Wireless Security Basics and Concepts

BOUSSETA HATIM

Understanding Wi-Fi, Bluetooth, Encryption, and How to Stay Safe

Introduction to Wireless Networks

Wireless networks allow devices to communicate without physical cables using **radio waves**.

This project helps test how secure wireless networks are by safely hacking them (with permission!). Just like doctors run tests to check your health, we run security tests to find weak spots in Wi-Fi networks before real hackers can exploit them.

Why This Matters?

- Wireless tech has come a long way since Marconi's first radio in 1896 - While we have security systems (like passwords and encryption), they need regular checkups - More Wi-Fi = More convenience... but also more risks (like digital pickpockets!)

How We Test?

1.We pretend to be hackers (but the good kind!) using special tools: - Kali Linux (digital Swiss Army knife for security testing) - ESP32 MARAUDER (a clever little device that can test Wi-Fi security) - PWNAGOTCHI (a Raspberry Pi-powered tool that learns about network weaknesses) - Wireshark (like a microscope for examining network traffic)

2. We carefully document all the vulnerabilities we find .
3. We help fix them to make networks safer .

Common Wireless Technologies

- **Wi-Fi (IEEE 802.11)**: For internet access in homes and offices
- **Bluetooth**: Short-range connections (headphones, keyboards)
- **Cellular (4G/5G)**: Mobile internet over long distances

Wi-Fi Standards Evolution

What is IEEE 802.11? IEEE 802.11 is the official name for Wi-Fi standards, created in 1997 to make wireless networking possible. Instead of using cables, it lets devices communicate using radio waves. Over time, it has improved in speed, range, and security through different versions like 802.11b, 802.11g, 802.11n, and more.

Wi-Fi has come a long way since its early days. The first version was slow (only 2 Mbps), but new versions kept improving: - 802.11b (1999): 2.4 GHz, 11 Mbps speed, good range. - 802.11a and 802.11g: Faster speeds, better performance. - 802.11n: The most widely used version for years. - 802.11ac and 802.11ad: Super-fast speeds and better reliability.

Table 1: Wi-Fi Standards Comparison

Standard	Year	Speed	Freq.	Key Feature
802.11b	1999	11 Mbps	2.4 GHz	First widely used
802.11a	1999	54 Mbps	5 GHz	Less interference
802.11n	2009	600 Mbps	2.4/5 GHz	MIMO antennas
802.11ac	2013	1.3 Gbps	5 GHz	Fast streaming
802.11ax	2019	10 Gbps	2.4/5/6 GHz	Wi-Fi 6

Wi-Fi Frequency Bands

Wi-Fi operates in several frequency bands:

- **2.4 GHz**: Used by 802.11b/g/n, has 14 channels, but many overlap.
- **5 GHz**: Used by 802.11a/n/ac, offers more channels and less interference.
- **Other Bands**:
  - **3.6 GHz / 4.9 GHz**: Used for public safety (mostly in the U.S.).
  - **60 GHz**: Used by 802.11ad for high-speed short-range communication.
  - **Below 1 GHz**: Used by 802.11af/ah for long-range IoT applications.

2.4 GHz Channels:

- Only channels 1, 6, and 11 are non-overlapping.
- Some channels are restricted in certain regions (e.g., Japan allows Channel 14 only for 802.11b).

5 GHz Channels:

- Provides 25 non-overlapping channels of 20 MHz each.
- Requires Transmit Power Control (TPC) and Dynamic Frequency Selection (DFS) to avoid interference with radar/military applications.

Regional Regulations:

- **EU (EN 301 893, 2005/513/EC)**: Regulates 5 GHz band use.
- **Germany & Austria**: Follow strict laws on 5.250-5.725 GHz usage.
- **U.S. (FCC)**: Enforces TPC/DFS to prevent interference with radar systems.

Emerging Wi-Fi Standards:

- **802.11y**: Uses 3.6 GHz for public safety.

- **802.11p (WAVE)**: Introduced for vehicular networks but rarely implemented.
- **802.11ad**: Uses 60 GHz for very high-speed short-range connections.
- **802.11ah**: Uses sub-GHz bands (900 MHz) for IoT and long-range applications.

NOTE : A Wi-Fi channel refers to a specific frequency range used for transmitting data over a wireless network. In simpler terms, it's like a specific lane on a highway that devices use to send and receive information. Multiple devices (such as laptops, phones, and access points) can use the same channel to communicate, but interference can occur if too many devices share the same or overlapping channels.

802.11 Frame Structure

The 802.11 frame structure is essential for transmitting data over wireless networks. The frame is divided into various parts, each with a specific function to ensure proper transmission and reception of data. Below is an overview of the key components within the 802.11 frame:

- **Frame Control (2 bytes)**:
  - **Purpose**: Manages the transmission process of the entire packet over the network.
  - **Subfields**: This field contains control bits that specify the frame's type, whether it is part of a sequence, if it has a retry, and other key control settings.
- **Duration/ID (2 bytes)**:
  - **Purpose**: Primarily used to define how long the channel should remain reserved for the current transmission, assisting in managing access to the channel.
  - **Secondary Role**: It may also serve to identify certain network characteristics such as power-saving features or device identification.
- **Address Fields (up to 6 bytes each)**:
  - **Purpose**: Stores the MAC addresses of devices involved in the transmission.
  - **Address 1**: The destination address (e.g., the access point's MAC address).
  - **Address 2**: The source address (e.g., the transmitter's MAC address).
  - **Address 3**: The receiver address (if applicable).
  - **Address 4**: An optional address used in complex frames like those used for bridging or mesh networks.
- **Sequence Control (2 bytes)**:

  - **Purpose**: Helps manage the ordering and fragmentation of frames.
  - **Sequence Number**: Used to order frames correctly when they are sent in sequence.
  - **Fragment Number**: If a frame is too large, it is split into fragments, and this field tracks the fragment numbers.

- **Frame Body (up to 2312 bytes)**:
  - **Purpose**: Contains the actual data being transmitted. The size can vary depending on the amount of data.

- **FCS (Frame Check Sequence, 4 bytes)**:
  - **Purpose**: Provides error checking through a Cyclic Redundancy Check (CRC). It ensures that the transmitted data has not been corrupted and can be validated at the receiving end.

Summary of Frame Structure

The 802.11 frame consists of several key parts that ensure efficient and accurate data transmission:

- **Frame Control**: Manages the packet's transmission process.
- **Duration/ID**: Sets the channel reservation time and can assist with power management.
- **Address Fields**: Store the MAC addresses of the involved devices.
- **Sequence Control**: Organizes frames and handles fragmentation.
- **Frame Body**: Contains the data being transmitted.
- **FCS**: Provides a checksum for error detection.

These fields ensure that data is transmitted efficiently, correctly, and with minimal interference in a wireless network.

Key Aspects of Secure Wireless Communication

Wireless networking is a rapidly evolving field, but ensuring security and reliability remains a priority. Here are the three fundamental aspects of secure communication:

- **Integrity:** Ensures that data remains unaltered during transmission, safeguarding it from tampering.

- **Confidentiality:** Protects data from unauthorized access or exposure, ensuring privacy.
- **Availability:** Guarantees that data and services are accessible whenever needed, ensuring reliability.

Common Wireless Security Threats

Wireless networks are prone to several types of security threats, which can be broadly categorized as follows:

- **Interception:** Unauthorized access to data during transmission, often by eavesdropping on the signal.
- **Interruption:** Disruption or jamming of communication between devices, leading to service denial.
- **Modification:** Altering data during transmission, which can compromise its integrity.
- **Fabrication:** Injecting false or malicious data into the network, potentially causing harm or misdirection.

Wi-Fi Specific Security Threats

In addition to general threats, wireless networks face specific risks that can directly impact the integrity and performance of Wi-Fi systems:

- **Wireless Phishing:** Fraudulent attempts to steal sensitive information by mimicking legitimate Wi-Fi networks (fake hotspots).
- **Evil Twin APs:** Malicious access points designed to look like trusted ones, tricking users into connecting and exposing their data.
- **Data Interception:** Unauthorized capture of data transmitted over the wireless network, often by man-in-the-middle attacks.
- **Denial of Service (DoS):** An attack that makes a network or service unavailable to legitimate users by overwhelming it with excessive requests.
- **Rogue Access Points (APs):** Unauthorized devices installed within the network, potentially bypassing security mechanisms.
- **Misbehaving Clients:** Devices that don't function as expected, which can create security vulnerabilities or degrade network performance.

- **Ad Hocs and Soft APs:** Unapproved peer-to-peer networks or misconfigured access points that can be exploited for attacks.
- **Wireless Intruders:** External attackers attempting to access the network without permission.
- **Misconfigured Access Points:** Incorrectly set-up APs that may leave the network vulnerable to exploits.

Security Threats

Security threats to wireless networks can be categorized into several types. These include **interception**, which involves unauthorized access to data during transmission; **modification**, which refers to altering transmitted data; **interruption**, which is the disruption of communication between devices; and **fabrication**, which involves injecting false data into the network. These threats pose serious risks to the integrity and reliability of wireless communication, making it crucial to implement strong security measures to prevent unauthorized access and ensure the safe transmission of data.

Encryption Techniques

Due to these threats, encryption plays a vital role in securing data during transmission. Modern encryption can be divided into two types:

- **Symmetric Key Algorithm:** The same key is used for both encryption and decryption. It is fast and efficient but requires secure key exchange.
- **Public Key Algorithm:** Uses a pair of keys (public and private) for encryption and decryption. It provides higher security but is slower than symmetric encryption.

With the increasing sophistication of attacks, strong encryption mechanisms have become essential to protect wireless communications and ensure the privacy and security of transmitted data.

Encryption Process

The encryption process:

- $E_k : M \to C$: This notation means that the encryption algorithm E (using the key k) takes the message M and turns it into ciphertext C. The k here represents the secret key used in the encryption process.

- $m \rightarrow E(k, m)$; for every $k \in K$: This part is saying that for every key k in the key space K, the message m is encrypted using the encryption algorithm E with that key. Here, m is the message being encrypted, and $E(k, m)$ means that the algorithm E uses the key k to encrypt the message m.

In simple terms:

1. You have a message (M) that you want to protect.
2. You use an encryption key (K) with an algorithm (E) to encrypt it.
3. The result is ciphertext (C).

The part "for every $k \in K$" just emphasizes that the process applies to all keys in the key space K, but in practical symmetric encryption, K is a single secret key that both the sender and receiver know and use.

Stream and Block Ciphers

Stream Ciphers: It encrypts every bit of the message as a single output one at a time.
Block Ciphers: It combines a number of messages at a time and encrypts them as one unit before.

Public Key Algorithm (RSA Encryption)

The Public Key Algorithm uses two separate cryptographic keys:

- **Public Key:** Used to encrypt plain text or verify digital certifications.
- **Private Key:** Used to decrypt cipher text or create digital certifications.

The two keys are mathematically related, but they are not the same.

Encryption and Decryption Process:

Encryption: Converts Plain Text (P) into Cipher Text (C) using the Public Key (K):

$$C \rightarrow K[P]$$

Decryption: Converts Cipher Text (C) back into Plain Text (P) using the Private Key (K_1):

$$P \rightarrow K_1[C]$$

Example:

If someone wants to send a secure message, they use the public key to encrypt it. Only the person with the private key can decrypt it and read the message.

RSA encryption is a well-known example of the Public Key Algorithm, developed in 1977 by Rivest, Shamir, and Adleman. It is widely used for secure communication over the internet.

Key Takeaways:

- Public Key Algorithm uses two keys: public for encryption, private for decryption.
- RSA is one of the most popular public key encryption methods.

Example with RSA (Simplified):

1. In RSA, the public key consists of two numbers: e (public exponent) and n (modulus), which are derived from two large prime numbers. 2. The private key consists of another number d, which is calculated using e and n.

The relationship is such that:

Encryption:

$$C = M^e \mod n \quad \text{(where } M \text{ is the message, and } C \text{ is the ciphertext)}$$

Decryption:

$$M = C^d \mod n \quad \text{(where } C \text{ is the ciphertext, and } M \text{ is the original message)}$$

The key point is that, using the properties of modular arithmetic, the public key and private key are inverses of each other. So, while e is used for encryption, d is the inverse operation that will decrypt the message correctly.

Why Is This Relationship Important?

- **Security:** Even though the public key is widely distributed, only the private key can decrypt messages encrypted with the public key, ensuring that only the intended recipient can read the message.
- **Authentication:** The private key can also be used to sign messages, and anyone with the public key can verify the signature. This helps ensure the authenticity of the sender.

Wired Equivalent Privacy (WEP)

WEP was introduced in 1997 as part of the IEEE 802.11 standard to provide security for wireless networks. The goal was to offer data confidentiality comparable to that of wired networks. However, WEP is now considered insecure due to various weaknesses. Here's a simplified explanation of how WEP works and its limitations:

How WEP Works:

- **WEP Key:** WEP uses a key (a sequence of hexadecimal digits) for encryption. The length of the key can vary:
  - 64-bit WEP (10-digit key)

  - 128-bit WEP (26-digit key)
  - 256-bit WEP (58-digit key)

- **Initialization Vector (IV):** A random number used to initialize the encryption process. It's combined with the WEP key to form a unique encryption key for each packet of data. The IV is short (only 24 bits), which is one of the reasons WEP is vulnerable.
- **RC4 Encryption:** WEP uses the RC4 encryption algorithm, which combines the WEP key and IV to generate a key stream. This key stream is then combined with the plaintext message to create ciphertext, which is transmitted over the network. The message is decrypted by the receiver using the same key and IV.

Security Issues with WEP:

While WEP was intended to provide encryption for wireless networks, several vulnerabilities were discovered over time that made it insecure:

- **Short Initialization Vector (IV):** The IV is only 24 bits long, which means it's repeated frequently, allowing attackers to easily predict the key stream and break the encryption.
- **Weak Encryption Protocol:** The RC4 algorithm used by WEP is weak by modern standards, and attackers can exploit its flaws to recover the encryption key.
- **Weak RC4 Algorithm Implementation:** Even though RC4 was initially thought to be secure, its poor implementation in WEP made it vulnerable to attacks like key recovery.
- **Short / Shared Keys:** WEP relies on a shared key for encryption, meaning both the sender and receiver use the same key. This leads to key reuse, which further compromises security.
- **No Key Management:** WEP does not have a system for regularly updating encryption keys. Once the key is known, it can be used to decrypt all subsequent messages, making it easy for attackers to crack.
- **Possibility of Message Modifications:** WEP lacks mechanisms to verify the integrity of the transmitted messages. This means an attacker can modify the message without detection.
- **Negative User Authentication:** WEP does not provide strong mechanisms for authenticating users. Anyone with the WEP key can connect to the network, allowing unauthorized access.
- **Eavesdropping:** Since WEP keys can be easily cracked, attackers can eavesdrop on the network traffic, gaining access to sensitive information.

Conclusion:

WEP was once a standard encryption protocol for wireless networks but is now outdated and insecure due to its weak encryption, poor key management, and other vulnerabilities. It has been largely replaced by more secure protocols like WPA2 (Wi-Fi Protected Access). Understanding WPA and WPA2 Security Protocols

1 Introduction

Wi-Fi Protected Access (WPA) and its successor, WPA2, are security protocols designed to secure wireless networks. They provide mechanisms for protecting data transmissions between wireless clients (e.g., laptops or smartphones) and access points (APs). WPA and WPA2 employ the same basic concepts but differ in encryption algorithms, with WPA2 offering stronger security.

2 WPA and WPA2 Key Generation Process

The key idea behind WPA and WPA2 is to generate a session key that encrypts data during communication. This session key is derived from a shared secret, known as the Pre-Shared Key (PSK), which is a password known to both the access point and the client.

2.1 Pre-Shared Key (PSK) and Pairwise Master Key (PMK)

The process starts by creating the Pairwise Master Key (PMK) from the PSK and the SSID (network name). The PSK is combined with the SSID using a hash function (usually PBKDF2) to generate the PMK:

$$\text{PMK} = \text{PBKDF2}(\text{PSK}, \text{SSID}, 4096, 256)$$

The PMK is not transmitted in the clear and is used to derive other keys, such as the Pairwise Transient Key (PTK), used in data encryption and integrity.

2.2 4-Way Handshake

WPA and WPA2 employ a 4-way handshake between the client and the access point to establish a secure communication channel. The handshake is used to generate the PTK, which will be used to encrypt data during the session.

The four messages exchanged during the handshake are:

- **Message 1:** The AP sends a nonce (ANonce) to the client.
- **Message 2:** The client responds with its own nonce (SNonce) and the MIC (Message Integrity Code).

- **Message 3:** The AP sends the GTK (Group Temporal Key) and confirms the client's response.
- **Message 4:** The client acknowledges the completion of the handshake.

During this handshake, the client and AP exchange nonces (random values) and MAC addresses. These values are later used to generate the PTK.

2.3 Pairwise Transient Key (PTK) Generation

The PTK is generated using the following formula:

$$\text{PTK} = \text{PRF-512}(\text{PSK}, \min(\text{AP MAC}, \text{Client MAC}), \max(\text{AP MAC}, \text{Client MAC}), \text{ANonce}, \text{SNonce})$$

Where:

- **PRF-512:** A Pseudo-Random Function (PRF) that generates a 512-bit key.
- **PSK:** The password (Pre-Shared Key) used to generate the PMK.
- **AP MAC and Client MAC:** The MAC addresses of the access point and the client.
- **ANonce and SNonce:** Random values generated by the access point and client, respectively, during the handshake.

The PTK is used to encrypt data frames and ensure message integrity between the client and the access point.

2.4 Cracking the PSK: Brute-Force and Dictionary Attacks

Although the PSK is not directly transmitted, it can be cracked through offline brute-force or dictionary attacks using the 4-way handshake data. The captured handshake contains information such as the AP MAC, Client MAC, ANonce, SNonce, and the MIC.

To crack the PSK, tools like `aircrack-ng` or `hashcat` use the following approach:

1. **Capture the 4-Way Handshake:** The attacker captures the handshake during the client's connection to the AP.
2. **Brute-Force the PSK:** The attacker tries multiple PSK guesses and applies the PRF-512 function to generate the PTK.
3. **Check the MIC:** The attacker uses the generated PTK to calculate the MIC and compares it to the MIC in the captured handshake. If they match, the correct PSK has been found.
4. **Stop When Correct PSK is Found:** The process stops once the correct PSK is identified.

3 Differences Between WPA and WPA2

While both WPA and WPA2 use similar processes, the main difference lies in the encryption method used:

- **WPA:** Uses TKIP (Temporal Key Integrity Protocol), which is considered less secure.
- **WPA2:** Uses AES (Advanced Encryption Standard), a more secure and modern encryption standard.

WPA2 is the recommended protocol as it provides better security.

4 Security Considerations and Mitigation

The security of WPA and WPA2 depends on the strength of the PSK. Weak PSKs (e.g., dictionary words or short passwords) are vulnerable to brute-force attacks. To mitigate this risk:

- Use long and complex PSKs (20+ characters, including a mix of letters, numbers, and symbols).
- Switch to WPA3, which uses a more secure key exchange protocol (SAE) and is resistant to offline dictionary attacks.

5 Conclusion

WPA and WPA2 provide secure wireless communication by generating session keys from the PSK and using those keys to encrypt data. The 4-way handshake ensures that the keys are unique for each session. However, attackers can still crack weak PSKs using brute-force or dictionary attacks. WPA2 offers stronger security than WPA, and WPA3 is the most secure option available today.

6 Bluetooth

- **Frequency Range:** Bluetooth operates in the 2.4 GHz to 2.485 GHz range, which is a crowded frequency band. However, Bluetooth devices avoid interference by rapidly hopping between 1600 different channels every second.
- **Range:** Typically, Bluetooth works within a range of 10 meters. However, with special antennas, it can extend up to 100 meters.
- **Pairing Process:**
  - Devices exchange a PIN or use a pairing process with random numbers.

- After pairing, an authentication key and a link key are created to secure communication.
- An encryption key is also created to ensure the data shared between devices is encrypted and secure.

7 How Bluetooth Works

1. **Pairing:** When two Bluetooth devices (e.g., a phone and a headset) want to connect, they go through a pairing process. The devices exchange a PIN code (or use other methods like random numbers) to authenticate each other.
2. **Piconet:** Once paired, the devices form a small network called a Piconet. The Piconet includes one master device and up to seven slave devices.
3. **Data Encryption:** After establishing a secure link, the devices can start sharing data. The data is encrypted using the encryption key to prevent eavesdropping.

8 Bluetooth Security Issues

Bluetooth faces several security challenges:

- **BlueSnarfing:** Attackers can steal information such as contacts and messages from a Bluetooth device.
- **BlueBugging:** Attackers can take control of a device, make calls, and listen to conversations.
- **Blueprinting:** Attackers can track Bluetooth devices based on their unique MAC address.
- **BlueSmack:** Attackers can knock devices offline by sending disruptive packets.

9 Security Measures

Bluetooth uses encryption and authentication during the pairing process to keep data secure. However, like all wireless technologies, it's vulnerable if not used properly. To improve security:

- Turn off Bluetooth when not in use.
- Use strong pairing methods and avoid using default PIN codes.

10 Conclusion

This is a high-level overview of Bluetooth communication, its pairing process, and the security challenges it faces. While Bluetooth provides a convenient means of wireless communication, it's important to understand its security implications and take necessary precautions.