Analysis of Wireless Key Management and Proposal

Michael Kwan, Andrew Ong and Derrick Yeung, *University of British Columbia*

***Abstract*—The 802.11 wireless security standard of using Wireless Equivalent Privacy (WEP) protocol as its authentication and encryption scheme has introduced security flaws and setup hassles for users. These drastically lowered the usability of WEP. New designs such as WPA, 802.1x, WEP*, TinyPeap and higher layer key management try to solve the usability issues with WEP's key management mechanism. In this paper, these new designs are analyzed. Built upon the knowledge obtained from these analyses of existing designs, a new key management system Wireless Public Key Protocol (WPKP) that provides robust key management mechanism for wireless network with the help of public key infrastructure is introduced.**

***Index Terms*—Mobile communication, Public key cryptography, Wireless LAN**

I. INTRODUCTION

In IEEE 802.11b, the Wireless Equivalent Privacy (WEP) claims to have the same level of security protection as wired networks, which should satisfy the basic objects of computer security: they are to ensure confidentiality, integrity, and availability. In this paper we will discuss the vulnerabilities that exist in the WEP security mechanism and how these vulnerabilities affect the usability of WEP, especially its key management aspect. Our main objective is to analysis different improvements to key management that have been developed and also to propose an alternative approach to replace the WEP key management mechanism. The existing WEP is an encryption checksum that prevents attackers from modifying packets, and authenticates and authorizes access to wireless networks. As we will illustrate how previous work has shown ways to retrieve the WEP key, these approaches have become threats and violate confidentiality and integrity. An adversary with a retrieved key can intercept messages and drop packets into a network. In addition, when a mobile station has revoked the right to access a wireless network, all the stations that are still on the network have to be reconfigured with a new WEP key. It greatly affects the systems availability during the setting up of the new key. Wireless network technologies have become so popular recently and 802.11b, which uses WEP, have been widely used in home or office networks. Therefore, it is important to address the threats in WEP and how may the mechanism be improved as our ultimate objective. However, since the main concern of this paper is on the usability wireless key management mechanism, we will just briefly describe the security issues inherit in WEP.

II. WEP ISSUES

A. Security Flaws

The protocol replies on an encryption algorithm and a WEP key, k, which is shared among an access point and mobile stations to establish secure connections. In the payload potion of a WEP packet, the message, M, is append to its checksum, c(M) to produce {M, c(M)} where , denotes concatenation. For each packet, it is assigned with a 24-bit Initial Vector (IV) and it is concatenated with the WEP key, k. The WEP key is either a 40-bits or a 128-bits key. RC4 stream cipher is then employed with the concatenated key to generate output bytes which then exclusive-ored with {M, c(M)} to produce the encipher text.

WEP utilizes RC4 stream cipher to provide security measures. The RC4 stream cipher consists of two major components, the key scheduling algorithm (KSA) and the pseudorandom byte generator. The key scheduling algorithm takes the concatenated key of the IV and WEP key as inputs and generates a state array. The pseudorandom byte generator then uses the state array to generate the pseudorandom byte sequence. The detailed algorithms for the KSA and the pseudorandom byte generator are shown in Fig. 1. Fluhrer et al. [1] presents a partial key exposure attack on the WEP protocol and shows how WEP key can be found in a short period of time. Ioannidis et al. [2] extends the work further to illustrate how might the actual implementation would be like and how techniques can be applied to further shorten the time for the WEP key recovery. As Arbaugh et al. [3] states, the attack of the WEP protocol can be prevented if

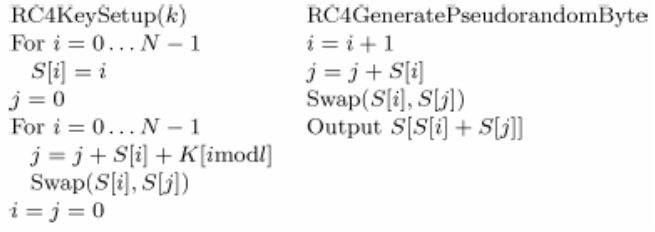

Fig. 1. Algorithms for the KSA and the pseudorandom byte generator.

the WEP key is frequently updated.

B. *Key Management of WEP*

As illustrated in the previous section, the WEP protocol is very weak in protecting a wireless network. Therefore it is essential for the network administrator to frequently change the WEP key. Unfortunately, the 802.11 standard does not offer a flexible mechanism to facilitate this operation.

The current key management scheme WEP employs is to manually distribute and input the WEP key in mobile stations in order to establish communication links among the stations and an access point (AP). Once a WEP key has been established and properly distributed, it is unlikely that the key would be changed regularly. The mechanism raises a security problem as Arbaugh et al. [3] suggested; human interventions can create threats since the WEP key is exposed to the users. Furthermore, within the IEEE 802.11b standard, there are two methods for WEP key distribution. The first method is a shared key method. Each mobile station is assigned with four different WEP keys. Each key can decrypt cipher text from the access point. However, there is one key available for transmission among stations. In fact, sharing WEP key would require changing the key of all stations when a mobile station is revoked from a network. By changing the WEP key manually for all stations would create overhead for users. The second method is using a key mapping table. Each MAC address is assigned with a separate entry in the table. Different WEP keys are assigned among the entries. Although the second method may resolve the revocation problem, both methods constrained by the need for frequent WEP key update.

III. EXISTING IMPROVEMENTS

A. *IEEE 802.1x*

The series of weaknesses and flaws had led to vendors of wireless products to provide a better method for security. Since there are numerous IEEE 802.11 wireless device already out in the market, the industry concentrated on improving the security mechanism while still using the IEEE 802.11 chipset which only supports WEP [5].

The vendor proposed combining two different protocols, Extensible Authentication Protocol (EAP) and IEEE 802.1x, to resolve the lack of convenient way of updating the shared key [5]. These two protocols together provide the framework for mutual authentication and session key delivery for both the access point base stations and wireless clients.

The IEEE 802.1x is a port-based, access control framework for wired or wireless networks that decides whether a client is authorized to use the network access service and then implements the decision [5]. Three entities exist in the IEEE 802.1x, supplicants (wireless clients who wish to use the network), authenticators (access points or base stations, separating the client from the network), and an authentication server (which grants/denies access to the network) [5].

IEEE 802.1x defines Extensible Authentication Protocol over LAN (EAPOL) for communication between the supplicant and the authenticator over the WLAN. When a wireless client (supplicant) wants to use a wireless network, the client first sends an EAPOL (Extensible Authentication Protocol over LAN) start message indicating interest of using the network. The access point (authenticator) responded by asking for the client's identity. The client sends its identity information to the access point. Upon retrieval of the identity information, the access point forwards it to the authentication server to see if access is granted [5].

All authentication messages exchanged are protected by encryption methods such as the Transport Layer Security

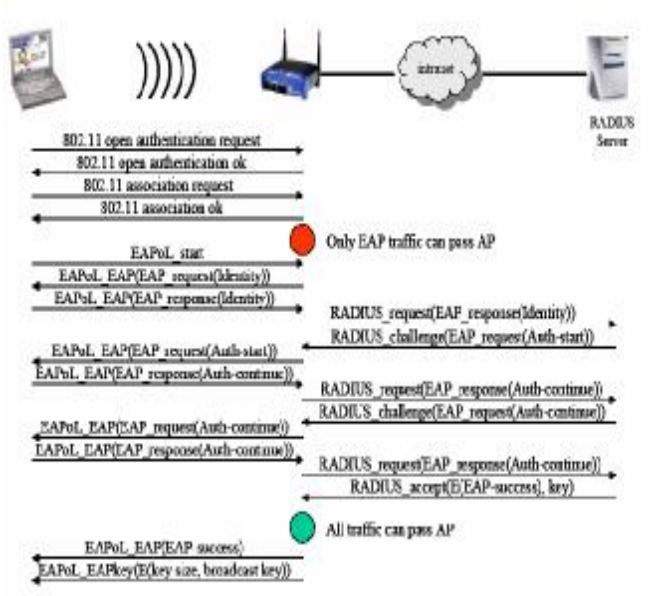

Fig. 2. 802.1x-based authentication and key generation [7]

(TLS). See Fig. 2. to see the complete message exchange.

The authentication phrase is completed; however, the connection is still susceptible to session high jacking. The authentication server, RADIUS server will generate a session key for a particular supplicant and authenticator to prevent session high jack [5].

B. *WIFI PROTECTED ACCESS (WPA)*

Before new hardware for 802.11i is available, new security method such as WIFI Protected Access (WPA) is introduced to partially implement some of the standards of 802.11i into the current 802.11 hardware. Firmware/Software updates to clients and access points allow WPA to interoperate with the hardware [6].

The fore-mentioned 802.1x and EAP together is required by WPA. There is two phrase to the WPA authentication process.

a. Open system authentication authenticates a wireless client to an access point.

b. There are two modes for 802.1x/EAP in WPA

1. Authenticate the user against authentication server as describe in section IEEE 802.1x (for enterprise) [6].

2. Authenticate the user by a pre-shared key (PSK) mode for small wireless networks which does not have an authentication server (home/small office) [6]

WPA still utilizes the RC4 algorithm but with temporal key integrity protocol (TKIP).

The TKIP enhances the WEP via the following [6]:

A 48-bit initialization vector (IV) is used and is separated from the overall key, which increases the key space to more than 500 trillion combinations. New sequencing rule is also applied [6].

An 8-byte message integrity code (MIC) protect the communication from a replay and man-in-the-middle replay attack [6].

Dynamic keying mechanism is used where a random key is automatically generated and distributes to each client. Each frame generated has a new key [6]

WPA also improves the way of using keys. Instead of using a single shared key between supplicant and authenticator, it utilizes four 128-bit keys to protect each communication [5]. A pair of key is used to protect the data and integrity of the data. Another pair is used to protect the initial handshake of the communication. The above keys are known as the Pairwire Transient Keys (PTK) for one to one communication; Group Transient Key (GTK) for one to many communication. As mentioned in the encryption section, these keys changes for every data packet sent [5].

Although WPA already provides a lot of improvement on top of WEP, there still remains weakness in WPA mechanism. WPA is vulnerable to dictionary attacks where passphrase of less than 20 characters in PSK [6].

C. Higher Layer Key Management System

Alternative approaches to the IEEE 802.11b WEP key management system have been proposed at a higher layer than the data link layer in OSI (Open System Interconnection) reference model. Arbaugh et al. [3] implements such mechanism at the application layer. In order to prevent an adversary to recover WEP key after collection of packets, short key period will prevent such an attack. However, in order to meet the IEEE 802.11b standard at the data link layer, Arbaugh et al. [3] proposes a higher layer key management solution. In addition, after a mobile station is authenticated to a wireless network, the station is assigned with a session key. The session key remains unchanged as long as the station stays connected. Therefore, there is need to update the key regularly at a fixed time interval even during a session. In Arbaugh el al. mechanism, DHCP is used to achieve the objective of allocating WEP key dynamically and also permit updates to the WEP key regularly. When a mobile station joins a wireless network, it obtains an IP address from a DHCP server and at the same time WEP key can be assigned by the DHCP server with the IP address request. In addition, the IP address for a mobile station remains valid till the lease expires. Prior to the expiry date of the lease, the IP address has to be renewed for which the DHCP can also provide a new WEP key. Although higher layer implementation does solve the key management problem in the IEEE 802.11b standard, the solution is practically infeasible since most small office networks do not have DHCP servers.

*D. Key Refresh and Host Revocation with WEP**

Instead of relying on application layer to manage WEP keys, Wool [4] presents another approach which does not rely on external authentication servers. The approach is called WEP*, which is claimed to be fully compatible with the existing IEEE 802.11b standard. In order to achieve short key period, an access point (AP) changes its WEP key periodically. It is the AP's responsibility to transfer new key to its hosts whatever the WEP key get updated. This scheme permits host revocation possible. After a mobile station is revoked from a network, it will not receive any updated keys. In addition, the updated key is generated in such a way that the sequence of the new keys is unpredictable. Furthermore, WEP* uses the IEEE 802.11 authentication protocol to transport the updated keys to the hosts securely. WEP* achieves transport security by having the hosts to individually share a long-term key with an access point (AP). The long-term keys for all the hosts are kept in the AP. The AP compares its storage of keys with the key of a host before it transmits its current WEP key to the host. Unfortunately, as Wool [4] admits, there are no specific implementations to show how WEP* can be incorporated into the IEEE 802.11b standard. Therefore, there is no way to guaranty that WEP* has efficiently improved the key management scheme in the widely used wireless standard.

E. TINYPEAP

The TinyPeap intent is to help the administrator in a way such that he does not need to physically update keys when a user/client no longer belongs to the wireless network. The TinyPeap is intended to target small wireless networks. Since the design of TinyPeap is similar to the design that will be proposed in this paper, we are going to look at it with greater details.

The current WPA model involves an authentication server which is separate from the base station. Home wireless network or small office wireless network may not have the resource to purchase an authenticate server to perform the authentication process. The modification made in TinyPeap is to have an authentication server built-in within the base station. Based on this model, wireless users can be authenticated based on username/password combination presented in the base station. When clients are being removed from the network, it will not require a physical key change; instead users can be removed from the authentication server

through the base station user interface.During the progress of the project, a topic in the www.broadbandnetworks.com forum titled "WPA cracker" caught the attention. It leads to a link to a homepage title "TinyPeap". The TinyPeap project matches the proposed solution of a built-in authentication server within wireless router/access point.

After the "TinyPeap" is integrated into a router's firmware (the router is limited to linksys's WRT54G/GS as of the date of this report), the router will have an interface of adding/modifying/removing users [10]. The system administrator can then manage users through this interface.

When an unauthenticated wireless connects to the access point, the access point automatically starts an 802.1x session by requesting the client its identity and password. The

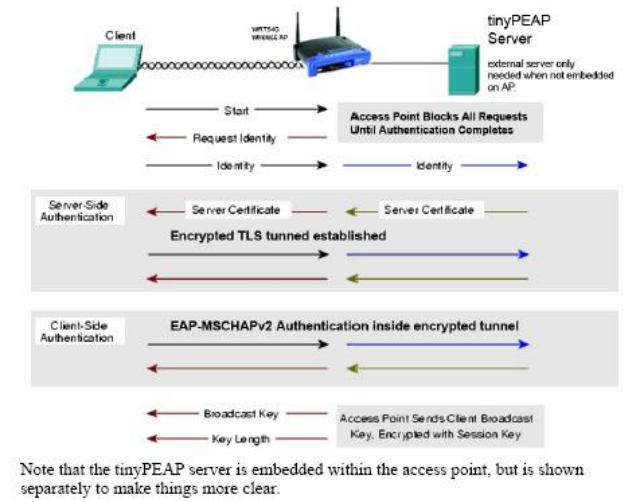

Fig. 3. Complete authentication process [10]

unauthenticated client will then send its userId and password to the router. Upon retrieval of the userId/password, the built-in small radius server, the password is hashed and compared with the hashed copy of the password. If the user is found to be valid, the router will initiate a TLS (SSH) connection the client sending the router's own unique signed certificate. [10]

After the client authenticates the access point's certificate (users can determine if the certificate is valid), it will complete the initiation of the TLS connection and begin the key exchange. The client will calculate a new pair of keys to be used for further communication. This new pair of key is sent back to the server encrypted using the user's userId/password. The server will derive the same password so that the key will not be resend through wireless connection. The TLS connection is ended and the client and router begin to communicate using the new keys. A key expiring and message count flags are associated with this new pair of keys. When the key is within the expiring time or the number of messages sent using this key is within a certain number, the re-keying process begins the same as initialization of the keys [10].

1) Advantage

The advantages of using the TinyPeap project are as follows:

a. It can associate the connection based on the user not the machine. Wireless adapters can use the machine's logon credentials provided by the user to initialize the wireless connection.

b. It disassociates the dependency of the client with a pre-shared key.

Since the key is disassociated with the common shared key, when users are removed from the system, the system administrator is only required to remove/disable user account in the access point.

a. It provides a quick fix to the weaknesses of the WEP mechanism.

b. It uses technologies currently available and does not require introduction of new techniques or technology.

2) Disadvantages

The disadvantage of a userId/password based system is always human factor. The authentication server is vulnerable to brute-force attack. To help resolving the issue, security policies is to be in place so that the choice of passwords and expiration dates of the passwords are set.

IV. PROPOSED SCHEME – WIRELESS PUBLIC KEY PROTOCOL

In this paper, we are going to propose a new key management scheme called wireless public key protocol (WPKP). WPKP replaces the role of the problematic WEP in providing security of a wireless network with a small scale public-key infrastructure (PKI). Similar to TinyPeap, an authentication server is built into a wireless access point (AP). The wireless access point acts as a certificate authority (CA) of the PKI and an authentication server, while all wireless clients are PKI clients. The access point serves the responsibility of issuing digital certificates to all wireless clients within the wireless network. Together with a client's media access control (MAC) address, a digital certificate assigns a unique identification to the wireless client. In order to gain access through an access point, the wireless client must be authorized and hold a certificate issued by the access point. Due to the scope of the project, we are only going to explain the key management aspect of WPKP.

A. Overview of WPKP Key Management

Key management is a pivotal factor that determines the usability of a wireless security mechanism. The limited key management capability of WEP, which was caused by the lack of definition in the 802.11 standard, is one of the major reasons that discourage the use of WEP. As a result, many of the wireless networks deployed either use a permanent fixed WEP key or no encryption key at all [11]. This phenomenon leaves many wireless networks vulnerable; therefore, the

TABLE I
KEY EXCHANGE MESSAGE SEQUENCE

Communication	Description
WC → AP: $\{request, req\_key\}_{AP\_Public}$	The wireless client submits a registration request.
WC ← AP: $\{\{app, sess\_key\}_{AP\_Private}\}_{req\_key}$	The access point approves the registration request.
WC → AP: $\{ack\}_{sess\_key}$	The client acknowledges the approval message.
WC ← AP: $\{digital\ certificate\}_{sess\_key}$	The access point transfer the digital certificate to the client
WC → AP: $\{transfer\ ack\}_{sess\_key}$	The client acknowledges the completion of transfer.

WPKP will deeply address this issue.

Unlike WEP, which has been designed to employ up to 4 static symmetric encryption keys [12], WPKP uses public and private key pairs, in the form of digital certificates. The WPKP key management involves exchange of digital certificates to newly joined clients, authorizing or revoking wireless clients' certificated, and updating expired certificates.

Before becoming part of a WPKP enabled wireless network, a wireless client needs to first register itself with the access point. The registration process begins by submitting a request to the access point. At this point, the wireless client can only be identified by its MAC address, since the client is new to the network and does not own any certificate recognized by the access point. Along with the request, a random request key is also sent to the access point for authentication and encryption purpose in the future. To protect the confidentiality of this request, the whole request message is encrypted with the public key of the access point; thus, only the access point can decrypt and read the request. Once the access point receives the registration request, the network administrator can approve or reject the request through the administration mechanism of the access point. If rejected, the request will be dropped and the corresponding wireless client will not be able to join the network, although a registration request may be submitted again later. If the request is approved, the access point will generate a new digital certificate for the wireless client and initiate a transfer sequence (see Table I) to securely send the new digital certificate to the client.

First, the access point sends an approval message back to the client. This message is first encrypted with the private key of the access point and then further encrypted by the random request key sent earlier with the request by the client. The reasons of encrypting the approval message with the random request key are to protect he message's confidentiality and to authenticate the client. Since the random request key is shared only among the access point and the wireless client, no one else besides the wireless client will be able to read the approval message. Similarly, the encryption with the access point's private key is for authenticating the access point to the client; being able to decrypt with the access point's public key and obtain a valid approval message gives the wireless client confident that the message was indeed sent by the access point.

After the approval message is validated by the client, a trusted relationship is established between the access point and the client. Included in the approval message is a session key. From this point onwards until the end of the digital certificate transfer sequence, all subsequent communication will be encrypted with this session key. The client responses to the access point by sending back an acknowledgment message. Upon receipt of the acknowledgment message, the access point starts the transmission of the digital certificate. The client receives this certificate and stores it into memory. Finally once the transmission is completed, the client replies another acknowledgment message, signifying the end of certificate transmission and the end of the whole transfer sequence.

Besides key exchange, key authorization and revocation are two elemental operations of key management. In a secure wireless network, the network administrator must have absolute control of who is authorized to access the network and who is not. With the use of a single shared static key among all wireless clients, WEP provides no easy support for authorizing and revoking keys. In contrast, WPKP offers a robust key management that supports flexible key authorization and revocation.

The key authorization and revocation processes used in WPKP are based on what is being done in a normal PKI. Under X.509 PKI, all digital certificates are valid unless they are expired or have been revoked. A certificate revocation list (CRL) is utilized to keep a record of the list of certificates that are no longer valid [13]. Similarly in the case of WPKP, the access point maintains a CRL of its own. Any WPKP digital certificate that satisfies all of the following conditions will be considered authorized with respect to an access point:

a. the certificate is a valid WPKP certificate

b. the certificate is not expired

c. the certificate is issued by the access point being accessed

d. the certificate is not on the CRL of the access point being accessed

Access to the AP will be granted to any wireless client who can present an authorized WPKP certificate during the authentication process.

Finally, the last vital operation of key management is key updating. There are two different ways to update the certificate of a WPKP wireless client. The first method of updating, whether or not the existing certificate expired, is to

simply re-register the client with the access point. By doing so, a new WPKP certificate will be given to the client. In addition, the WPKP provides an auto-update mechanism that serves as an alternative means of key updating. With the auto-update function, the access point will automatically offer a new WPKP certificate to a client if the AP detects that the client's certificate will soon expire. The check for expiry date will be done when a client authenticates for a session. However, if a client's certificate has already expired, the client must go through the registration process again.

B. Usability of WPKP

It is very obvious that the usability of WPKP is far more superior to that of WEP. The client enrolment process involves only two steps – registration request submission and registration request approval – hence minimizing the amount of human interaction. Also, key revoking and updating become two very simple processes in WPKP. Without the need of physically changing the key stored in all wireless clients, these two operations can be performed by the administrator at the access point and are completely transparent to the wireless client user.

Moreover, WPKP has a better usability, not only than WEP, but some other existing WEP alternatives as well, such as IEEE 802.1x and the deployment method developed by Palo Alto Research Center (PARC). Like WPKP, 802.1x also uses PKI as a security mechanism. However, a study has shown that a total of 38 steps are required to enrol a client into an 802.1x wireless network [14]. Even though the enrolment procedure proposed by PARC requires only a single step, it demands the wireless client to be physically moved to the proximity of the enrolment station, which may not be feasible if the client is a desktop computer instead of a laptop.

REFERENCES

[1] S. Fluhrer, I. Mantin, and A. Sharmir, "Weakness in the key scheduling algorithm of RC4," *In Eighth Annual Workshop on Selected Areas in Cryptography*, 2001.

[2] J. Ioannidis, A. D. Rubin, and A. Stubblefield, "A Key Recovery Attack on the 802.11b Wired Equivalent Privacy Protocol (WEP)," *ACM Transactions on Information and System security*, vol. 7, no. 2, pp. 319-332, May 2004.

[3] W. A. Arbaugh, N. Shankar, and K. Zhang, "A Transparent Key Management Scheme for Wireless LANs Using DHCP," Technical Report, Hewlett-Packard Laboratories, Palo Alto, Sep. 2001.

[4] A. Wool, "Lightweight Key Management For IEEE 802.11 Wireless LANs With Key Refresh and Host Revocation," Tel Aviv University, ISRAEL, July 2002.

[5] Lusheng Ji, Brian Feldman, Jonathan Agre, "Self-Organizing Security Scheme for Multi-hop Wireless Access Networks", http://www.flacp.fujitsulabs.com/Aerospace04-51.pdf, last accessed Nov 29, 2004

[6] Kevin Sacco, "WLAN Security: Insecurities of the WEP protocol and how WPA plans to overcome them", http://www.giac.org/practical/GSEC/Kevin_Sacco_GSEC.pdf, last accessed Nov 29, 2004

[7] Hui Luo, Paul Henry, "A Secure Public Wireless LAN Access Technique That Supports Walk-Up Users", http://ieeexplore.ieee.org/iel5/8900/28134/01258471.pdf, last accessed Nov 29, 2004

[8] Bruce Potter, Wireless Security's Future, *IEEE Security and Privacy*, pp 68-72, July/August 2003

[9] "Airsnort Homepage", http://airsnort.shmoo.com/, last access Nov 29, 2004

[10] Brian Lee, Jim Gruen, Takehiro Takahashi, Wenke Lee, and Richard Lipton, "TinyPeap", last accessed Nov 29, 2004, http://www.tinypeap.com/docs/TinyPEAP_White_Paper.pdf

[11] W. A. Arbaugh, N. Shankar and J. Wan, "Your 802.11 Wireless Network has No Clothes," Dept. Comp. Sci., Univ. of Maryland, Maryland, United States, 2001. Available: http://www.cs.umd.edu/~waa/wireless.pdf

[12] "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," ANSI/IEEE Standard 802.11, 1999 Edition, 1999.

[13] M. Bishop, *Computer Security*, Pearson Education, Boston, MA, 2003, pp. 245-272.

[14] D. Balfanz, G. Durfee, R. E. Grinter and D. K. Smetters, "In Search of Usable Security: Five Lessons from the Field," *IEEE Security & Privacy*, vol. 2, no. 5, Sept. 2004, pp. 19-24.