Annales UMCS Informatica AI 5 (2006) 407-416

Annales UMCS
Informatica
Lublin-Polonia
Sectio AI

http://www.annales.umcs.lublin.pl/

Cracking WPA? Is it possible?

Ryszard Haraszczuk*

Institute of Computer Science, Lublin University of Technology,
Nadbystrzycka 36B, 20-618 Lublin, Poland

Abstract

Wireless networks are commonly used by people. Market devices are equipped with the new security standard 802.11i. The standard announced in the middle of 2004 is commonly used to protect WLANs. Currently we can find the information that vulnerabilities were found in that standard. Then a question arises: Are WLANs compromised again? The article describes vulnerabilities which were discovered in the new WLAN security standard. It explains how the four-way handshake process works. Which part of that process can be used to break through our network? What kind of devices should we possess to check that vulnerability? What kind of protection can be applied to avoid such vulnerability?

1. Introduction

Recently WLAN technologies have found an important place in LAN architecture. Low costs of implementation of WLANs caused the fact they are commonly used by large teams of people. Wireless networks are almost everywhere, some of them are unsecured. So there is not a big problem to then assign to 802.11 networks. Very often wireless networks are secured by WEP. Vulnerabilities of WEP protocol are very well described and even a person with minimal knowledge of WLANs can break that security. Very often unsecured and not configured the WLAN networks are common in cities. Such networks provide an easy way to read data sent by open WLANs and create possiblity of losing data. Some users can say that they do not have any valuable data and ability to lose so it is not a problem. Most of such users do not realize that opened networks provide possibilities to intruders to connect to the Internet. As a result of unsecuring the network or improper security can be less efficient network connection. That is why built in WLANs protections should be implemented. In the middle of 2004 the new security standard 802.11i was introduced. Wi-fi protected access (WPA) is the common name of new 802.11i standard. Then new standard ensured proper protection during the last two years.

* *E-mail address*: r.haraszczuk@pollub.pl

But we can ask it 802.11i is still secure. Maybe there are some vulnerabilities in the new standard. If there are some, how can we protect against them? Finally how can we check the security level of our wireless network? This article describes vulnerabilities which where discovered in 802.11i during last years. Modern hardware is needed to check security level of WLAN. It describes hardware which can be used with open source tools to pass through WPA protected networks. The main aim of that article is to show that we should implement 802.11i standard properly. Avoid using dictionary words as pre-shared keys at home WLANs.

2. Hardware used to check wireless security

The device used to connect to the wireless network is wireless card. There are some limitations hidden in the statement that almost every network card can be used for wireless hacking. The additional amount of time used during the process of setting up hardware to work in different operational modes is one of those limitations. Another are different grades of efficiency after that process. Precise technical reasons which improve the process of setting up device to work properly in different modes are client card sensitivity in dBm, client card chipset, client card power emission and consumption level, presence of external connector socket for additional antenna and the most important compatibility with the Unix systems. The first question asked by beginners is which model and manufacturer of wireless card is the best for wireless penetration. Chipsets used currently in wireless cards are described below.

Card based on Prism chipset

Client cards based on Prism I designed for 802.11, Prism II or Prism III designed for 802.11b, Prism Indigo designed for 802.11a or dual-band Prism GT working in 802.11b/g or Prism Duette working in 802.11a/b standard, Prism Nitro working in 802.11g standard are commonly used by large groups of hackers. The Intersil manufacturer of Prism chipsets makes board documentation, reference designs, applications notes and other technical papers available to people. That kind of chipset is possesed by the biggest set of drivers due to open documentation. Wlan-ng drivers available for Linux users include Jouni Malinen's Hostap drivers – supporting the working client card in the Access Point mode. That kind of drivers can be used to set up honeypot or DoS and man-in-the-middle attack tests. Cards based on prism chipset support Abandon's airjack drivers allowing frame manipulation and determining close networks SSIDs [1].

Card based on Aironet chipset

Cards equipped with Aironet chipset are based on Intersil Prism chipset. Cisco Inc. added some new functionalities during the process of chipset modification.

The first additional Aironet card's ability is a regulation of output transmit power. The second function is ability to hop over all ISM band channels without software hopper. Drivers support wide configuration in Microsoft Windows and Unix like operating systems. Chipset specification is proprietary and is different from Prism chipset that's why host-ap drivers and Airjack drivers do not work properly. Nevertheless, users have got to use source code closed drivers supplied by Cisco cards equipped with that chipset which are excellent for rogue access point detection and multi channel traffic analysis [1].

Card based on Hermes chipset

Development of that chipset is supported by Lucent ltd. Cards equipped with that chipset have good receiving sensivity. Drivers support Windows platform but binary Linux drivers are also available. Linux drivers based in orinoco_cs driver work under open source platform. Unfortunately, working in different modes is not supported by drivers. Setting up cards in the monitor mode requires patching original drivers. Another drawback is that HostAp drivers do not support cards equipped with that chipset [1].

Card based on Atheros chipset

Cards based on Atheros AR5000 chipset work in 802.11a band. Linux drivers are available thanks to madwi-fi and vantronix project. Cards working in 5GHz band are fully supported in Linux. Working in raw sniffing mode and packet injection is supported. Unfortunately, cards working in dual channel are not fully supported. It is possible to set up cards to work in the monitor mode but packet injection is not supported [1].

Card based on AMD chipset

Another type of cards are AMDtek Inc chipset equipped. Cards equipped with that chipset work in 802.11 a/b/g combo cards. Although there are not many documents describing specification of AMD8211 chipset, manufacturer support Linux precompiled drivers [1].

Card based on other manufactures chipset

Cards equipped with chipset of another vendor are supported by driverloader. Driverloader is the wrapper which allows windows drivers to be used under Linux. Another loader is Ndiswrapper which supports the same functionality.

Unfortunately users have got to know that not all hacking functionality is available [1].

That paragraph describes different chipset types used in wireless cards. Some information about functionality supported by these cards was characterized. The user has got to think which type of card he should possess to check wireless security.

3. How does WPA-PSK work?

An important thing used to connect to wireless network is appropriate network card. The person without knowledge of operation rules governing 802.11i standard is unable to attack wireless network secured by that protection method. So let us look how new wireless security works.

Standard 802.11i describes the process of connecting a new client to the network. That process is composed of:

a) policy negotiation used for the process of authentication,
b) process of authorization with one of the earlier negotiated methods. There are a few EAP types of authorization used,
c) generation and distribution of keys.

That standard contains a set of rules forcing changing of keys used to encrypt the data sent over the wireless network. Figure 1 presents the phase a,b,c.

Both versions of WPA can work in two modes. The first is designed for home users and the second for institutions:

WPA-PSK – that mode is designed for home users who do not have authentication server. The key used to authenticate at Access Point is statically kept on workstation. That key is used in the process of generating Primary Transient Key (used to encrypt the data sent between the workstation and the gateway), and Group Master Key (used to encrypt the data sent to broadcast) [2].

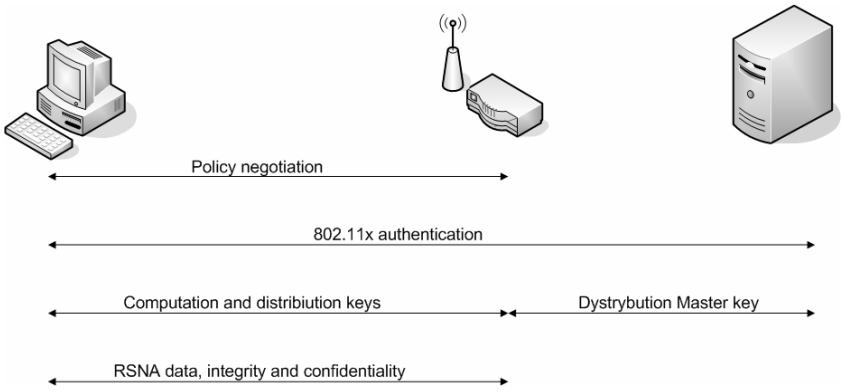

Fig. 1. Protocol 802.11i phases

WPA-EAP – that mode is designed for institutions. For example in EAP-TLS every user of WLAN has its own key. Authentication is made by Radius server (Remote Authentication Dial-In User Service). Radius stores users' keys. Access Point is Network Access Server (NAS) mediator between the authentication server (RADIUS) and the station (STA) or more often called supplicant. That mode resolves the problem of one shared keys for the whole institution. When one key is stolen the only device using that key is compromised. Another connections use different keys and are secure [2].

During the first phase of connecting the supplier to the network both Supplier and authenticator need to negotiate policy used to authenticate. Policies served by authenticator are announced by Beacon parameter or replied in Probe respond communicate. Policy used to authenticate is located in IE (Information Element) RSN frame. It contains:

- authentication served methods (802.11x, Pre-shared key),
- security protocols for group and individual communication (CCMP, TKIP),
- maintenance of preliminary authentication – supporting roaming of users between different access points.

When the first phase ends Authenticator and supplier go into Four-way handshake to negotiate temporary keys used to encrypt data. Let us look at the four way handshake process. Understanding that process is important to find insecurity which can be used by intruder.

The first step which is taken in the four-way handshake is for authenticator and supplicant to generate a nonce value. These values are random and different in every four-way handshake. Nonce value generated by authenticator is called Anonce. Nonce value generated by supplicant is called Snonce. After generating random value the authenticator starts the handshake process.

First message sent in the handshake process

Authenticator sent to the supplicant EAPOL-Key message containing Anonce. This message is not encrypted. Message is not secured against manipulation or change. If the attacker changes that message, the process of handshake falls down, that is why non protection is required.

Supplicant after receiving that message has all information needed to compute temporal keys from PMK, its own Snonce, his MAC address, Anonce from authenticator, MAC address of authenticator [3].

Second message sent from supplicant to authenticator

Authenticator can not compute temporal keys because he has no Snonce generated by supplicant. Unencrypted message sent from supplicant has Snonce in it. The second message sent to authenticator has protection which was not in

the first message. It contains MIC message integrity checksum. Supplicant has all keys needed to establish secure connection so it computes MIC of the second message and put it in it. MIC protects the second message against modifying by anyone. Secondly, MIC gives authenticator ability to check whether supplicant knows PMK. If the supplicant PMK's does not match that of the authenticator, MIC generated for the second message will be different. Authenticator has got to receive the second message and take Snonce of supplicant from it. Authenticator can compute temporal keys. After that he goes back and check MIC generated by supplicant. When it is correct, it means that supplicant has the same PMK as authenticator [3].

Third message sent from Authenticator to supplicant

Authenticator informs supplicant that he is ready to use new keys for encryption. It is important to synchronize operation of installing keys both on authenticator and supplicant. If one side turns on encryption before the second is ready, the established link will fall down. The third message includes MIC so supplicant can check whether authenticator has the same PMK key. The third message is also unencrypted and contains the sequence number (usually 0). That number is used to send the first encrypted message. That message is sent unencrypted until authenticator receives the fourth message. If the third message does not come to supplicant authenticator has got to resend that message that is why he does not install temporal keys [3].

Fourth message sent from supplicant to authenticator

That message is also unencrypted. Supplicant receives the third message and sends information message to authenticator. That message ends the four way handshake process. After that message supplicant installs temporal keys. Authenticator receives the fourth message and also installs temporal keys [3].

Authenticator and Supplicant have ability to encrypt messages sent in the network. After that process Supplicant and authenticator negotiate in the encrypted messages GTK group temporally keys supporting encryption of multicast and broadcast messages.

Temporary keys are destroyed when connection is closed by client or authenticator.

4. Example of WPA-PSK attack

Now we have theory, which can be used to perform the example attack on home WPA protected network using the dictionary Pre Shared key. The author of that attack is Richard Moskovitz.

The first thing is to possess proper network device. Most important thing is that chipset of our client card has to support working card in the monitor mode.

Monitor mode is equivalent to promiscuous mode in wired networks. Secondly our platform is a Unix-like operating system. And thirdly our drivers support packet injection. Attack is based on collecting all data needed to compute Primary transient keys. Hacker collects the following data: Access point MAC address, access MAC address of connected client, Anonce generated by Access point, Snonce generated by client. Next, a word from the dictionary is taken and primary transient key is computed. Message integrity checksum is generated and is compared to MIC of the second message sent during the four-way handshake process. When MICs from the handshake process is the same as generated by program it means that the word used to compute keys is the pre-shared key. Currently corporate WLANs are properly secured. There are not known vulnerabilities of WPA-EAP – mode designed for institutions.

More detailed phases of attack are:

Discover the presence of wireless network

First of all hacker has got to use his device to discover wireless networks. He can use kismet tool which shows all wireless networks. He needs to find on which channel WLAN works. Kismet is an open source tool. It can be downloaded from the internet. The process of installing is not difficult. It has also a graphical version called gkismet.

Kismet or gkismet can be used to discover SSID and MAC addresses of authenticator and supplicant. We can use a kismet tool to show detailed information about Access point and their clients.

The next step is to put wireless device in a monitor mode

It can be done by command

iwconfig interface monitor mode channel number_of_channel

Prepare deauthentication packets.

We can use an airforge tool to prepare a deauthentication packet. That tool is also an open source tool. If our Unix-like distribution does not have that tool in the standard commands, sources of that tool can be found in the internet and installed in our computer.

Generation of packet can be done by the command

airforge AP:MA:C_:AD:RE:SS CL:MA:C_:AD:RE:SS
name_of_deauthentication_packet

Enable packets collecting program

Before deauthentication packets are sent to the network program collecting all packets sent in the network should be enabled. There is very nice graphical tool called ethereal which can be used to do that. Ethereal is an open source tool and can be downloaded and installed in every Unix-like operating system.

Packet injection

There are a few tools which can be used to inject packets to the wireless network. For example tools which can be used to do that are packet2air and aireplay. The latter will be used to inject packets to the wireless network.

The command allowing us to send packets to the network looks like that presented below:

aireplay –m 26 –u 3 –w 0 –x 1 –r name_of_deauthentication_packet

Option: -m sets length of packet, -u specifies type of packet, -w turns on the use of WEP encryption or not, -x specifies a number of deauthentication packets sent per second

Derivation of pre shared key

Client card automatically tries to connect to the network after the process of deauthentication connected client. That is why we can collect all needed data to find pre-shared key. The open source tool which can be used to automatically find PSK is cowpatty. That tool uses dictionary pointed in the running command to check whether one of words from that dictionary is PSK.

5. Time-consumption of decryption

If our secret shared key is a dictionary word or simple 12345678 etc. chain it is not difficult to crack that password with a cowpatty tool. We have to know that the weaker and simpler the pre shared key is, the attack is shorter. We can ask how much time the process of brut force attack on the data collected in attack can take. Let us count how many possible sequences we can generate from the keyboard. The Number of different signs which can be typed from keyboard is 112 (without space bar – sometimes access point or client card rejects connections when the space bar is used in the pre shared key). The number of possible k – length subsequences put from the key board is given by the formula

$$C=W^K,$$

where K is the length of sequence C is the number of subsequences generated from all signs which can be typed, W is the number of all signs which can be typed.

Table 1 represents the number of possible subsequences, and the time which brut force attack can take. The number of seconds, minutes, hours, days is counted for the computer with four billion operations per second.

Access point allows minimal eight signs length pre-shared keys and the maximum is 64 signs length PSK.

Table 1. Number of possible k length pre-shared keys and time which process of checking them can take

K – length of subsequence	Number of subsequences	Seconds	Minutes	Hours	Days
7	2.21068E+14	55267.03519	921.1172531	15.35195422	0.639664759
8	2.47596E+16	6189907.941	103165.1323	1719.418872	71.64245302
9	2.77308E+18	693269689.4	11554494.82	192574.9137	8023.954738
10	3.10585E+20	77646205209	1294103420	21568390.34	898682.9307
11	3.47855E+22	8.69637E+12	1.4494E+11	2415659718	100652488.2
12	3.89598E+24	9.73994E+14	1.62332E+13	2.70554E+11	11273078682
13	4.36349E+26	1.09087E+17	1.81812E+15	3.0302E+13	1.26258E+12
14	4.88711E+28	1.22178E+19	2.0363E+17	3.39383E+15	1.41409E+14
15	5.47357E+30	1.36839E+21	2.28065E+19	3.80109E+17	1.58379E+16
16	6.13039E+32	1.5326E+23	2.55433E+21	4.25722E+19	1.77384E+18
17	6.86604E+34	1.71651E+25	2.86085E+23	4.76808E+21	1.9867E+20
18	7.68997E+36	1.92249E+27	3.20415E+25	5.34025E+23	2.22511E+22
19	8.61276E+38	2.15319E+29	3.58865E+27	5.98108E+25	2.49212E+24
20	9.64629E+40	2.41157E+31	4.01929E+29	6.69881E+27	2.79117E+26

6. Simple protections

The first proposed protection is to generate non dictionary SSID and PSK for home users. Insecurities described in the previous part of the article base on guessing PSK. If PSK is one of commonly used words and is present in a hackers' dictionary it is obvious that guessing it is not difficult. Many of us have to admit that it is difficult to generate a random sequence of signs by typing them on the keyboard. So one of the solutions of that problem is to use the program to generate PSK and SSID. After the process of generating them we can write that data on a writable compact disk and use it to set up our wireless devices. After the process of setting up devices we can destroy our secret data or keep it in a secure place and use to set up a new device.

Conclusions

Wireless networks are commonly used by a large group of people. While corporate mode using EAP-TLS or another method of authorization is very secure, home users have got to know about vulnerability which can be used to guess their pre-shared key. It is a very important thing to properly secure

WLANs using the pre-shared key mode. People using that mode have got to use keys which are less 20 signs long and are not based on words form dictionary. This article shows how the home mode of 802.11i standard works. It presents how the process of four-way handshake works. It describes process of the example attack.

References

[1] Vladimirov A.A., Gavrilenko K.V., Mikhailovsky A.A., *WI_FOO The secrets of Wireless Hacking*, Addison Wesley, (2004)
[2] Haraszczuk R., *Wireless Lan at home, in institutions and organizations*, Annales Informatica UMCS, Lublin, 1 (2005).
[3] Edney J., Arbaugh W.A., *Real 802.11 securityWi-fi Protected Access and 802.11i*, Addison Wesley, (2004).