Comenius University, Bratislava

Faculty of Mathematics, Physics and Informatics

Security of Wi-Fi Networks

Bachelor Thesis

Ľubomír Žák, 2012

Comenius University, Bratislava

Faculty of Mathematics, Physics and Informatics

Security of Wi-Fi Networks

Bachelor Thesis

Study program: Informatics

Branch of Study: 2508 Informatics

Department: Department of Computer Science

Advisor: RNDr. Peter Gaži, PhD.

Bratislava, 2012

Ľubomír Žák

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Ľubomír Žák
Študijný program: informatika (Jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor: 9.2.1. informatika
Typ záverečnej práce: bakalárska
Jazyk záverečnej práce: anglický

Názov: Security in Wi-Fi networks

Cieľ: The Wi-Fi technology based on the IEEE 802.11 family of standards is currently the most widespread wireless networking mechanism for personal computers. The goal of this thesis is to summarize existing means of securing Wi-Fi networks and to analyze the guarantees they offer. The thesis should survey all known attacks against secured Wi-Fi networks and the freely available tools implementing these attacks. It should also contain a summary of the author's own experiments with these tools or an independent implementation of a selected attack.

Vedúci: Mgr. Peter Gaži, PhD.

Dátum zadania: 25.10.2011

Dátum schválenia: 26.10.2011

doc. RNDr. Daniel Olejár, PhD.
garant študijného programu

študent

vedúci

I hereby declare I wrote this thesis by myself, only with the help of referenced literature, under the careful supervision of my thesis supervisor.

. . . .

I would like to thank my supervisor RNDr. Peter Gaži, PhD. for his great help, advices and supervising.

Abstrakt

Autor: Ľubomír Žák
Názov práce: Bezpecnost bezdrotových sietí WiFi
Škola: Univerzita Komenského v Bratislave
Fakulta: Fakulta Matematiky, Fyziky a Informatiky
Katedra: Katedra informatiky
Vedúci bakalárskej práce: RNDr. Peter Gaži, PhD.
Rozsah: 51 strán
Jún 2012, Bratislava

Cieľom tejto bakalárskej práce je podrobne popísať jednotlivé možnosti zabezpečenia bezdrôtových sietí a ich konfigurácie so zameraním na bezpečnostný štandard WPS. Práca je rozdelená do 5 kapitol. Úvodná kapitola popisuje vznik technológie bezdrôtových sietí a ich výhody. Druhá kapitola pojednáva o jednotlivých druhoch útokov na bezdrôtové siete. V tretej kapitole sa nachádza podrobný chronologický prehľad možností zabezpečenia bezdrôtovej siete. Štvrtá kapitola sa zameriava na štandard WPS a popisuje môžné riešenia nedostatkov implementácie tohto štandardu. V záverečnej kapitole je popísaná experimentálna čast tejto bakalárskej práce, ktorá zahŕňa popis útokov na štandard WPS v rôznych podmienkach a prezentuje z nej vyvodené závery.

KĽÚČOVÉ SLOVÁ: Wi-Fi, bezpečnosť, WPS, útok hrubou silou

Abstract

Author: Ľubomír Žák
Thesis title: Security of Wi-Fi Networks
University: Comenius University, Bratislava
Faculty: Faculty of Mathematics, Physics and Informatics
Department: Department of Computer Science
Advisor: RNDr. Peter Gaži, PhD.
Thesis length: 51 pages
June 2012, Bratislava

The main goal of this bachelor thesis is to provide an overview of wireless networks security options and their configuration with a focus on WPS standard. The thesis is divided into 5 chapters. The first chapter describes the 802.11 standard for wireless LANs and advantages of using WLAN. Next chapter provides an overview of attacks on WLANs in general. In the third chapter we give a chronological preview of approaches used for securing WLANs. Fourth chapter is focusing on the WPS standard and describes possible fixes of implementation flaws of this standard. In the last chapter there is an experimental part of the thesis, which includes description of an attack on WPS PIN in different conditions and presentation of the results.

KEYWORDS: Wi-Fi, security, WPS, brute force attack

Contents

List of Figures

Intro

Wireless LAN technology has rapidly become very popular all over the world. Thanks to the release of the IEEE 802.11 wireless LAN standard, wireless technology has transformed to an open solution for providing mobility and network services without the requirement of wired connection.

The wireless local area network (WLAN) protocol, IEEE 802.11, and associated technologies enable secure access to a network infrastructure. Until the development of WLAN, the network client needed to be physically connected to the network by using some kind of wiring.

With the rapid increase in use of WLAN technology it is important to provide a secure communication over wireless network. Since its creation the security of wireless networks went trough different stages of development, from MAC address filtering or WEP to WPA/WPA2.

The wireless technology was proven to be very practical (not only) for home users: Such a handy option to be comfortably connected to internet on a mobile device without the need of wires is still gaining in popularity. This led to an attempt to make a configuration of WLAN easier for regular user without any knowledge about computer science. The result of this was standard known as Wifi Protected Setup (WPS).

WPS, as a standardized technology, is implemented on wide variety of currently produced wireless access points. The incorrect designing of its standard led to fatal weakness which is discussed in this thesis in greater details. Furthermore, the thesis explains how the weakness can be exploited and provide some results from testing such an attack in different conditions.

1

WLAN Basics

The 802.11 specification as a standard for wireless local area networks (WLANs) was ratified by the Institute of Electrical and Electronics Engineers (IEEE) in 1997. It is a member of the IEEE 802 family, which is a series of specifications for LANs. The base specification for WLANs includes the 802.11 MAC and two physical layers, where the MAC layer is responsible for how to access the medium and send data while the physical layer works with transmission and reception of waves.

The 802.11 networks consist of four major components:[1]

- **Distribution system** - a logical component used to forward frames to their destination.
- **Access points (APs)** - devices performing wireless-to-wired bridging function.
- **Stations (STAs)** - device with wireless network interface communicating with other similar devices via APs.
- **Wireless medium** - medium used to transfer frames from station to station. Two radio frequency and one infrared physical layers were standardized.

The list of main benefits of using WLAN include: [2],[3]

- Mobility
- Easy installation in difficult-to-wire areas
- Reduced installation time

- Cost stability
- Scalability
- Range of coverage

Mobility

Mobility is a significant advantage of WLANs. Any user is able to access shared resources without the need of wired connection, anywhere in the organization. A wireless network allows users to be truly mobile as long as the mobile device is under the network coverage area.

Easy installation in difficult-to-wire areas

Implementation of WLANs can significantly decrease the cost of network installation. For example, connecting two networks in two different buildings separated by street, river, railroad etc. Furthermore, WLANs installation will not do any damage to historical buildings, since it requires neither cables nor drilling holes in walls.

Reduced installation time

In a wireless network, the only cable that needs to be installed in order to make the WLAN work is a cable connecting the wireless AP with the local network. There is no need to physically connect STAs to AP, so the time needed for installation works is greatly reduced. WLANs can even be often installed without professional help.

Cost stability

To install a new device in WLAN network, administrator does not need to buy additional cabling equipment to connect device to the network. Furthermore, it is possible to connect as many STAs as needed while still using only one AP.

Scalability

WLAN system can be configured in a variety of topologies and configuration to match the particular requirements. It can also be easily altered if needed and adding more APs into it can easily expand the network.

Range of coverage

Since most of the WLANs use radio frequencies, it is possible to access an AP even through walls or other obstacles. The range of a typical WLAN node is about 100 m. Coverage can be extended via roaming. If the coverage of APs overlap each other, the user can wander around and move from one coverage area to another while still being connected, without noticing any disturbance.

2

Attacks on WLAN

There are many security threats and attacks that can damage the security of WLANs. Those attacks can be basically classified into two categories: [4]

- Logical attacks
- Physical attacks

2.1 Logical Attacks

A logical attack is always related with the software, system and the sensitive data flowing in the network. The main goal of these attacks is to find and abuse the sensitive data flowing in the network. Some of the most common logical attacks are defined below:

- Brute Force Attacks Against Access Point Passwords
- Attacks against encryption
- MAC address spoofing
- Man in the Middle Attack
- Reconnaissance Attacks
- Dynamic Host Configuration Protocol Attack

Brute Force Attacks Against Access Point Keys

Most WLANs use a pre shared key or password that is used by all connected STAs. Brute force dictionary attacks attempt to recover this key by progressively testing each of the possible passwords. Once the key has been successfully recovered, an attacker gains access to the network. The example of the dictionary attack against WPA/WPA2 and various attacks on WEP are listed in Chapter 3.

Attacks against encryption

Attacker can use the known weakness in the way the data are being encrypted in the communication between the STAs and APs to recover various data or network password. We'll discuss used encryption standards and their weaknesses in chapter 3 in greater details.

MAC address spoofing

MAC address spoofing is a malicious technique used for gaining access to the wireless networks protected by MAC address filtering method. Since MAC addresses are transmitted unencrypted in 802.11 headers, an attacker can obtain MAC address of the authorized station by passively listening to the WLAN communication. Once the MAC address was obtained, the attacker can easily use it to gain access to the network.

Man in the Middle Attack

Man-in-the-middle attacks is an attack in which an attacker is able to read, modify and inject messages between two parties while neither of them knows that the communication has been attacked.

Using techniques like IEEE802.1x to achieve mutual authentications between APs and STAs as well as adopting an intelligent wireless Intrusion Detection System can help in preventing such attacks. Enforcing WEP or WPA across the wireless network is also a common solution to this problem.[5]

Reconnaissance Attacks

Reconnaissance attacks are used to gather information about a target network or system. Although such an attack may look basically harmless and is often overlooked, it is usually the information gained through reconnaissance attacks that is used in subsequent Access or DoS attacks.

Dynamic Host Configuration Protocol Attack

”DHCP starvation attack works by broadcasting DHCP requests with spoofed MAC addresses.”[6] If enough requests are sent, the address space available to the DHCP servers can get exhausted for a period of time. Subsequently, a legitimate user is denied an IP address requested via DHCP which results in dysfunctional network. DHCP starvation is usually a denial of service (DoS) attack. Additionally it may be used in conjunction with a malicious rogue server attack to redirect traffic to a malicious computer ready to intercept traffic. [7]

2.2 Physical Attacks

Rogue Access Points

Rogue access points are WLAN access points that are not authorized to connect to a target network. Rogue APs open a wireless hole into the network. An attacker can plant a rogue AP, or an employee may unknowingly create a security hole by plugging a non secure access point into the network. Any rogue AP can be used by anyone who can connect to the AP, including an attacker, giving them access to the wired

LAN.[8]

Physical placement of AP

The installation location of APs is another security issue because placing APs inappropriately will expose it to physical attacks. If an attacker can physically access the AP then he is able to switch the AP to its default settings- which is (in most cases) insecure. Therefore it is very important for network security administrators to carefully choose spots for placing the APs.[4]

Jamming attacks

Jamming attack is performed by transmitting a signal to the receiving antenna at the same frequency band or sub-band as the communications transmitter transmits. An attacker with the right tools and knowledge can easily jam the 2.4 GHz frequency in a way that drops the signal to a level where the wireless network can no longer function.[9]

3
Security in WLAN

In this chapter we will describe security/encryption methods used in wireless networks with their flaws and strong sides. We will also provide a detailed look on some of the possible attacks against them.

3.1 Weakest Security Mechanisms

Among the most commonly used security mechanisms to protect WLAN while them being no obstruction at all for an even unexperienced attacker are SSID hiding and MAC address filtering.

3.1.1 SSID Hiding

Many APs offer user an option to hide the SSID. If it is enabled, the AP in its beacon frames does not show the SSID - an empty string is shown instead. Although it looks like a good idea (if no one sees the WLAN it can not be attacked), it is not helpful at all.

Association request frame contains SSID in plaintext form. Also probe request and response frames which are being sent while the station is detecting the network do contain SSID.

Attacker can therefore perform one of the two possible attacks:

1. *passive* - an attacker silently monitors the traffic and waits until one of the station will try to associate with the AP or is searching for the targeted SSID
2. *active* - an attacker sends fake deauthentication or disassociation packet to any of the connected station while monitoring the traffic. The station needs to associate/authenticate itself again which leads to frame with plain SSID in it.

3.1.2 MAC Address Filtering

Like SSID hiding, MAC address filtering is also commonly used ”security” mechanism. Although it is better to use even weak protection than none at all, MAC address filtering can be easily broken by using MAC address spoofing technique described in Chapter 2, section MAC address spoofing.

3.2 WEP

Wired Equivalent Privacy (WEP) is a security algorithm for IEEE 802.11 wireless networks. It was a first attempt to provide security to wireless networks, ratified in September 1999. Even though its security weaknesses are well known and almost 13 years passed since its ratification, WEP is still widely used as a common option to secure communication on network.

3.2.1 Encryption Details

In a WEP protected network, all packets are encrypted using the stream cipher RC4 under the root key. This key is shared by all stations and a recovery of this key gives attacker full access to the network.

RC4 is a stream cipher, symmetric key algorithm. The same algorithm is used for both encryption and decryption as the data stream is simply XORed with the generated key sequence. The key stream is completely independent of the plain text used. It uses a variable length key from 1 to 256 bit to initialize a 256-bit state table. The state table is used for subsequent generation of pseudo-random bits and then to generate a pseudo-random stream which is XORed with the plain text to give the cipher text.

The algorithm can be broken into two stages: initialization, and operation. In the initialization stage the 256-bit state table, S is populated, using the key, K as a seed. Once the state table is setup, it continues to be modified in a regular pattern as data is encrypted. The initialization process can be summarized by the pseudo-code below: [10]

```
$j \leftarrow 0$
**for** $i \leftarrow 0$ to 255 **do**
    $S[i] \leftarrow i$
**end for**
**for** $i \leftarrow 0$ to 255 **do**
    $j \leftarrow (j + S[i] + K[i]) \bmod 256$
    Swap $S[i]$ and $S[j]$
**end for**
```

Once the initialization process is completed, the operation process may be summarized as shown by the pseudo-code below:

```
$j = i = 0$
**for** $k \leftarrow 0$ to $N - 1$ **do**
    $i \leftarrow (i+1) \bmod 256$
    $j \leftarrow (j+S[i]) \bmod 256$
    Swap $S[i]$ and $S[j]$
    $pr = S[(S[i] + S[j]) \bmod 256]$
    output $M[k]$ XOR $pr$
**end for**
```

The algorithm produces a stream of pseudo-random values. The input stream is XORed with these values. If the algorithm is fed with the encrypted data, it will produce the decrypted message output, and if it is fed with the plaintext message, the output will be encrypted data.

3.2.2 WEP Encryption

For each packet, a 24-bit initialization vector (IV) is chosen. The IV concatenated with the root key yields the per packet key. The CRC-32 checksum (Integrity Checksum Value- ICV) is calculated over the data to be encrypted. The per packet key is then used to encrypt the data followed by the ICV using RC4 stream cipher. The (unencrypted) IV is transmitted in the header of the packet. Figure 3.1 shows simplified version of frame.

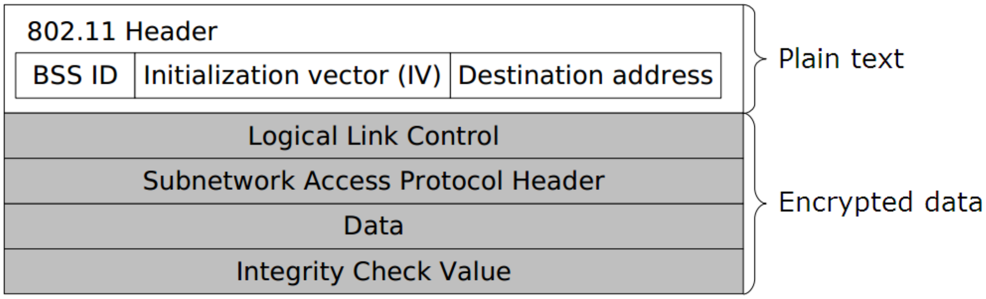


Figure 3.1: Simplified WEP Frame

The following figure illustrates the WEP encryption process.

3.2.3 WEP Decryption

The initialization vector (IV) is unencrypted in the header. The IV is appended to the root key. The combination of IV and the root key is used as an input for the pseudo-random number generator (PRNG) to generate a bit sequence. This sequence is XORed with the encrypted data plus ICV to decrypt the data. The ICV calculation then is run. If the value matches the value of ICV in the incoming frame, the data is considered to be valid.

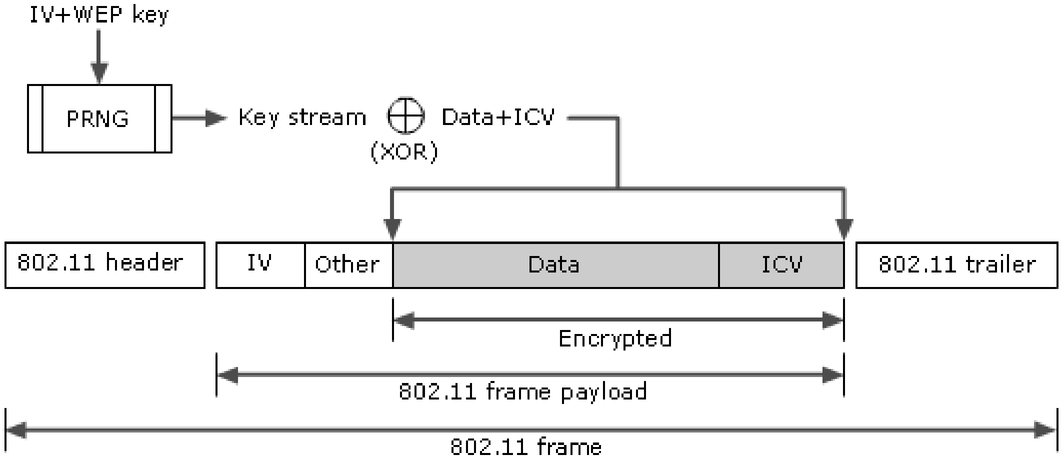


Figure 3.2: WEP Encryption

3.2.4 Flaws in WEP

WEP is considered to be a weak security technique to protect WLAN. The reason why WEP failed was using RC4 stream cipher for both the authentication and privacy functions. Even though RC4 is a very strong encryption algorithm, in WEP it was not implemented correctly. ”RC4 explicitly warns never to use the same key material twice, no matter what the payload is, since it is simply an XOR stream cipher.” [11]

Another fatal flaw is the 24-bit initialization vector (IV) that is transmitted clear in every packet. Two simple changes would have made a big difference in WEP’s security possibilities: using longer IV and also using another hash algorithm instead of CRC-32 for checksum.

Also WEP does not provide a defense mechanism against replay attacks. An attacker can passively listen to the network traffic and use recorded packets later on to derive information about the encryption key. Summing up, it has been proven both mathematically and practically that WEP is insecure. Nonetheless, it is still better than using no protection at all. [4]

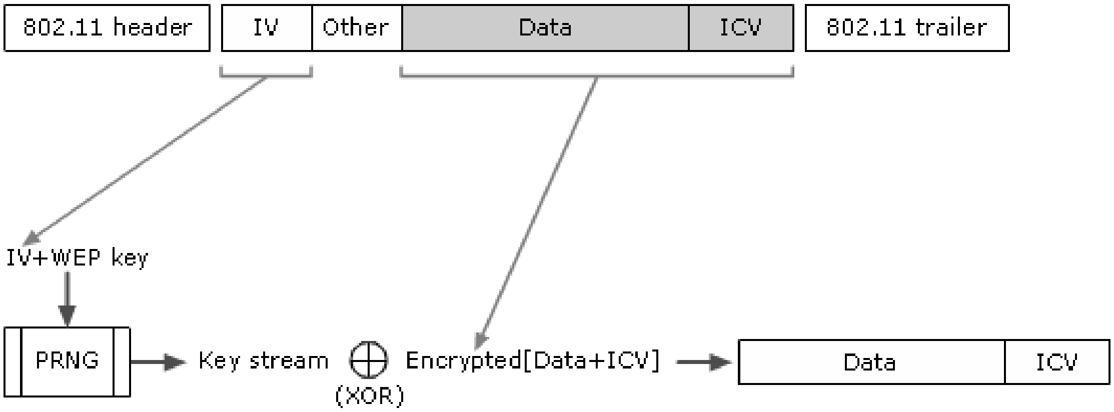


Figure 3.3: WEP Decryption

3.2.5 Attacks against WEP

FMS Attack

The first attack on WEP protocol was called the FMS attack. It was named after Fluhrer, Margin and Shamir who published it an article describing the weakness in 2001. The paper describes several “weak” IVs which have “a format of B+3::ff:X (where B is the byte of the key to be found, ff is the constant 255, and X is irrelevant).” [12] By using the knowledge of the plaintext in the headers of certain packets, such as ARPs, the attacker can determine the value of B. [13] To achieve a success probability of at least 50%, 4,000,000 to 6,000,000 packets needs to be captured. This number depends on the exact environment and implementation. [14]

The KoreK Attack

The user under the name KoreK posted this famous attack on the internet in 2004 by releasing an advanced WEP cracking tool. This implementation used 17 different attacks of which some were already known but most of them were were found by KoreK himself. The number of captured packets needed to achieve 50% success rate is reduced to about 700,000. [15]

The PTW Attack

The newest and most powerful attack on WEP is called the PTW attack, named after its creators Pyshkin, Tews, and Weinmann. The attack was released in 2007 and is much more efficient than all the other attacks because it uses every captured packet. The PTW attack is based on Klein's attack on the RC4 stream cipher, which was released in 2005. Basically, the PTW attack picks a set of number of likely keys and continues the RC4 algorithm on those instead of trying all possible combinations. The PTW attack needs approximately 70,000 packets only to achieve a 97% success rate. [13] [15]

The Chopchop Attack

The Chopchop attack is not a key recovery attack. Instead, it tries to obtain the information of a plaintext from given cipher text. The Chopchop attack focuses on a property of CRC-32 and "allows the attacker to interactively decrypt the last m bytes of the plain text of an encrypted packet by sending m*128 packets in average to the network." [14]

3.3 WPA

The Wifi Alliance (WFA) provided a new security protocol in 2003 in order to replace the weak WEP security. Fearing mass obsolescence and the backlash from consumers feeling they were being forced to buy new equipment only a short time after investing in it to achieve security, WPA was designed to operate on the limited hardware resources of APs designed for WEP. That is also a reason why WPA still uses RC4 as its stream cipher (while the WPA2 uses Advanced Encryption Standard - AES - block cipher in Counter Mode as its stream cipher). [16] The CRC-32 algorithm for checksum calculation was replaced with stronger Micheal algorithm.

The basic principle of WPA could be simplified as follows: transfer the data decrypted by Temporary Key Integrity Protocol (TKIP) before somebody can decrypt the key. While WEP was using single pre-shared key for all encryption, WPA changes the unicast encryption key for every frame and each change is synchronized between the wireless client and the wireless AP. For the global encryption key, WPA includes a facility for the wireless AP to advertise changes to the connected wireless clients.

WPA features two different operation modes:

- WPA-PSK (Pre-Shared Key) mode
- WPA Enterprise mode

In WPA-PSK, to access the network user needs to know the SSID of the network and the WPA key. The AP itself handles the authentication and contains secret key that both the client and the AP use to set up the secure connection. WPA uses 256-bit hexadecimal key (in opposite to 128-bit WEP key) generated with the PBKDF2 algorithm. The PBKDF2 algorithm generates a key called the Pairwise Master Key (PMK) that is then used to authenticate client devices to the network. The PBKDF2 basically works like this: the PMK is generated from the passphrase supplied by user (8 to 63 characters long), the SSID value and the length of SSID. All these values together are run trough SHA1 algorithm (4096 iterations), which produces the 256-bit value used as a handshake key.

This system provides a user a great benefit: all he does need to remember is a simple passphrase instead of some hexadecimal values. Although, it also has its drawbacks: If the attacker knows the SSID and the SSID length, he has several parts needed to recover the passphrase. Attacker can recover hashed version of the key from captured packet. In theory, he can run a dictionary brute-force attack where he inserts the SSID, SSID length and generated passphrase into PBKFD2 algorithm and compares the output with the captured hashed passphrase. Matching values would mean the attacker guessed the passphrase right. Since the passphrase is 8-63 characters long with 94 possible options per each (ASCII characters 32 to 126), it leads to very large number of possibilities (range 94^8 - 94^{63}). Also considering the fact that SHA1 is very CPU intensive, the brute force search could take thousand of years.

But now the human factor can help the attacker: most of the users use dictionary word passwords which greatly reduces the searching spectrum from thousands of trillions to few hundred thousand words. The dictionary attack was first implemented by Josh Wright in his tool coWPAtty. The improved version of the tool was later released by a wireless research group the Church of Wi-Fi with the following idea: if you run the attack against an SSID once, why not store the list of resulting test hashes to use again later if the same SSID was encountered. The result was the addition of genpmk program to coWPAtty to generate hash lookup tables.

In late 2009, Moxie Marlinspike launched wpacracker.com, the service using Amazon's EC2 cloud computing service to allocate processing power much cheaper compared to what it would cost to build it on our own. For $35, the service will run a dictionary attack using a 135 million word dictionary with results being sent back in less than 20 minutes (there is also possibility to use extended dictionary containing up to 520 million words).

To sum it up, the ability to crack WPA is based on two things: the quality and size of the dictionary and the amount of time that the attacker is willing to invest. It is worth to note that if the user uses passphrase which has nothing to do with dictionary words (combination of uppercase/lowercase letters, numbers and a few symbols), there is no chance of recovering the passphrase. The same concept of dictionary attacks applies also to WPA2-PSK passphrases. [16]

The WPA enterprise mode requires a RADIUS server to be used. The authentication server is connected to a wired local area network. [17] It is a component used for authentication and key management. In order to use the network, user needs to log in by entering username and password. After successful log in, user receives unique encryption key which is regularly updated. The enterprise mode obviously provides two important benefits over the PSK mode:

Increased key security

The keys are being regularly changed so it's nearly impossible to crack the actual encryption key.

Password-based authentication

The encryption key or the passphrase is not stored on the computer. These are derived securely in the background of the server. This helps when the user password is stolen, since changing username and corresponding password is easier than reconfiguring the whole network with new passphrase.

The following table shows how TKIP and Michael address the cryptographic weaknesses of WEP [18]. We'll discuss these in greater detail in the following subsection.

WEP Weakness	WPA solution
Short IV	The length of IV was doubled to 48 bits in TKIP
Weak data integrity	Michael algorithm replaced the CRC-32 algorithm used for checksum calculation. Michael calculates 64-bit message integrity code (MIC), which is further encrypted with TKIP.
Master key over derived key	TKIP uses a set of keys derived from the master key (the pre-shared key). Additionally, part of the input for the RC4 PRNG is changed for each frame.
No rekeying	WPA rekeys automatically to derive new key per each packet.
No replay protection	TKIP uses IV as a counter, therefore provides a protection against replay attacks.

3.3.1 Temporary Key Integrity Protocol (TKIP)

Temporary Key Integrity Protocol (TKIP) is a part of WPA certification. It addresses WEP weaknesses and provides solution for secure data encryption.

One of the flaws in WEP was too short IV. In WPA, the extra 32 bits are added to the original 24. However, in practice only 48 bits is used because one byte is thrown away to avoid weak keys. While adding 24 bits does not sound like much, it practically solves the IV rollover problem. For example, let's imagine the device sending 10000 packets per second. This could be done for example by using 64-byte packets at 11Mbps. While the 24-bit IV would roll over in less than half hour, the 48-bit one would not roll over for over 950 years. [19]

The per-packet key generation process consists of two phases and utilizes several inputs, such as the transmitting device MAC address, the 32 bits of the IV already mentioned, the first 16 bits of the IV, and the temporal session key. The first phase involves mixing the temporal session key, 32 IV bits, and the transmitter's MAC. In the second phase the output of the first phase is mixed with the temporal session key and 16 bits of the IV. Phase 1 eliminates the use of the same key by all connections, and the second phase reduces the correlation between the IV and per-packet key. Note that the key mixing results in different keys for each direction of communications over each link. [20]

Another improvement of the IV in TKIP is using it as a sequence counter. The TKIP IV, on the contrary to pseudo random IVs used in WEP, is incremented sequentially for each out-of-sequence IV packet discarded. This mitigates the replay attacks but also decreases the service quality. Since ACKing every received frame is highly inefficient, the improved called burst-ACK is used: a series of 16 frames is ACKed instead of ACKing each. In case that one of the frames was lost, selective ACKing is applied. A TKIP sequence number counter would reject such a frame if it received a frame with higher IV already- that's why TKIP employs a replay window which keeps track of the last 16 IV values received. If the received IV fits into the table and it wasn't received already, it is valid. [20]

TKIP uses the Michael algorithm for Message Integrity Code (MIC) caluclation instead of insecure WEP ICV computation. Michael was designed solely for WPA by a Dutch cryptographic engineer Niels Ferguson. It generates 64-bit MIC using a 64-bit secret key. Half of the Data MIC key is used as the 64-bit secret key for authenticating messages sent from the AP to the STA, and the other half is being used as the 64-bit secret key for authenticating messages sent from the STA to the AP. [17]

3.3.2 WPA Encryption

WPA needs following values in order to encrypt a wireless data frame:

- Initialization vector (IV)
- Data encryption key
- Source and destination addresses (SA, DA)
- Priority field value
- Data integrity key

The following figure 3.4 illustrates the simplified WPA encryption process:

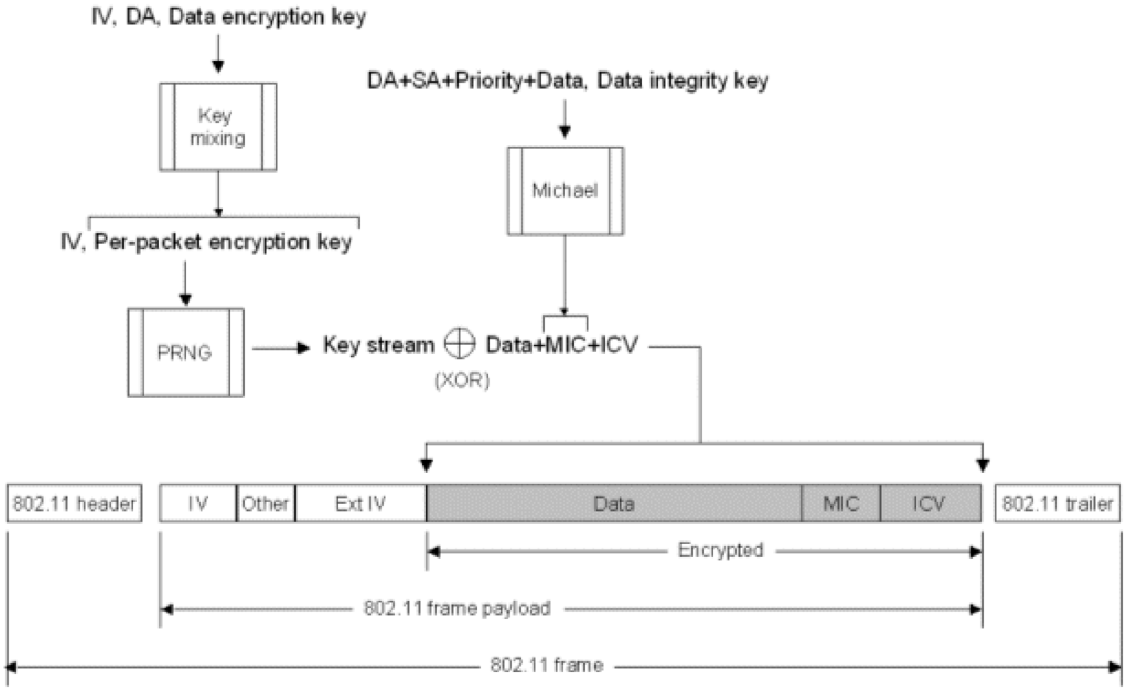


Figure 3.4: WPA Encryption Process for an Unicast Data Frame

In words, the IV, DA and the Data encryption key are used as an input into the WPA key mixing function, which outputs the per-packet encryption key. The DA, SA, Priority, the data (the unencrypted 802.11 payload), and the data integrity key are input into the Michael data integrity algorithm, which produces the MIC. The ICV is calculated from the CRC-32 checksum. The IV and per-packet encryption key are used as the input for the RC4 PRNG function, which produces the key stream of the same size as the data, MIC and ICV. To produce the encrypted portion of payload, the key stream is XORed with the data, MIC and ICV. Finally, the IV is added to the IV and Extended IV fields.

3.3.3 WPA Decryption

The WPA decryption process can be described as follows: The IV is extracted from the IV and extended IV fields. The IV, DA and the Data encryption key are used as the input for the key mixing function to produce the per-packet key. IV and per-packet key are used as the input for RC4 PRNG function to generate key stream of the same size as the encrypted data, MIC and ICV. The key stream is XORed with the encrypted data, MIC and ICV to produce unencrypted ICV, MIC and the data. The ICV is calculated and compared with the value of unencrypted ICV. If they do not match, the data is discarded. The DA, SA, Priority, Data and Data integrity key are input for the Michael algorithm, which produces the MIC. Calculated MIC value is then compared with the unencrypted MIC value of the incoming frame. If the MIC values do not match, the data is discarded. Otherwise the data is valid.

The following figure 3.5 illustrates the simplified WPA decryption process:

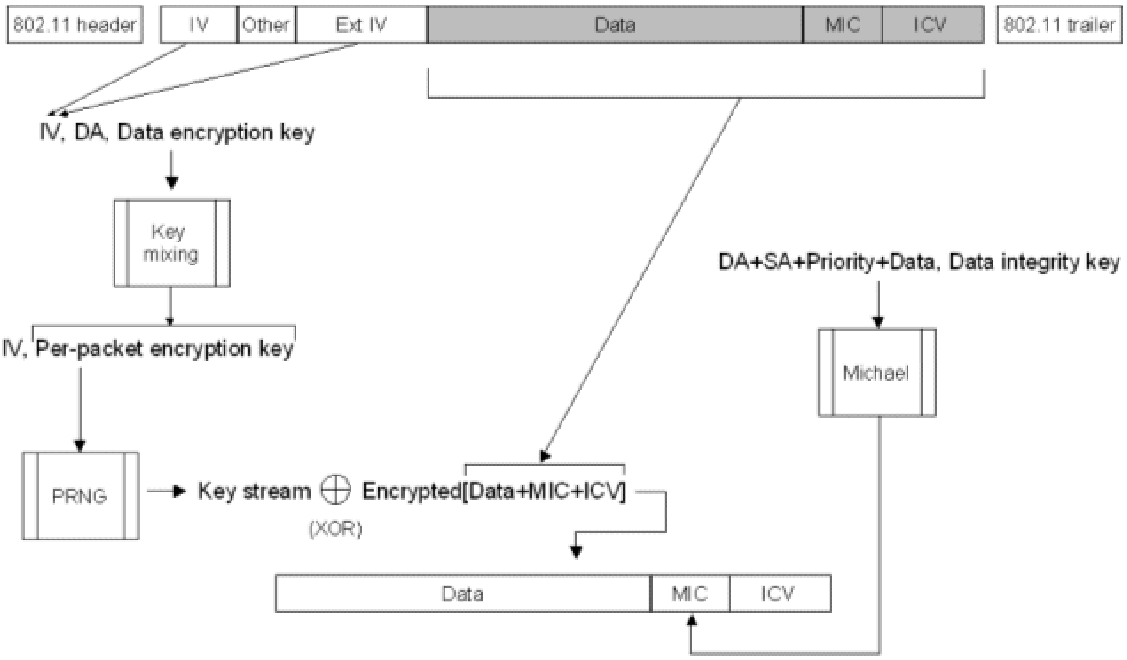

802.11 header
IV
Other
Ext IV
Data
MIC
ICV
802.11 trailer
IV, DA, Data encryption key
Key mixing
IV, Per-packet encryption key
DA+SA+Priority+Data, Data integrity key
Michael
PRNG
Key stream ⊕ Encrypted[Data+MIC+ICV]
(XOR)
Data
MIC
ICV

Figure 3.5: WPA Decryption Process for an Unicast Data Frame

3.3.4 Attacks on WPA

The dictionary attack on WPA passphrase was already discussed in the section above. Another attack on WPA encrypted packet data (not the passphrase itself) is improved Chopchop attack.

Chopchop attack on WPA

In WPA, a falsified encrypted packet that is made from an encrypted packet accepted in the past is discarded since the value of IV is checked (IV is used as a counter - replay attacks protection). Therefore, the attack can be executed only on WPA with the following conditions met: for client to AP communication is used TKIP, the IPv4 protocol is used in a way where most of the bytes are known to the attacker (for example 192.168.0.X), TKIP uses a long re-keying interval, and the network supports 802.11e Quality of Service features, which allows 8 different channels to be used for better data flow. If these conditions are met, the attacker can start the Chopchop attack. We will not discuss this attack in greater detail in this paper. To elude the TKIP defense against Chopchop attacks, the attacker needs to execute the attack on a different QoS channel. To prevent this attack, a very short rekeying time is suggested, since during 120 seconds the attacker is not able to recover full ICV value. [14]

3.4 WPA2

In September 2004, the Wi-Fi Alliance introduced Wi-Fi Protected Access 2 (WPA2), which is the second generation of WPA security. WPA2 still uses PSK authentication but instead of TKIP encryption it uses enhanced data encryption: a specific mode of the Advanced Encryption Standard (AES) known as the Counter Mode Cipher Block Chaining-Message Authentication Code (CBC-MAC) protocol (CCMP). [21],[22]

WPA2, similarly to WPA, offers two modes of operation:

- WPA2-PSK (Pre-Shared Key) mode
- WPA2 Enterprise mode

In WPA2-PSK mode the 256-bit key is generated from plain-text passphrase called pre-shared key (PSK). Like WPA, the same input is used to generate the pairwise master key (PMK), which is used to initiate the handshake: passphrase, SSID and SSID length.

On the other hand, when WPA2 Enterprise mode is used, much more complex authentication schemes can be supported since authentication is performed with the aid of a Authentication,Authorization and Accounting (AAA) backend server. This offers similar advantage over WEP as the WPA Enterprise mode: The wireless client and the server are the only owners of a master key (which is further used to derive temporal keys for communication). One additional advantage that the key derivation procedure brings to mobile users, is that when an roaming node connects to a WPA2 AP, it firstly checks if it has a collection of keys, called PMK Security Association (PMKSA),which it could use with this AP. If there is such an information cached from a previous association of the wireless client with the AP, then there is no need for a full authentication procedure with a AAA server (only the 4-way handshake needs to be done locally between the AP and wireless client). Summing it up, the PMKSA caching significantly speeds up the authentication procedure for roaming clients without compromising the security of the network. [23]

The following table shows how WPA2 addresses the cryptographic weaknesses of WEP. [22], [24]

WEP Weakness	WPA2 solution
Short IV	The IV was replaced with a Packet Number field with size of 48 bits.
Weak data integrity	The WEP-encrypted checksum calculation has been replaced with the AES CBC-MAC algorithm. It calculates the 128-bit value, where the high order 64-bites are used as a MIC.
Master key over derived key	AES CCMP uses a set of keys that are derived from master key and other values. Master key is derived from the EAP or Protected EAP (PEAP) authentication process.
No rekeying	AES CCMP rekeys automatically to derive new set of keys.
No replay protection	AES CCMP uses a Packet Number field as a counter, therefore provides a protection against replay attacks.

3.4.1 CCMP

CCMP, which stands for Counter Mode with Cipher-Block Chaining Message Authentication Code Protocol, is the security protocol that was created as a part of 802.11 security. It was primarily designed to replace TKIP and WEP. Because the AES cipher is processor-intensive, older hardware had to be replaced to support new CCMP/AES encryption processing.

CCMP is based on the CCM of the AES encryption algorithm. CCM combines Counter Mode (CTR) for data confidentiality and Cipher-Block Chaining Message Authentication Code (CBC-MAC) for authentication and integrity. As it was already mentioned before, CCM is used with the AES block cipher. When implemented as a part of CCMP encryption method, AES uses a 128-bit key and also encrypts data in 128-bit blocks. [25]

3.4.2 WPA2 Encryption

AES CCMP uses CBC-MAC to calculate the MIC and AES counter mode to encrypt the 802.11 payload and the MIC. AES CBC-MAC uses the following process to calculate a MIC value:

1. Encrypt a starting 128-bit block with data integrity key and AES. (Result1)
2. In the next step, XOR Result1 with next 128-bit block to produce XResult1.
3. Encrypt XResult1 with AES and data integrity key. (Result2)
4. XOR Result2 and the next 128-bit block of data.

The steps 3 and 4 are repeated until there is no more data. The output is a 128-bit block. The high-order 64-bits is the WPA2 MIC. The following figure 3.6 illustrates the afore mentioned process.

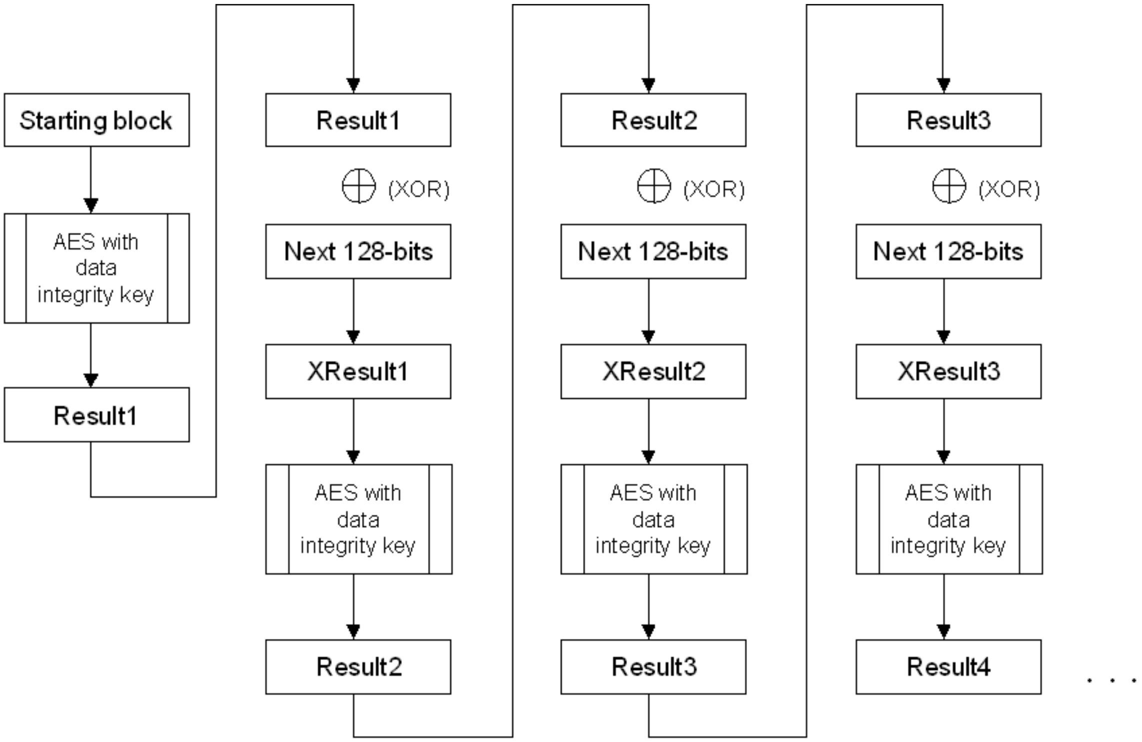


Figure 3.6: Calculating MIC Value

The AES Counter Mode (CTR) encryption algorithm works as follows:

1. Encrypt a starting 128-bit counter with AES and the data encryption key. (Result1)

2. In the next step, perform XOR operation between Result1 and the first 128-bit block of data that is being encrypted. The result is the first 128-bit encrypted block.

3. Increment the counter and encrypt it with AES and data encryption key. (Result2)

4. XOR Result2 and the next 128-bits of data. The result of the process is 128-bit encrypted block.

AES repeats steps 3 and 4 for all the 128-bit blocks in the data. For the final block, AES counter mode XORs the encrypted counter with the remaining bits, which produces the encrypted data with size of the last block of data. Figure 3.7 illustrates the whole process.

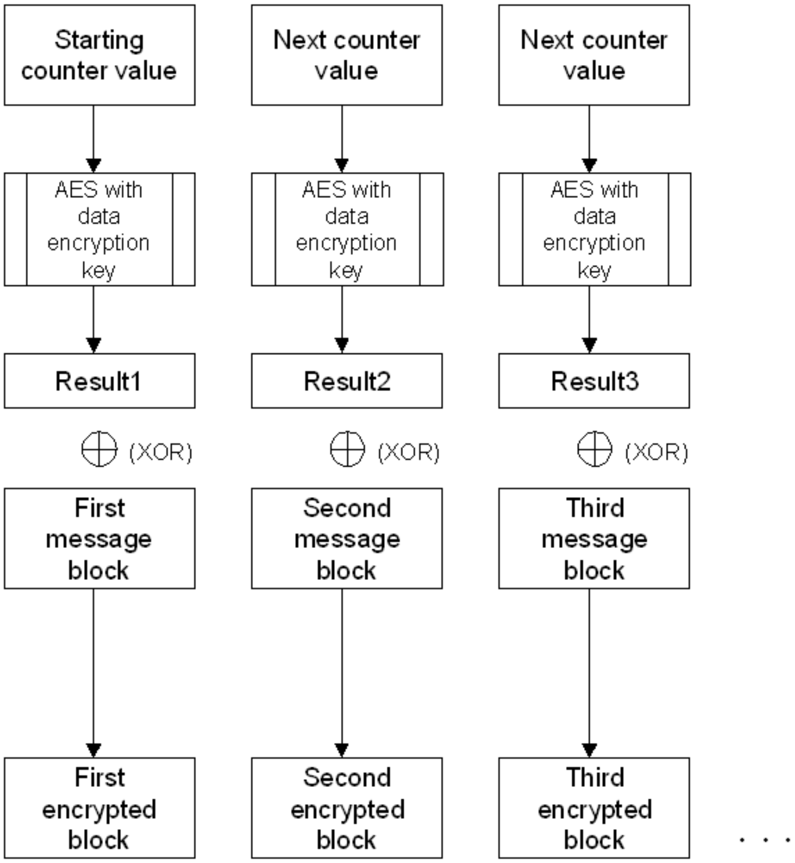


Figure 3.7: AES Counter Mode Encryption Algorithm

To encrypt an unicast data frame, WPA2 uses the following steps:

1. Starting block, 802.11 MAC header, CCMP header, data length and padding fields into CBC-MAC algorithm to get MIC as the result.

2. The starting counter value and the combination of the data with the MIC calculated in step 1 are used as an input for the AES Counter mode encryption algorithm with the data encryption key to produce the encrypted data and MIC.

3. CCMP header, which contains Packet Number, is added to the encrypted part of the payload.

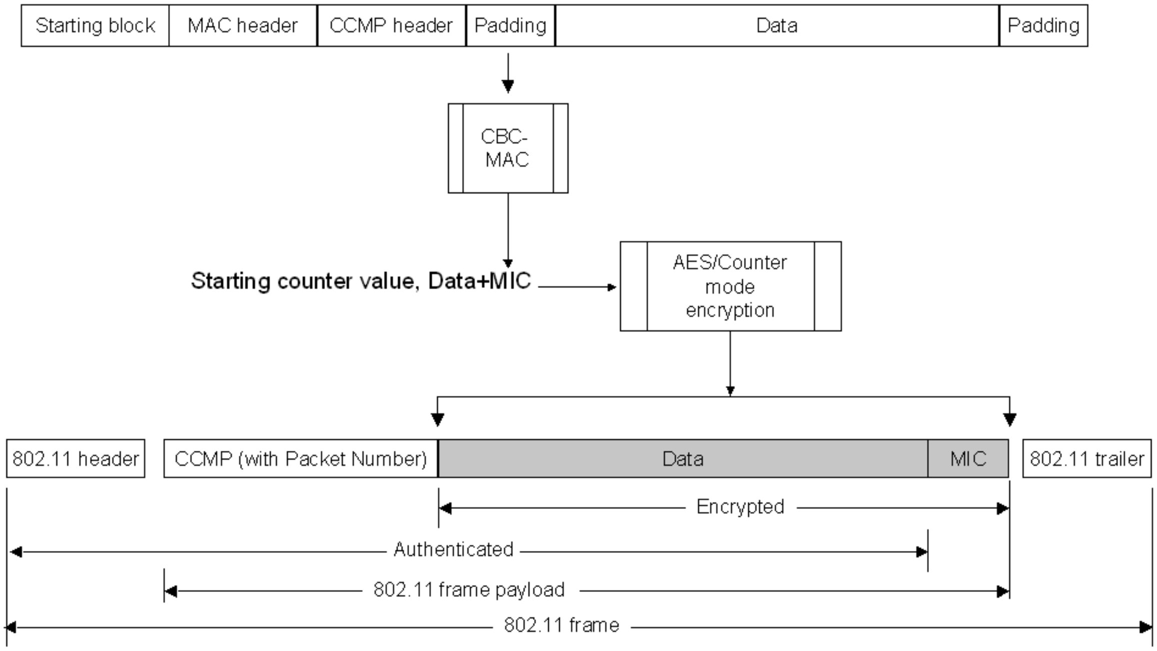


Figure 3.8: WPA2 Encryption

3.4.3 WPA2 Decryption

Decryption process can be summarized in these 4 steps:

1. Find the value of the starting counter from values in 802.11 header and MAC header.

2. The starting counter value and the encrypted portion of the 802.11 payload are used as an input for the AES counter mode decryption algorithm with the data encryption key. The result is the decrypted data and MIC. To produce the decrypted data block, AES counter mode XORs the encrypted counter value with the encrypted data block.

3. The starting block, 802.11 MAC header, CCMP header, data length, and padding fields are used as an input for the AES CBC-MAC algorithm with the data integrity key to calculate a MIC.

4. To find out if the data is valid, compare the unencrypted MIC with the calculated value of MIC. If the values do not match, WPA2 discards data.

The following figure 3.9 describes the decryption process: [22]

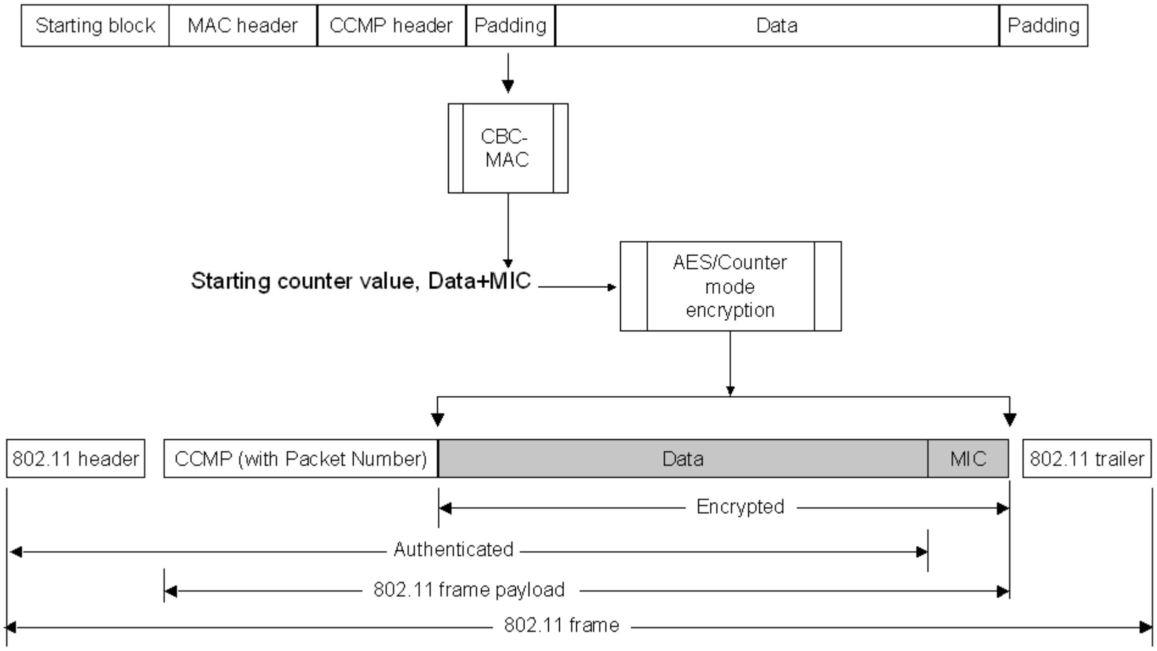


Figure 3.9: WPA2 Decryption

4

WPS

Wi-Fi Protected Setup (WPS) is Wi-Fi alliance's specification for secure association of wireless LAN devices. "The objective of WPS is to mutually authenticate the enrolling device with the Wi-Fi network and to deliver network access keys to the enrolling device. This is done by having the enrolling device interact with a device known as the "registrar", responsible for controlling the Wi-Fi network. The registrar may be, but does not have to be, located in the Wi-Fi access point itself." [26]

In other words, the standard was mainly designed to simplify configuring security on home networks. It automatically configures the WPA PSK without the user having to enter the key. The WPS protocol uses a PIN to automatically configure the PSK on the client device. The cleartext version of the PSK can be accessed once the PIN is known.

Almost all major vendors (including Cysco/Linksys, Zyxel, D-Link, Belkin, Buffalo, Technicolor and Netgear) have WPS certified devices. WPS is also activated by default on majority of the devices. Although the WPS is being marketed as a secure way of configuring the device, there are major implementation flaws which allow an attacker to perform a brute-force attack against the WPS PIN. We will discuss this attack in greater details in section 4.2.

4.1 WPS Authentication

During the WPS authentication process a **Diffie-Hellman key exchange protocol** is used. Main goal of the protocol is to define a secret value between two participants (let us call them A and B) which try to communicate each with other. This secret value is then used to establish a secret key, which is then used to encrypt the communication between A and B.

Before addressing the WPS setup methods, we will list the terminology used:

- The **enrollee** is a new device which wants to access the wireless network.
- The **registrar** provides wireless settings to the enrollee.
- The **access point** provides wireless hosting and also proxies message exchange between registrar and enrollee (often the AP is also a registrar at the same time).

Wi-Fi Protected Setup supports three setup methods: [27]

- Push Button Configuration
- PIN entry
- Out-of-Band

Push Button Configuration

Push Button Configuration (PBC) is a method that provides an unauthenticated key exchange. The user is required to push buttons (either an actual or virtual one) on both the registrar and the enrollee devices. The pushing causes both to initiate an unauthenticated Diffie-Hellman exchange: the enrollee device starts sending probe requests to all APs in range. The AP responds only if its registrars button has been pushed - conditioned by user. The timeout limit for an authentication (pushing buttons) is two minutes.

PIN Entry

PIN entry, sometimes also called In-band configuration, enables association based on a shared secret PIN. The PIN may be temporary (changeable, displayed by the enrollee) or static and is usually 8 digits long (alternatively, a 4 digits long PIN can be used). The authentication using PIN can be done in two different ways:

- Internal Registrar - user is required to submit a PIN of the connecting device into web interface of an AP.
- External Registrar - PIN of the registrar (AP in most cases) is entered into form on the connecting device itself. Since this method requires no other authentication but providing the PIN, it is vulnerable to brute force attack.

Out-of-Band

The Out-of-Band is intended to be used with channels like USB-flash drives, NFC-tokens[1] or two-way NFC interfaces. Such an alternate communication channel is used to transfer some information between the registrar and the enrollee. There's three possible scenarios: [26] [27]

1. Exchange of public key commitments[2], typically intended for two-way NFC interfaces, where the entire Diffie-Hellman exchange and the delivery of access keys takes place over the out-of-band channel.
2. Unencrypted key transfer - an access key is transmitted from a registrar to enrollees in unecrypted form by using either USB-flash drives or FC-tokens.
3. Encrypted key transfer - similar as the previous case, except that the key is encrypted using a key derived from the (unauthenticated) Diffie-Hellman key agreed in-band.

[1] NFC stands for Near Field Communication. The main characteristic of NFC is that it is a wireless device with limited range (generally about 10 cm). In this case, it uses Contactless Tokens to store configuration data to access Wi-Fi network, which helps new users to configure their devices in order to access the network. [28]

[2] Public keys of the devices or their hashes

The following tables illustrates the WPS authentication with PIN entry setup method used (external): [29]

IEEE 802.11			
	Supplicant → AP	Authentication Request	802.11 Authentication
	Supplicant ← AP	Authentication Response	
	Supplicant → AP	Association Request	802.11 Association
	Supplicant ← AP	Association Response	
IEEE 802.11/EAP			
	Supplicant → AP	EAPOL Start	EAP Initiation
	Supplicant ← AP	EAP-Request Identity	
	Supplicant → AP	EAP-Response Identity	
IEEE 802.11/EAP Expanded Type			
M1	Enrollee → Registrar	N1 || Description || PK_E	Diffie-Hellman Key Exchange
M2	Enrollee ← Registrar	N1 || N2 || Description || PK_R || Authenticator	
M3	Enrollee → Registrar	N2 || E-Hash1 || E-Hash2 || Authenticator	
M4	Enrollee ← Registrar	N1 || R-Hash1 || R-Hash2 || $E_{KeyWrapKey}$(R-S1) || Authenticator	prove possession of 1^{st} half of PIN
M5	Enrollee → Registrar	N2 || $E_{KeyWrapKey}$(E-S1) || Authenticator	prove possession of 1^{st} half of PIN
M6	Enrollee ← Registrar	N1 || $E_{KeyWrapKey}$(R-S2) || Authenticator	prove possession of 2^{nd} half of PIN
M7	Enrollee → Registrar	N2 || $E_{KeyWrapKey}$(E-S2 || ConfigData) || Authenticator	prove possession of 2^{nd} half of PIN, send AP configuration
M8	Enrollee ← Registrar	N1 || $E_{KeyWrapKey}$(ConfigData) || Authenticator	set AP configuration

Enrollee = AP
Registrar = Supplicant = Client/Attacker

PK_E = Diffie-Hellman Public Key Enrollee
PK_R = Diffie-Hellman Public Key Registrar

Authkey and KeyWrapKey are derived from the Diffie-Hellman shared key

Authenticator = $HMAC_{AuthKey}$(last message || current message)

$E_{KeyWrapKey}$ = Stuff encrypted with KeyWrapKey (AES-CBC)

PSK1 = first 128 bits of $HMAC_{AuthKey}$(1^{st} half of PIN)
PSK2 = first 128 bits of $HMAC_{AuthKey}$(2^{nd} half of PIN)

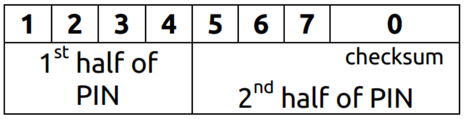
E-S1 = 128 random bits
E-S2 = 128 random bits

E-Hash1 = $HMAC_{AuthKey}$(E-S1 || PSK1 || PK_E || PK_R)

E-Hash2 = $HMAC_{AuthKey}$(E-S2 || PSK2 || PK_E || PK_R)

R-S1 = 128 random bits

R-S2 = 128 random bits
R-Hash1 = $HMAC_{AuthKey}$(R-S1 || PSK1 || PK_E || PK_R)
R-Hash2 = $HMAC_{AuthKey}$(R-S2 || PSK2 || PK_E || PK_R)



1	2	3	4	5	6	7	0
1st half of PIN				2nd half of PIN			checksum

The authentication process is rather complicated, because there is a lot of things this protocol is trying to solve:

- Suppressing replay attacks
- Providing mutual authentication
- Creating encrypted channel between client/AP
- Being invulnerable against offline attacks

The result of an attempt to solve all of the above led to another fatal weakness of WPS implementation discussed in the following section.

4.2 WPS Implementation Flaws

Although the idea of WPS was good, it was met with an incorrect implementation. An attacker can derive information about the correctness of the PIN he provided from the AP response:

- The 1^{st} half of PIN was incorrect if he receives an EAP-NACK message after sending M4.
- The 2^{nd} half of PIN was incorrect if he receives an EAP-NACK message after sending M6.

In other words, this decreases the maximum possible amount of authentication attempts needed to guess the PIN right from 10^8 (100.000.000) to $10^4 + 10^4$ (20.000).

Although since the 8^{th} number is always a checksum of digits one to seven, the maximum amount of attempts even without PIN being cut in half is 10^7. But knowing that, we get reduced real maximum number of attempts: from 20.000 to $10^4 + 10^3$ (11.000).

The recovery of PIN gives an attacker full access to the network. Furthermore, the PIN is something which can not be changed on majority of devices, therefore knowledge of PIN gives the attacker an opportunity to connect to the network even if the administrator has changed the password or SSID.

Summing up, attacking WPS provides several advantages over the direct attack on WPA/WPA2: [30]

1. Cracking the WPS PIN is considerably faster and not as luck-dependent.
2. Knowledge of PIN enables recovering of the passphrase instantly even if the owner changes it (this can be done with wpscrack[3] tool, where the user can choose an option to start from a given PIN or with slightly modified Reaver tool, where editing the source code was the only work around I was able to find).
3. Access points with multiple radios (2.4/5GHz) can be configured with multiple WPA keys. Since the radios use the same WPS pin, knowledge of the pin allows an attacker to recover all WPA keys.

4.3 How Many Devices Are Vulnerable?

According to [31], currently[4] around 74% of all devices which support WPS are vulnerable to this kind of attack on a wireless AP.

WPS is enabled on 87% of these devices by default. Furthermore, on 23% of the devices it is not possible to turn WPS off at all or turning it off has no effect whatsoever

[3] Download link can be found here: http://sviehb.wordpress.com/2011/12/27/wi-fi-protected-setup-pin-brute-force-vulnerability/

[4] 24.5.2012

on the vulnerability.

The average attack duration on various types of APs is around 7.5 hours.

4.4 Solutions

While discussing possible solutions it would be worth noting what can user do to prevent such an attack on his wireless AP and what is beyond his possibilities.

User's Options

The easiest solution is disabling WPS on AP manually via user interface or button. With WPS off, no one can abuse it to gain PSK. Although, it has been reported that even pushing it off manually does not really disable the feature on some of the APs (previous section). [32]

The only other thing the user can do (if he tried turning WPS off and it did not work or it did not prevent WPS brute force attack) is wait for a firmware upgrade and upgrade his AP once it is released.

Manufacturer's Options

Another obvious solution against brute force attack of any kind is the lockout policy against an attacking client. This could make an attack impractical, since instead of 2 seconds per attempt (a common rate at which an attacker sends pin and restart the process with good signal strength) we could achieve for example 120 seconds (or even more) needed per single attempt, which would slow down the attack significantly. Most of APs, though, do not implement any lockout policy by default.

Also disabling WPS PIN configuration after certain amount of wrong PIN entries (resulting in lockout for a hour for example) could work as a good solution.

Another solution comes to mind straight away: When there is something wrong with

AP's response to PIN authentication attempt, why not changing it so the attacker could not see whether he guessed the PIN right or wrong?

The answer is, though, that it is not possible due to mutual authentication. To explain this, let's simplify the authentication process as follows:

Lets assume that there is a secure channel between the client and server. For better understanding, we will also change the word "Registrar" to "Server" and "Enrollee" to "Client". The simplified protocol now looks like this:

1. Server → Client: Hash(Secret1 || first_half_of_PIN)
2. Client → Server: Hash(Secret2 || first_half_of_PIN), Secret2
3. Server → Client: Secret1

What is going on: Neither party wants to give the other the PIN, so they give each other hashes of the PIN combined with long secrets. After that they send each other the long secrets. Now, even with no PINs sent, they can each combine the secrets they have received with the secret PINs they already know (or they think they know), see that the hashes computed are equal to the hashes sent, and thus know that the other has knowledge of the PIN. This way the mutual authentication is achieved.

As it was stated above, the client might have no idea what the first half of the PIN looks like. He can send a random one and wait for the response. Now let's take a closer look at the possible options we have with response on a PIN authentication attempt:[33]

1. Do as it is specified in the protocol: refuse to send Secret1, because the PIN attempt was wrong. Then the client is able to differentiate a correct guess of the first four digits from an incorrect guess (this is the basic weakness of WPS).
2. We could try to change the response as follows: Let's lie and send the fake Secret1: fake_Secret1. Although it looks like a good idea, an attacker can simply compare Hash(Secret1 || first_half_of_PIN) from the first message with

this new Hash(fake_Secret1 || first_half_of_PIN) and he knows that the response is a ”lie”, therefore the PIN guess was wrong (so it is basically the same as the number 1).

3. The third option is to tell the truth: Let’s send the Secret1 in response even though the PIN guess was incorrect (so an attacker could think that his guess was correct). There is also a major flaw in this: an attacker would know both a Hash(Secret1, first_half_of_PIN) and a Secret1. With that information in hands, he could run an offline attack to determine what is the real first half of the PIN. Obviously, an offline attack trough 10,000 possibilities would be very fast.

Summing up, there is no other way how to change the APs response to the PIN attempt to make it work ”right”. It might take years until this security threat will be completely gone. We could mention a WEP encryption standard here as an example: even though many years passed since this protocol was deprecated and marked as insecure, there is still a lot of wireless networks being encrypted this way.

4.5 Brute Force Methodology

The following chart illustrates how the optimized brute-force attack works: [29]

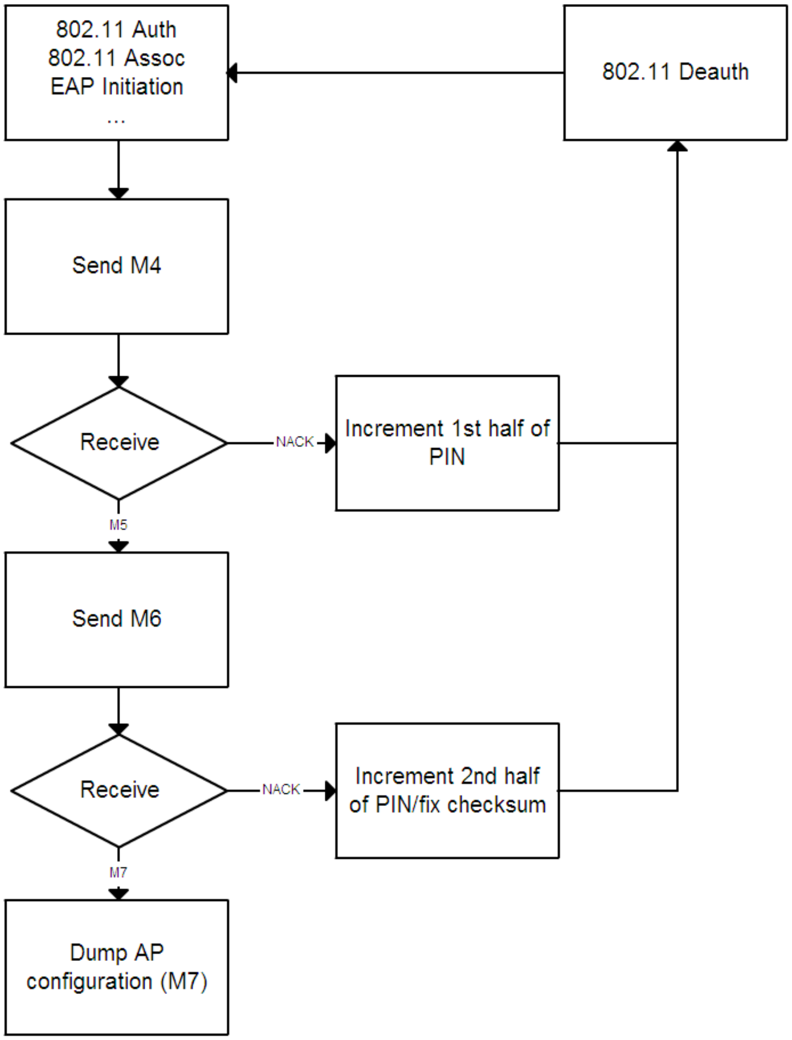


Figure 4.1: Brute Force Attack Chart

It is worth noticing that the attack is not trying to establish connection with the AP, therefore it does not send the final frame (M8). Instead, it simply displays the PIN and PSK on screen.

5 Tests and Experiments

In this chapter we will describe the experiments done, whether the tests were successful or not and what results we achieved from them.

5.1 Objective

Our objective was to perform successful brute force attack against a WPS PIN and describe the whole process. After that, we performed several tests on APs in two different organizations to find out how many of the APs used there were vulnerable to the attack.

5.2 Software and Hardware used

To successfully test a brute force attack experiment, there are two components needed: a wireless AP acting as a target of the attack and a client (an attacker) acting as a host with a network adapter capable of monitoring the WLAN traffic, inject packets and performing brute force attack.

For our experiments we used laptop Acer Aspire 5738Z with Atheros AR5B91 Wireless Network Adapter as an attacking device. The operation system used on the laptop was Linux distribution Debian GNU/Linux, since it is an open source free Linux distribution. Using Linux over Windows provides several advantages: switch the network adapter to monitor mode in order to monitor available WLAN networks is almost impossible under Windows OS. Also, the brute force tool chosen (Reaver) and an Aircrack-ng tools are made to be used under Linux distributions.

The Aircrack-ng [34] tool was used to enable monitor mode on the wireless network adapter. Since there is an open source brute-force WPS attacking tool, Reaver, released at [35], we used this implementation for our tests. To monitor the packet traffic between our laptop and the AP, the Wireshark [36] tool was used.

5.3 Reaver

Reaver is an open source tool for Linux distributions which implements the brute force attack against WPS PIN in order to receive the PSK. The source and free download can be found in [35] .

”Reaver has been designed to be a robust and practical attack against WPS, and has been tested against a wide variety of access points and WPS implementations.” [35]

Reaver in average needs 4 to 10 to recover the target AP’s passphrase. In practice, however, it will generally take less than a half of this time.

Factors influencing the length of the recovery process are:

- AP type
- Signal strength
- Lockout policy,

5.4 Example of Brute Force Attack

To prove our point that performing the attack itself is pretty easy task, as a first part of our experimental part we decided to test brute force attack at home against Zyxel NBG-416N Wireless N-Lite Home Router. This AP was configured to use WPA2-PSK with passphrase 14-60 characters long.

Let's assume that the Reaver and Aircrack-ng were both successfully installed and configured on attacking device using *apt-get install* command. As a first step, we disconnected the laptop from all networks (done in WLAN settings). After that, the wireless card was put into monitor mode. This was done by using Aircrack-ng utility, airmon-ng. The following command puts the wireless interface into monitor mode:

airmon-ng start wlan0

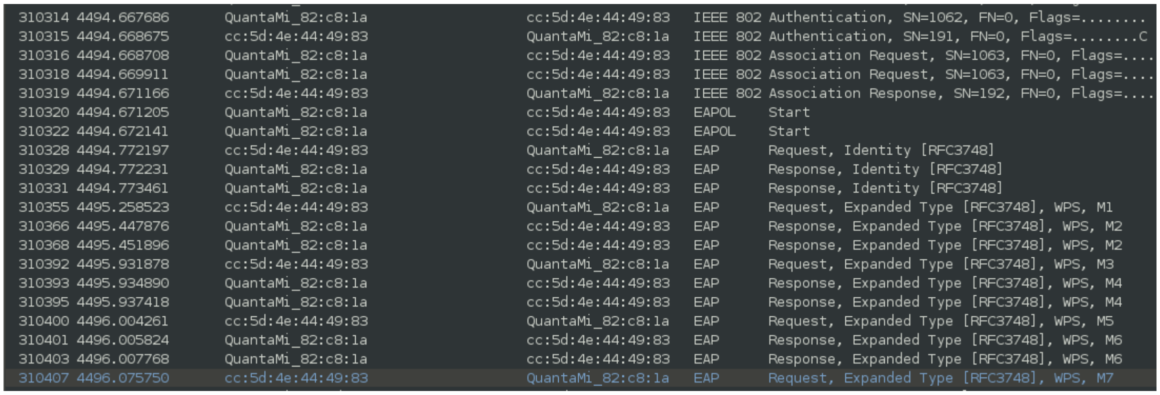
In words, it enables monitor mode on wlan0. On our laptop, the monitoring interface was called mon0. To monitor the network traffic, the airodump-ng tool was used:

airodump-ng mon0

This provided us with the list of wireless networks in range of our laptop. After finding the BSSID of the targeted network, we started Reaver to perform a brute force attack against PIN:

reaver -i mon0 -b XX:XX:XX:XX:XX:XX

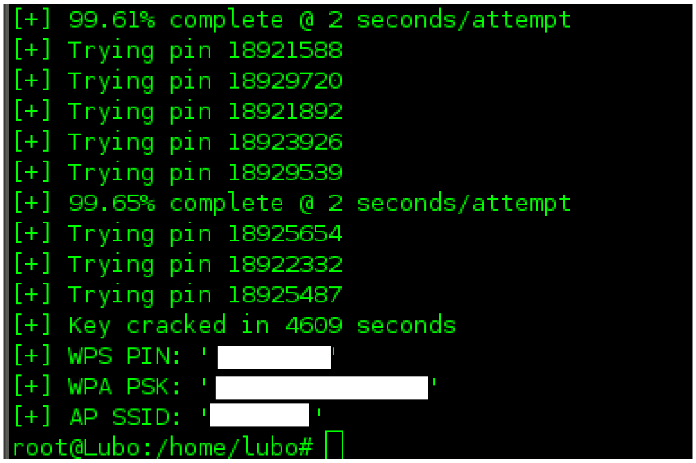
The figure 5.1 shows an example of frame exchange during the attacking process.



```
310314 4494.667686   QuantaMi_82:c8:1a   cc:5d:4e:44:49:83   IEEE 802 Authentication, SN=1062, FN=0, Flags=........
310315 4494.668675   cc:5d:4e:44:49:83   QuantaMi_82:c8:1a   IEEE 802 Authentication, SN=191, FN=0, Flags=........C
310316 4494.668708   QuantaMi_82:c8:1a   cc:5d:4e:44:49:83   IEEE 802 Association Request, SN=1063, FN=0, Flags=....
310318 4494.669911   QuantaMi_82:c8:1a   cc:5d:4e:44:49:83   IEEE 802 Association Request, SN=1063, FN=0, Flags=....
310319 4494.671166   cc:5d:4e:44:49:83   QuantaMi_82:c8:1a   IEEE 802 Association Response, SN=192, FN=0, Flags=....
310320 4494.671205   QuantaMi_82:c8:1a   cc:5d:4e:44:49:83   EAPOL    Start
310322 4494.672141   QuantaMi_82:c8:1a   cc:5d:4e:44:49:83   EAPOL    Start
310328 4494.772197   cc:5d:4e:44:49:83   QuantaMi_82:c8:1a   EAP      Request, Identity [RFC3748]
310329 4494.772231   QuantaMi_82:c8:1a   cc:5d:4e:44:49:83   EAP      Response, Identity [RFC3748]
310331 4494.773461   QuantaMi_82:c8:1a   cc:5d:4e:44:49:83   EAP      Response, Identity [RFC3748]
310355 4495.258523   cc:5d:4e:44:49:83   QuantaMi_82:c8:1a   EAP      Request, Expanded Type [RFC3748], WPS, M1
310366 4495.447876   QuantaMi_82:c8:1a   cc:5d:4e:44:49:83   EAP      Response, Expanded Type [RFC3748], WPS, M2
310368 4495.451896   QuantaMi_82:c8:1a   cc:5d:4e:44:49:83   EAP      Response, Expanded Type [RFC3748], WPS, M2
310392 4495.931878   cc:5d:4e:44:49:83   QuantaMi_82:c8:1a   EAP      Request, Expanded Type [RFC3748], WPS, M3
310393 4495.934890   QuantaMi_82:c8:1a   cc:5d:4e:44:49:83   EAP      Response, Expanded Type [RFC3748], WPS, M4
310395 4495.937418   QuantaMi_82:c8:1a   cc:5d:4e:44:49:83   EAP      Response, Expanded Type [RFC3748], WPS, M4
310400 4496.004261   cc:5d:4e:44:49:83   QuantaMi_82:c8:1a   EAP      Request, Expanded Type [RFC3748], WPS, M5
310401 4496.005824   QuantaMi_82:c8:1a   cc:5d:4e:44:49:83   EAP      Response, Expanded Type [RFC3748], WPS, M6
310403 4496.007768   QuantaMi_82:c8:1a   cc:5d:4e:44:49:83   EAP      Response, Expanded Type [RFC3748], WPS, M6
310407 4496.075750   cc:5d:4e:44:49:83   QuantaMi_82:c8:1a   EAP      Request, Expanded Type [RFC3748], WPS, M7
```

Figure 5.1: Example of Frames Captured by Wireshark

After the attack is successfully completed, we can see the following results in shell: PIN, PSK and BSSID of the target network. We can see the result of the attack on figure 5.2.



```
[+] 99.61% complete @ 2 seconds/attempt
[+] Trying pin 18921588
[+] Trying pin 18929720
[+] Trying pin 18921892
[+] Trying pin 18923926
[+] Trying pin 18929539
[+] 99.65% complete @ 2 seconds/attempt
[+] Trying pin 18925654
[+] Trying pin 18922332
[+] Trying pin 18925487
[+] Key cracked in 4609 seconds
[+] WPS PIN: '[redacted]'
[+] WPA PSK: '[redacted]'
[+] AP SSID: '[redacted]'
root@Lubo:/home/lubo#
```

Figure 5.2: Example of Reaver output

To prove the fact that the time needed to successfully complete the attack in same conditions is basically random (depends mostly on how fast is the first half of PIN guessed), we've repeated the attack 10 times with various PSK lengths and characters used. The results achieved are summarized in the following table :

No.	PSK Length	Duration
1.	14	96 minutes
2.	14	77 minutes
3.	63	226 minutes
4.	63	119 minutes
5.	47	204 minutes
6.	47	112 minutes
7.	25	84 minutes
8.	25	265 minutes
9.	20	189 minutes
10.	20	106 minutes

Note

It is worth noting that even though the estimated average time of an attack is 4-10 hours [35], during our tests there were attack attempts which were way out of this range- the attack was completed much faster. It was caused mainly by the fact that in these attempts we achieved almost ideal conditions for performing an attack: maximum strength signal, decent router CPUs capabilities, and no error messages (no lost frames, no timeouts occured) resulted in very fast PIN attempts (a little less than 2 seconds per one). At this rate, even the worst possible scenario (generating all 11.000 possible PINs) would take less than 6 hours to complete.

5.5 Attacks on WLANs in Two Different Organizations

In order to supply our work with some more data, we tried to test the attack on more than just one WLAN which was configured for that purpose. In order to do that, we were given a permission to perform attacks on wireless networks in two different organizations located in Bratislava, Slovakia (although due to legal reasons we are not eligible to provide the names of the organizations).

5.5.1 Organization A

In the first organization, there were 13 wireless networks detected in the building. After quick research we found out that only 8 of them were potential targets of our attack: networks using WPA/WPA2 in PSK mode. It was further confirmed by a system administrator of the company that none of the targeted wireless networks contains more than just one AP. From the rest, 4 of the networks were encrypted by WEP and one was not using any encryption at all.

To act as an attacker, we tried to perform attacks from the public section of the building: the section, which is not restricted area for employees only or is located outside. This approach caused considerably weaker signal strength on some of the target APs (signal strength varied from 34% to 76%). Due to such a low signal strength, a successful attack took longer than the attacks performed in previous section.

From the 8 WLANs we tested attacks on 6 were not successful: after an authentication the program could not test PIN requests (EAP messages), which means that the AP did not support WPS or it was turned off - in this case it was confirmed by an administrator that the APs are few years old and therefore they do not support WPS.

On the remaining two, the attacks were completed successfully in 422 (averaging 3 seconds/attempt) and 602 minutes (averaging 5 seconds/attempt) respectively. If we compare achieved results with the results from the previous section, there is a considerable increase in duration of attack caused by lower strength signal.

5.5.2 Organization B

In the second organization we were able to move freely around the building, since there were no restricted areas. From 21 wireless networks found in the organization, 14 were marked as potential targets. The rest of them were again using either WEP (2 networks), no encryption (1 network) or WPA/WPA2 in the Enterprise mode (4 networks).

We tried to brute force all WLANs with the following results:

- 11 of the WLANS were not vulnerable to WPS brute force attack.
- 3 of the WLANs were successfully attacked while the PIN and PSK were both recovered.

The results we achieved could be considered only as a minor security threat for the organization because:

- Majority of the WLAN users[1] can connect and use the WLAN safely, since the biggest (and most used) networks were being used in the Enterprise mode.
- The small local WLANs, which were vulnerable to our attacks, are being used only by small groups of users. It does not automatically imply that there is low or no chance of capturing important data, but we can nevertheless assume that with more users being potential targets the attackers chances would be considerably higher.

[1] Term ”users” in this case includes people using any kind of WLAN

Conclusion

The thesis provided a preview of mechanisms used for securing wireless networks. We discussed some basic terms about WLANs with possible attacks on WLANs in general and described how dangerous they are in the first chapter. This provided us with useful information which we used in the next chapter, which was dedicated to WLANs encryption standards and mechanisms. In the chapter we discussed how does each of the standards work: improvements compared to previous standard, encryption process, decryption process, flaws and possible attacks.

In fourth chapter we provided a detailed overview of WPS standard with possible authentication options, flaws and suggestions how these errors should be fixed in the future.

In the last chapter we performed an experimental part. The tests we were able to perform confirmed several thoughts we described in the previous chapter:

- It is possible to abuse the WPS implementation flaw to get full access to the wireless network.
- To do such a thing, all the potential attacker needs is a device with wireless network adapter, which is capable of injecting packets and monitoring traffic, and a little knowledge of working with any Linux distribution (since it is possible to boot Linux operating system from CD/DVD/USB, it is even easier).
- If the attack is not successful or possible, it is most probably caused by one of the following reasons:
  - The target AP does not support WPS or it is turned off manually on the device. Since the WPS is enabled by default on majority of devices which support WPS and regular users would not turn it off, the unsuccessful attack is more likely caused by the fact that the device is older and it does not support WPS at all.
  - The target AP implements a lockout policy, which makes a brute force attack impractical/impossible. However, during our tests there was no

such a device targeted. Either the attack got going and did successfully end or it did not start.

  - The target AP is being used in an Enterprise mode, which is not vulnerable to WPS PIN brute force attack.

Furthermore, our results support the assumption that in many companies (excluding those using only one network in Enterprise mode) there is going to be an AP which is vulnerable to this kind of attack. That means that there may be a weak spot in an otherwise good protected network, which enables malicious attacker to access confidential information.

Bibliography

[1] M. Gast, *802.11 Wireless networks: The definitive guide.* O'Reilly & Associates, Inc., Sebastopol, 2002.

[2] "What is 802.11 wireless?." http://technet.microsoft.com/en-us/library/cc785885(v=ws.10).aspx.

[3] T. Nghi, "Wireless local area network." http://www.netlab.tkk.fi/opetus/s38118/s00/tyot/25/index.shtml.

[4] A. M. A. Naamany, A. A. Shidhani, and H. Bourdoucen, "IEEE 802.11 wireless lan security overview," in *IJCSNS International Journal of Computer 138 Science and Network Security*, pp. 138–155, 2006.

[5] Q. Docter, E. Dulaney, and T. Skandier, *CompTIA A+ Complete Study Guide: Exams 220-701 (Essentials) and 220-702 (Practical Application).* Sybex, 2009.

[6] "DHCP starvation attack." http://www.networkdictionary.com/networking/DHCPStarvationAttack.php.

[7] "Cisco unified wireless network architecture—base security features." http://www.cisco.com/en/US/docs/solutions/Enterprise/Mobility/emob41dg/ch4_Secu.html.

[8] S. DeFino, B. Kaufman, N. Valenteen, and L. Greenblatt, *Official Certified Ethical Hacker Review Guide.* Cengage Learning, 2009.

[9] M. Stxhlberg, "Radio jamming attacks against two popular mobile networks," 2000.

[10] A. Mousa and A. Hamad, "Evaluation of the RC4 algorithm for data encryption," *Proc. Of International Journal Computer Science & Applications*, vol. 3, no. 2, 2006.

[11] B. Bing, *The world wide Wi-Fi : technological trends and business strategies.* John Wiley & Sons, Inc, 2003.

[12] H. Berghel and J. Uecker, "Wifi attack vectors," *Communications of the ACM*, vol. 48, no. 8, pp. 21–28, 2005.

[13] K. Hulin, C. Locke, P. Mealey, and A. Pham, "Analysis of wireless security vulnerabilities, attacks, and methods of protection," *Information Security Semester Project*, 2010.

[14] E. Tews and M. Beck, "Practical attacks against WEP and WPA," in *Proceedings of the second ACM conference on Wireless network security*, pp. 79–86, ACM, 2009.

[15] E. Tews, "Attacks on the WEP protocol," *IACR Eprint Server, eprint. iacr. org*, no. 2007/471, 2007.

[16] B. Haines, *Seven Deadliest Wireless Technologies Attacks.* Syngress/Elsevier, 2010.

[17] J. Wang, *Computer network security : theory and practice.* Springer, 2009.

[18] J. Davies, "Wi-fi protected access data encryption and integrity." `http://technet.microsoft.com/en-us/library/bb878126.aspx`.

[19] J. Edney and W. A. Arbaugh, *Real 802.11 Security: Wi-Fi Protected Access and 802.11i.* Addison Wesley, 2004.

[20] A. A. Vladimirov, K. V. Gavrilenko, and A. A. Mikhailovsky, *Real 802.11 Security: Wi-Fi Protected Access and 802.11i.* Addison Wesley, 2004.

[21] M. D. Ciampa, *Security+ Guide to Network Security Fundamentals.* Course Technology, Cengage Learning, 2012.

[22] J. Davies, "Wi-fi protected access 2 data encryption and integrity." `http://technet.microsoft.com/en-us/library/bb878096.aspx`.

[23] P. Georgopoulos, B. McCarthy, and C. Edwards, "Towards a secure and seamless host mobility for the real world," in *Wireless On-Demand Network Systems and Services (WONS), 2011 Eighth International Conference on*, pp. 134–141, IEEE, 2011.

[24] P. Chandra, A. Bensky, R. Olexa, D. Dobkin, and D. Lide, *Wireless networking.* Newnes, 2007.

[25] D. D. Coleman, *CWSP : certified wireless security professional official study guide.* Wiley Pub, 2010.

[26] J. Suomalainen, J. Valkonen, and N. Asokan, "Security associations in personal networks: A comparative analysis," *Security and Privacy in Ad-hoc and Sensor Networks*, pp. 43–57, 2007.

[27] C. Kuo, J. Walker, and A. Perrig, "Low-cost manufacturing, usability, and security: an analysis of bluetooth simple pairing and wi-fi protected setup," in *Proceedings of the 11th International Conference on Financial cryptography and 1st International conference on Usable Security*, pp. 325–340, Springer-Verlag, 2007.

[28] E. Haselsteiner and K. Breitfuß, "Security in near field communication (NFC)," in *Workshop on RFID Security RFIDSec*, vol. 6, 2006.

[29] S. Viehböck, "Brute forcing wi-fi protected setup." `http://sviehb.files.wordpress.com/2011/12/viehboeck_wps.pdf`.

[30] C. Heffner, "Cracking wifi Protected Setup with Reaver." `http://www.tacnetsol.com/news/2011/12/28/cracking-wifi-protected-setup-with-reaver.html`.

[31] "WPS flaw vulnerable devices." `https://docs.google.com/spreadsheet/lv?key=0Ags-JmeLMFP2dFp2dkhJZGIxTTFkdFpEUDNSSHZEN3c&type=view&gid=0&f=true&sortcolid=9&sortasc=true&rowsperpage=250`.

[32] T. Higgins, "Waiting for the WPS fix." `http://www.smallnetbuilder.com/wireless/wireless-features/31664-waiting-for-the-wps-fix`.

[33] D. Kaminski, "How the WPS bug came to be, and how ugly it actually is." `http://dankaminsky.com/2012/01/26/wps2/`.

[34] "Set of tools for auditing wireless networks." `http://www.aircrack-ng.org/`.

[35] "Brute force attack against WPS." http://code.google.com/p/reaver-wps/.

[36] "Wireshark: Protocol analyzer." http://www.wireshark.org/.

[37] P. Roshan and J. Leary, *802.11 Wireless LAN Fundamentals*. Cisco Press, 2003.