Wireless Network Security

Session 6

LEARNING OBJECTIVES

On completion of this session you should:

- Appreciate the IEEE 802.11 Wireless Protocol
- Appreciate the Wireless Applications Protocol
- Understand Wireless Transport Layer Security (WLTS)
- Appreciate end-to-end security between wireless clients and network servers

Contents

Reading

Prescribed readings

Reading 1: Read W. Stallings, Network Security Essentials - Application and Standard, 4th edition, Prentice Hall, 2011, pp. 191-196.

Reading 2: Read W. Stallings, Network Security Essentials - Application and Standard, 4th edition, Prentice Hall, 2011, pp. 197-210.

Reading 3: Read W. Stallings, Network Security Essentials - Application and Standard, 4th edition, Prentice Hall, 2011, pp. 211-218.

Reading 4: Read W. Stallings, Network Security Essentials - Application and Standard, 4th edition, Prentice Hall, 2011, pp. 218-227.

Reading 5: Read W. Stallings, Network Security Essentials - Application and Standard, 4th edition, Prentice Hall, 2011, pp. 228-230.

6.0 Introduction

IEEE 802.11 is a standard for wireless LAN. Wireless LAN (WLAN) allows users to move within an organization without losing network connectivity. This study guide looks at two important wireless network security schemes. First, we look at the IEEE 802.11i standard for wireless LAN security, also referred to as Wi-Fi. The remainder of the study guide is devoted to security standards for Web access from mobile wireless devices, such as cell phones. We start with Wireless Application Protocol (WAP), which is a set of standards for communication between mobile devices and Web servers. Then we examine the Wireless Transport layer Security (WTLS) protocol, which provides security between the mobile devices and

a gateway that operates between the cellular network and the Internet. Finally we shall cover end-to-end security services between WAP devices and Web servers.

IEEE 802 is a committee that has developed standards for a wide range of local area networks (LANs). In 1990, the IEEE 802 Committee formed a new working group, IEEE 802.11, with a charter to develop a protocol and transmission specifications for wireless LANs (WLANs). Since that time, the demand for WLANs, at different frequencies and data rates, has exploded. Keeping pace with this demand, the IEEE 802.11 working group has issued an ever-expanding list of standards.

6.1 IEEE 802.11 Wireless LAN Overview

Before proceeding, we need to briefly preview the IEEE 802 protocol architecture. IEEE 802.11 standards are defined within the structure of a layered set of protocols. This structure, used for all IEEE 802 standards, is illustrated in Stallings Figure 6.1.

The lowest layer of the IEEE 802 reference model is the **physical layer**, which includes such functions as encoding/decoding of signals and bit transmission/reception. In addition, the physical layer includes a specification of the transmission medium. In the case of IEEE 802.11, the physical layer also defines frequency bands and antenna characteristics.

Next is the **media access control (MAC) layer**, which controls access to the transmission medium to provide an orderly and efficient use of that capacity. The MAC layer receives data from a higher-layer protocol, typically the Logical Link Control (LLC) layer, in the form of a block of data known as the **MAC service data unit (MSDU).** The exact format of the MPDU differs somewhat for the various MAC protocols in use.

In most data link control protocols, the data link protocol entity is responsible not only for detecting errors using the CRC, but for recovering from those errors by retransmitting damaged frames. In the LAN protocol architecture, these two functions are split between the MAC and LLC layers. The MAC layer is responsible for detecting errors and discarding any frames that contain errors. The LLC layer optionally keeps track of which frames have been successfully received and retransmits unsuccessful frames.

6.1.1 IEEE 802.11 Network Components & Architecture

The smallest building block of a wireless LAN is a **basic service set (BSS)**, which consists of wireless stations executing the same MAC protocol and competing for access to the same shared wireless medium. A BSS may be isolated or it may connect to a backbone **distribution system (DS)** through an **access point (AP**). The AP functions as a bridge and a relay point. In a BSS, client stations do not communicate directly with one another. Rather the MAC frame is first sent from the originating station to the AP, and then from the AP to the destination station. Similarly, a MAC frame from a station in the BSS to a remote station is sent from the local station to the AP and then relayed by the AP over the DS on its way to the destination station. The BSS generally corresponds to what is referred to as a cell. The DS can be a switch, a wired network, or a wireless network. When all the stations in the BSS are mobile stations that communicate directly with one another, not using an AP, the BSS is called an **independent BSS (IBSS)**. An IBSS is typically an ad hoc network. In an IBSS, the stations all communicate directly, and no AP is involved.

A simple configuration is shown in Figure 17.3, in which each station belongs to a single BSS; that is, each station is within wireless range only of other stations within the same BSS. It is also possible for two BSSs to overlap geographically, so that a single station could participate in more than one BSS. Further, the association between a station and a BSS is dynamic. Stations may turn off, come within range, and go out of range. An **extended service set (ESS)** consists of two or more basic service sets interconnected by a distribution system. The extended service set appears as a single logical LAN to the logical link control (LLC) level.

6.1.2 IEEE 802.11 Services

1.The service provider can be either the station or the DS. Station services are implemented in every 802.11 station, including AP stations. Distribution services are provided between BSSs; these may be implemented in an AP or in another special-purpose device attached to the distribution system.

2.Three of the services are used to control IEEE 802.11 LAN access and confidentiality. Six of the services are used to support delivery of MSDUs between stations. If the MSDU is too large to be transmitted in a single MPDU, it may be fragmented and transmitted in a series of MPDUs.

We next discuss the services in an order designed to clarify the operation of an IEEE 802.11 ESS network. **MSDU delivery**, which is the basic service,

has already been mentioned. **Distribution** is the primary service used by stations to exchange MPDUs when the MPDUs must traverse the DS to get from a station in one BSS to a station in another BSS. **Integration** enables transfer of data between a station on an IEEE 802.11 LAN and a station on an integrated (wired) IEEE 802.x LAN. To deliver a message within a DS, the distribution service needs to know where the destination station is located. **Association** establishes an initial association between a station and an AP. **Reassociation** enables an established association to be transferred from one AP to another, allowing a mobile station to move from one BSS to another. **Disassociation** is a notification from either a station or an AP that an existing association is terminated.

Reading 1:

W. Stallings, Network Security Essentials - Application and Standard, 4th edition, Prentice Hall, 2011, pp. 191-196.

6.2 IEEE 802.11i Wireless LAN Security

The differences between wired and wireless LANs (in that wireless traffic can be monitored by any radio in range, and need not be physically connected) suggest the increased need for robust security services and mechanisms for wireless LANs. The original 802.11 specification included a set of security features for privacy and authentication that were quite weak. For privacy, 802.11 defined the **Wired Equivalent Privacy (WEP)** algorithm. The privacy portion of the 802.11 standard contained major weaknesses. Subsequent to the development of WEP, the 802.11i task group has developed a set of capabilities to address the WLAN security issues. In order to accelerate the introduction of strong security into WLANs, the Wi-Fi Alliance promulgated **Wi-Fi Protected Access (WPA)** as a Wi-Fi standard. WPA is a set of security mechanisms that eliminates most 802.11 security issues and was based on the current state of the 802.11i standard. The final form of the 802.11i standard is referred to as **Robust Security Network (RSN)**. The Wi-Fi Alliance certifies vendors in compliance with the full 802.11i specification under the WPA 2 program.

6.2.1 IEEE 802.11i Services

The 802.11i RSN security specification defines the following services:

• **Authentication**: A protocol is used to define an exchange between a user and an AS that provides mutual authentication and generates temporary keys to be used between the client and the AP over the wireless link.

• **Access control**: This function enforces the use of the authentication function, routes the messages properly, and facilitates key exchange. It can work with a variety of authentication protocols.

• **Privacy with message integrity**: MAC-level data (e.g., an LLC PDU) are encrypted, along with a message integrity code that ensures that the data have not been altered.

6.2.2 IEEE 802.11i Phases of Operation

The operation of an IEEE 802.11i RSN can be broken down into five distinct phases of operation. One new component is the authentication server (AS). The five phase are:

• **Discovery**: An AP uses messages called Beacons and Probe Responses to advertise its IEEE 802.11i security policy. The STA uses these to identify an AP for a WLAN with which it wishes to communicate. The STA associates with the AP, which it uses to select the cipher suite and authentication mechanism when the Beacons and Probe Responses present a choice.

• **Authentication**: During this phase, the STA and AS prove their identities to each other. The AP blocks non-authentication traffic between the STA and AS until the authentication transaction is successful. The AP does not participate in the authentication transaction other than forwarding traffic between the STA and AS. • **Key generation and distribution**: The AP and the STA perform several operations that cause cryptographic keys to be generated and placed on the AP and the STA. Frames are exchanged between the AP and STA only

• **Protected data transfer**: Frames are exchanged between the STA and the end station through the AP. As denoted by the shading and the encryption module icon, secure data transfer occurs between the STA and the AP only; security is not provided end-to-end.

• **Connection termination**: The AP and STA exchange frames. During this phase, the secure connection is torn down and the connection is restored to the original state.

We now look in more detail at the RSN phases of operation, beginning with the discovery phase. The purpose of this phase is for an STA and an AP to recognize each other, agree on a set of security capabilities, and establish an

association for future communication using those security capabilities (Confidentiality and MPDU integrity protocols for protecting unicast traffic, Authentication method, Cryptography key management approach). Confidentiality and integrity protocols for protecting multicast/broadcast traffic are dictated by the AP, since all STAs in a multicast group must use the same protocols and ciphers. The specification of a protocol, along with the chosen key length (if variable) is know as a *cipher suite*. The options for the confidentiality and integrity cipher suite are as follows: WEP, with either a 40-bit or 104-bit key (for backward compatibility), TKIP, CCMP, vendor-specific methods. The options for the authentication and key management (AKM) suite are: IEEE 802.1X, pre-shared key, vendor-specific methods). The discovery phase consists of three exchanges: Network and security capability discovery, Open system authentication, and Association.

The authentication phase enables mutual authentication between an STA and an authentication server (AS) located in the DS. Authentication is designed to allow only authorized stations to use the network and to provide the STA with assurance that it is communicating with a legitimate network.

6.2.3 IEEE 802.1X Access Control Approach

IEEE 802.11i uses the Extensible Authentication Protocol (EAP) that is defined in the IEEE 802.1X standard, designed to provide access control functions for LANs. Before a supplicant (wireless station) is authenticated by the AS (authentication server), using an authentication protocol, the authenticator (AP) only passes control or authentication messages between the supplicant and the AS; the 802.1X control channel is unblocked but the 802.11 data channel is blocked. Once a supplicant is authenticated and keys are provided, the authenticator can forward data from the supplicant, subject to predefined access control limitations for the supplicant to the network. Under these circumstances, the data channel is unblocked. 802.1X uses the concepts of controlled and uncontrolled ports. Ports are logical entities defined within the authenticator and refer to physical network connections. For a WLAN, the authenticator (the AP) may have only two physical ports, one connecting to the DS (backbone distribution system) and one for wireless communication within its BSS.

6.2.4 IEEE 802.11i Key Management Phase

Although the authentication is successful, the ports remain blocked until the temporal keys are installed in the STA and AP, which occurs during the 4-Way Handshake. During the key management phase, a variety of

cryptographic keys are generated and distributed to STAs. There are two types of keys: pairwise keys, used for communication between an STA and an AP; and group keys, for multicast communication. Stallings Figure 17.8 shows the two key hierarchies. Pairwise keys are used for communication between a pair of devices, typically between an STA and an AP. These keys form a hierarchy, beginning with a master key from which other keys are derived dynamically and used for a limited period of time. A **pre-shared key (PSK)** is a secret key shared by the AP and a STA, and installed in some fashion outside the scope of IEEE 802.11i. The other alternative is the **master session key (MSK),** also known as the AAAK, which is generated using the IEEE 802.1X protocol during the authentication phase, as described previously. The **pairwise master key (PMK**) is derived from the master key as follows: If a PSK is used, then the PSK is used as the PMK; if a MSK is used, then the PMK is derived from the MSK by truncation (if necessary). By the end of the authentication phase (on EAP Success message), both the AP and the STA have a copy of their shared PMK. The PMK is used to generate the **pairwise transient key (PTK)**, which in fact consists of three keys to be used for communication between an STA and AP after they have mutually authenticated. To derive the PTK, the PMK, the MAC addresses of the STA and AP, and nonces generated when needed are all input to the HMAC-SHA-1 function. Group keys are used for multicast communication when one STA sends MPDU's to multiple STAs.

6.2.5 IEEE 802.11i Protected Data Transfer Phase

IEEE 802.11i defines two schemes for protecting 802.11 MPDU data message integrity and confidentiality: the Temporal Key Integrity Protocol (TKIP), and the Counter Mode-CBC MAC Protocol (CCMP).

TKIP is designed to require only software changes to devices that are implemented with the older wireless LAN security approach called Wired Equivalent Privacy (WEP). TKIP adds a 64-bit message integrity code (MIC), generated by an algorithm, called Michael, to the 802.11 MAC frame after the data field. TKIP provides data confidentiality by encrypting the MPDU plus MIC value using RC4.

CCMP is intended for newer IEEE 802.11 devices that are equipped with the hardware to support this scheme. CCMP uses the cipher block chaining message authentication code (CBC-MAC) to provide message integrity. CCMP uses the CTR block cipher mode of operation, with AES for encryption. The same 128-bit AES key is used for both integrity and confidentiality. The scheme uses a 48-bit packet number to construct a nonce to prevent replay attacks.

6.2.5 IEEE 802.11i Protected Data Transfer Phase

At a number of places in the IEEE 802.11i scheme, a pseudorandom function (PRF) is used. For example, it is used to generate nonces, to expand pairwise keys, and to generate the GTK. The PRF is built on the use of HMAC-SHA-1 to generate a pseudorandom bit stream. Recall that HMAC-SHA-1 takes a message (block of data) and a key of length at least 160 bits and produces a 160-bit hash value. SHA-1 has the property that the change of a single bit of the input produces a new hash value with no apparent connection to the preceding hash value. This property is the basis for pseudorandom number generation. The IEEE 802.11i PRF takes four parameters (a secret key K, an application specific text string A, some data specific to each case B, and the desired number of pseudorandom bits Len) as input, and produces the desired number of random bits.

IEEE 802.11i PRF takes four parameters as inputs: *K, A, B, Len*. The parameter K serves as the key input to HMAC. The message input consists of four items concatenated together: the parameter A, a byte with value 0, the parameter B, and a counter i. The counter is initialized to 0. The HMAC algorithm is run once, producing a 160-bit hash value. If more bits are required, HMAC is run again with the same inputs, except that i is incremented each time, until the necessary number of bits is generated.

Reading 2:
W. Stallings, Network Security Essentials - Application and Standard, 4th edition, Prentice Hall, 2011, pp. 197-210.

6.3 Wireless Application Protocol (WAP)

The Wireless Application Protocol (WAP) is a universal, open standard developed by the WAP Forum to provide mobile users of wireless phones and other wireless devices, access to telephony and information services. WAP is designed to work with all wireless network technologies (e.g., GSM, CDMA, TDMA). WAP is based on existing Internet standards, such as IP, XML, HTML, and HTTP, as much as possible, & also includes security facilities. The current release of the WAP specification is version 2.0. Strongly affecting the use of mobile phones and terminals for data services are the significant limitations of the devices (in processors,

memory, and battery life) and the networks (relatively low bandwidth, high latency, and unpredictable availability and stability) that connect them. The user interface is also limited, displays are small, and all these features vary widely from terminal device to terminal device and from network to network. WAP is designed to deal with these challenges. The WAP specification includes:

- A programming model based on the WWW Programming Model
- A markup language, the Wireless Markup Language, adhering to XML
- A specification of a small browser suitable for a mobile, wireless terminal
- A lightweight communications protocol stack
- A framework for wireless telephony applications (WTAs)

The WAP Programming Model is based on three elements: the *client,* the *gateway,* and the *original* server. HTTP is used between the gateway and the original server to transfer content. The gateway acts as a proxy server for the wireless domain. Its processor(s) provide services that offload the limited capabilities of the hand-held, mobile, wireless terminals. For example, the gateway provides DNS services, converts between WAP protocol stack and the WWW stack (HTTP and TCP/IP), encodes information from the Web into a more compact form that minimizes wireless communication, and, in the other direction, decodes the compacted form into standard Web communication conventions. The gateway also caches frequently requested information.

Using WAP, a mobile user can browse Web content on an ordinary Web server. The Web server provides content in the form of HTML-coded pages that are transmitted using the standard Web protocol stack (HTTP/TCP/IP). The HTML content must go through an HTML filter, which may either be colocated with the WAP proxy or in a separate physical module. The filter translates the HTML content into WML content. If the filter is separate from the proxy, HTTP/TCP/IP is used to deliver the WML to the proxy. The proxy converts the WML to a more compact form known as binary WML and delivers it to the mobile user over a wireless network using the WAP protocol stack. If the Web server is capable of directly generating WML content, then the WML is delivered using HTTP/TCP/IP to the proxy, which converts the WML to binary WML and then delivers it to the mobile node using WAP protocols.

The **WAP architecture** is designed to cope with the two principal limitations of wireless Web access: the limitations of the mobile node (small

screen size, limited input capability) and the low data rates of wireless digital networks. Even with the introduction of 3G wireless networks, which provides broadband data rates, the small hand-held mobile nodes continue to have limited input and display capabilities. Thus WAP or a similar capability will be needed for the indefinite future.

6.3.1 IEEE 802.11i Wireless Markup Language

WML was designed to describe content and format for presenting data on devices with limited bandwidth, limited screen size, and limited user input capability. It is designed to work with telephone keypads, styluses, and other input devices common to mobile, wireless communication. WML permits the scaling of displays for use on two-line screens found in some small devices, as well as the larger screens found on smart phones. For an ordinary PC, a Web browser provides content in the form of Web pages coded with the **Hypertext Markup Language (HTML)**. To translate an HTML-coded Web page into WML with content and format suitable for wireless devices, much of the information, especially graphics and animation, must be stripped away. WML presents mainly text-based information that attempts to capture the essence of the Web page and that is organized for easy access for users of mobile devices. Important features of WML include: text and image formatting and layout commands, deck/card organizational metaphor, and support for navigation among cards and decks. In an HTML-based Web browser, a user navigates by clicking on links. At a WML-capable mobile device, a user interacts with cards, moving forward and back through the deck. A card specifies one or more units of interaction (a menu, a screen of text, or a text-entry field). A WML deck is similar to an HTML page in that it is identified by a Web address (URL) and is the unit of content transmission.

6.3.2 IEEE 802.11i WAP Architecture

The overall stack architecture implemented in a WAP client is a five-layer model. Each layer provides a set of functions and/or services to other services and applications through a set of well-defined interfaces. Each of the layers of the architecture is accessible by the layers above, as well as by other services and applications. Many of the services in the stack may be provided by more than one protocol. For example, either HTTP or WSP (Wireless Session Protocol) may provide the Hypermedia Transfer service. Common to all five layers are sets of services that are accessible by multiple layers. These common services fall into two categories: security services and service discovery. The WAP specification includes mechanisms to provide confidentiality, integrity, authentication, and nonrepudiation.

There is a collection of service discovery services that enable the WAP client and the Web server to determine capabilities and services.

The Wireless Application Environment **(**WAE) specifies an application framework for wireless devices such as mobile telephones, pagers, and PDAs. In essence, the WAE consists of tools and formats that are intended to ease the task of developing applications and devices supported by WAP.

6.3.3 IEEE 802.11i WAP Protocol Architecture

The WAP architecture dictates a collection of services at each level and provides interface specifications at the boundary between each pair of layers. Because several of the services in the WAP stack can be provided using different protocols based on the circumstances, there are more than one possible stack configurations. A common protocol stack configuration consists of a WAP client device that connects to a Web server via a WAP gateway. This configuration is common with devices that implement version 1 of the WAP specification but is also used in version 2 devices (WAP2) if the bearer network does not support TCP/IP. Refer to Figure 6.14 of Reading 3, which depicts a common protocol stack configuration.

6.3.4 IEEE 802.11i WAP Protocols

We now provide an overview of the WAP protocols.

The Wireless Session Protocol (WSP) provides applications with an interface for two session services. The connection-oriented session service operates above WTP, and the connectionless session service operates above the unreliable transport protocol WDP. In essence, WSP is based on HTTP with some additions and modifications to optimize its use over wireless channels. The principal limitations addressed are low data rate and susceptibility to loss of connection due to poor coverage or cell overloading. WSP is a transaction-oriented protocol based on the concept of a request and a reply.

The Wireless Transaction Protocol (WTP) manages transactions by conveying requests and responses between a user agent (such as a WAP browser) and an application server for such activities as browsing and e-commerce transactions. WTP provides a reliable transport service but dispenses with much of the overhead of TCP, resulting in a lightweight protocol that is suitable for implementation in "thin" clients using low-bandwidth wireless links. WTP is transaction oriented rather than connection oriented.

The Wireless Datagram Protocol (WDP) is used to adapt a higher-layer WAP protocol to the communication mechanism (called the bearer) used between the mobile node and the WAP gateway. Adaptation may include partitioning data into segments of appropriate size for the bearer and interfacing with the bearer network. WDP hides details of the various bearer networks from the other layers of WAP. In some instances, WAP is implemented on top of IP.

Reading 3:
W. Stallings, Network Security Essentials - Application and Standard, 4th edition, Prentice Hall, 2011, pp. 211-218.

6.4 Wireless Transport Layer Security

Wireless Transport Layer Security (WTLS) provides security services between the mobile device (client) and the WAP gateway. WTLS provides: data integrity, privacy, authentication, and denial-of-service protection. WTLS is based on the industry-standard Transport Layer Security (TLS) Protocol3, which is a refinement of the secure sockets layer (SSL) protocol. TLS is the standard security protocol used between Web browsers and Web servers. WTLS is more efficient that TLS, requiring fewer message exchanges. To provide end-to-end security, WTLS is used between the client and the gateway, and TLS is used between the gateway and the target server (see Figure 6.14 in Reading 3). WAP systems translate between WTLS and TLS within the WAP gateway. Thus, the gateway is a point of vulnerability and must be given a high level of security from external attacks.

Two important WTLS concepts are:

- **Secure connection**: A connection is a transport (in the OSI layering model definition) that provides a suitable type of service. For SSL, such connections are peer-to-peer relationships. The connections are transient. Every connection is associated with one session.

- **Secure session**: An SSL session is an association between a client and a server. Sessions are created by the Handshake Protocol. Sessions define a set of cryptographic security parameters, which can be shared among multiple connections. Sessions are used to avoid the expensive negotiation of new security parameters for each connection.

Between any pair of parties (applications like HTTP on client and server), there may be multiple secure connections. In theory, there may also be multiple simultaneous sessions between parties, but this feature is not used in practice.

There are a number of states associated with each session (see text for details). Once a session is established, there is a current operating state for both read and write (eg. receive and send). In addition, during the Handshake Protocol, pending read & write states are created. Upon successful conclusion of the Handshake Protocol, the pending states become the current states.

The connection state (see text) is the operating environment of the record protocol. It includes all parameters that are needed for the cryptographic operations (encryption/decryption and MAC calculation/verification).

6.4.1 IEEE 802.11I WTLS Protocol Architecture

WTLS is not a single protocol but rather two layers of protocols, as illustrated in Figure 6.15 of Reading 4. **The WTLS Record Protocol** provides basic security services to various higher-layer protocols. In particular, the Hypertext Transfer Protocol (HTTP), which provides the transfer service for Web client/server interaction, can operate on top of WTLS. Three higher-layer protocols are defined as part of WTLS: the **Handshake Protocol, The Change Cipher Spec Protocol, and the Alert Protocol**. These WTLS-specific protocols are used in the management of WTLS exchanges and are examined next.

The WTLS Record Protocol takes user data from the next higher layer (WTP, WTLS handshake protocol, WTLS alert protocol, WTLS change cipher spec protocol) and encapsulates these data in a PDU. The following steps occur (Figure 6.16):

1.The payload is compressed using a lossless compression algorithm.

2.A message authentication code (MAC) is computed over the compressed data, using HMAC. One of several hash algorithms can be used with HMAC, including MD-5 and SHA-1. The length of the hash code is 0, 5, or 10 bytes. The MAC is added after the compressed data.

3.The compressed message plus the MAC code are encrypted using a symmetric encryption algorithm. The allowable encryption algorithms are DES, triple DES, RC5, and IDEA.

4.The Record Protocol prepends a header to the encrypted payload.

The Change Cipher Spec Protocol is one of the three WTLS-specific protocols that use the WTLS Record Protocol, and it is the simplest. This protocol consists of a single message, which consists of a single byte with the value 1. The sole purpose of this message is to cause the pending state

to be copied into the current state, which updates the cipher suite to be used on this connection.

The Alert Protocol is used to convey WTLS-related alerts to the peer entity. Each message in this protocol consists of 2 bytes. The first byte takes the value warning(1), critical(2), or fatal(3) to convey the severity of the message. The second byte contains a code that indicates the specific alert. If the level is fatal, WTLS immediately terminates the connection, though other connections may continue, but no new connections may be established. A critical alert message results in termination of the current secure connection. Other connections using the secure session may continue and the secure identifier may also be used for establishing new secure connections. The connection is closed using the alert messages. Error handling in the WTLS is based on the alert messages.

The most complex part of WTLS is **the Handshake Protocol**. This protocol allows the server and client to authenticate each other and to negotiate a encryption and MAC algorithms and cryptographic keys to be used to protect data sent in a WTLS record. The Handshake Protocol is used before any application data are transmitted.

The Handshake Protocol consists of a series of messages exchanged by client and server. Stallings Figure 6.18 shows the initial exchange needed to establish a logical connection between client and server. The exchange can be viewed as having four phases. The **first phase** is used to initiate a logical connection and to establish the security capabilities that will be associated with it, and is initiated by the client. The **second phase** is used for server authentication and key exchange. The server begins this phase. The **third phase** is used for client authentication and key exchange. If all is satisfactory with the info from the server, the client sends one or more messages back to the server. The **fourth phase** completes the setting up of a secure connection. At this point the handshake is complete and the client and server may begin to exchange application layer data.

6.4.2 IEEE 802.11i WTLS Cryptographic Algorithms

Authentication in the WTLS is carried out with certificates. Authentication can occur between the client and the server or the client only authenticates the server. The latter procedure can happen only if the server allows it to occur. The server can require the client to authenticate itself to the server. However, the WTLS specification defines that authentication is an optional procedure. Currently, X.509v3, X9.68 and WTLS certificates are supported. The WTLS certificate is optimized for size.

The purpose of the WTLS protocol is for the client and server to generate a mutually shared pre-master key. This key is then used to generate as master key. A number of key exchange protocols are supported by WTLS. They can be grouped into those protocols that include a server_key_exchange message as part of the handshake protocol and those that don't. The server_key_exchange message is sent by the server only when the server certificate message (if sent) does not contain enough data to allow the client to exchange a pre-master secret, including for conventional Diffie-Hellman performed anonymously, elliptic curve Diffie-Hellman, or RSA key exchange without authentication. The server key exchange message is not sent for Elliptic curve Diffie-Hellman key exchange with ECDSA-based certificate, or for RSA key exchange with RSA based certificates.

The WTLS Pseudorandom Function (PRF) is used for a number of purposes. The PRF takes as input a secret value, a seed, and an identifying label, and produces an output of arbitrary length. WTLS PRF is implemented using only one hash algorithm (unlike TLS). Which hash algorithm is actually used, is agreed during the handshake as a part of the cipher spec. The PRF is based on a HMAC based data expansion function. See text for details.

Master Key Generation of the shared master secret, a one-time 20-byte value (160 bits) generated for this session by means of secure key exchange. First, a pre_master_secret is exchanged. Second, the master_secret is calculated by both parties, using the following function:

master_secret = PRF(pre_master_secret, "master secret",

ClientHello.random || ServerHello.random)

where the random numbers are exchanged during the first phase of the handshake protocol. The MAC and encryption keys are then derived from the master key, using the HMAC algorithm, and encompasses these fields:

HMAC_hash (MAC_secret, seq_number ||
WTLSCompressed.record_type

|| WTLSCompressed.length || WTLSCompressed.fragment)

Either MD5 or SHA-1 may be used for the HMAC hash function.

Encryption is applied to all of the WTLS record, except the header, using RC5, DES, 3DES, or IDEA encryption algorithms.

Reading 4:
W. Stallings, Network Security Essentials - Application and Standard, 4th edition, Prentice Hall, 2011, pp. 218-227.

6.5 WAP End-to-End Security

The basic WAP transmission model, involving a WAP client, a WAP gateway, and a Web server, results in a security gap, as illustrated in Stallings Figure 17.19. The mobile device establishes a secure WTLS session with the WAP gateway. The WAP gateway, in turn, establishes a secure SSL or TLS session with the Web server. Within the gateway, data are not encrypted during the translation process. The gateway is thus a point at which the data may be compromised.

There are a number of approaches to providing end-to-end security between the mobile client and the Web server. In the WAP version 2 (known as WAP2) architecture document, the WAP forum defines several protocol arrangements that allow for end-to-end security. Version 1 of WAP assumed a simplified set of protocols over the wireless network and assumed that the wireless network did not support IP. WAP2 provides the option for the mobile device to implement full TCP/IP-based protocols and operate over an IP-capable wireless network.

Figure 6.20 of Reading 5 shows two ways in which this IP capability can be exploited to provide end-to-end security. In both approaches, the mobile client implements TCP/IP and HTTP.

The first approach (Figure 6.20a) is to make use of TLS between client and server. A secure TLS session is set up between the endpoints. The WAP gateway acts as a TCP-level gateway and splices together two TCP connections to carry the traffic between the endpoints. However, the TCP user data field (TLS records) remains encrypted as it passes through the gateway and so end-to-end security is maintained.

Another possible approach is shown in Figure 6.20b. Here we assume that the WAP gateway acts as a simple Internet router. In this case, end-to-end security can be provided at the IP level, using IPsec (see SG 8).

Reading 5:
W. Stallings, Network Security Essentials - Application and Standard, 4th edition, Prentice Hall, 2011, pp. 228-230.

8.10 References

[1] Chen J, Jiang M and Liu Y, "*Wireless LAN Security and IEEE 802.11i*", IEEE Wireless Communications, February 2005.

[2] Frankel S, Eydt B, Owens L and Scarfone K, "*Establishing Wireless Robust Security Networks: A Guide to IEEE 802.11i*" NIST Special Publication SP 800-897, February 2007.

[3] Stallings W, "*Data and Computer Communications*" Eighth Edition, Prentice hall, 2007.