WiFi Network Security Protocols

476 Class Project: Team 07
Asa Wiese, Alex Gibson, Nicholas Kim
Oregon State University, Corvallis, OR, USA, wiesea,gibsona2,kimnich@oregonstate.edu

***Abstract*—Wireless Local Area Networks (WLANs) have become an integral part of daily life, offering convenient internet access. However, the inherent vulnerability of wireless communication to eavesdropping and unauthorized access poses significant security challenges. To mitigate these risks, various WiFi network security protocols have been developed. These protocols authenticate users, encrypt data, and maintain message integrity to protect against threats such as data breaches and network disruptions.**

This paper aims to explore and compare the evolution of WiFi security protocols, focusing on WEP (Wired Equivalent Privacy), WPA (Wi-Fi Protected Access), WPA2 (Wi-Fi Protected Access 2), and WPA3 (Wi-Fi Protected Access 3). It examines their authentication mechanisms, encryption algorithms, and message integrity checks to understand their strengths and weaknesses.

Starting with the foundational concepts of WiFi security, the paper delves into the specifics of each protocol. WEP, the first attempt at securing wireless networks, fell short due to vulnerabilities in its authentication, encryption, and message integrity mechanisms. WPA addressed many of these shortcomings by introducing stronger authentication methods, larger IVs, and a more robust message integrity algorithm. However, it still relied on the RC4 algorithm inherited from WEP, leading to potential vulnerabilities.

WPA2 marked a significant advancement by adopting the Advanced Encryption Standard (AES) and introducing the Counter Mode with Cipher Block Chaining Message Authentication Code Protocol (CCMP) for encryption. Despite its improvements, WPA2 remains susceptible to attacks such as dictionary attacks and Key Reinstallation Attacks (KRACK). WPA3, introduced to address WPA2's vulnerabilities, employs the Dragonfly handshake for authentication and AES-GCMP for encryption. However, it still faces challenges such as rogue access point attacks and ARP spoofing.

The paper provides insights into the strengths and weaknesses of each protocol, offering guidance for network administrators, security professionals, and WiFi users to make informed decisions about securing their networks. By understanding the evolution of WiFi security protocols and implementing appropriate measures, users can safeguard their data, maintain network integrity, and ensure a safe online experience.

I Introduction: Motivation and Objectives

A. Problem Description

Wireless Local Area Networks (WLANs) have become an indispensable part of modern life. From connecting to the internet at home to accessing public WiFi in cafes and airports, these networks offer ubiquitous and convenient internet access. However, the very nature of wireless communication introduces a significant security challenge: unlike wired networks, data transmissions over WiFi are inherently open to eavesdropping and unauthorized access. This vulnerability exposes users to a range of threats, from data breaches and identity theft to malware infections and network disruptions.

To combat these security risks, various WiFi network security protocols have been developed. These protocols implement a set of rules and mechanisms designed to authenticate users, encrypt data, and maintain message integrity. These protocols verify the legitimacy of devices attempting to connect to the network, ensuring only authorized users gain access. They scramble data transmissions using encryption algorithms, rendering them unintelligible to unauthorized parties. And, they guarantee that data remains unaltered during transmission, preventing attackers from manipulating information. A thorough understanding of these protocols and their evolution is instrumental to assuring one has the necessary context and information to keep their own digital information secure.

1) Protecting Sensitive Information:

Modern WiFi networks are often used to access a vast array of personal and professional data, including financial records, emails, confidential documents, and online accounts. Strong security protocols act as a vital shield, protecting this sensitive information from unauthorized access. Without proper encryption, anyone within range of the network could potentially intercept data transmissions, putting users at risk of identity theft, financial fraud, and exposure of sensitive data.

2) Ensuring Network Integrity:

Unsecured WiFi networks are susceptible to a variety of attacks. Malicious actors can exploit vulnerabilities to gain unauthorized access to a network's resources. This can potentially lead to denial-of-service attacks,

where attackers overwhelm the network with traffic, rendering it unusable for legitimate users. Additionally, compromised networks can be used as launchpads for further attacks on other devices and networks. Robust security protocols play a critical role in preventing unauthorized access and maintaining the integrity and stability of WiFi networks.

3) Maintaining User Privacy:

The increasing reliance on WiFi for online activities such as banking, shopping, and social media interactions necessitates a heightened focus on user privacy. Secure protocols ensure that data transmissions remain confidential. This prevents unauthorized monitoring of online activities, protecting users from targeted advertising, online tracking, and potential surveillance.

B. Technical Objectives and Goals

This report delves into the world of WiFi network security protocols, specifically focusing on the four major advancements: WEP (Wired Equivalent Privacy), WPA (Wi-Fi Protected Access), WPA2 (Wi-Fi Protected Access 2), and WPA3 (Wi-Fi Protected Access 3). We will analyze the security mechanisms employed by each protocol, exploring their approaches to authentication, encryption, and message integrity. By comparing and contrasting these protocols, we can understand their strengths and weaknesses, and identify the most suitable option for different use cases.

C. Structure

The following sections categorize our analysis of WiFi network security protocols:

1) Background and Fundamentals:

This section lays the groundwork for understanding WiFi security by introducing key concepts such as authentication, encryption, message integrity, and common threats. We explain the basic functionalities of WLANs and different types of WiFi networks (home, enterprise, public).

2) State-of-the-Art Solution Approaches:

We delve into the four major WiFi security protocols: WEP, WPA, WPA2, and WPA3. Each protocol will be analyzed in detail, explaining its approach to authentication, encryption, and message integrity. A comparison will summarize the strengths, weaknesses, and suitability for various use cases of each protocol.

3) Case Study:

The case study section focuses on the effectiveness of two techniques introduced in WPA3 against two security threats present in previous WiFi security protocols. It demonstrates a dictionary attack against a WPA2 network and the establishment of a rogue access point and analyzes the effectiveness of Simultaneous Authentication of Equals and Protected Management Frames to mitigate these threats.

Through this investigation, we aim to equip network administrators, security professionals, and everyday WiFi users with the knowledge necessary to implement strong security measures. By understanding the importance of secure protocols and selecting the appropriate one, users can safeguard their data, maintain network integrity, and ensure a safe and private online experience.

II Background and Fundamental Concepts

In the ever-evolving world of technology, securing our wireless connections is paramount. The need for robust security protocols becomes especially crucial when dealing with WiFi networks, which, by their very nature, operate in an open and accessible environment. This vulnerability exposes users to a range of threats, from data breaches and identity theft to malware infections and network disruptions. Understanding the Building Blocks of WiFi Security Before jumping into the specifics for each security protocol, let's establish a foundational understanding of the key concepts involved in securing WiFi networks.

User/Message Authentication: This process verifies the legitimacy of devices attempting to connect to the network. Only authorized devices are granted access.

Confidentiality/Privacy: This ensures that data transmissions remain encrypted, rendering them unintelligible to anyone eavesdropping on the network. Encryption essentially scrambles the data using a secret key, making it unreadable without the proper decryption key.

Message Integrity: This guarantees that data remains unaltered during transmission. It involves computing a checksum, a unique value based on the data itself, and attaching it to the message. The receiver then calculates its own checksum and compares it to the received value. Any discrepancies indicate potential tampering with the data.

Access Control: This defines the level of access granted to different users or devices on the network. It determines what resources and functionalities each user can utilize.

Now that we have a grasp of the fundamental security principles, let's explore how each security protocol addresses these factors starting with WEP.

The Wired Equivalency Protocol (WEP), introduced in 1997, was the first attempt at securing wireless networks. While its intention was to provide a level of security comparable to wired networks, WEP fell short on its promises. This background will delve into

the core functionalities of WEP, explaining its approach to user authentication, data encryption, and message integrity. We will also explore the inherent weaknesses that rendered it susceptible to various attacks.

For user/message authentication, WEP employs a shared key mechanism. All devices on the network need to possess the same secret key to authenticate and gain access. This approach, while seemingly simple, presents significant drawbacks [1]. For confidentiality/privacy, WEP utilizes the Rivest Cipher 4 (RC4) algorithm for encryption. This algorithm generates a pseudo-random stream of bits, which are then combined with the original data (plaintext) to create an encrypted version (ciphertext). For message integrity, WEP relies on a weak checksum algorithm called Cyclic Redundancy Check (CRC) to ensure message integrity. This algorithm computes a 32-bit value based on the data packet and a secret key. Lastly, for access control, WEP solely relies on shared keys for authentication, offering a single layer of security. This limited approach makes WEP vulnerable to unauthorized access if the shared key gets compromised.

Despite its initial intentions, WEP's shortcomings became increasingly evident over time. An attacker eavesdropping on the network traffic might potentially capture the shared key, and the limited size of the IV in WEP' s encryption process makes it susceptible to attacks that could exploit key reuse. The CRC algorithm employed by WEP can be easily manipulated by attackers, allowing them to alter data packets without detection. These vulnerabilities render WEP ineffective in securing WiFi networks, making it obsolete and unusable. These flaws prompted the development of more robust security protocols like WPA (WiFi Protected Access).

While WEP paved the way for securing wireless networks, its weaknesses necessitated advancements in security protocols. WPA, introduced in 2003, addressed many of WEP's shortcomings by implementing stronger authentication mechanisms, improved encryption algorithms with larger IVs, and a more robust message integrity check.

WPA introduced two significant advancements in user authentication. The first is Extensible Authentication Protocol (EAP) which is a more secure method that allows for various authentication mechanisms, such as username/password combinations or token-based authentication, offering greater flexibility and security compared to WEP's shared key approach. The second is Pre-Shared Key (PSK) where users share a passphrase, which is then converted into a stronger encryption key for authentication. WPA also utilizes a 4-way handshake, a secure process where devices exchange messages to verify their identities and establish a session key used for encrypting data transmissions. For confidentiality/privacy, WPA retains the RC4 encryption algorithm but with important changes. It contains an increased IV size of 48 bits, which is significantly larger than WEP's 24-bit IV. This larger size offers a greater number of variations, making key reuse attacks considerably more challenging. For encryption, the Temporal Key Integrity Protocol (TKIP) dynamically generates a new encryption key for each data packet transmitted [2]. For message integrity, WPA utilizes the Michael algorithm, a more robust method for message integrity compared to WEP's CRC. Michael incorporates several key security features including a stronger hash function, larger checksum size, sequence number, and frame counter. Lastly for access control, WPA leverages EAP mentioned earlier, which supports per-user authentication, enabling network administrators to define access levels for different users or devices. This allows for more granular control over network resources and functionalities.

Key shortcomings of WPA include its reliance on the RC4 algorithm, which is now considered less secure than newer options like AES. The Pre-Shared Key (PSK) mode, while convenient for home users, can be susceptible to brute-force attacks if weak passphrases are used.

WPA marked a significant improvement over WEP, but advancements in technology and the ever-evolving threat landscape led to the development of more sophisticated security protocols. WPA2, introduced in 2004, adopted the Advanced Encryption Standard (AES) algorithm, offering stronger encryption compared to RC4 used in WPA [3].

For user/message authentication, WPA2 uses the Advanced Encryption Standard (AES) along with EAP to authenticate users onto the network [3]. For confidentiality/privacy, Counter Mode with Cipher Block Chaining Message Authentication Code Protocol (CCMP) is used to encrypt data packets. It combines a 48-bit initialization vector with a nonce and MAC address to generate a key for each packet sent [4]. For message integrity, the MIC mechanism first introduced in WPA is used to maintain message integrity [3]. Lastly for access control, passphrase based authentication is used.

WPA3 was introduced in 2018 in order to fix some of the security vulnerabilities of WPA2. It also introduces more effective and efficient encryption algorithms for protecting data [3]. WPA3 uses the Simultaneous

Authentication of Equals (SAE) protocol for authenticating users when connecting to the network. This protocol uses the Dragonfly handshake to implement a zero-knowledge proof between the user and access point [3]. For confidentiality/privacy, WPA3 uses the Galois/Counter Mode Protection to perform 256-bit encryption on the packets [5]. For message integrity, each device on a WPA3 network has its own encryption key so that other devices cannot decrypt packets being transmitted [3]. Lastly for access control, client isolation is implemented in WPA3 to prevent clients on a network from directly communicating or even knowing about each other [3].

As technology continues to evolve, so too will the need for robust WiFi security protocols. Understanding the vulnerabilities of WEP and the advancements introduced by WPA and subsequent protocols empowers users to make informed decisions about securing their wireless networks. By implementing the latest security protocols, employing strong passwords or passphrases, and keeping software updated, users can create a more secure wireless environment and protect their data from unauthorized access.

III State-of-the Art Solution Approaches

WEP, WPA, WPA2, and WPA3 all approach the encryption process differently, with each iteration of security introducing an added layer of protection. This section will breakdown how each security protocol attempts to keep user's data safe over the air. A qualitative analysis on the effectiveness of each algorithm employed along with a synopsis of the key vulnerabilities found in each security protocol is provided here.

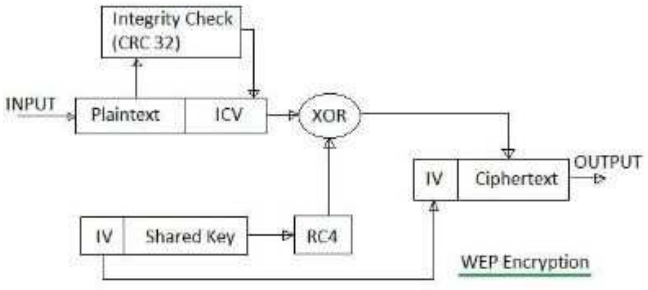

Fig. 1: WEP Encryption [1]

WEP Encryption works as follows:

1. First, a cipher key called the WEP Key is prepared. It is a character string or a hexadecimal number value.

2. The Initialization Vector of 24-bits is calculated with a PseudoRandom Number Generator

3. The WEP Key and Initialization Vector are then put into the Rivest Cypher 4.

4. An Integrity Check Value is calculated from plain text by using the 32-bit Cyclic Redundancy Check

5. A cipher text is obtained by an XOR operation of the output values of RC4 and the concatenation values of the ICV and the plain text.

WEP Decryption works as follows:

1. The received Initialization Vector and the WEP key are input to the Rivest Cypher once more.

2. The cipher text and output values of RC4 are input to an XOR operation.

3. The output values of XOR are the plain text and the ICV.

WEP has many vulnerabilites. WEP's user authentication relies on a single shared key. If an attacker eavesdrops on network traffic, they can capture this shared key, granting them unauthorized access and compromising the entire network's security. While WEP employs the RC4 algorithm for encryption, the initialization vector (IV) is only 24 bits long. This limited size allows for key reuse since there are not many variations possible, and the same key can be used even with slightly different IVs. By analyzing patterns in this encrypted data, attackers can exploit this repetition and decrypt communications, rendering the confidentiality WEP meaningless. WEP's message integrity relies on the weak checksum algorithm Cyclic Redundancy Check (CRC) which can be easily manipulated by attackers, allowing them to modify data packets without detection.

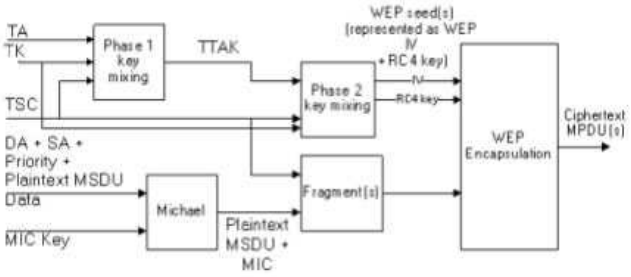

Fig. 2: TKIP Encryption [2]

WPA TKIP Encryption works as follows:

1. The TKIP Message Integrity Code or MIC is computed and appended to the data field

2. TKIP assigns the incremented TKIP Sequence Counter or TSC value to the frame

3. The encryption key is generated using the two phase key mixing function

4. The encryption key, the plain text data and the appended MIC are passed to the WEP engine

5. WEP generates an ICV, computed over the plaintext and MIC and appends this to the plain text, after the MIC.

6. The frame is encrypted by the WEP engine, and the ciphertext is sent.

WPA TKIP Decryption works as follows:

1. TKIP extracts the TSC. If it is out of order the frame is discarded.

2. The WEP key is generated using the two phase key mixing function, and is represented as a WEP IV and an RC4 key. These are passed to the WEP engine to decrypt the frame.

3. WEP checks the Integrity Check Value or ICV. If it is valid, the MIC is computed over the frame and compared with the MIC in the message. If they are different then the frame is dropped and countermeasures are triggered. Otherwise the frame is passed onto the upper layer.

The Michael algorithm is used by WPA for message integrity. This keyed hash function used generates a more complex and unique value (checksum) based on the data packet, making it significantly more resistant to manipulation than the CRC algorithm used by WEP. WPA employs a 64-bit checksum compared to WEP's 32-bit value, offering greater protection against potential attacks. The sequence number and framer counter help prevent replay attacks, where attackers attempt to resend captured data packets to gain unauthorized access.

Despite being an improvement upon WEP, WPA still has vulnerabilities. Poor passwords in WPA-PSK mode are vulnerable to brute force attacks or Dictionary attacks which attempt to break into the network by inputting a series of passwords using wordlists which contain a series of commonly used passwords. WPA's core encryption, TKIP, relies heavily on the RC4 algorithm inherited from WEP. This introduces a potential weakness, as attackers could exploit the key weaknesses found in RC4 itself. TKIP utilizes a relatively small initialization vector (similar to WEP), which while larger than WEP's, could still be susceptible to key reuse attacks with enough effort. Algorithms can recover the TK and MIC key with a few WEP keys derived from the same Initialization Vector.

The 4-way handshake used by WPA2 begins by generating the PMK from the PSK, SSID, and a Hash Message Authentication Protocol (HMAC). After this, the client requests to join the network and the AP responds with an acknowledgement and a random value known as a nonce. The client uses the nonce to calculate a PTK value and then generates its own nonce value that it sends along with a MIC value generated from the PTK. The AP then generates its own MIC value using the same process as the client and then confirms if the two MIC values match [3]. Lastly, the AP sends an installation request for the client to install the encryption and integrity keys and the client responds with a confirmation after it has done this [6].

The AES-CCMP encryption used by WPA2 works by taking the PTK, headers, packet number, and the MAC address of the device and sending them all through an AES encryption algorithm. The output of this algorithm is then XORed with the data packet before being sent across the wireless channel. AES works as a block cipher that performs many rounds of permutations and substitutions to encrypt data [4].

One vulnerability with WPA2 is the dictionary attack. This attack works by first capturing the 4-way handshake between a user and the AP. An attacker is then able to run through a large list of passwords and confirm if one of them matches the password used by the captured handshake or not. In the event that a password matches, an attacker is then able to connect to the network and take advantage of other vulnerabilities [3].

Another vulnerability is the Key Reinstallation Attack (KRACK). This attack works by first setting up a rogue access point so that the client connects to the attacker instead of the AP. The attacker then forwards data to and from the AP. During the 4-way handshake, the attacker is able to replay message 3 of the 4-way handshake. By capturing multiple responses to message 3, the attacker is then able to derive the key and decrypt network traffic [3].

		802.11i Methods	
	WEP	WPA	WPA2
Security Protocol	WEP	TKIP	CCMP
Cipher	RC4	RC4	AES
Key Length	40 or 104 bits	128 bits encryption, 64 bits authentication	128bits
Key Life	24 bit IV	48 bit IV	
Key Generation	Concatenation	Two phase mixing function	Not needed
Data Integrity	CRC-32	Michael	CBC-MAC
Header Integrity	None	Michael	CBC-MAC
Replay Protection	None	Packet Number	
Key Management	None	EAP-based	
Authentication	Open or Shared Key	802.1x or Pre-Shared Key (PSK)	

Fig. 3: Comparison of WiFi security protocols [2]

The dragonfly handshake protocol used in SAE authentication for WPA3 begins by deriving a PE value using the network password and elliptic curve parameters p and q where both parameter values are large prime numbers [7]. Each party in the handshake then generates random values r and m which are then used

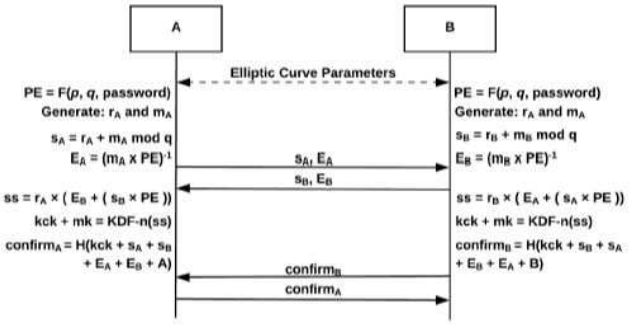

Fig. 4: Dragonfly handshake [3]

along with the PE to generate a scalar value (s) and an element value (E). Each party then exchanges their respective s and E value and uses the received values to derive a shared secret value (ss). The ss is then used by each party to calculate a confirmation key (kck) and master key (mk). The kck is used along with the previous s and E values to calculate a confirmation by each party. The parties then exchange these values to make sure both sides have the same ss value. After this confirmation, the mk value is used as the PMK in the following 4-way handshake previously described for WPA2. The reason that this protocol is so effective is that the usage of dot product operations makes it so that an m value cannot be computed by knowing the PE and E values sent between users [3].

AES-GCMP for WPA3 uses the AES algorithm previously discussed for the encryption of data. It also uses GCMP to check message integrity which acts as a faster and more efficient version of CCMP, allowing for higher throughput [5].

WPA3 addresses the dictionary attack from WPA2 by using the dragonfly handshake so that passwords cannot be guessed without direct communication to the AP. It also addresses the KRACK exploit by making it so multiple transmissions of message 3 from the 4-way handshake can be disabled [3].

One vulnerability that WPA3 is susceptible to is the rogue access point attack. By having an attacker setup a fake AP that pretends to be the real wireless AP, they can trick users into connecting to their AP. At this point, they can then prompt the user to input the network password, allowing them to gain access to the network [3].

After gaining the password, an attacker can then perform an evil twin attack by setting up a new rogue access point. Once a user connects to this AP, the attacker is able to connect to the real AP and forward data between the user and AP while also being able to decrypt all traffic being sent [3].

Another attack that can be performed after gaining the network password is an ARP spoofing attack where the attacker is able to connect to the network and send ARP messages to the AP in order to disconnect another user from the network. They can do this repeatedly to perform a denial of service attack against the user [3].

IV Case Study: Implementation and Evaluation

A. The Studied Techniques

The three main security threats against WPA2 are de-authentication attacks, dictionary attacks, and rogue access points. This section will provide a description of these threats and address two techniques introduced in WPA3 intended to mitigate these threats: Simultaneous Authentication of Equals and Protected Management Frames.

A de-authentication (or disconnect) attack allows the attacker to disconnect a client from its AP. In a WiFi network, Management Frames are used to manage client connection. Management frames to be sent without encryption in WPA2 and may be easily forged by an attacker [8]. An attacker may spoof a client MAC address and send a de-authentication frame to the AP to disconnect the client [3].

A dictionary attack allows the attacker to acquire encryption keys after capturing a handshake during client authentication. After capturing the 4-way handshake, the attacker will repeat key derivation processes over a list of candidate passwords. For each candidate password, the attacker will derive its corresponding Pre-Shared Key (PSK). The availability of information such as client and AP MAC addresses, the network SSID, and the random nonces captured in the 4-way handshake allow the attacker to derive all connection keys and Message Integrity Code (MIC) corresponding to the candidate password. By comparing the MIC derived from the candidate password and the MIC captured in the handshake, the attacker can determine whether or not the candidate password was correct [3]. Upon determining the correct password, the attacker may decrypt all traffic on that connection and other connections whose handshakes were captured. The attacker may also use the password to establish a connection with the AP. The use of de-authentication attack improves the efficiency and scope of the dictionary attack as it allows the attacker to force a client to reconnect to the AP in order to capture its handshake [8].

A rogue access point allows the attacker to employ various methods to acquire connection keys and decrypt traffic. The attacker may set up an access point using the same SSID as a genuine AP to trick clients into

connecting to it. The phishing technique is an example of key acquisition in which connected clients are sent to a landing page where they are prompted for the network password [3]. The attacker may send an unencrypted Channel Switch Announcements (CSA) management frame to cause clients to route traffic through their rogue access point [8].

In WPA3, two techniques are introduced to mitigate these threats. The Simultaneous Authentication of Equals (SAE) protocol implements the Dragonfly Handshake as described in Section III, offering protection against offline dictionary attacks and providing forward secrecy to client connections [3]. The independent generation and evaluation of key computational elements during the Dragonfly Handshake restricts the attacker from determining connection keys. The vulnerabilities of unencrypted management frames such as de-authentication and CSA were resolved by the implementation of Protected Management Frames (PMF). An AP employing PMFs will prompt for an encrypted response upon receipt of an unencrypted management frame, preventing the attacker from performing such actions in a non-keyed state [3].

B. Evaluation and Analysis

This section will demonstrate a dictionary attack on a WPA2 network and the implementation of a rogue access point. Both demonstrations will be performed on a virtual machine running Kali Linux. Kali Linux is an open source Linux distribution designed to assess WiFi security and equipped with network tools such as Aircrack-ng [9]. The setup for these demonstrations consists of a Windows laptop running Kali Linux on a virtual machine, a WiFi router running a network with WPA2-PSK (AES) encryption, and a compatible WiFi USB adapter.

1) Dictionary Attack Demonstration

A dictionary attack can be easily implemented on a WPA2 network using the Wifite auditing tool which automates many network attack processes [10]. Install Wifite on Kali Linux with the following command.

```
sudo apt install wifite
```

Run the Wifite program. A list of discovered networks will begin to populate as shown in Figure 5. Press Ctrl+C to stop the search when the target network ESSID (network name) appears in the list. Enter the list number corresponding to the target network when prompted.

```
sudo wifite
```

Wifite will execute a series of attacks on the target network. The WPA Handshake capture will implement a dictionary attack by running aircrack-ng tools on captured handshakes for every candidate password in a wordlist. The attack may use a handshake captured during execution by de-authenticating connected clients (Figure 6) or a handshake saved from a previous attack. The attacker may edit the wordlist by changing the wordlist file path or directly editing the default wordlist text file. Upon discovery of the correct network PSK, Wifite will display the network password and finish execution.

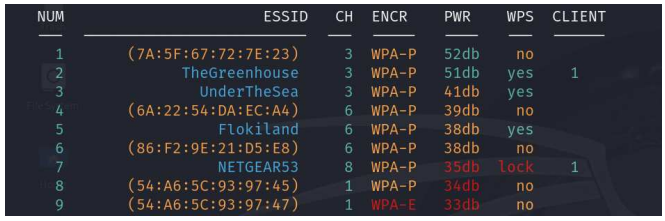

Fig. 5: Discovered networks list following Wifite program start

Fig. 6: Wifite WPA Handshake Capture using De-Authentication

2) Rogue Access Point Demonstration

A rogue access point may be implemented using a WiFi USB adapter with Ad-Hoc mode, meaning it is capable of acting as an access point. Use the airbase-ng command to establish a new access point on the WiFi adapter (wlan0) and to create a tap interface (at0) for the virtual machine. Specify a channel number with the -c flag.

```
sudo airbase-ng --essid
    name_of_rogue_AP -c 11 wlan0
```

Confirm the established access point by using the following command.

```
sudo airodump-ng wlan0
```

To provide access for network packets to enter the virtual machine, a virtual bridge must be established and linked between the tap interface and the ethernet interface (eth0). Install the virtual bridge utilities and establish the virtual bridge link with the following commands.

```
sudo apt-get install bridge-utils
```

```
sudo brctl addbr name_of_bridge

sudo brctl addif
    name_of_bridge at0

sudo brctl addif
    name_of_bridge eth0
```

Enable the tap and ethernet interfaces and enable IP forwarding to make the access point accessible to external clients. Ensure that the ethernet interface is configured with an IP address within the same subdomain as the genuine access point.

```
sudo ifconfig at0 0.0.0.0 up

sudo ifconfig eth0
    IPv4_address up

sudo sysctl -w
    net/ipv4/ip_forward=1
```

The rogue access point with the attacker-specified network name is now accessible to nearby devices. From this point, a variety of threats as discussed in the previous sections may be employed against a nearby network and connected clients. Figure 7 shows the airbase-ng print out from the access point establishing command after a client connection. Figure 8 shows packet sniffing on the new client connection.

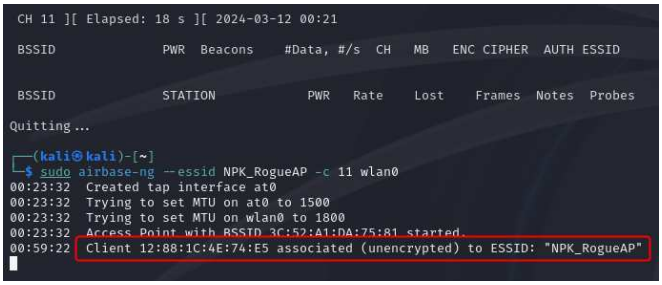

Fig. 7: Client Connects to Rogue Access Point

3) *Analysis of Attack Demonstrations and WPA3 Techniques*

To analyze the effectiveness of SAE and PMF, this section will refer to the research on the feasibility of the two demonstrated threats against WPA3 as well as conclusions to be drawn from the attack implementations.

In the first demonstration implementation, executing a dictionary attack with a wordlist of 200000 candidate passwords takes approximately 90 seconds to complete. It is difficult to assess the time it takes to successfully

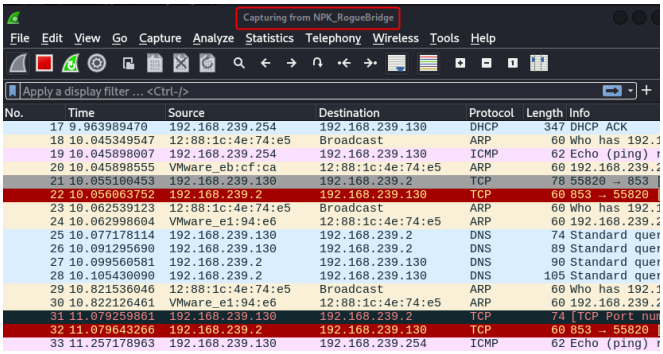

Fig. 8: Client Connections Packets in Wireshark

crack a network password in WPA2 and WPA3 due to the many wide ranging factors such as password complexity and attacker computing power; however, it may be concluded from the research that performing a dictionary attack on WPA3 with the same level of threat as shown in the demonstration is infeasible for two reasons: key encryption complexity and forward secrecy. The computational intractibility of deriving particular key elements from a password and handshake make SAE much more complex than the 4-way handshake alone [3]. The 4-way handshake utilizes the PSK and random nonces transmitted over the air to generate connection keys. However, the random elements generated in SAE are never directly transmitted over the air, but are rather derived by client and AP independently. Thus, even if the attacker were to obtain the network password and derive the encryption keys for one connection, they would not have the ability to decrypt traffic captured from previous connections since those connections would have their own independently-derived random elements [3]. Additionally, the use of PMFs would mitigate the ability of the attacker to capture handshakes to begin with since the attacker cannot de-authenticate connected clients and would have to wait for new client connections.

Unlike the dictionary attack, rogue access points remain feasible, though less effective, against SAE and PMF techniques in WPA3. Nothing inherent to SAE encryption prevents an attacker from deploying a rogue access point and phishing unsuspecting clients for the network password. However, as previously discussed, PMFs mitigate the effectiveness of certain rogue access point techniques such as the use of CSAs to force clients into connection.

Overall, the implementation of SAE and PMF severly limits the potential threats against a WPA3 network as opposed to WPA2. The cost- and time-efficient dictionary attack method demonstrated in this section is

infeasible against a WPA3 network. While rogue access points remain as a point of attack, the scope of its threat has been mitigated by the PMF technique.

V Conclusion

Overall, by performing research and implementations on WiFi network security, it was found that clear improvements were made with each iteration of the WiFi security protocols. WEP was found to be the least secure due to how it only uses one encryption key for all devices which made it easy for attackers to figure out these keys. WPA improved upon this protocol by using TKIP to generate new encryption keys instead of using a static key like in WEP. WPA also introduced message integrity checking so that it could be determined if the packets had been modified. WPA2 then improved upon WPA by replacing the TKIP algorithm with AES-CCMP for a more complex encryption. This was achieved through hardware improvements that allowed the new encryption method to be done efficiently. Lastly, WPA3 improves upon WPA2 by introducing PMFs and the SAE protocol for authentication which uses the dragonfly handshake to create a more secure handshake between the user and AP, preventing vulnerabilities such as dictionary attacks and the KRACK attack.

One challenge faced in this project was implementing attacks and assessing security mechanisms on a WPA3 network. There are not as many accessible tools for WPA3 penetration testing compared to WPA2. Despite this, the rogue access point demonstration in the case study serves as an implementation of a security threat that is common to both WPA2 and WPA3.

Some future work recommendations for those who wish to perform further research on WiFi network security would be to research how rogue access point attacks can be prevented since those attacks are the main entry point for attackers on a WPA3 WiFi connection. Along with this, more research should be done on possible security vulnerabilities of WPA3 so that all of the vulnerabilities of the protocol can be identified and fixed in a possible future encryption protocol.

VI Group Member Contributions

For the Survey presentation, Asa presented information describing WEP, WPA, and WPA2 processing, including methods of encryption, level of security, and the varying security protocols each employs. The Implementation presentation had Asa perform a dictionary attack using Kali Linux on his home wifi Network. He created a template for the presentation and helped populate the WPA vulnerability slides. For the final Report Asa wrote the Abstract, Introduction, and the Background, Fundamental Concepts, and State of the Art Solution approaches for WEP and WPA.

Alex worked on the conclusion and the WPA2 and WPA3 portions of the background and state of the art solutions sections for this report. He also did the WPA3 description, vulnerabilities, and improvements over WPA2 for the survey presentation. Lastly, for the implementation presentation, he reviewed WPA3 vulnerabilities, created a python script to demonstrate a WPA3 man-in-the-middle attack, and presented that script.

Nicholas developed the Case Study sections including the study of SAE and PMF techniques, the demonstrations for the dictionary attack and rogue access point establishment, as well as the analysis on the effectiveness of SAE and PMF against the demonstrated threats. Nicholas demonstrated the rogue access point setup in the implementation presentation and discusses several security threats against WEP, WPA, and WPA2 in the survey presentation.

References

[1] A. Kan, "Attacking wpa enterprise wireless network," Dec 2016. [Online]. Available: https://pentest.blog/attacking-wpa-enterprise-wireless-network/

[2] U. of Canterbury, "The performance of the ieee 802.11i security specification on wireless lans," 2005. [Online]. Available: https://www.csse.canterbury.ac.nz/research/reports/HonsReps/2005/hons_0505.pdf

[3] C. P. Kohlios and T. Hayajneh, "A comprehensive attack flow model and security analysis for wi-fi and wpa3," 2018. [Online]. Available: https://doi.org/10.3390/electronics7110284

[4] R. Awati, "Counter mode with cipher block chaining message authentication code protocol (ccmp)." [Online]. Available: https://www.techtarget.com/searchsecurity/definition/CCMP-Counter-Mode-with-Cipher-Block-Chaining-Message-Authentication-C

[5] A. Gillis, "Wpa3." [Online]. Available: https://www.techtarget.com/searchsecurity/definition/WPA3

[6] "Authentication types for wireless devices," 2008. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/routers/access/wireless/software/guide/SecurityAuthenticationTypes.html

[7] D. Harkins, "Dragonfly key exchange," 2015. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc7664#section-2.1

[8] P. Ebbecke, "Protected management frames enhance wi-fi network security." [Online]. Available: https://www.wi-fi.org/beacon/philipp-ebbecke/protected-management-frames-enhance-wi-fi-network-security

[9] g0tm1lk, "What is kali linux," Nov 2023. [Online]. Available: https://www.kali.org/docs/introduction/what-is-kali-linux/

[10] kimocoder, "Wifite homepage," Mar 2024. [Online]. Available: https://github.com/kimocoder/wifite2