace. Globálně rozšířené protokoly obsahují mnoho slabin, od nedostatků původního návrhu až po kritická pochybení programátorů. Cílem této práce je analýza opakujících se hrozeb za účelem vytvoření ucelené klasifikace a obranných direktiv. Na základě těchto znalostí jsme zkonstruovali přenosné demonstrační zařízení, které je určené k plně automatizovanému spuštění útoku Evil Twin, s možností využít další moderní nástroje pro testování zabezpečení.

Klíčová slova Wi-Fi, Evil TwinBerries, zabezpečení Wi-Fi, Wi-Fi útoky, Evil Twin, KRACK, KR00K, Rogue Access Point

Abstract

The wireless network compromise presents a serious threat to traffic confidentiality, integrity, and authenticity. The globally widespread protocols already have a well established attack surface filled with various pitfalls, ranging from poor design decisions to critical programming errors. In this thesis, we focus on researching the recurring threats to provide a modern taxonomy overview and general protection guidelines. To demonstrate its functionality, we constructed a portable device capable of a fully automated Evil Twin kill chain execution along with the contemporary Wi-Fi auditing toolkit options.

Keywords Wi-Fi, Evil TwinBerries, Wi-Fi security, Wi-Fi attacks, Evil Twin, KRACK, KR00K, Rogue Access Point

Contents

1 Introduction **1**

1.1 Motivation 1

1.2 Goal 2

2 IEEE 802.11 **3**

2.1 OSI model 5

2.2 WNIC 6

2.3 Frame Types 6

3 Protocols **9**

3.1 WPA2 11

3.2 WPS 12

3.3 IEEE 802.1X 14

3.4 WPA3 16

4 Wi-Fi Attack Surface **19**

4.1 Taxonomy Overview 20

4.2 General protection guidelines 22

4.3 Established threats 24

4.4 WarDriving 28

5 Evil Twin Attack **31**

5.1 Principle 31

5.2 Exploitability 34

5.3 Protection 46

5.4 Advancements 50

6 Evil TwinBerries **53**

6.1 What is it 53

6.2 Why does it exist 55

6.3 How does it work 56
6.4 Testing 60

Conclusion **65**

Bibliography **67**

A Acronyms **77**

B Contents of the enclosed SD **83**

List of Figures

2.1 2.4 GHz channel span 4
2.2 5 GHz channel grouping 4
2.3 802.11 OSI model outline 5
2.4 Simplified 802.11 frame with WPA2 header 6

3.1 WPA2 communication 13
3.2 802.1X communication 15
3.3 WPA3's SAE handshake 17

4.1 Wireless rig 20

5.1 Airodump-ng 32
5.2 Kismet 33
5.3 Evil Twin in series 34
5.4 Evil Twin parallel 34
5.5 Wigle SSID Heatmap 36
5.6 Deauthentication frames 38
5.7 Malicious captive portal 42
5.8 BiRe Flowchart 49

6.1 Evil TwinBerries 54
6.2 Encryption Timeline 56
6.3 iw summary 57
6.4 SSLStrip2 credential stealing 59
6.5 Evil TwinBerries inner look 61
6.6 Evil TwinBerries size 62

List of Tables

3.1 Wi-Fi Security Protocols 10

4.1 Proposed taxonomy 21

5.1 Beacon frame fingerprinting fields 47

6.1 Toolkit 63

Chapter 1

Introduction

WPA2 remains the most used protocol globally for Wi-Fi security and it may take years for the WPA3 to take over. The globally widespread protocols already have a well established attack surface filled with various pitfalls, ranging from poor design decisions to critical programming errors.

Consequences include the loss of traffic confidentiality, integrity, and authenticity. Therefore, the risks of eavesdropping, packet forgery, Denial-of-Service, device infection, or credential theft require special attention. The wireless network compromise presents a serious threat to both personal and enterprise modes. Among other things, adversaries may leverage enterprise variants to gain an initial foothold in the target network and move laterally to obtain sensitive resources.

The goal of this thesis is to explore the logic, availability, and prevention, along with the malicious potential of known Wi-Fi attacks. We aim to use such research to construct a proof-of-concept security testing device and provide practical defense guidelines.

1.1 Motivation

Preserving personal privacy is critical in the modern world and the wireless flow of data is no exception. Popular retail Wi-Fi access solutions do not guarantee sufficient protection against recurring attack models out of the box (sometimes even after the latest update and proper configuration). Thus, they leave the customer potentially unsafe. Without the necessary knowledge, users can hardly prevent or recognize visible signs of malicious intent as a direct result of Wi-Fi network compromise.

Effective education of public and system administrators on the security risks associated with this technology is essential due to the vast Wi-Fi utilization and high availability of effective security auditing tools. To better protect also means to periodically monitor evolving trends in wireless security and update/patch the networking equipment. By maintaining a state-of-the-art penetration testing toolkit, network administrators can evaluate and report the effectiveness of ongoing network fortification efforts.

1.2 Goal

The goal of this thesis is to explain the current Wi-Fi security threats to construct a device capable of demonstrating bad actor influence over a wireless network for lawful protection development and organized educational purposes.

First, we need to establish the necessary technical groundwork for the Wi-Fi communication principles so we explore the IEEE 802.11 standard and the security protocols/modes. Based on that, we discuss the current Wi-Fi attack surface and propose general protection guidelines. The next step is a detailed analysis of the unique Evil Twin attack vector with a focus on exploitability, advancements, and the available detection schemes. The important focus was to put in context the attacker's point of view with known effective countermeasures and further educate on the risks and protections of Wi-Fi communication.

After the theoretical chapters, we present a device specialized for Wi-Fi security testing and provide a technical overview. Demonstration of choice is a rogue access point in the form of a highly customizable Evil Twin pipeline and a contemporary wireless auditing toolkit. To showcase the accessibility of such devices, we chose a publicly known attack surface and the common Raspberry Pi family (ARM architecture).

Chapter 2

IEEE 802.11

We dedicate the first chapter to the IEEE 802.11 set of media access control (MAC) and physical layer (PHY) protocols outlined in [23] for wireless local area network (WLAN) implementation. This is to set the necessary technical groundwork for the upcoming Wi-Fi communication principles. Such information is important because Wi-Fi is essentially a brand based on IEEE 802.11 with its own trademark and various certifications for security standards.

The 802.11 family consists of a series of half-duplex, over-the-air modulation techniques that transmit over various frequencies, including but not limited to 2.4 GHz, 5 GHz, 6 GHz, and 60 GHz frequency bands. Furthermore, fourteen channels are designated in the 2.4 GHz range, spaced 5 MHz apart from each other except for a 12 MHz space before channel 14.[23]

A standard speed Wi-Fi signal occupies five channels in the 2.4 GHz band, therefore in order to avoid interference from other nearby APs, the selected channel number should differ by five or more.[6] Image 2.1 was taken from [6] and shows how the 2.4 GHz band Wi-Fi channels overlap. However, there may always be regulations imposed by a local government to specify the transmission capabilities.

5 GHz wireless access systems including the RLAN (Radio Local Area Network) equipment are used in wireless local area networks to provide a high-speed data communications in between devices connected to the wireless infrastructure. The European standard EN 301 893 [17] covers operation in the 5.15–5.725 GHz band (adopted in May 2017). In the image 2.2, we show the grouping options for bonded channels to allow for simultaneous transmission for Load Based Equipment as described in the standard.

Development and overall advancement of this technology is done in versions and labeled through letter based suffixes, e.g. 802.11b, 802.11a, 802.11g, 802.11n, 802.11ac and 802.11ax. Recently, the Wi-Fi Alliance decided to generalize a novel easy-to-understand naming approach where versions are referred to using numerical sequence, e.g. 802.11n as Wi-Fi 4, 802.11ac as Wi-Fi 5 and 802.11ax as Wi-Fi 6.[83]

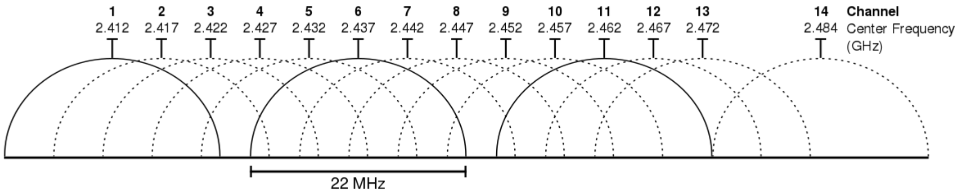

1 2 3 4 5 6 7 8 9 10 11 12 13 14 Channel
2.412 2.417 2.422 2.427 2.432 2.437 2.442 2.447 2.452 2.457 2.462 2.467 2.472 2.484 Center Frequency (GHz)
22 MHz

Figure 2.1: Wi-Fi channels in the 2.4 GHz band.

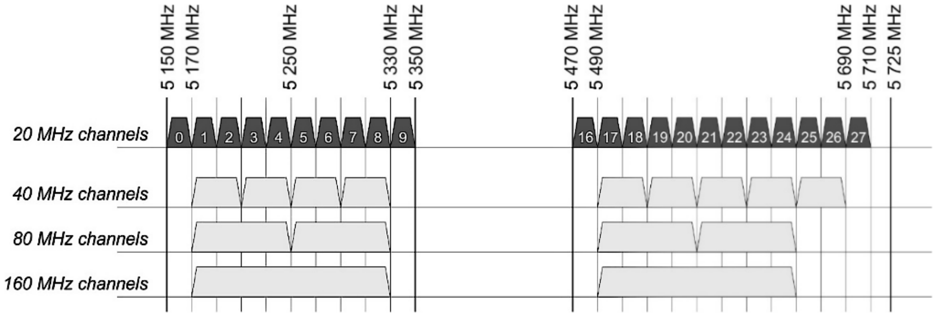


Figure 2.2: Grouping options for bonded Wi-Fi channels in the 5 GHz band.

Additionally, the IEEE 802.1X defines encapsulation of the Extensible Authentication Protocol (EAP) over IEEE 802.11 (EAPOL) and 802.11i specifies security operations (implemented as WPA2).

The 802.11 protocol also defines two entities that can be present in a wireless network: base stations and access points. Base Stations are client devices that connect to an access point, for example, laptops, phones, etc. In this thesis we may also use the terms station, client and client device to describe the same thing. Finally, Access Point (AP) refers to the networking hardware that provides stations with access to the distribution system (i.e. network).[58]

These entities can be used to build one of three kinds of wireless network, or basic service set (BSS). The configurations are Independent BSS (IBSS), Mesh BSS and finally the Extended Service Set (ESS) which realises a collection of BSSs connected using a common Distribution System (DS).[57] Basic Service Sets within infrastructure network consist of a collection of zero or more clients connected to an access point, and are identified by their Basic Service Set Identifier (BSSID). Similarly, ESS is identified by its Extended Service Set Identifier (ESSID), often known as the "network name". [55]

2.1 OSI model

The Open Systems Interconnection (OSI) model is a conceptual model created by the International Organization for Standardization which enables diverse communication systems to communicate using standard protocols.[9] There are seven abstraction layers, each handles a specific job and communicates with the layers above and below itself. We are not going to discuss the whole application of the OSI model, instead we focus on the 802.11 changes in the lowest layers, i.e. Data-Link and Physical. Image 2.3 taken from [73] graphically outlines these changes.

Data-Link layer defines the format of data on the network by managing packets in smaller pieces called frames. The 802.11 Data-Link layer is divided into two sublayers. Upper portion is the IEEE 802.2 Logical Link Control (LLC) and the bottom portion is the Media Access Control (MAC). When the data is handed off to the LLC (from layer 3) it becomes known as the MAC Service Data Unit (MSDU), i.e. a data payload that contains the IP packet plus some LLC data. Once LLC sends the MSDU to the MAC sublayer, it is encapsulated in a MAC Protocol Data Unit (MPDU) and handed further down as a 802.11 frame.[23]

Then, the Physical layer is in charge of transmitting raw bit streams over the chosen physical medium in two sublayers. In this case, the upper portion is known as the Physical Layer Convergence Procedure (PLCP) and the lower portion as the Physical Medium Dependent (PMD). In short, PLCP sublayer prepares the frame for transmission as it takes the frame from the MAC sublayer and creates the PLCP Protocol Data Unit (PPDU) accordingly. The PMD sublayer then modulates and transmits the data as bits.[23]

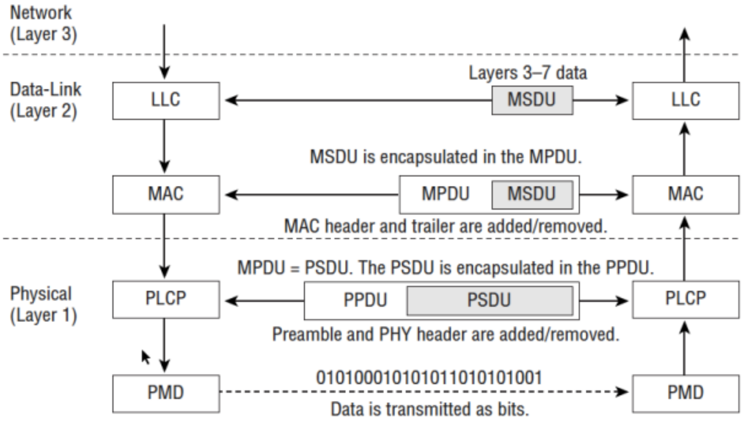

Network
(Layer 3)
Data-Link
(Layer 2)
Physical
(Layer 1)
LLC
Layers 3–7 data
MSDU
LLC
MSDU is encapsulated in the MPDU.
MAC
MPDU
MSDU
MAC
MAC header and trailer are added/removed.
MPDU = PSDU. The PSDU is encapsulated in the PPDU.
PLCP
PPDU
PSDU
PLCP
Preamble and PHY header are added/removed.
PMD
01010001010101101010101001
Data is transmitted as bits.
PMD

Figure 2.3: IEEE 802.11 OSI model outline.

2.2 WNIC

Wireless Network Interface Controller (WNIC) connects to a wireless network using the layers 1 and 2 of the OSI model and uses an antenna to communicate via radio waves. To suit to different needs, manufacturers offer varying hardware capabilities and chipsets. Their popularity is often based off of the driver support and the level of performance/flexibility one gets at the given price point.

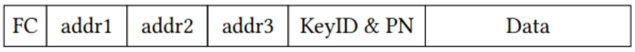
Such controllers support multiple communication modes (i.e., Master, Managed, Promiscuous and Monitor). For our purpose, the most relevant is the monitor mode. Unlike promiscuous mode, the monitor mode enables a WNIC to capture wireless packets without the need to associate it with an AP.[20] Hence it is possible to effectively monitor surrounding traffic with lesser digital footprint.

2.3 Frame Types

This section is dedicated to showcasing different frame types and relevant theory behind them. Many of the variants described here appear later in the thesis as building blocks to various exploitations.

There are 3 types of frames used in the 802.11 MAC layer 2 communications happening over the air which manage and control the wireless link. They are the Management Frames, Control Frames and Data frames.[45] Simplified layout of a 802.11 frame with a WPA2 header (taken from [45]) is shown in Figure 2.4.

First, the Frame Control (FC) field contains several bit-flags, then the next three fields contain the address of the receiver (addr1), the address of the sender (addr2), and the address of the final destination (addr3). For example, when a client sends an outbound IP packet, addr1 equals the MAC address of the AP, addr2 that of the client itself, and addr3 that of the router.[71]



FC	addr1	addr2	addr3	KeyID & PN	Data

Figure 2.4: Simplified 802.11 frame with WPA2 header.

The content of each frame, along with its source and destination address, is stored in the data field. When a frame is encrypted, its plaintext header includes the KeyID and Packet Number (PN) field. The PN field stores the replay counter used by the encryption algorithm, and the KeyID identifies which key was used to protect the frame.[71]

Traffic coordination became a large issue in WLANs, therefore management and control packets are dedicated to assist in this difficult task. The following list showcases common management frames used to establish and maintain Wi-Fi communications. Broader scope with examples can be found at [45]. Each entry also contains corresponding wlan.fc.type_subtype listed in parentheses for easier filtering.

Authentication (0x0b) AP decides on the identity of a radio NIC (Network Interface Controller) to allocate resources

Deauthentication (0x0c) announcement packet to another station in order to terminate secure communications (contains reason code, VSI and 802.11w info)

Association req / res (0x0)/(0x01) enables the AP to allocate resources and synchronize

Reassociation req / res (0x02) similar to a association, often utilized during client roaming

Disassociation (0x0a) terminates the association from either AP or device

Beacon (0x08) AP periodically announces its presence and relay information

The purpose of management frames varies greatly, for example, the beacon frame is an essential management component transmitted by an AP for broadcasting its capabilities to client devices in a network. [26] Whereas, probe requests and probe responses are used to obtain information from another station or AP. On the other hand, deauthentication or disassociation have a history of Denial-of-Service oriented misuse. (see image 5.6 from section 5.2.2). It is also possible to see control frames in the image referenced above as their goal is to assist the delivery of data frames between stations and collision management. These consist of Request to Send (RTS), Clear to Send (CTS) packets to reduce collision and ACK to inform sending station if no errors are found.

Finally, Protected Management Frames (PMF) is a feature based off the 802.11w amendment to provide integrity for both unicast and broadcast management frames, while also being responsible for unicast management frame encryption.[15] In effect, it enables protection for Disassociate and Deauthenticate frames, making it harder for an adversary to deauthenticate clients from a network.[74] However, not all clients will support this feature; therefore, the usage is negotiated by the client and access point. [1]

[1] WPA3 will make this security feature mandatory

Chapter 3

Protocols

An absence of wires indicates that any capable device within the range of a wireless network can attempt to intercept or send packets. Furthermore, the data intercepted in this way can be analyzed for illegal purposes, for example, stealing credit card information or login credentials.

The goal of this chapter is to provide general information about the protocols used to secure Wi-Fi networks. These protocols provide user authentication and mechanisms to ensure both confidentiality and integrity of transmitted data. We decided to exclude further analysis of legacy protocols due to their broken security and deprecation. Table 3.1 compares current practices in terms of important features.

Global deployment and utilization of Wi-Fi technology require a continuous effort to protect the client's privacy and device security. Therefore, the supported protocols/devices have to be thoroughly evaluated from a security standpoint. There are other issues related to global security maintenance, for example, the problematic patching process or product implementation diversity by manufacturers. Also, the now established growing trend of connecting vast amounts of appliances and other smart devices to the Internet (known as the Internet of Things or IoT) has only made wireless technologies more relevant.

The first wireless network encryption standard, Wired Equivalent Privacy (WEP), was introduced as a part of the original 802.11 specifications ratified in 1997.[27] Vulnerabilities which were discovered in 2001 in this encryption method, required the development of a new encryption standard.[18] Due to the cybersecurity weakness of WEP, it is recommended to avoid its usage.

In 2002, Wi-Fi Alliance released a new encryption method the Wi-Fi Protected Access (WPA), which was compatible with the old hardware and thus only required a software upgrade.[80] However, this was only a temporary workaround due to its weak cryptographic hash-algorithms and the Wi-Fi Alliance kept on improving WPA.

The wireless specification 802.11i ratified in 2004 included the improved

Wi-Fi Protected Access 2 (WPA2) that currently holds the top position for the most widely used protocol (as seen in chapter 6.2). Additionally, in 2007 Wi-Fi Alliance created an additional safety method Wi-Fi Protected Setup (WPS). WPS makes it possible to connect to a wireless network just by pressing a hardware button, thus avoiding the need to enter a password entirely.[4]

Wi-Fi Security Protocols		
	WPA2	WPA3
Security modes	PSK or Enterprise	Personal or Enterprise
Handshake[2]	4-way, group key	Dragonfly
Packet encryption suite	AES-CCMP	AES-CCMP, AES-GCMP
Encryption key size[3]	128-bit	128-bit or 256-bit
Integrity techniques	CBC-MAC	SHA-2
Secure setup	WPS	DPP
Protected Management Frames	Optional	Required
Forward Secrecy	No	Yes
Year of release	2004	2018

Table 3.1: Wi-Fi Security Protocols

The latest generation, WPA3 brings cutting-edge security protocols to the market. WPA3 adds new features to simplify Wi-Fi security, enable more robust authentication, deliver increased cryptographic strength for highly sensitive data markets and maintain resiliency.[85]

Modern security protocols offer different options for personal and enterprise networks. For example, the Wi-Fi Alliance claims that users of WPA3-Personal receive increased protections from password guessing attempts while WPA3-Enterprise users can take advantage of higher grade security protocols for sensitive data networks.[85] Compared to WPA2, both security modes use the same encryption method (TKIP or CCMP) but the inner mechanisms change.

The main difference between the security modes is how clients authenticate themselves to the network. Personal networks are designed for home networks and use a PSK for authentication. Enterprise networks use the IEEE 802.1X standard which provides authentication mechanisms that relay the authentication towards an authentication server (RADIUS).[74]

[2] please refer to [71] and [72] to see a complete list of handshakes.

[3] WPA3-Enterprise mandates longer key sizes (the equivalent of 192-bit security).[12]

3.1 WPA2

The original security standard Wi-Fi Protected Access (WPA) was introduced in 2003 as an interim solution to the limited protection offered by WEP. All modern protected Wi-Fi networks rely on the 802.11i amendment which defines both the 4-way handshake and two encryption protocols. However, due to the slow standardization of this amendment, the Wi-Fi Alliance already started certifying devices based on a draft of 802.11i under the Wi-Fi Protected Access (WPA) program.[71]

The WPA program added support for Temporal Key Integrity Protocol (TKIP) encryption, an older form of security technology vulnerable to cryptographic attacks. Once 802.11i had been finished, the WPA2 certification program was created. The main difference is that WPA2 mandates support for the more secure (AES-)CCMP encryption protocol and optionally allows the (WPA-)TKIP encryption protocol while the reverse is true for WPA.[71]

The 802.11i amendment defines two data-confidentiality protocols. The first is called the Temporal Key Integrity Protocol (TKIP), however, this protocol is deprecated due to security concerns. The second protocol is commonly called (AES-)CCMP and the AES block cipher replaces the RC4 cipher.[70] In 2012 the 802.11ad amendment added a new data-confidentiality protocol called the Galois/Counter Mode Protocol (GCMP).

The CCMP protocol is based on the AES cipher operating in the CCM mode (counter mode with CBC-MAC). It is an Authenticated Encryption with Associated Data (AEAD) algorithm and is considered secure as long as no Initialization Vector (IV) repeats under a particular key. In CCMP, the IV is the concatenation of the sender is MAC address, a 48-bit nonce, and some additional flags derived from the transmitted frame.[70]

All three protocols behave like stream ciphers where the keystream is generated and XORed with the plaintext data. The generated keystream depends on the TK subkey of the PTK, and a 48-bit packet number. This packet number, commonly called a nonce, is incremented by one for every transmitted packet, starting at zero or one depending on the specific protocol, it is used as a replay counter by the receiver.[71]

The 4-way handshake provides mutual authentication based on a shared secret called the Pairwise Master Key (PMK) and negotiates a fresh session key called the Pairwise Transient Key (PTK). Every message in the 4-way handshake is defined using EAPOL frames which we describe later in section 3.3.1. During this handshake, the client is called the supplicant and the AP is called the authenticator. We use these terms interchangeably and the description below provides the usage and source of various keys that take a part in the handshake.[2]

PSK creation as part of the 802.11i passphrase-to-PSK mapping scheme.

Usage: shared secret.

PMK is derived from a pre-shared password in a personal network, or negotiated using 802.1X in an enterprise network.

Usage: the highest order key used within the 802.11i amendment.

PTK is formed as a combination of the PMK, the Authenticator Nonce (ANonce), the Supplicant Nonce (SNonce), and the MAC address of both the supplicant and client.

Usage: later split into a Key Confirmation Key (KCK), KeyEncryption Key (KEK), and Temporal Key (TK).

GTK The GTK may be derived from a group master key (GMK) but it is a random value assigned by the broadcast/multicast source.

Usage: protects broadcast/multicast medium access control (MAC) protocol data units (MPDUs) from that source.

The KCK and KEK from PTK split are used to protect handshake messages, while the TK is used to protect normal data frames with a data-confidentiality protocol (see [87] for TK-related vulnerabilities). Note that in an existing connection, the PTK can be refreshed by initiating a new 4-way handshake. During this rekey, all 4-way handshake messages are encrypted by the data-confidentiality protocol using the current PTK.[70]

Details on the exchanged messages and key installations can be found in standards issued by the Wi-Fi Alliance [23] or in materials oriented on handshake evaluation such as [70] and [71]. Image 3.1 taken from said materials, shows a graphical overview of the handshake.

If WPA2 is used, the 4-way handshake also transports the current Group Temporal Key (GTK) to the supplicant. The authenticator periodically refreshes the group key and distributes it to all clients using the group key handshake. This ensures that the GTK works for the recently authorized clients only. The most defensive scenarios renew the key once a client leaves the network.

3.2 WPS

Wi-Fi Protected Setup™ (WPS) is a WPA2 feature that lets the user easily connect WPS-supported client devices, such as wireless printers, to the their router wirelessly.[31] Administrators may choose from multiple methods to connect new devices to their network. The traditional network setup requires clients to know the pre-shared key and SSID to establish connection. WPS methods are based on different identification mechanisms to enable flexibility, e.g. Personal Identification Number (PIN), Push-Button Configuration (PBC), or Near Field Communication (NFC).[37]

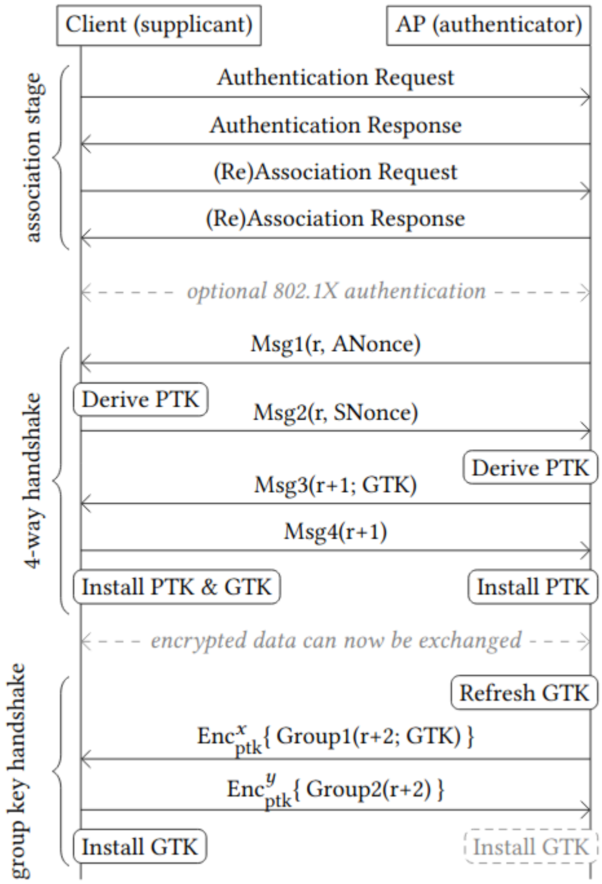


Figure 3.1: Messages transmitted when a client connects with an AP using WPA2.

PIN method relies on the secrecy of an 8-digit identifying code to allow devices into the network [4].

Push-Button activates WPS by pushing a physical button on the board (few boards have such a button marked on the board case/label).

Push-Button-Virtual represents the “WPS Accept” virtual button alternative in the wireless interface menu.[37]

NFC (Near Field Communication) capable device is added to the network when in contact with another NFC-tag or device [75]

[4] Such code may be static and provided as part of the device packaging/labeling, or dynamically generated by the device.

WPS is not intended for use in Enterprise networks where separate authentication servers are used to control network access. It is strongly recommended to consider the negative security implications associated with using this technology (see section 4.3.2).

Along with WPA3, the Wi-Fi Alliance will launch the Device Provisioning Protocol (DPP) as a simple way to onboard “headless” IoT devices into the network; this will make the provisioning more manageable and user friendly.[43]

The protocol strives to protect against known threats, e.g. passive adversaries/eavesdroppers and active adversaries that could deny provisioning service without alerting the user, create new networks and, manipulate device connections.[81]

Also, Wi-Fi Easy Connect simplifies the device configuration with the use of QR codes, NFC tags or downloaded device information from the cloud and a user-chosen configurator (such as a smartphone or tablet) to manage network access.[82]

3.3 IEEE 802.1X

IEEE 802.1X is a part of the IEEE 802.1 group of networking protocols and represents an IEEE Standard for port-based Network Access Control (PNAC). Its main goal is to enable authenticated access to the IEEE 802 media, including Ethernet, Token Ring and 802.11 wireless LANs. Although Remote Authentication Dial-In User Service (RADIUS) support is optional within the IEEE 802.1X, it is expected that many authenticators will function as the RADIUS clients as seen in [24].

For central management, it is possible to deploy backend authentication, and accounting as a part of the authentication, authorization, and accounting (AAA) practices. If that is the case, authenticators will function as the AAA clients.

Extensible Authentication Protocol (EAP) is used as the authentication framework for the 802.1X and further defines EAPOL as the encapsulation of EAP over IEEE 802.11 (EAP over LAN).[25] In image 3.2, we show the 802.1X communication layout as a guiding material.

In this thesis, we briefly discuss enterprise solutions just to provide a technical contrast and initial guidance for future work. The image 2.4 showcases how the previously defined EAPOL is used between the supplicant and the authenticator. EAP is then usually tunneled over Radius between the authenticator and the authentication server, but it can also be done over other AAA protocols (Diameter).[48]

3.3.1 EAP

The most commonly used EAP implementations are the EAP-PEAP and the EAP-TTLS. They provide a secure tunnel to protect the authentication pro-

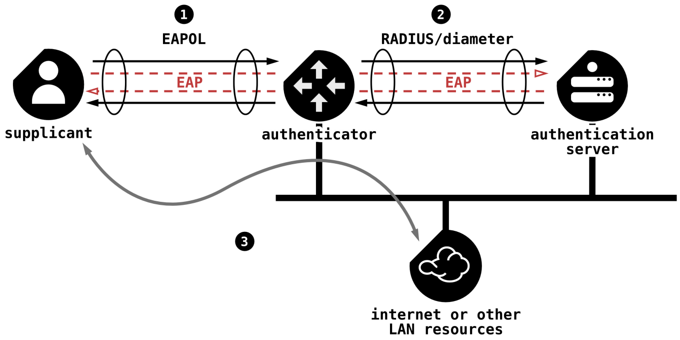


Figure 3.2: IEEE 802.1X communication layout.

cess that takes place between the supplicant and the authentication server.

Once a client device attempts to connect to the network, the authentication server presents the supplicant with an x.509 certificate. After the client accepts the certificate, a secure encrypted tunnel is established between the authentication server and the supplicant to proceed with authentication.[57]

However, the supplicant and the authentication server do not communicate directly. All their communication is relayed by the authenticator (multiple OSI layers). More specifically, the supplicant and the authenticator communicate using a Layer 2 protocol such as IEEE 802.11X and the authenticator communicates with the authentication server using RADIUS, which is a Layer 7 protocol.[49]

3.3.2 RADIUS

The RADIUS (Remote Authentication Dial-In User Service) is a network protocol that defines rules and conventions for AAA management Its implementations serve as the basis for multiple commercial offerings and supply the authentication, authorization, and accounting needs of many companies and ISPs.[24] A server is generally used to authenticate users or devices before allowing them to access the network. Clients present their credentials via a customizable login prompt or using the link framing protocol such as the Point-to-Point Protocol (PPP) that uses authentication packets instead.[49] These users or devices are then authorized for specific network services and subsequently logged.

To provide an example, it is possible to use the RSA RADIUS with the RSA Authentication Manager to directly authenticate users attempting to access network resources through the RADIUS-enabled devices. When a RADIUS server receives remote user access requests from clients (for example a

VPN) it forwards the access requests to the RSA Authentication Manager for validation.[52] Authentication Manager then sends accept or reject messages to the server.

3.4 WPA3

In 2018 the WPA3 was released as the late addition to the Wi-Fi Security practices. It is important to remark that the WPA3 does not define new protocols but instead serves as a certification that defines which existing protocols a device must support. This certification mandates support of the Dragonfly handshake, also known as the Simultaneous Authentication of Equals (SAE).[72]

In contrast to the 4-way handshake of WPA2, the SAE handshake provides forward secrecy and is resistant to dictionary attacks.[71] Dragonfly supports Elliptic Curve Cryptography (ECC) with elliptic curves over a prime field (ECP groups), and Finite Field Cryptography (FFC) with multiplicative groups modulo over a prime (MODP groups).[86] Dragonfly is also used in networks that authenticate clients using EAP-pwd, i.e the AP initiates the handshake and the commit and confirm frames are encapsulated in 802.1X frames.[72]

Before initiating the Dragonfly handshake, the pre-shared password is converted to a group element using a hash-to-element method. A method for MODP groups is called the hash-to-group and for elliptic curves the hash-to-curve. The Password Authenticated Key Exchange (PAKE) converts the processed passwords into a high-entropy keys and after executing SAE the negotiated high-entropy key is used in a 4-way handshake to derive a fresh session key. Image 3.3 showcases the graphical overview of the WPA3's SAE handshake as proposed in [72]. More detailed information about the WPA3 can be found in the original Wi-Fi Alliance documents [84] [86] or the published security research [72].

Although the WPA3 still uses the WPA2's 4-way handshake, it is not vulnerable to dictionary attacks because the key generated by SAE has a much higher entropy than an ordinary password. [72] Backward compatibility with WPA2 is offered as a transition mode feature where WPA2 and WPA3 are simultaneously supported using the same password [84]. However, in section 4.3.5 we discuss the potential security implications.

Due to its built-in protection against known side-channel attacks, the Dragonfly overhead is high compared to similar algorithms. Adversaries can abuse this resource intensity to conduct a Denial-of-Service attack. To combat some of the issues, Dragonfly designers implemented a reactive anti-clogging defense that utilizes secret cookies.

The AP declares a "cookie demanding" state as a result of an ongoing Denial-of-Service attempt and binds generated strings to message senders.

The first SAE message will not be processed unless the message contains a valid cookie.[84] However, researchers were able to bypass this anti-clogging mechanism as seen in [72].

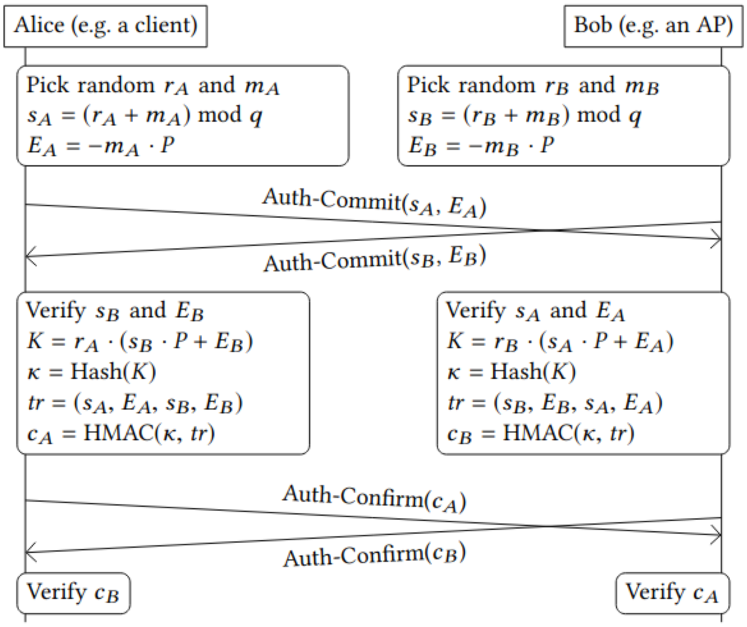

Alice (e.g. a client)
Bob (e.g. an AP)
Pick random r_A and m_A
$s_A = (r_A + m_A) \bmod q$
$E_A = -m_A \cdot P$
Pick random r_B and m_B
$s_B = (r_B + m_B) \bmod q$
$E_B = -m_B \cdot P$
Auth-Commit(s_A, E_A)
Auth-Commit(s_B, E_B)
Verify s_B and E_B
$K = r_A \cdot (s_B \cdot P + E_B)$
$\kappa = \text{Hash}(K)$
$tr = (s_A, E_A, s_B, E_B)$
$c_A = \text{HMAC}(\kappa, tr)$
Verify s_A and E_A
$K = r_B \cdot (s_A \cdot P + E_A)$
$\kappa = \text{Hash}(K)$
$tr = (s_B, E_B, s_A, E_A)$
$c_B = \text{HMAC}(\kappa, tr)$
Auth-Confirm(c_A)
Auth-Confirm(c_B)
Verify c_B
Verify c_A

Figure 3.3: Simultaneous Dragonfly handshake initiation with elliptic curves.

Chapter 4

Wi-Fi Attack Surface

Previously, we defined the technical background for the Wi-Fi technology and popular security protocols. This chapter is dedicated to the attack surface associated with this widespread technology, where we classify exploitation methods and showcase established threats.

Most importantly, we discuss general protection guidelines for modern Wi-Fi networks to help network administrators and Wi-Fi consumers better secure their wireless perimeter. Attack surface magnitude is closely tied to the protocols selected; however, this thesis primarily covers non-deprecated protocols such as WPA2/3 in mostly the personal mode. Enterprise exploits are of similar nature with the difference of more advanced structure (authentication server), protocols (EAP-PEAP/EAP-TTLS, LLMNR, NBT-NS, NTLM) and communication complexity (Active directory, RADIUS authentication, network traversal/lateral movement).

We can see wireless attacks having impact on state level espionage as well. The US Department of Justice filed a formal accusation (indictment) citing the usage of portable state of the art wireless rig installed into a rented vehicle.[13] The documents represent a rare glimpse into how an elite group may adapt and incorporate the efficiency of Wi-Fi attack surface. The following image 4.1 is a legal attachment from the official indictment.

The idea of high stakes Wi-Fi hacking effort is relevant if the target network is significantly more expensive to breach via traditional means. With the secure deployment of traditional protection technologies such as firewalls, intrusion prevention systems (IPSs), content filters, and anti-virus/anti-malware detection tools, the enterprise networks are in some cases more difficult targets. As a result, adversaries are now exploiting less secure end user devices and Wi-Fi networks to penetrate the enterprise networks.[69]

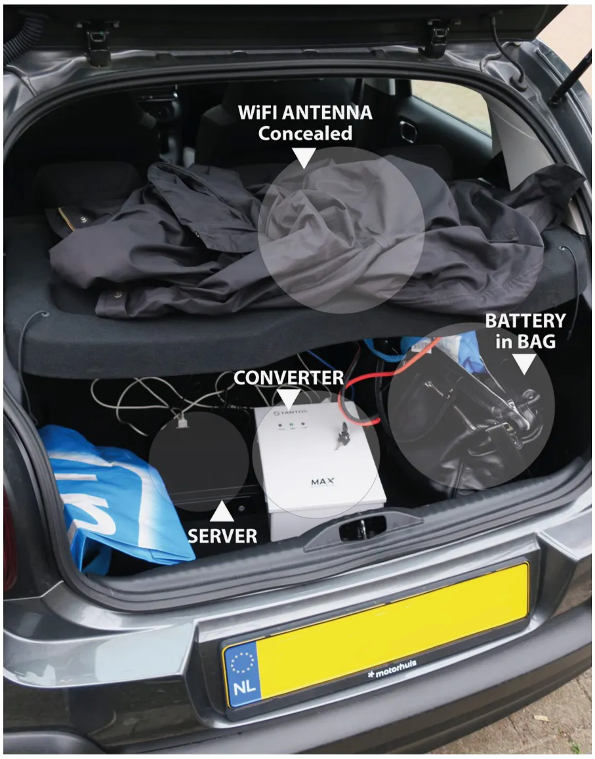


Figure 4.1: State of the art wireless rig.

4.1 Taxonomy Overview

Concurrent with our research, a thesis dedicated to a comprehensive taxonomy of Wi-Fi attacks was published prior to releasing our own materials for the Evil TwinBerries device. We would like to direct anyone interested in the broad scope of Wi-Fi attacks to see the publication [74] based at the Radboud University Nijmegen. To support the global unification of nomenclature and categorization perspectives, we decided to retain the classification approach proposed by [74] and also quote some of their quality taxonomy tables.

Attacks can be separated into multiple types categorised by their goals and leverage mechanisms. The four common types are Man-in-the-Middle (MitM), Denial-of-Service (DoS), Traffic Decryption and Key-recovery. Ultimately, these types are often combined either as a mandatory execution step or an efficiency booster. Further classification focuses on the attack features such as its type, protocol and efficiency. From the practical standpoint it is important how available and sophisticated the exploits are, e.g., the level of automation and protection evasion.

The following table 4.1 showcases some of the current threats to Wi-Fi networks.

Taxonomy Overview		
Type	Protocol	Name
Man-in-the-Middle	*-*	Evil Twin
	-	KARMA Attack
	-	MANA Attack
	WPA*-Open	Known Beacon Attack
Key-recovery	WPA2-PSK	Dictionary Attack
	WPA2-PSK	WPS Brute-force Attack
	WPA2-PSK	WPS Pixie Dust Attack
	WPA2-PSK	PMKID Hash Dictionary Attack
	WPA3-PSK	Dragonblood
Traffic Decryption	WPA2-*	KRACK Attacks
	WPA2-*	KR00K vulnerability
Denial of Service	*-*	Rseource Exhaustion Attack
	WPA2-*	Deauthentication Flooding Attack
	WPA3-*	Dragonfly Resource Exhaustion

Table 4.1: Proposed taxonomy

In the context of Wi-Fi, Man-in-the-Middle is a type of attack focused on rerouting traffic through the adversary or leveraging a rogue access point (RAP) with the intent to eavesdrop on traffic and manipulate packets. A common approach is to either hijack routing properties of connected network or use any variant of the Evil Twin Attack (ETA). Such attacks focus on constructing a RAP to impersonate a legitimate access point that is preferred/trusted by nearby stations to lure wireless clients. The topic of ETA is so vast we decided to dedicate chapter 5 to it.

Denial-of-Service is traditionally used to decrease overall network stability by flooding the target system with management frames or specially crafted packets to cause resource exhaustion. Some Wi-Fi exploitations rely heavily on disconnecting clients from the victim AP because of the way stations attempt to provide clients with wireless connection afterwards. Common motives are handshake collection, automatic network selection (see section 5.2.1), hardware (registry) flush [88], triggered reinicialization of cryptographic values [70]. The variety of DoS attacks grew over the years across the OSI model often with the introduction of vulnerable functions and general availability of hardware (SDR, external WNIC, IoT platforms) capable to do so. More on Denial-of-Service can be found in the Evil Twin chapter (section 5.2.2).

Traffic encryption is one of the main factors driving security protocol evolution forward because confidentiality of encryption keys is an essential factor for data integrity and packet authenticity. As a result, to crack the contents of packets with the purpose of obtaining its contents in plaintext is more difficult in modern networks compared to other more historical configurations (WEP RC4 support). For example, WPA3 cryptography relies on elliptic curves, making it the strongest of Wi-Fi protocols mathematically. However, the currently prevalent protocols have advanced encryption mechanisms in place but the problem lies in improper realisation of such mechanisms rather than a flawed design (see section 4.3). Therefore, short sighted implementation decisions may lead to a weakened encryption.

Finally, the adversary may associate with a Wi-Fi network using a recovered pre-shared key. Similarily to other principles, weaknesses in the authentication protocol (offline dictionary attack) play a role in an unlawful key recovery, but also the statistical analysis of the encrypted traffic.[74] It is important to state that unauthorized access is unacceptable and may lead to a wider and serious network compromise. This led the the newer WPA3 certification to take precautionary prevention measures, leaving mainly WEP, WPA/2 or WPS especially vulnerable.

4.2 General protection guidelines

In this section we highlight protection measures that can be applied for client devices and Wi-Fi networking infrastructure, some even in the enterprise environment. Among other things, this advisory is also based on attacks and techniques we describe in the thesis. Target audience are not only network administrators but the general public as well. With the growth rate of wireless communications it is necessary to educate everyone on concurrent privacy threats associated with Wi-Fi networks to help advance global security.

For personal devices, disable the auto-connect feature and any wireless interfaces when not actively connected to a network. This may prevent clients from automatically connecting to rogue access points. Preferred Networks List (PNL) should only contain essential entries to reduce the surface of victims to spoof, so forgetting networks after establishing connection is helpful. Restricting PNL entries to specific MAC addresses only requires adversaries to know the network they are spoofing.

To prevent malicious certificate behaviour such as SSLStrip, clients should always communicate over HTTPS with a trusted and valid domain. Recently, Google Chrome (Chrome 90) started to prefer HTTPS to HTTP when not specified in the address bar.[67] Additionally, proper HSTS and certificate pinning can help.

Routing traffic through a Virtual Private Network (VPN) makes it harder for an adversary to inspect traffic through a Man-in-the-Middle position since only the encrypted traffic from the tunnel with the VPN is observable.[28]

Security-oriented solutions for smart homes are also commercially available, for example, the Avast Omni[5] or the Trend Micro Home Network Secur-

[5] urlhttps://www.avast.com/en-us/omni

ity[6] may provide an additional layer of security.

As for the actual Wi-Fi infrastructure, migration to a modern security protocol is essential. WPA and WPA2 are still available; however, it is advisable to use equipment that specifically supports WPA3, as relying on the other protocols could leave a network open to exploitation.[8] Some devices already support the WPA3 certification and their deployment should be globally prioritized due to the introduction of stronger encryption and crucial security measures.

The manufacturers of wireless APs periodically release updates and patches for software and firmware. Check the customer support from the manufacturer or the ISP (internet service provider) about instructions, suggestions, and security options. Always keep the network up to date and make sure the system has relevant patches applied.

Change passwords regularly to prevent adversaries from gaining unauthorized access into the network. Dictionary attack and brute force in general relies on the pre-shared key's complexity. Stronger passphrases (at least 20 characters according to the 802.11i) are time and hardware-consuming to attackers and may be spared from generic cracking approaches.

The WPA3 security considerations say that WPA3-Personal should also limit authentication attempts when an implementation identifies an active attack. Repeated authentication failures may indicate that an active attack is underway, allowing implementations to respond appropriately, including throttling the authentication attempts and/or issuing alerts such as Simple Network Management Protocol (SNMP) trap, log message, or others.[84]

Avoid publicizing the Service Set Identifier (SSID) to prevent outsiders from easy access. At the very least, change the SSID to something unique that does not leak any information that could lead to network or PII (personally identifiable information) compromise. Manufacturer's default value attracts attackers as a sign of poorly configured AP that is easy to identify and potentially exploit.[8]

Allow access only for authorized users in order to maintain the privacy of the primary network. Try to utilize the guest option for guest devices, this grants access on a separate wireless channel with a separate password.

Consider installing a firewall directly on wireless devices (a host-based firewall), as well as on the home network (a router or modem-based firewall). Attackers who can directly tap into a wireless network may be able to circumvent a network firewall. Hence the host-based firewall will add a layer of protection to the data.[8] Also, disable remote administration on all nodes unless necessary because it allows outside access to change, for example, the router administrator settings.[16]

If WPA2 is really necessary, preferably deploy separate networks for WPA2 and WPA3 with different pre-shared key to mitigate downgrade issues associ-

[6] `https://www.trendmicro.com/en_us/forHome/products/homenetworksecurity.html`

ated with WPA3 deployment.[84] Disable WPS as it is deemed insecure and should be replaced by a newer standard. Ideally, the device should support Protected Management Frames (PMF) that offers data confidentiality, integrity, origin authenticity and replay protection for management frames[7] (as described in chapter 2.3).

Organizations can deploy Wireless Intrusion Detection System to enhance monitoring and defense capabilities. A Guide to Securing Networks for Wi-Fi by the U.S. Department of Homeland Security states that in response to the growing number of attacks on networks and the risks associated with the pervasive nature of wireless technologies, the major recommendation in the guidance [69] is to deploy a wireless intrusion detection system (WIDS) and wireless intrusion prevention system (WIPS) on every network, to detect and automatically disconnect devices using unauthorized wireless services.

Organizations deploy these systems to create and enforce wireless security policies in enterprise networks Also the ability to centrally monitor and manage enterprise wireless security (with respect to the threats listed later in the thesis) allows the security operations center (SOC) to act upon automated alerts and better understand the perimeter during incident remediation.

4.3 Established threats

The current wireless threat landscape is built on vast research initiative and the short sighted development decisions that defined essential practices for adversaries to leverage.

4.3.1 Password Recovery

Recovering the pre-shared key (PSK) allows the attacker to circumvent authorization layer of victim 802.11 network. Attackers build on various brute force methods to recover the PSK and the final approach is shaped by the exploitable properties of target environment. The goal is to obtain the 4-way handshake between a client and victim AP to brute-force the PSK. This is available because of the (plaintext) transmitted nonce and the MIC over it.[74]

In case of a busy network, the passive approach of sniffing ongoing traffic for handshakes rather than using active Denial-of-Service methods may be sufficient. However, if the situation calls for more active measures to be employed, then the Evil Twin pipeline could be adjusted to collect just enough information out of the handshake to do an offline brute-force attack against the PSK (refer to section 5.2).

Additionally in 2018, a new approach appeared on the forums of a popular password cracker (hashcat) that excludes clients and the attacker directly com-

[7] WPA3 security protocol requires this feature by default.

municates with an AP.[63] Instead of collecting complete 4-way handshakes, the research shows that some routers append too much information (in the form of PMKID) to the Robust Security Network Information Element (RSN IE) hence a single EAPOL frame is sufficient to obtain the PMK (Pairwise Master Key).

PMKID is computed as follows, where “PMK Name” string is constant and both BSSIDs are known to the attacker.[63]

```
PMKID = HMAC-SHA1-128(PMK, "PMK Name" | MAC_AP | MAC_STA)
```

The client-less PMKID attack was also covered by the author of bettercap in his blogpost [34] to highlight the exploitation differences from handshake collection. The advantages of capturing the PMKID hash instead of handshakes make for a more efficient, independent and stealthier process.

4.3.2 WPS Bruteforce Attack

We’ve already discussed the technical elements of the WPS system. Here we present the attacks that cause it to be often disabled as a best practice due to its unreliability. The issue lies in the fact that the design allows adversaries to easily recover the secret PIN and thus learn the WPA/WPA2 pre-shared key. Despite the manufacturers effort to reduce the exploitable attack surface by adding lock-out mechanisms, they often introduce weak methods to generate the secret nonces.[5]

On specific chipsets, random numbers are not really random but are derivations of the hashes, therefore it is possible to guess the nonces and then compute the PIN. Tools like Bully[8] or Reaver[9] have the option to perform an offline attack Pixie Dust (pixiewps), by automatically passing the PKE, PKR, E-Hash1, E-Hash2, E-Nonce and Authkey. According to the documentation, the pixiewps tool will then try to attack Ralink, Broadcom and Realtek chipsets.

4.3.3 KRACK Attacks

The KRACK attacks publication was a groundbreaking research because it affected multiple handshakes on a level where the hijacking of TCP streams and data injections were made possible. When a client joins a network it executes the 4-way handshake (see section 3.1 for a more detailed description) to provide mutual authentication and negotiate a fresh session key (PTK). Such handshake consists of multiple messages that are referred to by numbers according to their order of transmission. The key is then installed after receiving message 3 of the handshake to encrypt normal data frames using the data-confidentiality protocol.

[8] https://gitlab.com/kalilinux/packages/bully

[9] https://github.com/t6x/reaver-wps-fork-t6x

Researchers found multiple mechanisms and race conditions to manipulate the handshake messages to reinstall an already-in-use key. Each time the message 3 is received, the same session key is reinstalled and thereby the incremental transmit packet number (nonce) and receive replay counter used by the data-confidentiality protocol are reset.[70]

Adversaries can then force this reuse to violate communication privacy as a result of severe shortcomings of the globally used authentication processes. More specifically, AES-CCMP was marked as susceptible to replay and decryption of packets but not forgery. TKIP and GCMP failed to protect against packet forgery as well. This type of attacks was also used to attack the group key, PeerKey, and fast BSS transition handshake.[70]

Their work was later extended by further systematic analysis to no longer rely on hard-to-win race conditions but rather employ more practical methods of exploitation. The authors improve key reinstallation attacks (KRACKs) by generalizing known attacks, analyzing all hand-shakes, bypassing 802.11's official countermeasure, auditing (flawed) patches, and enhancing attacks using implementation-specific bugs.[71]

In this advancement, the adversaries no longer rely on plain-text transmission of message 3 but the sleep feature of WNM (Wireless Network Management power-save) allows them to generate the message 3 encrypted under the newly negotiated session key. Attacker then controls the flush of buffered messages for client by unsetting the sleep flag with an empty data frame. Additionally, it was proven that other handshakes, e.g. the Fast Initial Link Setup (FILS) and the Tunneled direct link setup PeerKey (TPK), were also vulnerable to key reinstallations. Some APs also accept replayed message 4's of the 4-way handshake.[71]

We conclude that preventing key reinstallations is harder than expected and believe that (formally) modeling 802.11 would help to better secure both implementations an the standard itself.

The weaknesses are in the Wi-Fi standard, that means they are widespread and will take a lot of global effort to patch. Mainly due to the fact that the standard progressively grows with new features and requires a specific knowledge to understand. Original researchers also stated that the prevention is harder than initially assumed. They believe that the 802.11 standard would benefit from formal modelling or simplification.[71]

4.3.4 KR00K Vulnerability

A recently discovered vulnerability in commonly used Wi-Fi chips, primarily FullMAC WLAN chips manufactured by Broadcom and Cypress. Both WPA2-Personal and WPA2-Enterprise protocols with AES-CCMP encryption are affected. The vulnerability (CVE-2019-15126) was first discovered by the ESET researchers and published as a white paper in February 2020 [88].

Subsequent followup of similar bugs in different chip brands went public in August.[87]

Visible naming similarity to KRACK suggests common exploitation ground, yet these two are not interchangeable as there are important differences. KR00K embodies the worst case scenario achievable by KRACK attacks – all-zero key encryption of transmitted data. However, there are significant changes and constraints in place.

We are no longer discussing a series of key reinstallation attacks but a chip based vulnerability exploitation. Instead of relying on the 4-way handshake, KR00K uses victim disassociation as the trigger. This gives it a strict limit on the amount of data acquirable, thus only a part of the communication is actually exposed.

Once a station's WLAN session gets disassociated, the session key (TK) stored in the Wireless Network Interface Controller's (WNIC) Wi-Fi chip is cleared in memory (set to zero).[88] Consequentially, the data left in the Tx buffer are transmitted with their confidentiality flawed by this insecure encryption practice.

To make things worse, monitor mode enabled WNIC does not require the attacker to know the Pre-Shared Key (PSK), or have any connection to the network at all. This important approach distinction results from the ability to forge management frames and independently capture vulnerable traffic.

An adversary may periodically trigger disassociation of vulnerable devices. By anticipating the Tx buffer flush, the attacker can bypass one data protection layer and decrypt the intercepted communication. From personal experience, the period of actions resulting in KR00K trigger has to take into account the importance of reassociation and the time needed for a Tx buffer refill.

4.3.5 Dragonblood

WPA3 has also shown significant security issues despite being the latest certification for Wi-Fi communication. Among other things, the specification mandates a new handshake to be used and overall security guarantees are unclear. The Dragonfly variant used in WPA3 is also known as Simultaneous Authentication of Equals (SAE) and along with the EAP-pwd, they use the Dragonfly handshake to provide forward secrecy and resistance to dictionary attacks.

The authors of [72] presented vulnerabilities in all WPA3 and EAP-pwd implementations in their important research initiative. After the publication, new draft of the protocols incorporating proposed design changes took place.

For example, their results include Denial-of-Service, novel side-channel leaks (timing and cache), downgrade attacks and offline password brute force.

WPA3 was found vulnerable in transition mode where it accepts connections using WPA3-SAE and WPA2 with the same password. Adversary may forge its origins and transmit beacons to trick the client into thinking the AP only supports WPA2. Despite the ability to detect forged RSNE in beacon frames the authors still carried out a successful dictionary attack with no Man-in-the-Middle position necessary. All they needed was to collect a single authenticated 4-way handshake message.

Novel side-channel attacks involve the password encoding methods (hash-to-group and hash-to-curve) that are part of the password derivation process to leak valuable data. For timing attacks, obtained info about the number of executed iterations can be used to recover the victim's password. On the other hand, cache-based approach aims to help offline brute-force by checking memory access patterns, e.g., whether the Quadratic Residue (QR) test in the first iteration of the hash-to-curve algorithm succeeded or not.[72]

Ironically, when compared with similar algorithms, the Dragonfly overhead is high as a result of covered defenses against known side-channel attacks. This could increase DoS attacks efficiency when spoofing commit frames and because of that its authors put prevention mechanisms in place. However, Dragonblood bypasses the anti-clogging defense (described in section 3.4) based on reflecting the secret cookie by forging MAC addresses and effectively proceeds with the Denial-of-Service attack.[72]

4.4 WarDriving

WarDriving is the act of moving around a specific area and mapping the population of wireless access points for statistical purposes. These statistics are then used to raise awareness of the security problems associated with these types of networks (typically wireless).[22] This often misleading term is tied to the 1983 movie WarGames where the phone number discovery practice WarDialing is depicted. Suffixes added to the 'War' keyword form a nomenclature based on the type of selected transportation (WarWalking, WarFlying, WarCycling) or even exploitation (Warkitting[10]).[66]

The common misconception about WarDriving is the inherent expectation of malicious activity, for example, the unauthorized access of private wireless networks or eavesdropping. We believe the definition above (published by the original DEF CON WarDriving Contest organizer Chris Hurley in his 2004 book) is more accurate and the intent should always be highlighted when discussing this term.[22] There is no doubt that WarDriving is being weaponized not only for statistical purposes but for malicious use as well. However, an individual responsible for malicious actions on wireless networks is most importantly a criminal. The techniques and principles of WarDriving are only misused with ill intent. All statistical efforts of this nature should have their goals firmly set with an appropriate legal review. Deployed code has to be under control and no sensitive or private user data should be involved under any circumstances.

[10] Warkitting combines the notions of WarDriving and rootkits.

The Evolution of the Wi-Fi wireless network landscape moves towards more secure solutions and the auditing techniques had to evolve too. Significantly better performing hardware is currently available with a wide variety of different WNIC chipsets. Additionally, software support is well established, ranging from the operating systems to drivers and robust penetration testing frameworks.

To prove the point, some solutions require only a few steps to setup on a 10$ device (Raspberry Pi Zero W) with trivial setup and provide a fully automated pocket size handshake sniffer (pwnagotchi[11]) focusing on maximizing the crackable WPA key material it captures. Ultimately, the Kali Linux OS supports ARM architecture, thus enables preconfigured weaponization of other Raspberry Pi computers.

[11] pwnagotchi.ai/

Chapter 5

Evil Twin Attack

The Evil Twin attack (ETA) has been a persistent security threat for decades in wireless local area networks (WLANs). An ETA refers to a rogue access point impersonating a legitimate access point to lure in wireless users. This unique Man-in-the-Middle attack vector is often deployed to eavesdrop and alter network traffic of target Wi-Fi networks with both EAP and PSK variants of WPA/2 affected.[32]

Stations are typically optimized to respond to this scenario by selecting the access point within their chosen ESS (Extended Service Set) that has the best signal strength and signal to noise ratio. This opens up options to make devices roam to the attacker's access point, providing a Man-in-the-Middle position between the stations and a network gateway. [55]

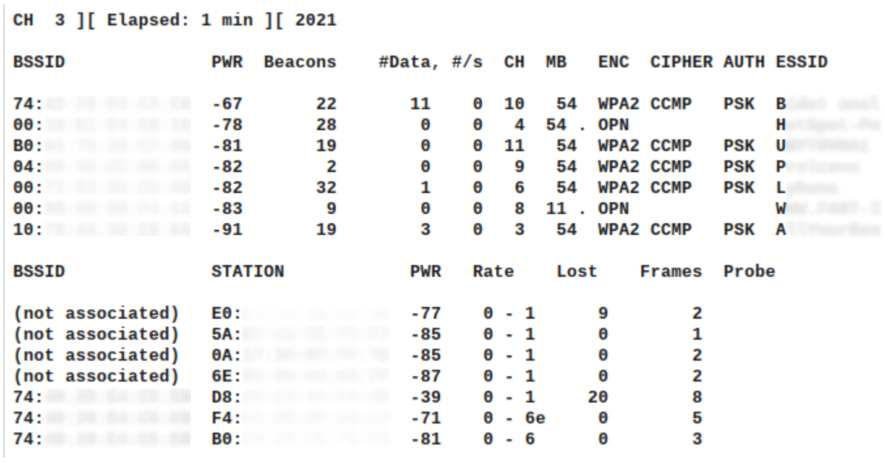
Impersonation of LAP or client hijacking is done in multiple variants, often driven by changes in the management frame behaviour and device protection development. Attack vectors need to consider differences in corporate and personal networks, restricted probing, Preferred Network List (PNL) maintenance and evasion from different detection techniques. Defense against wireless Man-in-the-Middle bad actors also greatly varies and some protections are not fully capable of detecting advanced forms of RAPs alone.

In this chapter we describe the problem of the Evil Twin attack and technical differences among popular RAP advancements. We discuss the pillars of ET and its capabilities in the current wireless environment. Important focus is to put in context the attackers point of view with known effective countermeasures to the current threat landscape and further educate on the risks and protections of Wi-Fi communication.

5.1 Principle

In practice, this attack has multiple mandatory stages to deploy and some optional, based on the projected goals. The subset of key components that enhance stealth and effectiveness is heavily influenced by the wireless environment constraints. To be more specific, such factors could be the configuration of nearby devices, protocol suite setup, existing defensive solutions, network/infrastructure scale, collateral damage and the level of security training amongst victims.

A skilled attacker first scouts the surrounding Wi-Fi environment and collects enough information to classify nearby networks/devices. In the images 5.1 and 5.2 we show some of the observable information using the common household tools (airodump-ng[12] and kismet[13]). The goal is mostly to inspect the management frames and derive conclusions from the observed characteristics.



```
CH  3 ][ Elapsed: 1 min ][ 2021

BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

74:[redacted]      -67       22       11    0  10   54  WPA2 CCMP   PSK  B[redacted]
00:[redacted]      -78       28        0    0   4  54 . OPN              H[redacted]
B0:[redacted]      -81       19        0    0  11   54  WPA2 CCMP   PSK  U[redacted]
04:[redacted]      -82        2        0    0   9   54  WPA2 CCMP   PSK  P[redacted]
00:[redacted]      -82       32        1    0   6   54  WPA2 CCMP   PSK  L[redacted]
00:[redacted]      -83        9        0    0   8  11 . OPN              W[redacted]
10:[redacted]      -91       19        3    0   3   54  WPA2 CCMP   PSK  A[redacted]

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

(not associated)   E0:[redacted]      -77    0 - 1      9        2
(not associated)   5A:[redacted]      -85    0 - 1      0        1
(not associated)   0A:[redacted]      -85    0 - 1      0        2
(not associated)   6E:[redacted]      -87    0 - 1      0        2
74:[redacted]      D8:[redacted]      -39    0 - 1     20        8
74:[redacted]      F4:[redacted]      -71    0 - 6e     0        5
74:[redacted]      B0:[redacted]      -81    0 - 6      0        3
```

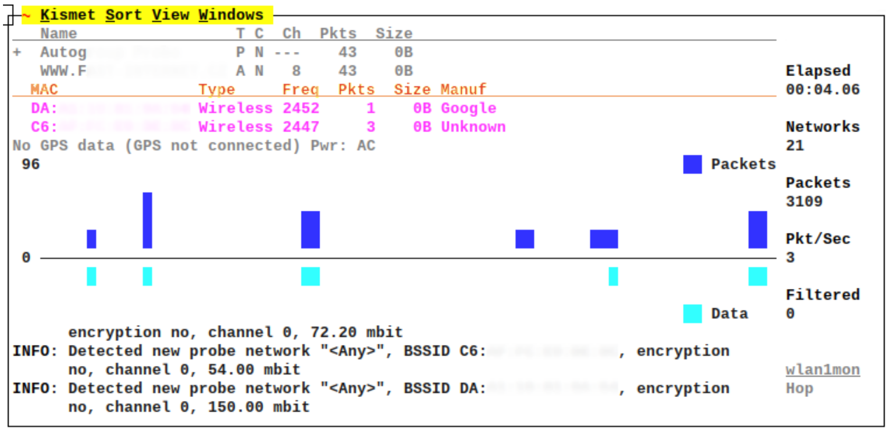
Figure 5.1: Airodump-ng wireless traffic processing with channel hopping enabled.

Monitoring the ongoing wireless traffic to further analyze the types of relationships between nearby communicating devices (on multiple layers) is critical and also provides additional contextual information. For example, signal strength, device naming conventions, OUIs (Organizational Unique Identifiers), SSID geolocation, or observable BSSID patterns are often effective indicators of associations between devices. Recently, the spotlight shifted more towards reconstruction or prediction of Preferred Network List (PNL) entries. In the section 5.4, we discuss additional techniques that focus on limited probing scenarios or networks operating out of range (e.g., KARMA, MANA, Known Beacons Attack).

Up until now, the presence of a malicious actor may not be apparent to the defenders. This depends on how aggressive and loud the selected scouting methods are. Completely passive monitor mode approach may take too long or miss some frames completely. Therefore attackers often choose more aggressive

[12] http://aircrack-ng.org/doku.php?id=airodump-ng

[13] https://kismetwireless.net/



```
~ Kismet Sort View Windows
   Name                     T C  Ch  Pkts  Size
+  Autog[redacted]          P N ---   43    0B
   WWW.F[redacted]          A N  8    43    0B                 Elapsed
  MAC                 Type      Freq  Pkts  Size Manuf         00:04.06
  DA:[redacted]       Wireless 2452     1     0B Google
  C6:[redacted]       Wireless 2447     3     0B Unknown       Networks
No GPS data (GPS not connected) Pwr: AC                        21
 96                                                  Packets
                                                               Packets
                                                               3109

                                                               Pkt/Sec
0                                                              3

                                                               Filtered
                                                     Data      0
      encryption no, channel 0, 72.20 mbit
INFO: Detected new probe network "<Any>", BSSID C6:[redacted], encryption
      no, channel 0, 54.00 mbit                                wlan1mon
INFO: Detected new probe network "<Any>", BSSID DA:[redacted], encryption   Hop
      no, channel 0, 150.00 mbit
```

Figure 5.2: Kismet wireless traffic processing through terminal view.

methods of DoS like deauthentication to uncover hidden networks (hidden by not broadcasting ESSID in either its beacons or broadcast probe response), collect handshakes and observe device behaviour under pressure at the cost of possibly alerting the Wireless Intrusion Detection System.[77]

After sufficient reconnaissance, the attacker reevaluates next steps of the Cyber Kill Chain framework [35] (or MITRE ATT&CK [39] for that matter), i.e. weaponization, delivery and exploitation. In the case of attacking a specific nearby access point, the attempt is to configure and build a RAP capable of routing traffic and luring unsuspecting users. Applied tools and configurations depend heavily on the attacker platform and as an example we show a Linux based environment later in the thesis (see chapter 6).

The constructed fake AP needs to meet specific configuration criteria and retain access point functionality to appear legitimate in order to lure in victim devices. This requires a DHCP server to provide addresses, exploitation dependent DNS handling and appropriate firewall rules to restrict routing. To also offer internet connectivity, rogue access points either establish connection with a different gateway to forward traffic (image 5.3), or rely on their own supply of mobile internet connection (image 5.4).

Throughout this whole process, attackers try to match the capabilities of a victim access point to avoid detection and successfully further exploit insecure mechanisms/practices. Once the exploitation decisions are made, the defenders can expect other tools and services put in place to extend the functionality, often with a malicious captive portal (webserver), database of provisioned devices or downgrade stages for encrypted communication.

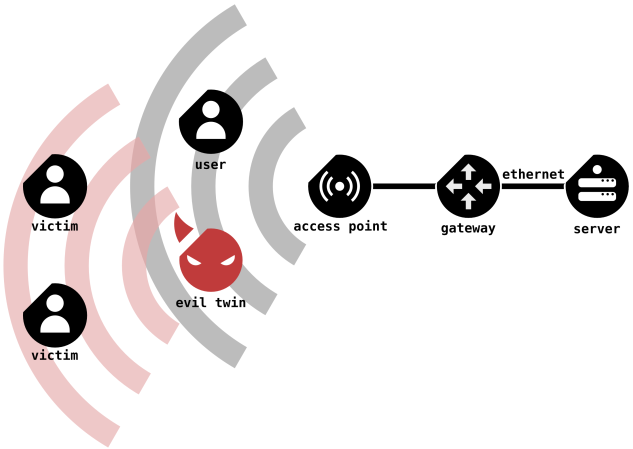


Figure 5.3: Evil Twin in series with another access point.

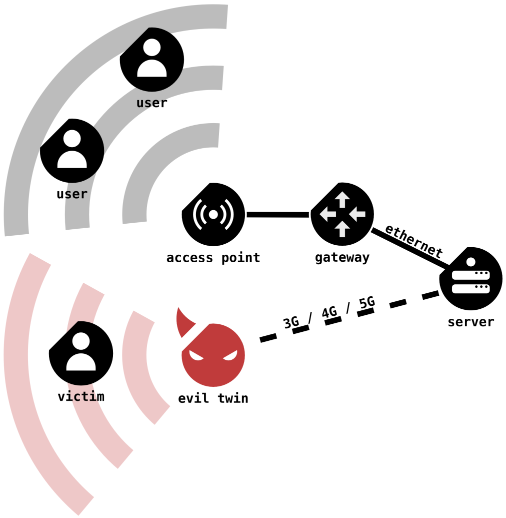


Figure 5.4: Parallel Evil Twin relies on its own internet connection supply.

5.2 Exploitability

Wireless attacks are inherently risky and in red team[14] assessments where time is limited, the aim is to maximize impact in the shortest time frame possible.

[14] A group of people authorized and organized to emulate a potential adversary's attack or exploitation capabilities against an enterprise's security posture.[42]

Long term eavesdropping attempts such as SSLStripping or Credential Sniffing are time consuming and potentially less rewarding due to the widespread adoption of HSTS and Certificate Pinning.[46] Rogue access point can offer robust and effective platform to deliver payloads to victims. Once the attacker has forced a device to connect, she gains the ability to act as either a captive portal or an internet gateway. As a captive portal, it is possible to e.g. redirect users to malicious pages that may prompt them to install implants (updates), exploit browser features or leak personal information and restrict their internet access until they comply. Gateway position allows to inject malicious code into static content (i.e. modify unencrypted JavaScript files in transit, etc).[61]

Evil Twin attacks are the primary means of attacking WPA/2-EAP networks, and can also be effective for attacking WPA/2-PSK networks. We decided to evaluate enterprise approaches in a separate work, however the author Gabriel Ryan (s0lst1c3) of popular toolkit for performing targeted Evil Twin attacks against WPA2-Enterprise networks (EAPHammer[15]) has done an important research [54] on the state of EAP (Extensible Authentication Protocol).

When used for lawful demonstration purposes, attacks should be supervised throughout their entire execution cycle and focused on specific targets within the approved scope. One can limit the affected devices through the use of Management Frames Access Control Lists (MFACL) which allow to whitelist/blacklist traffic specified by MAC address or SSID (see [60] for a thorough description).

The mechanisms leveraged in persuading victim devices into connecting to a malicious AP are introduced with the goals of providing stable connection to users and a level of flexibility to hardware manufacturers. Devices tend to perform some sort of passive scanning, active probing or ESS roaming, making it vulnerable to bad actors looking to automatically establish a connection.

5.2.1 Attraction phase

To obtain the Man-in-the-Middle foothold, the attacker must make their network present when the victim is looking for a wireless network to join. Discovering what the client looks for or taking over relevant SSID is the general idea. If the user is currently associated to a nearby network, the attacker may forcibly cause the victim to restart her search for available networks (see section 5.2.2). In some cases, it is at this point that the adversary may collect information about the preferred/trusted networks or present a counterfeit more appealing to the automatic wireless network selection.[11]

Devices look for known or trusted networks using either active or passive scanning. Active scanning relies on the use directed probe requests to check

[15] https://github.com/s0lst1c3/eaphammer

whether specific Wi-Fi network is nearby, however this technique is not as common anymore due to its severe exploitation potential (see section 5.4).

Probe request frame contains enough information for adversaries to respond with a forged response tricking the client into believing it found a match and proceed with the connection process. Additionally, there is an implied leak of potentially sensitive information since it is possible to use WarDriving plaforms such as wigle[16] to geolocate collected SSIDs and fill in Open Source Intelligence (OSINT) investigation based on context. In image 5.5, we show the wigle SSID heatmap for central Europe.

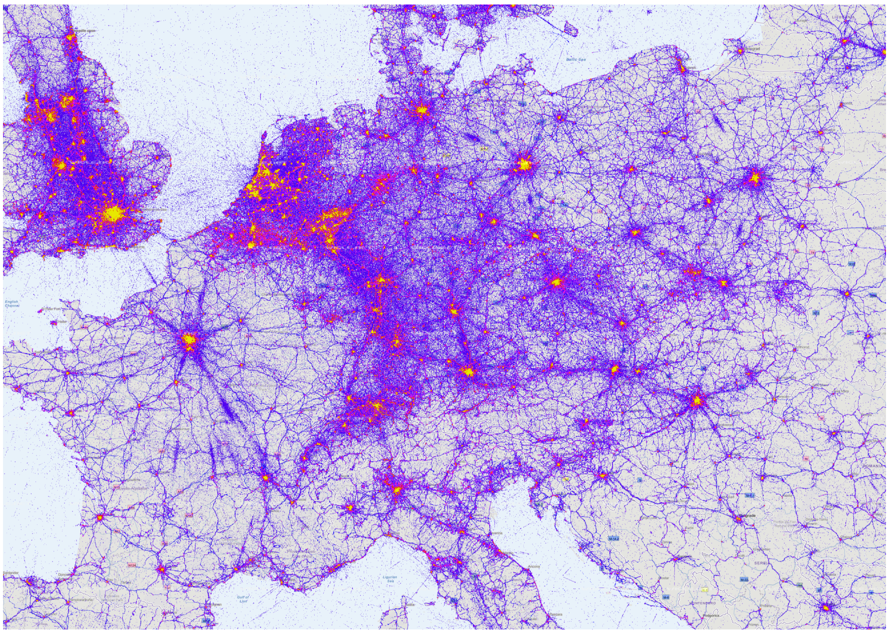


Figure 5.5: SSID heatmap for central Europe on the wigle platform.

Passive scanning, on the other hand, is listening for access points announcing their presence and capabilities via beacon frames. Clients are often configured to compare received ESSIDs with entries in their PNL and initiate automatic connection on match. This practice could be exploited to a certain degree by, for example, the Known Beacons attack that is described later in the thesis.[7]

Additionally, the 802.11 protocol allows stations to roam between access points in the same ESS. Devices are typically optimized to respond to multiple APs within the ESS by selecting the access point that provides the best connection. Users can then move across large facilities with wide ESS reach without the need to explicitly select the best possible access points whenever

[16] https://wigle.net

the previous is out of reach. Best candidate selection usually boils down to a combination of signal strength, throughput, and signal to noise ratio.[58]

Android, for example, recognises a wireless network by its SSID (Service Set Identifier). If any wireless network spoofs the SSID (creates an ET) and arrives in the vicinity of the device, android then assumes it to be the genuine and connects to it directly.[26]

Adversaries may also chain relevant Denial-of-Service attack methods to increase the overall effectiveness through coercion (see subsection 5.2.2). The motivation is to paralyze the original host (possible on various OSI layers) and establish connections with affected clients. Since stations rely on easily fabricated configuration fields (such as the ESSID) to determine which ESS an access point belongs to. It is possible to misuse inherent roaming behavior and provide more appealing AP, thus enabling unsuspecting devices to prefer the malicious access point instead. [55]

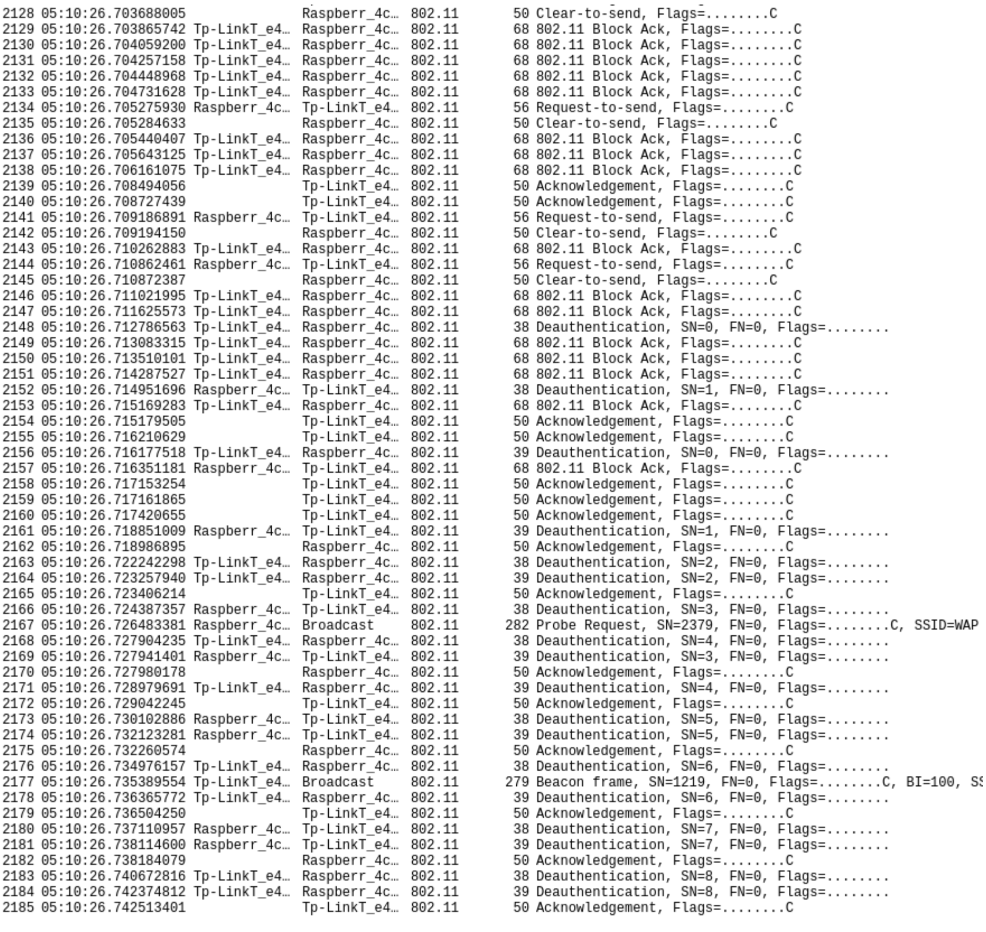
Once an unauthorized party has a fully operating rogue access point with the identified victims connected, then comes the potential to inflict severe damage to the client privacy and device security. Enterprise modifications are also a significant threat with a plethora of potentially devastating consequences to the Active Directory setup or the network infrastructure as a whole. The techniques of indirect wireless pivoting, LLMNR/NBT-NS poisoning and traffic relays (e.g., SMB relay) present a serious attack vector [56], however enterprise modifications are a topic of future work because of their significant impact and more advanced techniques.

5.2.2 Denial of Service

The significance of a stable Wi-Fi connection is not only important for public LANs or small home networks but for devices with critical functionality as well. For example, health monitor sensors allow healthcare personnel to remotely monitor important physiological signs of their patients in real time and assess health conditions. Surveillance cameras along with other security solutions may also be dependent on a specific data rate in order to remain of value. Not to mention that drones rely heavily on wireless communication too and their use case ranges from harmless hobbyists to a well established market for industrial defense solutions against Unmanned Aerial Vehicles (or their morally less appropriate counterparts).

Denial of service is also essential in Evil Twin attack as a coercion method to paralyze the victim legitimate access point (LAP). This makes room for the rogue access point (RAP) constructed in the foreground to take over unsuspecting clients. Again, attackers may leverage many principles with each having a different footprint and effect for the defender to act upon. Intuitively, as is with most Denial of Service vectors, the target system only has a limited amount of resources available. If the attacker can flood an AP with a large volume of problematic management frames, it could result into a complete resource exhaustion, rendering the network inoperable by its users.

The very protocol design is partially at fault as the deauthentication principle is mandatory for legitimate and security oriented purposes (refer to section 2.3 for more technical description of the frames). Besides, the lack of sufficient protection (e.g., protected management frames) brings in undesired consequences. In an attempt to disconnect all AP clients, the device can spoof and send deauthentication or disassociation packets to stations based on observed data traffic. Figure 5.6 shows how the collected frames during such attack and the corresponding time difference in the second leftmost column



```
2128 05:10:26.703688005                  Raspberr_4c…  802.11       50 Clear-to-send, Flags=........C
2129 05:10:26.703865742 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2130 05:10:26.704059200 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2131 05:10:26.704257158 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2132 05:10:26.704448968 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2133 05:10:26.704731628 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2134 05:10:26.705275930 Raspberr_4c…  Tp-LinkT_e4…  802.11       56 Request-to-send, Flags=........C
2135 05:10:26.705284633                  Raspberr_4c…  802.11       50 Clear-to-send, Flags=........C
2136 05:10:26.705440407 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2137 05:10:26.705643125 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2138 05:10:26.706161075 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2139 05:10:26.708494056                  Tp-LinkT_e4…  802.11       50 Acknowledgement, Flags=........C
2140 05:10:26.708727439                  Tp-LinkT_e4…  802.11       50 Acknowledgement, Flags=........C
2141 05:10:26.709186891 Raspberr_4c…  Tp-LinkT_e4…  802.11       56 Request-to-send, Flags=........C
2142 05:10:26.709194150                  Raspberr_4c…  802.11       50 Clear-to-send, Flags=........C
2143 05:10:26.710262883 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2144 05:10:26.710862461 Raspberr_4c…  Tp-LinkT_e4…  802.11       56 Request-to-send, Flags=........C
2145 05:10:26.710872387                  Raspberr_4c…  802.11       50 Clear-to-send, Flags=........C
2146 05:10:26.711021995 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2147 05:10:26.711625573 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2148 05:10:26.712786563 Tp-LinkT_e4…  Raspberr_4c…  802.11       38 Deauthentication, SN=0, FN=0, Flags=........
2149 05:10:26.713083315 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2150 05:10:26.713510101 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2151 05:10:26.714287527 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2152 05:10:26.714951696 Raspberr_4c…  Tp-LinkT_e4…  802.11       38 Deauthentication, SN=1, FN=0, Flags=........
2153 05:10:26.715169283 Tp-LinkT_e4…  Raspberr_4c…  802.11       68 802.11 Block Ack, Flags=........C
2154 05:10:26.715179505                  Tp-LinkT_e4…  802.11       50 Acknowledgement, Flags=........C
2155 05:10:26.716210629                  Tp-LinkT_e4…  802.11       50 Acknowledgement, Flags=........C
2156 05:10:26.716177518 Tp-LinkT_e4…  Raspberr_4c…  802.11       39 Deauthentication, SN=0, FN=0, Flags=........
2157 05:10:26.716351181 Raspberr_4c…  Tp-LinkT_e4…  802.11       68 802.11 Block Ack, Flags=........C
2158 05:10:26.717153254                  Tp-LinkT_e4…  802.11       50 Acknowledgement, Flags=........C
2159 05:10:26.717161865                  Tp-LinkT_e4…  802.11       50 Acknowledgement, Flags=........C
2160 05:10:26.717420655                  Tp-LinkT_e4…  802.11       50 Acknowledgement, Flags=........C
2161 05:10:26.718851009 Raspberr_4c…  Tp-LinkT_e4…  802.11       39 Deauthentication, SN=1, FN=0, Flags=........
2162 05:10:26.718986895                  Raspberr_4c…  802.11       50 Acknowledgement, Flags=........C
2163 05:10:26.722242298 Tp-LinkT_e4…  Raspberr_4c…  802.11       38 Deauthentication, SN=2, FN=0, Flags=........
2164 05:10:26.723257940 Tp-LinkT_e4…  Raspberr_4c…  802.11       39 Deauthentication, SN=2, FN=0, Flags=........
2165 05:10:26.723406214                  Tp-LinkT_e4…  802.11       50 Acknowledgement, Flags=........C
2166 05:10:26.724387357 Raspberr_4c…  Tp-LinkT_e4…  802.11       38 Deauthentication, SN=3, FN=0, Flags=........
2167 05:10:26.726483381 Raspberr_4c…  Broadcast     802.11      282 Probe Request, SN=2379, FN=0, Flags=........C, SSID=WAP
2168 05:10:26.727904235 Tp-LinkT_e4…  Raspberr_4c…  802.11       38 Deauthentication, SN=4, FN=0, Flags=........
2169 05:10:26.727941401 Raspberr_4c…  Tp-LinkT_e4…  802.11       39 Deauthentication, SN=3, FN=0, Flags=........
2170 05:10:26.727980178                  Raspberr_4c…  802.11       50 Acknowledgement, Flags=........C
2171 05:10:26.728979691 Tp-LinkT_e4…  Raspberr_4c…  802.11       39 Deauthentication, SN=4, FN=0, Flags=........
2172 05:10:26.729042245                  Tp-LinkT_e4…  802.11       50 Acknowledgement, Flags=........C
2173 05:10:26.730102886 Raspberr_4c…  Tp-LinkT_e4…  802.11       38 Deauthentication, SN=5, FN=0, Flags=........
2174 05:10:26.732123281 Raspberr_4c…  Tp-LinkT_e4…  802.11       39 Deauthentication, SN=5, FN=0, Flags=........
2175 05:10:26.732260574                  Raspberr_4c…  802.11       50 Acknowledgement, Flags=........C
2176 05:10:26.734976157 Tp-LinkT_e4…  Raspberr_4c…  802.11       38 Deauthentication, SN=6, FN=0, Flags=........
2177 05:10:26.735389554 Tp-LinkT_e4…  Broadcast     802.11      279 Beacon frame, SN=1219, FN=0, Flags=........C, BI=100, SS
2178 05:10:26.736365772 Tp-LinkT_e4…  Raspberr_4c…  802.11       39 Deauthentication, SN=6, FN=0, Flags=........
2179 05:10:26.736504250                  Tp-LinkT_e4…  802.11       50 Acknowledgement, Flags=........C
2180 05:10:26.737110957 Raspberr_4c…  Tp-LinkT_e4…  802.11       38 Deauthentication, SN=7, FN=0, Flags=........
2181 05:10:26.738114600 Raspberr_4c…  Tp-LinkT_e4…  802.11       39 Deauthentication, SN=7, FN=0, Flags=........
2182 05:10:26.738184079                  Raspberr_4c…  802.11       50 Acknowledgement, Flags=........C
2183 05:10:26.740672816 Tp-LinkT_e4…  Raspberr_4c…  802.11       38 Deauthentication, SN=8, FN=0, Flags=........
2184 05:10:26.742374812 Tp-LinkT_e4…  Raspberr_4c…  802.11       39 Deauthentication, SN=8, FN=0, Flags=........
2185 05:10:26.742513401                  Tp-LinkT_e4…  802.11       50 Acknowledgement, Flags=........C
```

Figure 5.6: Deauthentication attack frames captured in Wireshark.

Most popular tools nowadays can mount deauthentication attacks. For example, the popular open-source aircrack-ng suite contains the aireplay-ng binary which the authors describe as responsible for generating traffic for the later use in aircrack-ng for cracking the WEP and WPA-PSK keys. Documentation of aireplay-ng states that for directed deauthentications it sends out a total of 128 packets for each deauth the user specifies (64 packets are sent to the AP itself and 64 packets are sent to the client). [1] Different attacks are available that can cause deauthentication for the purpose of capturing WPA handshake data, fake authentications, interactive packet replay, hand-crafted ARP request injection and ARP-request reinjection (packetforge-ng tool also enables creation of arbitrary frames). [1]

As the protocols evolved, more deauthentication mechanisms were introduced to bypass new countermeasures. Since common deauthentication packets are easily detected when monitoring network activity, some more advanced techniques appeared in the publicly circulating codebase. For example, the following methods are implemented and listed in MDK4[17], the proof-of-concept tool to exploit common IEEE 802.11 protocol weaknesses.

Beacon Flooding sends beacon frames to show fake APs at clients (sometimes crashes network scanners and drivers).

Authentication Denial-Of-Service sends authentication frames to all APs found in range (too many clients can freeze or reset several APs).

SSID Probing and Bruteforcing probes APs and checks for answer, useful for checking if SSID has been correctly decloaked and if AP is in sending range (bruteforcing of hidden SSIDs with or without a wordlist is also available).

Deauthentication and Disassociation sends deauthentication and disassociation packets to stations based on data traffic to disconnect all clients from an AP.

Michael Countermeasures Exploitation sends random packets or re-injects duplicates on another QoS queue to provoke Michael Countermeasures on TKIP APs (AP will shutdown for a whole minute).[21]

EAPOL Start and Logoff Packet Injection floods an AP with EAPOL Start frames to keep it busy with fake sessions and thus disables it to handle any legitimate clients (or logs off clients by injecting fake EAPOL Logoff messages).

Attacks for IEEE 802.11s mesh networks on link management and routing (flood neighbors and routes, create black holes and divert traffic).

WIDS Confusion to confuse/abuse Intrusion Detection and Prevention Systems by cross-connecting clients to multiple WDS nodes or fake rogue APs.

Packet Fuzzer with multiple packet sources and modifiers.

[17] `https://github.com/aircrack-ng/mdk4`

The tool is based on the aircrack-ng osdep library and greatly expands the available Denial-of-Service options. Some of the functions are deployed along with aireplay-ng as part of the popular multi-use Wi-Fi testing script airgeddon[18].

Additionally, it is possible to selectively jam 802.11 wireless networks on the physical layer using, for example, a Software Defined Radio (SDR) such as HackRF one by The Great Scott Gadgets. Substantial research [64] on Wi-Fi jamming via SDR demonstrates three different scenarios that differ in the occupied bandwidth (between 10 MHz and 20 MHz) and the channel power of the applied signal (20 MHz or 40 MHz).

Radio jamming represents the action of deliberate electromagnetic emission for blocking or interference over an authorized radio communication. Physical layer for wireless Denial-of-Service is an important part of the OSI model. In situations when data rate is extremely important, the type of jamming presented here could create connectivity problems and seriously affect the user perceived quality of service (even when the jammer is placed at a considerable distance from the targeted device).[64] This way, an attacker leaves an unusual digital footprint which may pose an unique evasion advancement.

5.2.3 Captive Portal

A captive portal is meant to hold newly connected users captive before they are granted network access based on their compliance with the administrators policies and requirements. The motivation behind captive portals is driven by various factors and ultimately represents a platform for multi-level privacy compromise. Associated authentication practices provide a point of entry for malicious actors looking to gain access to a guest user's device and possibly move laterally to the larger corporate network.

Deployment of this functionality is often done on specially managed or public hotspots. Typically, to prompt for access credentials, customer information or acceptance of internal policy/ToS (Terms of Service). After doing so, the client is redirected to a landing page as configured by the administrator.

A wide variety of captive portals occurs in the wild, mainly due to the unique and specific requirements different hotspot providers have. Development complexity (and cost) depends on the robustness of employed feature set (e.g., social media integration, advanced fingerprinting, PII identification, customer identity) and additional third party involvement for further data processing/analytics.

Captive portals can also take on a more offensive role in the access point setup since they implement a web page with the potential to authenticate and account the activities of the user in a network. Many web based attacks can be launched through the portal. The current history of malicious intent

[18] https://github.com/v1s1t0r1sh3r3/airgeddon

contains captive portal cloning (phishing) with subsequent credential/identity theft, browser history reconstruction attacks [62], android syscall triggers [26], occasional sandbox escapes (CVE-2021-21261[19]) or the browser exploitation – BeEF[20].

Device-to-Device (D2D) attack is in this context defined as an exploit where one device launches a malicious activity on another through the wireless communication channel.[40] This attack was showcased in [26] as a part of protection proposal with the purpose of infecting an Android device before the relay of network traffic through it occurs (may bypass some WIDS). The following image 5.7 was taken from [76] and shows a copy of a legitimate captive portal design used to deploy a malicious script injection.

Their Evil Twin attack was configured to launch a malicious component of an already installed app in the device on submission of the portal page. The malicious component may be a service which opens a port or sends an SMS to premium number or exfiltrates sensitive information to malicious server. Connection is terminated immediately after execution takes place.[26]

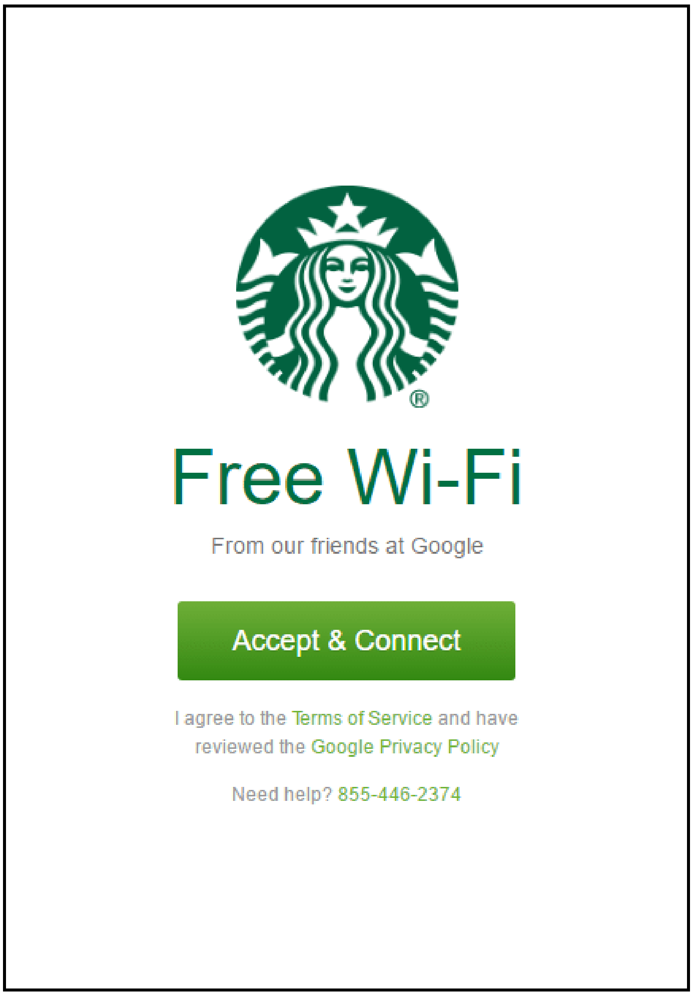
Browsers rely on many layers of caching to speed up web applications; by caching a resource like an HTML document or a video, browsers avoid the overhead of re-fetching that resource the second time a user visits a page. Dabrowski et al. (2016) [10] successfully launched a browser history stealing attack through captive portal with a new variant later presented by [62] in 2018. This branch of attacks relies on embedding cross-origin requests for common URLs and monitoring indicators that would suggest this page has been visited (e.g., measuring when re-paint events happen for links that have already been visited).[62]

From the criminal side of things, in 2019 the threat researchers at IBM X-Force IRIS reported a Wi-Fi criminal activity aimed at commercial-grade layer 7 (L7) routers that are widely deployed.[29] Attacks leak E-commerce data through ads and JavaScript injections, making it hard for victims to point out the initial point of compromise (especially, if they visit such networks regularly).

However, it is important to keep in mind that privacy concerns are emerging about malicious practices not only from criminal actors but from hotspot providers themselves. A study on privacy risks of public Wi-Fi Captive portals conducted by Suzan Ali et al.[3] reveals the collection of a significant amount of privacy-sensitive personal data through the use of social login (e.g., Facebook and Google) and registration forms, and many instances of tracking activities, sometimes even before the user accepts the hotspot's privacy and terms of service policies. Most hotspots use persistent third-party tracking cookies within their captive portal site; these cookies can be used to follow the user's browsing behavior long after the user leaves the hotspots, e.g., up

[19] `https://nvd.nist.gov/vuln/detail/CVE-2021-21261`

[20] `https://beefproject.com/`



Free Wi-Fi

From our friends at Google

I agree to the Terms of Service and have reviewed the Google Privacy Policy

Need help? 855-446-2374

Figure 5.7: A common captive portal phishing with injected data draining script.

to 20 years.[3]

Authors conducted measurements in Montreal Canada, mostly in cafes, restaurants, shopping malls, retail businesses, banks, and transportation companies (bus, train and airport), some of which are local to the city, but many are national and international brands. 40 hotspots (59.7%) used third-party captive portals that appear to have many other business customers across Canada and elsewhere. Thus the results might be applicable to a larger geographical scope.

Except for a few exceptions, all examined hotspots employ varying levels of user tracking technologies on their captive portals and landing pages. On average, the research shows 7.4 third party tracking domains per captive portal (max: 34 domains, including 10 known trackers).

It further proves that social login providers may share several privacy-sensitive PII items. The US Department of Homeland Security defines personally identifiable information or PII as any information that permits the identity of an individual to be directly or indirectly inferred, including any information that is linked or linkable to that individual.[68]

Authors showcase LinkedIn (a business-oriented social network) to share the user's full name, email address, profile picture, full employment history, and the current location. Most hotspots also share PII and browser/device information with third-parties via the referrer header, the request-URL, HTTP cookie or WebStorage. As for device fingerprinting, 35.8% hotspots perform some form of fingerprinting on captive portals and 76.1% on landing pages.

Such practices are a serious privacy threat to users, especially when the general client awareness of possibly bulky ToS is taken into consideration. But even those who carefully inspect the terms of service may be completely unaware of the real data manipulation. Although McDonald's tracks users in their captive portal (9 known trackers, 28 fingerprinting attributes), the captive portal lacks a privacy policy stating their use of web tracking.[3]

5.2.4 Eavesdropping

Among other things, the Man-in-the-Middle position allows adversaries to eavesdrop and alter traffic routed through the compromised node. If done right, this could lead to malicious actions like the use of online services on behalf of affected victims or serving malformed content. Criminal intentions often focus on downgrading the traffic encryption or certificate forgery to leak sensitive information such as credentials, personal/payment information or visited domains.

With the wide adoption of TLS certificates and HSTS (HTTP Strict Transport Security) the attackers need to go an extra mile to circumvent the HTTPS protection.

Common approach is to proxy and modify the processed HTTP(S) communication (additionally DNS if HSTS is present). Tools capable of most of these eavesdropping techniques have their source code and documentation publicly available online in most cases. Hence their availability is not strictly restricted to well funded organizations (see section 6.1).

5.2.4.1 SSLStrip

If a website accepts a connection through HTTP and redirects to HTTPS, visitors may be susceptible to one of the well known encryption downgrade techniques called the SSLStripping[21] made popular by Moxie Marlinspike who later became the creator of the private messenger Signal.

[21] `github.com/moxie0/sslstrip`

When the victim makes a request to access a secure resource, such as a login page, the attacker receives the request and forwards it to the server. The server establishes an encrypted tunnel with the attacker due to falsified origin. Afterwards, the victim is forwarded a modified server's response, converted from HTTPS to HTTP with the original server destination listed as the responder. As a result, all subsequent requests from the victim and the server will occur over an unencrypted HTTP connection through the attacker. They can be viewed or modified before forwarding to the server since the traffic is sent in plaintext.[56]

HSTS is an enhancement of the HTTPS protocol that was designed to mitigate the weaknesses exploited by tools such as SSLStrip. When a HTTP client requests a resource from a HSTS enabled web server, the server adds the following header to the response:

```
Strict-Transport-Security: max-age=31536000
```

The HSTS header informs the browser that it should never load a site using HTTP and automatically convert all attempts to access the site using HTTP to HTTPS requests instead.[56] [36]

Developers also use preload lists to preload domains into browsers as HSTS eligible. Most major browsers (Chrome, Firefox, Opera, Safari, IE 11 and Edge) have HSTS preload lists based on the Chrome list. Domain scanner used to evaluate proposed requirements (hstspreload[22].org) has its source code publicly hosted on GitHub under The Chromium Project If a site sends the preload directive in an HSTS header, it is considered to be requesting inclusion in the preload list and may be submitted via the form on this site.[47]

The following technique advancement caught traction after a talk at Black-Hat Asia 2014 by Leonardo Nve (Exploiting changes on DNS server configuration).[44] His contribution was to also proxy and modify DNS traffic in order to partially bypass the HSTS. When a victim navigates to `www.evilcorp.com`, for example, the attacker redirects the user to `wwwww.evilcorp.com` over HTTP.[56] Accomplishing this can be as simple as responding with a 302 redirect that includes the location header. However, this assumes certificate pinning is not used and that the user does not notice a communication with different subdomain.[56] We successfully performed credential theft in a controlled environment using this newer technique and the output log can be seen in figure 6.4.

Certificate pinning restricts which certificates are considered valid for a particular website, by "pinning" the certificate authority (CA) issuer(s), public keys or even end-entity certificates of their choice. Clients connecting to that server will treat all other certificates as invalid and refuse to make an HTTPS connection. [51] Despite the security enhancement when correctly implemented, certificate pinning may also represent a security threat in case of

[22] `https://github.com/chromium/hstspreload`

device configuration compromise that allows to setup an unauthorized HPKP (HTTP Public Key Pinning) policy.

5.2.4.2 SSL/TLS certificate forgery

SSL/TLS certificates are X.509 digital files issued by an independent third party and later installed on a web server. X.509 is a standard format for public key certificates, digital documents that securely associate cryptographic key pairs with identities such as websites, individuals, or organizations. [53]

Server usually responds to clients with a certificate where values like domain name (DN), public key, or signature are provided. Before the key is used, the clients perform validation of e.g. the certificate CA trust, expiration or domain name. To attack certificates, popular and still continuously developed tools such as SSLsplit ([23]) and mitmproxy ([24]) represent well documented certificate interception tools.

For SSL HTTPS connections, SSLsplit generates and signs forged X509v3 certificates on the fly, mimicking the original server certificate's subject DN, subjectAltName extension and other characteristics. SSLsplit has the ability to use existing certificates of which the private key is available, instead of generating forged ones. There is an existing support for NULL-prefix CN certificates but otherwise the authors do not implement exploits against specific certificate verification vulnerabilities in SSL/TLS stacks.[50]

SSLsplit also implements a number of defences against mechanisms which would normally prevent MitM attacks or make them more difficult. E.g., deny OCSP (Online Certificate Status Protocol) requests in a generic way, mangle headers to prevent server-instructed public key pinning (HPKP), avoid HSTS, avoid Certificate Transparency enforcement (Expect-CT) and prevent switching to QUIC/SPDY, HTTP/2 or WebSockets (Upgrade, Alternate Protocols).[50]

A full CA implementation that also generates interception certificates on the fly is included in mitmproxy. To generate trust, mitmproxy is registered as a trusted CA with the device manually. After the 4-way handshake, it inspects the certificates used. First the Server Name Indication (SNI) indicates the connected hostname, and then the Common Name (CN) with Alternative Name (SAN) in the upstream certificates are used to generate the dummy certificate for the client.[38]

For the attack to be successful, the victim client has to accept the provided forged certificate. Improper certificate validation by the client side or irresponsible adding of untrusted certificates as exceptions are bad practices with dire consequences. To make things worse, there are signs indicating that a significant number of Let's encrypt certificates (issued by the non-profit certificate authority Let's Encrypt[25]) are being issued for phishing purposes.

[23] `roe.ch/SSLsplit`

[24] `mitmproxy.org`

[25] https://letsencrypt.org/

According to [33], “encrypting everything” includes the bad sites and the widespread use of HTTPS on malicious sites has been on the rise. Authors took 15,270 certificates issued to the keyword “PayPal” and estimate that 96.7% of those are used for phishing purposes. The article highlights then the opinion that for many years, the security industry as a whole has incorrectly taught users to associate HTTPS and the green padlock with a “safe” site. This proposed generalization may have an impact on users not correctly recognizing such phishing attempts.[33]

5.3 Protection

The fact that this attack is still here 20 years later after its initial attempts only proves that the idea of an Evil Twin or Rogue Access Point requires its own protection defense branch, strategy and solutions.

As previously mentioned, defense mechanisms against wireless Man-in-the-Middle bad actors vary in detection approaches. Currently known Evil Twin attack detection schemes often look for specific connection characteristics, network behaviour and traffic monitoring in order to determine gateway validity. These are observed through client devices or additional hardware implants made specifically to monitor wireless communication channels. To provide a controlled reference point, some solutions utilize online hosted services to uncover malicious network behaviour.[32]

The evaluated Indicators of Compromise (IoCs) are often a subset of duplicate SSIDs [20], deauthentication frames, time metrics or path anomalies for packet forwarding detection (e.g., traceroute, Round Trip Time (RTT), number of hops) [19]. Some solutions also fingerprint devices (beacon frames [26], RSSI [65]), maintain device whitelists or modify protocol features.[26]

Detections are generally separated into categories mostly depending on the chosen approach. Common defense identification is whether the mechanism leverages an administrator or client position with either pre-association or post-association measures. On the network administrator side of ETA detection, the administrator will be responsible for detecting and/or assisting network clients to detect ETA. Since the network administrator may have all information about the Wi-Fi network, she can have a list of fingerprints of all devices constructing Wi-Fi network.[41] In practice we can also observe hybrid approaches that are designed to also utilize the unique features of client side monitoring.

Most detections take place either after or during the association process.[26] To anticipate and prevent Evil Twin attacks from happening is complicated and slow reactions open a window for the attacker (e.g, D2D attacks). The established RAP needs to be preemptively identified with high confidence (false positives are expensive) and blocked from association attempts with affected devices in the future.

Solutions such as ETGuard [26] claim to provide pre-association detection mechanism with beacon frame fingerprinting and subsequent deauthentication flood to cripple the RAP. It is important to note, that we do not enforce any of these solutions, since their testing is out of the thesis scope, but we chose to build on their educational value instead.

The ETGuard authors proclaimed that existing ET detection solutions on APs were incapable of preventing this attack due to two reasons. Either because they analyze an ET after the relay of user traffic through it or they can detect this attack only for hardware ETs. They chose to improve the situation by presenting an online, fingerprinting based pre-association detection mechanism which works as a client-server mechanism in real-time. The server accommodates beacon frame fingerprints of legitimate APs and passively scans the surroundings across all available channels. Scan results are compared to the stored fingerprints and once ET is detected, deauthentication frames are continuously transmitted to prevent clients from connecting to an ET.[26]

Similarly, in [41] the proposed ETA detection system creates two Virtual Wireless Clients where the first monitors multiple Wi-Fi channels in a random order looking for a specific data packets sent by a server on the internet. In parallel, the second warns the clients when network switches gateway from one AP to another in the middle of a secure connection.[41]

The following table 5.1 lists which fields are considered during the ETGuard identification process.[26]

Identification fields	
Static	BSSID, Beacon Interval, Capability Information, SSID
Dynamic	timestamp, sequence number, TIM (Traffic Indication Map), DTIM (Delivery Traffic Indication Message)
Default	Country, Supported Rates, Extended Support Rates, Vendor-Specific

Table 5.1: Beacon frame fingerprinting fields

Only those fields whose values remain constant across all the beacon frames transmitted by an AP are considered for fingerprinting. The dynamically changing fields represent the network parameters and load of an AP. Whereas the default configuration fields are similar in APs belonging to same OEM.[26]

Becon frame fingerprinting may not always be sufficient, hence the authors also enforce signal strength indicator fingerprinting to detect, for example, ETs belonging to same OEM (Original Equipment Manufacturer) as a result of SSID and BSSID forging (radiotap header includes information about the AP signal strength). The signal strength of two APs can never be the same and cannot be forged. Two APs located at two different places will always transmit different signal strengths. However, RSSI may oscillate due to hazy effects of radio signals.[26] Received Signal Strength Indicators are also utilized in [65] as an access point fingerprint.

Latest research suggests that the limitations of admin-side detection methods are mainly twofold, i.e., requiring dedicated equipment and lacking real-time protection.[32] The equipment has to be capable of specific operations to properly evaluate its surroundings and protect clients. Absence of real-time protection is typical in cases where automated evaluation and fingerprint collection are lacking and manual operation prevails.

On the other hand, client-side detection solutions may be solely targeted at individual Evil Twin models which makes it insufficient as a standalone high detection rate system. Most of the existing client-side solutions simply focus on the series model and that significantly decreases the overall detection rate.[32] The parallel approach has expanded with the availability of 4G or 5G networks to attackers, thus defenders can never rely on Evil Twin to be strictly in series.

In some publications, protocol modifications and timing/route measurements are considered another classification. Timing was also used to help distinguish wireless connection from a wired network.[41] Inherent transmission delay difference proposed by [30] between 4G mobile APs (10-20ms) and wired APs (less than 1ms) provides context to the measured RTT values between AP and the subsequent node in the search for an ongoing parallel ETA. Needless to say, this method may lead to false negatives and loss of relevance due to the shortened delay difference in 5G mobile communications.

However, most detections operate under the assumption that the transmission characteristics brought by evil hop are different from the legitimate networks.[32] Generally, time metrics may be influenced by pre-fetching, network topology, traffic volume, or network types. Evil Twin features such as packet forwarding cannot distinguish malicious rogue APs because they behave just like a legitimate AP.[19] Network fluctuation and route selection are also a potential interference along with advanced attackers tuning the parallel Evil Twin configuration.[32]

Gateway recognition is an important aspect of detecting unauthorized gateway switch. One could compare data traffic at different locations of the Wi-Fi network with a known authorized list, and check what type of network is the source coming from. As for routing, detection of a malicious rogue AP based on different reverse traceroute information (collected by a remote server) is shown in [19]. Comparing route paths to the same server via multiple target APs may reveal different gateways. Although, some route packets may be dropped on the firewall due to inherent security policies.

Recently, a client-side solution (2020) [32] called BiRe claims to have 100% detection rate in multi-model ETA scenarios. BiRe acknowledges many of the protection effectivity issues and comes up with exploiting TCP handshakes and NAT gateway behavior. The novel bi-directional SYN reflection is proposed to determine the existence of an ETA and differentiate among various attack models. Bi-directional SYN reflection refers to asynchronously emitting a specific SYN packet in both forward and reverse directions, so it is fully capable of working in current half-duplex networks. A pair of wireless adapters is employed to cooperatively initiate and monitor TCP handshake processes to see whether an Evil Twin prevents any communication. Intrusion can be detected when the expected TCP SYN-ACK packets are absent in the monitored TCP handshake processes.[32] The following image 5.8 presents a graphical representation of BiRe detection algorithm.

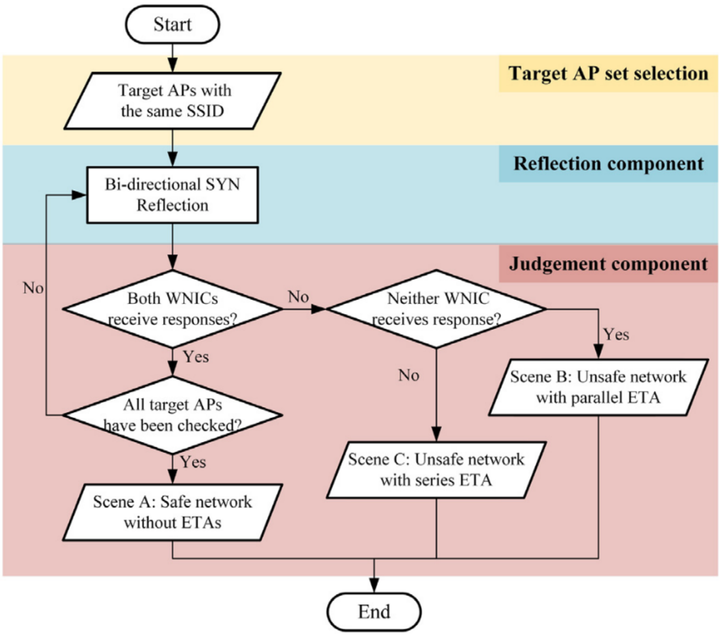

Start
Target AP set selection
Target APs with the same SSID
Reflection component
Bi-directional SYN Reflection
Judgement component
No
Both WNICs receive responses?
No
Neither WNIC receives response?
Yes
Yes
No
Scene B: Unsafe network with parallel ETA
All target APs have been checked?
Yes
Scene C: Unsafe network with series ETA
Scene A: Safe network without ETAs
End

Figure 5.8: Bi-directional SYN reflection detection algorithm.[32]

Additionally, the hardware information is also useful to defenders. Most modern networking hardware uses 802.11ac (Wi-Fi 5), altough it's not uncommon to see 802.11n (Wi-Fi 4) deployed in production as well. However, the vast majority of wireless pentesting hardware is limited to 802.11n and earlier, e.g. 802.11g or 802.11a. The Best practice is to look out for new ESS operating in the 802.11g/a or with a default BSSID of either 00:11:22:33:44:00 or 00:11:22:33:44:55.[61] Fixed BSSIDs in similar fashion are common for most publicly available RAP tools including the Wi-Fi Pineapple or EAPHammer (whose author implements it by choice to help detect its usage).

Finally, Rogue AP attacks are often executed using external hardware made by manufacturers such as Alfa Networks, TP-Link, and Panda Wireless.[61] In our research we also ended up using some of the Alfa Networks or Panda Networks equipment due to its compatibility and availability (See section 6.3).

As such, it's typically a good idea to monitor for devices that have OUIs that from these types of manufacturers.

5.4 Advancements

Previously we have discussed problems of automatic network selection in devices scanning for known or trusted networks. Over the time, multiple exploitation approaches and mitigations arose, mostly around the PNL. Each time a station connects to a wireless network, the network's ESSID is stored in the station's PNL. It represents an ordered list of every network that the station has connected to in the past with network-specific configuration for each entry attached.[58]

Not protecting the PNL entries sufficiently is a serious personal privacy risk. Apart from exposing configuration details for Evil Twin to act upon, the wiggle community WarDriving platform can provide geolocation for each SSID and show a map of networks visited. This could potentially lead to personal whereabouts and sensitive information being revealed via Open Source Intelligence (OSINT) investigation. One should consider active evaluation of PNL contents to delete unnecessary or vulnerable entries.

5.4.1 KARMA Attack

This Evil Twin attack variant has also been relevant for over a decade. It allows to target specific client devices regardless of target SSID visibility or range. KARMA takes advantage of clients that send direct probe requests to determine which wireless networks are nearby and what are their capabilities. RAP is faked through forged response to pose as a known AP listed in the broadcasted PNL of the victim device. The overall execution is more subtle since the attacker solely responds to the probes she finds appropriate and no SSID broadcast is necessary. [14] However, KARMA attacks are no longer as effective due to modern devices restricting directed probe requests for networks on their PNL. [77]

5.4.2 MANA Attack

KARMA was an improvement on the simple Evil Twin attack and fortunately, the handling of probe requests has changed for the better in modern devices. Currently, it is common for devices to expect a response on a broadcast probe first in order to prove validity of subsequent directed probe responses, or neglect the use of active probing altogether and rely on passive scanning.[79] Broadcast probe requests work almost exactly the same way as directed probes, but are sent with the SSID field set to NULL. Mana aims to preemptively respond to such broadcasts by reconstructing PNL of reachable devices and spoofing probe responses.

The rogue access point uses a series of hash tables to keep track of surrounding requests. Each probing device has its MAC address paired with the ESSIDs it has probed for. When the RAP receives a probe request, it first determines whether it's a broadcast or directed probe. If it's directed probe, the ESSID is saved under client MAC address and the AP responds back with a directed probe response. In the case of a broadcast probe, the access point responds with probe responses for each of the networks in that device's PNL.[59]

The default behaviour is to build up a view of each individual PNL and only respond to broadcasts with networks matched to that device. However, if stealth is of no concern, then the loud MANA variant increases the overall reach by building a unified global PNL. Attacker responds to broadcasts with every network every device has probed for to get a connection. This approach is based on the idea that client devices within close physical proximity to one another are likely to have at least some common entries in their PNLs.[77] Such attacks also allow to target a relatively secure device as a consequence of shared PNL entries with nearby vulnerable stations.

Attackers also need to make sure that devices with randomized MAC addresses are taken into consideration in PNL hashtables and MFACL. Tools often provide a way to specify fixed sections of the observed address (OUI or locally administered characteristics) and the randomized bits. The loud mode is less prone to proportionality issues where the effect of randomisation is not that significant due to the unified global PNL.[78]

5.4.3 Known Beacons Attack

Nowadays, most modern network managers have taken countermeasures against the KARMA and MANA attack by switching to passive scanning instead of arbitrarily sending probe requests. Network managers now wait to receive a beacon frame with a familiar ESSID before associating with a wireless network.

However, George Chatzisofroniou (Census labs, 2018) came with an approach of attempting to predict which open networks a device may be looking for by constructing a large list of known open Wi-Fi networks. For example, common instances are "iPhone" and public hotspots found in hotels, restaurants, coffee shops or transportation. In a more sophisticated version of the attack, the adversary may use a "dictionary" of common ESSIDs that the victim has likely connected to in the past. The adversary then transmits probe-response frames for all the networks in its lists.[7]

Additionally, in an attempt to reduce the amount of high noise Known Beacons attack is responsible for, the tool EAPHammer implements a burst variation that can be used to transmit a burst of forged beacon packets over a short period of time (ideally paired with MFACL). To protect themselves from this attack, users are strongly advised to make sure no ESSIDs of open networks are listed in their network manager's Preferred Network List.[59]

Chapter 6

Evil TwinBerries

In the previous chapters, we examined the underlying technical groundwork necessary to understand the Wi-Fi technology and the problems of the common attack surface. Additionally, we discussed the development of security countermeasures and overall best practices for Wi-Fi network configuration. This chapter is dedicated to the practical results of our offensive research and represents the physical manifestation of some of the described techniques, especially with focus on the Evil Twin attack and traffic decryption.

6.1 What is it

Ultimately, we decided to pursue a project that would best demonstrate the importance of a secured wireless perimeter. Target audience are not only cybersecurity professionals but technical students and the general public as well. With the growth rate of wireless communications it is necessary to educate everyone on concurrent privacy threats associated with Wi-Fi networks to help clients protect themselves.

Evil TwinBerries is an adjustable purple team[1] device actively used for the development and research of Wi-Fi security. The overall focus of the project prototype is to utilize commonly available hardware while preserving portability. In figure 6.1 we present the appearance of our prototype. OS environment is configured appropriately with selected wireless auditing tools to cover the up to date penetration testing toolkit.

Among other things, we built a portable rogue access point solution capable of a parallel handshake collection and a fully automated Evil Twin kill chain. To showcase the accessibility of such devices, we chose a publicly known attack surface and the common Raspberry Pi family (ARM architecture).

[1] Purple Team is referred to as an alignment of both blue team (defense oriented development, protection maintenance, threat detection) and the red team (penetration testing, vulnerability detection and reporting) classification of security operations.

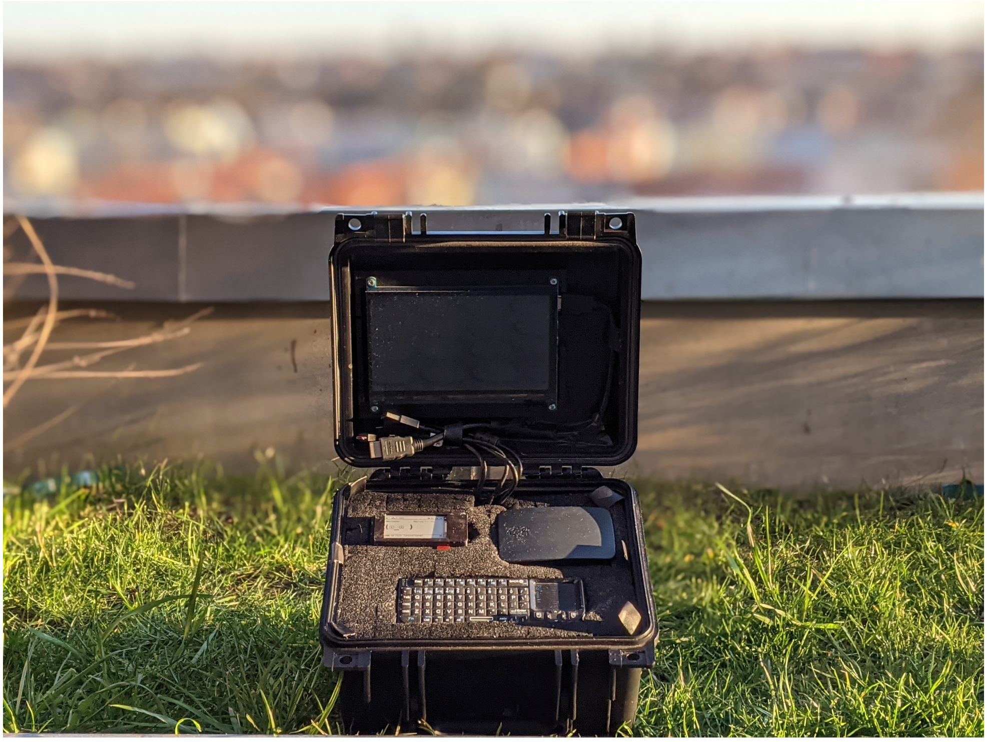


Figure 6.1: The Evil TwinBerries functional prototype.

The software approach is modular, realised through an automation pipeline where the modules are provisioned. Each module is a batch of Bash or Python scripts responsible for separate segments of the Evil Twin behaviour. It is possible to broaden the demonstration scope with any relevant Bash or Python code since their transformation into corresponding plug-in modules is supported. The project layout and process management allows to maintain stability in more dynamic scenarios. Modules are systematically loaded on startup and subsequently available for scheduling as part of the Evil TwinBerries execution cycle.

Other modules like the KR00K exploitation, ARP injection and a simplistic DNS amplification had their proof of concept deployed in a controlled academic research environment with reviewed process documentation (see section 6.4).

The design direction for Evil TwinBerries environment is to allow legal, highly customisable, up to date wireless bad actor demonstrations with proper logging and continuous integration support. Both commercial and academic parties are involved in this process to provide industry deployment with an established university foundation.

Community driven open source efforts provided a significant starting point for this research and they collectively form an amazing resource pool to represent the common Wi-Fi penetration testing framework. Practical write-ups, publications, software and hardware are available for everyone to take advantage of. Nowadays, building a versatile toolkit is not only possible but openly enabled by online resources often labeled for educational purposes. The recurring attack vectors require well educated attention, thus it makes sense to research these tools for the real educational value (see table 6.1).

6.2 Why does it exist

The device is made to demonstrate bad actor influence over a wireless network (not always on the network) for lawful protection development and organized educational purposes. We hope to inspire a security enhancing initiative by providing a detailed technical description of known exploitations with appropriate countermeasures. For this reason, we chose hardware from common manufacturers in an attempt to showcase the influence of rapid IoT growth on the seriousness and possibilities of wireless intrusion.

The protocol deployment and security of Wi-Fi networks has changed significantly over the last ten years (see figure 6.2 generated from wigle data). For example, the adoption of WPA3 is on its way although according to wiggle, the current majority of Wi-Fi networks are secured with WPA2. Encryption layout went from 30% WEP, 10% WPA, and 20% WPA2 at the end of 2011 to 5% WEP, 5% WPA and 69% WPA2 with WPA3 currently mapping 1020 (0.00%) instances (out of 725,661,586 networks).

Because the vulnerability surface for WPA2 is already well established it allows blue teams[26] to better prepare for possible intrusions by studying common exploitation vectors and securing the wireless perimeter accordingly (see section 4.2). To ease this process, we attempt to present the logic, availability and prevention, along with the malicious potential of known attacks to provide a technical background for defenders. At the same time, effective education of public on the security risks associated with Wi-Fi wireless networks is essential.

By the time of this writing, the severity of pandemic restrictions is forcing large portions of the world population to heavily utilize home based connections for sensitive activities or work. It is not guaranteed for a popular retail wireless router to provide sufficient protection against recurring attack models (sometimes even after the latest update), thus leaving the customer potentially vulnerable. Without the necessary knowledge, customers can hardly prevent or recognize visible signs of malicious intent as a direct result of Wi-Fi network compromise.

Preserving personal privacy is critical in the modern world and the wireless flow of data is no exception. For this reason, we are working with the antivirus

[26] The group responsible for defending an enterprise's use of information systems by maintaining its security posture [42]

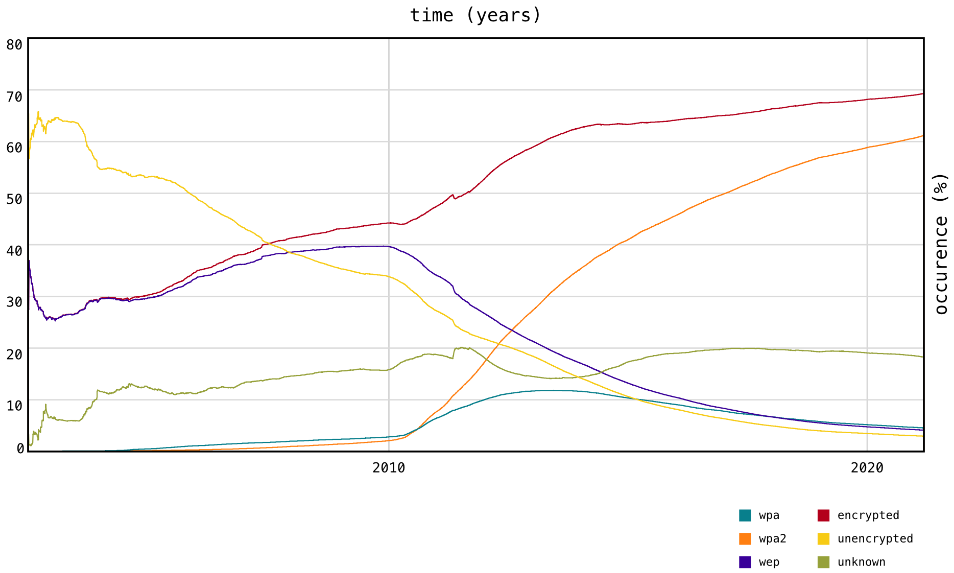


Figure 6.2: Wi-Fi encryption over time.

giant Avast software to fully utilize the device potential and help protect global communications.

It is important to monitor evolving trends in wireless security and understand the underlying technical aspect. By maintaining a relevant penetration testing toolkit we are able to evaluate the effectiveness of ongoing network fortification efforts. For large scale operations, the WarDriving principles and techniques are an industry proven foundation. However, support for the wireless AP operations en masse is limited by multiple legal constraints imposed by the Czech government. Needless to say, our project provides logging and functional design necessary for scaling out for broader statistical analysis.

6.3 How does it work

We developed our own configurable pipeline around the Evil Twin kill chain and set up a contemporary wireless auditing toolkit with respect to industry standards. ETA is ideal for demonstration purposes as it represents an unusual Man-in-the-Middle threat with multiple stages and variations. Corresponding protection approaches are also diverse in nature and complexity, hence the ET requires openness towards modifications.

Some automated methods are already present in the popular auditing tools and do not require the operator to know the fundamental principles or configuration details to construct a malicious RAP. This approach is not acceptable in our situation as it limits research flexibility and could later cause undesired patterns identifiable as Indicators of Compromise (IoC) for the Wireless Intrusion Detection Systems (WIDS) or victims to act upon.

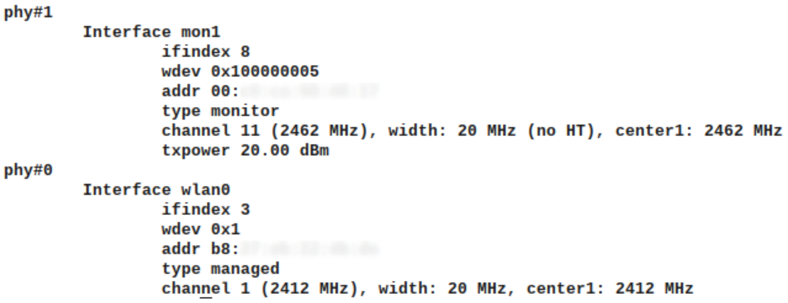
However, the wide range of techniques along with ease of access makes these tools worth exploring and definitely part of our penetration testing toolkit. We summarize most of our used tools in the table 6.1 at the end of this section. The proposed ET pipeline hopes to analyze and control otherwise implied behavior of an uneducated toolkit usage to truly test the audited defenses in the foreseeable future.

6.3.1 Rogue Access Point

In the following sections we describe the services used to construct a Rogue Access Point prototype with its own captive portal.

We built primarily on the Debian based operating system (Linux), more specifically on a customized Raspberry Pi OS image. Kali Linux was the backup operating system due to initial driver related crashes and the unnecessary resource load. Tools described in this chapter are listed in table 6.1. Due to the sensitive nature of this thesis, we decided to provide trimmed code examples and configuration files as attachments. The system environment is configured to allow for Evil Twin execution and other wireless attacks using external WNIC by Alfa Networks (AWUS036NHA[27]). Therefore, the loaded drivers are made for the Atheros AR9271 chipset.

First we verify WNIC connectivity using tools like iw (image 6.3). Once we made sure the external WNIC is connected, we create an additional wireless interface with the monitor mode enabled. Optionally, the launch of necessary services for LTE connectivity and GNSS system synchronization may take place.



```
phy#1
        Interface mon1
                ifindex 8
                wdev 0x100000005
                addr 00:[redacted]
                type monitor
                channel 11 (2462 MHz), width: 20 MHz (no HT), center1: 2462 MHz
                txpower 20.00 dBm
phy#0
        Interface wlan0
                ifindex 3
                wdev 0x1
                addr b8:[redacted]
                type managed
                channel 1 (2412 MHz), width: 20 MHz, center1: 2412 MHz
```

Figure 6.3: CLI configuration utility for wireless devices.

Target identification is performed through the tools for classification of surrounding wireless traffic as seen in 5.1. Once the target is identified and

[27] https://alfa.com.tw/products/awus036nha?variant=36473966166088

collateral damage contained (possibly by using MFACL), we scrape the victim access point characteristics and subsequently spoof identification directives in the ET configuration files. The goal is to appear indistinguishable from the victim AP to deceive automatic network selections nearby.

The constructed rogue access point needs to be capable of routing communication and control critical services such as the DHCP (Dynamic Host Configuration Protocol) provisioning or DNS (Domain Name System) handler.

Hostapd is the chosen user space daemon for access point and authentication servers. According to the documentation[28], it implements IEEE 802.11 access point management, IEEE 802.1X/WPA/WPA2/EAP Authenticators, RADIUS client, EAP server, and RADIUS authentication server.

The DHCP functionality toolkit is the dhcpd package (Internet Systems Consortium DHCP Server) Its configuration declares an authoritative server, lease times, IP ranges and other routing information (e.g., subnets, broadcast, routers, domain name servers). Finally, the path to dhcpd leases file is specified and represents a persistent database of leases that the server assigned. A common alternative to dhcpd is the dnsmasq as the easier-to-use option which also provides a DNS server. For the DNS sinkhole, we are using the dnsspoof[29] binary to forge replies to arbitrary DNS address or pointer queries on the LAN.

If coercion is necessary, Denial-of-Service attacks are executed against the target. The effectiveness of each method depends on the target environment. Initially we utilized the aireplay-ng for periodic deauthentication or mdk4 to adapt in complicated scenarios but the options are vast.

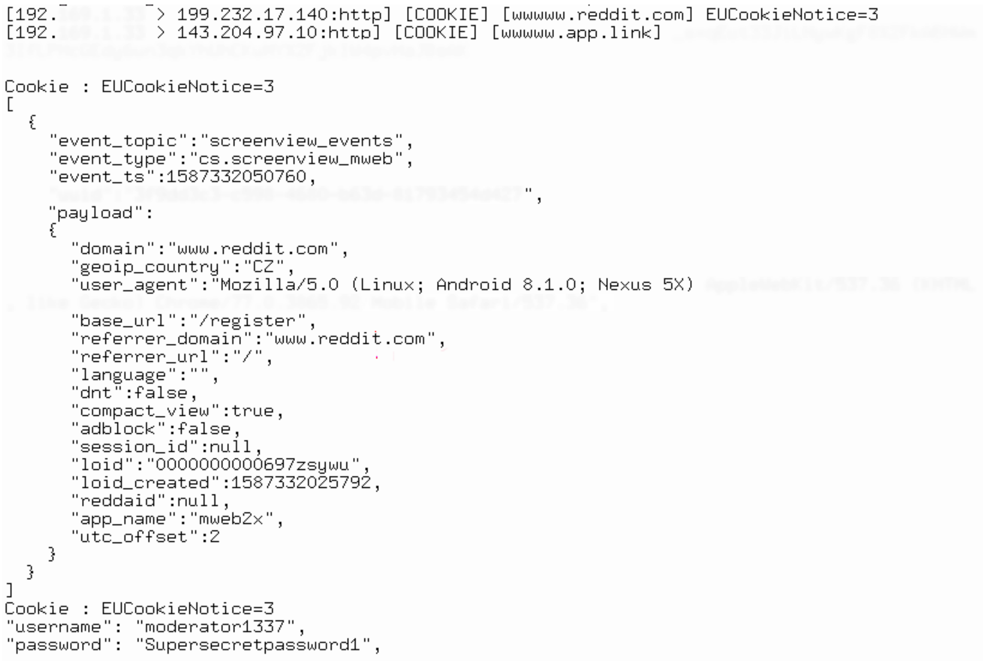
The captive portal, on the other hand, has no strict outline and the final product is open to interpretation. Phishing-based attacks have their functionality and graphical design influenced by online collections of popular commercial standards. Other specialized portals focus more on the underlying impact of operations embedded into them (see section 5.2.3).

However, it is necessary to establish additional firewall rules (iptables) and a webserver (lighttpd) to provide the initial platform and restrict the users from the network while provisioning captive portal with a landing page. Our actual captive portal website was designed in cooperation with Avast Software and its implementation remains confidential. Other generic tools may provide simplistic login form in an attempt to match the intercepted passwords against previously captured victim handshakes.

In other exploitation attempts we also redirected traffic to SSLStrip with DNS proxy enabled by using SSLstrip2 and dns2proxy. Image 6.4 shows a sample log entry with stolen login credentials from the popular website Reddit.

[28] `https://w1.fi/hostapd/`

[29] `https://linux.die.net/man/8/dnsspoof`



```
[192.[redacted] > 199.232.17.140:http] [COOKIE] [wwwww.reddit.com] EUCookieNotice=3
[192.[redacted] > 143.204.97.10:http] [COOKIE] [wwwww.app.link] [redacted]

Cookie : EUCookieNotice=3
[
  {
    "event_topic":"screenview_events",
    "event_type":"cs.screenview_mweb",
    "event_ts":1587332050760,
    [redacted]",
    "payload":
    {
      "domain":"www.reddit.com",
      "geoip_country":"CZ",
      "user_agent":"Mozilla/5.0 (Linux; Android 8.1.0; Nexus 5X) [redacted]
      "base_url":"/register",
      "referrer_domain":"www.reddit.com",
      "referrer_url":"/",
      "language":"",
      "dnt":false,
      "compact_view":true,
      "adblock":false,
      "session_id":null,
      "loid":"0000000000697zsywu",
      "loid_created":1587332025792,
      "reddaid":null,
      "app_name":"mweb2x",
      "utc_offset":2
    }
  }
]
Cookie : EUCookieNotice=3
"username": "moderator1337",
"password": "Supersecretpassword1",
```

Figure 6.4: Stolen credentials using the SSLStrip2 method.

6.3.2 Pipeline

To allow for efficient testing, we constructed a fully automated self-reconfiguring ET pipeline to subsequently deploy the scheduled stages. Rogue Access Point configuration is based on the parsed target characteristics and the goal is to pose as the original device by spoofing its key identification directives. This entire process is initiated by submitting the victim information (SSID) to any of the listening kill chain service invokers (BlueTooth server daemon mostly).

Additionally, we employ a dynamic plug-in module inclusion mechanism (allows more stages), documented logging policy and system/networking environment check. Logging supports different severity levels and is structured through stages into multiple files. All spawned processes get checked for possible runtime issues and eventually indexed through a PID based stack structure. Exit handlers return well documented codes with attached messages and ultimately ensure the final graceful termination cleanup.

The language of choice for automation and orchestration is Bash (Google Shell Style Guide compliant). Python helps make more complex functionality practical and fill in for Bash limitations. Intercommunication of targets between pwnagotchi and the demonstration invoker is possible using our custom pwnagotchi bluetooth communication module (server&client).

The handshake collection and analysis is an extended scenario realized by a combination of bettercap (pwnagotchi) and a GPU accelerated hashcat cloud instance. The Selection of the VM platform provider depends on its billing plan value, control and availability of a GPU intensive resource pool. For example, Amazon EC2 P3 instances[30] or Azure Machine Learning NC-series instances[31].

6.3.3 Casing

As previously stated, the overall focus with the initial prototype is to utilize commonly available hardware to provide a digestable presentation of the project. All while preserving the core project values, the selected waterproof briefcase is a tribute to the great but no longer maintained Warberry Pi project[32]. Our public relations friendly prototype consists of the following items:

Wireless Toolbox: Raspberry Pi 3

WNIC: Alfa AWUS036NHA (atheros chipset and monitor mode)

Handshake collection: Raspberry Pi Zero WH (pwnagotchi build)

Power management: ATX style PCB by Pi Supply (soldering required)

LCD Touchscreen: Waveshare 13,3" HDMI IPS

Physical impact protection: Layered foam template

Photographs 6.5 and 6.6 provide inside look at the hardware layout and a size comparison.

6.4 Testing

Previously in section 6.3, we described the technical details of our prototype. Due to the sensitive legal nature of this thesis, the testing process requires a specific and controlled lab environment. Therefore, we have deployed a separate wireless network on the least used nearby Wi-Fi channel with only testing devices' MAC addresses allowed access to the network. Additionally, some testing methods and results remain private due to the partially confidential industry collaboration with Avast Software. In the table 6.1, we summarize most of the tools we considered essential during development.

First, we verified that the Evil TwinBerries device is capable of intercepting and producing 802.11 traffic. To test this, we have successfully executed the ARP poisoning attack with concurrent telnet snooping using the Scapy library.

[30] `https://aws.amazon.com/blogs/aws/new-amazon-ec2-instances-with-up-to-8-nvidia-tesla-v100-gpus-p3/`

[31] `https://docs.microsoft.com/en-us/azure/virtual-machines/sizes-gpu`

[32] `https://github.com/secgroundzero/warberry`

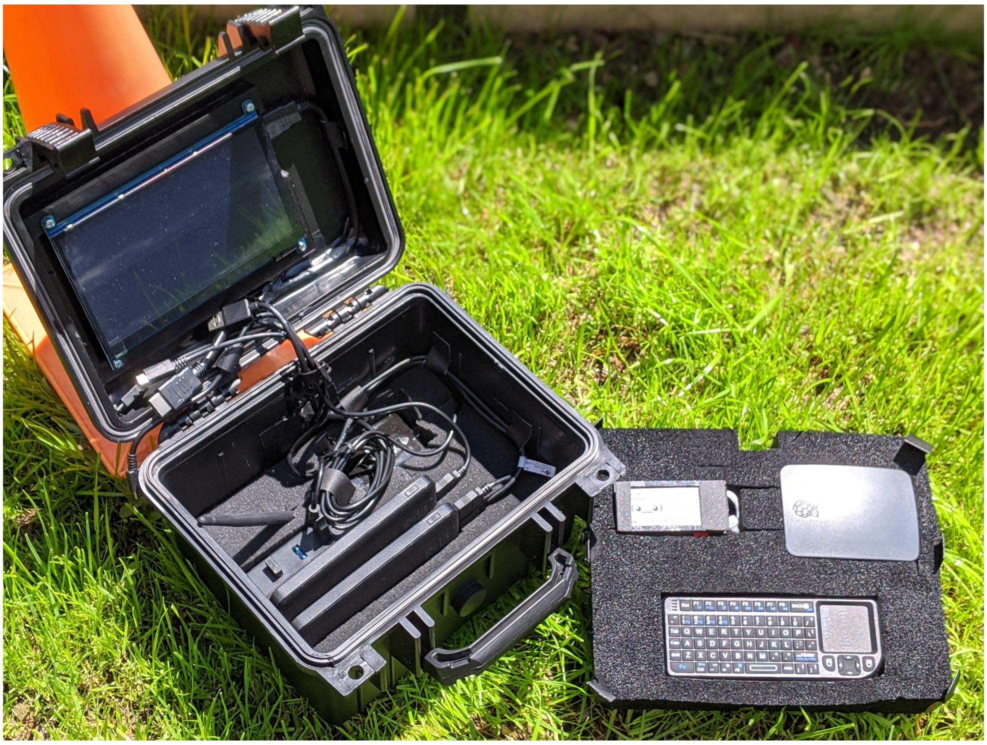


Figure 6.5: Evil TwinBerries inner look.

The consequences of such attacks were stolen credentials as well as the ability to hijack network flow.

Then, we monitored the prototype's physical properties and Denial-of-Service capability by implementing some common techniques (Deauthentication Attack, TCP/SYN Flood and DNS Multiplication Attack) using both the Scapy and Libpcap libraries. Factors such as overheating and power consumption came into play once we increased the attack intensity and multiple physical adjustments had to be done.

The attacker was the Evil TwinBerries staging configuration (Raspberry Pi 3B) with a USB connected WNIC (Alfa AWUS036NHA) in monitor mode. Similarly, the victims were a different Raspberry Pi 3B (both running the Raspberry Pi Buster OS image) and the jailbroken Nexus5X Android phone. The isolated wireless environment was simulated mostly using the vulnerable TL–WR841N router AP in factory settings.

To confirm our theoretical Evil Twin research, we constructed a rogue access point and flooded the victim with spoofed deauthentication frames. This made our access point a better connection candidate and allowed us to successfully attack the Nexus5X device. Moreover, we deployed multiple configurations and threat model variations, e.g. SSLstrip2 or captive portal phishing. Compared to other solutions, we primarily focused on a specific goal

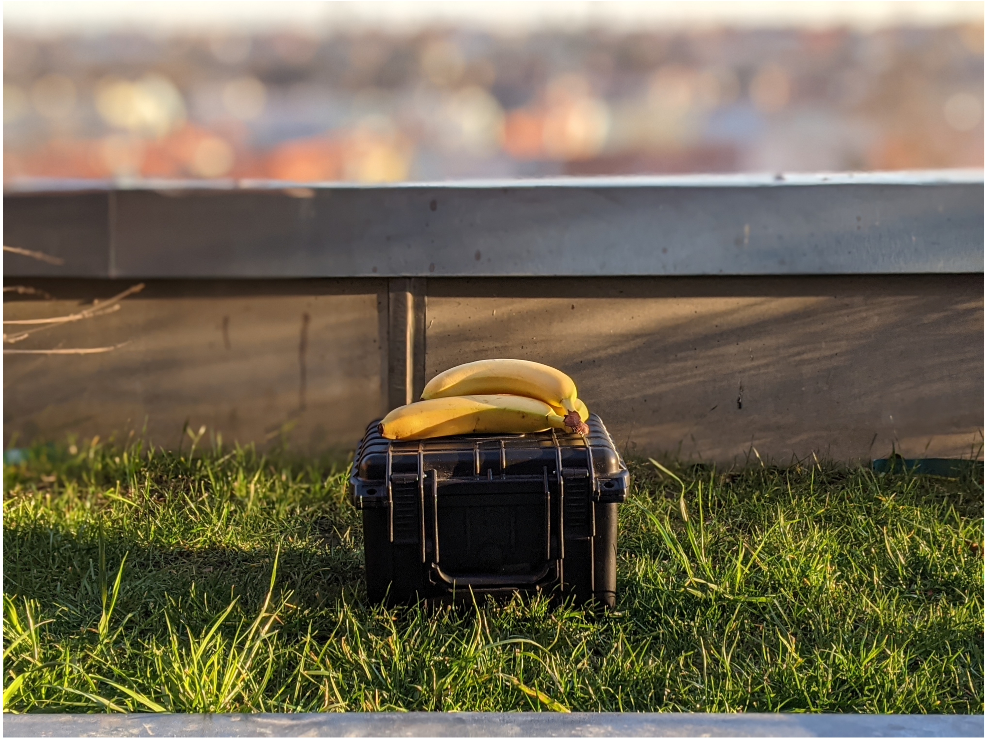


Figure 6.6: Evil TwinBerries size comparison.

of replicating our theoretical findings while exploring the potential of modern ARM-based platforms.

Unfortunately, later in the development, we observed an issue with the WNIC not properly capturing data packets while in monitor mode. Due to limited time constraints, we were only able to narrow down the cause of this issue and the KR00K exploitation pipeline has not been fully tested for an all-zero key packet decryption yet (despite its seemingly complete Evil TwinBerries module).

Reconnaisance	
Airmon-ng	`https://www.aircrack-ng.org/doku.php?id=airmon-ng`
Kismet	`https://kismetwireless.net/`
Bettercap	`https://www.bettercap.org/`
RF signal analysis	
GNU Radio	`https://www.gnuradio.org/`
SDR receiver	`https://gqrx.dk/`
URH	`https://github.com/jopohl/urh`
Infrastructure	
AP	`https://w1.fi/hostapd/`
DNS	`https://linux.die.net/man/8/dnsspoof`
Firewall	`https://linux.die.net/man/8/iptables`
Penetration testing	
Aircrack-ng	`https://aircrack-ng.org/`
Airgeddon	`https://github.com/v1s1t0r1sh3r3/airgeddon`
EAPHammer	`https://github.com/s0lst1c3/eaphammer`
Wifite2	`https://github.com/derv82/wifite2`
KRACK	`https://github.com/vanhoefm/krackattacks-scripts`
KR00K	`https://github.com/hexway/r00kie-kr00kie`

Table 6.1: Toolkit

Conclusion

We have outlined the principles of Wi-Fi technology and the popular security protocols/mechanisms. Later, we researched the attack surface associated with this widespread technology to classify exploitation methods and showcase established threats. The analysis was focused on the Wi-Fi attack surface where we excluded the legacy protocols due to their broken security and deprecation. We then discuss recurring threats along with the general protection guidelines for modern Wi-Fi networks to help network administrators and Wi-Fi consumers better secure their wireless perimeter.

The next step was a detailed analysis of the unique Evil Twin attack vector with the focus on exploitability, advancements, and the available detection schemes. Based on our findings, we discuss the pillars of ET and its capabilities in the current wireless environment as well as the available detection schemes. The important focus was to put in context the attacker's point of view with known effective countermeasures to further educate on the risks and protections of Wi-Fi communication.

We also presented a device specialized on Wi-Fi security testing with appropriate technical description. To showcase the accessibility of such devices, we chose a publicly known attack surface and the common Raspberry Pi family (ARM architecture). Among other things, we were able to demonstrate a portable rogue access point solution capable of a parallel handshake collection and a fully automated Evil Twin kill chain.

The chosen software approach is modular where each module is a batch of Bash or Python scripts responsible for separate segments of the Evil Twin behavior. It is possible to broaden the demonstration scope with any relevant Bash or Python code since their transformation into corresponding plug-in modules is supported. The project layout and process management allow maintaining stability in more dynamic scenarios. Additionally, the device is capable of utilizing any of the other contemporary toolkit options.

Enterprise advancements and WPA3 are a subject for future work. During testing, we have observed an issue with the WNIC not properly capturing data packets while in monitor mode. Due to time constraints, we were only able to narrow down this issue and its resolution remains a part of future work. Further improvements involve integration to existing business infrastructure and corresponding hardware modifications based on issues observed in this prototype (such as better power and heat management). Additionally, we are in the process of ongoing legal negotiations about a controlled deployment for analytical purposes.

Bibliography

[1] Aircrack-ng. *Aircrack-ng docs - deauthentication*, 2010. Accessed: 14th May 2020.
URL `https://aircrack-ng.org/doku.php?id=deauthentication`

[2] Aktin, Devin. *802.11i Key Management v3 - Certified Wireless Network*, 2005. Accessed: 18th January 2020.
URL `https://cwnp.com/uploads/802-11i_key_management.pdf`

[3] Ali, Suzan, Osman, Tousif, Mannan, Mohammad, and Youssef, Amr. *On Privacy Risks of Public WiFi Captive Portals.* Lecture Notes in Computer Science, p. 80–98, 2019. doi:10.1007/978-3-030-31500-9_6.

[4] Benton, Kevin. *The Evolution of 802.11 Wireless Security - Kevin Benton*, 2010. Accessed: 18th January 2020.
URL `https://benton.pub/research/benton_wireless.pdf`

[5] Bongard, Dominique. *Offline bruteforce attack on WiFi Protected Setup*, 2014. Accesed 25th Aug 2021.
URL `http://archive.hack.lu/2014/Hacklu2014_offline_bruteforce_attack_on_wps.pdf`

[6] Butler, Jane. Wireless networking in the developing world: a practical guide to planning and building low-cost telecommunicaitons infrastructure. Limehouse Book Sprint Team, 2013.

[7] Chatzisofroniou, George. *The Known Beacons Attack (34th Chaos Communication Congress)*, 2018. Accessed: 4th July 2020.
URL `https://census-labs.com/news/2018/02/01/known-beacons-attack-34c3/`

[8] CISA. *Security tip (st05-003) - Securing Wireless Networks*, 2020. Accessed 21st April 2021.
URL `https://us-cert.cisa.gov/ncas/tips/ST05-003`

[9] Cloudflare. *What Is The OSI Model?*, 2019. Accessed: 15th February 2020.
URL https://cloudflare.com/learning/ddos/glossary/open-systems-interconnection-model-osi/

[10] Dabrowski, Adrian, Merzdovnik, Georg, Kommenda, Nikolaus, and Weippl, Edgar. *Browser History Stealing with Captive Wi-Fi Portals.* 2016 IEEE Security and Privacy Workshops (SPW), 2016. doi:10.1109/spw.2016.42. Accessed: 15th November 2020.

[11] Dai Zovi, D.A. and Macaulay, S.A. *Attacking automatic wireless network selection.* Proceedings from the Sixth Annual IEEE Systems, Man and Cybernetics (SMC) Information Assurance Workshop, 2005., 2005. doi:10.1109/iaw.2005.1495975.

[12] Diffen. *WPA2 vs WPA3*, 2019. Accessed: 21st March 2021.
URL https://diffen.com/difference/WPA2_vs_WPA3

[13] DoJ. *Documents and Resources from the October 4, 2018 Press Conference*, 2018. Accessed 3rd February 2021.
URL https://justice.gov/opa/documents-and-resources-october-4-2018-press-conference

[14] Dormann, Will. *Instant karma might still get you*, 2015. Accessed: 22nd March 2020.
URL https://insights.sei.cmu.edu/blog/instant-karma-might-still-get-you/

[15] Ebbecke, Philipp. *Protected Management Frames enhance Wi-Fi® network security*, 2020. Accessed 20th April 2021.
URL https://wi-fi.org/beacon/philipp-ebbecke/protected-management-frames-enhance-wi-fi-network-security

[16] ESET. *Cybersecurity in the home – securing your WiFi router*, 2017. Accessed 12th March 2021.
URL https://eset.com/au/about/newsroom/press-releases1/eset-blog/cybersecurity-in-the-home-securing-your-wifi-router/

[17] ETSI. *EN 301 893 - ETSI*, 2017. Accessed: 24th April 2020.
URL https://etsi.org/deliver/etsi_EN/301800_301899/301893/02.01.01_60/en_301893v020101p.pdf

[18] Fluhrer, Scott, Mantin, Itsik, and Shamir, Adi. *Weaknesses in the key scheduling algorithm of RC4*, 2001. Accessed: 18th January 2020.
URL https://link.springer.com/content/pdf/10.100%2F3-540-45537-X_1.pdf

[19] Hsu, Fu-Hau, Hsu, Yu-Liang, and Wang, Chuan-Sheng. *A solution to detect the existence of a malicious rogue AP*. Computer Communications, 142-143:62–68, 2019. doi:10.1016/j.comcom.2019.03.013.

[20] Hsu, Fu-Hau, Wang, Chuan-Sheng, Hsu, Yu-Liang, Cheng, Yung-Pin, and Hsneh, Yu-Hsiang. *A client-side detection mechanism for evil twins*. Computers & Electrical Engineering, 59:76–85, 2017. doi: 10.1016/j.compeleceng.2015.10.010.

[21] Huang, Jianyong, Seberry, Jennifer, Susilo, Willy, and Bunder, Martin. *Security analysis of Michael: The IEEE 802.11i message INTEGRITY CODE*. Embedded and Ubiquitous Computing – EUC 2005 Workshops, p. 423–432, 2005. doi:10.1007/11596042_44.

[22] Hurley, Chris. WarDriving drive, detect, defend: a guide to wireless security. Syngress Publishing, 2004.

[23] IEEE. *IEEE 802.11-2016 - 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*. Accessed 20th April 2021.
URL `https://standards.ieee.org/standard/802_11-2016.html`

[24] IEEE. *IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines*. Internet Requests for Comments, 2003. Accessed: 16th March 2021.
URL `https://tools.ietf.org/html/rfc3580`

[25] IPCisco. *EAPoL: 802.1X: Extensible Authentication Protocol over LAN*, 2021. Accessed: 26th March 2021.
URL `https://ipcisco.com/lesson/eapol-extensible-authentication-protocol-over-lan/`

[26] Jain, Vineeta, Laxmi, Vijay, Gaur, Manoj Singh, and Mosbah, Mohamed. *ETGuard: Detecting D2D attacks using wireless Evil Twins*. Computers & Security, 83:389–405, 2019. doi:10.1016/j.cose.2019.02.014.

[27] Kalniņš, Rūdolfs, Puriņš, Jānis, and Alksnis, Gundars. *Security evaluation of wireless network access points*. Applied Computer Systems, 21(1):38–45, 2017. doi:10.1515/acss-2017-0005.

[28] Kaspersky. *What is VPN? How It Works, Types of VPN*, 2021. Accessed 5th May 2021.
URL `https://kaspersky.com/resource-center/definitions/what-is-a-vpn`

[29] Kiefer, Christopher, Kessem, Limor, and Wong, Reginald. *Leading Magecart Group Targeting Captive Wi-Fi Users via L7 Routers*, 2019. Accessed: 15th November 2020.
URL https://securityintelligence.com/posts/leading-magecart-group-targeting-captive-wi-fi-users-via-l7-routers/

[30] Kim, Iluk, Seo, Jungtaek, Shon, Taeshik, and Moon, Jongsub. *A novel approach to detection of mobile rogue access points.* Security and Communication Networks, 2013. doi:10.1002/sec.756.

[31] Linksys. *Linksys Official Support - Connecting devices using Wi-Fi Protected Setup™ (WPS) on your Linksys router*, 2015. Accessed: 25th May 2020.
URL https://linksys.com/us/support-article?articleNum=143300

[32] Lu, Qian, Jiang, Ruobing, Ouyang, Yuzhan, Qu, Haipeng, and Zhang, Jiahui. *BiRe: A client-side Bi-directional SYN Reflection mechanism against multi-model evil twin attacks.* Computers & Security, 88:101618, 2020. doi:10.1016/j.cose.2019.101618.

[33] Lynch, Vincent. *Let's Encrypt Has Issued Certificates to Over 14,000 PayPal Phishing Sites*, 2020. Accessed: 9th January 2021.
URL https://thesslstore.com/blog/lets-encrypt-phishing/

[34] Margaritelli, Simone. *Pwning WPA/WPA2 Networks With Bettercap and the PMKID Client-Less Attack*, 2019. Accessed: 8th November 2020.
URL https://evilsocket.net/2019/02/13/Pwning-WiFi-networks-with-bettercap-and-the-PMKID-client-less-attack/

[35] Martin, Lockheed. *Cyber Kill Chain®*, 2014. Accessed: 22nd March 2020.
URL https://lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html

[36] MDN, Contributors. *Strict-Transport-Security*, 2021. Accessed: 9th April 2021.
URL https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security

[37] MikroTik. *Manual:Interface/Wireless*, 2021. Accessed: 25th May 2020.
URL https://wiki.mikrotik.com/wiki/Manual:Interface/Wireless

[38] mitmproxy. *How mitmproxy works.* Accessed: 7th March 2020.
URL https://docs.mitmproxy.org/stable/concepts-howmitmproxyworks/

[39] MITRE. *ATT&CK for Enterprise Introduction*, 2017. Accessed: 22nd March 2020.
URL `https://attack.mitre.org/resources/enterprise-introduction/`

[40] Mohanan, Vasuky, Budiarto, Rahmat, and Aldmour, Ismat. Powering the internet of things with 5G networks. IGI Global, 2018.

[41] Nakhila, Omar, Amjad, Muhammad Faisal, Dondyk, Erich, and Zou, Cliff. *Gateway independent user-side wi-fi Evil Twin Attack detection using virtual wireless clients*. Computers & Security, 74:41–54, 2018. doi: 10.1016/j.cose.2017.12.009.

[42] NIST. *Computer Security Resource Center - Glossary*, 2018. Accessed 3rd May 2021.
URL `https://csrc.nist.gov/glossary/`

[43] NSA. *WPA3 will Enhance Wi-Fi Security*, 2018. Accessed: 18th January 2020.
URL `https://apps.nsa.gov/iaarchive/library/reports/wpa3-will-enhance-wi-fi-security.cfm`

[44] Nve, Leonardo. *OFFENSIVE: Exploiting DNS Servers Changes*, 2014. Accessed: 3rd June 2020.
URL `https://blackhat.com/asia-14/briefings.html#Nve`

[45] Parthipattu, Srinivasaraghavan. *802.11 Sniffer Capture Analysis - Management Frames and Open Auth*, 2020. Accessed: 24th April 2020.
URL `https://community.cisco.com/t5/wireless-mobility-documents/802-11-sniffer-capture-analysis-management-frames-and-open-auth/ta-p/3120622`

[46] Petrov, Ivan, Peskov, Denis, Coard, Gregory, Chung, Taejoong, Choffnes, David, Levin, Dave, Maggs, Bruce M., Mislove, Alan, and Wilson, Christo. *Measuring the rapid growth of HSTS and HPKP Deployments*, 2018. Accessed: 9th January 2021.
URL `https://cs.umd.edu/sites/default/files/scholarly_papers/Petrov,%20Ivan_1801.pdf`

[47] Project, The Chromium. *HTST submission requirements*. Accessed: 9th January 2021.
URL `https://hstspreload.org/#submission-requirements`

[48] RF-Wireless. *RF Wireless World - radius vs diameter-difference between radius and diameter protocols*, 2018. Accessed: 21st March 2021.
URL `https://rfwireless-world.com/Terminology/Radius-protocol-vs-Diameter-protocol.html`

[49] Rigney, Willens, Livingston, Rubens, Merit, Simpson, and Daydreamer. *Remote Authentication Dial In User Service (RADIUS).* Internet Requests for Comments, 2000. Accessed: 21st March 2021.
URL `https://tools.ietf.org/html/rfc2865#page-5`

[50] Roethlisberger, Daniel. *SSLsplit - transparent SSL/TLS interception*, 2019. Accessed: 7th March 2020.
URL `https://roe.ch/SSLsplit`

[51] Rowley, Jeremy. *What is certificate pinning?*, 2020. Accessed: 4th April 2021.
URL `https://digicert.com/dc/blog/certificate-pinning-what-is-certificate-pinning/`

[52] RSA. *RSA Authentication Manager Documentation*, 2021. Accessed: 26th March 2021.
URL `https://community.rsa.com/t5/rsa-authentication-manager/tkb-p/authentication-manager-documentation`

[53] Russell, Aaron. *What Is an X.509 Certificate?*, 2021. Accessed: 12th April 2021.
URL `https://ssl.com/faqs/what-is-an-x-509-certificate/`

[54] Ryan, Gabriel. *The black art of wireless post-exploitation - bypassing port-based access controls using indirect wireless pivots.* Accessed: 9th January 2021.
URL `https://blog.gdssecurity.com/labs/2017/8/31/whitepaper-the-black-art-of-wireless-post-exploitation-bypas.html`

[55] Ryan, Gabriel. *whitepaper: Identifying rogue access point attacks using probe response patterns and signal strength*, 2017. Accessed: 10th January 2021.
URL `https://blog.gdssecurity.com/labs/2017/1/17/whitepaper-identifying-rogue-access-point-attacks-using-prob.html`

[56] Ryan, Gabriel. *Advanced Wireless Attacks Against Enterprise Networks (AWAE) (v3.0.1)*, 2019.
URL `https://solstice.sh/workshops-advanced-wireless-attacks/`

[57] Ryan, Gabriel. *II. Attacking And Gaining Entry To WPA2-EAP Wireless Networks*, 2019. Accessed: 9th January 2021.
URL `https://solstice.sh/ii-attacking-and-gaining-entry-to-wpa2-eap-wireless-networks/`

[58] Ryan, Gabriel. *Modern Wireless Tradecraft Pt I - Basic Rogue AP Theory - Evil Twin and Karma Attacks*, 2019. Accessed: 9th January 2021.

URL `https://posts.specterops.io/modern-wireless-attacks-pt-i-basic-rogue-ap-theory-evil-twin-and-karma-attacks-35a8571550ee`

[59] Ryan, Gabriel. *Modern Wireless Tradecraft Pt II - MANA and Known Beacon Attacks*, 2019. Accessed: 6th January 2021.
URL `https://posts.specterops.io/modern-wireless-attacks-pt-ii-mana-and-known-beacon-attacks-97a359d385f9`

[60] Ryan, Gabriel. *Modern Wireless Tradecraft Pt III - Management Frame Access Control Lists (MFACLs)*, 2019. Accessed: 15th January 2021.
URL `https://posts.specterops.io/modern-wireless-tradecraft-pt-iii-management-frame-access-control-lists-mfacls-22ca7f314a38`

[61] Ryan, Gabriel. *Modern Wireless Tradecraft Pt IV - Tradecraft and Detection*, 2019. Accessed: 15th January 2021.
URL `https://posts.specterops.io/modern-wireless-tradecraft-pt-iv-tradecraft-and-detection-d1a95da4bb4d`

[62] Smith, Michael, Disselkoen, Craig, Narayan, Shravan, Brown, Fraser, and Stefan, Deian. *Browser history re:visited*, 2018. Accessed: 15th November 2020.
URL `https://usenix.org/conference/woot18/presentation/smith`

[63] Steube, Jens Atom. *hashcat advanced password recovery*, 2018. Accessed: 8th November 2020.
URL `https://hashcat.net/forum/thread-7717.html`

[64] Sârbu, Annamaria and Neagoie, Dumitru. *Wi-Fi jamming using software defined radio.* International conference KNOWLEDGE-BASED ORGANIZATION, 26(3):162–166, 2020. doi:10.2478/kbo-2020-0132.

[65] Tang, Zhanyong, Zhao, Yujie, Yang, Lei, Qi, Shengde, Fang, Dingyi, Chen, Xiaojiang, Gong, Xiaoqing, and Wang, Zheng. *Exploiting Wireless Received Signal Strength Indicators to Detect Evil-Twin Attacks in Smart Homes.* Mobile Information Systems, 2017:1–14, 2017. doi:10.1155/2017/1248578.

[66] Tsow, Alex, Jakobsson, Markus, Yang, Liu, and Wetzel, Susanne. *Warkitting: The Drive-by Subversion of Wireless Home Routers.* Journal of Digital Forensic Practice, 1(3):179–192, 2006. doi:10.1080/15567280600995832.

[67] Tung, Liam, 2021. Accessed 14th April 2021.
URL `https://msn.com/en-us/news/technology/google-releases-chrome-90-with-https-by-default-and-security-fixes/ar-BB1fGiWd`

[68] (US), Department of Homeland Security. *What is Personally Identifiable Information?*
URL `https://dhs.gov/privacy-training/what-personally-identifiable-information`

[69] (US), Department of Homeland Security. *A Guide to Securing Networks for Wi-Fi (IEEE 802.11 Family)*, 2017. Accessed 21st April 2021.
URL `https://us-cert.cisa.gov/sites/default/files/publications/A_Guide_to_Securing_Networks_for_Wi-Fi.pdf`

[70] Vanhoef, Mathy and Piessens, Frank. *Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2*, 2017. Accessed: 14th February 2020.
URL `https://papers.mathyvanhoef.com/ccs2017.pdf`

[71] Vanhoef, Mathy and Piessens, Frank. *Release the Kraken.* Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, 2018. doi:10.1145/3243734.3243807. Accessed: 14th February 2020.

[72] Vanhoef, Mathy and Ronen, Eyal. *Dragonblood: Analyzing the Dragonfly Handshake of WPA3 and EAP-pwd.* Cryptology ePrint Archive, Report 2019/383, 2019. Accessed: 4th January 2021.
URL `https://eprint.iacr.org/2019/383`

[73] Vergès, François. *Wireshark: How to check if a data frame is sent using 802.11n*, 2015. Accessed 25th October 2020.
URL `https://semfionetworks.com/blog/wireshark-how-to-check-if-a-data-frame-is-sent-using-80211n/`

[74] Vink, Mark. *A comprehensive taxonomy of wi-fi attacks*, 2020. Accessed 26th April 2021.
URL `https://ru.nl/publish/pages/769526/mark_vink.pdf`

[75] Want, Roy. *Near field communication.* IEEE Pervasive Computing, 10(3):4–7, 2011. doi:10.1109/MPRV.2011.55.

[76] Wasil, Dean, Nakhila, Omar, Bacanli, Salih Safa, Zou, Cliff, and Turgut, Damla. *Exposing vulnerabilities in mobile networks: A mobile data consumption attack.* 2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), 2017. doi: 10.1109/mass.2017.76.

[77] White, Dominic. *Improvements in rogue ap attacks – mana 1/2*, 2015. Accessed: 11th May 2020.
URL `https://sensepost.com/blog/2015/improvements-in-rogue-ap-attacks-mana-1/2/`

[78] White, Dominic. *Handling randomised mac addresses in mana*, 2016. Accessed: 11th January 2021.
URL `https://sensepost.com/blog/2016/handling-randomised-mac-addresses-in-mana/`

[79] White, Dominic and Villiers, Ian de. *Manna from Heaven; Improving the state of rogue AP attacks*, 2014. Accessed: 18th December 2020.
URL `https://media.defcon.org/DEF%20CON%2022/DEF%20CON%2022%20presentations/DEF%20CON%2022%20-%20Dominic-White-Ian-de-Villiers-Manna-from-Heaven.pdf`

[80] Wi-Fi, Alliance. *Wi-Fi Protected Access: Strong, standards-based, interoperable security for today's Wi-Fi networks*, 2003. Accessed: 19th January 2020.
URL `https://dell.com/downloads/global/shared/whitepaper_wi-fi_security4_29_03.pdf`

[81] Wi-Fi, Alliance. *Device Provisioning Protocol specification*, 2018. Accessed: 16th May 2020.
URL `https://wi-fi.org/download.php?file=/sites/default/files/private/Device_Provisioning_Protocol_Specification_v1.1_1.pdf`

[82] Wi-Fi, Alliance. *Fi Easy Connect*, 2018. Accessed: 17th May 2020.
URL `https://wi-fi.org/discover-wi-fi/wi-fi-easy-connect`

[83] Wi-Fi, Alliance. *Wi-Fi Alliance® introduces Wi-Fi 6*, 2018. Accessed 20th April 2021.
URL `https://wi-fi.org/news-events/newsroom/wi-fi-alliance-introduces-wi-fi-6`

[84] Wi-Fi, Alliance. *WPA3 Security Considerations - Wi-Fi Alliance*, 2019. Accessed 22nd April 2021.
URL `https://wi-fi.org/download.php?file=/sites/default/files/private/WPA3_Security_Considerations_201911.pdf`

[85] Wi-Fi, Alliance. *Security*, 2020. Accessed: 24th April 2020.
URL `https://wi-fi.org/discover-wi-fi/security`

[86] Wi-Fi, Alliance. *WPA3 Specification version 3.0 - Wi-Fi Alliance*, 2020. Accessed 23rd April 2021.
URL `https://wi-fi.org/download.php?file=/sites/default/files/private/WPA3_Specification_v3.0.pdf`

[87] Čermák, Miloš and Lipovsky, Robert. *Beyond KrØØk: Even more Wi-Fi chips vulnerable to eavesdropping*, 2020. Accessed: 16th February 2021.

URL https://welivesecurity.com/2020/08/06/beyond-kr00k-even-more-wifi-chips-vulnerable-eavesdropping/

[88] Čermák, Miloš, Svorenčík, Štefan, Lipovský, Róbert, and Kubovič, Ondrej. *KR00K - CVE-2019-15126*, 2020. Accessed: 14th March 2020. URL https://welivesecurity.com/wp-content/uploads/2020/02/ESET_Kr00k.pdf

Appendix A

Acronyms

AAA Accounting Authentication Authorization

AEAD Authenticated Encryption with Associated Data

AES Advanced Encryption Standard

ANonce Authenticator Nonce

AP Access Point

ARM Advanced RISC Machines

BSS Basic Service Set

BSSID Basic Service Set Identifier

CA Certificate Authority

CBC Cipher Block Chaining

CCMP Counter Mode Cipher Block Chaining Message Authentication Code Protocol

CTS Clear to Send

CVE Common Vulnerabilities and Exposures

D2D Device-to-Device

DHCP Dynamic Host Configuration Protocol

DN Domain Name

DNS Domain Name System

DoJ Department of Justice

DoS Denial-of-Service

DPP Device Provisioning Protocol

DS Distribution System

DTIM Delivery Traffic Indication Message

EAP Extensible Authentication Protocol

EAPOL Extensible Authentication Protocol over LAN

ECC Elliptic Curve Cryptography

ECP Elliptic Curves over a Prime field

ESS Extended Service Set

ESSID Extended Service Set Identifier

ETA Evil Twin Attack

ET Evil Twin

FC Frame Control

FFC Finite Field Cryptography

FILS Fast Initial Link Setup

GCMP Galois Counter Mode Protocol

GMK Group Master Key

GTK Group Temporal Key

HMAC Hash-based Message Authentication Code

HPKP HTTP Public Key Pinning

HSTS HTTP Strict Transport Security

HTTP Hypertext Transfer Protocol

HTTPS Hypertext Transfer Protocol Secure

IBSS Independent Basic Service Set

IEEE Institute of Electrical and Electronics Engineers

IoC Indicator of Compromise

IoT Internet of Things

IP Internet Protocol

ISP Internet Service Provider

IV Initialization Vector

KCK Key Confirmation Key

KEK KeyEncryption Key

LAP Legitimate Access Point

LLC Logical Link Control

LLMNR Link-Local Multicast Name Resolution

MAC Media Access Control

MFACL Management Frames Access Control Lists

MitM Man-in-the-Middle

MPDU MAC Protocol Data Unit

MSDU MAC Service Data Unit

NBT-NS NetBIOS Name Service

NFC Near Field Communication

NIC Network Interface Controller

NTLM Windows NT LAN Manager

OCSP Online Certificate Status Protocol

OEM Original Equipment Manufacturer

OSINT Open Source Intelligence

OSINT Open Source Intelligence

OSI Open Systems Interconnection

OUI Organizational Unique Identifiers

PAKE Password Authenticated Key Exchange

PBC Push-Button Configuration

PHY Physical OSI layer

PID Process Identifier

PII Personally Identifiable Information

PIN Personal Identification Number

PLCP Physical Layer Convergence Procedure

PMD Physical Medium Dependent

PMF Protected Management Frames

PMK Pairwise Master Key

PNAC Port-based Network Access Control

PNL Preferred Networks List

PN Packet Number

PPDU PLCP Protocol Data Unit

PPP Point-to-Point Protocol

PSK Pre-Shared Key

PTK Pairwise Transient Key

QR Quadratic Residue

RADIUS Remote Authentication Dial-In User Service

RAP Rogue Access Point

RLAN Radio Local Area Network

RSA Rivest–Shamir–Adleman

RSN IE Robust Security Network Information Element

RSN Robust Security Network

RSSI Received Signal Strength Indication

RTS Request to Send

RTT Round Trip Time

SAE Simultaneous Authentication of Equals

SDR Software Defined radio

SHA Secure Hash Algorithm

SMB Server Message Block

SNI Server Name Indication

SNMP Simple Network Management Protocol

SNonce Supplicant Nonce

SOC Security Operations Center

SSID Service Set Identifier

SSID Service Set Identifier

SSL Secure Sockets Layer

TCP Transmission Control Protocol

TIM Traffic Indication Map

TKIP Temporal Key Integrity Protocol

TK Temporal Key

TLS Transport Layer Security

ToS Terms of Service

TPK Tunneled direct link setup PeerKey

VPN Virtual Private Network

WEP Wired Equivalent Privacy

WIDS Wireless Intrusion Detection System

WIPS Wireless Intrusion Prevention System

WLAN Wireless Local Area Network

WNIC Wireless Network Interface Controller

WNM Wireless Network Management

WPA Wi-Fi Protected Access

WPS Wi-Fi Protected Setup

Appendix B

Contents of the enclosed SD

- README.md .... SD contents description
- src .... source files
  - snippets .... implementation source snippets
    - startup.sh .... exploitation chain starter
      - launch_fake_ap.sh .... fake AP launcher
      - launch_dhcpd.sh .... DHCP server launcher
      - launch_dns_blackhole.sh .... DNS sinkhole launcher
      - launch_webserver.sh .... webserver launcher
    - README.txt .... description of attached functions
  - thesis .... thesis LaTeX source code
- text .... thesis text directory
  - thesis.pdf .... thesis in PDF format