MASQUERADING TECHNIQUES IN IEEE 802.11 WIRELESS LOCAL AREA NETWORKS

by

OMAR NAKHILA
B.Sc. University of Mosul, 2004
M.Sc. University of Mosul, 2007

A dissertation submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy
in the Department of Electrical and Computer Engineering
in the College of Electrical Engineering and Computer Science
at the University of Central Florida

Fall Term
2017

Major Advisor: Cliff C. Zou

ABSTRACT

The airborne nature of wireless transmission offers a potential target for attackers to compromise IEEE 802.11 Wireless Local Area Network (WLAN). In this dissertation, we explore the current WLAN security threats and their corresponding defense solutions. In our study, we divide WLAN vulnerabilities into two aspects, client, and administrator. The client-side vulnerability investigation is based on examining the Evil Twin Attack (ETA) while our administrator side research targets Wi-Fi Protected Access II (WPA2).

Three novel techniques have been presented to detect ETA. The detection methods are based on (1) creating a secure connection to a remote server to detect the change of gateway's public IP address by switching from one Access Point (AP) to another. (2) Monitoring multiple Wi-Fi channels in a random order looking for specific data packets sent by the remote server. (3) Merging the previous solutions into one universal ETA detection method using Virtual Wireless Clients (VWCs). On the other hand, we present a new vulnerability that allows an attacker to force the victim's smartphone to consume data through the cellular network by starting the data download on the victim's cell phone without the victim's permission.

A new scheme has been developed to speed up the active dictionary attack intensity on WPA2 based on two novel ideas. First, the scheme connects multiple VWCs to the AP at the same time-each VWC has its own spoofed MAC address. Second, each of the VWCs could try many passphrases using single wireless session. Furthermore, we present a new technique to avoid bandwidth limitation imposed by Wi-Fi hotspots. The proposed method creates multiple VWCs to access the WLAN. The combination of the individual bandwidth of each VWC results in an increase of the total bandwidth gained by the attacker. All proposal techniques have been implemented and evaluated in real-life scenarios.

Keywords– Wi-Fi Security; Virtual Wireless Client; Wi-Fi Traffic Shaping; Evil Twin Attack, Wireless Protected Access

To my parents

Without whom none of my success would be possible

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Cliff C. Zou for his guidance, support and encouragement throughout my PhD program at University of Central Florida.

I would also like to thank members of my dissertation committee: Dr. Mostafa Bassiouni, Dr. Mainak Chatterjee, Dr. Damla Turgut and Dr. Morgan Wang for their valuable guidance and suggestions on my dissertation.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF PSEUDO CODE

CHAPTER 1: INTRODUCTION

1.1 Problem Statement/Motivation

Nowadays, 802.11-based wireless local area networks are everywhere [1]. Enterprise WLAN market share is expected to grow to 21.10 Billion USD by 2021 when the market share in 2016 was 5.53 Billion USD [2]. This growth was driven by the advent of the Internet of Things (IoT). Also, people rely on the wireless network in their daily life bases, shopping online and paying bills to name a few. These features make Wi-Fi networks an attractive target for intruders to compromise wireless client information [3][4].

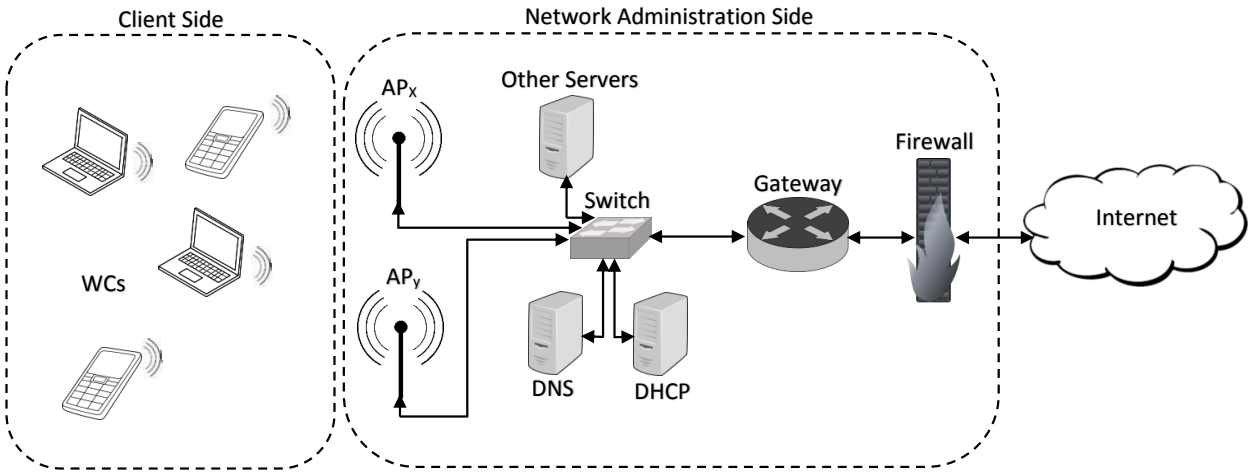


Figure 1.1: Typical WLAN diagram were Wireless clients (WCs) connect to the Internet through Access Points (APs). DHCP and DNS servers are used to assign network configuration and resolve domain names, respectively. The network administrator may add other servers to the network based on the WLAN design, for example, Remote Authentication Dial-In User Service (RADIUS) server. Gateway is used to route network traffic to the Internet while firewall is used to protect the WLAN from the Internet.

In this dissertation, we inspect the current WLAN security challenges and their solutions. We divided the WLAN network into two parts, client and network administration as shown in figure 1.1. The division is based on which part of the network being attacked. For the client's side,

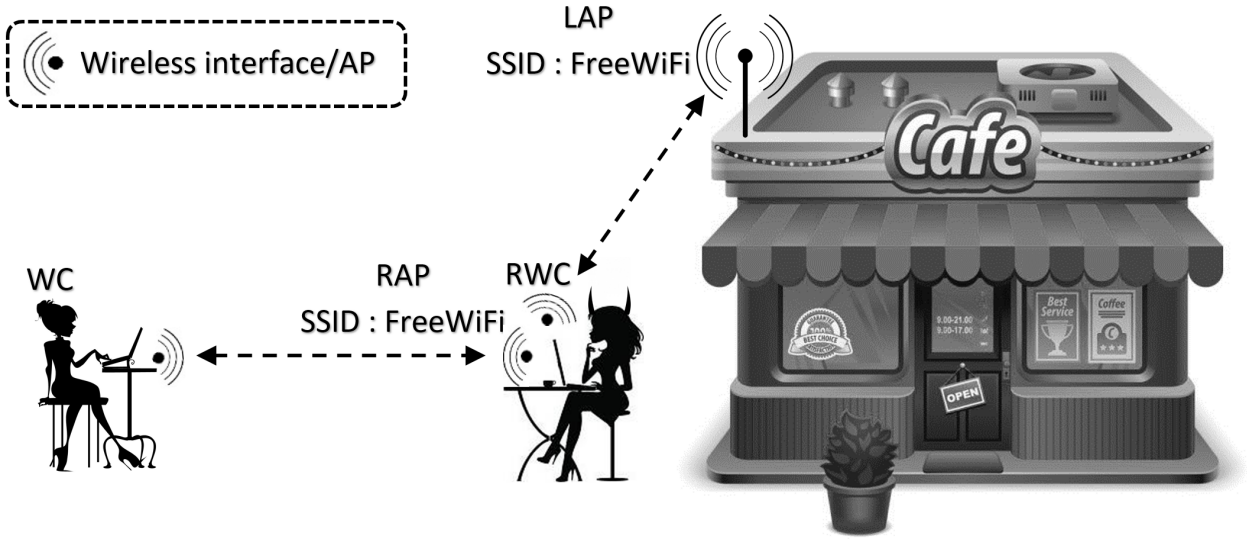
we analyzed Evil Twin Attack (ETA), a popular attack on open Wi-Fi networks. Furthermore, we propose new data consumption attack on mobile wireless clients. While, on the network administrator side, we improved an online dictionary attack on the current wireless security protocol suite, Wireless Protected Access (WPA2). In addition, we introduce a new attack on the wireless traffic shaping imposed by the network administrator.

1.2 Evil Twin Attack

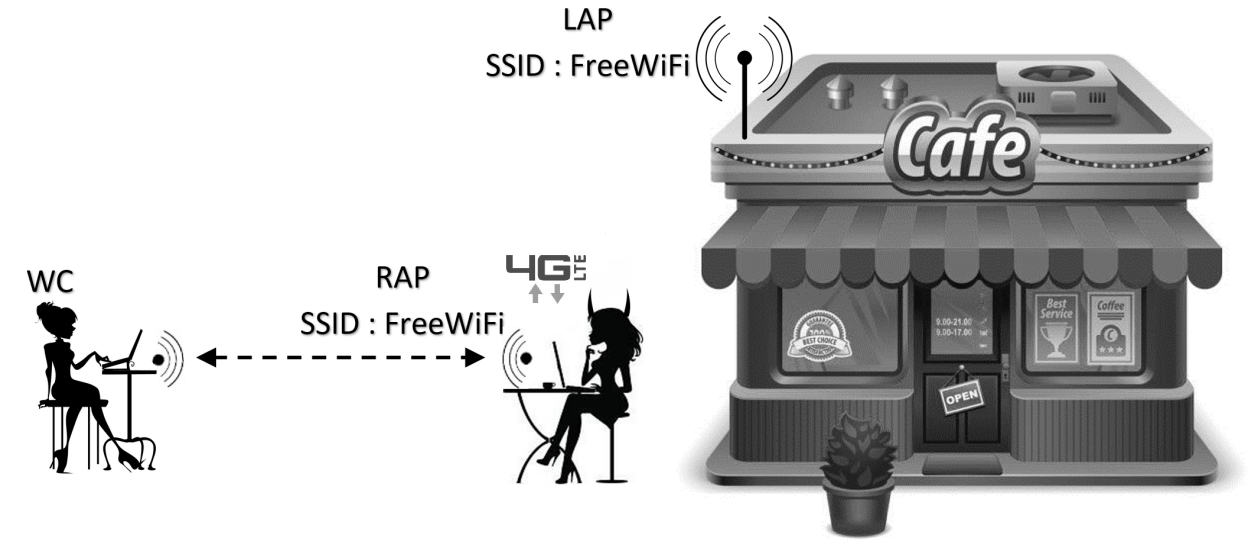
Wireless networks provide connectivity to the Internet for smart phones, mobile PCs and tablets. The growth and use of wireless devices has increased data traffic on cellular networks [5]. Some businesses such as coffee shops, fast food restaurants and airports offer free Wi-Fi services to their clients. Besides offloading data traffic from cellular networks [6], the use of Wi-Fi provides a fast and budget friendly alternative to a wireless client (WC) when it comes to accessing the Internet [7]. However, for ease of access, these Wi-Fi networks provide no security in terms of authentication or encryption. When a WC wants to access a Wi-Fi network, she must agree to the "Public Wi-Fi Access Terms and Conditions" in which the Wi-Fi provider assumes no responsibility for the security/privacy of the WC's information [8].

Insecure Wi-Fi networks provide a tempting environment for attackers to initiate many attacks, one of them is called Evil Twin Attack (ETA) as illustrated in Figure 1.2. ETA refers to a Wi-Fi rogue access point (RAP) impersonating a legitimate access point (LAP) to eavesdrop WC's Wi-Fi data [7, 8, 9, 10, 11]. Since a Wi-Fi network can only be recognized by its SSID and MAC address, the attacker can set up an RAP with the same SSID of the LAP. Furthermore, the attacker's RAP may have better and more powerful signal than the LAP, which will lure the WC to connect to it first [12].

After the WC connects to the RAP, the attacker can snoop on the WC's data traffic and/or launch man-in-the-middle attack (MIMA). For example, using ETA, an attacker can infer mo-



(a) Evil twin attack using single ISP gateway. The attacker pass through WC data to the Internet using LAP.



(b) Evil twin attack using different ISP gateways.The attacker uses her own mobile data connection (4G-LTE) to pass through WC data to the Internet.

Figure 1.2: Illustration of ETA scenarios. The RAP can successfully lure WC connecting to it instead of the LAP when it provides stronger/better signal to those WCs.

bile keystroke by detecting the change in the channel state information when the wireless client moves her hand and fingers. Mobile keystroke can be recognized even the the wireless client is using HTTPS. Another example, SSL strip attack [13] that force the WC to use HTTP instead of

HTTPS. Furthermore, DNS spoofing attack [14] where the WC receives incorrect IP address when requesting a certain domain. This results in directing the WC to visit malicious website rather than the actual website.

Once the WC is connected to the RAP, the attacker have two options to direct WC's data traffic to the Internet. First, the attacker can use another Wi-Fi interface card and connect to the LAP as a rogue wireless client (RWC). The attacker use the RWC to pass the WC traffic to the Internet. Both LAP and RAP use the same ISP gateway as shown in Figure 1.2a. Hence, we call this attack option as *ETA using single ISP gateway*.

The attacker has another option to avoid connecting to the LAP. Due to the increase in Internet access speed of mobile broadband connections, such as 4G Long Term Evolution (LTE) or WiMAX, the attacker can use her own cellular broadband link to connect the WC to the Internet [9, 11]. In this scenario, the attacker is placed between the RAP and her broadband connection as illustrated in Figure 1.2b. We call this attack option as *ETA using different ISP gateways*.

1.3 Mobile Data Consumption Attack

Cell phones are becoming a necessary piece of equipment in our lives. In 2016, there were seven billion active cellular telephone subscriptions worldwide with three and half billion cell phones that have a subscription to access the Internet [15]. To reduce congestion on the cellular network, most cell phone ISP carriers throttle customer data after exceeding a certain data limit or impose a monthly limit data cap.

On the other hand, as a complimentary service, coffee shops, fast food restaurants, and airports provide free Wi-Fi network access to their customers. These open access networks provide budget-friendly Internet access which helps offload data traffic from the cellular network [6][16]. However, for their ease of access, these types of networks are insecure in terms of lacking authentication and encryption. Instead, when the customer first accesses the Wi-Fi network, he or she must

agree to the Public Wi-Fi Terms and Conditions, where the ISP provider claims no responsibility for the customer information security/privacy [8].

The absence of wireless security in open access Wi-Fi networks provides tempting environments for attackers [17][18][19][10]. An Evil Twin Attack (ETA) can be initiated by an attacker to impersonate the role of a legitimate Wi-Fi access point (illustrated in Figure 1.2). Such an impersonation is simple since an open Wi-Fi network can only be recognized by its MAC address and Service Set Identifier (SSID). Furthermore, the attacker's fake access point (AP) may provide a better and more powerful signal to the victim in which case it will cause the victim to connect to the attacker's network instead of the legitimate network. Such a switch is automatic and can happen without the victim's intervention [12].

After the victim connects to the attacker's AP, the attacker can snoop the victim's wireless data and apply a Man In The Middle Attacks (MIMA). For example, DNS spoofing attack [18] is a popular MIMA that allows the attacker to response to DNS query coming from the victim. The attack can redirect the victim web browser to a malicious website rather than the legitimate one by sending her a wrong IP address.

1.4 Wireless Protected Accesses II

WLAN's security evolved over three major stages throughout its road to protect wireless clients. First, Wireless Equivalent Privacy (WEP), is the first security protocol used to protect IEEE 802.11 WLAN [20][21]. Although WEP uses Rivest Cipher 4 (RC4) stream cipher to encrypt wireless data, the size of initializing vector (IV) used was small, which led to IV conflict. Furthermore, the master keys are directly used to encrypt data with no key management. Researchers have demonstrated ways to break the security in WEP in less than a minute [22]. These variabilities led to the emergence of the second stage security standard of Wi-Fi Protected Access (WPA). WPA was created to support legacy wireless devices and at the same time to patch WEP

defects[23]. The current and the third WLAN security stage accomplished by introducing WPA2. The design of WPA2 was not limited by the hardware constraints like WPA. WPA2 uses AES (Advanced Encryption Standard) and CCMP (Counter Mode CBC-MAC Protocol) by default, which provides stronger encryption than WPA[20][23].

Both WPA and WPA2 have two modes of operation. The first mode is the Pre-shared key (PSK) or personal mode, which is designated for small office/home office (SOHO) wireless networks. An access point (AP) will use only one pass-phrase (8 to 63 characters in length) to authenticate wireless clients. Each client should use the same exact passphrase stored in AP to pass the authentication process successfully. If a WLAN's administrator wants to change the pass-phrase, he needs to change the pass-phrase in all wireless clients and APs. For WLANs in large cooperations, changing the passphrase on all wireless clients and APs is not practical[24].

The second mode, also called Enterprise mode, needs administrators to set up a dedicated Remote Authentication Dial-In User Service (RADIUS) server. Each user will have a unique username and password to be authenticated by the RADIUS server. After the authentication process completes successfully, the AP will receive a random key from the RADIUS server to protect the wireless communication[24].

Dictionary pass-phrase attack is one of the common attacks on WPA2-PSK[20]. Since PSK will be the primary key to protect WLAN, the attacker will try to guess the passphrase used to generate PSK. This attack can be done by capturing the initial WPA2-PSK handshaking between a legitimate wireless client and the AP. After capturing the handshaking frames, the attacker will use offline dictionary word guessing software to recover the passphrase.

On the other hand, most attacks on WPA-II enterprise are based on man in the middle attack (MIMA) [25][26][27]. The attacker positions herself between the WC and the AS to capture the WC credentials. However, using a digital certificate on the RADIUS server side with the proper configuration on the WC side prevents most of these attacks [27]. In this case, the attacker can initiate an active brute force attack to gain access to the wireless network. The downside of such

an attack is the very low-level password guessing speed, which makes such a brute-force attack little threat to the wireless network.

1.5 Wireless Traffic Shaping

Staying connected to the Internet has become a priority in our daily routines. At the same time, the increase in Internet data traffic, due to the widespread of high-definition multimedia, has pushed us to search for high-speed Internet access [28]. Clients can use cellular service to have high-speed Internet access through their mobile data connection. Although mobile broadband is convenient, it is also expensive and may fluctuate based on the wireless coverage area. On the other hand, businesses such as fast-food restaurants, coffee shops, hotels, and airports, may provide complementarity Internet access to their clients through the use of Wi-Fi hotspots. Using these Wi-Fi hotspots to access the Internet offers a budget-friendly alternative to mobile data connection [16].

Wi-Fi hotspots allow clients to simultaneously connect different wireless devices to the Internet [6]. However, network administrators may impose wireless bandwidth limitations on the wireless devices accessing the Internet through these Wi-Fi hotspots [29][30][31]. Each wireless device will be assigned a certain download and upload speed to access the Internet. The reason behind these limitations is to prevent customers from abusing the complimentary Internet service, to provide fair bandwidth allocation, and to make the customer pay to have a faster Internet connection.

Different commercial software are available to increase the client Internet connection speed. For example, an increase in the Internet downloading speed can be achieved by initiating different connections simultaneously to the same file on the Internet [32][33]. The summation of all connection's speed will result in a faster file download speed. However, bandwidth limitations are implemented at the data link and network layer which make it difficult for these tools to take advantage of the multiple data connections. The bandwidth controller will detect that all of these

connections are initiated from one single wireless client and thus reduce the speed of all the connections to a single one.

However, an attacker may circumvent the traffic shaping policy applied by the wireless network administrators by using virtual wireless clients technique. Although the virtual wireless clients technique was developed to improve the wireless network performance and privacy [34], in our work, it is used as a tool to attack wireless network infrastructure [35][36].

1.6 Contributions

This dissertation addresses the current WLAN vulnerabilities. Our investigation targeted vulnerabilities on the wireless client and the network administration side.

First, in [17] we presented a novel detection method to deal with the second type of ETA (ETA using different gateway). The technique detects whether or not different gateways are used by multiple APs in one hotspot location that has the same SSID. As far as we know, each hotspot will always use the same gateway for Internet access no matter how many legitimate APs have been set up in the same hotspot [37]. The detection method is a secure client-side approach that does not rely on any support from hotspot networks or dedicated servers. In addition, no training data or authorized trusted AP list would be used in the ETA detection. Finally, our detection method was implemented and evaluated in a real-life environment.

Second, in [18] we focused on the ETA using single ISP gateway. We proposed a real-time procedure ETA detection that examines all nearby access points (APs) in a parallel manner. At the end of the detection process, each AP marked as either LAP or RAP. The proposed ETA detection monitors multiple Wi-Fi channels in a random order looking for particular wireless frames. These frames are sent from a dedicated public server on the Internet. By capturing these particular wireless frames, WC can detect the RAP instantaneously. Our ETA detection is a client-side solution, that is more appropriate than the network administrator side solutions [9][10] because it allows

the WC to guarantee her security without any assistance from network administrators. Also, the WC does not need to have any information about the Wi-Fi network configuration or any training data or Wi-Fi network fingerprint as required by other solutions [8][38][39]. Finally, our detection technique effectiveness was mathematically modeled, prototyped and evaluated in a real-life environment.

Third, we present a novel detection method to detect both types of ETA simultaneously. Basically speaking, the detection technique modifies the previous detection methods and merge them into one comprehensive ETA detection using virtual wireless clients (VWCs). Using one wireless interface card, WC creates two VWCs to detect both ETA scenarios simultaneously. Each VWC is responsible for identifying one ETA type. This yield one complete solution to prevent ETA. The system was implemented and tested using off the shelf devices.

Fourth, in [40] we introduce a new vulnerability that depletes customer mobile data quota. The attack targets customers that use free open Wi-Fi networks instead of their cellular data connection. Using different types of man in the middle attacks, an attacker may trigger the wireless client to download large data from the Internet using her own cellular data connection. Our proposed attack was implemented and evaluated in a real-life environment.

Fifth, in the administrator side vulnerabilities analysis, we presented a new scheme to apply online dictionary attack on WPA2-PSK [41]. To our knowledge, all the available implementations of the dictionary passphrase attack on WPA2-PSK are offline based attacks, and they will fail if there is no legitimate wireless client connected to the AP or in the process of connecting to the AP. In this scenario, all offline brute force implementation will not work since they will need the initial WPA2-PSK four-way handshaking frames between the AP and a legitimate wireless client. On the other hand, an online dictionary attack can still work under this scenario. We present two novel techniques to speed up the online dictionary attack process. First, we create parallel virtual wireless clients (VWC) simultaneously authenticating to an AP. Each VWC emulates a standalone wireless client. Second, we enable each VWC to guess the PSK multiple times within a single wireless

session. Each VWC keeps guessing the WPA2 passphrase until it receives a de-authentication frame from the AP. Finally, our online dictionary attack was implemented and evaluated in a real-life environment using different off-the-shelf wireless APs. Our technique showed that it could speed up the password guessing process by 100-fold compared to the traditional online single client attack.

Sixth, we expand the novel technique in [41] to speed up the active dictionary attack process on WPA2-enterprise [36]. By using only one wireless interface card, we can create many parallel virtual wireless clients (VWCs) simultaneously authenticating to a RADIUS server. Each VWC emulates a standalone wireless client, and hence, increasing the attacker's active dictionary attacking power. Although by default, an authentication server, such as RADIUS server, may delay rejection response to slow down the online dictionary attack [42][43], using VWC technique lowered the impact of such a protection feature. The delay time imposed by the RADIUS server will be utilized by the attacker to start a new connection, and test other passwords. Finally, our active dictionary attack has been implemented and evaluated in a real-life environment using different off-the-shelf wireless APs and one of the most popular RADIUS servers. Our technique showed that it could speed up the password guessing speed by 1700% compared to the traditional single wireless client attack.

Finally, in [44], we introduce a network vulnerability to avoid Wi-Fi hotspot bandwidth limitation by using multiple Virtual Wireless Clients (VWCs). Using only one wireless network interface card, an attacker can create multiple virtual wireless clients. Each VWC emulates a standalone wireless device. The VWCs start multiple connections to a remote file on the Internet. The bandwidth allocated to each VWC is separate from other VWCs which allows the attacker to overload the hotspot using only one physical wireless interface card. The proposed technique was implemented and evaluated in real-life scenarios using off the shelf devices.

1.7 Dissertation Organization

The dissertation organized as follows. Chapter 2 discusses the related works for both sides, client and administrator, which include previous ETA detections, mobile data consumption attack, WPA-II vulnerabilities and wireless traffic shaping techniques. In chapter 3 we presented several intuitive solutions and showed why they are not effective in ETA detection. We also show the design of the new detection method for both ETA using single ISP gateway and ETA using different ISP gateway. Data consumption attack is presented in Chapter 4. While in chapter 5 we explain how WPA2 works and introduce the design of the new online parallel dictionary attack on both WPA2-PSK and WPA-2 enterprise. In chapter 6 we illustrate a novel technique to by pass wireless traffic shaping controller. Finally, the conclusion and future works will be present in the chapter 7.

CHAPTER 2: LITERATURE REVIEW

WLAN is the most popular wireless network for clients to access Internet [45]. Through the past years, WLAN increased its potentials to accommodate the growth of bandwidth demands [46]. However, since the invention of WLAN by the IEEE 802.11 committee, securing Wi-Fi networks is considered by researchers as an ongoing challenge [47]. In this section, we examine closely multiple attacks on WLAN. The first attack is Evil Twin Attack that can be implemented on open Wi-Fi networks. Second, we introduce a new attack on mobile users that force data consumption. On the other hand, we illustrate a parallel active dictionary attack that can be carried out on WPA2. Finally, a unique attack is presented to circumvent wireless traffic shaping controller.

2.1 Evil Twin Attack

ETA in wireless networks is a threat that can transfer the privilege from a legitimate wireless network administrator to an attacker to become the gateway of a wireless client (victim). In this scenario, all the wireless traffic from the victim will pass through the attacker node. At this point, the attacker can apply the desired man in the middle attack (MIMA) to exploit any vulnerability that can leak information about the victim. MIMA in this situation will be hard to detect since the victim will be on a separate wireless network (attacker wireless network) than the legitimate wireless network.

The detection of ETA was under the spotlight for many years. Researchers have been investigating detection methods that can alert the wireless network administrator or client about the presence of this type of attack. However, most ETA detection methods are bound to work in particular environments. In [7], researchers divided ETA detection into three different categories: protocol modification, hardware fingerprinting and non-hardware identification. On the other hand, [9][10] divide ETA detection into two groups: comparing data traffic at different locations of the

Table 2.1: Acronyms

Acronym	Definition	Acronym	Definition
ETA	Evil twin attack	LWC	Legitimate wireless client
WC	Wireless client	WIC	Wireless interface card
VWC	Virtual wireless client	AS	Authentication server
AP	Access point	RAS	Rogue authentication server
ISP	Internet service provider	WR	Wireless router
LAP	Legitimate access point	EAP	Extension authentication protocol
RAP	Rogue access point	EAPoL	EAP over lan
RWC	Rogue wireless client	WS	Wireless sessions
P_d	Detection probability	RADIUS	Remote Auth. Dial-In User Service
P_m	Detection missing probability	D	WC switching time between APs
N	Number of wireless channels	k	Attacker dis/connect from/to LAP
RTT	Round trip time		
PIS	Public information server		

Wi-Fi network with a known authorized list, and checking if the source of the data traffic is coming from a wireless or a wired network.

In this dissertation, we classify ETA detection into two main categories: network administrative, and client detection side. In network administrative side ETA detection, the network administrator is responsible for detecting and/or assisting the WC to detect ETA. Since the network administrator has all the information about the Wi-Fi network, she can have a list of fingerprints of all devices constructing the Wi-Fi network. While in the client side ETA detection, the wireless client is the one responsible for detecting ETA without any help from the network administrators.

In the first category, administrators are the one responsible for ensuring wireless client protection from ETA. Administrators scan the airwaves and match between APs found transmitting nearby with an authenticated APs list that has been previously created on the administrator side.

Each AP should have a fingerprint that can be used to identify itself. A fingerprint is any information that can be used to distinguish a single device or a group of devices from one another. For example, AP hardware and location can be used as a fingerprint [9].

The strength of this type of protection depends on the fingerprint used to recognize the AP. For example, if the location is the fingerprint of an AP, this kind of detection may trigger a false positive alert of a potential ETA if there is a nearby AP that transmitting in close range to the authenticated APs [10]. Also, an attacker may change the rogue AP characteristics to match the ligament AP. For instance, the attacker can change the MAC address of a rogue AP to one of the authenticated APs. Researchers were investigating different types of fingerprints that can be used to distinguish one AP from another. In [38], AP clock skew was used as a fingerprint. Using clock skew as a fingerprint was further improved by [39]. However, without having an authorized AP list beforehand, this ETA detection fail.

Furthermore, the network administrative side detection adds more cost to the Wi-Fi network construction. The Network operator may have to install wireless sensors and collect traffic data at the switch/router to be compared with the available fingerprint authorized list. Another key point in this type of detection, is that the WC still unaware of the level of protection, (if any) that a specific Wi-Fi network is using against ETA. To sum up, administrative side ETA detections are limited, expensive and not available in many scenarios [9].

The second category of ETA detection methods is user side detection. This type of detection is preferable than the administrator side detection since the wireless clients will ensure their protection against ETA. One of the detecting method techniques that fall into this type of category [9] propose that by measuring the travel time of packets between the wireless client and a nearby server, the wireless clients can detect the presence of ETA. This is because when an attacker uses the rogue AP to pass through wireless client data, there will be an extra wireless hop between the wireless client and the legitimate AP. This extra wireless hop will add more time compared to the direct connection between the wireless client and the legitimate AP.

However, this method assumes that the attacker will use the legitimate wireless network gateway to pass through client data traffic. This detection will fail especially when the attacker uses faster Internet connection compared to the legitimate wireless network. In this scenario, the attacker can delay the response time of the propagating packets between the server and the wireless client to match the propagation time of the packets passing through the legitimate AP. In addition, this method suffers from wireless signal strength fluctuations and the data traffic load on the APs that may vary the response time between the wireless client and the server [9].

Another ETA detection method that belongs to the second category and can be used to detect different gateway is traceroute command ETA detection method [48]. In this detection method, traceroute command will be used to find route information between the wireless client and a random remote server. In the beginning, the wireless client connects to any AP and use the traceroute command to find the route information between himself and any remote server. Then, the wireless client switches to another AP and use traceroute command to record the route information between herself and the same remote server used by the first AP. Using two different APs for the same wireless network should return the same route information [11].

Nevertheless, this type of detection may fail since network administrators may configure network firewall to drop these traceroute packets for security purposes [49]. Also, an attacker can easily pass traceroute ETA detection method by simply monitoring the wireless data traffic. This monitoring is possible because traceroute uses the unencrypted ICMP protocol to gather route information between the wireless client and the remote server. An attacker can capture traceroute results sent to the wireless client using the legitimate wireless network. After that, the attacker can send these results to the wireless client using the rogue wireless network. This give the same route information for both gateways which will pass ETA detection method without triggering any alarm.

On the other hand, a wireless client can set up a VPN connection through the wireless hotspot. In this case, all the traffic between the wireless client and the hotspot will be encrypted.

However, VPN is not available for all users and have numerous points of failure [50].

Finally, Open WiFiHop [51] is a WC-based ETA detection that listens to different Wi-Fi channels looking for watermarked packets. We address the vulnerabilities found in Open WiFiHop and present our ETA detection solution in section 3.2.

2.2 Mobile Data Consumption Attack

A mobile data consumption attack can drain a victim's data cap in a short period of time. Such an attack can prevent the victim from accessing the Internet after reaching his or her monthly allocated data limit, which leads to denial of service. If the victim does not have a data cap, he or she will be continuously charged for the data used by the attack. Also, keeping the mobile device transmitting/receiving data results in battery power consumption.

Stealth spam attacks can abuse the fact that many connections that are formed do not tell the network that they are closing [52]. Thus an attacker can use a connection made by the victim that the network still thinks is open, even though the phone had closed it. The attacker can then send data over this connection as a spam attack, which consumes the victim's data.

Other similar attacks are introduced in [53]. The first is the cloak-and-dagger spamming attack where the victim's data is consumed by either spoofing the victim's IP address and using data as the victim, or by sending an MMS message which opens up a connection to spam the victim over, using up the victim's data. The second attack in [53] is the hit-but-no-touch attack where data packets are sent to the victim with a shortened time-to-live value. The packets then pass through a mobile network's billing system, but never make it to the victim, thus invisibly using the victim's data.

Mobile billing vulnerability based on TCP packet re-transmission is presented in [54]. Many mobile service providers, such as AT&T, Verizon, T-Mobile, and Sprint, bill customers based on the total amount of data traffic that has been sent and received to the Internet, including

retransmitted data packets. An attacker can force the victim to consume more data by increasing the number of TCP data packets retransmitted.

The attacker sends a text message to the victim with a link to a malicious server. When the victim opens the link, a TCP connection between the victim's cell phone and the remote server will be established. The connection is based on TCP protocol, and thus it can keep forcing the victim to retransmit TCP packets. TCP packet retransmission can be initiated when three acknowledgment packets were received to the same TCP sequence number or by the timeout of the TCP connection.

A similar attack on a mobile billing system is when the attacker sends a spoofed IP packet to an external server [55]. The remote server uses the victim's spoofed IP address to start a network connection to the victim's cell phone. The remote server then sends a large amount of data to the mobile network that will be directed to the victim cell phone.

All the previous attacks targeted victims that use a private IP address. The impact of these attacks can be intensified when the victim is using a public IP address [56]. With the spread of IPv6, cell phone devices will have direct access to the Internet. In this situation, the attacker can directly send data packets to the victim that deplete cell phone data quota.

In this dissertation, we present a new attack that exploits the inaccuracy of mobile network billing systems.

2.3 Wireless Protected Access II

Following the vulnerabilities found in WEP, Wi-Fi Protected Access I (WPA-I) and Wi-Fi Protected Access II (WPA-II) were introduced [57]. WPA-I is used to provide a temporary solution to legacy wireless devices, and WPA-II is the current standard security protocol for 802.11 wireless networks. In publications, WPA-II is also referred to as robust security network (RSN) or IEEE 802.11i-2004 [58]. WPA-II deployments can be different between Small Office / Home Office (SOHO) and enterprise wireless network. WPA-II Pre-shared key (PSK) is used in SOHO

where only one passphrase is used to protect the wireless traffic. However, in WPA-II enterprise, each wireless client has her username and password to protect their wireless traffic. Network administrator sets up an authentication server (AS), such as Remote Authentication Dial-In User Service (RADIUS), to authenticate each wireless client.

WPA2-PSK uses state of the art AES/CCMP to protect wireless client data. PSK length is 256 bits or 64 octets represented as a hex number. However, since it is more convenient for users to remember ASCII keys than hex numbers, users will use a pass-phrase that consist of 8 to 63 characters. The pass-phrase then mapped to PSK. This mapping drops the security of WPA2-PSK to about 2.5 bits per character [59][47]. Pass-phrase less than 20 characters are vulnerable to dictionary attack.

The most feasible technique to bypass WPA2-PSK security is by recovering the pass-phrase from the four-way handshaking communication. Most of the available implementation are based on the offline dictionary attack against the four-way handshake. Attacks on WPA2-PSK are categorized into two parts, offline and online.

For the offline attack, Aircrack-ng [60] software suite is one of the most popular software used to brute force PSK using dictionary word list. First, the four-way handshaking must be captured between legitimate wireless client willing to connect to the AP. Capturing the four-way handshaking can be accomplished by using the Airodump-ng software. If the wireless client already connected to the AP then, the attacker can use Aireplay-ng which force the wireless to de-authenticate and start the four-way handshake again [61].

After the attacker capture the four-way handshake, Aircrack-ng software start the offline dictionary passphrase guessing attack to recover the passphrase. Other offline software can speed up the offline pass-phrase guessing attack by using GPU like Hashcat [62] software.

All the previous attacks will fail if there is no legitimate wireless client willing to connect to the AP. Furthermore, even if there is an already connected wireless client, if the network is protected using 802.11W [63], the attacker will not be able to de-authenticate the connected clients.

In contrast, our proposed technique is not based on the condition of having a legitimate wireless client.

For the online attack, Wi-Fi Alliance introduced Wi-Fi Protected Setup (WPS), which is an optional feature to help wireless clients connect to the WLAN with ease, while providing protection at the same time [64]. One of the methods used by WPS to authenticate a user is by asking her to enter an eight-digit PIN number written on the back of the AP. Knowing the PIN will reveal the passphrase used to drive the WPA2-PSK keys. However, due to poor design of WPS, using Reaver [65] software, the attacker can apply an online brute force attack and recover the PIN without having a legitimate wireless client present.

Since WPS is an optional feature, an AP may not support it. Also, some manufacturers limit the number of times a wireless client can enter a wrong PIN number. If the wireless client exceeded that limit, the WPS method would be locked for a certain amount of time. Both of these cases limit the attack on WPS. On the other hand, our proposed technique is not affected by the availability of WPS. Furthermore, WPA2-PSK is not limited by the number of times a wireless client can enter an incorrect passphrase.

For enterprise WLAN, network administrator avoid using WPA2-PSK since she doesn't have control on each WC interdependently. For example, in WPA2-PSK, to revoke the access of a particular WC to the WLAN, the network administrator has to change the PSK on all APs. Also, she has to update the new PSK on all other WCs. Furthermore, WPA2-enterprise support different methods of WC authentications which used to generate the encryption master key.

IEEE 802.11i enterprise consists of two main parts: the AS, such as RADIUS server, and the authenticator, which is the AP. When the WC, also called supplicant, wants to access the WLAN, she should be authenticated first by the AS. The communication between the AS and the WC pass through the AP. Extensible Authentication Protocol (EAP) is used to define the authentication method between the AP and the AS. EAP and its authentication method will be encapsulated in the RADIUS protocol between the AS and the AP. On the other hand, between the AP and the

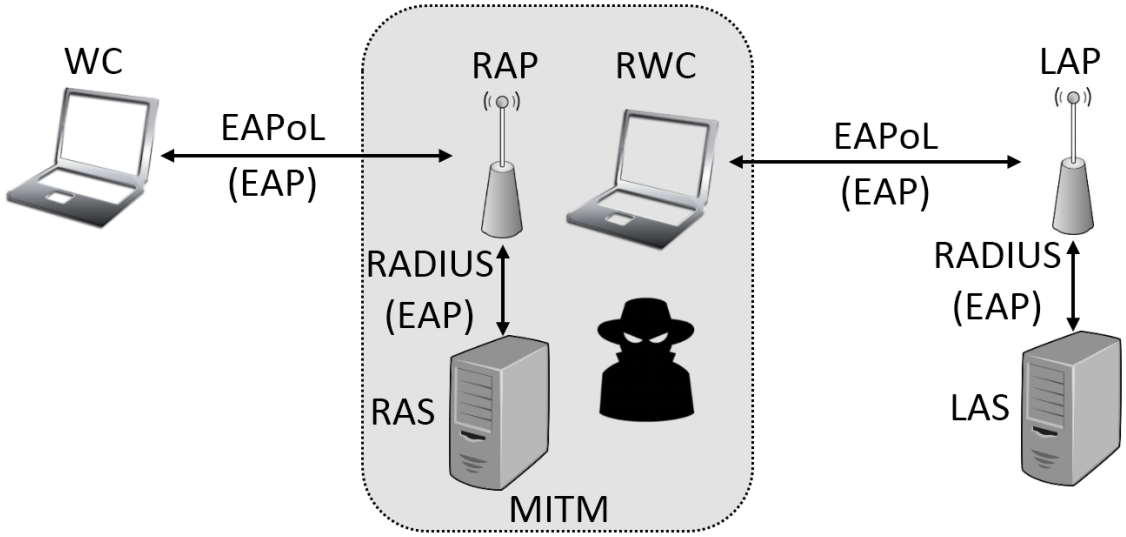


Figure 2.1: Typical MITM attack on WPA-II enterprise

WC, EAP and its authentication method is sent using EAP over IEEE 802 protocol, which is known as “EAP over LAN” or EAPOL [58].

After the authentication phase finishes successfully, both the WC and the AP generate a random Pair Master Key (PMK). At this point, 4-way handshaking procedure starts between the WC and the AP only. Both of the WC and AP will use PMK to generate Pair Temporary Key (PTK) which is used to protect the four-way handshaking communication and the WC data. Finally, a Group Transient Key (GTK) is generated by the AP and sent to the WC to protect the wireless broadcast traffic [58].

802.11 enterprise WLAN depends on the 4-way handshake and 802.1x protocol to secure WC data. This WPA2 type should not be confused with 802.11 personal, where WLAN depends only on the 4-way handshake to authenticate WC traffic [58]. We divided the attacks on WPA-II enterprise into two main categories: MITM attacks [25][26][27][66][67][68] and denials of service attacks (DOS)[69][41].

In the first category, an attacker sets up a rogue AP (RAP) and a rogue AS (RAS) as shown

in figure 2.1. The RAP impersonates the legitimate AP (LAP) by broadcasting the same WLAN SSID. This attack can also be refer to as Evil Twin attack [27][17]. the WC may connect first to the RAP when it offers better signal than the LAP.

When the WC connects to the RAP first, she will be authenticated using the RAS. At the same time, the attacker can start connecting to the LAP and be authenticated to the LAS using the WC credentials. After successfully capturing the WC credentials, the attacker can turn off the RAP allowing the WC to connect to the LAP. This is the basic implementation behind most MIMA on IEEE 802.11i.

Most of the MIMA succeed only when the WC has misconfiguration that is exploited by an attacker. For example, authentication protocols such as EAP - Tunneled Transport Layer Security (TTLS) and Protected EAP (PEAP) allows the WC to check the AS digital certificate [17]. In [27] the attacker took advantage of the WC not checking the Common Name (CN) string of the digital certificate offered by the AS to have successful MITM attack. The attack would fail if the WC checks and rejects the RAP digital certificate [27].

Another successful type of MITM attacks is when the attacker makes the WC use a less secure EAP authentication protocol. For example, in [26] the attacker's RAS authenticated the WC using Light EAP protocol, which is a less secure protocol compared to both EAP-TTLS and PEAP. This attack will fail if the WC only used EAP-TTLS or PEAP as the main authentication methods with proper AS digital certificate checking [27][70].

DOS is the second category of attacks on WPA-II enterprise. Although this type of attack does not compromise the WC credentials, it will prevent her from accessing the WLAN. In both [69] and [41] the attacker sent crafted EAP frames to prevent the WC from successfully completing the authenticated phase. This type of attack is out off the scope of this dissertation.

The current proposal used the same concept in [35] to apply the attack on WPA-II enterprise. Such an attack is important when others attacks such MITM is not feasible.

2.4 Circumventing Wireless Traffic Shaping

A high-speed connection is an attractive option when it comes to accessing the Internet. One of the convenient methods to connect to the Internet is to use the cellular data connection. The client can also use her mobile as a Wi-Fi hotspot and share the data connection with other users. However, most cellular companies charge a lot of money when it comes to accessing the Internet; while other cellular companies even limit the amount of data being downloaded or uploaded to/from the Internet.

On the other hand, businesses such as fast food restaurants, coffee shops, hotels, and airports may provide complimentary connection to the Internet through public Wi-Fi hotspots. These public Wi-Fi hotspots may impose traffic shaping to limit the bandwidth of their wireless clients. Such a limitation features can be freely available in many commercial wireless devices through Guest Wi-Fi option [71][72].

Wireless clients can use different techniques to increase the Internet connection speed. For example, the wireless client throughput can be increased by using UDP-based Data Transfer Protocol (UDT) [73]. The UDT technique employs UDP protocol to transfer files instead of using TCP protocol. The removal of the connection-oriented protocol overhead will reduce the amount of control traffic and increase the actual data traffic. However, the connection will be still throttled by the bandwidth limiter since the protocol does not change the physical and logical address of the wireless client. Furthermore, UDT is designed to be used with high-speed networks.

Another method that can be used by the wireless client is to employ a commercial software such as Internet Download Manager (IDM) [33]. IDM accelerates the file transfer up to five times by initiating multiple connections to the same file on the Internet [32]. Each connection starts from different parts of the file. The total download speed equals to the summation of all the connection's speeds to the file. However, this technique is also limited by the bandwidth controller since the wireless client can still be identified by her IP and MAC address.

Increasing the Internet connection speed can be also achieved when the wireless client uses both, the mobile data and the Wi-Fi hotspot connections simultaneously. In [74], clients can combine both Internet connections using a proxy server. Data request will be sent to a proxy server that is used to load balance the download/upload speeds between the two network connections on the wireless client. However, this technique uses the mobile data connection, and it also limited by the speed of the hotspot bandwidth limiter.

In this dissertation, we present an attack to bypass the bandwidth limitation used in public Wi-Fi hotspot by using virtual wireless clients technique.

CHAPTER 3: Client-side Evil Attack Detection

3.1 Introduction

In recent years, businesses, such as fast-food restaurants, coffee shops, retail stores, have set up Wi-Fi access points to provide free wireless Internet service to attract and better serve their customers. These sites are also called hotspots. Most of the time, Wi-Fi hotspots have no or very limited security protection. Clients only need to search the airwave and connect to the wireless network. No mean of encryption or authentication used besides the wireless network name (SSID). Because of the lack of security protection, hotspots are vulnerable to the famous and well-known Evil Twin Attack.

When the rogue AP hijacks the Wi-Fi connections from clients, the rogue AP usually has two options to connect to the Internet. First, the rogue AP can itself behaves like a regular Wi-Fi client and uses the legitimate AP to connect to the Internet. This is the classical ETA [9] [10] [11] as shown in Figure 1.2a

The second Internet access option for an ETA is to use cellular broadband connection [9] [11] as illustrated in Figure 1.2b. This type of ETA become more popular nowadays due to the increase in the Internet access speed of mobile connections, such as 4G Long Term Evolution (LTE) or WiMAX[60]. In this approach, the attacker uses a different gateway compared with the legitimate AP.

3.2 Evil Twin Attack Using Single ISP Gateway

In this section, we first present the adversary model. Then we present several intuitive detection schemes and show that all of them have inherent security holes, making them unfeasible solutions to the ETA using single ISP gateway.

ETA was assumed to be implemented by an attacker with the capability to mimic the legit-

imate wireless network specifications. For example, the IP and the MAC addresses of the DHCP, DNS and the gateway provided by the rogue AP (RAP) are the same as the ones found in the legitimate wireless network. Also, the propagation time between the wireless client (WC) and any other servers can be tuned by the attacker to give a similar result as the legitimate wireless network.

As introduced previously, a RAP in ETA has two options to connect to the Internet: using the same ISP gateway as the legitimate AP (LAP), or by utilizing a different ISP gateway. In this section, our ETA detection focuses on the second type of ETA that uses a different ISP gateway compared with the LAP wireless network.

3.2.1 Intuitive Detection Schemes and Their Security Problems

1) Detection based on route option in IP packet header:

One of the intuitive detection methods that can be used to detect ETA using different ISP gateway is by taking advantage of the record route option found in IP header [6]. When this option is enabled in a packet, routers on the path between the source and destination insert their IP address in the packet IP header. The WC sends an IP packet through a given Access Point (AP_x) that belongs to the legitimate wireless network. Then, the WC switches to another access point (AP_y) that has the same SSID of the legitimate wireless network and send the second packet. The record route option should be enabled in these two packets, and the destination address of these two packets is a special server on the Internet. When the server at the other end receives these packets, it will match between the routers' addresses recorded in the packet's IP header. The WC can view the results on the server using a secure protocol.

However, similar to the traceroute packets [48], record route packets may be dropped or ignored by many firewalls for security reasons [49]. In addition, only at most nine IP addresses can be registered along the route while the average number of routers in any given route on the Internet is 19 to 21 [75].

2) Detection based on TCP connection:

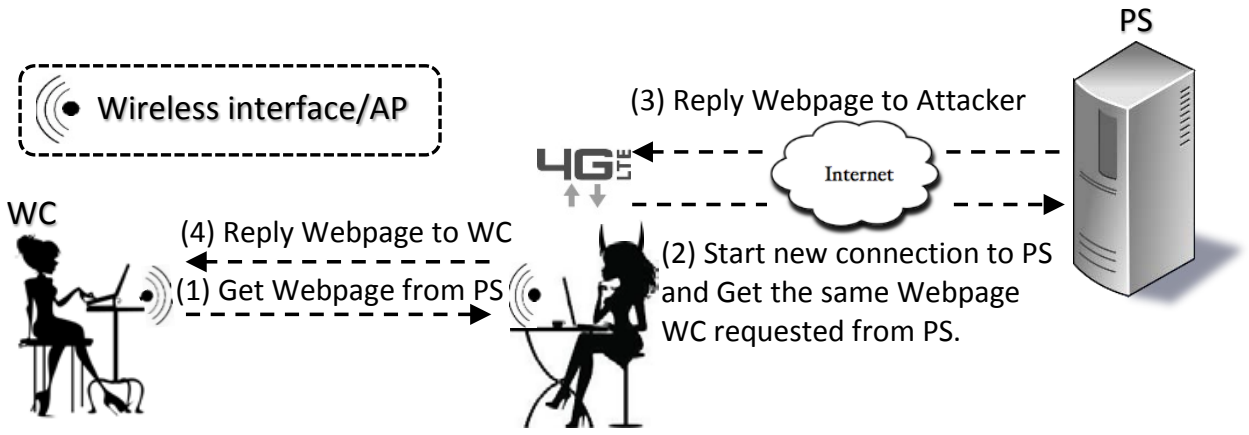


Figure 3.1: Possible man-in-the-middle attack on the ETA detection that relies on TCP connection without security.

The second intuitive detection method that can be proposed to detect ETA using different gateway is by dividing TCP communication. The detection procedure will start after a wireless client initiates a wireless connection to a nearby AP. This AP (we call it AP_x) should have the wireless SSID name such as FreeWiFi (Figure 1.2) that belongs to the legitimate wireless network. After connecting to AP_x, the WC starts a TCP 3-way handshake to a random remote web server such as www.google.com. Each side (the WC and Google server) creates a socket connection that contains the IP address and the Port number for the other side.

After completing a successful TCP 3-way handshake through AP_x, the WC switch to a different AP (we call it AP_y) that has the same SSID name (FreeWiFi). The WC does not start a new TCP 3-way handshake to the remote web server since the TCP connection is already established using AP_x. Changing the AP does not have any effects on the socket information stored on each side of the connection. After switching to AP_y, the WC sends a GET HTML request to download an index web page from the remote web server.

If the two APs use the same ISP gateway, the TCP connection will not break, and the WC start downloading the index web page from the remote web server successfully. Otherwise, if the

TCP connection dropped or the WC didn't receive any response from the remote server, it means that these two APs are using different ISP gateways. Using different gateways prevented the web server from sending the index web page to the WC because the IP address and/or the port number of the WC is different through each APs.

An attacker can conduct MIMA to the above detection method by impersonating the remote web server role. This MIMA can take place when the WC starts downloading the index web page though AP_y (which is the RAP). The attacker at this point can catch the GET HTML request from the WC and start a new connection to the remote web server and retrieve the index web page. Then, because the attacker can monitor the TCP connection setup between the WC and AP_x, the attacker can send the index web page to the WC by continuing the existing TCP connection. This MIMA is illustrated in Figure 3.1.

3.2.2 Proposed Detection Design

3.2.2.1 Design

The design of the proposed ETA detection method for detecting different gateways is based on the following assumptions: network administrators may deploy more than one AP for better quality and wider coverage. However, all APs belonging to the same wireless network will always use a single gateway for Internet access. This type of wireless network topology can be found in coffee shops, hotels and airports [37]. Also, network administrators in these wireless networks usually assign private IP addresses to their wireless customers. These private IP addresses will eventually translate into the public IP of the gateway using network address translator (NAT) or port address translator (PAT) [76].

3.2.2.2 Proposed ETA Detection

The detection relies on secure TCP connection for web page retrieval in a similar way as the second intuitive TCP-based detection method introduced in Section 3.2.1. When the WC starts the detection procedure, it initiates a TCP 3-way handshake though AP_x using a secure connection to an arbitrary remote web server that supports HTTPS connections (such as to https://www.google.com). Then, the WC switches Internet access via AP_y and issues HTTPS GET command to retrieve web page content.

By using a secure connection, we can prevent an attacker from applying the MIMA illustrated in Figure 3.1 since the attacker does not have the current TCP session's information to continue the secure TCP connection with the WC.

Our proposed detection method distinguish whether two access points with the same SSID use the same network gateway or not. If there are more than two APs existed in a wireless network, our detection schemes work in the same way by checking each AP one after another to find whether all existing APs use the same gateway or not.

The detection method will be on the WC side which is more desirable than the administrator-side detection. The client-side design gives a security-sensitive user more control over her wireless connection security and can be used in any wireless network regardless of what security mechanism has been implemented.

In addition, no fingerprint is used in the detection. The client does not need to have any previous information about the APs installed in the wireless network. Furthermore, the detection method is not based on a protocol or a protocol option (such as ICMP or record route option) that might be blocked by network administrators for security purposes.

Pseudo Code 3.1: Proposed Evil Twin Attack Detection on ETA using different ISP gateway.

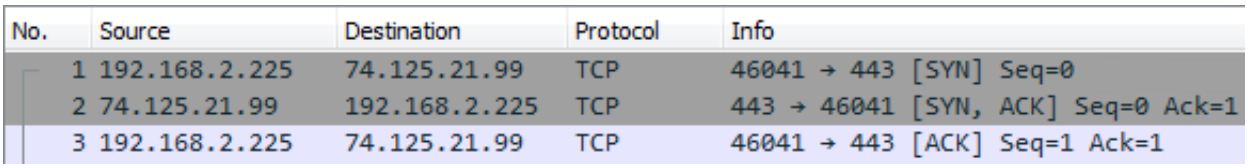
```
1  Connect to AP_x
2  Start secure TCP 3-way handshake to www.google.com
3  Verify www.google.com certificate
4  if www.google.com certificate is valid then
5  |  Switch to AP_y
6  |  GET command to download index.html
7  |  if www.google.com starts sending the index.html webpage then
8  |  |  Print no ETA detected
9  |  else
10 |  |  The connection was dropped or rejected
11 |  |  Print ETA detected
12 |  end
13 else
14 |  Print server certificate error!!
15 end
```

3.2.2.3 Implementation

The ETA detection client software prototype was implemented using C language and executed on a Linux machine. In our implementation, we used LORCON2 [77] library to communicate with the web server. The program automatically starts a secure TCP socket connection through the first AP with an arbitrary web server, and then start downloading the index web page from the web server using the second AP.

The web server used in our prototype was www.google.com because it is more reliable than most other web servers, and most importantly, it has a long time-to-live secure TCP connection (240 seconds based on our measurements). This give the WC plenty of time to switch from one AP to another without the secure TCP connection to have a timeout.

Since there will be more than one AP with the same SSID, WC connects first to the AP_x using its MAC address. The MAC address is used as a reference to switch between different APs that belongs to the same SSID. After finishing the secure TCP 3-way handshake, the program will automatically switch to the second AP (AP_y) and start downloading the index web page from the



No.	Source	Destination	Protocol	Info
1	192.168.2.225	74.125.21.99	TCP	46041 → 443 [SYN] Seq=0
2	74.125.21.99	192.168.2.225	TCP	443 → 46041 [SYN, ACK] Seq=0 Ack=1
3	192.168.2.225	74.125.21.99	TCP	46041 → 443 [ACK] Seq=1 Ack=1

Figure 3.2: Secure TCP 3-way handshake using AP_x.

web server. The Pseudo Code 1 illustrates our proposed ETA detection.

3.2.3 Evaluation

3.2.3.1 Evaluation Procedure

In our testbed set up, two APs (Dlink DIR-890L and Asus AC1900) were used to represent both AP_x and AP_y, respectively. Wireshark [78] is used to capture network traffic.

The first part of our evaluation procedure was to verify if both APs that belongs to the same ISP gateway would not trigger any alarm using our proposed design. We connected both APs (AP_x and AP_y) to the same ISP gateway. The WC software recorded the MAC addresses of both APs and randomly connected to one of the APs, in our case, it was AP_x. Through AP_x, the WC obtained network configuration from the DHCP server. The connection information between the WC and the web server is shown in Figure 3.2. The IP address obtained by the WC was in the private IP range (192.168.2.225) and the source port address that was used in the 3-way handshake was (46041). However, both of the WC IP and port were translated to the public IP address and port of the gateway using NAT/PAT. On the other side, the remote Google server had an IP address of 74.125.21.99 with the port 443.

At the end of the 3-way handshake procedure, the web server created a socket connection using the public IP address and port given to the WC at the ISP gateway. The WC also created a socket connection using Google public IP and port. During the handshaking, the WC verified Google server digital certificate. The WC stops communicating with AP_x at this point and switched

No.	Source	Destination	Protocol	Info
15	192.168.2.225	74.125.21.99	TLSv1.2	Application Data
16	74.125.21.99	192.168.2.225	TCP	443 → 46041 [ACK]
17	74.125.21.99	192.168.2.225	TLSv1.2	Application Data
18	192.168.2.225	74.125.21.99	TCP	46041 → 443 [ACK]

Figure 3.3: Successfully downloading index webpage using AP_y.

to AP_y.

Although the WC switched in the middle of an active secure TCP connection between the two APs, the Wireshark did not catch any connection termination packets sent from the web server or the AP_x to the WC. The WC can use the active connection to the web server through AP_y. The WC used the active connection on AP_y to send a GET command to retrieve the HTML index page from the Google web server as shown in Figure 3.3.

The ETA detection software notifies the WC of a safe wireless network when the remote server (Google) replies to the HTML get request. The remote server can only response to the WC get HTML request when the socket connection information used thought AP_x matches the socket connection information used through AP_y.

The second part of our evaluation procedure was to make AP_x AP_y use different ISP gateways. In this scenario, the WC private IP address was also 192.168.2.225 and Google web server IP was 216.58.192.68. Similar to the first part of our evaluation procedure, the WC started a secure TCP connection through AP_x and switched to AP_y. The WC sent GET HTML request through AP_y. However, since AP_y used different ISP gateway than the ISP gateway used by AP_x, the HTML GET request socket information did not match the socket information used to create the secure TCP connection. Based on the configuration of the remote web server, it can send a rest TCP connection to the WC or drop the HTML GET request as shown in figure 3.4. The WC sends multiple GET request to Google server but no response was received. Without receiving a positive

No.	Source	Destination	Protocol	Info
15	192.168.2.225	216.58.192.68	TLSv1.2	Application Data
16	192.168.2.225	216.58.192.68	TCP	[TCP Retransmission]
17	192.168.2.225	216.58.192.68	TCP	[TCP Retransmission]
18	192.168.2.225	216.58.192.68	TCP	[TCP Retransmission]

Figure 3.4: Wireless client unable to download the webpage when AP_x and AP_y used different gateways.

response from the remote server, the WC software notifies the client of a possible ETA on the given wireless network.

3.2.3.2 Detection Time Delay Analysis

Unlike [9], our proposed detection method is not based on time measurements. The fluctuation of APs response time due to the increase/decrease of wireless traffic would not interfere with the detection performance. Nevertheless, the time delay is still a vital performance metric. Therefore, we have analyzed time delay in our evaluation. The wireless APs used in our testbed were Dlink DIR-890L and Asus1900. These APs also operated as DHCP, DNS servers, and gateway. The WC software installed on Linux-based OS with a Penguin wireless N USB adapter.

We measured the time delay for four main steps in the detection procedure:

- The time to connect to AP_x and obtain a valid network configuration from the DHCP server.
- The time to finish the secure 3-way TCP handshake.
- The time spent to switch from AP_x to AP_x and obtain/reuse a valid IP from the DHCP server.
- The time to receive a response from the web server.

The test was repeated 50 times for each measurement. At the beginning of each trial, the APs (including DHCP, DNS, and the gateway) was turned off and back on to ensure fresh reading.

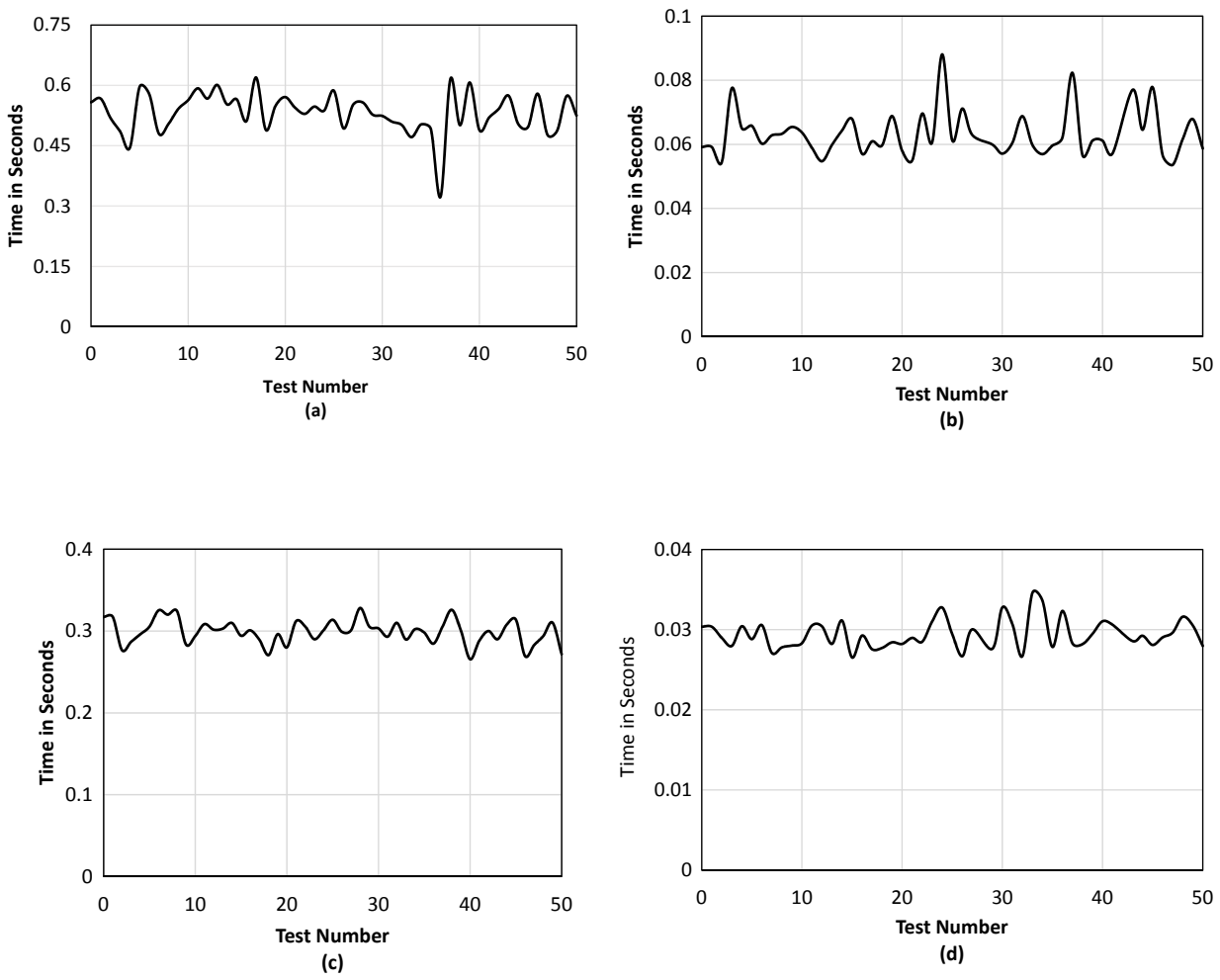


Figure 3.5: ETA detection procedure time duration when AP_x and AP_y use the same gateway. (a) connecting to AP_x. (b) secure 3-way TCP handshake. (c) switching to AP_y. (d) receiving a response from the web server.

The results of the testbed measurements when the two APs used the same gateway is illustrated in Figure 3.5 .

The time spent to connect to AP_x was 0.5 seconds. This time includes 1) the authentication and association time to AP_x, 2) the duration time to receive a valid network configuration from the DHCP. The average time to switch from AP_x wireless channel to AP_y wireless channel was about

0.3 seconds. The WC didn't request a new IP address but utilize the one used in AP_x.

The time duration to finish the 3-way handshake and to receive a response from the web server was relatively shorter than the connecting time. In our testbed, fast Internet speed link was used ($>$10 Mbps). The average time duration to complete the 3-way handshake was 0.06 seconds, while the average time to receive a response from the web server was 0.03 seconds. These time values depend on many factors such as the Internet speed, DNS response time and web server's response time.

In the end, we want to emphasize that although the test time of the detection method may vary according to many factors as explained above, these factors will not affect the detection efficiency of the proposed technique.

3.2.4 Discussion

Although we had only two APs in our testbed LAP and RAP, If the client receives more than two AP signals, our detection method can be used without any change to switch between each reachable APs one by one. Each AP switching should be done in the middle of the secure TCP connection. If anyone of these APs uses a different gateway, the secure TCP connection will break, and WC will be notified.

The time needed to connect to AP_x was larger than AP_y. After connecting to AP_x, the WC received valid network configuration from the local DHCP. To speed up the detection process, after connecting to AP_y, the WC reused the same network configuration gained from AP_x. The connection time from one wireless network to another may vary, and it depends on the manufacturing types and models of the wireless network devices.

In our method, client software verified the remote server's certificate to prevent the attacker from creating a fake remote server to bypass our detection procedure. Our ETA detection is not vulnerable to MIMA such as SSL strip attack [13]. Our proposed ETA detection starts its communication on port 443, SSL strip attack [13] is not feasible since that attack is based on the transition

between port 80 and port 443.

Our proposed ETA detection scheme has its limitations. We discuss these limitations below.

First, we stated that our detection method was focused on detecting ETA using different gateways. If the attacker uses the same legitimate gateway to pass client data, our detection method will not work. However, combining our detection method with our ETA detection method in section 3.3 that were used to detect ETA using single ISP gateway would produce an effective and comprehensive ETA detection system.

Second, the proposed detection method spends about 0.3 seconds when switching from one AP to another as shown in Figure 3.5-C. This requires that the web server should have a long time to live (TTL) secure TCP session to allow the client to switch between the APs without dropping the connection. In our prototype evaluation, google web servers were selected because they support secure TCP protocol such as TLS/SSL and they also have a long TTL TCP session. We measured the TTL value of TCP session for www.google.com, and the result was 240 seconds.

Third, upon detecting the presence of ETA, our detection method is not able to identify which AP is rogue and which one is legitimate. Because both the legitimate AP and the rogue AP provide Internet access that could have the similar quality, it is very challenging to further distinguish them apart with only client-side actions.

Finally, if the client receives only rogue AP(s) signals without any legitimate AP, our detection method will not work as well. This weakness can be found in all client-based ETA detections that do not use authorized AP-list. The client cannot detect ETA since all the AP(s) will give the consistent fake results.

3.3 Evil Twin Attack Using Single ISP Gateway

In this section, we present our ETA detection on ETA using single ISP gateway. The attacker uses the LAP to pass through WC data instead of using her mobile broadband connection.

In this case, all data sent from the wireless network will be originated from one ISP gateway.

3.3.1 Intuitive detection schemes and their security vulnerabilities

Open WiFiHop [51] is a client-side ETA detection of ETA using single ISP gateway. The detection structure is composed of a WC and a dedicated public server. First, the WC connects to a nearby AP and send a watermarked packet to the public server. The watermarked packet is a random bit stream that is only known to the WC. After the WC sends the watermarked packet to the public server, the WC immediately switches to other Wi-Fi channels looking for any transmission of the watermarked packet. The public server will keep replying this watermarked packet to the WC. If the WC captures the watermarked packet in other Wi-Fi channels then the initial AP is RAP, else it is LAP.

Based on the procedure described above, Open WiFiHop has the following vulnerabilities and limitations.

First, open WiFiHop is vulnerable to replay attack. The public server only reply the watermarked packet to the WC without any modification. When the WC sends the watermarked packet to the public server, the attacker can store the watermarked packet and then disconnect from the LAP. The attacker can then start sending the stored watermarked packet to the WC. Since the attacker disconnected from the LAP, no watermarked packet is sent on other Wi-Fi channels. In addition, when the WC returns back to the initial AP, the attacker can connect to the LAP. In this scenario, Open WiFiHop will fail to detect ETA.

Second, the attacker can avoid Open WiFiHop detection by gathering information about the watermarked packets replay arrivals time and, the round trip time between the public server and the WC. When the WC sends the watermarked packet to the public server, she immediately switch to other Wi-Fi channels looking for the watermarked packet [51]. The attacker can simply disconnect from the LAP without even replying the watermark packet since the WC is checking other Wi-Fi channels. When the WC returns back to the initial AP, the attacker can reconnect

to the LAP. At this point, the WC will start receiving the watermarked packets from the public server. The attacker can also estimate when the WC returns to the initial AP simply by capturing the communication between the WC and the public server, which will pass through the attacker in the first place.

In [51], when the public server receives the watermarked packet, it will delay each reply by D time units, which is the time needed by the WC to switch from one AP to another. By measuring the time differences between two public server replies, the attacker can calculate D. Also, the WC will monitor each wireless channel by time $\geqslant (D + RTT)$ where RTT is the round trip time from the WC to the public server. RTT can be easily calculated since the initial communication between the WC and the public server went through the RAP.

In general, ETA detection security should not be based on information that can be gained, calculated and/or estimated by the attacker. In the next section, we propose an ETA detection procedure that overcomes the above vulnerabilities found in [51].

3.3.2 Proposed ETA detection

3.3.2.1 Assumption

Our proposed detection takes advantage of the unique network architecture deployed by the first attack option of ETA using a single ISP gateway: when a WC sends/receives data through RAP, the same wireless data is be sent/received between the attacker's RWC and the LAP. A network administrator may extend 802.11 wireless coverage by installing more than one LAPs, however, these LAPs are connected to network using cables.

Furthermore, our ETA detection is based on a fundamental 802.11 architecture design. When an AP fails to receive an acknowledgment response from a WC, it assumes the transmitted frame was lost due to collision or weak signal [79][80]. The AP keep sending unacknowledged frames for a certain amount of time until it determines that the WC is offline, and then disconnects

it from the wireless network.

3.3.2.2 Proposed Detection Design

Our ETA detection system design overcomes the vulnerabilities in WiFiHop discussed in section 3.3.1. The effectiveness of the detection procedure is not based on parameters that can be gained or estimated by the attacker. Furthermore, the ETA detection is a real-time client-side method that does not rely on training data and/or Wi-Fi network fingerprint.

The proposed ETA system detection is composed of two parts: a WC and the public information server (PIS). First, by listening to the Wi-Fi beacon frames, the WC records the MAC address and the working Wi-Fi channel for all nearby APs that belong to the Wi-Fi network being tested. For simplicity, let us assume we have only two APs in the target Wi-Fi network, AP_x and AP_y. Wi-Fi SSID is used to determine if an AP belongs to the target Wi-Fi or not. The first step does not involve any communication between the WC and any APs.

Second, the WC randomly connects to one of the recorded APs, for example, AP_x. Once the WC is connected to AP_x, the Wi-Fi network DHCP assigns network configuration such as IP address to the WC. Now that the WC is connected to the Wi-Fi network, she establish a connection to the PIS and sends a "hello" packet. Data traffic between the WC and the PIS is encrypted. The PIS will assign a unique ID to the WC, e.g., XYZ. Such ID is capable of telling apart the communication between the WC and PIS from the communication of other WCs that may start the ETA detection at the same time on the same Wi-Fi network. After the WC receives her ID, she sends APx's MAC address along with the WC's ID to the PIS. In the meantime, the WC saves the Wi-Fi network connection information. Likewise, PIS keeps AP's MAC address that belongs to the connection.

Third, the WC switches randomly to other recorded APs (in our scenario is AP_y). At the same time, the WC changes her MAC address. After receiving network configuration using the new MAC address from AP_y, WC starts a new connection to the PIS. After that, the WC

Table 3.1: Info Packet Data

Packet Seq.	WC ID	AP MAC Address
1	XYZ	AP_x
2	XYZ	AP_y
3	XYZ	AP_x
4	XYZ	AP_y

sends AP_y's MAC address along with his/her ID to the PIS. Also, the WC saves the network configuration related to AP_y. In case there are more than two APs, the WC keeps repeating the previous procedure until going through the last recorded AP. As can be seen at this point, the WC is having two completely separate connections to the PIS.

Fourth, through the last connected AP (in our scenario is AP_y), the WC sends "Info Start" packet which signals to PIS to start sending info packets. PIS starts sending info packets to the WC through each connection separately. Info packets contain the MAC address of the AP being used to establish the connection between the PIS and the WC. Also, each info packet has increment sequence numbers to prevent the replay attack, as shown in Table 3.1.

Fifth, immediately after the WC sends info start packet, she randomly switches to one of the APs (AP_x or AP_y) channel and starts listening to the info packets sent by the PIS for a certain amount of time. WC filters all the incoming packets based on the WC's ID. As a result, all filtered wireless frames should have their destination MAC address pointing to one of the WC's MAC addresses. If not, then that frame was sent to an RWC. WC can then extract the MAC address inside the info packet to mark it as RAP. Also, if the WC did not receive an info packet from the AP that belongs to the listening channel, then that AP is also a RAP. Otherwise, the AP is LAP. In addition, the WC checks the sequence number of the info packets and ignores any packet with a sequence number that is less than or equal to the last one received.

Even if the attacker has all the timing information of the PIS sending interval and the WC

switching/listening time, the ETA will be detected because the WC's channel switching is random. The attacker cannot tell if the WC is listing to the RAP or the LAP. If the attacker stops sending info packets while the WC is listening to the RAP channel, our detection will detect the ETA. Also, if the attacker starts sending info packets while the WC is listening to the LAP Wi-Fi channel, the proposed detection will detect that the LAP is sending info packets to other WCs (attacker Wi-Fi interface). Furthermore, every info packet has its own sequence number, the attacker can't apply the replay attack on info packets.

At the end of the detection procedure, the WC marks every recorded AP as RAP or LAP. The WC now can freely connect to any of the LAPs. The PIS deletes all the information related to the WC's ID XYZ. This makes the PIS simple to implement and maintain.

3.3.2.3 Proposed Detection Efficiency

In our ETA detection, the WC monitors all the recorded APs' Wi-Fi channels randomly. Given the attacker has all our ETA detection timing, she should decide when to disconnect/connect from the LAP to avoid being detected. Since info packets have encrypted sequence numbers, the attacker cannot save a copy and reply it to the WC. When the attacker disconnects from the LAP, she cannot send any info packets using the RAP. Since the WC monitors each APs' Wi-Fi channel for one time unit, the WC ETA detection missing probability P_m can be calculated as:

$$P_m = \frac{k}{N} \times \frac{N-k}{N} \tag{3.1}$$

where N is the number of recorded APs' Wi-Fi channels and k is the number of times the attacker disconnect/connect from/to the LAP. The attacker's goal is to find the best value for k in order to maximize the detection missing probability P_m. This can be calculated by finding the roots of the

Table 3.2: Proposed ETA using single ISP gateway detection/missing probability

Monitor Ch. Freq.	Miss Probability	Detection Probability
1	25%	75%
2	6.25%	93.75%
3	1.5625%	98.4375%
4	0.390625%	99.609375%

P_m's derivative, given as:

$$\frac{dP_m}{dk} = \frac{N - 2k}{N^2} \tag{3.2}$$

The roots of Equation (3.2) is 0 and N/2. Applying k = N/2 to Equation (3.1) yields $P_m = 0.25$. Given that $P_m = 0.25$, the WC's ETA detection probability $P_d = 1$ - $P_m = 0.75$. To increase P_d, we increased the number of times the WC monitors each recorded AP's Wi-Fi channel as shown in Table 3.2. Monitoring each recorded AP's Wi-Fi channel for four times makes our proposed ETA detection probability $\approx 100\%$.

3.3.2.4 Implementation

The ETA detection WC and PIS software were implemented using C language. LORCON2 [77] is used to allow the WC to inject/receive frames into a Wi-Fi network. Both WC and PIS software were installed on Linux OS based machines. TCP protocol is used to carry out communication between the two of them.

WC starts by using LORCON2 to inject/receive wireless frames using Wi-Fi interface card. As soon as the WC connects to the AP, she starts communicating using UDP protocol with the Wi-Fi DHCP server. The Wi-Fi network DHCP server sends the network configuration to the WC. Immediately, the WC initiates a connection to the PIS and receives her ID. We used TCP protocol to implement the communication between the WC and the PIS. Although UDP can be used

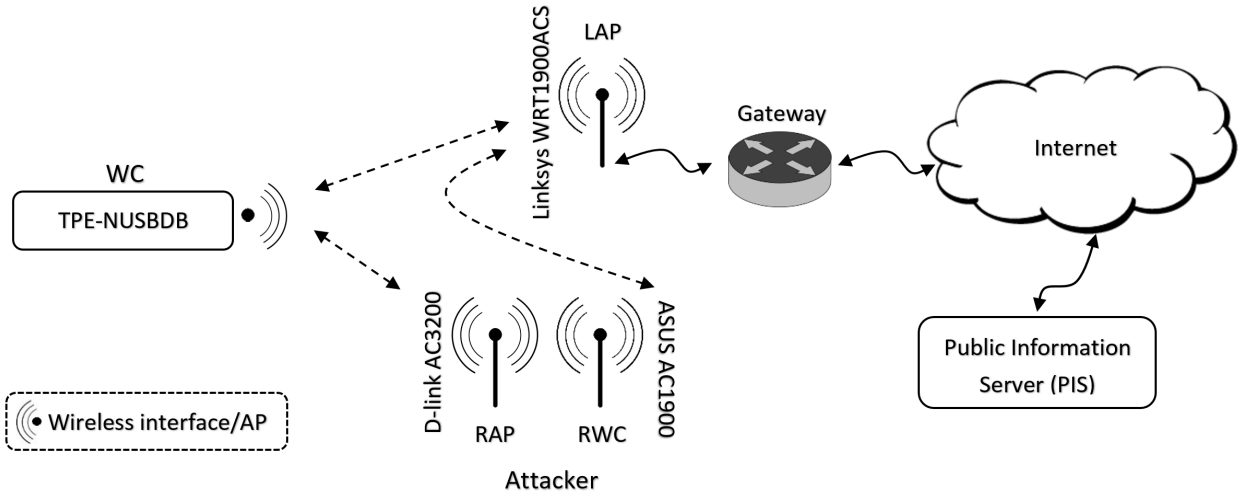


Figure 3.6: Proposed ETA on single ISP gateway evaluation testbed set up.

to establish the connection between the WC and the PIS, TCP is preferred since it is a more reliable protocol compared to UDP. Furthermore, the data between the WC and the PIS is encrypted. Pseudo Code 3.2 illustrates the proposed ETA detection procedure.

3.3.3 Evaluation

We implemented a Wi-Fi network testbed to evaluate our proposed ETA detection. Wireshark software [78] was used to monitor all communications between the WC and the PIS. Both the WC and the PIS software were installed on Linux based OS. The WC interface card is wireless N dual-band USB adapter (TPE-NUSBDB). We assumed the attacker used D-link AC3200 Wi-Fi router to set up the RAP, and ASUS AC1900 Wi-Fi router to connect to the LAP were the LAP is Linksys WRT1900ACS Wi-Fi router. Figure 3.6 illustrates the testbed set up.

First, the WC listened to the Wi-Fi beacon and recorded the APs information such as the working channel and the MAC address. In our testbed, the WC recorded the working channels and MAC addresses of D-link AEnterpriseUserSideETA200 (RAP) and Linksys WRT1900ACS (LAP). After that, the WC randomly connected to one of the APs, e.g., RAP. After receiving

Pseudo Code 3.2: Proposed ETA detection Procedure on ETA using single ISP gateway.

```
Recored nearby APs info.  forming target SSID
Randomly connect to one of the recorded APs
Get network conf.  from DHCP server
Establish secure connection to PIS
Send "hello" pkt.  to PIS
Get WC ID from PIS
Send current AP MAC Addr.  and WC ID to PIS
Save connection info.
while not connected to all other recoreded APs do
    Change WC MAC Addr.
    Randomly connect to one of the remaining APs
    Get network conf.  from DHCP server
    Establish secure connection to PIS
    Send current AP MAC Addr.  and WC ID to PIS
    Save connection info.
end
Send "Info start" pkt.  to PIS
PIS Start sending Info pkts each D sec
while Each AP channel should be monitored four times do
    Randomly switch to one of the APs ch.
    Filter traffic based on WC ID
    Read all Filtered Info pkts
    if Info pkt was found then
        if Info pkt Seq. ≤ than previous one then
            Ignore Info pkt.
        end
        else
            if Wireless frame not sent to WC then
                Extract AP MAC addr.  from info pkt Mark extracted AP MAC
                 Addr.  as RAP.
            end
            else
                Ignore Info Pkt.
            end
        end
    end
    else
        Mark AP belongs to current ch.  as RAP
    end
    Mark non RAP marked APs as LAP
end
```

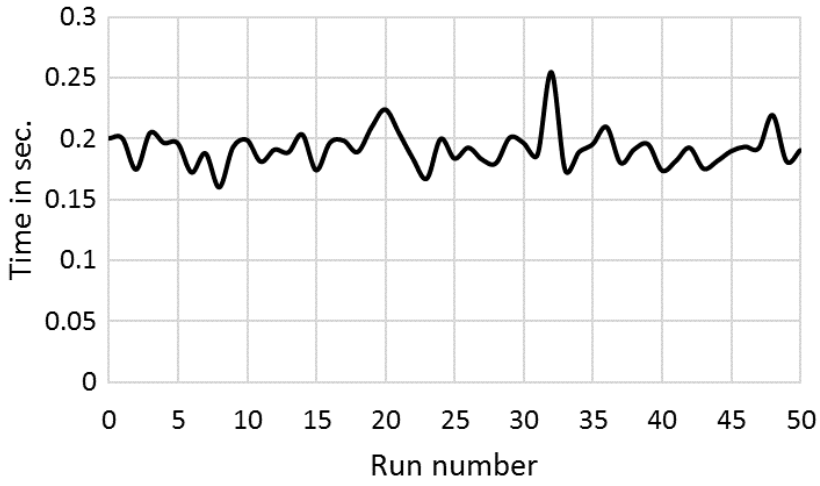


Figure 3.7: WC channel switching time form one AP to another.

network configuration from the DHCP server, the WC established a secured connection to the PIS and received her ID. Immediately, the WC sent RAP MAC address along with her ID. The WC saved network configuration.

Second, the WC changed the Wi-Fi interface MAC address and switched to the LAP. Since the MAC address was changed, new network configuration received from the DHCP server. The WC started a new connection to the PIS and sent LAP MAC address with her ID to the PIS. Now, the WC has two active connections to the PIS through both, the RAP and the LAP. Until now, the real testing has not started yet.

Our ETA detection started when the WC sent "info start" packet to the PIS. For comparison purposes, we used the same timing technique used in [51]. The PIS started sending Info packets at an interval of D seconds each, where D is the time required for the WC to switch from one AP to another. In our testbed, which based on 50 runs, the average value of D was ≈ 0.2 second with a standard deviation of 0.015 seconds as shown in Figure 3.7. Also, the WC should spend longer than $(D + RTT)$ second to monitor each Wi-Fi channel [51], where RTT is the Round Trip Time between the WC and the PIS. Based on 50 runs, Figure 3.8 shows the RTT measured between the

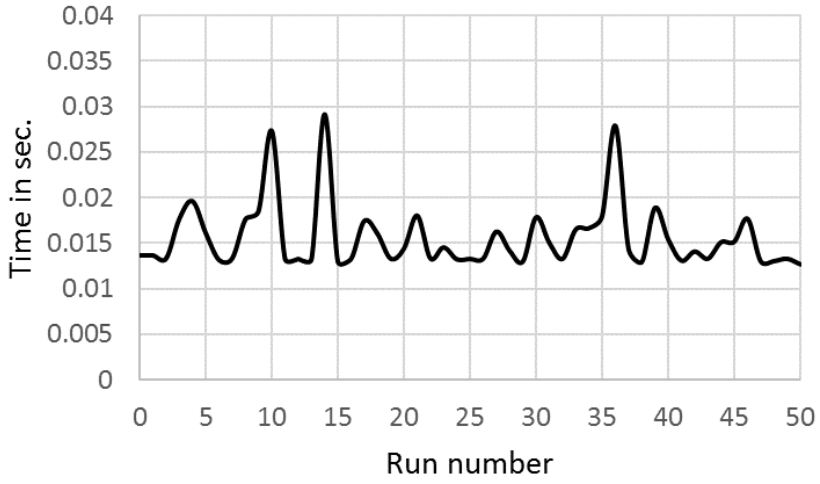


Figure 3.8: Round trip time between WC and PIS.

WC and the PIS which was ≈ 0.016 second with a standard deviation of 0.0037 seconds. As a result, the WC should monitor each Wi-Fi channel longer than (0.2 + 0.016) second. Based on that, we chose for the WC to monitor each Wi-Fi channel for 0.4 seconds. Furthermore, to avoid being affected in case the info packets were lost/dropped along the route between the PIS and the WC, the PIS continuously sends multiple info packets once in every D time.

Since each channel should be monitored four times, Equation (3.4) calculated our ETA detection time based on the number of APs Wi-Fi channels available in the network.

$$DetectionTime = N * (2.4) \tag{3.3}$$

Where N is the number of Wi-Fi channels to be tested, and 2.4 is the total time to test each Wi-Fi channel which came from calculating $4 \times (0.4 + 0.2)$. For example, based on Equation (3.4), our ETA detection spend 26.4 seconds to monitor all the 11 Wi-Fi channels in 802.11 b/g network.

Although WC had to wait 0.4 sec on each wireless channel, in our 50 runs, WC was able to capture LAP, RAP and RWC info packets sent by PIS in an average of ≈ 0.06 second with a

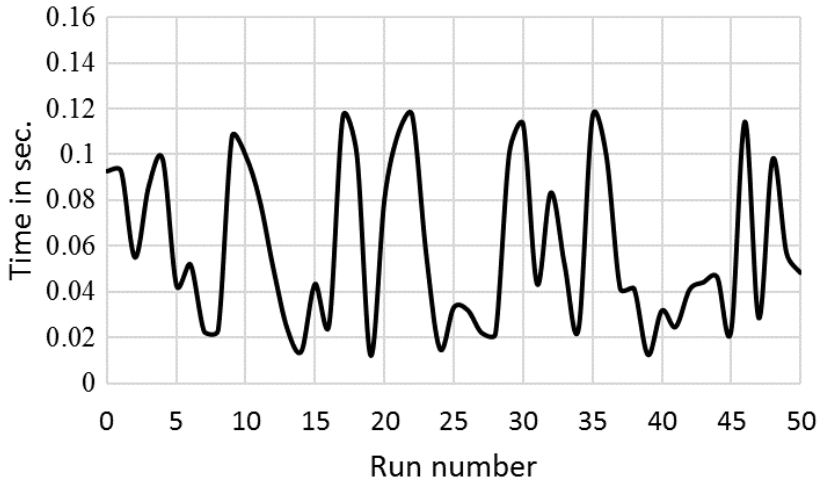


Figure 3.9: Info frames capturing time.

standard deviation of 0.03 second as shown in Figure 3.9 . WC captured info packets in less than 0.4 seconds because PIS will keep sending multiple packets to the WC each time D which is equal to the switching time of the WC. By the time WC switch from one AP to another, info packets should have been already sent by the PIS and on its way to the WC.

3.3.4 Discussion

In this section, we presented an effective ETA detection of ETA using single ISP gateway. If the attacker uses her broadband network connection, this ETA detection will fail. However, combining our detection with ETA detections of ETA using different ISP gateways presented in section3.2, will produce a complete detection tool that can be used to detect both ETA scenarios.

Our ETA detection can test all the 11 802.11 b/g WiFi channels in around half a minute with a detection rate close to 100%. Meanwhile, in Open WiFiHop [51], spend around the same time to test only one AP. Furthermore, our proposed detection is more secure since it was not based on parameters that can be projected by an attacker. For example, unlike Open WiFiHop [51], if the attacker has all the procedure timing information, our ETA detection efficiency will not be affected

and is always approximated to 100 %.

The proposed ETA detection does not rely on training data and/or the Wi-Fi's network fingerprint, which makes it preferable for customers (such as travelers) who visit the Wi-Fi network for the first time. Furthermore, the WC will be the one who ensures his/her security. In addition, the PIS used in our ETA detection is simple to implement and maintain. No WC data will be saved on the PIS, which ensures user privacy in case the PIS was compromised.

Network administrators may extend a Wi-Fi network coverage by setting up repeaters. In general, Wi-Fi repeaters are installed in places that do not have Ethernet port. In IEEE 802.11, Wi-Fi repeater traffic uses all the four address fields in the wireless traffic frame; however, LAP, WC, and RAP use only three address fields [81]. Our proposed detection can check the number of addresses used in the Wi-Fi frame to distinguish between the two types of traffic.

Finally, the WC should be within the wireless coverage area of both the LAP and RAP to detect the ETA. We assumed the network administrators wirelessly covered the designated network area (such as coffee shops, etc.) by using LAPs. When the attacker set up her RAP, she will be within that designated wireless network area. The same assumption applies to the WC.

3.4 Gateway Independent Evil Twin Attack Detection

Insecure Wi-Fi networks provide a tempting environment for attackers to initiate many attacks, one of them is called Evil Twin Attack (ETA). In this section, our proposed ETA detection design is an extension of both [82, 18], wheres [82] used to detect ETA using different ISP gateways while [18] used to detect ETA using single ISP gateway. In this work, we combined these two techniques using virtual wireless clients [35, 36], a novel technique to overcome a major limitation in client side ETA detection. Most of the client side ETA detections that does not relay on training data or pre authorized fingerprint list are gateway dependent [9, 10, 11, 82, 18, 51]. The WC will fail to detect ETA, when she use an ETA detection different from the ETA type the attacker is

Table 3.3: Illustrate different types of ETA detections. ETA detections that receive support from the legitimate network administrator (such as fingerprint list) is categorized as administrator side ETA detection since the detection method would fail without that support.

ETA	Category		ETA ISP Gateway	
Detection	Administrator	Client	Single	Different
[39], [38]	X		X	X
[9, 10]		X	X	
[11]		X		X
[82]		X		X
[18]		X	X	
[51]		X	X	
Proposed		X	X	X

running. However, our new comprehensive design is a gateway independent which limits the ETA false negative. Table 3.3 summarize different ETA detections.

3.4.1 Comprehensive ETA detection

Our proposed ETA using single ISP gateway detection and ETA using different ISP gateway detection can work in parallel using only one physical wireless interface card. To achieve that, a WC creates two virtual wireless clients (VWCs) in which each VWC emulates one standalone wireless client [35][36]. The first VWC (VWC1) implements the ETA detection procedure using single ISP gateway detection while the other VWC (VWC2) implements the ETA detection procedure using different ISP gateway detection.

To make both ETA detection techniques work together, we had to modify the detection procedure from the previous sections. For example, ETA detection using different gateway relies on creating a secure connection for sending and receiving heartbeats to/from PIS instate of Google server. When the WC starts the detection procedure, it initiates a TCP 3-way handshake though AP_x using a secure connection to PIS. Then, the client switches Internet access via AP_y and issues

a heartbeat request to PIS and receive the response from the PIS.

Using our previous scenario of two APs (AP_x and AP_y), both VWCs connect to AP_x using different MAC addresses. The Wi-Fi network DHCP server assigns network configuration such as IP address to both VWCs. Each VWC receives different IP addresses since they have different MAC addresses. Both VWCs start a secure connection to the PIS. VWC1 keeps communicating with PIS to get the unique ID and sends AP_x information. After that, both VWCs switch to AP_y.

During the transition from AP_x to AP_y, VWC1 changes her MAC address while VWC2 keeps her previous MAC address. When both VWCs connect to AP_y, VWC1 receives a new IP address from the DHCP server. VWC1 starts a new connection to the PIS using the newly received network configuration. Then, VWC1 sends AP_y's MAC address along with her ID to the PIS. The VWC1 saves the network configuration related to AP_y. On the other hand, VWC2 reuses her original IP address and sends a heartbeat request to PIS using the secure connection that was created through AP_x.

If VWC2 does not receive heartbeat response from PIS through AP_y, the proposed detection stops and gives the WC a warning that ETA using different ISP gateways is ongoing on the current Wi-Fi network. However, if the heartbeat was received from AP_y then, both VWCs switch to the next recorded AP. In our scenario the last AP was AP_y so, VWC2 informs the WC that both APs (AP_x and AP_y) are using the same ISP gateway. At this point, detection of ETA using different ISP gateways stops, while VWC1 continues the detection process of ETA using single ISP gateway. VWC1 sends info start packet and randomly switches to one of the APs (AP_x or AP_y) channel and starts listening to the info packets sent by the PIS as shown in Figure 3.10

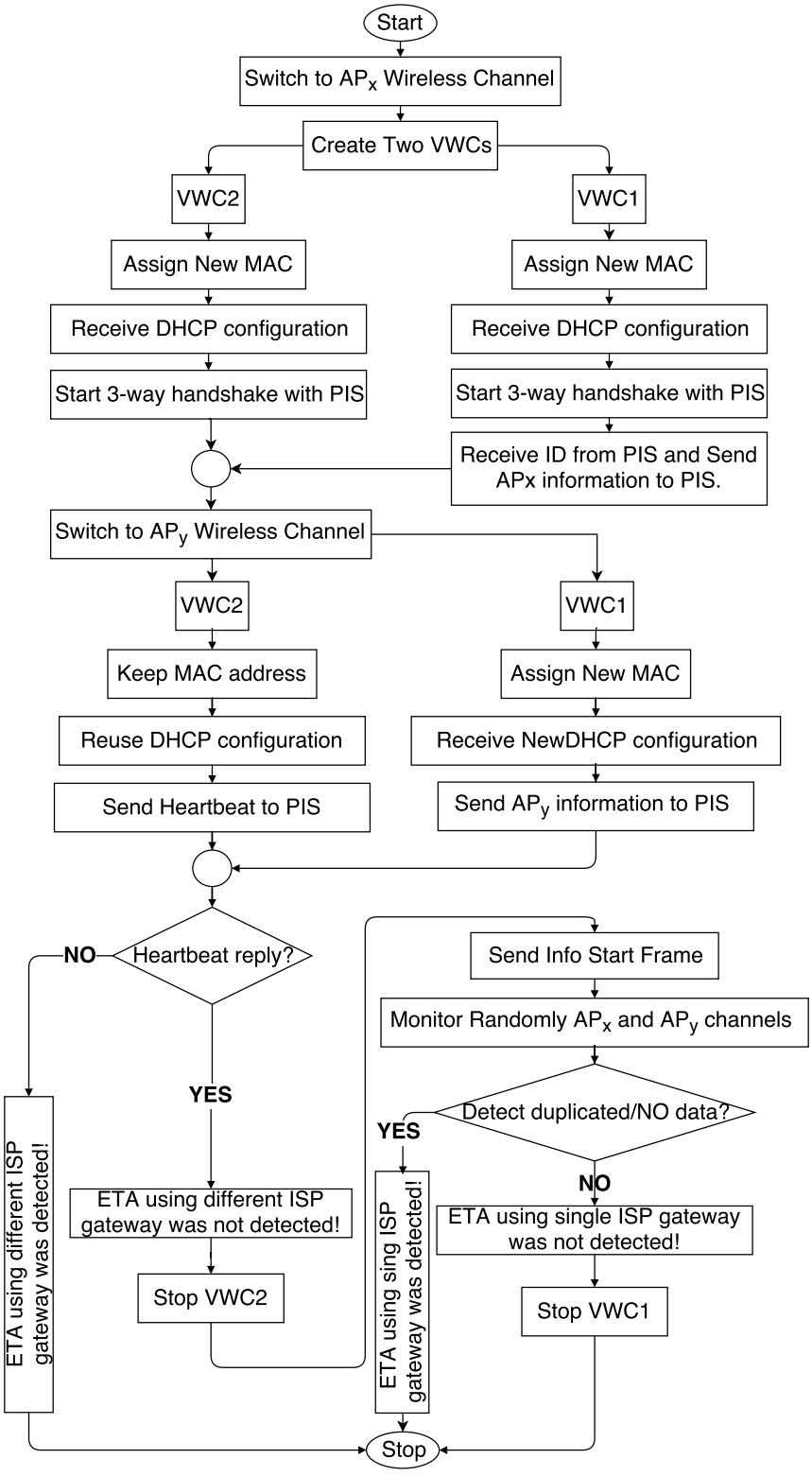


Figure 3.10: Proposed ETA detection on both ETA using single ISP gateway and ETA using different ISP gateways.

Pseudo Code 3.3: Proposed ETA detection Procedure.

```
Record nearby APs info.  having target SSID
Create VWC1 and VWC2
Set different MAC addresses to both VWCs
Both VWCs connect to one of the recorded APs
Both VWCs receive network conf.  from DHCP server
Each VWC establishes a secure connection to PIS
VWC1 Sends "hello" pkt.  to PIS
VWC1 Gets WC ID from PIS
VWC1 Sends current AP MAC Addr.  and WC ID to PIS
VWC1 Saves connection info.
while not connected to all other recoreded APs do
    VWC1 assigns new MAC Addr.
    VWC2 keeps original MAC addr.
    Both VWCs connect to one of the remaining APs
    VWC1 gets network conf.  from DHCP server
    VWC2 reuses pervious network conf.
    VWC2 sends heatheats to PIS
    if No heartbeat reply recived from PIS then
        Display ETA using single ISP was detected
        Exit both ETA detection procedures
    end
    VWC1 establishs a new secure conn.  to PIS
    VWC1 sends AP MAC Addr.  and WC ID to PIS
    VWC1 saves connection info.
end
Display ETA using single ISP was not detect
Stop VWC2
VWC1 Sends "Info start" pkt.  to PIS
PIS Start sending Info pkts each D sec
while Each AP channel should be monitored four times do
    VWC1 randomly switchs to one of the APs ch.
    VWC1 filters traffic based on VWC1 ID
    VWC1 reads all filtered Info pkts
    if Info pkt was found then
        if Info pkt Seq. ≤ than previous one then
            Ignore Info pkt.
        end
        else
            if Wireless frame not sent to VWC1 then
                Extract AP MAC addr.  from info pkt Mark extracted AP MAC
                 Addr.  as RAP.
            end
            else
                Ignore Info Pkt.
            end
        end
    end
    else
        Mark AP belongs to current ch.  as RAP
    end
    Mark non RAP marked APs as LAP
end
```

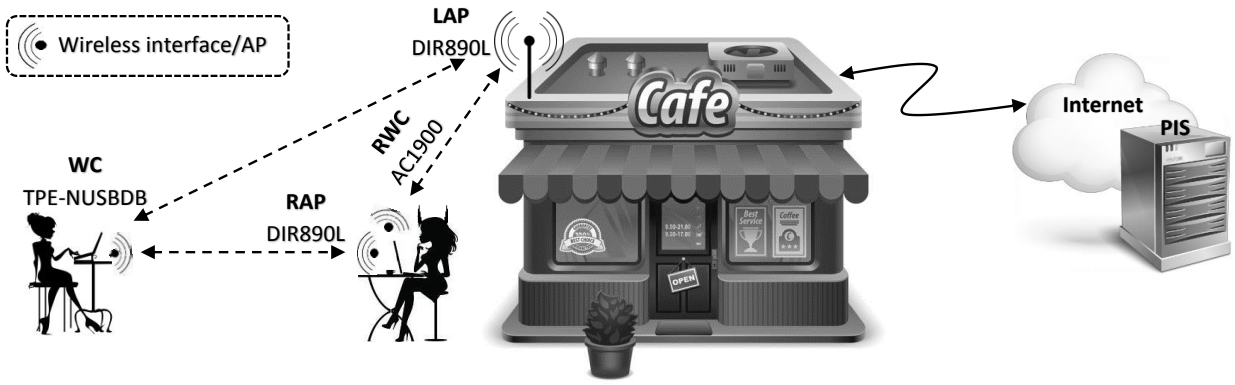


Figure 3.11: Proposed ETA evaluation testbed set up.

3.4.2 Implementation

The ETA comprehensive detection WC/PIS software ware implemented using C language. Both WC/PIS were installed on Linux OS based machines. TCP protocol is used to carry out communication between the two of them. We used Loss Of Radio CONnectivity (LORCON2) [77] library to create multiple VWCs. LORCON2 is an open source library used to create crafted 802.11 wireless frames. WC uses LORCON to inject/receive wireless frames using Wi-Fi interface card. As soon as VWCs connects to the AP, they start communicating using UDP protocol with the Wi-Fi DHCP server. The Wi-Fi network's DHCP server sends the network configuration to both VWCs. Each VWC follows the different procedure to detect the ETA. Pseudo Code 3.3 illustrates the proposed ETA detection design.

3.4.3 Evaluation Procedure

Our proposed ETA detection was tested in real workplaces such as Dunkin' Donuts, Starbucks, and Panera Bread. We also implemented a Wi-Fi network testbed to evaluate our proposed ETA detection. Wireshark software was used to monitor all communications between the VWCs and the PIS. Both the VWCs and the PIS software were installed on Kali Linux OS. The WC Wi-Fi interface card is wireless N dual-band USB adapter (TPE-NUSBDB). We assumed the attacker

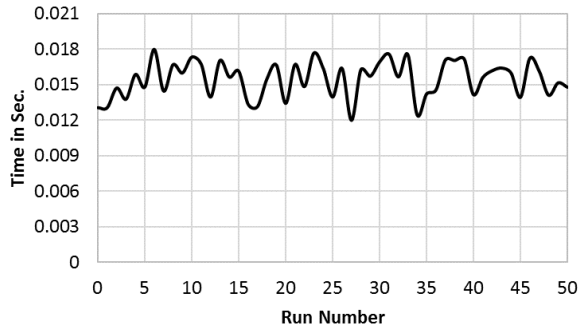


Figure 3.12: Initialize client wireless interface card to operate on RAP wireless channel.

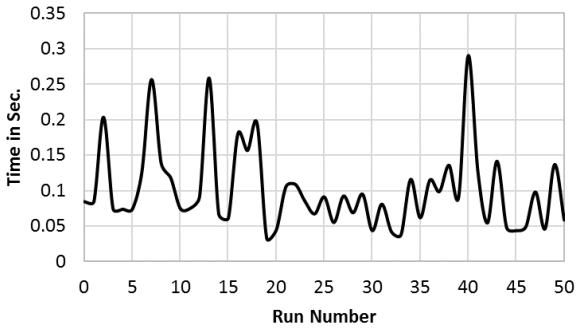


Figure 3.13: Average time for VWC1 and VWC2 to authenticate to RAP.

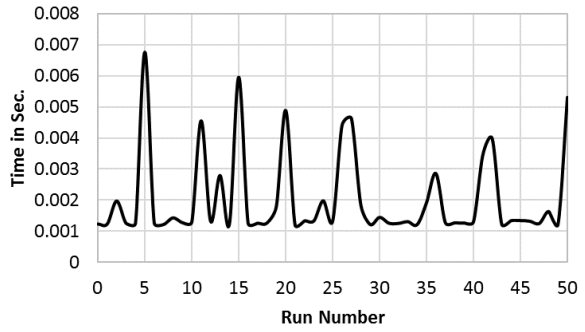


Figure 3.14: Average time for VWC1 and VWC2 to associate phase to RAP.

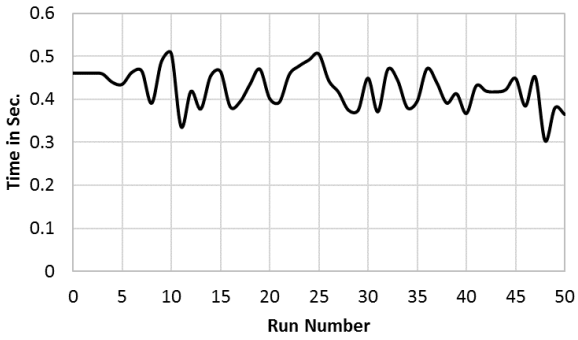


Figure 3.15: Average time for VWC1 and VWC2 to receive DHCP configuration.

used D-link DIR890L Wi-Fi router to set up the RAP, and ASUS AC1900 Wi-Fi router to connect to the LAP. Where the LAP is also D-link DIR890L Wi-Fi router. However, our ETA detection mechanism will work with any other Wi-FI router that can be bought off-the-shelf. Figure 3.11 illustrates the testbed set up. We repeated our proposed ETA procedure trails for 50 runs.

First, the WC listens to the Wi-Fi beacon and records the APs information such as the working channel and the MAC address. In our testbed, the WC recorded the working channels and MAC addresses of RAP and LAP. After that, the WC created two VWCs and randomly connected to one of the APs, e.g., RAP. The average time needed for both VWC1 and VWC2 to complete

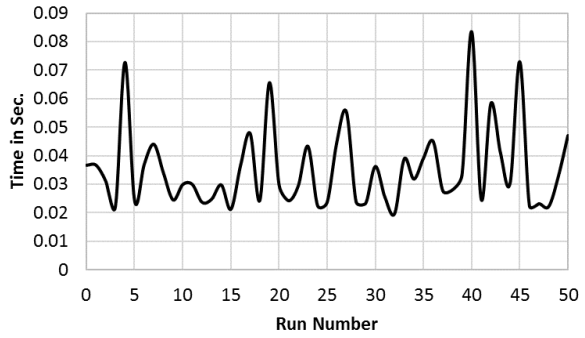


Figure 3.16: Time duration for VWC1 to finish communicating with PIS.

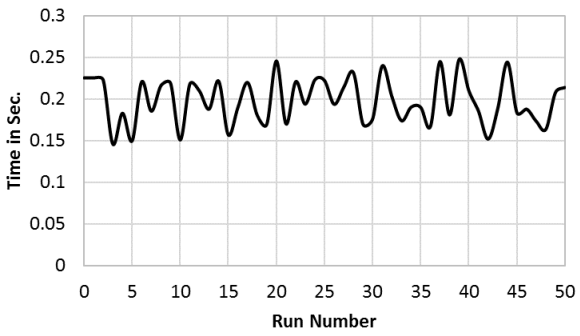


Figure 3.17: Time for WC to switch from RAP operating Wi-Fi channel to LAP Wi-Fi channel

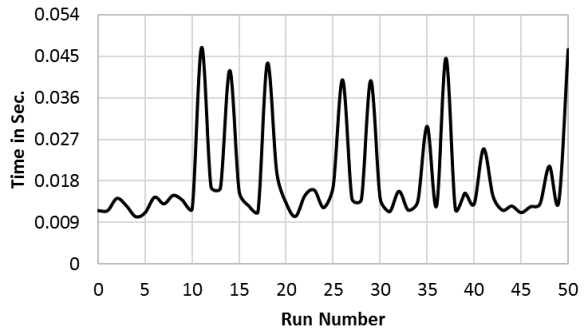


Figure 3.18: Time delay until VWC2 received heartbeats from PIS.

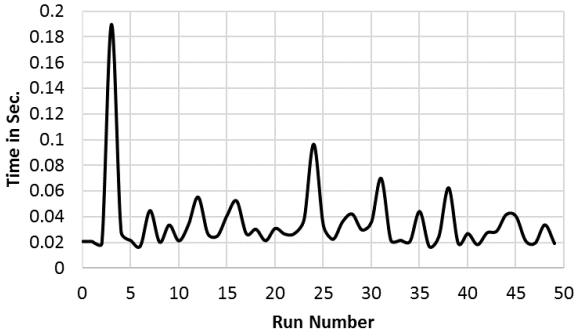


Figure 3.19: VWC1 communication time with PIS including sending ”Info Start” frame.

(1). initialize the wireless interface card to work on the RAP wireless channel., (2). pass the authentication phase, (3). pass the association connection phase, was 0.12 seconds with variance of 0.003 seconds as shown in Figures 3.12,3.13,3.14 respectively. After both VWCs were connected to RAP, they both received network configuration from the DHCP server. The average time to obtain a valid IP address using RAP was 0.42 seconds with variance of 0.0019 seconds as shown in Figure 3.15. After that, both VWCs established a separate secured connection to the PIS. However, only VWC1 received her ID. Immediately, VWC1 sent RAP MAC address along with her ID to the PIS. Both VWCs should finish their procedures at each AP to be able to switch to the next AP.

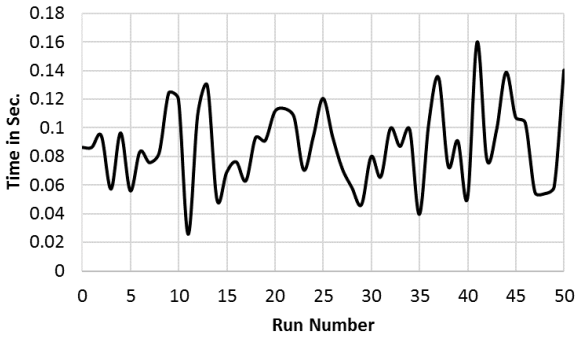


Figure 3.20: Average time delay before VWC1 capture Info frames from LAP, RAP and RWC.

The time needed to finish communicating to PIS on the Internet through RAP was 0.043 seconds with variance of 0.0007 seconds as shown in Figure 3.16.

Second, both VWCs finished communicating with RAP and started switching to LAP. During the switching from RAP to LAP only VWC1 changed her Wi-Fi interface MAC address. The average switching time between RAP and LAP was 0.2 seconds with variance of 0.0008 seconds as shown in Figure 3.17. Since the MAC address of the VWC1 was changed, new network configuration was received from the DHCP server. On the other hand, VWC2 kept its original network configuration because she used the same MAC address. We assumed that both RAP and LAP gave the same exact authentication, association and DHCP response time when communicating with VWCs.

At this point, VWC2 reused the previous connection (network socket) which was set up through the RAP and sent a heartbeat request to the PIS. VWC2 received a heartbeat reply from the PIS since both RAP and LAP used the same public IP address to communicate with the PIS. VWC2 displayed a message to the WC that both RAP and LAP are using the same ISP gateway. The time needed for VWC2 to receive a positive reply from the PIS was 0.018 seconds with variance of 0.00012 seconds as shown in Figure 3.18. VWC2 spent about 0.9 seconds to finish detecting ETA using different gateways.

In the meanwhile, VWC1 started a new connection to the PIS and sent LAP MAC address with her ID to the PIS. Now, the VWC1 has two active connections to the PIS through both the RAP and the LAP. Until now, the real testing of ETA using single ISP gateway has not started yet.

Our ETA detection for the ETA using single ISP gateway started when VWC1 sent "info start" packet to the PIS. "info start" packet was sent after VWC1 finished communicating with PIS which was around 0.035 seconds with variance of 0.0007 seconds as shown in Figure 3.19. For comparison purposes, we used the same timing technique used in [51]. The PIS started sending Info packets at an interval of D seconds each, where D is the time required for the VWC1 to switch from one AP to another. In our testbed, which was based on 50 runs, the average value of D was $\approx$ 0.2 seconds with standard deviation of 0.0008 seconds as shown in Figure 3.17. Also, the VWC1 should spend longer than $(D + RTT)$ seconds to monitor each Wi-Fi channel [51], where RTT is the Round Trip Time between the VWC1 and the PIS. The RTT measured between the VWC1 and the PIS was $\approx$ 0.016 seconds with a standard deviation of 0.005 seconds. As a result, the VWC1 should monitor each Wi-Fi channel longer than (0.2 + 0.016) seconds. Based on that, we chose for the VWC1 to monitor each Wi-Fi channel for 0.4 seconds. Furthermore, to avoid being affected in case the info packets were lost/dropped along the route between the PIS and the VWC1, the PIS continuously sent info packets once every D seconds.

Since each channel should be monitored four times to have $\approx$ 100 detection rate (Table 3.2), our ETA detection time based on the number of APs Wi-Fi channels available in the network can be calculated as:

$$DetectionTime = N * (2.4) \tag{3.4}$$

where N is the number of Wi-Fi channels to be tested, and 2.4 is the total time to monitor each Wi-Fi channel which came from calculating $4 \times (0.4 + 0.2)$. For example, based on Equation (3.4), VWC1 spend about half a minute to monitor all the 11 Wi-Fi channels in 802.11 b/g network.

Although VWC1 had to wait 0.4 seconds on each wireless channel, VWC1 was able to

capture LAP, RAP and RWC info packets sent by PIS in average of ≈ 0.08 seconds with a standard deviation of 0.0001 seconds as shown in Figure 3.20. This is due to the fact that PIS will keep sending multiple packets to the VWC1 every D time intervals which is equal to the switching time of the VWC1. By the time VWC1 switches from one AP to another, info packets should have already been sent by the PIS and on its way to VWC1.

3.4.4 Discussion

Virtual Wireless Clients (VWCs) has been proposed previously to improve wireless performance and privacy [34], however, utilizing VWCs in securing wireless networks is unique. In this chapter, we have presented a comprehensive ETA detection technique. The proposed detection can effectively detect ETA regardless of the gateway type used by the attacker. Both procedures of detecting ETA using single ISP gateway and ETA using different ISP gateways work in parallel using VWC technique.d

Wi-Fi network coverage may be extended by setting up relays such as repeaters or creating wireless distribution system (WDS). This type of wireless coverage extension is avoided by Network administrators due to the lack of standardization [81, 83, 84, 85, 86]. However, our proposed detection can detect whether a specific AP is a relay or an AP by checking the wireless frame headers. In IEEE 802.11, Wi-Fi relay traffic uses all the four address fields in the wireless frame; however, LAP, WC and RAP use only three address fields [87].

Attacker can hide the info packets by setting up a VPN tunnel between the RWC and a VPN proxy server on the Internet. In this case all data traffic between the RWC and the VPN proxy server will be encrypted. VWC1 will be unable to decrypt info packet anymore. However, using VPN proxy will modify the public IP address of VWC1 on the Internet. This behavior will be detected by VWC2.

Another tactics an attacker may undergo on our proposed ETA detection is to exhaust all the available association identifiers AID on each LAP to prevent the VWCs from connecting to it.

Each AP can have up to 2,007 AIDs [88]. Each AID is given to a WC. In this case, the RWC must generate many VWCs and connect to the LAP all at the same. The RWC must maintain all these connections since the LAP timeout and drops idle connection for certain amount of time. To alert the WC of such condition, our proposed ETA detection could count the number of connections to each LAP by monitoring the wireless traffic.

The proposed ETA detection using different ISP gateways is light, fast and effective. However, after detecting the existence of ETA, VWC2 cannot tell which AP is LAP and which AP is RAP. Since both the LAP and the RAP provide Internet access that could have the same specifications, it is very challenging to distinguish them with only client-side actions.

Finally, having PIS server in our detection design is vital. An attacker may initiate a Denial of service attack (DoS) to block all the connection from the wireless clients to the PIS server. To overcome this scenario, multiple PIS servers can be created and installed in different locations. Since the design and implementation of PIS server is simple, no synchronization between the servers is needed. The wireless client randomly select any available PIS server to start our proposed ETA detection technique.

CHAPTER 4: Mobile Data Consumption Attack

4.1 Introduction

Smartphone carrier companies rely on mobile networks for keeping an accurate record of customer data usage for billing purposes. In this section, we present a vulnerability that allows an attacker to force the victim's smartphone to consume data through the cellular network by starting the data download on the victim's cell phone without the victim's knowledge. The attack is based on switching the victim's smartphones from the Wi-Fi network to the cellular network while downloading a large data file.

4.1.1 Preliminaries

Our mobile data consumption attack is designed for use where there is a nearby public Wi-Fi hotspot, such as a coffee shop, hotel, fast food restaurant, or store that has a captive portal.A captive portal is a web page that network users are redirected to accept these network usage conditions or similar terms. They are often used in coffee shops, fast food restaurants, and airports. They can be seen directly after a user connects to the network, as the user will be redirected to the captive portal upon attempting to use the Internet. If the customer doesn't accept the terms and conditions on the captive portal, he or she will be denied Internet access to the free Wi-Fi network.The attacker can target a victim or set of victims at the Wi-Fi network. If the attacker targeted a particular victim, the attacker could wait until the victim enters an area with a nearby public Wi-Fi hotspot.

In the current version of the attack, any customer connected to the open Wi-Fi network is a potential target for our proposed attack. However, not every person will be attacked. This work focuses on attacking one victim, as attacking a set of victims requires running the attack multiple times, once on each victim. Selecting the victim or setting up the attack can happen in

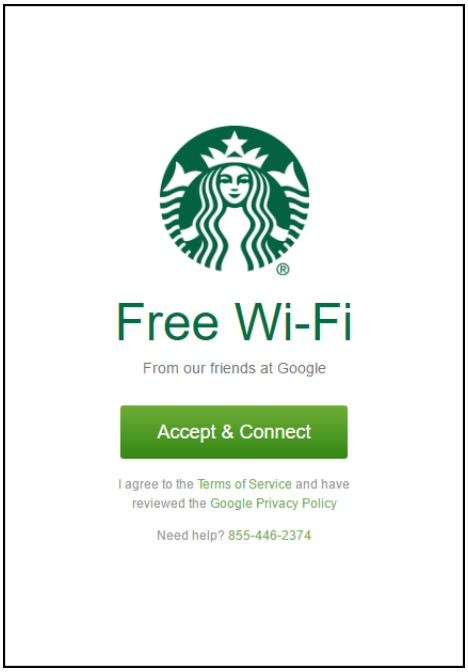


Figure 4.1: A common Starbucks captive portal with an injected malicious script. Once this page opens, a download begins in the victim's smartphone's background.

either order. If the attacker chooses to attack an individual that happens to be at the location, it is recommended that the attacker sets up the attack before choosing a random victim. If the attacker has prior knowledge of a particular victim's plan to go to a certain location, the attacker can set up the attack before the victim arrives. Having the attack setup before the victim arrives or before the victim is chosen increases the chance of the attack's success.

4.1.2 Design

The proposed attack in this chapter is designed based on the following three attacks:

- The attacker creates a fake web server that serves a captive portal web page that is similar to the original Wi-Fi network. The captive portal web page includes a malicious code that forces the victim to download a large data file from the Internet.
- Using an Evil Twin Attack, the attacker lures the victim to switch to the fake network. Such

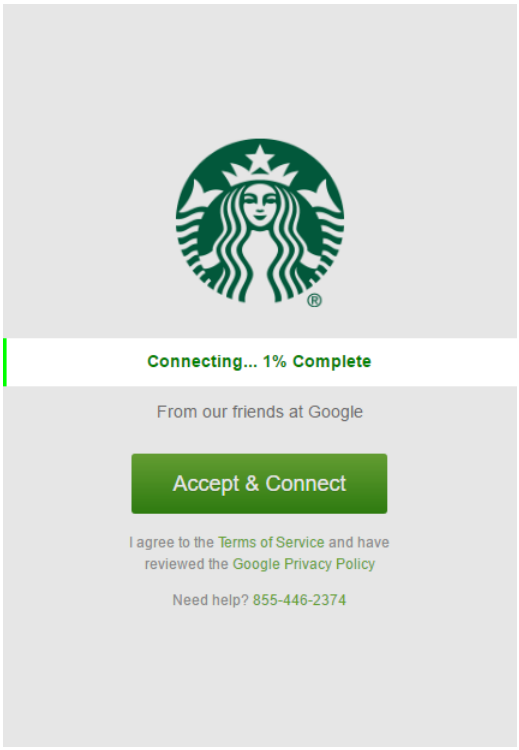


Figure 4.2: The overlay displayed by the malicious script. The loading bar moves to 100% using a logarithmic progression designed to keep the victim on the page so that the attacker can perform the attack.

a switch can also happen in an automatic manner especially when the attacker AP is near to the victim's location. After the victim connects to the fake network, the attacker can spoof the victim's DNS request by sending the malicious captive portal web page whenever the victim requests an URL.

- To make sure the victim is only downloading data through the cellular network, deauthentication attacks target the victim's smartphone preventing him or her from connection to any Wi-Fi network after the captive portal is delivered. Deauthentication is easy to implement because 802.11 WLAN management wireless frames are sent without any protection [88].

4.1.3 Implementation

A laptop with an off-shelf network interface card is used in our attack proposal. Linux operating system was used to implement all the attacks illustrated in the design section. First, the attacker starts an Evil Twin Attack on the victim. We can predict which open Wi-Fi network the victim is connected to, based on his or her current location. For instance, if the victim is in a coffee shop, it is likely that the victim is connected to the coffee shop's public Wi-Fi. The attacker must connect to the open Wi-Fi network beforehand and capture the captive portal used on the network. A captive portal can be captured by connecting to the original Wi-Fi network and accessing the Internet using a free web browser, such as Chrome. The original Wi-Fi network sends the captive portal web page to the attacker which is downloaded and used to create the malicious captive portal.

Not every captive portal that is downloaded will be displayed to the victim exactly the way it appears on the Wi-Fi. Because of this, the attacker may have to manually adjust the captive portal's code in order to make the downloaded captive portal look very similar to the original captive portal. The closer the downloaded captive portal looks to the original, the better the chance is that a victim will not notice that the downloaded captive portal is not the original one.

Once a downloaded copy of the captive portal is obtained, the attacker needs to inject malicious code into the captive portal. The code has been written for this attack, and it can be injected into most captive portals without altering the integrity of the web page. The code does not change how a captive portal looks. However, the new malicious captive portal adds functionality that causes the web page to download data in the background. Functionality is also added that causes the web page to display an overlay upon clicking the button on the captive portal that allows a user to connect to the network.

The overlay is designed to give time for the attacker to conduct the attack while attempting to keep the victim on the captive portal and not realize that the attack is occurring. Also, the captive

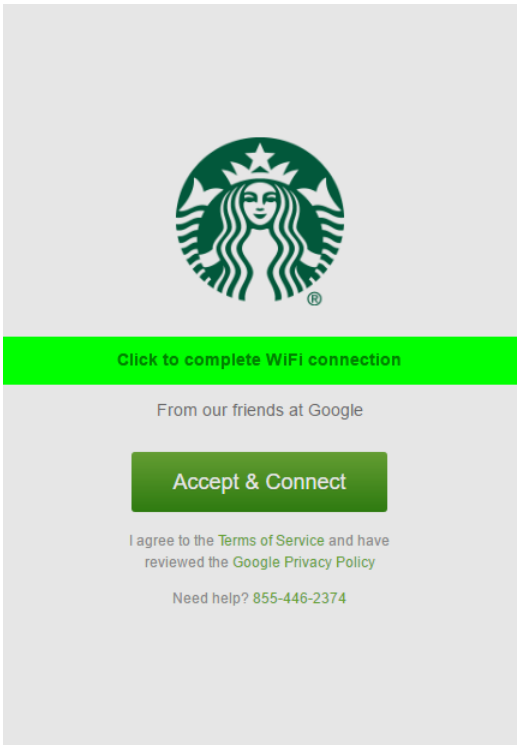


Figure 4.3: The captive portal after the loading bar finishes. When the victim clicks on the page, the victim is redirected to her desired URL, but this page is left open in another tab, downloading in the background.

portal is designed to cause the victim to leave the web page open for an extended period of time by redirecting the victim away from the captive portal, leaving it open and downloading in another tab. The code has also been designed to bypass most pop-up blockers which are built into smartphone browsers. An example of a malicious captive portal is shown in Figure 4.1 along with examples of the overlay shown in Figure 4.2 and Figure 4.3.

4.2 Results

After the malicious captive portal is implemented, the attacker is ready to start our proposed attack. The attacker will begin the mobile data consumption attack by initiating an Evil Twin Attack [17][18] in which the attacker's laptop Wi-Fi impersonates the original access point that

the victim is connected to. The program Airedump-ng can be used to find information about the AP the victim is connected to, such as the AP's MAC address and the AP's channel. The program Airemon-ng [89] can be used to mount the attacker's laptop wireless card into monitor mode to prepare the card to be used as an AP. The program Airebase-ng [89] can then be used to make the card work as an AP with the same MAC address and name of the public AP. The attacker should now have a replica of the public AP running on his or her laptop.

The attacker also needs to start up a DHCP server, DNS proxy, and host the malicious captive portal on his or her laptop. A DHCP server can be set up using the program ISC-DHCP-Server [90]. The server hands out IP addresses to the victim. The network configuration sent by the DHCP server needs to match the one sent by the legitimate Wi-Fi network exactly. A DNS proxy is set up by using the DnsChef software [91]. A DNS proxy is needed to resolve URL requests of the victim to the IP address of the attacker's captive portal by applying a DNS spoofing attack. The Apache web server [92] is used to host the captive portal on the attacker's laptop.

If the victim is already connected to the public Wi-Fi, the attacker can disconnect him or her by initiating a deauthentication attack. This is done by sending continuous deauthentication packets. The program Aireplay-ng [89] is used to send out these packets. The packets are sent to the victim and the AP of the victim. The packets going to the victim are spoofed as the AP and notify the victim that the AP wants him or her to disconnect from the AP. The packets are also sent to the AP informing it that the victim is disconnecting from the AP. Thus the victim and the AP both disconnect from each other. These packets can target a particular individual if the attacker has the MAC address of the victim, or it can disconnect every individual from the AP.

Once the victim is disconnected from the AP, he or she would start searching for another AP from the same WiFi network to connect to. As long as the victim has not previously connected to any other nearby network, and as long as the attacker's AP has a stronger signal than the public AP, the victim will connect to the attacker's fake AP. The attacker can power up their wireless card in order to increase the signal strength of their AP to attract the victim. The attacker can also move

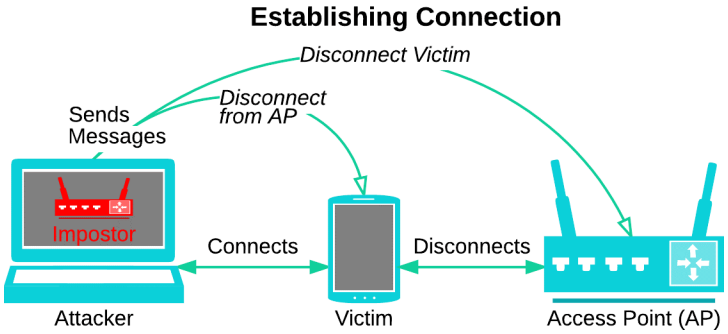


Figure 4.4: A diagram displaying how the attacker establishes connection with the victim.

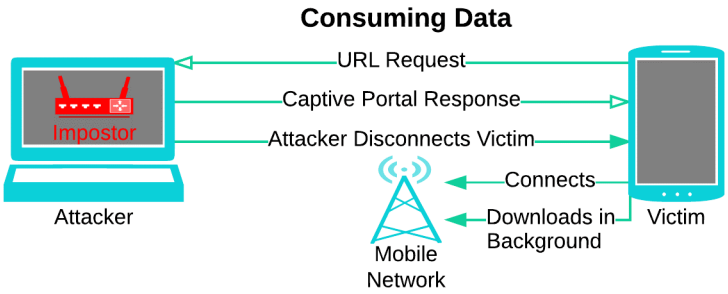


Figure 4.5: A diagram displaying how the attacker begins consuming data from the victim.

closer to the victim in order to increase signal strength. Figure 4.4 illustrates how the attacker establishes a connection with the victim.

Once the victim connects to the attacker's network, the victim will send an URL request to access a particular website. The attacker DNS proxy would capture the URL request and resolve the request to the IP address of the attacker's captive portal. The victim then requests the IP address which happens to be the malicious captive portal of the attacker. The attacker sends back the captive portal as a response to the victim's request. This is a typical MIMA of the wireless victim.

The victim's browser loads the captive portal and immediately starts trying to download data through the attacker. However, the attacker would not have a connection to the internet, and therefore nothing is yet downloaded by the victim. The victim is expected to click the button on

the captive portal that allows a user to connect to the network.

Once the victim has clicked the the Public Wi-Fi Terms and Conditions accept button, the attacker disconnects the victim from the Evil Twin Attack access point. This can be done by simply turning off Airebase-ng. The victim now has no nearby networks to connect to, as the attacker begins sending out deauthentication packets as needed to keep the victim from reconnecting to the public AP. With no nearby network to connect to, the victim will switch by default to the mobile data network. The mobile network grants Internet access to the victim and the captive portal. The captive portal now has a connection to download data using the victim's cell phone data plan. At this point, our proposed attack is consuming data from the victim as shown in Figure 4.5.

A laptop and online security auditing tools are used in implementing our attack. The attack does not require any modification to a protocol or device firmware. The current implementation of the attack redirects users to a malicious captive portal by poisoning DNS requests.

The attack was tested using two smartphones running Android 6. Both phones used the browser Chrome app to open the captive portal. The rate of the mobile data consumption from the attack varies from one device to another. It also depends on the type of data plan the victim is using. For example, if the victim is enrolled in a high-speed cellular data plan, he or she will consume more data than a slow speed data plan. Other variables to the rate of mobile data consumption include how fast the server pushing data out to the device can do so, and how good of a connection the device has to its local mobile tower. However, our tests demonstrated that the rate of mobile data consumption was often high enough to cause a severe amount of data consumption.

Our attack can run on a victim's mobile device for an extended time, which will most likely cause a severe amount of data consumption. The attack exploits a vulnerability in the mobile networks' data usage billing system that allows an attacker to cause serious data depletion of customer data quota. Our tests demonstrate that the proposed attack is feasible when a victim connects to a free open Wi-Fi network offered by coffee shops, fast food restaurants, and airports. The attack will keep going as long as the victim does not stop the cellular data connection.

However, the victim may notice that she is not using the Wi-Fi network in two different ways. First, a pop-up message that says Wi-Fi disconnected may appear at the bottom of the screen for about a second. Second, an indicator at the top of the screen will show that there is no Wi-Fi connection and that mobile data is being used. The attack attempts to cover indicators through social engineering by showing different messages through the malicious captive portal.

Furthermore, our attack can be only implemented when the victim is connected to an open Wi-Fi network. If the customer is connected to a secure Wi-Fi network that uses WEP/WPA/WPA2, our attack will fail. The attacker will not be able to start an Evil Twin Attack because he or she does not have the wireless network encryption key. In addition, the attack may fail when the customer can detect an Evil Twin Attack [17][18].

4.3 CONCLUSIONS

A vulnerability in the mobile networks' data usage billing system was demonstrated by using a mobile data consumption attack. The attack works by delivering a malicious captive portal to the victim, forcing them to connect to their mobile data plan, and causing them to use data via a download initiated by the captive portal. Our attack would work when the victim connects to a free open Wi-Fi network that is available in most coffee shops, fast food restaurants, and airports.

Our attack evaluation was based on attacking the victim for short period of time, using Android mobile OS and Chrome web browser. Further testing is needed to explore the extent of the proposed vulnerability. For example, initiating our attack on different mobile OS and various web browsers.

CHAPTER 5: Parallel Active Dictionary Attack on WPA-II

5.1 Introduction

Wireless protect access II is the state-of-the-art security protocol suite used in WLAN. Unlike open Wi-Fi access discussed in section 3, WPA-II provide secure transmission media between the wireless clients and the access point. WPA-II use different types of authenticate and encryption methods to protect wireless clients data. WPA2-PSK, also called WPA2-Personal, was designed to simplify the implementation of WPA2 in small/network office network. While WPA2-Enterprise is designed to be implement in lager wireless network which may require network administrator to add spacial type of servers to authenticate client before accessing the WLAN. In this section, we present a novel technique to increase the active dictionary attack on both types of WPA-II.

5.2 Parallel Dictionary Attack on WPA2-PSK

WPA-2 PSK provides a simple implementation for the complex design of WPA-2 Enterprise [93]. In this section, we proposed a new attack that targets WPA2-PSK.

5.2.1 Background of WPA2-PSK Protocol

The aim of our techniques is to improve the online dictionary attack speed on WPA2-PSK. The online attack doesn't require a legitimate wireless client to be present. In this section we will explain how a wireless client and a AP generate and exchange the keys used to protect WLANs using WPA2-PSK suite.

5.2.1.1 Keys Generation

The pass-phrase of WPA2-PSK is pre-installed in both of the AP and the wireless client. The pass-phrase is a secret information that will be used to derive all the required keys to protect

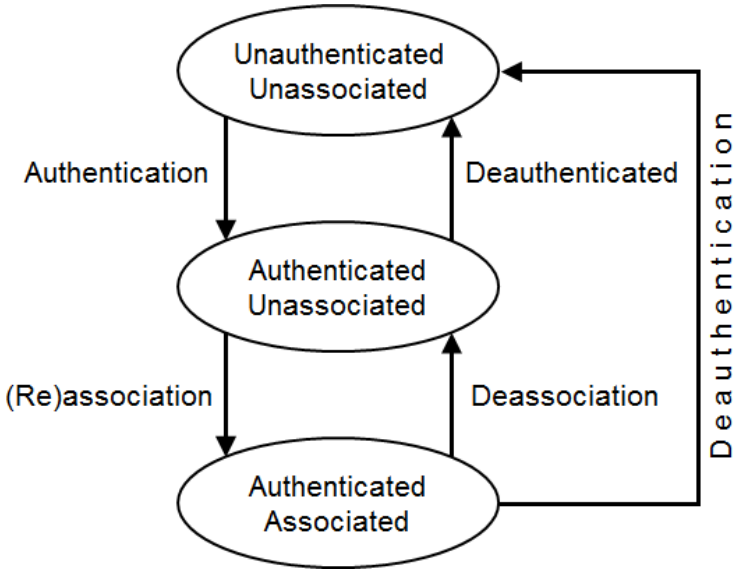


Figure 5.1: 802.11 Authentication and Association states.

WLAN. More than one key will be generated and each one of them is used for different purposes. In general, there are seven keys involved in the protection of WPA2-PSK networks[94][95].

First, before WPA2-PSK key generation starts, an 802.11 wireless client has to authenticate and associate to the AP as shown in Figure 5.1[96]. WPA2-PSK four-way handshaking procedure starts when the wireless client passes the authentication and the association states. The names of these two states are somewhat misleading since both states do not have any type of security. It is merely a formality procedure used by wireless clients and an AP to exchange capability information.

Second, after the wireless client is authenticated and associated to the AP, WPA2-PSK four-way handshake start. WPA2-PSK uses a Pre-shared key (PSK) which is derived form the passphrase that was entered manually to both wireless client and AP. The passphrase length is 8 to 63 characters. Using Password-Based Key Derivation Function 2 (PBKDF2), passphrase, SSID and SSID length are to be hashed 4096 times to produce 256 bit Pair Master Key (PMK) as shown in Figure 5.2. PMK is the same for every pair of SSID and passphrase.

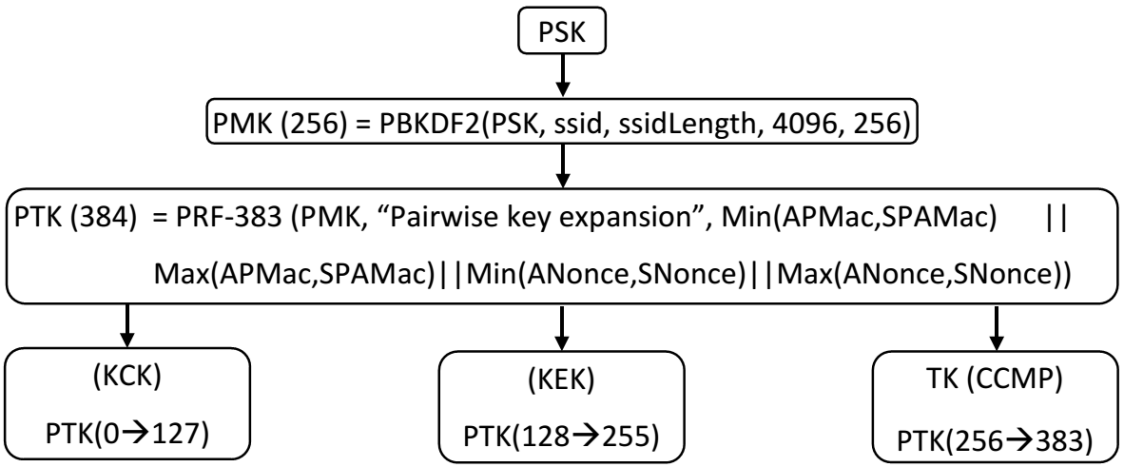


Figure 5.2: WPA2-PSK key generation.

Third, PMK, the phrase ”Pairwise key expansion”, AP’s MAC address and the wireless client’s MAC address, a random number generated by the AP (ANonce) and a random number generated by the wireless client (SNonce) will be fed to Pseudo-random function (PRF) to produce Pair Temporary Key (PTK). The length of the PTK in the WPA2-PSK(AES/CCMP) is 384 bits.[94].

Fourth, PTK will be divided into three keys as shown Figure 5.2 where :

- Key Confirmation Key (KCK 128 bits) which is used to provide data integrity in the four-way handshaking communication.
- Key Encryption Key (KEK 128 bits) which is used to protect the four-way handshaking communication.
- Temporal Key (TK 128 bits) is used to protect wireless data.

All the previous keys are used to ensure the integrity and confidentially and used in unicast communication between the AP and the wireless client. On the other hand, the AP will generate a Group Temporal Key (GTK) and send it to the wireless client. GTK is used by wireless clients

and AP to send broadcast data to the wireless network. The AP uses KEK to protect GTK while sending it to the wireless client.

5.2.1.2 Keys Exchange

In WPA2-PSK, the AP starts the four-way handshaking messages exchange by sending Message 1. Both the AP and the wireless client relays on the four-way handshake communication to confirm the possession of PSK. Four-way handshake procedure starts after the wireless client authenticates and associates (Figure 5.1) to the AP. Four-way handshake consists of four messages as shown in Figure 5.3[59].Extensible Authentication Protocol (EAP) over LAN (EAPoL) is used to carry out the four-way handshaking messages between both parties. First, AP sends Message 1 which contains an ANouse using EAPOL. ANouse is a 32 digit random number generated by the AP. When the wireless client receives Message 1, she will have all the required parameters to derive PMK from PSK as shown in Figure 5.2. At this point, KCK, KEK, and TPK are generated on the wireless client side. The wireless client then creates Message 2 which contains SNonce and the Message Integrity Code (MIC). Where SNonce is also a 32 digit random number which is generated by the wireless client.

MIC is used to ensure the integrity of Message 2. MIC is calculated on the whole EAPOL header plus the KCK (MIC(EAPOL, KCK)). When AP receives Message 2, it extracts SNonce and derives KCK, KEK, and TPK. Furthermore, the AP will calculate Message 2 MIC and compare it with the MIC received from the wireless client.

Message 3 is sent from AP to the wireless client, and it contains the GTK encrypted using KEK and MIC. Message 4 will be sent from the wireless client to the AP to confirm a successful end of the four-way handshaking. When the attacker receives Message 3 from the AP, she can confirm that the passphrase used in the creation of Message 2 was correct.

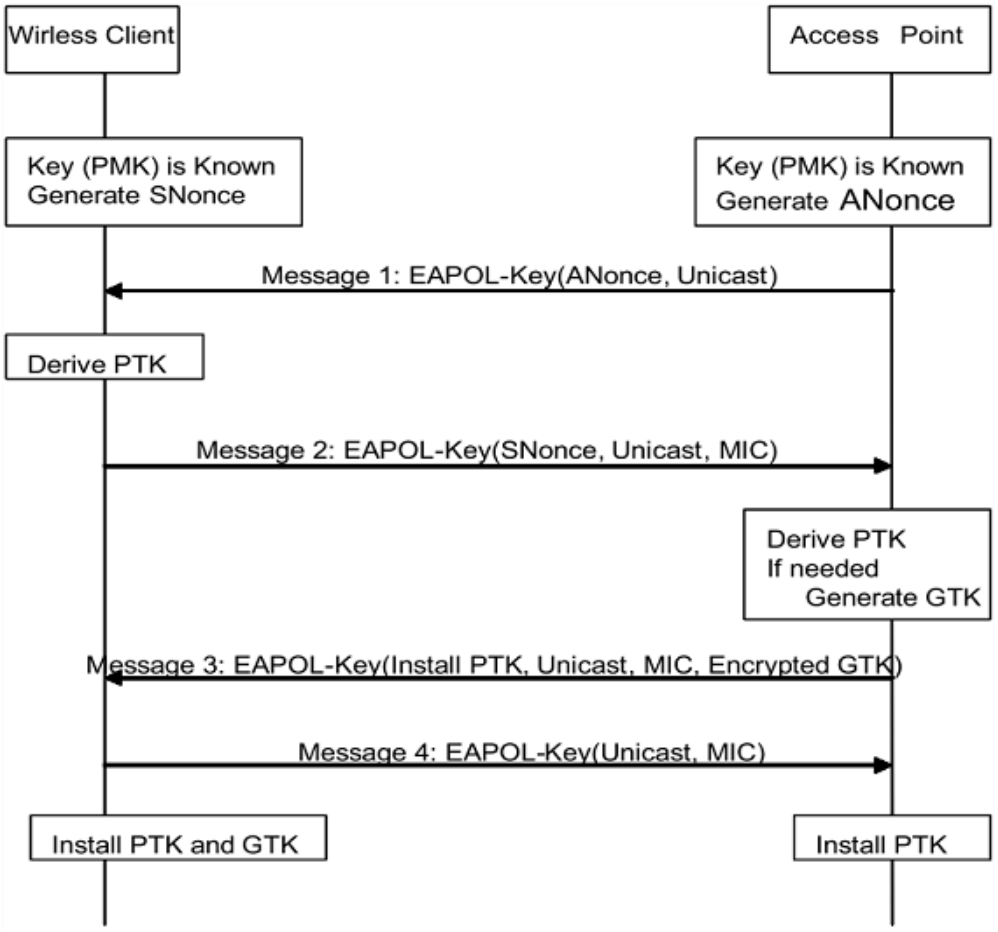


Figure 5.3: WPA2-PSK Four-Way Handshaking.

5.2.2 Active dictionary attack

Active dictionary attack on the passphrase of the WPA2-PSK can be applied since most APs do not limit the number of trials a wireless client can input using an incorrect passphrase. In this section, we present two novel techniques to speed up the active dictionary attack. The following two subsections illustrate the design and the implementation of proposed methods.

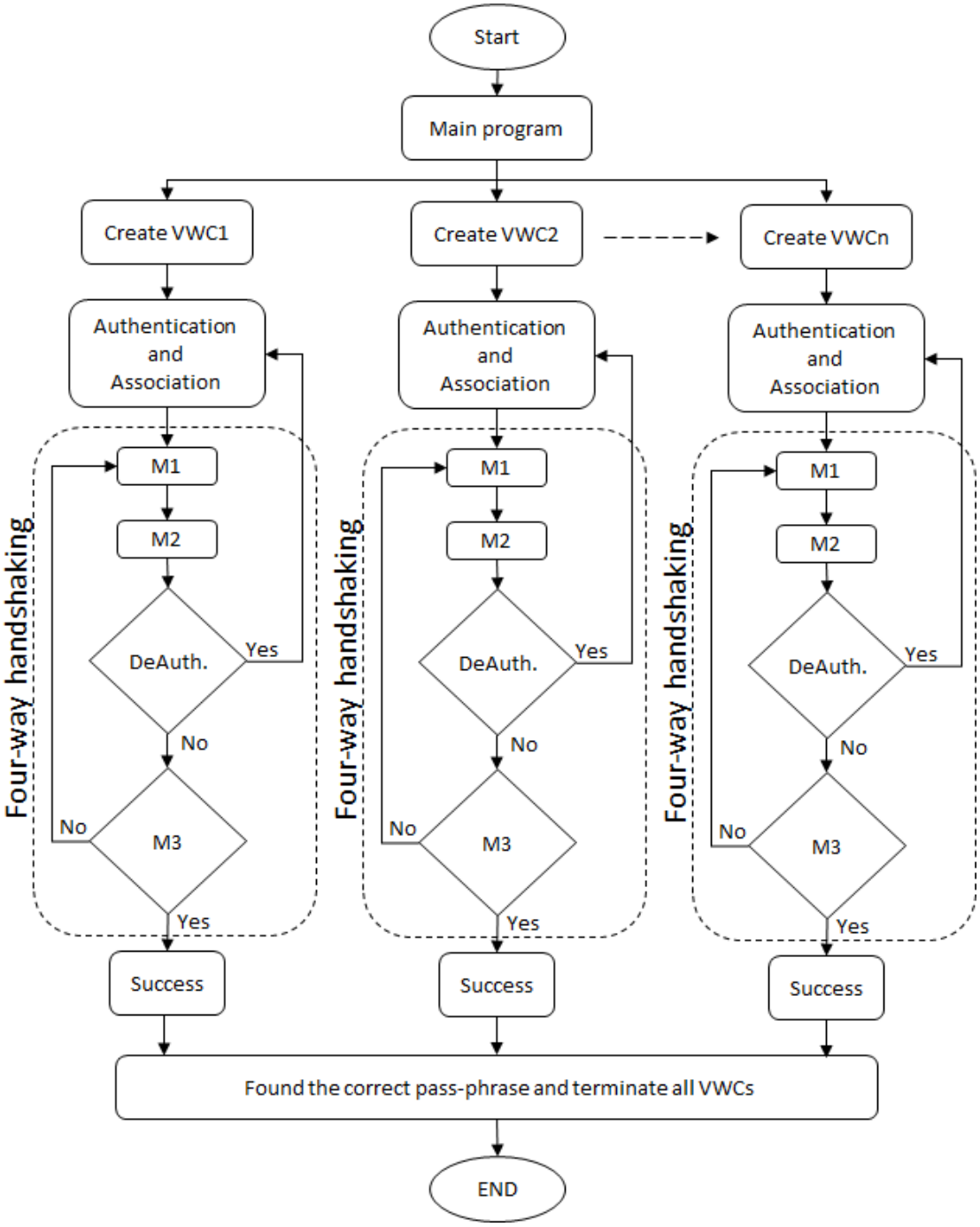


Figure 5.4: Our proposed parallel active WPA2-PSK attack design. Where M1, M2 and M3 are the first three messages of the four-way handshaking. M4 message was omitted since it is only a confirmation frame from a VWC to the AP to indicate a successful end of the four-way handshaking procedure.

5.2.2.1 Proposed design

WPA2 was designed to provide security to WLAN. WPA2-PSK is designated for small office/home office networks and to be used without the need of a RADIUS server. The strength of WPA2-PSK security depends on how complicated the passphrase is. In this section, we introduce a new proposed design that utilizes two novel techniques to speed up online pass-phrase guessing speed.

The proposed design is based on applying an active dictionary attack against WPA2-PSK. The attack aims to recover the passphrase without the need of capturing the four-way handshaking between a legitimate wireless client and the AP.

Our software tries to automatically guess the passphrase by selecting a passphrase from a dictionary word list and creating Message 2 of the four-way handshaking. The program then sends Message 2 to the AP and waits for a reply. If the AP responds with Message 3 then, we have guessed the correct passphrase. When the AP replies with Message 1 to our Message 2 then, the passphrase used to create Message 2 was incorrect.

The major hurdle of the active dictionary attack is the passphrase guessing speed. Some APs will take a certain amount of time to reply to Message 2 of the four-way handshake, especially when the passphrase used to build Message 2 was wrong. Also, our program on the attacker machine will take some time to filter responses received from the AP since the attacker will receive all the Wi-Fi frames transmitted on that channel. Furthermore, transmission propagation will add more delay time to pass-phrase guessing speed.

To speed up the WPA2-PSK passphrase guessing process, the first novel technique we present in our active dictionary attack is to let the attacking program initiate multiple virtual wireless clients (VWCs). Each VWC acts as a real client trying to connect to the AP. All these VWCs are generated from one wireless interface card. A VWC will use a spoofed MAC address when communicating with the AP.

To further speed up the PSK guessing process, the second novel technique we present in our active dictionary attack is to enable each VWC to try more than one passphrase for every wireless session. This method speeds up the attack since the VWC will not have to pass 802.11 authentication and association states every time a new passphrase is to be tested. A single VWC will keep trying different passphrases until it is de-authenticated from the AP as shown in Figure-5.4.

5.2.2.2 Implementation

Our technique was implemented using C language on a Linux machine. Using LORCON2[77] library, we were able to inject and receive 802.11 wireless frames. LORCON2 is a cross-platform virtual interface that allows us to send and receive crafted 802.11 frames.

Our main program creates multiple processes were each process acts as a standalone wireless client. Each VWC picks a random spoofed MAC address and start a wireless session to the AP. The main program keep monitoring the state of each process.

After a VWC passes the authentication and association stages of the 802.11 WLANs, the VWC begin the four-way handshake to the AP. Using a dictionary word list, the VWC creates Message 2 and send it to the AP. If the AP responds with Message 3 then the passphrase was correct otherwise the VWC will try another passphrase from the dictionary word list.

When the AP receives an incorrect passphrase, it will respond with Message 1. The VWC will disconnect from the AP and start a new wireless session to the AP with a different MAC address. After that, the VWC can inject another passphrase to the AP.

To further speed up the attack , we noticed that since the AP didn't send any de-authentication frames due to the incorrect passphrase in Message 2, we can inject another passphrase using Message 2. This will speed up the attack even more since the VWC doesn't have to send authentication and association frames again. The program will keep trying passphrases until the AP sends a de-authentication frame with reason code 02 (previous authentication no longer valid). At this point,

the VWC will stop the current wireless session and start a new wireless session with a different MAC address.

5.2.3 Evaluation

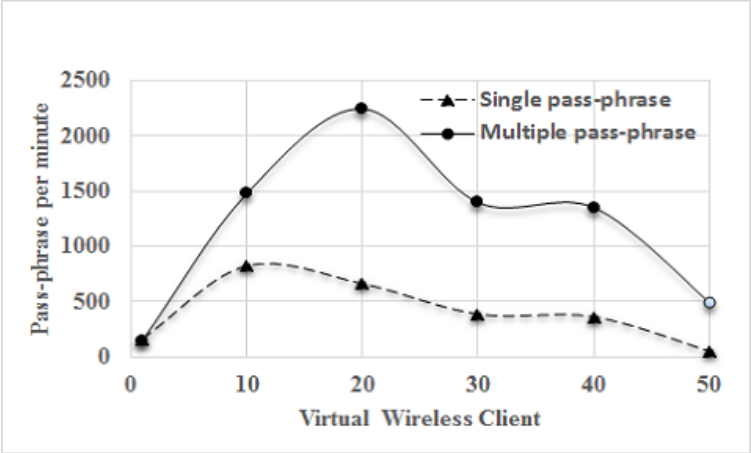
We evaluated our proposed technique by initiating the attack on three different wireless routers. The wireless routers used in the test bed were DLink 601, Cisco Linksys EA3500 and Xiaomi Router Mini. Each wireless router was restored to its default setting, then we enabled the WPA2-PSK protection in each router with a certain passphrase. The attacking machine has an Atheros chipset WLAN card and was installed with Linux based OS. The APs and the attacker's WLAN card used 802.11g as wireless communication standard.

During the attack, our prototype program test our two techniques at the same time. For each AP, the first technique starts by creating multiple VWCs where each one of them try only one pass-phrase at a time and wait for the response from the AP. After the client sends Message 2 of the four-way handshaking to the AP, if the AP replied with Message 3 then the passphrase was correct. However, if the AP replied with Message 1 then the pass-phrase was wrong. The VWC will be de-authenticated from the AP and change its MAC address and start a new wireless session.

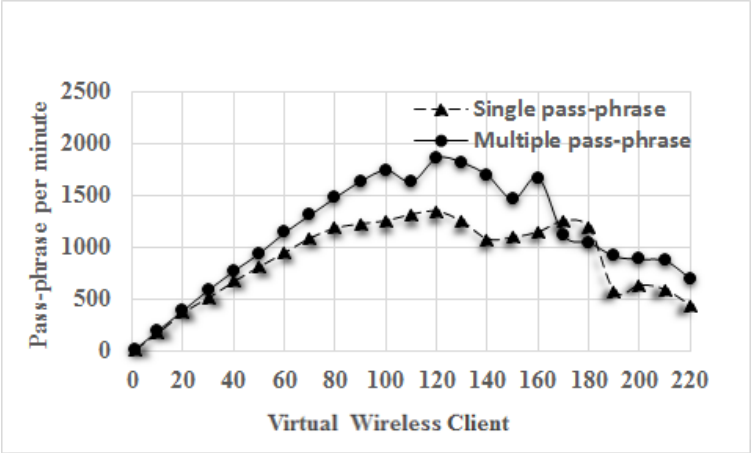
The second technique also create multiple VWCs. However, when one VWC receives Message 1 as response to Message 2 (guessed passphrase is incorrect), it will not proceed with de-authentication. Instead, the VWC will pick another passphrase and create Message 2 and send it to the AP again. The VWC keep sending Message 2 repeatedly until it receives de-authentication frame from the AP. At this point the VWC will change its MAC address and start a new wireless session.

To measure how many passphrases we can test at the same time using both techniques, for each trial,the program increases the number of VWCs from 1 to a certain number. During the test, each AP responded differently to our attack as shown in Figure 5.5.

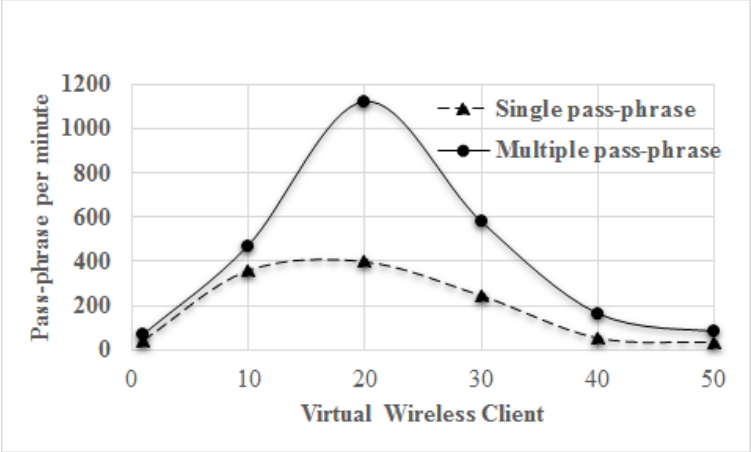
For the three APs, the attack speed of the traditional online dictionary attack (one wireless



(a)



(b)



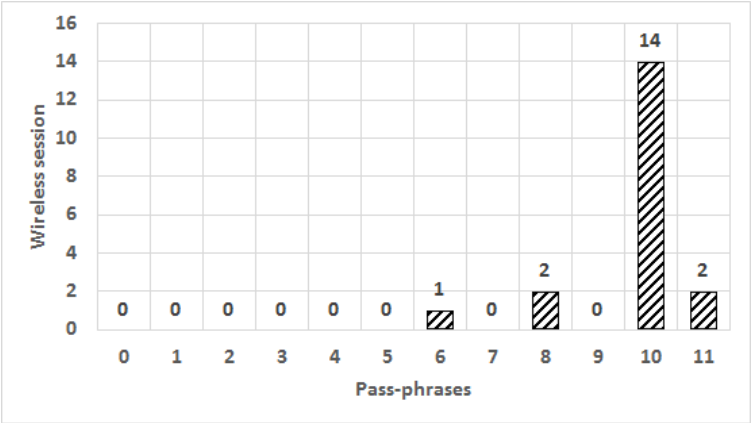
(c)

Figure 5.5: Comparison between three different wireless routers against our proposed attack where (a) Cisco Linksys EA3500, (b) Dlink DIR-601 (c) Xiaomi Router Mini.

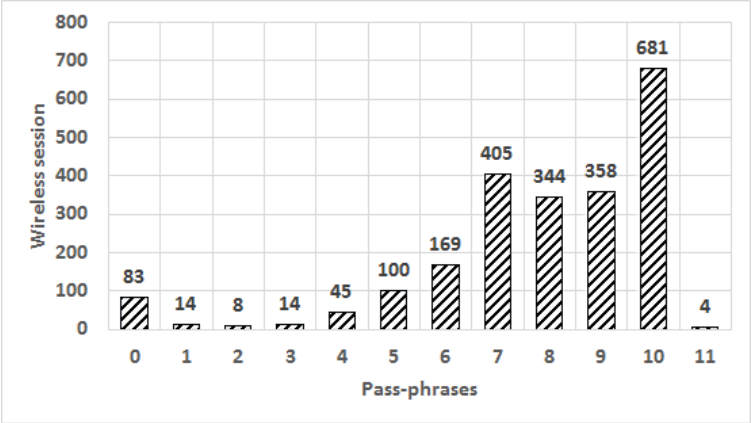
client and single passphrase per wireless session) is shown as the first data point in each graph of Figure 5.5. For example, the traditional attack speed for Dlink wireless router, as shown in the first data point on Figure 5.5b, is 18 passphrases per minute. Increasing the number of VWC increased the intensity of the active dictionary attack. When each VWC tests more than one passphrase per wireless session, the attack effectiveness also increased as shown in Figure 5.5-5.6.

When a single VWC tries multiple pass-phrase guessing at the same wireless session against Dlink wireless router, the attack intensity was on average 18 pass-phrase per minute as shown in Figure 5.5b-5.6a. In Figure 5.6, the average passphrase guessing speed can be calculated by dividing the total number of passphrases by 10 minutes. Increasing the number of VWC to 120 gave us the maximum pass-phrase attack guessing for the Dlink wireless router—on average 1833 passphrase per minute as shown in Figure 5.5b-5.6b. The pass-phrase guessing attack speed improvement for the Dlink wireless router at this point is about 100-fold. However, further increasing the number of VWC more than 120 had negative impact on the pass-phrase guessing attack speed. As shown in Figure 5.5b-5.6c, when we have more than 120 VWC attacking Dlink wireless router, the pass-phrase guessing speed drops.

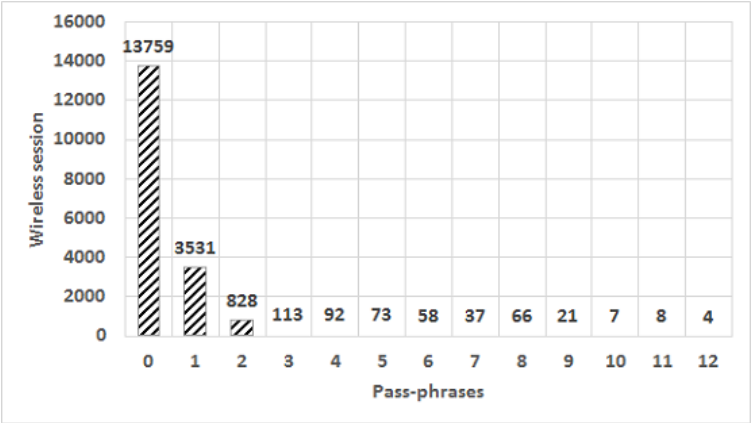
Both Figures 5.5 and Figure 5.6 show that the number of passphrase guessing will drop when the number of VWCs passes a certain threshold. This is because increasing the number of VWCs for each AP will increase the traffic on the wireless channel. Delay time and frame loss will increase when the wireless channel becomes congested up to a certain point that many wireless sessions will time out. To prove that, Figure 5.7 shows a comparison between attacking Dlink wireless router with 120 VWC before and after the wireless channel being relatively busy. We say relatively busy because 802.11g wireless channel during our test may get busy since it is a shared medium by other wireless clients. However, in Figure 5.7 we applied a continuous wireless data transmitted from another wireless client during the full length of the attack to simulate a busy channel. The pass-phrase guessing speed when we have 120 VWC attacking at the same time dropped from 1833 pass-phrases per minute (Figure 5.5b-5.6b) when the channel is relatively idle



(a)



(b)



(c)

Figure 5.6: Pass-phrases guessing trails per each wireless session against Dlink wireless router where (a) One VWC, (b) 120 VWC and (c) 220 VWC.

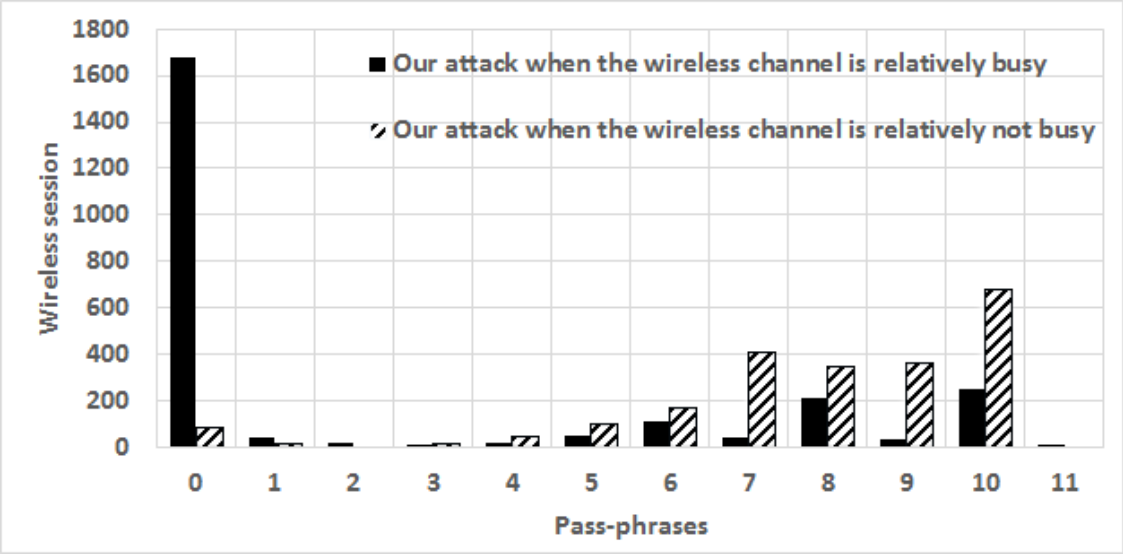


Figure 5.7: Comparison between pass-phrases guessing trails per each wireless session when we have congested vs uncongested wireless channel using the same number of VWCs (120) againt Dlink wireless router.

to 247 pass-phrases per minute when the channel is relatively busy.

5.2.4 Discussion

In this section, we presented an online active dictionary attack to tackle the current Wi-Fi home security (WPA2-PSK). Available dictionary attacks are based only on the offline scheme, and they may not work if an attacker was unable to capture the four-way handshaking frames of a legitimate client. Furthermore, we proposed two novel techniques to speed up the online dictionary attack.

Our proposed attack is based on the following assumptions. First, by default, the AP does not filter the wireless client MAC addresses. Second, WPA2-PSK does not limit the number of trials a wireless client can take to enter the pass-phrase.

Furthermore, WLAN administrators may install more than one AP to expand the wireless coverage signal[37]. Since all APs belongs to the same Extended Service Set Identification (ES-

SID), our attack can be distributed to all APs. In this scenario, the attack speed will further increase with the increase in the number of APs in the ESSID.

Our proposed attack has its own limitation. We discuss them below:

First, Our proposed attack will be limited by the wireless channel bandwidth and the response time of the AP. However, nowadays, the new 802.11ac standard provides high bandwidth wireless channels that can reach up to 1 Gbps [46] compared to 54Mbps for the 802.11g. In addition, more powerful SOHO APs are being developed that have more processing power which will reduce the response time of the AP.

Second, offline dictionary attack is faster than online dictionary attack since the offline attack is not limited by AP and the wireless channel bandwidth. However, offline dictionary attack will fail if the attacker is unable to capture the four-way handshaking between a legitimate wireless client and the AP. In this scenario, our technique will be a feasible solution to recover the WPA2-PSK pass-phrase.

5.3 Parallel Dictionary Attack on WPA2-Enterprise

WPA2-Enterprise is the current security suite used in protecting large WLAN. It provide more authentication methods than WPA2-PSK. In this section, we will expand our parallel active dictionary attack on WPA2-PSK to target WPA2-Enterprise.

5.3.1 Background of 802.1x Protocol

IEEE 802.11i standard was developed to overcome the vulnerabilities found in WEP. IEEE 802.1x standard and 4-way handshaking procedure are the main components of IEEE 802.11i (WPA-II enterprise) standard. IEEE 802.1x standard is mainly used for authenticating the WC, and the 4-way handshaking procedure is used for exchanging cryptography keys[58]. In this section, we present a novel technique to attack the authentication part of IEEE 802.11i standard.

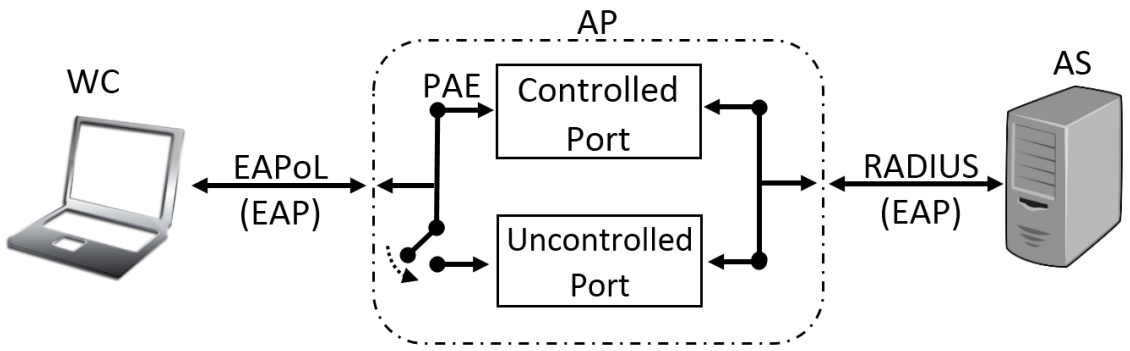


Figure 5.8: 802.11i port access entry authentication.

When the WC (supplicant) authenticates to the AS (RADIUS), the communication will pass through the AP (authenticator) as shown in Figure 5.8. IEEE 802.1x standard uses port access entry (PAE) on the AP to allow the WC to send/receive frames to the AS. During the authentication phase, all data traffic from the WC will be forwarded only to the AS. After the WC finishes the authentication phase successfully, she switches from the controlled port to the uncontrolled port in which they can access services offered by the wired network.

One of the most popular authentication methods used by RADIUS is EAP-MD5. Since EAP-MD5 is based only on Message Digest 5 hashing function, it is considered fast and straightforward to implement [70] [97]. EAP-MD5 authentication starts after the WC finishes 802.11 authentication and association states with the AP as shown in Figure 5.9. The names of 802.11 authentications and associations are somewhat misleading since both communications don't have any security. It is merely a formality procedure used by WCs and an AP to exchange capability information.

EAP-MD5 begins when the AP sends EAP-Request (Identity) frame to the WC. Also, the WC can ask for EAP-Request (Identity) frame by sending EAPoL Start frame. At this point, the WC sends his/her username to the AP. The username is passed to the AS server using RADIUS protocol. The AS generates a random challenge string and an ID, which represents a small number, and sends it to the AP. After receiving the random challenge and the ID from the AP, the WC hashes

(ID + Password + MD5 Challenge) using MD5 hashing function and sends it to the AP. The AS successfully accepts the access request when the password used in the hashing function matches the one stored in the AS; otherwise, the AS rejects the access request. Also, the WC can send EAPoL Logout frame to de-authenticate from the AP.

Although EAP-MD5 is attractive and simple, it is considered vulnerable to be used in the WLAN for many reasons [70][97]. For example, the attacker can apply replay attack by capturing the hash message from the WC and send it to the AP. Furthermore, the attacker can sniff the hashed message and use an offline dictionary attack. The WC can reject EAP-MD5 authentication method by responding to the MD5 challenge by Nak frame [17].

The EAP-(TLS and TTLS) and PEAP provide better protection when compared to the EPA-MD5 in the WLAN. The EAP-TLS is considered the most secure method in WLAN [27][70]. Both, the WC and the AS, should have their digital certificate. EAP-TLS perform authentication by exchanging the digit certificate of the WC and the AS. The complexity added by requiring the WC to have a digital certificate makes EAP-TTLS and PEAP a better alternative.

EAP-TTLS and PEAP are the most common authentication methods in 802.11i [27]. They both use two phases of authentication. The first authentication phase provides a secure channel so that the WC can pass her credentials using the second authentication phase. The first authentication phase also can be referred to as the outer authentication, and the second authentication phase is called the inner authentication. The inner authentication can use a less secure EAP authentication method, such as EAP-MD5 since the outer phase protects it. Table 5.1 compares between the different types of EAP authentication methods [70].

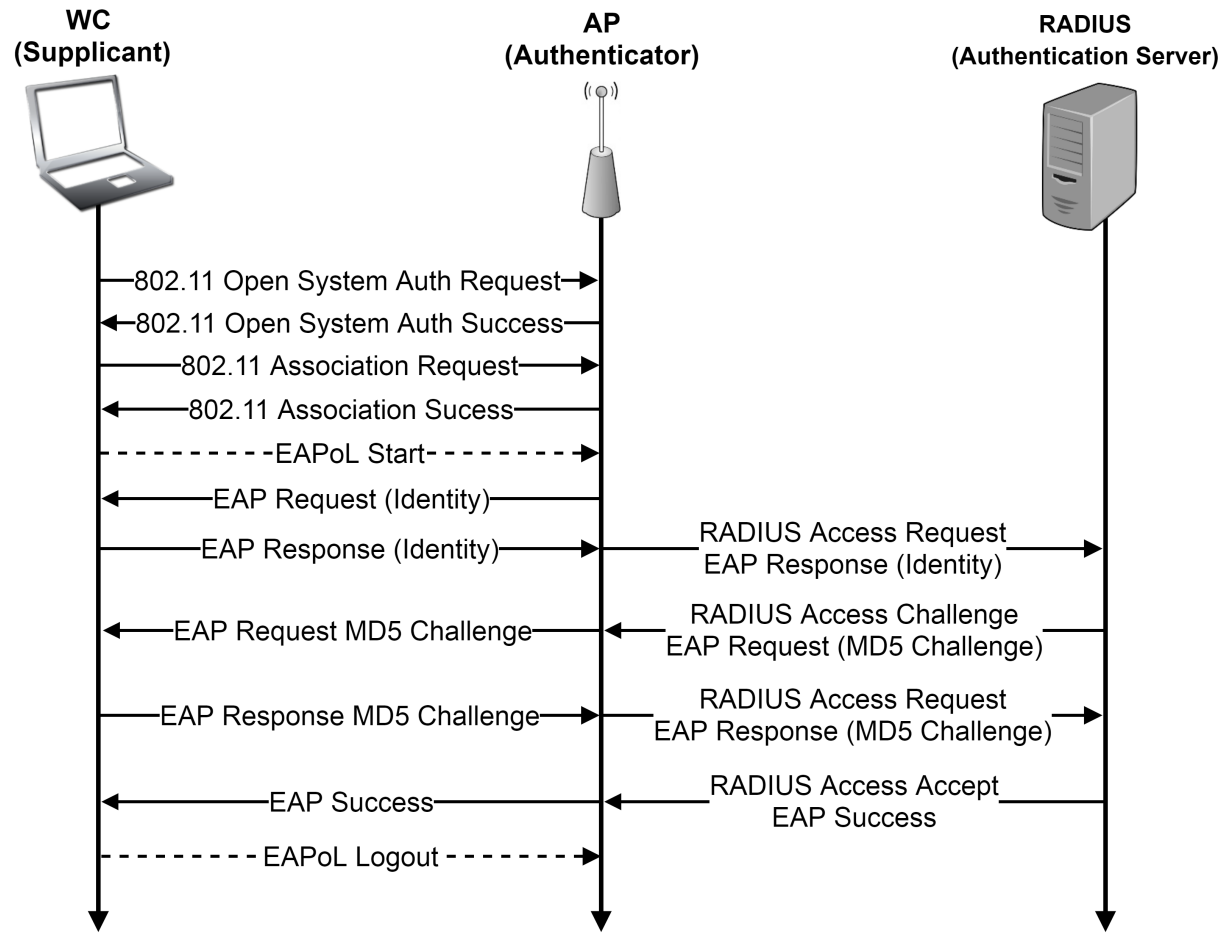


Figure 5.9: EAP-MD5 authentication method.

5.3.2 Active Dictionary Attack Design

5.3.2.1 Design

Most EAP authentication methods require each WC to provide her username and password to be allowed to access the WLAN. The username is used to locate the WC account and the password to authenticate her. To obtain both the username and password, our active dictionary attack was divided into two main steps.

The first step of our attack procedure is to capture the WC username. This goal accom-

Table 5.1: Comparison between common EAP authentication methods

Property	*EAP Authentication Method*			
	MD5	*TLS*	*TTLS*	*PEAP*
Authentication attributes	Unilateral	Mutual	Mutual	Mutual
Deployment difficulties	Easy	Hard	Moderate	Moderate
Dynamic re-keying	No	Yes	Yes	Yes
Requires server certificate	No	Yes	Yes	Yes
Requires client certificate	No	Yes	No	No
Tunnelled	No	No	Yes	Yes
WPA compatible	No	Yes	Yes	Yes
WLAN security	Poor	Strongest	Strong	Strong

plished by monitoring the authentication communication between a legitimate WC (LWC) and the LAP. The LWC is required to send her Identity when she receives EAP-Request (Identity) from the AP at the beginning of the EAPoL protocol, as shown in Figure 5.9. To simplify the implementation/management of the WLAN, most network administrators use the LWC username as her Identity [98]. Furthermore, most EAP authentication methods send LWC Identity in a plain text [70].

After capturing the LWC username, we start the second step of our proposed procedure by initiating parallel active dictionary attack on the AS. Using only one wireless interface card, we created multiple VWCs. Each VWC communicates with the AS as a standalone WC and starts a dictionary attack on the password of the captured LWC username. To speed up the attack speed, VWCs use the least time-consuming EAP authentication method such as EAP-MD5 when communicating with the AS. EAP-MD5 is considered to be faster compared to both EAP-TTLS and PEAP because it interacts less with the AS. A VWC can reject other EAP authentication methods offered by the AS by sending a NAK frame at the beginning of the authentication process. This will enforce the AS to use EAP-MD5 for the communication.

By using the fastest available EAP authentication method, each VWC starts authenticating

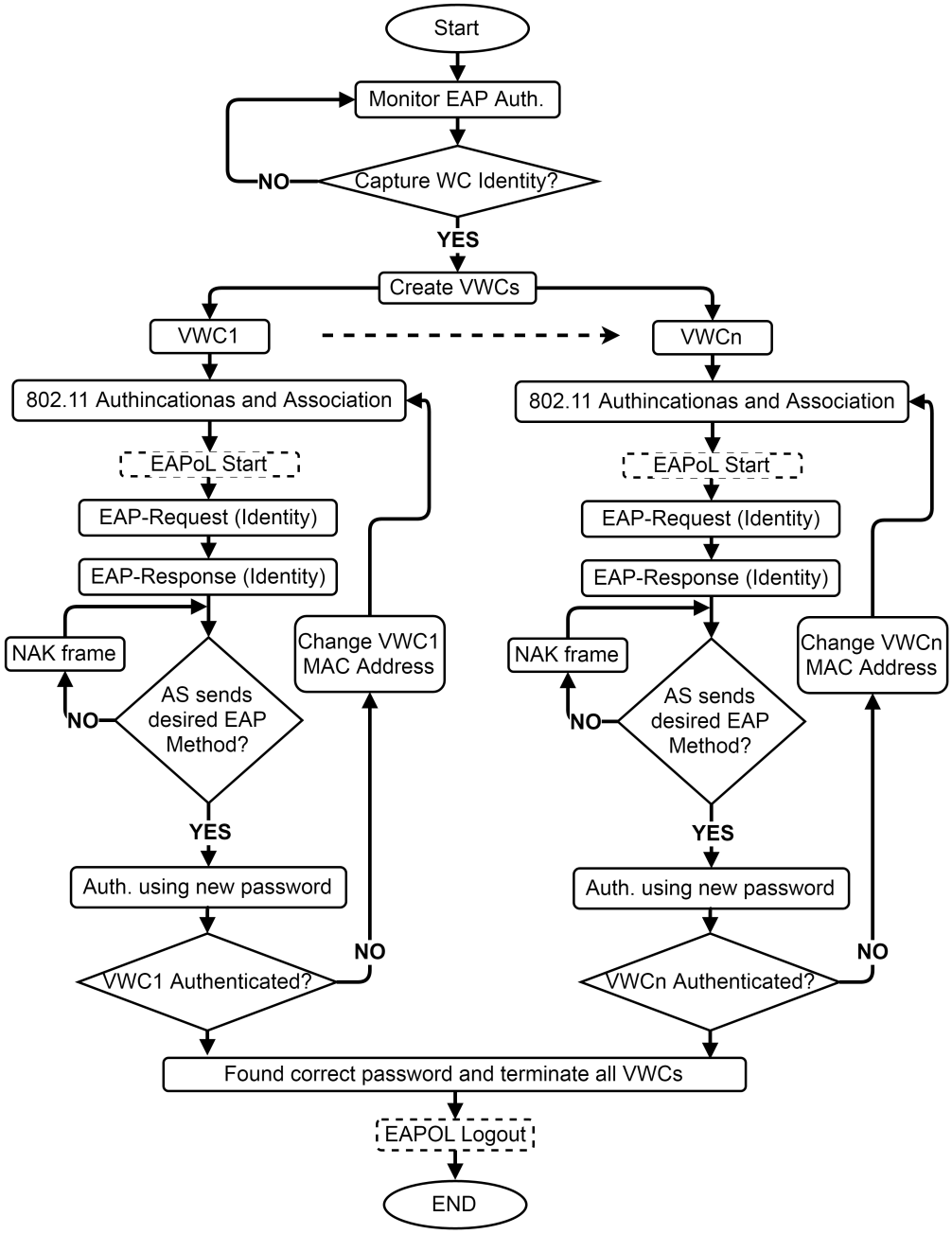


Figure 5.10: Our proposed parallel active dictionary attack using one wireless interface card (WIC)

to the AS using different passwords. When a VWC fails to authenticate using the selected password, it changes the MAC address and starts a new EAP authentication session. The attack stopped when one of the VWCs authenticated successfully to the AS as shown in Figure 5.10.

5.3.2.2 Implementation

Our proposed parallel active dictionary attack is implemented using C language. We used Loss Of Radio CONnectivity (LORCON) 2 library to create multiple VWCs. LORCON 2 is an open source library used to inject/receive 802.11 wireless frames [77].

Each VWC emulates a single WC with a unique MAC address. All VWCs send/receive frames using only one wireless interface card (WIC) at the same time. Whenever one of the VWCs passes the authentication phase, the attack stops.

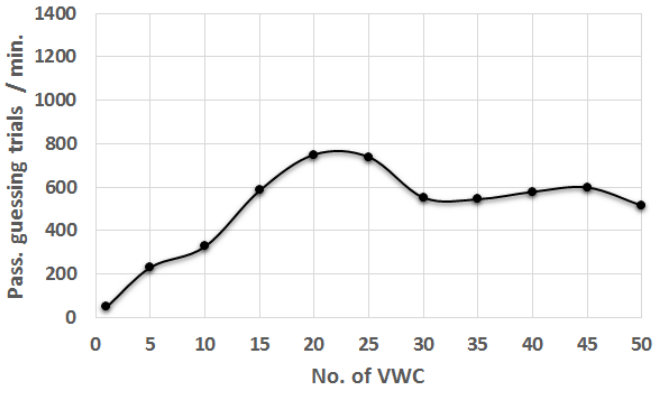
5.3.3 Evaluation

We set up a WLAN testbed to evaluate our proposed parallel active dictionary attack. The testbed consisted of an AP and an AS. Three different types of wireless routers (WR) (ASUS-RT-AC68U, Dlink-DIR890L, and Linksys WRT54) were used in the evaluation as an APs. Furthermore, we implemented the AS by installing on a server the current version of FreeRADIUS server, which is the most popular open source RADIUS server [42][43].

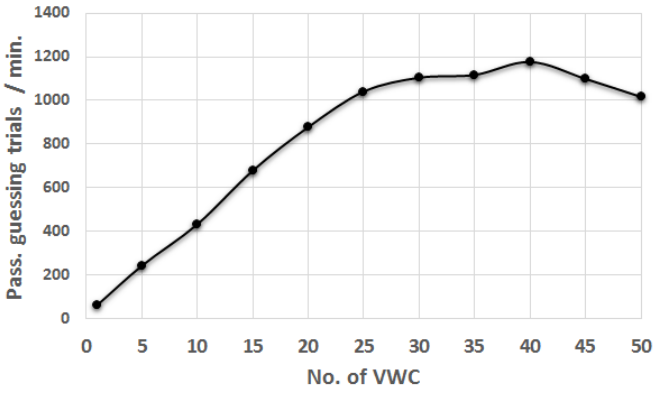
The AP was configured to use WPA-II enterprise as the WLAN security protocol. The AS used RADIUS protocol on port 1812 to communicate with the AP. For the RADIUS server configuration, we added the AP as a client and the LWC as a user, which is the typical FreeRADIUS set up [43]. All other settings in both the AP and the RADIUS server set to default.

On the attacker side, our proposed parallel active dictionary attack code was installed on Linux based OS. The attacker used Penguin Wireless N Dual-Band USB Adapter as the WIC. We used Wireshark to monitor the traffic between the LWC, VWCs, the AP, and the AS.

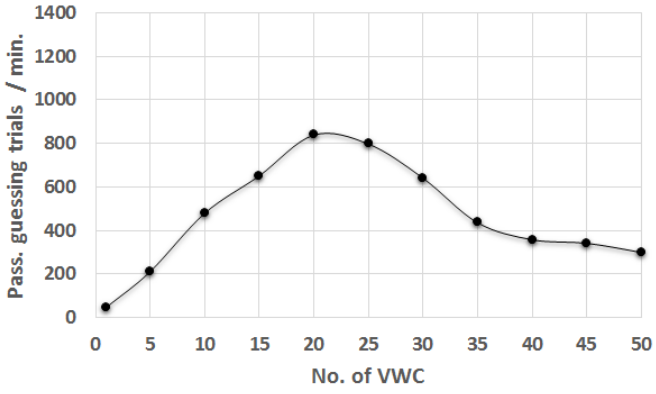
The first step of our proposed attack is to capture the username of the LWC. From the LWC PC, we connected to the testbed WLAN using the most common EAP authentication methods (TTLS and PEAP). Our proposed attack program successfully captured the LWC username each time the LWC sent her Identity to the AP. We also observed that the AS requested the LWC to use



(a)



(b)



(c)

Figure 5.11: Comparison between three different APs against our proposed attack where (a) Dlink-DIR890L, (b) ASUS-RT-AC68U (c) Linksys WRT54. The traditional active dictionary attack intensity is represented by the first data point on each figure.

EAP-MD5 as the initial EAP authentication method. However, the LWC rejected using EAP-MD5 by sending a NAK frame and accepted one of the two other authentication methods (TTLS and PEAP).

The second step of our proposed attack started after capturing the LWC username. First, our proposed attack code created many VWCs that connected to the testbed AP and started EAPoL. The attack code used the LWC username in all Identity response frames when communicating with the AP. Unlike the LWC, the proposed attack code accepted EAP-MD5 authentication method requested by the AS. EAP-MD5 is simple to implement and requires less time to finish the authentication process compared to both PEAP and EAP-TTLS.

To illustrate the increase in the dictionary attack speed using our proposed technique, we started authenticating to the AS using only one VWC. This resembles the traditional single WC active dictionary attack. Then, we increased the number of VWCs connecting to the AP until the password guessing rate started to drop. Each AP tested for a total time of one hour and a half. We repeated the previous procedure for the three different types of APs used in the testbed, and the results shown in Figure 5.11.

The increase in the intensity of guessing trials for the three different APs reached its maximum when there was a certain number of VWCs authenticating at the same time. That number was different from one AP to another. For example, the rate of guessing trials in ASUS-RT-AC68U, AP when we had only one WC was 65 passwords per minute. The password guessing rate increased to 1176 passwords per minute when we had 40 VWCs. Such an increase in the intensity of guessing speed is equal to 1700% as shown in Figure 5.11b. However, increasing the number of VWCs beyond that point (40 VWCs) will reduce the password guessing speed.

Increasing the number of VWCs will increase the number of concurrent wireless sessions to the AP. Consequently, wireless sessions started to timeout, then dropped after exceeding a certain number of active VWCs. The ratio between the number of successful wireless sessions (i.e., password guessing trials) to the total number of all wireless sessions (WS) was calculated and rep-

resented in Figure 5.12. Almost all wireless sessions were successful when we had fewer VWCs. The ratio started to drop when we further increased the number of VWCs.

5.3.4 Discussion

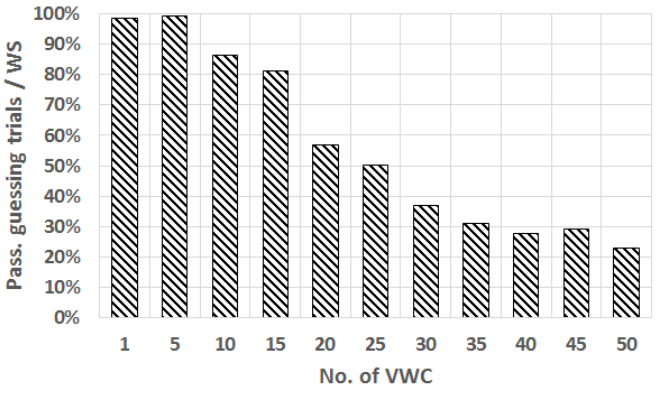
In this section, we presented a new technique to increase the intensity of the active dictionary attack on WPA-II enterprise in WLAN. The attacker can improve the password trial guessing speed by creating multiple virtual wireless clients authenticating to the AS at the same time. Such an improvement can reach up to 1700% increase in the guessing trials.

In WPA-II enterprise, obtaining the PMK from the 4-way handshaking is unpractical. The PMK is a random 256-bit key in length that changes every time the WC connects to the WLAN. Furthermore, retrieving the PMK will not compromise the WC password. On the another hand, our proposed technique reveals the actual password of the LWC.

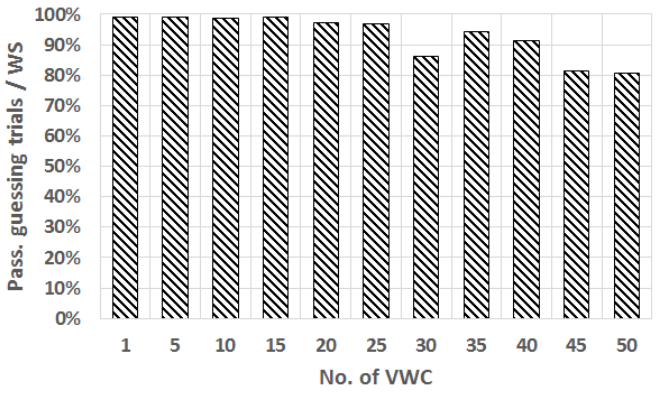
The proposed technique may fail if the username not captured in the first step of the attack. The network administrator can hide the username of the LWC by using Network Access Identifier (NAI) [98] in the outer authentication phase and use the actual LWC in the inner authentication phase. However, this requires a more complicated WLAN network implementation and can be only used with tunneled EAP authentication methods such EAP-TTLS and PEAP.

The network administrator may use locking mechanism to prevent brute force attack. However, no locking feature activated on FreeRadius server. By default, Radius server only delayed responding to VMCs requests to slow down the brute force attack. Our proposed attack downgrades the impact of such a protection feature. Each time a VWC is waiting for a response from the AS, another VWC can be created to test a different password.

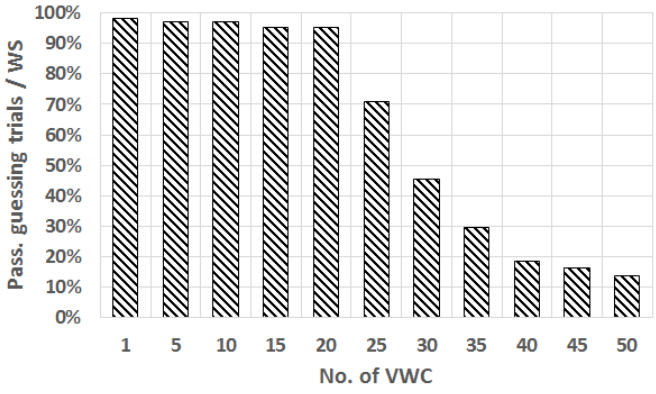
Our proposal attack intensity can be affected by the AP type, the wireless medium and the attacker/AS station performance. To have better results, an attacker can use a high-performance workstation and start the attack to the least congested AP. Also, an attacker can initiate a distributed attack using our proposed technique to all nearby APs that use the WIFI channel.



(a)



(b)



(c)

Figure 5.12: The ratio between the number of successful password guessing trials to the total number of all wireless sessions (WS) for (a) Dlink-DIR890L, (b) ASUS-RT-AC68U (c) Linksys WRT54.

Finally, the proposed attack can authenticate each VWC to the AS using different EAP authentication methods including PEAP and EAP-TTLS. However, the EAP-MD5 authentication method used in our testbed because of its fast speed and simplicity. Also, EAP-MD5 was the initial authentication method offered by the AS.

CHAPTER 6: Circumventing Wireless Traffic Shaping

6.1 Introduction

Accessing the Internet through Wi-Fi networks offers an inexpensive alternative for offloading data from mobile broadband connections. Businesses such as fast food restaurants, coffee shops, hotels, and airports, provide complimentary Internet access to their customers through Wi-Fi networks. Clients can connect to the Wi-Fi hotspot using different wireless devices. However, network administrators may apply traffic shaping to control the wireless client's upload and download data rates. Such limitation is used to avoid overloading the hotspot, thus providing fair bandwidth allocation. Also, it allows for the collection of money from the client to have access to a faster Internet service. In this chapter, we present a new technique to avoid bandwidth limitation imposed by Wi-Fi hotspots. The proposed method creates multiple virtual wireless clients using only one physical wireless interface card. Each virtual wireless client emulates a standalone wireless device. The combination of the individual bandwidth of each virtual wireless client results in an increase of the total bandwidth gained by the attacker.

6.2 Assumption

Our proposed attack targets Wi-Fi hotspots that imposed a bandwidth limitation on their wireless clients. Wireless network administrators avoid network overload by assigning a dedicated bandwidth to each wireless client. Based on the complexity of the wireless network design, a network administrator may use IP and MAC addresses to identify wireless clients. This type of bandwidth limitation is common in public Wi-Fi hotspots such as fast food restaurant, coffee shops, hotels, and airports.

6.2.1 Attack scenarios

The attacker can use the VWCs technique to pass the Wi-Fi hotspot traffic shaping in many scenarios. For example, whenever an application wants to access the Internet, a VWC is created and assigned to that application. In a web browser, each opened tab can be assigned to a separate VWC. However, some VWCs may still suffer from bandwidth limitation when they exceed the bandwidth allocated to them. For example, when an open browser tab requests to download a file, the VWC assigned to that browser tab cannot exceed the bandwidth limitation allocated to it.

Another scenario is when multiple VWCs work together to download a single file from the Internet. Each VWC starts downloading the single file from a different starting byte location. Some file servers allow clients to request a file from a specific byte number [32]. In this case, the VWCs will start downloading the file simultaneously from different locations. The parts received by the VWCs will be combined at the client's device. However, this scenario may not work when the server does not support byte-serving technique.

Finally, an attacker can set up a special server on the Internet to overcome the limitations in the previous scenarios. The attacker communicates directly with the special server while the special server retrieves the online resources (such as a file) from other servers on the Internet. The special server can obtain online resources faster than the VWCs, because the Internet connection speed between the special server and other servers, is not restricted by the bandwidth limitation such as the one between the attacker and the special server. After that, the special server can divide the online resource into multiple parts and send them to the attacker's VWCs.

In this chapter, we focused on avoiding the traffic shaping technique used by the hotspot when a client downloads a specific file on the Internet.

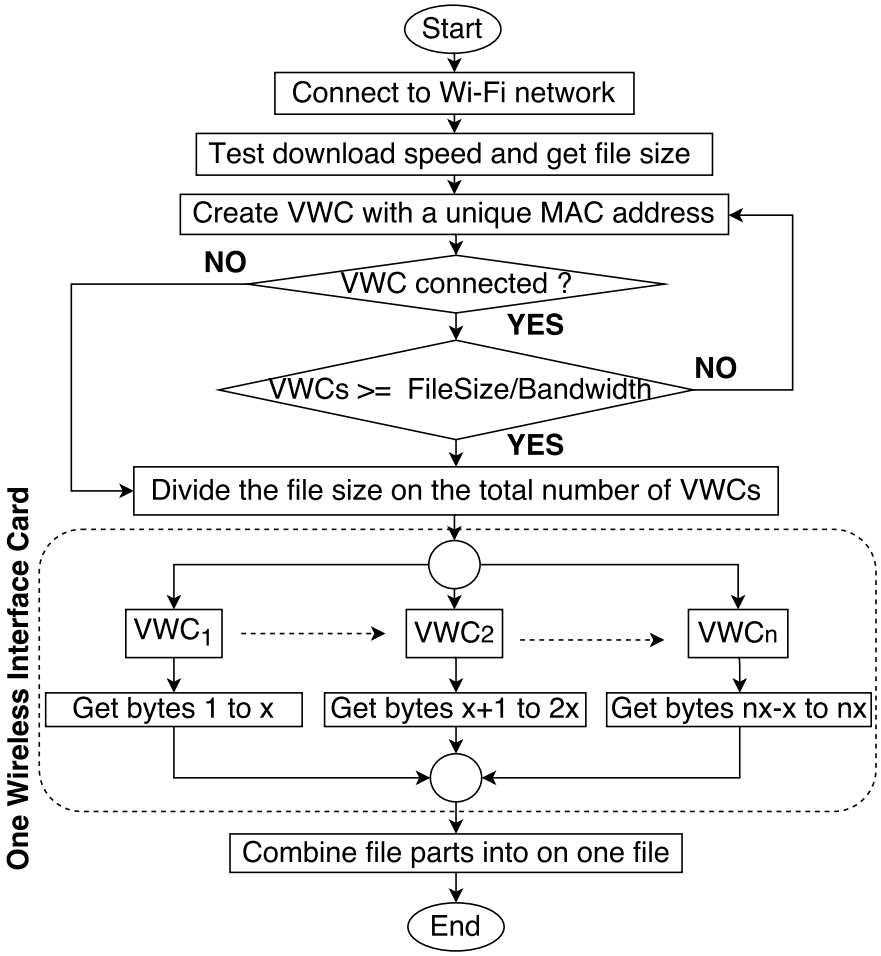


Figure 6.1: Proposed attack design on Wi-Fi hotspot traffic shaping using Virtual Wireless Clients.

6.2.2 Design

The proposed attack is based on the Virtual Wireless Clients technique. Using only one wireless network interface card, the attacker creates multiple Virtual Wireless Clients that each will have a unique IP and MAC address. All VWCs connect simultaneously to the Wi-Fi hotspot to access the Internet and start downloading the file as shown in figure 6.1.

First, the attacker connects to the Wi-Fi hotspot and test the bandwidth assigned to her by the wireless network administrator. The attacker can calculate the bandwidth limitation by measuring the time needed to download a small file from the Internet. After that, the attacker gets

Table 6.1: Software used in our testbed evaluation. Software were installed on Linux O.S except IDM which was installed on Windows O.S.

Protocol	Transfer Software	File Server	Port
TFTP	tftp	Xinetd	69
FTP	ftp	VsFTPd	20,21
HTTP	IDM	Apache2	80
HTTP	VWC (Proposed)	Apache2	80

the size of the actual file that will be download using the VWCs.

The maximum number of VWCs that will be used to download the file is based on the file size and the bandwidth allocation as shown in equation 6.1.

$$NumberofVWCs = \frac{FileSize}{AllocatedBandwidth} \tag{6.1}$$

Since the hotspot may limit the number of wireless clients to connect to it, our proposed attack keeps testing if the newly created VWC is able to reach the Internet.

After the attacker finishes creating the VWCs, each VWC starts requesting different parts of the file using a byte serving technique [32]. Since the number of the created VWC may be less than the number from equation 6.1, each VWC request part size equals to equation 6.2.

$$RequestingPartSize(x) = \frac{FileSize}{TotalVWCs} \tag{6.2}$$

After the VWCs finishes downloading all file parts, the software combines them into one.

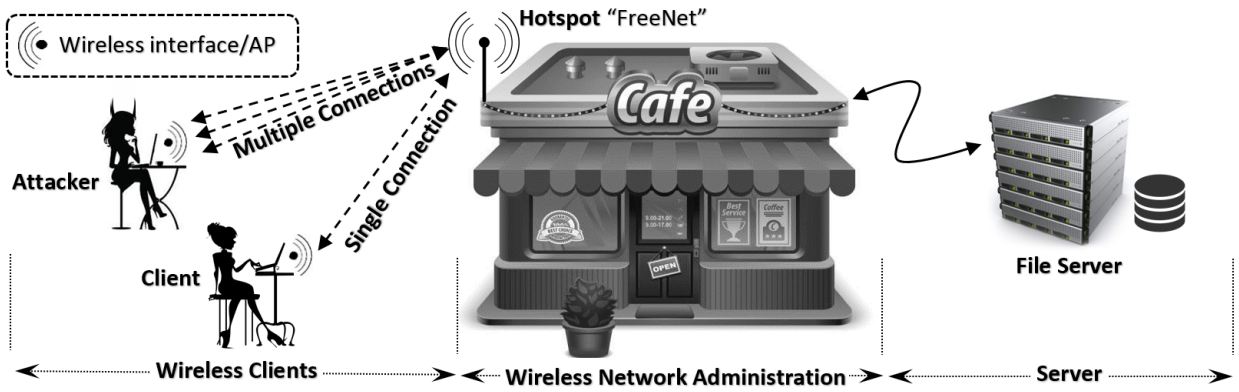


Figure 6.2: Proposed attack testbed set up. The attacker and the client used Laptops with TPE-NUSBDB wireless network interface card to connect to the wireless network. Dlink DIR-890L was used as a hotspot and bandwidth controller. We used a Linux based workstation to create the File Server.

6.2.3 Implementation

We have developed a software written in C language with the help of Loss Of Radio CONnectivity (LORCON2) library [77]. LORCON2 is an open source library used to allow the wireless client to inject crafted wireless frames and at the same time capture wireless traffic on the operating wireless channel.

First, the software authenticates and associates to the AP. After that, using DHCP protocol, the software obtains the network configuration from the DHCP server. Finally, using DNS and HTTP protocol, the software access the Internet. The developed software repeats the previous procedure for each created virtual wireless client.

6.3 Evaluation

Our proposed attack on the Wi-Fi hotspot bandwidth controller was evaluated in a real-life testbed set up shown in figure 6.2. The testbed set up consisted of three main parts, wireless clients, wireless network administration and file server.

The wireless client's side contains two laptops: one represents a regular wireless client, and

the other one resembles an attacker. We installed our proposed software on the attacker's laptop, while on the other laptop, we installed a different file transfer software such IDM. Both laptops connect to the wireless network side and start downloading files from the file server side. In this way, we can compare our downloading software speed with others. Table 6.1 illustrates software used in our evaluation.

On the wireless network administration side, we used D-Link DIR-890L with DD-WRT firmware to create the Wi-Fi hotspot. The hotspot assigned a specific download and upload speed to each wireless client using Quality Of Service (QoS) option. QoS use different packet scheduler algorithms such as Hierarchical Token Bucket (HTB) [99]. Any Wireless client that connects to the hotspot will be allocated 10 Kbytes data rate limit for upload and another 10 Kbyte for download. This uplink and downlink speed can be set to any arbitrary number, however, having a higher bandwidth limitation in our evaluation might produce inconsistent results since other factors such as channel congestion may affect the download/upload speed which is not part of the bandwidth limitation policies.

On the Server side, we created a standard file server. We installed TFTP, FTP and HTTP services on a Linux-based workstation. These services are standard file transfer protocols used to transfer data on the Internet [100]. The server response to TFTP on UDP port 69, FTP on TCP port 20 and 21 and HTTP on TCP port 80. The file server held different file size to be downloaded from the laptops at the wireless client side. All the traffic from the wireless client side to the file server side pass through the wireless network.

Our proposed attack took advantage of the byte serving technique used in HTTP/1.1 protocol. The wireless client can request a specific part of a file to be downloaded. If the requested range is valid, the server starts sending the file. Each virtual wireless client starts downloading different portions of the file simultaneously.

We tested the time needed for the regular wireless client and the attacker to download different files from the file server using TFTP, FTP, and HTTP. Each test was carried separately.

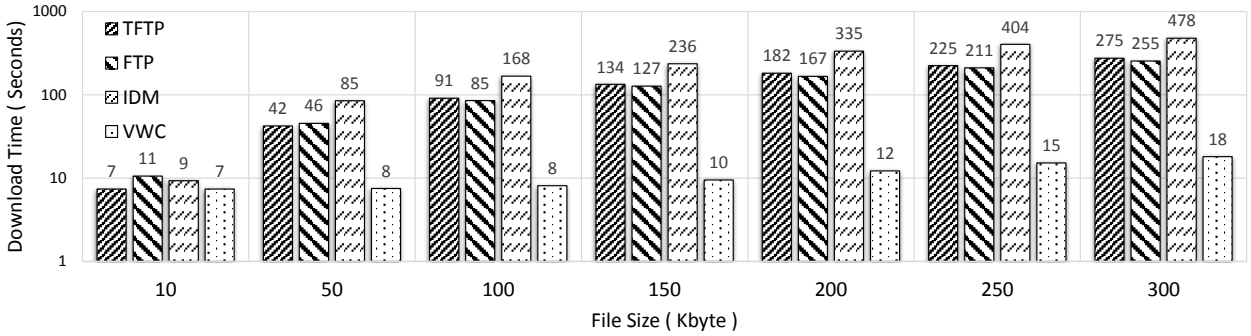


Figure 6.3: The time needed to download different files from the file server using the software in Table 6.1. Y-axis is log-10 scale.

We used the default TFTP client software in Linux O.S on the wireless client laptop while Xinetd software was used on the server side. VsFTPd was used on the server side to provide FTP protocol service, while the default Linux FTP software was used on the wireless client laptop. IDM software was used on the wireless client laptop to download the files using HTTP protocol while Apache server was used on the file server side. Finally, our proposed VWC software was installed on the attacker laptop and utilized to download files from the Apache server on the file server. Table 6.1 illustrates the client/server software used in our testbed evaluation.

The files on the server side were 10 to 300 Kbytes in size with 50 Kbytes increment. The link speed between the file server and the Wi-Fi hotspot was 100 Mbytes/second. However, the download and the upload speed between the wireless client side and the wireless network administration side was set to 10 Kbits/second. We started downloading each file using the software shown in Table 6.1.

Using our VWC technique, we set the number of virtual wireless clients based on equation 6.1. For example, for 10 Kbytes file size, we only created one VWC. Since the bandwidth limit was set to 10 Kbit/second, the time needed to download the file was 7.5 seconds. All other methods used to download the 10 Kbytes file size on the regular wireless client were able to finish in about 10 to 7 seconds. This is because the actual file size is 80 Kbits which need 80 Kbits / 10 Kbit/seconds =

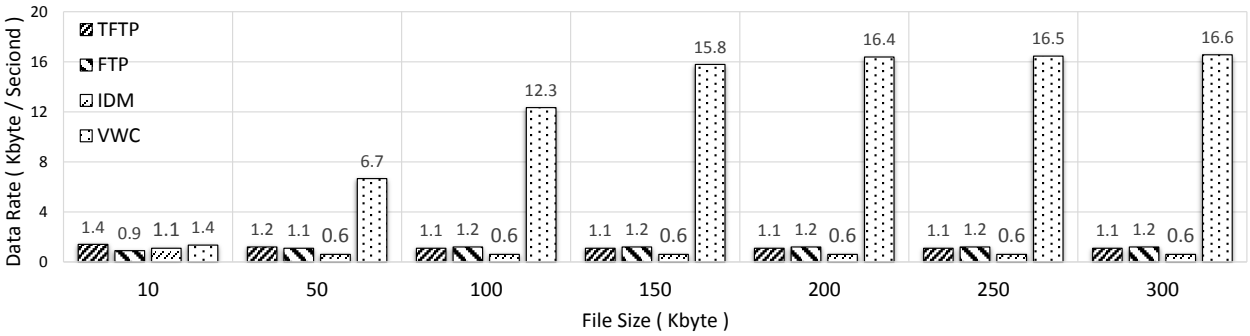


Figure 6.4: Comparison between download data rate for each software (Table 6.1) used in our testbed evaluation.

8 seconds to finish downloading.

We further increased the file size to 50 Kbytes. Since the file size is 50 Kbytes, our proposed technique created 5 VWC based on equation 6.1. In our proposed method, the time needed to download the 50 Kbytes was similar to the time needed to download the 10 Kbytes file. On the other hand, the methods used by the regular wireless client to download the 50 Kbytes file size increased by five folds to the time needed to download the 10 Kbytes file. Figure 6.3 illustrates the measured time to finish downloading different file sizes on both the attacker and the regular wireless client laptop.

However, during the increase of the number of VWCs, we noticed that the attacker started to receive a constant data rate from the Wi-Fi hotspot. When the number of VWCs were more than 20, the wireless connection to the hotspots started to timeout and drop as shown in figure 6.4. By using our software, the attacker was able to gain almost 16 folds bandwidth increase, while all other transfer methods had a constant download speed.

6.4 Discussion

In this chapter, we illustrated a practical attack on the traffic shaping protection used in public Wi-Fi network. We tested our attack effectiveness by comparing it with different file transfer

methods. By using VWCs technique, an attacker can bypass bandwidth limitation imposed by network administrators. The attacker creates multiple virtual wireless clients and connects them simultaneously to the Wi-Fi hotspot. The VWCs technique increased the wireless link speed up to 16 folds. However, the following limitation may affect the performance of such an attack.

First, our proposed attack is based on downloading files from servers that support a byte-serving technique, which is available in the HTTP/1.1 standard. Our attack will not work when the file server is using FTP or TFPT since both protocols do not support such a feature. In this case, the attacker can implement a proxy server on the Internet. When the attacker requests a resource from the Internet, the request will be sent to the proxy server. Since the connection speed between the proxy server and the Internet resource is fast, the proxy server acquires the resource, divides it and sends it to the attacker's VWCs. On the attacker's end, all the parts of the resource will be combined. In this case, the attacker can download files even when the file server does not support a byte serving technique.

Second, the wireless network administrator that provides credentials to their wireless clients may impose bandwidth limitations using the wireless client's username and password instead of using the IP and MAC address of the wireless client. In this case, our proposed attack will not work. However, this requires the network administrator to set up a more complex wireless network infrastructure and assign and give each wireless client a unique username and password.

Finally, increasing the number of VWCs will increase the traffic on the wireless channel that can affect the download/upload speed. Also, certain APs limit the number of the wireless clients that can connect to it simultaneously.

CHAPTER 7: Conclusion and Future work

Securing WLAN is a challenging task. In this dissertation, we investigated the current WLAN vulnerabilities and their solutions by targeting both sides of the wireless network, the wireless client, and network administrator. We also took into consideration both types of WLANs, open and secure. We presented different attacks and solution to improve the security of Wi-Fi networks.

First, a novel ETA detection technique was proposed to detect ETA using different gateways. The proposed method is a lightweight client-side approach. The detection method was prototyped and evaluated in real-world scenarios. The procedure detection time is short, and its variance does not affect the detection efficiency.

Second, we presented a real-time client-side ETA detection of ETA using single ISP gateway. In our ETA detection, the wireless client can scan the whole 11 Wi-Fi channels of 802.11 b/g network for ETA in approximately half a minute. No training data and/or network fingerprint used in the detection. Our proposed detection efficiency was mathematical modeled and implemented in real life scenario with a detection rate of $\approx 100\%$.

Third, based on the previous ETA detection techniques, a comprehensive real-time client-side ETA detection was proposed. Both, ETA using different and single ISP gateways can be detected in parallel using virtual wireless clients technique. Having both detections running simultaneously prevented attacker maneuvers and reduced the other all detection time.

Fourth, a vulnerability in the mobile networks' data usage billing system was demonstrated by using a mobile data consumption attack. The attack works by delivering a malicious captive portal to the victim, forcing them to connect to their mobile data plan, and causing them to use data via a download initiated by the captive portal. Our attack would work when the victim connects to a free open Wi-Fi network that is available in most coffee shops, fast food restaurants, and airports.

Fifth, we introduced an active WPA2-PSK dictionary attack that can be utilized to recover

passphrase when the attacker is unable to capture the four-way handshaking frames between the AP and an authorized user. The speed of the active WPA2-PSK dictionary guessing attack improved by implementing two novel techniques. First, the attacker created multiple virtual wireless clients (VWCs) using a single WLAN interface card. Each VWC emulated a standalone wireless client to the AP. All the VWCs started guessing the passphrase of the WPA2-PSK in a parallel manner. Second, as long as the wireless session is active, a VWC kept guessing the passphrase until a de-authentication frame is received from the AP. Our proposed attack was implemented and evaluated using different types of off-the-shelf wireless APs. Our results showed that the two proposed techniques might improve the attack speed up to 100-fold compared to the traditional single client active dictionary attack.

Sixth, we used the same technique to increase the dictionary attack intensity on WPA-II enterprise in WLAN. Such an attack is significant when other attacks, such as MITM, are not feasible. The attack uses only one WIC to create multiple VWCs. Each VWC authenticates to the AS as a standalone WC. Our proposed technique implemented and evaluated using different off-the-shelf APs. The most popular RADIUS server (FreeRadius) was used as an AS in the testbed set up. The final results showed an improvement of 1700% in the intensity of the active dictionary attack by using VWC technique, compared to the traditional one wireless client.

Finally, network administrators may impose traffic shaping techniques to protect their wireless network from being overloaded and offer fair bandwidth allocation. Also, they may require the client to pay in order to increase their Internet network connection speed. However, using a VWC technique, an attacker can bypass such a limitation by creating multiple virtual wireless clients using only one physical wireless interface card. Each VWC connects to the wireless network as a standalone wireless client and reserve a separate bandwidth. The total bandwidth that is being used by the attacker, in this case, equals to the summation of all the VWCs bandwidths. Our proposed technique was implemented and evaluated using off the shelf devices. The result shows that the attacker can speed the Internet connection up to 16 folds compared to other file transfer methods.

7.1 Future work

In our proposals, it is important for the WC to be able to monitor wireless traffic (WiFi in promiscuous mode). Such a condition depends on the WC OS, wireless interface card driver, and chipset. In our experiments, we used Linux OS and LORCON2 driver with Atheros based WiFi USB interface card. As future work, our proposed system can be ported to mobile O.S, e.g., Android, or on a Windows system using different wireless interface card drivers and chipsets. For example, Windows O.S users can use Winpcap driver with supported interface cards [101]. Android O.S users can use PCAP library [102] on RTL8187 chipset based wireless interface card.

LIST OF REFERENCES

[1] N. Zhang and H. Bao. Wireless network technology and its applications. In *2009 International Conference on Networks Security, Wireless Communications and Trusted Computing*, volume 2, pages 635–638, April 2009.

[2] Enterprise wlan market by component, hardware (wireless access points, ap antennas, wireless lan controllers, multigigabit switching, wireless location appliance), software, service, vertical, and region - global forecast to 2021. Technical report, April 2017.

[3] A. Kumar and H. Om. A secure, efficient and lightweight user authentication scheme for wireless lan. In *2016 International Conference on Emerging Trends in Engineering, Technology and Science (ICETETS)*, pages 1–9, Feb 2016.

[4] Marcel Medwed. Iot security challenges and ways forward. In *Proceedings of the 6th International Workshop on Trustworthy Embedded Devices*, TrustED '16, pages 55–55, New York, NY, USA, 2016. ACM.

[5] Ericsson mobility report. Technical report, June 2015.

[6] M. Seufert, T. Griepentrog, V. Burger, and T. Hofeld. A simple wifi hotspot model for cities. *IEEE Communications Letters*, 20(2):384–387, Feb 2016.

[7] F. Lanze, A. Panchenko, I. Ponce-Alcaide, and T. Engel. Hacker's toolbox: Detecting software-based 802.11 evil twin access points. In *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*, pages 225–232, Jan 2015.

[8] H. Mustafa and W. Xu. Cetad: Detecting evil twin access point attacks in wireless hotspots. In *Communications and Network Security (CNS), 2014 IEEE Conference on*, pages 238–246, Oct 2014.

[9] C. Yang, Y. Song, and G. Gu. Active user-side evil twin access point detection using statistical techniques. *IEEE Transactions on Information Forensics and Security*, 7(5):1638–1651, Oct 2012.

[10] Y. Song, C. Yang, and G. Gu. Who is peeping at your passwords at starbucks? to catch an evil twin access point. In *2010 IEEE/IFIP International Conference on Dependable Systems Networks (DSN)*, pages 323–332, June 2010.

[11] S. Nikbakhsh, A. B. A. Manaf, M. Zamani, and M. Janbeglou. A novel approach for rogue access point detection on the client-side. In *Advanced Information Networking and Applications Workshops (WAINA), 2012 26th International Conference on*, pages 684–687, March 2012.

[12] Intel. *What is Wi-Fi roaming aggressiveness*. August 2014.

[13] M. Marlinspike. More tricks for defeating ssl in practice. *Black Hat USA*, 2009.

[14] Roland van Rijswijk-Deij, Anna Sperotto, and Aiko Pras. Dnssec and its potential for ddos attacks: A comprehensive measurement study. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, IMC '14, pages 449–460, New York, NY, USA, 2014. ACM.

[15] International Telecommunication Union. Global and regional ict data. Report, 2016.

[16] L. Qiu, H. Rui, and A. Whinston. When cellular capacity meets wifi hotspots: A smart auction system for mobile data offloading. In *2015 48th Hawaii International Conference on System Sciences*, pages 4898–4907, Jan 2015.

[17] O. Nakhila, E. Dondyk, M. F. Amjad, and C. Zou. User-side wi-fi evil twin attack detection using ssl/tcp protocols. In *Consumer Communications and Networking Conference (CCNC), 2015 12th Annual IEEE*, pages 239–244, Jan 2015.

[18] O. Nakhila and C. Zou. User-side wi-fi evil twin attack detection using random wireless channel monitoring. In *MILCOM 2016 - 2016 IEEE Military Communications Conference*, pages 1243–1248, Nov 2016.

[19] C. Yang, Y. Song, and G. Gu. Active user-side evil twin access point detection using statistical techniques. *IEEE Transactions on Information Forensics and Security*, 7(5):1638–1651, Oct 2012.

[20] Halil Ibrahim Bulbul, Ihsan Batmaz, and Mesut Ozel. Wireless network security: Comparison of wep (wired equivalent privacy) mechanism, wpa (wi-fi protected access) and rsn (robust security network) security protocols. In *Proceedings of the 1st International Conference on Forensic Applications and Techniques in Telecommunications, Information, and Multimedia and Workshop*, e-Forensics '08, pages 9:1–9:6, ICST, Brussels, Belgium, Belgium, 2008.

[21] Y. Zou, J. Zhu, X. Wang, and L. Hanzo. A survey on wireless security: Technical challenges, recent advances, and future trends. *Proceedings of the IEEE*, 104(9):1727–1765, Sept 2016.

[22] Erik Tews, Ralf-Philipp Weinmann, and Andrei Pyshkin. Breaking 104 bit wep in less than 60 seconds. In *Proceedings of the 8th International Conference on Information Security Applications*, WISA'07, pages 188–202, Berlin, Heidelberg, 2007. Springer-Verlag.

[23] S. Bohn, S. Grob, R. Nubgen, and P. Schwann. An automated system interoperability test bed for wpa and wpa2. In *2006 IEEE Radio and Wireless Symposium*, pages 615–618, Jan 2006.

[24] Andrew Gin and Ray Hunt. Performance analysis of evolving wireless ieee 802.11 security architectures. In *Proceedings of the International Conference on Mobile Technology, Applications, and Systems*, Mobility '08, pages 101:1–101:6, New York, NY, USA, 2008. ACM.

[25] H. Hwang, G. Jung, K. Sohn, and S. Park. A study on mitm (man in the middle) vulnerability in wireless network using 802.1x and eap. In *Information Science and Security, 2008. ICISS. International Conference on*, pages 164–170, Jan 2008.

[26] Pieter Robyns, Bram Bonné, Peter Quax, and Wim Lamotte. Short paper: Exploiting wpa2-enterprise vendor implementation weaknesses through challenge response oracles. In *Proceedings of the 2014 ACM Conference on Security and Privacy in Wireless; Mobile Networks*, WiSec '14, pages 189–194, New York, NY, USA, 2014. ACM.

[27] Sebastian Brenza, Andre Pawlowski, and Christina Pöpper. A practical investigation of identity theft vulnerabilities in eduroam. In *Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, WiSec '15, pages 14:1–14:11, New York, NY, USA, 2015. ACM.

[28] Cisco Inc. *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016 to 2021*, 2017.

[29] D. J. Henry. *CCNP Wireless (642-732 CUWSS)*. Cisco Press.

[30] Wifi hotspot for single & chain coffee shop and cafes. Technical report, 2017.

[31] James Eades. Recognising the challenges and trends faced in technology within the hotel industry. Technical report, 2017.

[32] Roy T. Fielding, Yves Lafon, and Julian Reschke. Hypertext Transfer Protocol (HTTP/1.1): Range Requests. RFC 7233, June 2014.

[33] Tonec Inc. *Internet Download Manager*, 2016.

[34] F. Zhang, W. He, and X. Liu. Defending against traffic analysis in wireless networks through traffic reshaping. In *2011 31st International Conference on Distributed Computing Systems*, pages 593–602, June 2011.

[35] O. Nakhila, A. Attiah, Y. Jinz, and C. Zou. Parallel active dictionary attack on wpa2-psk wi-fi networks. In *Military Communications Conference, MILCOM 2015 - 2015 IEEE*, pages 665–670, Oct 2015.

[36] O. Nakhila and C. Zou. Parallel active dictionary attack on ieee 802.11 enterprise networks. In *MILCOM 2016 - 2016 IEEE Military Communications Conference*, pages 265–270, Nov 2016.

[37] SMC Network. *Wireless Hotspot Solutions*. 2008.

[38] S. Jana and S. K. Kasera. On fast and accurate detection of unauthorized wireless access points using clock skews. *IEEE Transactions on Mobile Computing*, 9(3):449–462, March 2010.

[39] Fabian Lanze, Andriy Panchenko, Benjamin Braatz, and Thomas Engel. Letting the puss in boots sweat: Detecting fake access points using dependency of clock skews on temperature. In *Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security*, ASIA CCS '14, pages 3–14, New York, NY, USA, 2014. ACM.

[40] Salih Bacanli Cliff Zou Dean Wasil, Omar Nakhila and Damla Turgut. Exposing vulnerabilities in mobile networks: A mobile data consumption attack. In *The 14th International Conference on Mobile Ad-hoc and Sensor Systems (IEEE MASS 2017): The Fourth National Workshop for REU Research in Networking and Systems*, pages 449–460, 2016.

[41] A. Alruban and E. Everitt. Two novel 802.1x denial of service attacks. In *Intelligence and Security Informatics Conference (EISIC), 2011 European*, pages 183–190, Sept 2011.

[42] Dirk van der Walt. Freeradius project, 2016.

[43] Dirk van der Walt. *FreeRADIUS Beginner's Guide*. Packt Publishing, 9 2011.

[44] O. Nakhila and C. Zou. Circumvent traffic shaping using virtual wireless clients in ieee 802.11 wireless local area network. In *Military Communications Conference, MILCOM 2017 - 2017 IEEE*, Oct 2017.

[45] Jyrki Penttinen. *The telecommunications handbook : engineering guidelines for fixed, mobile, and satellite systems*. John Wiley & Sons Inc, Chichester, West Sussex, UK, 2015.

[46] D. J. Deng, K. C. Chen, and R. S. Cheng. Ieee 802.11ax: Next generation wireless local area networks. In *10th International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*, pages 77–82, Aug 2014.

[47] M. C. Ghanem and D. N. Ratnayake. Enhancing wpa2-psk four-way handshaking after re-authentication to deal with de-authentication followed by brute-force attack a novel re-authentication protocol. In *2016 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (CyberSA)*, pages 1–7, June 2016.

[48] Ankit Panch and Santosh Kumar Singh. A novel approach for evil twin or rogue ap mitigation in wireless environment. *International Journal of Security and Its Applications*, 4(4):33–38, October 2010.

[49] Robert Sherwood. *Discovering and Securing Shared Resources on the Internet*. Univ. of Maryland, 2008.

[50] Charles Scott, Paul Wolfe, and Mike Erwin. *Virtual Private Networks*. O'Reilly & Associates, Inc., Sebastopol, CA, USA, 1998.

[51] Diogo Mónica and Carlos Ribeiro. *WiFiHop - Mitigating the Evil Twin Attack through Multi-hop Detection*, pages 21–39. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[52] M. Mayhew and R. Muresan. Implementation of a decoupling based power analysis attack countermeasure. *IET Circuits, Devices Systems*, 10(6):528–535, 2016.

[53] Chunyi Peng, Chi-Yu Li, Hongyi Wang, Guan-Hua Tu, and Songwu Lu. Real threats to your data bills: Security loopholes and defenses in mobile data charging. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, CCS '14, pages 727–738, New York, NY, USA, 2014. ACM.

[54] Y. Song, C. Yang, and G. Gu. Who is peeping at your passwords at starbucks? to catch an evil twin access point. In *IEEE/IFIP International Conference on Dependable Systems Networks (DSN)*, pages 323–332, June 2010.

[55] Chae T. Im Wan S. Yi Yoo J. Won Dong W. Kang, Joo H. Oh. A practical attack on mobile data network using ip spoofing. In *Applied Mathematics & Information Sciences*, pages 2345–2353, November 2013.

[56] Wai Kay Leong, Aditya Kulkarni, Yin Xu, and Ben Leong. Unveiling the hidden dangers of public ip addresses in 4g/lte cellular data networks. In *Workshop on Mobile Computing Systems and Applications*, pages 16:1–16:6, 2014.

[57] Andrew Gin and Ray Hunt. Performance analysis of evolving wireless ieee 802.11 security architectures. In *Proceedings of the International Conference on Mobile Technology, Applications, and Systems*, Mobility '08, pages 101:1–101:6, New York, NY, USA, 2008. ACM.

[58] Iso/iec international standard - 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 amendment 6: Medium access control (mac) security enhancements. *ISO/IEC 8802-11, Second edition: 2005/Amendment 6 2006: IEEE STD 802.11i-2004 (Amendment to IEEE Std 802.11-1999)*, pages c1–178, July 2004.

[59] David Coleman. *CWSP Certified Wireless Security Professional Official Study Guide Exam PW0-204*. John Wiley & Sons, New York, NY, 2011.

[60] Thomas d'Otreppe de Bouvette. *Aircrack-ng*, 2017.

[61] I. P. Mavridis, A. I. E. Androulakis, A. B. Halkias, and P. Mylonas. Real-life paradigms of wireless network security attacks. In *2011 15th Panhellenic Conference on Informatics*, pages 112–116, Sept 2011.

[62] Jens Steube. *Hashcat*, 2017.

[63] 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 amendment 4: Protected management frames. *IEEE Std 802.11w-2009 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, and IEEE Std 802.11y-2008)*, pages 1–111, Sept 2009.

[64] D. Zisiadis, S. Kopsidas, A. Varalis, and L. Tassiulas. Enhancing wps security. In *2012 IFIP Wireless Days*, pages 1–3, Nov 2012.

[65] *Reaver WPS Projectr*, 2011.

[66] K. Hoeper and L. Chen. An inconvenient truth about tunneled authentications. In *Local Computer Networks (LCN), 2010 IEEE 35th Conference on*, pages 416–423, Oct 2010.

[67] Fanzheng Kong and Weili Huang. Ieee802.1x of protocol analysis and improvement. In *Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on*, volume 3, pages V3–282–V3–285, Aug 2010.

[68] M. Cai, Z. Wu, and J. Zhang. Research and prevention of rogue ap based mitm in wireless network. In *P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2014 Ninth International Conference on*, pages 538–542, Nov 2014.

[69] P. Q. Ding, J. N. Holliday, and A. Celik. Improving the security of wireless lans by managing 802.1x disassociation. In *Consumer Communications and Networking Conference, 2004. CCNC 2004. First IEEE*, pages 53–58, Jan 2004.

[70] Khidir M. Ali and Thomas J. Owens. Selection of an eap authentication method for a wlan. *Int. J. Inf. Comput. Secur.*, 1(1/2):210–233, January 2007.

[71] Cisco. Captive Portal Configuration Guide. Technical report, 2014.

[72] Unifi enterprise wifi system. Technical report, 2017.

[73] Se-young Yu, Nevil Brownlee, and Aniket Mahanti. Performance and fairness issues in big data transfers. In *Proceedings of the 2014 CoNEXT on Student Workshop*, CoNEXT Student Workshop '14, pages 9–11, New York, NY, USA, 2014. ACM.

[74] Alex Gizis. *Speedify Software*, 2017.

[75] Reka Albert, Hawoong Jeong, and Albert-Laszlo Barabási. Internet: Diameter of the World-Wide Web. *Nature*, 401(6749):130–131, September 1999.

[76] Paramvir Bahl, Ranveer Chandra, Jitendra Padhye, Lenin Ravindranath, Manpreet Singh, Alec Wolman, and Brian Zill. Enhancing the security of corporate wi-fi networks using dair. In *Proceedings of the 4th International Conference on Mobile Systems, Applications and Services*, MobiSys '06, pages 1–14, New York, NY, USA, 2006. ACM.

[77] Joshua Wright and Michael Kershaw. Lorcon2 project, 2016.

[78] Angela Orebaugh, Gilbert Ramirez, Jay Beale, and Joshua Wright. *Wireshark & Ethereal Network Protocol Analyzer Toolkit*. Syngress Publishing, 2007.

[79] S. Rayanchu, A. Mishra, D. Agrawal, S. Saha, and S. Banerjee. Diagnosing wireless packet losses in 802.11: Separating collision from weak signal. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, April 2008.

[80] S. C. Wang and A. Helmy. Beware: Background traffic-aware rate adaptation for ieee 802.11. *IEEE/ACM Transactions on Networking*, 19(4):1164–1177, Aug 2011.

[81] David Coleman. *CWNA Certified Wireless Network Administrator study guide : (exam PWO-100*. Wiley John Wiley distributor, Hoboken, N.J. Chichester, 2006.

[82] O. Nakhila, E. Dondyk, M. F. Amjad, and C. Zou. User-side wi-fi evil twin attack detection using ssl/tcp protocols. In *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*, pages 239–244, Jan 2015.

[83] Alan Holt. *802.11 wireless networks : security and analysis*. Springer, London, 2010.

[84] Jorge Olenewa. *Guide to wireless communications*. Course Technology/Cengage Learning, Boston, MA, 2014.

[85] Praphul Chandra. *Wireless security*. Newnes/Elsevier, Amsterdam Boston, 2009.

[86] Chwan. *Introduction to computer networks and cybersecurity*. CRC Press, Boca Raton, FL, 2013.

[87] Ben Miller Peter Mackenzie David A. Westcott, David D. Coleman. Cwap certified wireless analysis professional official study guide: Exam pw0-270l. In *John Wiley & Sons*, March 2011.

[88] Ieee standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements - part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications. *IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999)*, pages 1–1076, June 2007.

[89] Thomas d'Otreppe de Bouvette. *Aircrack-ng*, 2016.

[90] Internet Systems Consortium. *Dhcpd*, 2016.

[91] DNSchef. *Iphelix*, 2014.

[92] Robert McCool. *Apache HTTP Server*, 2016.

[93] L. Ge, L. Wang, and L. Xu. A method for cracking the password of wpa2-psk based on sa and hmm. In *2016 3rd International Conference on Information Science and Control Engineering (ICISCE)*, pages 59–62, July 2016.

[94] 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: Amendment 6: Medium access control (mac) security enhancements. *IEEE Std 802.11i-2004*, pages 1–190, July 2004.

[95] M. Vanhoef and F. Piessens. Key reinstallation attacks: Forcing nonce reuse in wpa2. In *The ACM Conference on Computer and Communications Security (CCS)*, Oct 2017.

[96] Mathy Vanhoef and Frank Piessens. Practical verification of wpa-tkip vulnerabilities. In *Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security*, ASIA CCS '13, pages 427–436, New York, NY, USA, 2013. ACM.

[97] Jyh-Cheng Chen and Yu-Ping Wang. Extensible authentication protocol (eap) and ieee 802.1x: tutorial and empirical experience. *IEEE Communications Magazine*, 43(12):supl.26–supl.32, Dec 2005.

[98] A. DeKok. The network access identifier. RFC 7542, RFC Editor, May 2015.

[99] J. L. Valenzuela, A. Monleon, I. San Esteban, M. Portoles, and O. Sallent. A hierarchical token bucket algorithm to enhance qos in ieee 802.11: proposal, implementation and evaluation. In *IEEE 60th Vehicular Technology Conference, 2004. VTC2004-Fall. 2004*, volume 4, pages 2659–2662 Vol. 4, Sept 2004.

[100] Behrouz Forouzan. *Data communications and networking*. McGraw-Hill Higher Education, New York, 2007.

[101] WinPcap Team. *WinPcap project 4.1.3*, 2017.

[102] Mike Kershaw. *Kismet project*, 2017.