

The Performance of the IEEE 802.11i Security Specification on Wireless LANs

Andrew Gin
Supervisor: Ray Hunt
November 2005

Abstract

Wireless networks are popular due to the mobility afforded by having connectivity without wires. Wireless networks are inherently vulnerable and the IEEE 802.11i security standard was established to address security. Previous research has evaluated the effects of WEP encryption and its variants on wireless network performance. However no previous work has evaluated the effects of the 802.11i specification (implemented as WPA and WPA2) on performance. The aim of this research is to discover the effects of the new WPA2 specification on wireless network throughput, latency, error rate and interaction with multiple clients. It also makes comparisons with the existing security methods, WPA and WEP. Synthetic and real network applications were used to generate network traffic. Various security levels were defined. These ranged from No Security to WPA2, and also included different authentication methods (Pre-shared key and 802.1x EAP-TLS certificate authentication). Performance was measured against these security levels. The results show that while there were statistically significant differences between the security levels, they are small enough to be realistically ignored. This research shows it is possible to establish a secure wireless network, without any noticeable compromise in performance. Using hardware with hardware accelerated security features, there is no reason to use anything less than the WPA2 security specification.

Acknowledgements

I would like to thank Mayank Keshariya and Nilufar Baghaei for their helpful suggestions, Craig Miller for his assistance in setting up the access point and Chris Harrow for his ideas setting up the server. Many thanks go to Pascal Lequeux and Yves Legendre of ZTI Telecom, for providing me with technical support and new license keys each time the hardware failed. I would like to thank my supervisor, Associate Professor Ray Hunt for supervising my project and in particular his relentless efforts in obtaining the WPA2 hardware. I would like to thank my friends and family for putting up with me in what has been a challenging and stressful year. Finally I would like to extend my thanks to the Honours room 'BrainTrust' students for making this year enjoyable.

Contents

1	Introduction	1
1.1	Overview	1
2	Wireless Network Security Methods	2
2.1	Wired Equivalent Privacy	2
2.1.1	Flaws with WEP	4
2.2	Wi-Fi Protected Access	7
2.3	Wi-Fi Protected Access 2	14
2.4	Related Work	16
3	Research Goals	18
3.1	Motivation	18
3.2	Objectives	18
4	Experimental Design	19
4.1	Network Setup	19
4.2	Security Levels	20
4.3	Network Measurement Methodology	21
4.4	Traffic Generator Configuration	22
5	Results and Discussion	24
5.1	Effects on Throughput	24
5.2	Effects with Multiple Clients	24
5.3	Effects on Packet Errors and Round Trip Times	26
5.4	Discussion	27
5.5	Limitations	28
6	Future Work	30
7	Conclusion	31

1 Introduction

Wireless networks have increased in use dramatically over the recent years. Mobility and the freedom to connect without any wires have contributed to its popularity. However wireless networks are inherently more vulnerable to attack than wired networks. Where a wired network needs to have its lines tapped, or direct physical access to the hub or switch to have its security compromised, a wireless network broadcasts over the open airwaves. Anyone within transmission range can receive this signal.

Because of this inherent vulnerability, the 802.11 wireless network standard [1] includes an encryption mechanism, known as Wired Equivalent Privacy (WEP). WEP's intention is to give wireless networks the same degree of security as an unsecured wired network [2]. The flaws in the implementation of WEP and its encryption key are widely published. This includes key reuse and the key length being inadequate. This can lead to the key being discovered through a brute force attack, using freeware tools available on the internet [3, 4].

To address these shortcomings, a comprehensive security standard, IEEE 802.11i was developed [2, 3, 5]. A specification, known as Wi-Fi Protected Access (WPA) was developed by the Wi-Fi Alliance¹, as a stopgap measure to address the flaws in WEP until 802.11i was finalised [6]. WPA is a 'subset' of the standards specified in IEEE 802.11i, and requires only a software upgrade, so it can work on existing hardware [7].

In late 2004, the Wi-Fi Alliance began interoperability certification testing on hardware supporting WPA2, representing the complete 802.11i specification. While it has some mechanisms in common with WPA, it has several improvements, including the use of the Advanced Encryption Standard (AES), a stronger encryption method. Since AES is a much stronger cryptographic algorithm, it requires a more powerful processor. This means much of the existing network hardware will need to be upgraded (for example access points and network cards) to cope with the increased computational load [8].

This paper investigates the effects of the WPA2 security specification on the performance (throughput, latency and errors) of wireless networks and compares this performance with the existing WEP and WPA architectures. It builds on previous work carried out in [9] by including the full WPA and WPA2 specifications.

1.1 Overview

Section 2 details WEP, WPA and WPA2. Section 3 justifies the motivations and goals of this project. Section 4 describes the experimental setup and methodology. Section 5 presents the results, statistical analysis and a discussion of the results. Section 6 describes avenues for future work and section 7 describes the conclusions.

¹<http://www.wi-fi.org>. The Wi-Fi Alliance is a non-profit organisation, certifying wireless products for interoperability.

2 Wireless Network Security Methods

This section details the current wireless LAN link layer security methods: Wired Equivalent Privacy (WEP), Wi-Fi Protected Access (WPA) and Wi-Fi Protected Access 2 (WPA2). All three of these methods use what is known as an Initialisation Vector (IV). This will be described before the methods are discussed.

An IV is used to ensure that ciphertext (encrypted data) encrypted with the same key is harder to recover [10]. With a stream cipher² an IV ensures that no encryption key is used more than once. When two plaintexts are encrypted with the same key, an exclusive-or (XOR) function can be applied to the two ciphertexts, which will give the same result as applying an XOR function to the original plaintexts. Brute force methods can then be used to recover the original two plaintexts, given an XOR of the original plaintexts. To overcome this, if the same encryption key is used, a sequence of bits are added to the start of the plaintext and these bits are also added to the key. The sequence of bits, or IV, are different for each plaintext, and this results in a different key for each plaintext.

With a block cipher³, when two plaintexts with common beginnings are encrypted with the same encryption key, the resulting ciphertext would be the same up until the first difference. This can result in a pattern that can aid intruders [11]. Adding an IV to the start of the plaintext will result in different ciphertexts, even when the two plaintexts are encrypted with the same key. As before, the IV is different for each plaintext.

Therefore the notion behind an IV is to protect against XOR methods for stream ciphers and revealing patterns for similar plaintexts when encrypted with block ciphers, as the IV is different for each plaintext.

2.1 Wired Equivalent Privacy

Wired Equivalent Privacy (WEP) was intended to make a wireless connection as secure as an unsecured wired network [2]. WEP attempts to bring confidentiality, access control and data integrity to a wireless connection.

WEP uses the RC4 algorithm to encrypt data. RC4 is a symmetric⁴ stream cipher and encrypts 8 bits at a time. The RC4 key used by WEP is made up of a shared secret key (which is usually either 40 or 104 bits in length) that only the sender and receiver know, and a 24 bit IV. This results in an RC4 encryption key of either 64 or 128 bits in length. A pseudo-random number generator uses this key (made up of the shared secret key and the IV) and the data length value of the plaintext to generate a key stream the same length as the plaintext data to be encrypted [12]. This key stream is XOR'd with the plaintext to produce the ciphertext. This part of WEP addresses confidentiality, as it prevents outsiders from viewing the messages as they are encrypted.

WEP affords data integrity by using an integrity check value (ICV) to ensure data is not manipulated during transmission (for example during a man-in-the-middle attack [13]). The ICV is essentially a CRC-32 integrity checksum and is

²A stream cipher is one that operates on a byte or a few bits at a time.

³A block cipher is one that operates on a block of bits (usually 64 or 128) at a time.

⁴A symmetric cipher is one which both the sender and receiver use the same key to encrypt and decrypt data.

32 bits long. It is calculated over the plaintext data only. ICVs generated from the same plaintext will be identical. Hence if the ICVs differ, then the plaintext must have been tampered with from its original. In this way the ICV provides for data integrity.

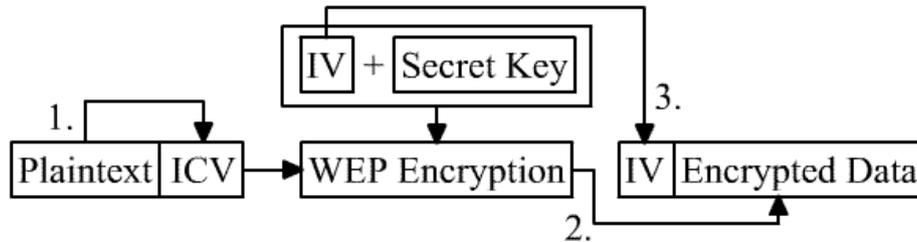


Figure 1: The WEP Encryption Process

Figure 1 illustrates how WEP works and how the RC4 algorithm and ICV are used.

1. The ICV is computed over the plaintext data, and added to the end of the plaintext data.
2. The plaintext and the ICV are encrypted with RC4. The IV is appended to the shared secret key to create the encryption key. The IV is different for each frame sent, therefore each frame is encrypted with a different key.
3. The IV is prepended to this encrypted data. The IV is viewable by anyone, as it is not encrypted. This is so the receiver knows which IV to use when decrypting.
4. This frame (made up of an IV and encrypted data) is then sent.

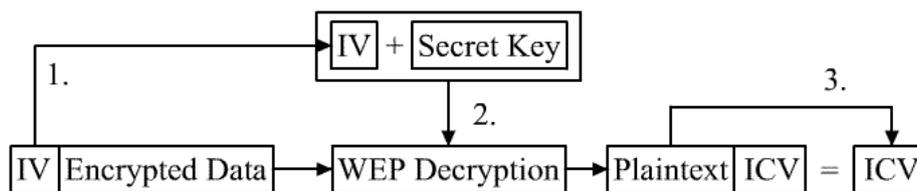


Figure 2: The WEP Decryption Process

When the recipient receives this frame (Figure 2), the following steps are performed:

1. The receiver has the same secret shared key as the sender. This secret key is combined with the IV in the received frame to create the key used to decrypt the data.
2. The frame is decrypted, resulting in plaintext data and the ICV.

3. An ICV is computed over the plaintext data. This ICV is compared with the ICV in the decrypted packet. If they are not identical, then the data is considered to have been modified in transit, and the frame is dropped and ignored. Otherwise it is passed onto the upper layer.

Access control in WEP is provided by two authentication methods. The first method, Open System Authentication, is not really an authentication method; it is a null authentication algorithm. Essentially any station that requests authentication is authenticated. This method has been provided so that other authentication protocols can be used instead of the WEP methods. There are two messages in this method:

- The requester (a station initiating the authentication exchange) sends a request to authenticate.
- The responder (the station that receives this response, for example an access point) replies, and the two stations are considered to be mutually authenticated [1].

The second method is Shared Key Authentication. For a station to be authenticated, it requires knowledge of the shared secret key. There are four messages exchanged in this method:

- The requester sends a request to authenticate.
- The responder generates a sequence of 128 random octets. This sequence is known as a challenge and is different with each request. This is sent to the requester.
- The requester encrypts this challenge using the WEP steps described previously, and sends it back to the responder.
- The responder receives this and decrypts it. If the ICV check is successful, the decrypted challenge is compared to the challenge sent in the second frame. If they are the same, then the requester is authenticated.

2.1.1 Flaws with WEP

WEP is flawed in each of the security aspects it seeks to address: confidentiality, access control and data integrity.

In terms of confidentiality, there are several problems. WEP uses the RC4 algorithm to perform the encryption. RC4 is the algorithm used in systems such as Secure Sockets Layer (SSL), often used to encrypt internet banking transactions. There is nothing wrong with the RC4 algorithm itself; it is the way it has been implemented in WEP that causes problems. One of the fundamental rules of RC4, and stream ciphers in general, is that an encryption key must never be used more than once [14]. As stated earlier, if a single key has been used to encrypt two different frames, the original plaintexts can be recovered. The use of an IV overcomes this. While the shared secret key is static for each frame, concatenating an IV to the static shared key will result in a different encryption key for each frame, as the IV is different for each frame. Herein lies part of the problem with WEP. The IV used in WEP is only 24 bits long. A 24 bit IV means there are almost 17 million (2^{24}) combinations for the IV. This means,

when combined with a static shared key, there can be 2^{24} frames transmitted before the IV space is exhausted. Once all the possible IVs have been used, the IVs will begin to cycle through previously used values for the IV; there are no unused IVs left. While 2^{24} frames may seem like a lot, a busy 802.11b access point, constantly sending 1500 byte frames at 11Mbps⁵, will take about 5 hours to send 2^{24} frames before it reuses IVs:

$$\begin{aligned} 11 \times 10^6 \text{ bits per second (bps)} / 8 &= 1375000 \text{ bytes sent per second} \\ 1375000 / 1500 &= 916.67 \text{ frames sent per second} \\ 2^{24} / 916.67 &= 18302 \text{ seconds} = \text{about 5 hours} \end{aligned}$$

This time is likely to be less, as many frames are significantly smaller than 1500 bytes (such as TCP acknowledgement frames [15]).

With an 802.11g or 802.11a access point, which operates at 54Mbps, the IV space will be exhausted in as little as an hour:

$$\begin{aligned} 54 \times 10^6 \text{ bps} / 8 &= 6750000 \text{ bytes sent per second} \\ 6750000 / 1500 &= 4500 \text{ frames sent per second} \\ 2^{24} / 4500 &= 3728 \text{ seconds} = \text{about 1 hour} \end{aligned}$$

When an IV is reused with the same shared secret key, the resulting encryption key will be the same as the encryption key created when the IV was first used. As stated before, two plaintexts can be recovered when they have been encrypted with the same encryption key. The IV length means this can happen in as little as after one hour

Another problem with the IV is that the 802.11 standard does not specify how IVs are selected. Some implementations set the IV to zero at initialisation and increment by one for each frame. While this ensures each frame has a different key, it will mean the lower values of the IV are used more often than the higher values. This results in a larger set of duplicate IV values (and therefore more encryption key reuse) [16]. Other implementations may select the IV randomly, however there is a 50% chance of IV reuse after 5000 frames [17].

The size of the shared secret key is also a problem for WEP. When WEP was first introduced, the US Government had export restrictions on encryption [18]. Only encryption up to 56 bit could be exported (which was why there was an international/export version of web browsers with 56 bit encryption and domestic versions for use in the US and Canada with 128 bit encryption). The law stated that any system using encryption larger than 40 bit and up to 56 bit required a key recovery system [19]. The IEEE needed to comply with these laws if it wanted products exported freely and accepted as an international standard. The problem is that a key of only 40 bits in length could be cracked using a single computer in as little as 50 hours [3] using brute force methods (where every possible combination is tested).

When the US Government lifted these restrictions, vendors extended the shared secret key length to 104 bits (resulting in a 128 bit key when combined with the 24 bit IV). Quite often these 104 bit implementations were interoperable [5]. Some vendors extended this to 152 bits, and even as large as 256 bits [2]. However these larger implementations were proprietary and were not

⁵Mbps = megabits per second, MB/s = megabytes per second

interoperable. A brute force attack on a 104 bit key would not be feasible; it would require 10^{19} years [20].

However since the implementation of RC4 is fundamentally flawed, increasing the key size does little to improve WEP. Using newer statistical methods, instead of brute force methods, and techniques focused on *unique* IVs, even a 104 bit key can be cracked in as little as three minutes [21].

Another problem with WEP is that the shared secret key is used directly in the encryption key (as it is appended to the IV to create a full length encryption key). Shared secret keys (sometimes known as master keys) should never be used directly; they should only be used to derive other keys [22].

Finally in terms of confidentiality, WEP has no key management. Every device on the wireless network needs to have knowledge of the same shared secret key. If this shared secret key is compromised, every other session on the wireless network can be decrypted. Changing the shared secret key requires manually changing it on every device on the wireless network. While this is feasible for a small home network, it may become inconvenient to change it every hour to accommodate for IV recycling. This has often resulted in the shared secret key remaining unchanged for the entire lifetime of the wireless network. While feasible on small networks, it is unpractical when trying to distribute the key across a large wireless network with many users.

With respect to access control and authentication, as stated earlier, there are two methods: open and shared key. While open is completely *open*, leaving the authentication to other protocols, there are problems with shared key authentication. One of these is that the authentication is performed in one way only. This can lead to man-in-the-middle attacks [3], as the requester does not know if they are authenticating with a real access point, or an outsider posing as an access point. Thus if attackers send a challenge to the requester, and receive an encrypted challenge, they will be able to determine the key stream used to encrypt the data. The attackers can then request to authenticate with a valid access point, and since they have knowledge of the key stream used, they will be able to authenticate properly with the access point. This is because the access point will see that the attacker can encrypt the challenge correctly and therefore authenticates the attacker as a legitimate user. This is known as authentication spoofing.

Another method for an attacker to spoof authentication is to simply observe the shared key authentication process passively [17]. Since anyone can observe wireless traffic, an attacker will be able to watch a legitimate user get authenticated. They will be able to view the challenge sent by the authenticator (for example an access point) and the encrypted challenge sent by the user. As before, knowing the unencrypted and encrypted challenge means the attacker can derive the key stream, and then authenticate as a legitimate user.

WEP is also flawed in terms of data integrity. The ICV in WEP is a CRC-32. The CRC-32 checksum is normally used for error detection, which it does well, however it is inappropriate to rely on a CRC-32 to check whether a message has been modified in transit. One problem with the CRC-32 is that it is linear. This means the CRC is spread out across the entire message. Flipping a bit in the encrypted message results in a known set of bits in the CRC that must also be flipped. This means an attacker could modify a message and adjust the CRC/ICV to make the modified message appear unchanged [23].

Another flaw with the ICV is that it is unkeyed; knowledge of a shared key

is not required to calculate a valid CRC. This means attackers can decrypt the messages and modify them. A valid CRC can then be computed (as all that is needed to calculate a valid CRC is knowledge of the plaintext) and then the forged frame can be passed on, without appearing to have been modified [24].

WEP is flawed in terms of confidentiality, access control and data integrity - the very features it was designed for. In terms of confidentiality, the IV is too small, which results in key stream reuse. IVs are also often initialised to a constant value and incremented, meaning some IVs are used more often than others. The original 40 bit WEP keys could be attacked successfully with brute force methods, so many vendors often extended the key length to 104 bit, which were uncrackable using brute force methods. However since the underlying principles of the IV are weak, new statistical methods can crack a 104 bit key faster than the time it takes to crack a 40 bit key using brute force methods. Key administration is not provided for; changing the shared secret key means manually entering it into each device on the wireless network. This has frequently resulted in keys remaining unchanged for the whole life of the network. The ICV is also flawed; messages can be tampered with, and still appear valid, by appropriately modifying the ICV. In terms of access control, using shared key is worse than open authentication paired with another protocol (for example 802.1x), as attackers can view the authentication process. This means they will have copies of the encrypted and unencrypted challenge and from these, can derive the key stream used to encrypt. Authentication is also only performed in one direction.

2.2 Wi-Fi Protected Access

The problems in WEP led to the creation of the 802.11i standard [5]. The 802.11i standard seeks to address all of the security issues concerning wireless LANs and is essentially split into three main parts: Temporal Key Integrity Protocol (TKIP) and Counter Mode with CBC-MAC Protocol (CCMP) both offer confidentiality and data integrity, and IEEE 802.1x provides for authentication. TKIP is designed for legacy devices and hardware that can only support WEP, while CCMP is a more advanced, robust protocol designed for all new devices. Either of these can be combined with 802.1x authentication; when 802.1x is combined with TKIP, it is known as WPA, and when 802.1x is combined with CCMP, it is known as WPA2. Wi-Fi Protected Access (WPA) was introduced in October 2003 as an interim solution, to immediately address the security flaws in WEP while the 802.11i standard was still under development. In order to effectively secure wireless LANs, WPA needed to work on existing hardware and infrastructure, and uses TKIP to achieve this. WPA also uses the IEEE 802.1x framework for authentication.

WPA has two modes of operation in terms of authentication: IEEE 802.1x and Pre-Shared Key (PSK). IEEE 802.1x authentication (also known as WPA Enterprise mode) is aimed at corporations with existing authentication infrastructure in place, such as RADIUS⁶ servers. IEEE 802.1x [25] uses the Extensible Authentication Protocol (EAP, [26]) as the transport protocol used for authentication; EAP is a transport method, not an authentication method.

⁶Remote Authentication Dial-In User Service. These are often used by organisations to authenticate dial-in users.

There are many⁷ actual authentication methods which can use EAP, for example smart cards and passwords. Common secure authentication types include EAP-TLS (Transport Layer Security), EAP-TTLS (Tunnelled TLS) and PEAP (Protected EAP). These secure methods utilise X.509 certificates and provide for mutual authentication. IEEE 802.1x is a port based authentication framework. It operates on the concept of a controlled and uncontrolled port. The topology of a wireless LAN may be arranged as in Figure 3.

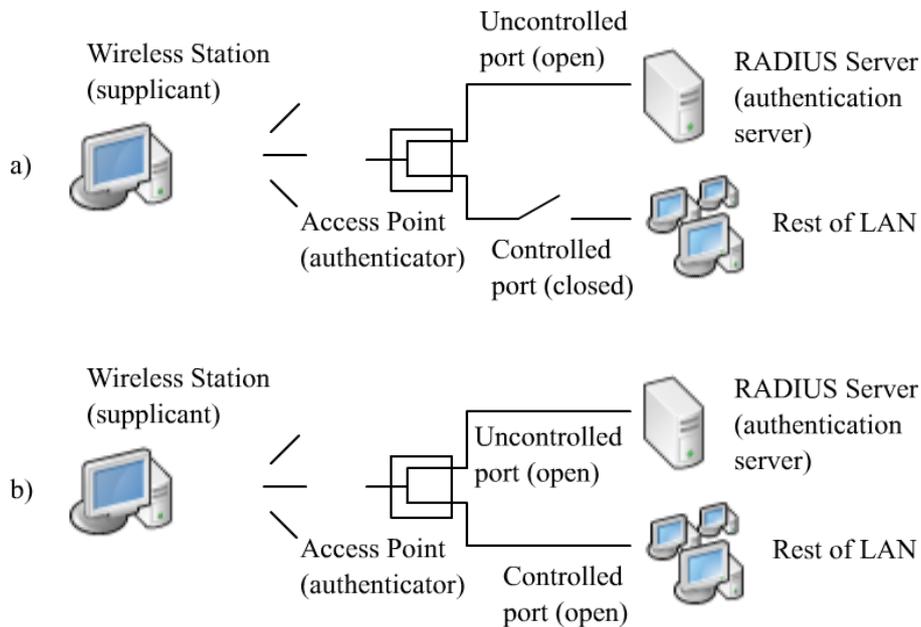


Figure 3: A typical LAN arrangement. In a) before the station has authenticated, the controlled port (used to access the LAN) is closed. In b) after the station has authenticated, the controlled port is open, and the station can access the LAN. The two ports are logical entities and both use the same physical connection to access the other LAN resources.

Before a station is authenticated, the only traffic permitted is EAP over LAN (EAPOL) traffic to the authentication server (Figure 3(a)). This essentially blocks users from accessing the network until they have been authenticated. A station that requests authentication contains what is known as a supplicant. A supplicant responds to authenticator data. An access point, which receives this authentication request, contains an authenticator which authorises access to the controlled LAN resources. The authenticator however does not necessarily need to reside in an access point [27].

Using an EAP method, the supplicant communicates with the authentication server, normally a RADIUS server. All communication is sent through the authenticator; the authenticator (which may be an access point) simply relays messages. Once the authentication is successful, the authenticator authorises

⁷<http://www.iana.org/assignments/eap-numbers> contains all of the registered EAP types.

access to the authenticated user. The user can then access the controlled resources on the rest of the LAN (Figure 3(b)).

When 802.1x is used with 802.11i, a key hierarchy is defined during the authentication process as follows:

Master Key (MK) A private symmetric key, that only a station and an authentication server can possess. An MK is bound to a specific session between a station and an authentication server. The MK is referred to as an AAA⁸ key in the 802.11i standard.

Pairwise Master Key (PMK) This key may be derived from the MK, or can use the PSK directly. Only a station and an authentication server can derive the PMK from the MK. A PMK is bound to a specific session between a station and an access point. Once the authentication server derives the PMK, it is moved to the authenticator (for example an access point).

The MK is not the same as the PMK, and an access point cannot derive the PMK, otherwise the access point could make access control decisions instead of the authentication server [28].

Pairwise Transient Key (PTK) It is derived from the PMK and a 4-way handshake between the supplicant (for example a station) and the authenticator (for example an access point), and is used between the station and the authenticator. The PTK is made up of a collection of keys:

- Key Confirmation Key (KCK): 128 bits - Used to prove the possession of the PMK and binds the PMK to a particular station and access point.
- Key Encryption Key (KEK): 128 bits - Used to distribute the Group Transient Key (GTK).
- Temporal Encryption Key: 128 bits - The key used to encrypt the traffic.
- Temporal Data Origin Authenticity Key: 128 bits - When TKIP is used, two 64 bit keys, one in each link direction, are used to encrypt the Message Integrity Code (MIC). The TKIP MIC is described later.

These temporal keys are unique to each station.

Group Transient Key (GTK) The key used to encrypt multicast/broadcast traffic. The access point creates a random GTK, encrypts it with the KEK and sends it to the station. This GTK is used by all stations associated with the access point.

This process provides secure key management, and the final GTK step completes the authentication process.

As stated before, EAP-TLS is an example of a secure mutual authentication method. Figure 4 illustrates the steps involved.

⁸Authentication, Authorisation and Accounting.

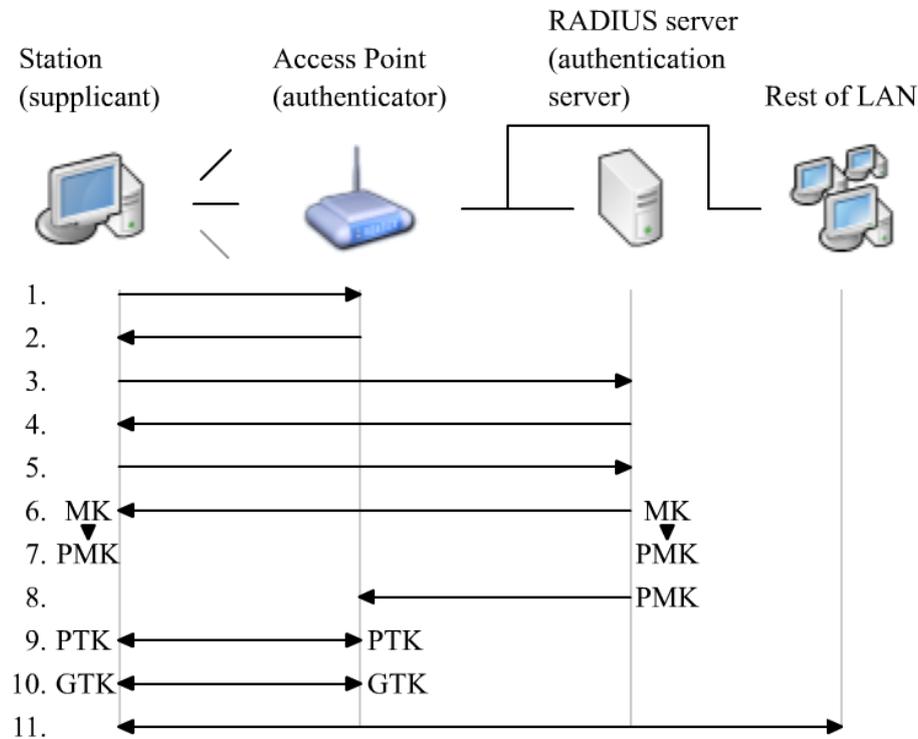


Figure 4: The steps used to authenticate a station under EAP-TLS.

1. The station associates with the access point.
2. The access point blocks all traffic from the station, and requests its ID.
3. The station responds, and the access point forwards this to the RADIUS server.
4. The RADIUS server retrieves the station's public key from a Certificate repository. The RADIUS server sends a challenge to the station (i.e. "Here is something encrypted with your public key. Decrypt it with your private key, perform a function on it and then return it encrypted with my public key").
5. The station responds to this challenge.
6. If the RADIUS server accepts the challenge, an MK is jointly negotiated between the station and the RADIUS server, and is sent securely from the RADIUS server to the station.
7. Both the RADIUS server and the station derive the PMK from the MK.
8. The RADIUS server moves the PMK to the access point.
9. The station and the access point use the PMK and a 4 way handshake to derive (not transport), bind and verify a PTK. The Temporal Encryption

Key part of this PMK is used during this session to encrypt communications between the station and the access point.

10. The access point uses another 4 way handshake and the KEK part of the PTK to encrypt the GTK and sends it to the station.
11. This completes the authentication process, and the access point grants LAN access to the station.

The session keys can be refreshed automatically, a huge improvement over WEP's manual rekeying. The access point can be configured to force stations to re-authenticate at regular intervals. This may be defined in terms of the number of seconds passed, or number of packets sent. This can also be deferred to the authentication server.

Under the PSK method (also known as WPA Personal mode), a key is manually entered into each device on the wireless network. The PSK is used directly as the PMK. WPA under PSK authentication is designed for small office/home office (SOHO) wireless networks that do not have authentication servers, but still want the security benefits that WPA has over WEP. A station is authorised to use the network only if the station's password matches the access point's password. A wireless network can be configured to have one single PSK for every device, or one PSK per station, which is more secure (for example against internal attacks).

Since the PSK method is simpler, it has some disadvantages that are not present when using 802.1x authentication. Since the keys are manually entered into stations, like WEP, if the PSK changes, it must be changed manually on each device on the wireless network. Since WPA-PSK is designed for smaller networks this is not much of a problem. It is not designed for large networks, as it has the same key administration problems as WEP. In addition, since the PSK is stored on the device itself, if the device is stolen or the PSK is ever compromised, every other device using the wireless network is compromised as well. The PSK used must be chosen carefully as wireless networks using WPA-PSK can be compromised using offline dictionary attacks [29].

The Temporal Key Integrity Protocol (TKIP) still uses an IV as well as RC4 in order to operate on the same hardware originally designed for WEP. However the IV is used differently to WEP. The IV has been extended to 48 bits, and is used as what is known as a TKIP Sequence Counter (TSC). The first 16 bits of the TSC are stored in the WEP IV field, while the remaining 32 bits are stored in a field known as the extended IV. As a result the protocol data unit is expanded to accommodate this additional field. The 48 bit TSC is an increasing counter initialised to 1 when the TKIP Temporal Key is initialised or changed. Each frame received must have a TSC greater than the TSC in the previous frame received from the same sender. This provides protection from replay attacks. The 802.11i standard states that once the TSC space is exhausted, communications must end, or a new temporal key generated, as reusing the TSC will mean key reuse (however frames that are retransmitted with the same key do not compromise security). However unlike WEP's 24 bit IV, the TSC space is 48 bits. This means over 280 trillion frames can be sent before all TSC values are used for a single temporal key. An access point operating at 54Mbps continuously sending 1500 byte packets will require almost 2000 years to exhaust the TSC space:

$$54 \times 10^6 / 8 = 6750000 \text{ bytes sent per second}$$

$$6750000 / 1500 = 4500 \text{ frames sent per second}$$

$$2^{48} / 4500 = 1983 \text{ years.}$$

Unlike WEP, where the encryption key was generated by concatenating the 24 bit IV to the 40 or 104 bit secret key, TKIP uses a two phase key mixing function to generate the key. The first phase combines the temporal key (from the key hierarchy), the transmit address and the 48 bit TSC, to generate an 80 bit TKIP-mixed transmit address and key (TTAK). The second phase combines the 80 bit TTAK again with the temporal key and the TSC, resulting in the final 128 bit key used to encrypt the current frame. This results in a per-packet key, unlike WEP which used a static shared key. This key mixing function is described in more detail in [5].

TKIP has a 64 bit Message Integrity Code (MIC) called Michael, to protect messages from being modified in transit. The MIC is calculated over the destination and source address, a priority field, three reserved octets and the entire plaintext message payload. The MIC key (Temporal Data Origin Authenticity Key section of the PTK) is used to encrypt this MIC. The 802.11i standard states that since TKIP was intended to be implemented on older and existing hardware designed for WEP, the MIC is not considered complete protection against message forgery [5]. To compensate for this, the MIC detects active attacks (unlike WEP's ICV) and countermeasures can be employed to prevent further attacks. The WEP ICV is still used in conjunction with the MIC to prevent false detection of MIC failures, and therefore false countermeasure initiation.

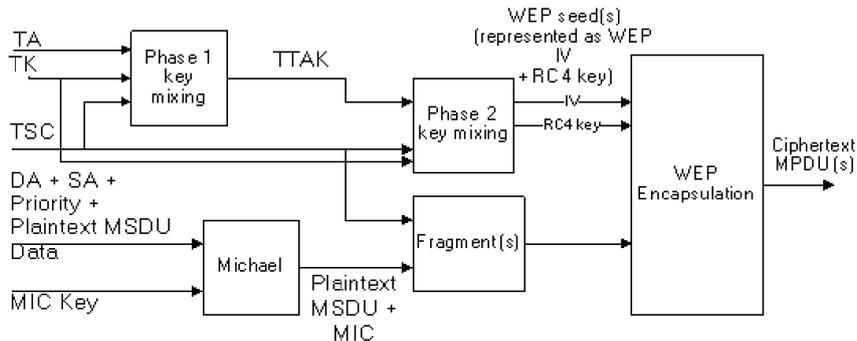


Figure 5: The WPA Encryption Process (from [5]).

Figure 5 (from [5]) illustrates how WPA works.

1. The TKIP MIC is computed and appended to the data field.
2. TKIP assigns the incremented TSC value to the frame.
3. The encryption key is generated using the two phase key mixing function.

4. The encryption key (represented as a WEP IV and an RC4 key), the plain text data and the appended MIC are passed to the WEP engine.
5. WEP generates an ICV, computed over the plaintext and MIC and appends this to the plain text, after the MIC.
6. The frame is encrypted by the WEP engine, and sent.

When the recipient receives this frame (Figure 6, from [5]) the following steps are performed.

1. TKIP extracts the TSC. If it is out of order (less than or equal to the current value of the receiver's counter) the frame is discarded.
2. The WEP key is generated using the two phase key mixing function, and is represented as a WEP IV and an RC4 key. These are passed to the WEP engine to decrypt the frame.
3. WEP checks the ICV. If it is valid, the MIC is computed over the frame and compared with the MIC in the message. If they are different then the frame is dropped and countermeasures are triggered. Otherwise the frame is passed onto the upper layer.

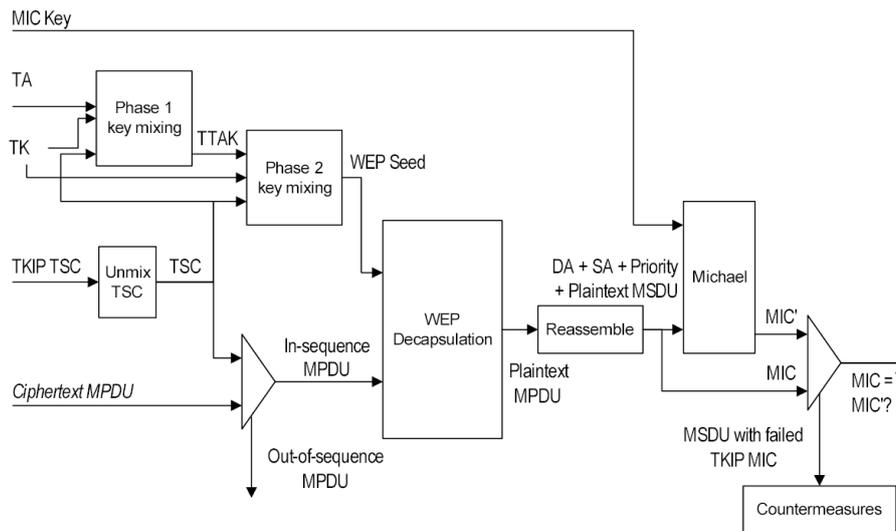


Figure 6: The WPA Decryption Process (from [5]).

Countermeasures are used as the 802.11i standard states that the MIC “provides only weak protection against active attacks”. When an MIC failure occurs within 60s of a previous MIC failure, the following happens:

- All stations/supplicants de-authenticate themselves.
- Access points/authenticators de-authenticate all stations/supplicants associated with it and disallows new associations for at least 60s after the second MIC failure.

- All devices stop transmitting and receiving TKIP encrypted frames for at least 60s after the second MIC failure. During this period devices are only allowed to send and receive unencrypted 802.1x messages.

Normal network operation can resume at least 60s after the second MIC failure. To prevent these countermeasures from being used as a base for a denial of service attack (since each device stops sending and receiving for a short period and then must re-authenticate and re-associate with an access point after), the MIC is checked last. Any frame that does not have a valid ICV and TSC are discarded before the MIC is verified. The ICV ensures that noise and transmission errors do not erroneously trigger the countermeasures.

WPA addresses the weaknesses in WEP by using TKIP for confidentiality and data integrity and 802.1x for authentication. One of the major problems with WEP was key reuse, caused by a small IV space. This has been addressed by the TSC. The TSC also provides protection from replay attacks. Shared secret keys are also no longer used directly in the encryption key. Instead, a key hierarchy is created, for example if a PSK is used, it is the dynamically generated temporal key (which is derived from the PSK) that is used to encrypt. The key hierarchy also provides a degree of key management, as the keys that are used to encrypt are derived from keys higher in the hierarchy. These higher layer keys are also dynamic as they are valid only for a particular session. The addition of an MIC and countermeasures improves protection against forgeries and message modifications. Finally authentication has been improved with PSK for small networks and 802.1x for large networks.

WPA has a few implementation issues. It may require a firmware upgrade on all wireless WEP devices on the network, which is time consuming and can also damage the hardware. Older WEP hardware may not even be able to support WPA. WPA may also be incompatible with some operating systems that do not contain supplicants (needed for both WPA modes of authentication). Like other networks, WPA is also vulnerable to denial of service attacks, especially due to the effects of the countermeasures. However additional validation checks are used to ensure the countermeasures are not falsely triggered.

2.3 Wi-Fi Protected Access 2

While WPA was designed within the constraints of existing hardware created to run WEP, Wi-Fi Protected Access 2 (WPA2) was designed from scratch. The complete 802.11i specification was approved in June 2004, and the Wi-Fi Alliance began interoperability testing for WPA2 in September 2004. WPA2 represents the complete implementation of the 802.11i standard. WPA2, like WPA, addresses confidentiality, access control and data integrity. WPA2 can use the IEEE 802.1x authentication framework (for WPA2 Enterprise mode), or PSK (for WPA2 Personal mode) in the same way as WPA to perform authentication.

WPA2 uses the Counter Mode with CBC-MAC Protocol (CCMP) instead of TKIP and is required for devices claiming Robust Security Network (RSN) compliance. The Advanced Encryption Standard (AES) algorithm is used for both encryption (in Counter Mode (CTR)) and data integrity (CBC-MAC) and both processes use the same temporal key from the key hierarchy.

Counter Mode (CTR) is the method of encryption used, and uses AES instead of the RC4 encryption algorithm. AES is a block cipher and is defined in [30]. AES was selected by the United States National Institute of Standards (NIST) to replace the Data Encryption Standard (DES and 3DES/Triple DES) algorithm [31]. It is also the preferred encryption method in the US Federal Information Processing Standards (FIPS) 140-2. Some organisations require compliance with FIPS 140-2 (such as government agencies), or desire security based on the complete 802.11i standard, and WPA2 was introduced for this reason. AES in CCMP uses 128 bit keys and operates on a block size of 128 bits. AES is significantly stronger than RC4; if a computer could try 2^{55} keys per second, it would require about 149 thousand billion years (the age of the universe is less than 20 billion years) to crack a 128 bit AES key [32].

Cipher Block Chaining with Message Authentication Code (CBC-MAC) provides for data integrity and like the MIC in TKIP, protects the sender and destination addresses as well as the frame data from modification. CCMP also uses a 48 bit Packet Number (PN) as part of the IV. The PN increments for each frame sent, and never repeats for a single temporal key. Like the TSC in TKIP, the PN provides protection from replay attacks and works in the same way. A frame with an out of sequence PN, or a frame which has a PN equal or smaller than the current value of the replay counter, is discarded.

Figure 7 (from [33]) illustrates how WPA2 works.

- The data in the frame is split into 128 bit blocks.
- To calculate the MIC, an IV is created by concatenating the Priority and Reserved bits, the address of the sender, the PN and other header data⁹. This IV is fed into an AES block, and an XOR operation is applied to the resulting key stream with specific elements in the header of the frame. The output of this is then XOR'd with the first 128 bit block of data. This continues for the entire length of the frame resulting in a 128 bit CBC-MAC value. The first 64 bits of this value are used as the MIC.
- The encryption process is initialised by the IV and a counter set to 1. The combined IV and counter value (known as the preload value) are fed into the AES block and an XOR operation is applied to the resulting key stream and the first 128 bits of data. The counter is incremented, and this continues for the length of the entire frame.
- The counter is set to 0, and fed into the AES block. An XOR operation is applied to the resulting key stream and the 64 bit MIC value.
- The encrypted MIC value is appended to the encrypted frame.
- The frame is sent.

Decryption is the reverse of this, with some additional steps. One of these is PN checking. The PN is extracted and if it is not greater than the current value of the replay counter, the frame is discarded. The last step is MIC checking. An MIC is calculated in the same way, and this MIC is compared to the MIC

⁹This additional header data is referred to as Additional Authentication Data (AAD) in the 802.11i standard.

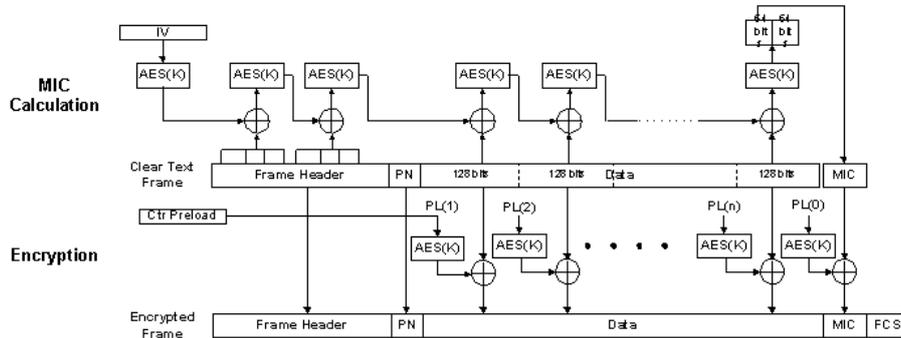


Figure 7: The WPA2 Encryption Process (from [33]).

in the received frame. If they are identical then the frame is passed on to the higher layers, otherwise it is discarded.

Unlike WPA, which activates countermeasures after an MIC failure, WPA2 does not invoke countermeasures if the calculated MIC does not match the received MIC. This is because while WPA was designed within the limits of WEP hardware and therefore constrained to its capabilities, WPA2 is completely new and is not designed in terms of WEP.

While WPA addresses the weaknesses in WEP adequately, it was designed to work on WEP hardware. WPA2 was designed without this constraint and as a result, uses AES, a stronger encryption algorithm than RC4. The MIC in WPA2, CBC-MAC, is also better than WPA's MIC. Like WPA, WPA2 uses 802.1x or PSK for authentication and uses a PN to protect against replay attacks. While there are no security flaws in WPA [8] (as long as a strong password is chosen when using the PSK mode), WPA2 avoids the design flaws of WEP altogether. WPA2 also satisfies organisations that desire wireless security based on the complete 802.11i specifications, or require FIPS 140-2 compliance.

Like WPA, WPA2 has a few implementation issues as well. WPA2 will require new hardware (for example access points and network cards) which can be expensive. Firmware upgrades may be possible on existing hardware, but not all hardware can be firmware upgradeable to WPA2. These upgrades are required to support the processing power required by AES. Instead of performing the AES encryption in hardware mode, the AES encryption that WPA2 requires may be performed in software mode, however this requires sufficient processing power (such as a Pentium 2.5GHz, [34]). Since WPA2 products have just been released, they have not had the benefits of rigorous testing, and as a result may have undiscovered loopholes that can be exploited.

These wireless network security methods are summarised in Table 1 (adapted from [28]).

2.4 Related Work

While there is a wealth of research around wireless network performance ([35, 36, 37, 38]), such work has not focused on the impact of various security mechanisms.

	WEP	802.11i Methods	
		WPA	WPA2
Security Protocol	WEP	TKIP	CCMP
Cipher	RC4	RC4	AES
Key Length	40 or 104 bits	128 bits encryption, 64 bits authentication	128bits
Key Life	24 bit IV	48 bit IV	
Key Generation	Concatenation	Two phase mixing function	Not needed
Data Integrity	CRC-32	Michael	CBC-MAC
Header Integrity	None	Michael	CBC-MAC
Replay Protection	None	Packet Number	
Key Management	None	EAP-based	
Authentication	Open or Shared Key	802.1x or Pre-Shared Key (PSK)	

Table 1: Summary of wireless network security methods (adapted from [28]).

Wong [39] investigated the effects of Virtual Private Network (VPN) and IEEE 802.1x security frameworks on network performance. Various configurations of each framework were evaluated. The general pattern found was that the stronger the security level, the lower the network performance. While VPN offered end to end security (compared to 802.1x, which provided client to access point security only), it was more complex to implement and had a larger impact on performance than 802.1x. This research was performed with one client sending to one client (representative of a single user, lightly loaded network).

Baghaei [9] extended this research by using multiple clients. It also evaluated the effects of packet length on the network throughput with different security mechanisms. This study showed that WEP encryption significantly reduced network performance when the network was congested. Network performance was also reduced as more clients were on the network under all security levels tested.

No previous research analysed the effects of IEEE 802.11i (WPA and WPA2) security specifications on network performance. This study builds on Wong's and Baghaei's research by focusing on the effects of this specification.

3 Research Goals

3.1 Motivation

WPA2 has two modes of operation: enterprise mode and personal mode (detailed in section 2.3). Enterprise mode may not be appropriate for all networks; for example, EAP-TLS, one of the more secure EAP authorisation types in the 802.1x specification, requires a RADIUS authentication server, as well as a system in place for digital certificate administration [40].

This infrastructure may be excessive for a small office or ad hoc network set up just to transfer a few files; the time required to set up the required infrastructure would take longer than the actual data transfer itself. WPA2 would be used in pre-shared key mode (PSK) in this situation.

It is not known how much faster or slower WPA2 is in comparison to existing specifications (WEP and WPA). Knowing this may influence the design of wireless networks, so each configuration will have an appropriate balance between performance and security.

3.2 Objectives

The aim of this research is to uncover and compare the effects of the 802.11i security standard on wireless network performance with existing security configurations (WEP and WPA). This will be achieved by measuring the throughput and latency while traffic is pushed through the network. The traffic will consist of data generated by a traffic generator, as well as traffic generated by typical applications, for example a large file FTP transfer.

A combination of both ‘synthetic’ data and ‘real world’ application data will be used as traffic. This provides a more complete picture of the performance than just relying on one form of measurement, as there may be factors which cause the two measurements to be different. The measurements taken from the application data would be an example of the performance the average user can expect, while using ‘synthetic’ data enables full control over the characteristics of the traffic.

Both of these traffic types will also be used to evaluate the existing security specifications in order to make comparisons, which range from no security and the two types of WEP, right up to the complete WPA2 specification with AES. It may also be of interest to see what effects the high performance hardware built for AES will have on the existing security specifications.

4 Experimental Design

4.1 Network Setup

The topology used to perform the experiments is illustrated in Figure 8. The server was a Pentium 4 2.4GHz with 512MB RAM, running Microsoft Windows Server 2003 Enterprise Edition with Service Pack 1. This server operated as a RADIUS server via Microsoft's Internet Authentication Service.

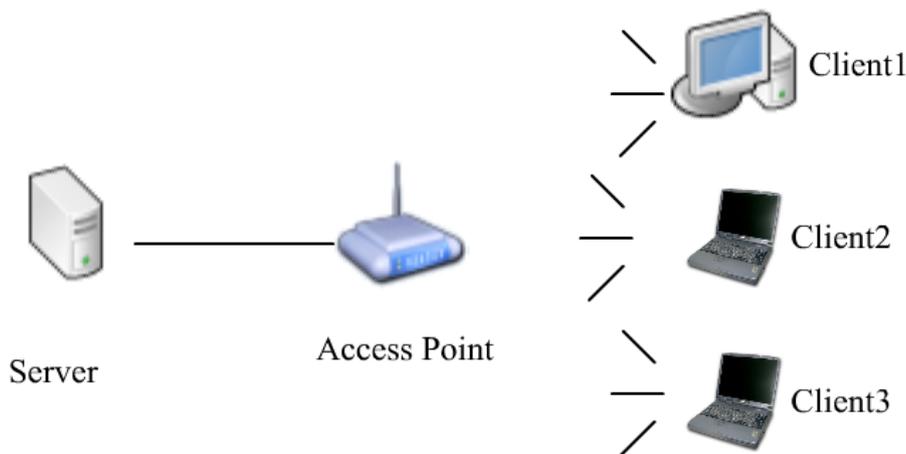


Figure 8: The topology used to perform the experiments.

A Cisco Aironet 1130AG series access point was connected to the server via a 100Mbps Ethernet connection. The access point was operating in the 802.11a 5GHz 54Mbps mode. The 5GHz frequency range is less crowded than the 2.4GHz range (for example some cordless phones and Bluetooth devices use the 2.4GHz range), and is therefore less susceptible to interference. Noise generated by other electronic devices also tend to affect 2.4GHz frequencies more than 5GHz frequencies. However [41] shows there is little difference between 2.4GHz and 5GHz in terms of propagation in an indoor office environment. The Cisco 1130AG has hardware acceleration for both AES and TKIP. Re-authentication on the access point was disabled. There are many ways to define the re-authentication level (such as in terms of seconds passed, or packets sent), and the 1130AG could only define it in terms of seconds passed. This was undesirable, as any impact of re-authentication overhead would best be described as a percentage of traffic sent: in terms of packets.

Client1 was an Athlon 800MHz with 768MB RAM, running Microsoft Windows XP Professional with Service Pack 2. The wireless adapter for Client1 was a Cisco Aironet 802.11a/b/g wireless PCI card (PI21AG). The Cisco Aironet drivers were used to configure the connection, rather than the Windows Network Connection Properties. Dedicated drivers offer more control over the adapter than the more general Windows options.

Client2 was a Compaq Evo N160; a Pentium 3 M 1.2GHz with 256MB RAM. Client3 was a Hewlett Packard Compaq nx9040; a Pentium M 1.6GHz with 480MB RAM. Both Client2 and Client3 were notebooks, and each used a

Cisco Aironet 802.11a/b/g wireless CardBus adapter (CB21AG). The OS and wireless network adapter on both Client2 and Client3 were configured the same as Client1.

As all of the wireless network devices used in the experiment were Cisco based, vendor interoperability problems were avoided. The wireless network adapters in the 3 clients were based on the Atheros AR5212 and AR5112 chipsets, which, like the access point, has hardware accelerated encryption¹⁰. These configurations are summarised in Table 2. None of the computers were running any processes which could affect processor or network utilisation.

System	Processor	RAM	Interface
Server	Pentium 4 2.4GHz	512MB	PCI 100mbit ethernet
Client1	Athlon 800MHz	768MB	PCI Cisco Aironet
Client2	Pentium 3 M 1.2GHz	256MB	CardBus Cisco Aironet
Client3	Pentium M 1.6GHz	480MB	CardBus Cisco Aironet

Table 2: Summary of Systems used in the experiment.

As the access point and the server were the only devices directly connected to each other, the network was isolated from other LANs. A scanning utility verified that there were no other wireless networks close enough to be a direct source of interference. All of the hardware was in a single room to ensure that walls and other obstacles would not be a factor in the measurements. All hardware locations were kept the same for each experiment, and no hardware was moved during the running of the experiments. Antennas were positioned no closer than 1.5m to each other since [42] states some wireless products have low data rates at very close ranges, as they restrict signals that are too strong.

While not every wireless network can be deployed in such a controlled manner, the number of factors that may affect results is reduced. This enables analysis to be limited to the effects of the security specifications on the performance of wireless networks. Limiting the experiments to a base case, ideal set up also facilitates comparisons with other network scenarios.

4.2 Security Levels

The following seven security layers were used to test the performance of the wireless network.

1. **No Security:** This is often the ‘out of the box’ configuration of most wireless network devices.
2. **WEP shared key authentication and 40 bit encryption:** This is the original 802.11 security specification.
3. **WEP shared key authentication and 104 bit encryption:** This improves on WEP by using a larger key.
4. **WPA with PSK authentication and RC4 encryption:** This uses WPA under the pre-shared key method of authentication.

¹⁰<http://www.atheros.com/pt/AR5002XBulletin.htm>

5. **WPA with EAP-TLS authentication and RC4 encryption:** This uses certificates for mutual authentication. It was expected that the throughput would be similar to the PSK method of authentication, as the authentication phase is completed at the start of the session, and more importantly, re-authentication was disabled.
6. **WPA2 with PSK authentication and AES encryption:** This uses WPA2 under the pre-shared key method of authentication.
7. **WPA2 with EAP-TLS authentication and AES encryption:** This uses certificates for mutual authentication. As with WPA-EAP-TLS, it was expected that this method of authentication would result in similar throughput, as authentication is completed when the client completes association with the access point. This has been included for completeness.

The first layer (No Security) was tested as a base case in order to ascertain whether the different security layers reduce throughput by adding overhead.

4.3 Network Measurement Methodology

As described in Section 3, this project's main aim was to ascertain the extent to which the security mechanisms reduce throughput in a wireless LAN. A traffic generator was used to generate traffic on Client2 and Client3 to send to Client1. Several traffic generators were evaluated, and these included Iperf¹¹, NetPerf¹² and IPTraffic¹³. Ethereal¹⁴ was used to analyse whether each tool generated traffic according to the parameters given. IPTraffic was the only tool to generate reliably consistent traffic based on the parameters. These parameters are detailed in Section 4.4.

Three sets of experiments were conducted. Each experiment was repeated 8 times. To reduce the effects memory caching and disk paging, the first three readings were not recorded, leaving five readings in each cell. As stated before, no other applications, such as Ethereal, were running when these tests were run.

The first set were synthetic benchmarks. IPTraffic was configured as described in Section 4.4. This test consisted of Client2 sending to Client1 at each of the seven security layers, recording throughput. This represented the base case of one client sending to one client. This test was repeated, but with both Client2 and Client3 sending to Client1. This test represented multiple clients associated with the access point, to create a loaded network. Due to a lack of wireless adapters, the speed of the access point was reduced from 54Mbps to 24Mbps (only for this test) to represent a loaded network with multiple clients. While adding more clients to an access point reduces throughput for each client, this test would indicate whether the security layers were affected differently by a loaded situation.

The second set of experiments was similar to the first set. This experiment consisted of Client2 downloading an 11.2MB video file from Client1 via FTP. Client1 was running Cerberus FTP server¹⁵. While synthetic benchmarks are

¹¹<http://www.dast.nlanr.net/Projects/Iperf>

¹²<http://www.netperf.org>

¹³<http://www.zti-telecom.com/pages/iptraffic-test-measure.htm>

¹⁴<http://www.ethereal.com>

¹⁵<http://www.cerberusftp.com>

useful as they test only a specific aspect (in this case network throughput) they are not representative of ‘real world’ applications. File transfer is a popular application of networks, and this test gave insight into one aspect of ‘real world’ throughput and whether the security layers affected it.

The third set of experiments was again a synthetic test, measuring latency and errors as reported by IPTraffic. Errors as reported by IPTraffic were the sum of lost and out of sequence packets. This test was performed with each of the seven security layers. Here, latency and errors were recorded.

4.4 Traffic Generator Configuration

The traffic generator used in this study was IPTraffic. It is a software testing tool that can generate, receive, capture and replay traffic. It can also be used to measure end to end traffic performance. IPTraffic was configured as follows:

Total Number of Packets: Preliminary tests were run with packet numbers ranging from 5000 and 100,000. The number of packets did not affect the throughput. The experiments were conducted with Client1 receiving a total of 20,000 packets (20,000 from Client2 during the single sender tests, and 10,000 from Client2 and Client3 each, during the multiple client tests), chosen as an arbitrary value.

Outgoing Bandwidth: No bandwidth limiting software was used, nor was there any other process running which used the network. This meant the tool was configured to push as much traffic as possible through the interface.

Traffic Protocol: TCP is the predominant transport protocol used on the internet [43], while UDP is used for time sensitive applications, such as voice over IP. With the number of applications which stream data increasing, the proportion of UDP traffic on the internet may change. Accordingly both TCP and UDP were used in the tests. No other transport protocols were used.

TCP Windows: The TCP windows were left at 8192 bytes. While the TCP window size has an effect on the throughput of a network, 8192 bytes is the default for Windows XP. Leaving the TCP window at 8192 bytes means only the effect of the security levels was being tested. Not all applications can change the TCP window size, and this experiment aimed to give an idea of a typical, un-tuned, base case scenario.

Traffic Type: The content of the traffic generated by the traffic generator was not important to the running of the tests. A lot of time and effort was spent considering the size of the packets generated. It was important to use a traffic profile that would be representative of typical network behaviour. However this was difficult to characterise; what was a typical network traffic profile? This depended on many factors, such as purpose of the network, the time of day and week, as well as the types of users and applications using the network. Replaying traffic captured from the University’s LAN was considered briefly, however this was not explored further due to the reasons above. Attempting to characterise a wireless network traffic profile was beyond the scope of this project.

Research [44, 15] indicates that packet sizes on the internet are bi-modally distributed. The research shows that the two most frequently occurring packet sizes are 40 bytes (which are typically header only TCP acknowledgement packets in response to data received) and 1500 bytes (which corresponds to the maximum ethernet MTU size). The remaining traffic is distributed between these extremes depending on how much data remains in the last fragment of the data transfer.

The packet payloads used in the synthetic tests were 6 and 1460 for TCP and 1472 bytes for UDP. A payload of 6 bytes was used, as this was the minimum IP traffic would use without crashing. A payload of 1460 and 1472 bytes correspond to the maximum payload sizes of TCP and UDP respectively. The UDP tests did not include a small packet payload, as UDP does not have acknowledgement packets. While this model was not the most realistic, it was simple enough to be used by many benchmarking tools¹⁶. It also makes the tests more repeatable, as while replaying real, captured traffic would be the most realistic, the repeatability of such a study would depend on the availability of the identical capture file.

In the second set of synthetic benchmarks, where latency and errors were examined, packet sizes were again fixed, but to 1460 bytes for both TCP and UDP. Since throughput was not being measured, the payload size was kept constant in order to limit any effects it may have on error rate and latency.

While the tool used was able to test with duplex connections (where both clients send to each other), for the scope of this project, each connection was configured to send in one way only. Results were analysed using ANOVA at the 95% confidence interval.

¹⁶Most network benchmarking tools such as those evaluated for use in this study accept maximum packet size as a parameter.

5 Results and Discussion

5.1 Effects on Throughput

In the first set of synthetic benchmarks, Client2 was configured to send data as fast as possible to Client1. In Figures 9, 10 and 11, the one to one bars show the mean throughput for each of the security levels defined in section 4.2. With the TCP transfers, the differences are statistically significant for both the 6 byte payload (Figure 9, one to one); ($F_{6,24} = 21.28, p < 0.01$) and the 1460 byte payload (Figure 10, one to one); ($F_{6,24} = 11.09, p < 0.01$). While statistically significant, in practical terms, the difference is not large; across all security levels, the mean throughput is 83.06Kbps (s.d. 1.09) and 8.12Mbps (s.d. 0.26) for the 6 and 1460 byte TCP payloads respectively. A standard deviation of 0.26Mbps corresponds to approximately 33.28KB/s. This variance is not large when considering the mean is 8.12Mbps, which corresponds to 1015KB/s. It is also typical of a wired Ethernet LAN [45]. However it is interesting to note that both the WPA-PSK and WPA-EAP-TLS security levels have lower throughputs than the WPA2 security levels for both the 6 and 1460 byte TCP, one to one client transfer.

There are no statistically significant differences in throughput between the various security levels in the one client to one UDP transfer (Figure 11); ($F_{6,24} = 0.997, p = 0.45$). The mean throughput is 9.14Mbps (s.d. 0.10).

These results differ hugely from [39] and [9], which found stronger security levels decreased the throughput. Here, it is clear that applying the different security levels has little tangible effect on the throughput, and is discussed in section 5.4.

These results are reproduced in the second experiment, the FTP transfer (Figure 12), where Client2 downloaded an 11.2 MB file from Client1. There are no statistically significant differences between the security levels ($F_{6,24} = 2.00, p = 0.11$). The mean throughput across all security layers is 9.73Mbps (s.d. 0.29).

5.2 Effects with Multiple Clients

Similar results are found when this test was repeated with multiple clients. In this situation, Client2 and Client3 were configured to send as fast as possible to Client1. In Figures 9, 10 and 11, the two to one bars show the mean throughput of both senders, for the three traffic configurations. Moving from one sender to two senders effectively halves the outgoing throughput from each sending client. The combined throughput of both clients is lower than the throughput of one client in the previous test, as the access point is limited to 24Mbps (for this test only) as detailed in section 4.3. The differences between the security levels are again statistically significant for both the 6 byte TCP payload ($F_{6,24} = 32.70, p < 0.01$) as well as the 1460 byte TCP payload ($F_{6,24} = 2.82, p < 0.05$). As with the previous test, in practical terms this difference is not large; across all security levels, the mean throughput is 78.08Kbps (s.d. 1.57) and 6.91Mbps (s.d. 0.30) for the 6 and 1460 byte TCP payloads respectively. In the 6 byte TCP multiple client transfer, the throughputs of both WPA security levels are again slightly lower than the WPA2 throughputs.

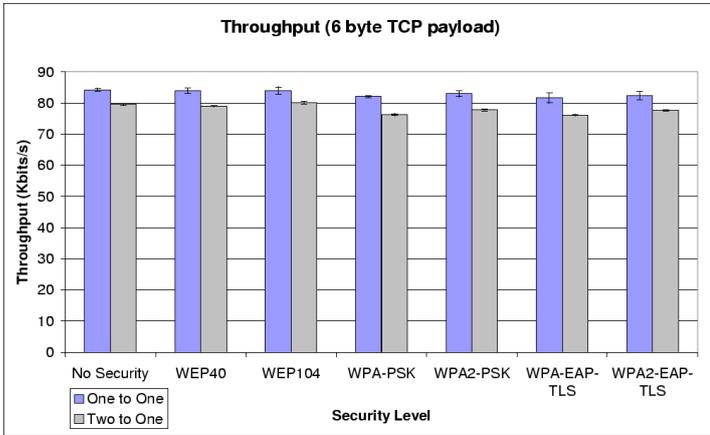


Figure 9: Throughput with a 6 byte TCP payload. Two to one is the combined throughput of both sending stations.

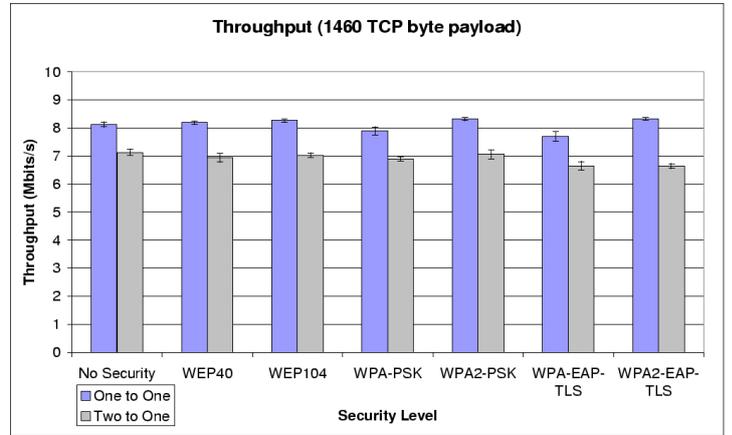


Figure 10: Throughput with a 1460 byte TCP payload. Two to one is the combined throughput of both sending stations.

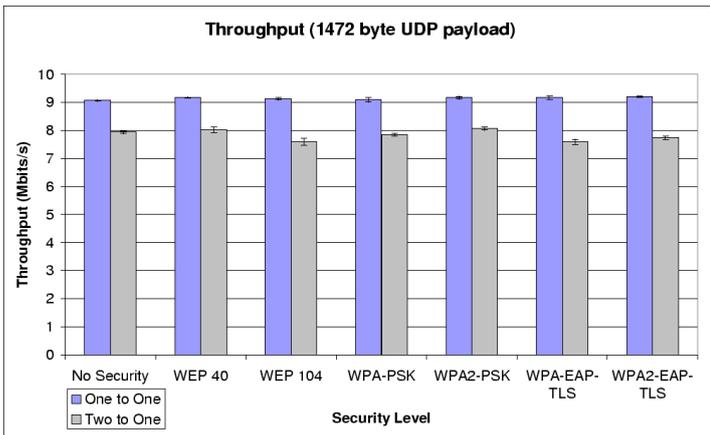


Figure 11: Throughput with a 1472 byte UDP payload. Two to one is the combined throughput of both sending stations.

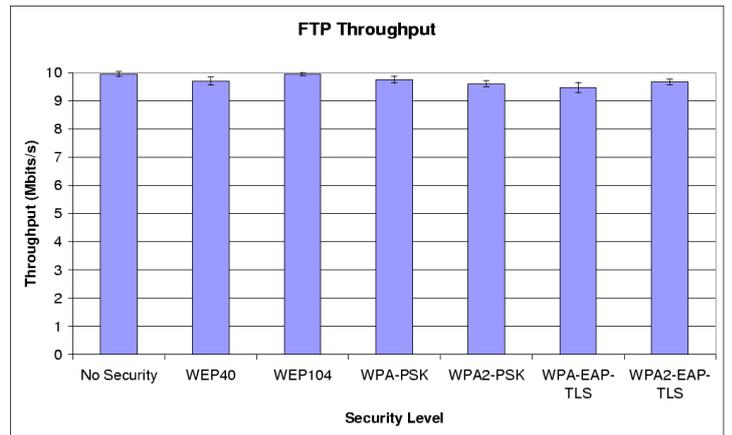


Figure 12: Throughput with the FTP file transfer.

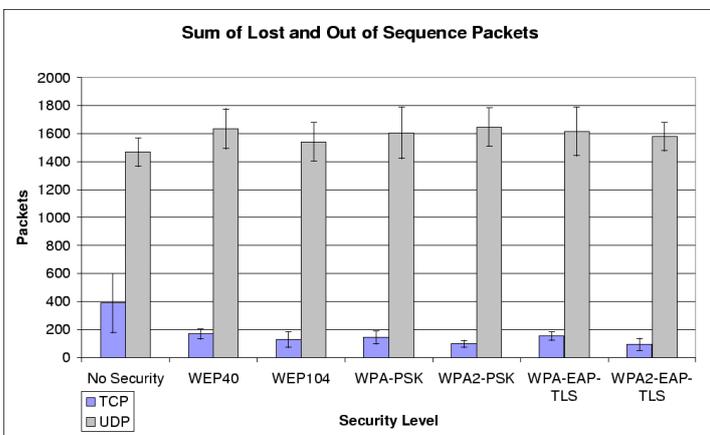


Figure 13: Sum of Lost and Out of Sequence Packets.

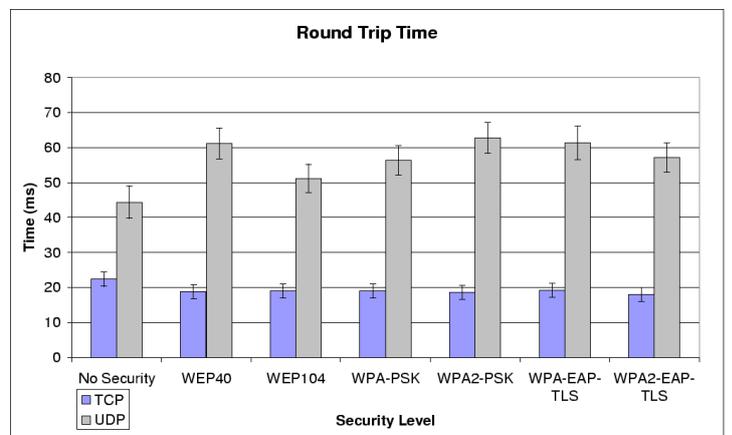


Figure 14: Mean Round Trip Time.

The UDP transfer with multiple clients also yielded statistically significant differences between the various security levels ($F_{6,24} = 5.77, p < 0.01$). Much like the TCP tests with multiple clients, this difference is not a practical difference; across all security levels, the mean throughput for UDP with multiple clients is 7.83Mbps (s.d. 0.24). Here the WPA2 throughputs are also slightly higher than the WPA throughputs.

ANOVA reports statistically significant interactions between the seven security levels and the two levels of client transfer (one client sending to one and two clients sending to one) for all three traffic configurations. This means the various security levels have statistically different effects on the throughput depending on the number of clients on the network.

For the 6 byte TCP transfer by number of clients, the interaction is significant at the 0.05 level ($F_{6,24} = 3.12, p < 0.05$). Figure 9 shows that the 802.11i security levels (WPA-PSK and above) have slightly larger reductions in throughput compared to the others when there were more clients on the network (i.e. moving from the one client sending to one transfer, to the two clients sending to one transfer). However at this speed, the overall reduction in throughput across all security levels is around 5Kbps, or 625B/s, and small variations on this is clearly not a huge difference.

For the 1460 byte TCP transfer by number of clients, the interaction is also significant ($F_{6,24} = 3.20, p < 0.05$). Figure 10 shows the differences in throughput between one to one and two to one transfers. The WPA2-EAP-TLS security level has the largest decrease in throughput as a result of the multiple client test; it drops by 1.68Mbps, while the other security levels drop by an average of 1.21Mbps. Unlike the previous statistically significant differences, which were realistically insignificant, this difference may have an effect in real performance terms. This test shows that WPA2-EAP-TLS drops by almost half a megabit more than the other security levels when more clients are on the network. This half a megabit difference (equivalent to approximately 60KB/s) corresponds to roughly 7% of the WPA2-EAP-TLS throughput with multiple clients (6.64Mbps or 850.18KB/s). Whether this difference is really significant depends on the typical usage of the LAN.

For the 1472 byte UDP transfer by number of clients, the interaction is significant at the 0.01 level ($F_{6,24} = 5.95, p < 0.01$). While there is no statistically significant difference between the security levels with only one client sending, there are when there are two clients sending. This indicates the throughputs are affected differently by the security levels depending on the number of clients when using UDP. WPA-EAP-TLS and WEP fall the most, by 1.58Mbps and 1.53Mbps respectively, while WPA2-PSK falls the least, by 1.10Mbps.

5.3 Effects on Packet Errors and Round Trip Times

In the second set of synthetic tests, packet loss and the round trip time (RTT) were measured. Client2 was configured to send time-coded packets to Client1. Figure 13 shows the sum of sequence and lost packets at the application level, for both TCP and UDP as reported by IPTraffic.

For TCP, the differences in errors (lost packets and out of sequence packets) between the different security levels are statistically significant ($F_{6,24} = 6.09, p < 0.01$). The two WPA2 security levels have slightly lower errors than the other security levels. It is not known what caused the relatively large variance and

errors with the No Security level. The mean number of errors is 166.51 packets (s.d. 101.45).

With UDP, the differences in errors between the security levels are not significant ($F_{6,24} = 0.96, p = 0.47$). Error rates with UDP are on average 11 times that of TCP; the mean number of errors is 1584.63 packets (s.d. 62.22).

Figure 14 shows the mean RTT for both TCP and UDP at each security level. For TCP, the differences are statistically significant ($F_{6,26} = 7.71, p < 0.01$). With the exception of the No Security level, all of the security levels have a mean RTT of between 18 and 19 ms; No Security has a mean RTT of 22.4 ms. Within each security level, the jitter (the mean two way variance as reported by IPTraffic) is 2 ms and is indicated by the error bars in Figure 14.

With UDP, the differences in RTT between the security levels are not significant ($F_{6,24} = 1.81, p = 0.14$). The mean RTT for UDP is 56.37 ms. Within each security level, the mean jitter is 4.37 ms and is also indicated by the error bars in Figure 14.

5.4 Discussion

These experiments show that wireless network throughput is largely unaffected when the various security levels are applied. This is in stark contrast to findings in [39] and [9], where throughput fell as more complex security was used. For the single client transfers, the differences between the security levels are statistically different in the two TCP transfers (labelled one to one in the graph legends in Figures 9 and 10). These differences are not considered realistically significant, and the fact that there are no statistical differences found between the security levels in the single client UDP (Figure 11) and the FTP transfer (Figure 12) is a further indication that any statistical differences between the security levels is not of practical significance.

Where the security levels are statistically different, the WPA throughputs are lower than the respective WPA2 throughputs. This suggests that the chipset in the WLAN adapters has better support for AES/CCMP than RC4/TKIP. This is most noticeable in the one to one 1460 byte TCP transfer. The transfers with two senders also support this; in the UDP and 6 byte TCP transfer, the WPA2 throughputs are slightly higher than the corresponding WPA throughputs.

As [9] identified, the implementation of the security specification has a large impact on the throughput performance. In [9], the wireless LAN client adapters performed the encryption and decryption in the firmware. The client adapters used in this experiment performed the encryption and decryption in the hardware, and this explains the minimal performance degradation when security is used.

In terms of security level by number of senders interaction, they are statistically significant in all three transfers. Across all three transfers, the throughputs of the WPA and WPA2 security levels fall slightly more than the lower security levels (No Security, WEP40 and WEP104). This is noticeable in the 1460 byte TCP transfer, where WPA2-EAP-TLS falls by almost half a megabit more than the other levels. This interaction is also noticeable in the 1472 byte UDP transfer; with another client sending, it appears that the processing capacity of the WLAN client adapter is reached for certain security levels, resulting in different levels of degradation. However since only two senders at most were tested, it cannot be assumed that the 802.11i security level throughputs (in particular

WPA2-EAP-TLS) will continue to fall more than the other security levels as more clients are added.

Nevertheless, while the interaction may be realistically significant, the differences between the security levels in the two senders transfer are not hugely significant in real terms.

Error rate is also largely unaffected in real terms by the security levels. With TCP, all of the security levels have similar error rates, with the exception of No Security, which has a large range of values. Here, the WPA2 levels have fewer errors than the WPA levels, adding further support that the hardware used has better provision for WPA2 than WPA. No statistical differences were found with the UDP transfer, providing further evidence that the different security levels have little real effect on the throughput. The error rate with UDP is much higher than that of TCP as TCP uses congestion control mechanisms to reduce packet loss. Errors are reduced at the application level as TCP can request retransmissions when errors are present at the TCP level. These retransmissions at the TCP level are not represented in the graph. Since UDP has no such method available, the receiver ends up receiving more data than it can process and as a result, packets are dropped.

Round trip times show a similar situation, with the TCP times effectively the same (again with the exception of No Security) and the UDP times showing no statistically significant differences.

In terms of authentication type, both [39] and [9] found that the 802.1x authentication methods resulted in a large performance degradation. This is clearly not the case here; as stated previously, the security levels have little real effect on the throughput. Overall, the EAP-TLS security methods have similar throughputs to the other security levels, and in several instances, cannot even be statistically differentiated from the other levels. However these experiments were performed with dynamic rekeying disabled and this must also be taken into consideration. It is expected that as the re-authentication and dynamic rekeying frequency during a session increases, the performance will decrease, due to the additional overhead.

These results show that it is possible to have a wireless network with robust security measures in place, without any real compromise in performance.

5.5 Limitations

There are several limitations with these experiments. Some of these provide avenues for further work, and are described in section 6.

One limitation is that the clients used to send and receive were not of uniform hardware specification. Processor speeds ranged from 800Mhz to 1.6GHz. While previous research ([37]) suggests that the processor of the client has less of an impact on network performance than the implementation of the 802.11 standard on the wireless adapter, ensuring the clients were of uniform hardware specification would guarantee that the throughput was independent of client hardware specifications.

Another limitation is that the multiple client tests were only approximations. This was due to a lack of suitable client WLAN adapters. The access point was limited to 24Mbps, rather than the full speed of 54Mbps, to approximate the congestion of multiple clients connected to the access point (the single sender

tests were performed at full speed however). This approximation may have different results compared to the equivalent test with multiple real clients.

IPTraffic was the only software used in the synthetic tests. Other packet generators may give different throughputs. However while the absolute values may change, the relative values are expected to stay the same; if the security levels have relatively similar throughputs using IPTraffic, it is expected that the throughputs will also be relatively similar using another packet generator. The FTP transfer supports this; the relative FTP throughput values are realistically similar to the single sender 1460 byte TCP transfer (the closest synthetic test), while the absolute values for the FTP throughput are almost 2Mbps higher.

With FTP being the only ‘real’ network application used, error rates for this transfer could be considered more valid than the error rates reported by IPTraffic. However Ethereal crashed each time it captured data, which meant the error rates and retransmissions for an FTP transfer were not analysed.

Re-authentication and dynamic rekeying was disabled. This is an important feature of the 802.11i security specification. However not all access points can change the re-authentication interval; some access points have this period hard coded into them. Others can specify the re-authentication period in terms of packets passed, while others specify it in the number of seconds passed. Normally an access point would want this feature enabled, however since these tests were aimed at revealing the base relative performance differences between the security levels, periodic re-authentication was disabled.

Finally, these results are only representative of an ideal base case infrastructure mode set up. Radio and physical interference will have an effect on the performance. The type of client adapter and access point will also have an effect on the throughput, as detailed in [37].

6 Future Work

This research has accomplished its objective of comparing the 802.11i security specifications with existing security configurations with respect to wireless network performance. However, since the network setup used in this experiment represented an ideal base case scenario, different or more complex scenarios can be explored. These different conditions paired with the various security levels can be used to evaluate their effects on wireless network performance.

Such conditions include scenarios which are more realistic, such as including radio and physical interference. Devices could also be moved while transmitting, to see movement has an effect on network performance.

In this experiment, all of the network hardware used were Cisco products. Further experiments could analyse whether connecting devices from different vendors would result in throughput changes with the security levels applied.

Related to this would be using client adapters which did not have hardware acceleration for encryption and decryption. Such devices sometimes rely on the main CPU for processing. With faster CPUs, network performance degradation may be small as well, even without hardware acceleration. CPU usage with the various security levels was not measured in this experiment, and this could be taken into account in future work.

Using multiple clients on the network (rather than approximating, as done in this experiment) is another avenue for further work. Due to a lack of WLAN client adapters, this experiment was not able to discover the effects with more than two sending clients. An experiment with multiple senders associated with an access point would reveal whether the 802.11i security levels (WPA and WPA2) would continue to degrade at a slightly higher rate than the lower security levels.

Re-authentication and dynamic rekeying was disabled in this experiment. Enabling this with the interval specified in terms of packets sent would identify the effect on network performance. The performance could be evaluated with different re-authentication intervals, in order to determine the optimum balance between performance and security. This could also be combined with multiple clients in order to analyse the interaction; multiple clients re-authenticating frequently would definitely have an impact on network performance.

This experiment used only one network benchmarking tool (IPTraffic) and one network application (FTP). Other software could be used to confirm the relative performance differences. Other protocols and applications could be used as well, such as HTTP and other transaction centred protocols. Related to this is measuring the error rates and latency of the real protocols or applications. This was not performed in this experiment as Ethereal crashed while capturing traffic during an FTP transfer.

Another opportunity for further work is to evaluate authentication using other 802.1x authentication methods. This experiment used EAP-TLS, which uses digital certificates to mutually authenticate the user with the authentication server. Future work could compare the various 802.1x EAP based authentication methods. This could be combined with the multiple clients by re-authentication frequency experiment described earlier. This would reveal any multiple client by re-authentication frequency by authentication type interactions. In other words, it could find out which 802.1x authentication type is fastest, based on number of clients as well as re-authentication frequency.

7 Conclusion

This research project aimed to discover the effects of the 802.11i security specification on wireless network performance, and compare it with existing security configurations.

The results show that while there are statistically significant differences between the security levels in some situations, it is unlikely that users will notice these small differences. The differences are so small that in under some conditions, the various security levels cannot even be statistically differentiated. All of the security levels, ranging from No Security to WPA2-EAP-TLS had realistically similar throughput, latency and error rates under all transfers.

The results also indicate that the WLAN client adapters and access point have an effect on the performance. Both the client adapters and access point used in this study had hardware accelerated encryption and decryption, resulting in minimal network performance degradation when security was enabled. Where the security levels have been statistically different, the WPA2 levels generally have better performance than the WPA levels, and this suggests the particular hardware used has better support for WPA2 than WPA.

Future work includes investigating the effects on throughput with multiple clients, as well as enabling re-authentication and dynamic rekeying. Other 802.1x authentication methods could be used in combination with this, in order to discover how the re-authentication frequency and number of clients affects network performance. Performing these tests with different hardware as well as software or network applications will also provide further insights into the effects of the security levels on wireless network performance.

This research shows it is possible to establish a wireless network with robust security, without any noticeable compromise in performance. Using modern hardware with hardware accelerated security features, there is no reason to use anything less than the complete WPA2 security specification with AES/CCMP encryption.

References

- [1] “Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.” ANSI/IEEE Std 802.11, Institute of Electrical and Electronics Engineers, 1999 (Reaffirmed 2003).
- [2] “Securing Wi-Fi wireless networks with today’s technologies.” Wi-Fi Alliance White Paper, http://www.wi-fi.org/OpenSection/pdf/Whitepaper_Wi-Fi_Networks2-6-03.pdf, February 2003.
- [3] B. Brown, “802.11: the security differences between b and i,” *IEEE Potentials*, vol. 22, pp. 23–27, October - November 2003.
- [4] H. Boland and H. Mousavi, “Security issues of the IEEE 802.11b wireless LAN,” in *Canadian Conference on Electrical and Computer Engineering, 2004*, vol. 1, pp. 333–336, May 2004.
- [5] “Amendment 6: Medium Access Control (MAC) Security Enhancements.” ANSI/IEEE Std 802.11i, Institute of Electrical and Electronic Engineers, 2004.
- [6] “Wi-Fi Protected Access Q&A.” Wi-Fi Alliance Document, http://www.wi-fi.org/OpenSection/pdf/Wi-Fi_Protected_Access_QA.pdf, December 2004.
- [7] B. Potter, “Wireless security’s future,” *IEEE Security & Privacy Magazine*, vol. 1, pp. 68–72, July - August 2003.
- [8] “WPA2 Q&A.” Wi-Fi Alliance Document, http://www.wi-fi.org/OpenSection/pdf/WPA2_Q_A.pdf, March 2005.
- [9] N. Baghaei and R. Hunt, “Security performance of loaded IEEE 802.11b wireless networks,” *Computer Communications, Elsevier, U.K.*, vol. 27, no. 17, pp. 1746–1756, 2004.
- [10] “Initialization vector.” Wikipedia, http://en.wikipedia.org/wiki/Initialization_vector, July 2004.
- [11] J. Geier, “802.11 WEP: Concepts and Vulnerability.” Wi-Fi Planet, <http://www.wi-fiplanet.com/tutorials/article.php/1368661>, June 2002.
- [12] H. Berghel and J. Uecker, “Wifi attack vectors,” *Communications of the ACM*, vol. 48, pp. 21–28, August 2005.
- [13] X. Gu, “Wireless LAN attacks and vulnerabilities,” University of Canterbury, August 2004.
- [14] L. Barken, *How Secure is Your Wireless Network? Safeguarding Your Wi-Fi LAN*. Prentice Hall, 2003.
- [15] C. Williamson, “Internet Traffic Measurement,” University of Calgary, November 2001.
- [16] D. Laverty, “Why is wep crackable?.” Open Xtra, <http://www.openxtra.co.uk/articles/wep-weaknesses.php>, Accessed June 2005.

- [17] “What’s Wrong With WEP?.” Interopt White paper, http://www.ilabs.interop.net/LANSec/papers/14_Whats_wrong_with_WEP-LV04.pdf, 2004.
- [18] S. Weatherspoon, “Overview of ieee 802.11b security,” in *Intel Technology Journal*, Intel Corporation, May 2000.
- [19] P. Coleman, “The Encryption Balancing Act.” Computer Publishing Group, <http://swexpert.com/webserver/features/f2/2.6/>, July 1997.
- [20] T. Newsham, “Cracking WEP Keys.” http://lava.net/~newsham/wlan/WEP_password_cracker.pdf, 2001.
- [21] H. Cheung, “The Feds can own your WLAN too.” TomsNetworking, <http://www.tomsnetworking.com/Sections-article111-page1.php>, March 2005.
- [22] D. Laverty, “Why is WEP crackable?.” OpenXtra, <http://www.openextra.co.uk/articles/wep-weaknesses.php>, April 2004.
- [23] N. Borisov, I. Goldberg, and D. Wagner, “Security of the wep algorithm.” www.isaac.cs.berkeley.edu/isaac/wep-faq.html, 2001.
- [24] L. Loeb, “What’s up with WEP?.” IBM, <http://www-106.ibm.com/developerworks/library/s-wep/?article=wir>, April 2001.
- [25] “Port-Based Network Access Control.” IEEE Std 802.1X-2004, Institute of Electrical and Electronic Engineers, 2004 (Revision of IEEE Std 802.1X-2001).
- [26] B. Aboba, L. Blunk, J. Carlson, E. Levkowitz, and J. Vollbrecht, “Extensible authentication protocol (eap).” RFC 3748, June 2004.
- [27] L. Strand, “802.1X Port-Based Authentication HOWTO,” in *The Linux Documentation Project*, <http://www.tldp.org/HOWTO/8021X-HOWTO/>, October 2004.
- [28] N. Cam-Winget, T. Moore, D. Stanley, and J. Walker, “IEEE 802.11i Overview,” in *NIST 802.11 Wireless LAN Security Workshop*, http://csrc.nist.gov/wireless/S10_802.11i%20Overview-jw1.pdf, December 2002.
- [29] R. Moskowitz, “WLAN Testing Reports “PSK as the Key Establishment Method”.” ICSA Labs, https://www.icsalabs.com/icsa/docs/html/communities/WLAN/wp_PSKStudy.pdf, November 2003.
- [30] “Advanced Encryption Standard (AES) (FIPS PUB 197).” National Institute of Standards and Technology (NIST), November 2001.
- [31] “Building A Secure Wireless Network.” Atheros Communications White Paper, http://www.atheros.com/pt/whitepapers/atheros_security_whitepaper.pdf, April 2004.
- [32] “Advanced Encryption Standard (AES) Questions and Answers.” National Institute of Standards and Technology (NIST), <http://csrc.nist.gov/CryptoToolkit/aes/aesfact.html>, January 2002.

- [33] D. Eaton, "Diving into the 802.11i Spec: A Tutorial." CommsDesign, http://www.commsdesign.com/design_library/cd/hn/0EG20021126S0003, November 2002.
- [34] "Wi-Fi Protected Access, WPA2 and IEEE 802.11i." Cisco Q&A, http://www.cisco.com/application/pdf/en/us/guest/products/ps430/c1167/cdcont_0900aecd801e3e59.pdf, 2004.
- [35] B. Bing, "Measured performance of the IEEE 802.11 wireless LAN," in *Conference on Local Computer Networks, 1999. LCN '99*, pp. 34–42, October 1999.
- [36] L. Chandran-Wadia, S. Mahajan, and S. Iyer, "Throughput performance of the distributed and point coordination function of an IEEE 802.11 wireless LAN," in *IEEE International Conference on Computer Communications (ICCC)*, August 2002.
- [37] A. Vasani and A. U. Shankar, "An empirical characterization of instantaneous throughput in 802.11b WLANs," Department of Computer Science, University of Maryland, September 2002.
- [38] G. Xylomenos and G. C. Polyzos, "TCP and UDP performance over a wireless LAN," in *Eighteenth Annual Joint Conference of the IEEE Computer and Communication Societies. Proceedings. IEEE*, vol. 2, pp. 439–446, March 1999.
- [39] R. Hunt, J. Vargo, and J. Wong, "Impact of security architectures on wireless network performance," in *5th IEEE International Conference on Mobile and Wireless Communications Networks (MWCN 2003)*, October 2003.
- [40] "Wi-Fi protected access: Strong, standards-based interoperable security for today's Wi-Fi networks." Wi-Fi Alliance White Paper, http://www.wi-fi.org/OpenSection/pdf/Whitepaper_Wi-Fi_Security4-29-03.pdf, April 2003.
- [41] D. Dobkin, "Indoor propagation and wavelength," tech. rep., WJ Communications, September 2002.
- [42] "Methodology for Testing Wireless LAN Performance." Atheros Communications White Paper, www.atheros.com/pt/atheros_benchmark_whitepaper.pdf, 2003.
- [43] M. Fomenkov, K. Keys, D. Moore, and K. Claffy, "Longitudinal study of internet traffic in 1998-2003," in *WISICT '04: Proceedings of the winter international symposium on Information and communication technologies*, pp. 1–6, 2004.
- [44] C. Shannon, D. Moore, and K. Claffy, "Characteristics of fragmented ip traffic on internet links," in *IMW '01: Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement*, pp. 83–97, 2001.
- [45] D. R. Boggs, J. C. Mogul, and C. A. Kent, "Measured capacity of an ethernet: myths and reality," *SIGCOMM Comput. Commun. Rev.*, vol. 25, no. 1, pp. 123–136, 1995.