

12 Wireless Security¹

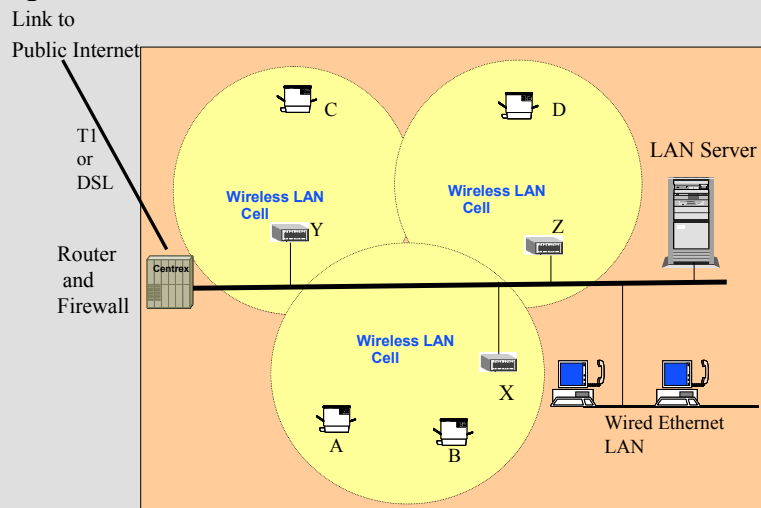
12.1	INTRODUCTION	12-3
12.2	SECURITY PRINCIPLES	12-5
12.2.1	Overview.....	12-5
12.2.2	Security Tradeoffs and a Design Procedure.....	12-6
12.2.3	Overview of Core Security Technologies.....	12-8
12.2.4	Information Protection (Privacy and Integrity).....	12-10
12.2.5	Authentication and PKI.....	12-13
12.2.6	Authorization and Access Control.....	12-14
12.2.7	Accountability and Assurance	12-14
12.2.8	Mapping Security Technologies to Security Needs.....	12-15
12.3	WIRELESS LAN SECURITY.....	12-16
12.4	CELLULAR WIRELESS NETWORK SECURITY	12-20
12.5	SATELLITE COMMUNICATION SECURITY.....	12-22
12.6	WLL AND CORDLESS SECURITY.....	12-24
12.6.1	Wireless Local Loop (WLL) Security.....	12-24
12.6.2	Cordless Phone Security.....	12-26
12.7	EMERGING WIRELESS NETWORK SECURITY	12-27
12.7.1	Free Space Optics (FSO) Security.....	12-27
12.7.2	Mobile Ad Hoc Network Security.....	12-28
12.7.3	Wireless Sensor Network Security.....	12-30
12.8	INTERNET LAYER SECURITY PROTOCOLS – VPNs AND IPSec	12-31
12.8.1	Virtual Private Networks (VPNs)	12-31
12.8.2	IPSec	12-33
12.9	WIRELESS MIDDLEWARE SECURITY	12-35
12.9.1	Overview.....	12-35
12.9.2	Secure Socket Layer (SSL) for Wireless Web Security.....	12-35
12.9.3	WAP Security and WTLS.....	12-38
12.9.4	i-mode Security.....	12-41
12.9.5	Wireless VPN Versus WAP Security	12-41
12.10	WIRELESS APPLICATION SECURITY.....	12-42
12.10.1	Overview.....	12-42
12.10.2	Mobile Client Security	12-43
12.10.3	Web Server Tier (Middle Tier) Security.....	12-44
12.10.4	Back-end System and Transaction Security Through SET	12-46
12.11	SHORT EXAMPLES AND CASE STUDIES	12-47
12.11.1	Wireless in Government Services.....	12-47
12.11.2	Wireless Security in the Health Sector.....	12-48
12.11.3	Wireless LANs at Texas A&M University.....	12-49
12.12	ANALYSIS AND DESIGN OF WIRELESS SECURITY	12-50

¹ This chapter is based on the wireless security material in the book: A. Umar, *Information Security and Auditing in the Digital Age*, 2nd ed., NGE Solutions, June 2004.

12.12.1	<i>Analysis of Security Tradeoffs and Design Guidelines</i>	12-50
12.12.2	<i>Simple Design Procedure for Wireless Security</i>	12-52
12.13	SUMMARY AND CONCLUSIONS.....	12-56
12.14	REVIEW QUESTIONS AND EXERCISES.....	12-56
12.15	REFERENCES.....	12-56

Example: Securing a Small Office Network

The following figure shows a network in a small office that needs to be secured. The network consists of three 802.11 cells, handled by access points X, Y, and Z. The access points (APs) are connected to a Fast Ethernet LAN that serves as a backbone and controls access to a LAN server. The backbone is also connected to a few wired Ethernet LANs, and to the public Internet through a router.



The company wants to secure this network, especially mobile users, to eliminate unauthorized access. The following approaches are worth reviewing:

1. No physical network security at AP level, but ID/PW are required before any mobile user can access the LAN server. This is very convenient because a mobile user can roam between different cells without having to log on to various APs. This approach, however, is weak in terms of security because anyone who wanders into a cell can access the backbone and launch attacks on the internal LAN from wireless users. Since no encryption is used, a wireless user can “listen” to the traffic in a cell and on the backbone.
2. Same as option 1, but introduce encryption between mobile users and LAN server when a user signs on by using ID/PW. This makes the network a bit more secure because the network traffic on the backbone is encrypted. However, a mobile user can still access the wired Ethernet LAN and also access the Internet without additional checking.
3. Physical network security at the AP level, i.e., each AP has its own ID/PW and encryption code. In addition, all other LAN segments, including the wired Ethernet LAN, also use their own ID/PW and encryption. This presents strong security because the traffic in each cell as well as the LAN segment is authenticated and encrypted. However, this can be an operational nightmare for users because they have to log on/log off every time they move from one cell to another.
4. A totally different, but commonly used, approach is to treat all internal WLANs as external

networks that have to go through the company firewall. The main advantage of this approach is that even if an intruder wanders into a WLAN cell, the intruder cannot access any internal corporate resources without going through the firewall. To support this approach, the access points X, Y, and Z would be connected to the router and firewall (this configuration is not shown above).

12.1 Introduction

The growth of wireless networks and mobile services over the last few years has been tremendous. Naturally, the security concerns are becoming more serious with the growth of wireless. As more people access critical information, and as consumers begin to do their business and banking on devices that are connected over wireless LANs, MANs, and WANs, wireless security has moved to the forefront. In fact, user surveys indicate that security is the largest possible deterrent to the use of wireless in corporations. For example, in a December 2000 survey of 101 IT and business managers, *Internet Week* found that security was ranked as the number-one concern regarding wireless use.

In essence, wireless networks face the same type of security issues (e.g., privacy, integrity, authentication) as the wired networks. Wireless security is not much different from wired security. The same security concerns exist, wired or not: authenticate whom you are talking to, secure the data as it travels from the handheld device to the destination host, and ensure that the traffic has not been altered en-route. Companies such as Amazon.com and E-Trade do this in the wired world. The main differentiating issue of wireless network security is that the information is transmitted over a common medium (the air), so is easier to tap into and alter while on transit between end-points.

There are a number of stories about eavesdropping of wireless traffic. For example, competitors have been able to capture the emails between HP personnel by simply sitting in the office parking lot with an antenna. Something similar also happened to Sun Microsystems. In addition, information sent by a federal agency wirelessly was intercepted and then used against the agency in a future negotiation. My own students, from a wireless network class that I taught, spent a day in Manhattan and captured a disk full of plain text (unencrypted data) by simply driving around the Manhattan business district in a car with a simple antenna. They were just doing research to demonstrate how vulnerable wireless communications are (well, that is what they told me!).

The issues of wireless security are not evenly divided. Some areas are more vulnerable than others. For example, Free Space Optics (FSO) systems transmit information by using laser beams – a very difficult technology to intercept – while wireless Ethernet LANs are more susceptible to intervention because of several security weaknesses [Arbaugh 2001]. In addition, there are issues at different levels (networks, middleware, and applications). The objective of this chapter is to provide enough details so that a sound security solution based on a comprehensive checklist can be developed. The checklist must include the enterprise applications and corporate databases, computing platforms (e.g., computers, operating systems), middleware (e.g., web servers), and network elements (e.g., routers, wireless access points). After reviewing security principles, the principles are applied to all layers of a wireless systems – starting from network layers and proceeding to the application layer. The following questions guide the discussion:

- What are the core security principles and how they can be developed into a design approach?
- What are the security issues specific to wireless LANs, cellular networks, satellites, WLLs, and cordless systems?
- How can TCP/IP security through VPNs and IPSec be used to secure wireless communications?
- How do higher-level security solutions such as WAP security and SET (Secure Electronic Transactions) interact with wireless network security?
- Can a comprehensive wireless security procedure be developed that considers security at all layers?

Chapter Highlights

- Wireless security can be discussed at several levels (wireless physical network, TCP/IP, middleware, and applications).
- Numerous technologies exist to deal with the issues at various levels. Some techniques are better than others.
- Wireless networks impose special problems because the information is carried through the air and thus is easier to tap and alter.
- There are many areas of vulnerability in wireless networks:
 - Location services (HLR/VLR) introduce privacy concerns.
 - Wireless access points present numerous security exposures.
 - Mobile ad hoc networks are difficult to secure.
 - Several products use the un-authenticated Diffie-Hellman (DH) algorithm which suffers from a well-known *man-in-the-middle* attack.
 - The Wired Equivalent Privacy (WEP) algorithm, part of the IEEE802.11 standard for wireless LANs, is designed to protect wireless communication from eavesdropping.
 - Several weaknesses of WEP have been demonstrated.
- Approaches to deal with wireless network security include:
 - Turn on security at wireless links to avoid eavesdropping even if it has weaknesses.
 - Use WEP because it does provide some security – make up for WEP security by providing higher layers of security (e.g., SSL).
 - Make sure that all access points are themselves monitored and controlled so that no one sets up rogue access points.
 - Treat wireless networks as untrusted networks. Thus put the internal WLANs outside the firewall so that they are treated as outsiders.
 - Minimize placing critical applications and databases on wireless networks – move them to wired networks behind firewalls.
 - Make sure that the passwords on wireless networks are different than those used on the wired networks. Hackers usually capture passwords from wireless networks and then use them to gain access over wired networks.
- Some emerging wireless networks (FSO, UWB) offer very strong security features – use them if needed.
- Mobile ad hoc networks (MANET) and wireless sensor networks (WSNs) raise many security issues that are not resolved – stay away from them if you need highly secure networks, or use strong application-level security.
- Mobile IP allows you to maintain the same Internet connection address (“care of” address) as you move around. Many security issues and evolving solutions exist in Mobile IP.
- The WAP specification ensures that a secure protocol is available for transactions on a

wireless handset. WAP uses the Wireless Transport Layer Security (WTLS) protocol .



The Agenda

- Security Principles
- Wireless Network Security
- Higher Layer Security and Security Design

12.2 Security Principles

12.2.1 Overview

The issues of security are of vital importance for mobile services and need more attention. Basically, security involves the following aspects, called PIAAA:

- **Privacy:** assure confidentiality of information (i.e., no one other than the authorized people can see the information) when transmitting it over a network or storing it in an insecure place.
- **Integrity:** avoid corruption of information (i.e., no unauthorized modification allowed).
- **Authentication:** identify for certain who is communicating with you (i.e., make sure that if someone logs on as John, he in fact is John)
- **Authorization (Access control):** determine what access rights that person has (i.e., can John only read given information or can he also update, delete, and add information?).
- **Accountability:** assure that you can tell who did what when, and convince yourself that the system keeps its security promises. This includes *non-repudiation (NR)* – the ability to provide proof of the origin or delivery of data. NR protects the sender against a false denial by the recipient that the data has been received. It also protects the recipient against false denial by the sender that the data has been sent. In other words, a receiver cannot say that he/she never received the data, and the sender cannot say that he/she never sent any data.
- **Availability:** assure that the users can use the system when they need to. Attacks such as denial of service attempt to minimize the system availability.

It is also important to diligently administer the security system, i.e., to define and enforce the security policies that are consistent across all elements of applications, middleware services, and networks. These, and other aspects of security, are supported at various layers (network, middleware, application) by using a wide range of technologies (see Figure 12-1). Security is needed at these different layers since security at each layer fulfills different requirements. Figure 12-1 can serve to build a comprehensive checklist for security design. Let us briefly review the security at various layers (details will be given later).

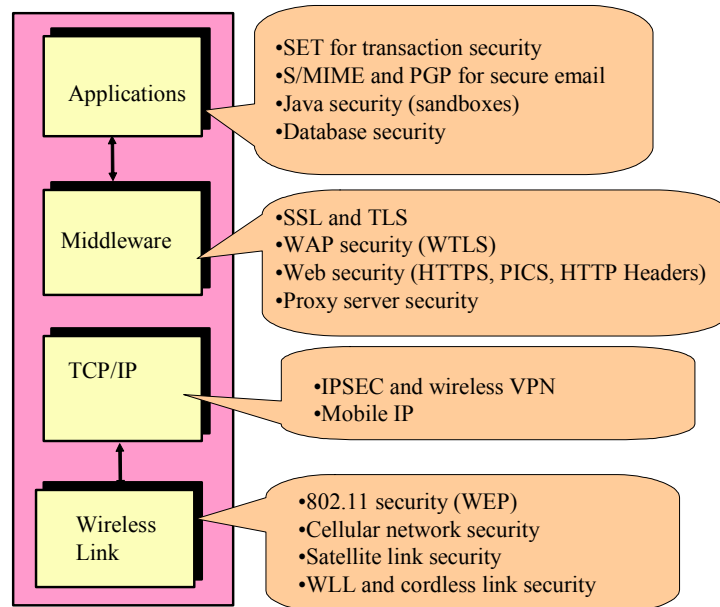


Figure 12-1: Levels of Security

Wireless link security protects information transfer at layer 1 and 2, i.e., over radio links used by wireless LANs, cellular networks, satellites, and wireless local loops. Sections 12.3 through 12.7 discuss these security issues in some detail. Independent of the radio link, the network traffic can be encrypted at higher layers (TCP/IP) by using IPsec and VPNs (Virtual Private Networks). At the middleware layers, SSL (Secure Socket Layer) is used for secure Web browser-Web server exchanges, and WTLS (Wireless Transport Layer Security) is used to secure WAP applications. A variety of security approaches exist at the application layers, in which case authorization controls are used within applications to regulate access to specific data, and cryptographic infrastructures are built to strongly authenticate users and provide confidentiality. Application level security is provided, for example, by database managers, Java Virtual Machines, email security packages (e.g., S-MIME), and SET (Secure Electronic Transactions). In particular, applications themselves provide access control and strong user authentication. These issues are discussed in Sections 12.8 through 12.10.

Security must be considered at all layers. Securing a higher layer while keeping lower layers unsecured makes the system vulnerable to intrusions from the lower layers. In general, lack of security at a certain layer might compromise the overall system even if other layers are secured. Consider, for instance, a system where the application data is secure – it can only be accessed through proper ID-PW – but is transmitted over an insecure network. In this case, the overall security of the application could be suspect because an intruder could tap the network and read the data while in transit. Specifically, application security protects application data (e.g., database security mechanisms allow the data to be stored on the hosts in a protected manner), while network security protects data as it is transmitted on the network. Both are needed for secure operations.

12.2.2 Security Tradeoffs and a Design Procedure

There are different approaches to achieving wireless security. Figure 12-2 shows the main approaches (darker areas indicate that security, such as encryption, has been applied). If you use the network layer 1 and 2 security (let us say, encryption by using the 802.11 WEP) as shown in Figure 12-2a, then *all traffic on that network segment* is encrypted (emails, Web

browsing, etc.). But once your messages go beyond that network segment (say, an office LAN), then you cannot assume that other network segments on the path are secure. All “hops” between the two end points have to be encrypted for end-to-end security. In addition, WEP security is weak, as we will discuss later; thus another layer of security is needed.

Another option is to move security to a higher level, i.e., to the IPSec level as shown in Figure 12-2b. Since most applications run on top of IP, all communications between these applications are encrypted, IPSec provides a good end-to-end security solution over IP and is used in most VPNs. However, IPSec requires a re-engineering at the router level to assure that the routers can understand IPSec packets and handle IPSec encryption/decryption.

Yet another option is to move security to still a higher layer, i.e., above the TCP layer as shown in Figure 12-2c. The most widely used option is the use of SSL (Secure Socket Layer). SSL could be made transparent to the applications by using it as part of the TCP/IP stack, or it can be packaged in specific applications such as email, file transfer and Web browsing. This presents several choices:

- SSL is currently available as an option with most HTTP packages. For example, almost all Web browsers and servers at present support SSL. In Figure 12-2c, application A1 is not encrypted and uses unsecure HTTP (without SSL support), thus it is not secure at all. A2 uses its own encryption (e.g., SET – Secure Electronic Transactions), although it also uses plain HTTP. A3 does not use its own encryption but it uses SSL with HTTP; thus it uses SSL encryption. A4 uses its own encryption plus the SSL security; thus it is theoretically more secure.
- SMTP (Simplified Mail Transport Protocol) by itself is not encrypted, but PGP and S/MIME are used for encryption.

It can be seen from this discussion that to provide complete wireless security, more than one layer should be secured. For example, in addition to WEP security for 802.11, IPSec as well as SSL can be used for added security. Although it is possible to encrypt at every layer, too much encryption can add significant overhead.

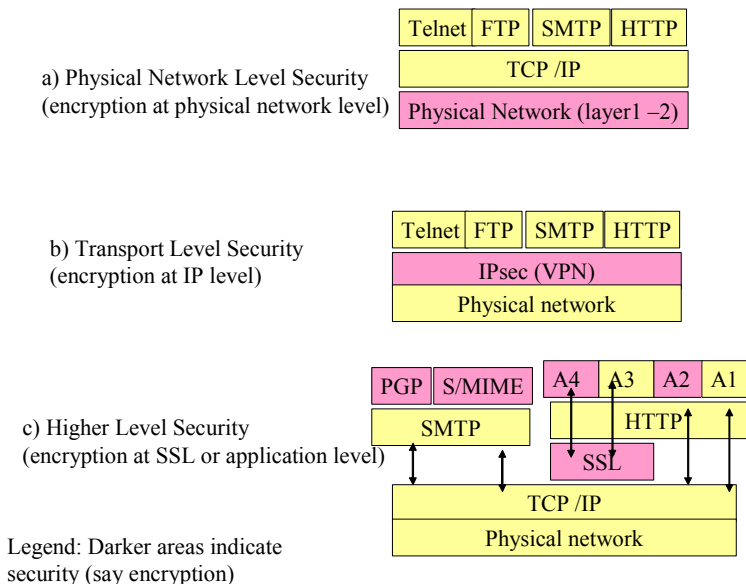


Figure 12-2: Approaches to Wireless Security

A **security design approach** is needed to include various security tradeoffs at different layers and address various issues. For example, is it better to secure a lower layer than a higher

layer? In addition, is it important that the business logic of a Web application runs on a server and not on the client? Can the Web application server, being used to integrate access to back-end resources (databases, etc.), provide greater security of the resources? Finally, how should the application be structured to take advantage of the network filters (“firewalls”)?

A good design protects the Web server (providing presentation services to the customers) behind an outer firewall, and the remaining servers and databases (supporting business logic) behind a second, inner firewall. This structure, shown in Figure 12-3, is known as a demilitarized zone, or DMZ. In most cases, a Web server sits alone in the DMZ, handling requests from the Web and passing them along to the secure internal network. The applications and internal business systems behind the inner firewall contain all the remaining business logic and data of the enterprise. In addition, you can gain performance benefits by caching frequently requested data inside the DMZ rather than retrieving it from back-end systems each time it is requested. For example, a catalog of frequently asked questions with answers could be stored in the DMZ. Naturally, resources in the DMZ are at higher risk, thus sensitive information should not be stored in the DMZ. We also need to consider security of clients. For example, mobile devices typically need another level of security before they can enter the DMZ. See Section 12.12 for a security design procedure.

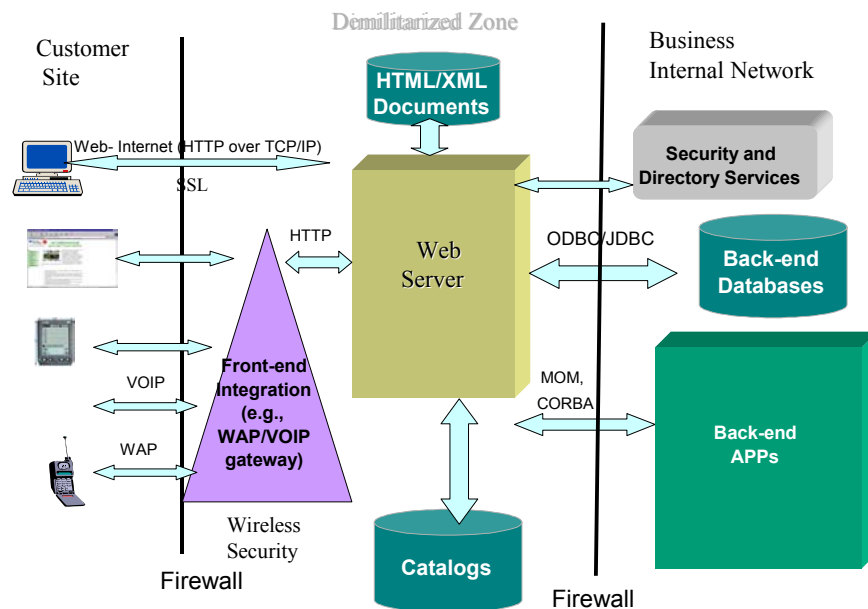


Figure 12-3: Security Design

12.2.3 Overview of Core Security Technologies

Wireless systems use the core secure technologies in a variety of ways to achieve as good security as possible. We have mentioned many of these technologies previously. Here is a quick overview.

User logon and password is one of the oldest and still most commonly used technologies. This technology enables the use of existing systems with minimal disruption to existing infrastructure and applications.

Encryption is another technology that has been used for a number of years to mask the messages so that the interveners cannot see/modify the messages. Due to e-commerce,

encryption/decryption has become a major area of active work. In the simplest case, data is transformed by a key into an encrypted message. The encrypted message is then transmitted and decrypted on the other side by using the same key. Encryption/decryption can be performed by hardware and/or software. Modern computing systems have the ability to implement very sophisticated encryption/decryption techniques. The same encryption can be used on all data in a system, or encryption keys can be more “personalized.” For example, instead of using the same encryption/decryption key on all data from all stations in a network, each station or user can use its own encryption/decryption key. A user can have his or her own encryption card which is inserted into a workstation before the user logs on. This card encrypts the data before sending it across the network. The encrypted data can be read only by those users or programs with access to the same encryption key. Encryption is generally discussed in two different formats:

- **Secret Key:** In a secret-key (also known as symmetric or private) encryption scheme, the same key is used by the sender to encrypt the message and by the receiver to decrypt it. While secret-key encryption is usually very fast and efficient, the problem is with key management. In other words, since the sender and receiver have to agree on the same key, sending the key from one side to the other might compromise it.
- **Public key:** In a public-key (asymmetric) system, the encryption key E and the decryption key D are different – hence the name “asymmetric.” Each user has a pair of keys, a private key D that he keeps secret and a public key E that he publishes. When sender Bob needs to send a message to a user Joe, he encrypts the message with Joe’s public key E(J) . This encrypted message can only be decrypted with Joe’s private key D(J). Therefore if this encrypted message is delivered to user Pat who does not have key D(J), then Pat cannot decrypt it. Thus when Joe receives the message, he can decrypt the message by using D(J) and read the message. Notice that in this key system the decryption key is private and not transmitted over the network. While public-key systems solve the problem of key management, they are usually significantly slower than private-key systems. The RSA (Rivest, Shamir and Adleman) algorithm, developed in 1976, is by far the most widely used public-key encryption algorithm.

Digital signature is used to authenticate the source of a message. It is essentially the same as a public-key system except that the order in which the keys are applied is reversed. A sender “signs” the message by applying his private key to it. The sender sends the message and the signature to the receiver. The receiver checks the signature by applying the sender’s public key to it. If the receiver gets the original message back, he is sure that the message was signed by the sender’s private key, and therefore, was sent by the receiver himself. In essence, a digital signature is a block of data created by applying a cryptographic signing algorithm to some data using the signer’s private key. Digital signatures may be used to authenticate the source of the message and to assure message recipients that no one has tampered with a message since the time it was sent by the signer.

Message Digesting is used to make sure that a certain message was not changed along the way between the sender and the receiver. A message digest algorithm produces a fingerprint of the message by applying a hashing function to it. The receiver can check for the integrity of the message by reapplying the hash function and comparing with the original fingerprint. The hash functions used in these schemes are such that the fingerprint changes dramatically if a single bit of the message changes.

A **digital certificate** binds an entity’s identification to its public key and is issued by the certification authority. Digital certificates, based on the X.509v3 standard, enable Internet applications and other users to verify the identity of an entity. Unfortunately, certificates produced by one vendor’s product may not interoperate with other vendors’ certificates

because X.509 does not define the formats of the certificate entries and other necessary provisions. PKIX, the X.509 standard by IETF, defines the contents of public-key certificates and is intended to resolve these interoperability issues.

12.2.4 Information Protection (Privacy and Integrity)

Information must be protected and its integrity maintained at least at two levels: a) the sites where it exists, and b) when it is transmitted. In addition, the encryption keys themselves need to be protected.

Site Protection. Information must be protected at the sites where it exists. Access control (allowing authorized users to access needed data) protects data at various sites. Most database managers have security features that allow only authorized users to access needed data. In some cases, data is encrypted and stored for additional security. An important aspect of site protection at present is **Java Security**. Security of Java code has been an area of concern for a while. Current Java security is defined at the following levels:

- **Java 1.1's security management system** – All local code is trusted. All remote code is untrusted, unless it is digitally signed by a trusted source. Untrusted code runs in a “sandbox,” and has limited access to local system resources.
- **Java 2's security management system** – Local and remote code is checked by the same security management system. It supports fine-grained, flexible and easy-to-specify security and permission policies.

b) Transmission Protection. When data must travel outside of a secure system environment, it needs to be protected so that the policies governing its use cannot be violated. Secure communications – ensuring data privacy, data integrity, and origin authentication – are an important aspect of information protection. Examples of the technologies used for secure communications are:

- **Firewalls** – the network filters that police “who” enters and leaves an enterprise network and “what” gets in and out. A firewall is essentially a software package that is installed on network routers. This software checks each IP packet and determines if it should enter the system. Firewalls provide a logical and physical separation of the public Internet and internal IT systems. A good security design generally has two firewalls: an outer firewall that exposes some services to the outside world, and a second, inner firewall, that keeps the inner resources. The zone between the two firewalls is known as a demilitarized zone, or DMZ.
- **SSL** – The Secure Sockets Layer (SSL) protocol uses encryption and authentication techniques to ensure that communications between a client and a server remain private and to allow the client to identify the server and vice versa. SSL runs on top of TCP/IP and manages secure messaging on the network. SSL client and server negotiate encryption scheme and key size. SSL is currently used heavily to protect the traffic between Web clients and servers. It uses RSA (Rivest, Shamir, and Adleman) Public encryption for key session negotiation and DSA (Digital Signature Algorithm) for session encryption. See the sidebar “SSL” for additional information.
- **VPN and IPSec** – Virtual Private Networks (VPN) are private networks (e.g., networks internal to corporations) that use public communication infrastructure. In other words, you set up a private network over a public network by using encryption. VPNs use IETF IPSec (RFC 2401) and related standards to transport encrypted messages over shared networks. IPSec provides security at the packet level, instead of security at the application layer. It encrypts and signs Headers and/or Data parts of IP Header. It provides security without requiring changes to applications and thus is suitable for Virtual Private Networks (VPN). VPN differs from SSL in that it creates a secure channel between two

TCP/IP hosts over which multiple TCP/IP connections can be established. Each TCP/IP session itself may or may not use SSL.

- **S/MIME** – Most email client and server programs using Internet systems such as SMTP send email as clear text. The Secure Multipurpose Internet Mail Extensions (S/MIME), a specification for secure electronic messaging, can be used to prevent the interception and forgery of email.
- **Middleware Security** – CORBA security is a good example of protection at the middleware level. OMG specifies three levels of security for applications that use CORBA.
 - CORBA Security level 0 provides authentication and session encryption using SSL.
 - CORBA Security level 1 provides security to applications that are not aware of security. It automatically introduces authentication, session encryption, access control, and simple auditing. CORBA security level 1 uses CORBA interceptors.
 - CORBA Security Level 2 provides API to interface to CORBA security objects. Security-aware applications can specify security requirements (e.g., access control) at the object and method level.
- **SET** – The Secure Electronic Transaction (SET) protocol, developed jointly by Visa, MasterCard, IBM, and other technology providers, is used to protect the transfer of bankcard payment information over open networks like the Internet. This is an application-layer security protocol.

c) Key Protection. In addition to secure communications, protection of the keys that in turn are used to protect the assets is also important. Private keys and shared secrets, once acquired, must be protected. End-to-end security must include consideration of the security of the end user device. Private keys stored on a personal computer disk file may be stolen via access to the file system or outright theft of the device. Security can be enhanced by the use of smart cards. Another approach is to use a security chip embedded in end-user systems. In addition, server-side hardware devices can provide tamper-resistant key storage as well as assistance for encrypting and decrypting messages and public/private key operations, etc. that require heavy computational load.

Basic Security Services: SSL and Digital Certificates

At present, most Web browsers and servers use Secure Sockets Layer (SSL) technology to provide a safe way to transmit sensitive information, such as credit card numbers, online banking, email messages, surveys and other personal information.

The SSL protocol provides data encryption, server authentication, message integrity, and optional client authentication for a TCP/IP connection, allowing for the secure transfer of sensitive information over the Internet. SSL consists of software installed in browsers and on servers and can be obtained by subscribing to a Secured Service Provider such as ssl.com or by obtaining a Server Certificate from ssl.com and installing it on an existing secured server.

All major browsers and servers today are “SSL capable.” SSL technology was developed by Netscape Communications Corporation and has become the industry-standard method for protecting Web communications.

SSL Overview.

SSL uses public key encryption to provide security at the packet level. At the receiving end, SSL provides Server Authentication Message Integrity checks.

The basic public-key scheme assumes that both parties have a private/public key pair, and that they can trade these pairs. Then, they use their private keys to encrypt messages to be sent, and the public ones to decrypt received messages. Additionally, message integrity can be verified when checksum is generated for each packet, signed with the private key, and sent along with the packet.

However, this kind of encryption is too time-intensive. A faster encryption scheme consists of encrypting the main part of the packet, and then sending the key used to encrypt the packet. This key is itself encrypted using the private key and so is well-protected. This way, the heavy-duty encryption is used only for the “session key,” which is very short, making the whole transaction close in speed to a non-encrypted one.

SSL adopts this latter scheme. A “master key” is generated using some random data. The master key is used to generate a session key for each session, from which a client write key and a server write key are generated (you read with the other party’s write key). The server’s public key is used to encrypt the master key during the initial handshake. From then on, the packets are encrypted with the server or client write key, depending on who is sending it; a digest is taken; and the whole thing is finally packaged up with a session number. A faster version is obtained by using DES, a faster form of encryption than RSA.

SSL comes in two strengths, 40-bit and 128-bit, which refer to the length of the “session key” generated by every encrypted transaction. The longer the key, the more difficult it is to break the encryption code. Most browsers support 40-bit SSL sessions, and the latest browsers, including Netscape Communicator 4.0, enable users to encrypt transactions in 128-bit sessions – trillions of times stronger than 40-bit sessions. Global companies that require international transactions over the Web can use a global server certificates program to offer strong encryption to their customers.

SSL use in practice.

In order to use SSL between an individual using a Web browser and a business hosting a website on a secured server, the following must occur:

A Root Certificate must be installed on the individual’s local Web browser. Certificate Authorities (CA) such as ssl.com provide Root Certificates for public downloading by individuals.

A business must have a Server Certificate installed on their secured server. There are two ways a business can obtain access to a secured server: they can subscribe to a Secured Service Provider (SSP) such as ssl.com, or they can obtain a Server Certificate from a CA. In the latter case, the business must first ask their Internet Service Provider (ISP) to generate a Certificate Signing Request (CSR); then the CSR must be submitted to the SSP. The SSP will verify that the business is legitimate and will issue the business a Server Certificate. This certificate is then installed on the secured server, and SSL transactions are then enabled.

Digital certificates encrypt data using Secure Sockets Layer (SSL) technology. Because SSL is built into all major browsers and Web servers, simply installing a digital certificate turns on their SSL capabilities.

For further reading: the SSL site (www.ssl.com) has comprehensive documentation on the mechanics of SSL and certificates.

12.2.5 Authentication and PKI

In e-commerce/e-business, you need to authenticate the consumers who buy your products or services, employees who access internal systems from remote locations via the public Internet, or business partners who are tightly integrated into your supply chain and ERP systems.

For authentication, many applications choose to make use of one-time passwords. However, the use of such one-time passwords often requires the deployment of token cards – an expensive and labor-intensive effort. This is why software-based solutions are more popular. Most systems enforce authentication by developing a *session key* that establishes the identity of partners at the start of a session and is used for its duration. But then this session key needs to be encrypted. Should a private or public key system be used? Given the advantages and disadvantages of these approaches (private key is efficient but not very secure and public keys are not efficient but secure), in practice, a public key system is used to exchange the session key between the two sides. Then this key is used in a private key system only for that session. Many current systems, such as SSL (Secure Socket Layer) uses this technique. See the sidebar “SSL” for more details.

Many applications use cryptographic software to incorporate public-key cryptography for encryption and authentication. A number of such software packages exist, including the following:

- Kerberos (<http://www.mit.edu/kerberos/>), a cryptographic authentication scheme using a third-party authentication server to grant cryptographic “tokens” that authenticate users to a given service. Kerberos is an open standard designed to provide strong authentication by using secret-key cryptography. Used primarily for secure interoperation of existing systems, Kerberos is used for user authentication.
- Entrust (www.entrust.com), a subsidiary of Entrust Technologies that offers a portfolio of service solutions to securely manage e-business transactions. Solutions include secure e-business transactions from e-commerce websites to interactive cell phones. Entrust also recently entered the secure transaction business for wireless. Entrust.net manages personal, Web and WAP (for wireless) certificates. In particular, the new WAP Server Certificates are digital certificates that enable WAP servers to establish Wireless Transport Layer Security (WTLS) sessions with mobile phones and micro-browsers that support the WAP standard.
- PGP (Pretty Good Privacy), a popular program available on the Internet that uses public-key cryptography to authenticate users to each other without the use of certificates.
- A number of public-key based cryptographic infrastructure tools, such as the Microsoft and Netscape Certificate Servers, which allow for the inclusion of public-key certificates in various applications.

It is important to understand the role of **Public Key Infrastructure (PKI)** in authentication. Authentication mechanisms include a wide range of options such as user ID and password, one-time pass-tokens, digital certificates, and biometrics. These mechanisms are typically part of Public Key Infrastructure (PKI). PKI capabilities help create and manage asymmetric cryptographic keys or public/private key pairs required by applications. The following major PKI components provide the necessary capabilities to establish, maintain, and protect trusted relationships.

The Certification Authority (CA) creates and signs digital certificates, maintains a list of certificates that have been revoked before the expiration date (certificate revocation lists), makes these certificates and revocation lists available, and provides an interface so administrators can manage certificates.

The Registration Authority (RA) evaluates the credentials and relevant evidence that a person requesting a certificate is who they claim to be. The RA approves the request for issuance of a certificate by a CA. CA and RA functions are provided by a wide range of PKI providers such as Tivoli SecureWay Public Key Infrastructure.

Directory Services, based on the Lightweight Directory Access Protocol (LDAP), define and implement a common schema for users and groups. The directory service is the point of integration for user authentication among products in many security systems. This has a positive effect on reducing administrative costs and complexity. A user can be defined once within an enterprise, and information about that user can be accessed in a consistent manner by multiple different applications. By comparison, in today's environment, common objects must be defined and administered on a per-application basis.

12.2.6 Authorization and Access Control

Authorization is concerned with assuring that only authorized users can access a particular system privilege. Authorization relies heavily on access control – the process of checking whether an authenticated user's privileges permit the execution of a particular operation on a particular protected resource. For example, can Alice withdraw money from account zc-11-35? The access control is typically enforced through access control lists (ACLs) that may look something like the following:

User name	Resource Name
Joe	Payroll
Sam	Accounting
Tim	Inventory control

Scalability of ACLs is a major issue because modern applications may scale to dozens or hundreds of Web servers and potentially tens of millions of end users. The administration of ACLs can be very complex if they must be configured on each Web server system. Authorization to back-end data or subsystems must be handled as well, including systems that have existing authorization mechanisms. In addition, authorization to other key e-business resources such as objects and message queues must be incorporated.

Due to the complexity of managing ACLs, many applications provide access control on their own because it is not always possible to provide intra-application access control using Kerberos or public-key schemes. Some products have been released that make use of the Distributed Computing Environment (DCE) access control policies. These products, such as HP's Praesidium, make use of the fine-grained access control capabilities of DCE and link them to the deployment of Kerberos within a system. Other products such as the Tivoli Secureway Policy Director provides a centralized authorization service that is the point for administering access controls for Web servers, Web applications servers, firewalls, EJBs, and other systems.

12.2.7 Accountability and Assurance

A system needs to log all attempts to access corporate resources to ensure that the system is secure. This logging can also facilitate management decisions by allowing analysis of use patterns. A comprehensive, distributed logging and audit facility for Internet-based applications is needed.

In essence, an e-business must provide assurance that the infrastructure and application resources, including systems, networks, and data, are protected with regard to confidentiality and integrity. This includes protecting the enterprise network and systems from various forms of attacks, and also requires that the communications between the consumer or business partner and the application are secure and confidential. A solution architect can choose from the set of mechanisms discussed so far to satisfy the specific security requirements for the solution. Two additional considerations are:


- Intrusion detection – These services emphasize early detection of intrusions. Should a DMZ, extranet, or any internal system be compromised, you need to detect that fact early, and take necessary actions to prevent the launching of a further attack into the private network.
- Virus detection – Computer viruses can enter your systems in a variety of ways: via email attachments, from software installs, from files brought by employees from home, etc. They can quickly proliferate from system to system or user to user, causing damage to data, applications and networks. Viruses must be identified quickly and isolated, and the damage repaired.

12.2.8 Mapping Security Technologies to Security Needs

The modern cryptographic techniques reviewed so far are used to secure wireless systems. The following table shows a mapping of various security technologies to security needs (i.e., which technologies address which needs).

Table 12-1: Security Considerations – Mapping Technologies to Needs

Technologies	Privacy	Integrity	Authentication and Authorization	Accountability (Non-repudiation)	Availability and Denial of service
Encryption	X		X		
Password protection	X		X		
Digital signatures		X	X		
Message Digest		X			
Digital certificates	X	X	X		
ACLs			X		
Audit trails				X	
Redundancy					X



Time to Take a Break

- ✓ • Security Principles
- Wireless Network Security
- Higher Layer Security and Security Design

12.3 Wireless LAN Security

12.3.1.1 Overview

The wireless LAN industry has grown at a notable rate of between 40 and 60% per year since the mid 1990s and represents around \$2 billion at the time of this writing. The result is a very widespread use of wireless LANs (WLANs), especially the ones based on the IEEE 802.11 standard. Visit any major office building, department store, or hospital, and you will discover 802.11 cards in most PCs and access points hanging from the ceilings. The popularity of Wi-Fi LANs is driven by several factors. First, product prices have decreased dramatically over the past year. Second, new wireless LAN applications are continually being adopted with more corporate and individual reliance on mobile computing platforms. Finally, a strong grassroots movement is building numerous open and free Wi-Fi hotspots around the globe. These no-fee hotspots are springing up in coffee shops, university campuses, and residential areas and are competing with expensive wireless connections from telecom providers [Schmidt 2003]. For example, in mid-2003, the NYC wireless open network (95 active nodes) was competing with the two main pay WLANs operated by T-Mobile USA (120 nodes) and Wayport (3 nodes). See Schmidt [2003] and the websites www.nodedb.com and www.t-mobile.com for additional information on free WLANs.

Figure 12-4 shows a simple wireless LAN configuration. Each station in the wireless LAN has a wireless LAN adapter that operates in certain frequency ranges; connectivity to wired networks is provided through an “*access point*.” Wireless communication is limited by how far signals carry for a given power output. In fact, the longer the distance, the more security concerns arise because more intruders can intercept the signals. This is one of the reasons why Bluetooth is supposed to be more secure than 802.11; as Bluetooth signals cover up to 10 meters, while 802.11 can go as far as 100 meters. At any point in time, a mobile PC equipped with a wireless LAN adapter is associated with a single access point and its microcell, or area of coverage.

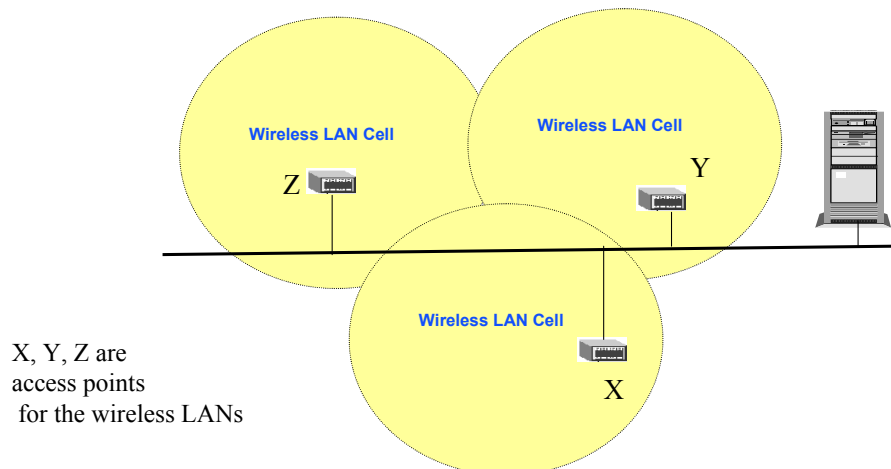


Figure 12-4: A Simple Wireless LAN Environment

There are several areas of concern in wireless LAN security. In particular, the current wireless *access points* present a large security exposure. Although some organizations believe that the security provided by their deployed wireless access points is sufficient to prevent

unauthorized access and use, many researchers have shown otherwise [Cam-Winger 2003, Housley 2003, Arbaugh 2001, LAN97, Walker 1997]. A number of vendors are releasing high-end access points claiming that they address the wireless security problems. Unfortunately, very few products provide enough information to determine the level of assurance that the product will provide. In addition, several products use the un-authenticated Diffie-Hellman (DH) algorithm which suffers from a well-known *man-in-the-middle* attack. The problem is that an attacker can insert himself in the middle of the key exchange between the client and the access point – obtaining the session key, K. Thus the use of un-authenticated Diffie-Hellman introduces a greater vulnerability to the organization's network.

Due to these and other wireless security problems, organizations with deployed wireless networks are vulnerable to unauthorized use of, and access to, their internal infrastructure. The specific areas of vulnerability for 802.11-based wireless LANs are:

- **Random Connectivity.** A user can potentially walk into a building and be connected to the access point by just being in the vicinity. This is unlike wired networks where the computer has to be physically connected to a corporate network.
- **Identity Issues.** Identity is an important part of a security system – without it a malicious outsider can potentially masquerade as a valid user. In WLANs, the MAC address of the WLAN card is used as the only form of identity for both devices and users. Most current open source device drivers allow the users to change the MAC address [Housley 2003]. This creates a security problem.
- **Access Control Issues.** Access control is usually based on ACLs (access control lists) that are based on identity (i.e., the MAC address). Since a MAC address can be changed, a malicious user can access someone else's ACL. Another approach is a "closed network," where a user presents a secret to the access point before gaining access. Unfortunately, the "secret" in WLANs is the access point address that can be easily sniffed.
- **Authentication Issues.** WLANs use a shared key with a challenge and a response for authentication. Several products use the un-authenticated Diffie-Hellman (DH) algorithm for such an approach, but DH suffers from the well-known *man-in-the-middle* attack as stated previously.

Different approaches to wireless LAN security are reviewed in this section. As we will see, many vulnerabilities in wireless LAN security exist that are being addressed at present, but many problems still exist. Practical approaches to achieve WLAN security are also discussed.

12.3.1.2 Wired Equivalent Privacy (WEP)

The Wired Equivalent Privacy (WEP) algorithm, part of the IEEE 802.11 standard for wireless LANs, is designed to protect wireless communication from eavesdropping. WEP is also intended to prevent unauthorized access to a wireless network. Although this is not an explicit goal of the 802.11 standard, it is frequently considered to be a feature of WEP.

WEP is a cipher and relies on a secret key that is shared between a mobile station (e.g., a laptop with a wireless Ethernet card) and an access point (i.e. a base station). The secret key is used to encrypt packets before they are transmitted, and an integrity check is used to ensure that packets are not modified in transit. The standard does not discuss how the shared key is established. In practice, most installations use a single key that is shared between all mobile stations and access points. More sophisticated key management techniques can be used to protect the wireless LAN from attacks; however, commercial systems do not support such techniques widely.

A number of flaws in the WEP algorithm have been found which could seriously undermine the security claims of the system. In particular, a group of researchers at Berkeley

(www.drizzle.com/~aboba/IEEE/wep-draft.zip) found that the following types of attacks against WEP are practical to mount using only inexpensive off-the-shelf equipment:

- Passive attacks to decrypt traffic based on statistical analysis.
- Active attack to inject new traffic from unauthorized mobile stations, based on known plain text.
- Active attacks to decrypt traffic, based on tricking the access point.
- Dictionary-building attack that, after analysis of about a day's worth of traffic, allows real-time automated decryption of all traffic.

Note that these attacks apply to both the 40-bit and 128-bit versions of WEP. They also apply to networks that use the 802.11g standard (802.11g leaves the WEP algorithm unchanged). Based on these experiments, it is recommended that anyone using an 802.11 wireless network not rely on WEP for security, and employ higher-level (e.g., application) security measures to protect their wireless network data. See Cam-Wingert [2003], Arbough [2001], Walker [2000, 2001] for additional discussion of WEP security.

12.3.1.3 802.11i – Overcoming the WEP Weaknesses

All 802.11a, b, and g devices support WEP despite its flaws. To overcome the WEP weaknesses, the IEEE 802.11 Working Group instituted Task Group i to produce a security upgrade for the 802.11 standard. The 802.11i standard includes two main developments (see the 802.11i specifications for additional details):

- **Short Term Solution (WPA).** Most existing 802.11 systems implement WEP in hardware. To address the WEP vulnerabilities on the already-deployed 802.11 networks, the Wi-Fi Alliance has taken a subset of the draft 802.11i standard, calling it WPA (Wi-Fi Protected Access), as a short-range solution. WPA uses Temporal Key Integrity Protocol (TKIP) as the protocol and algorithm to improve security of keys used with WEP. WPA is an interim solution because it does not replace WEP – it adapts WEP protocols to address well-known WEP problems. Users of this short-range solution go through a firmware/driver upgrade to include the TKIP algorithms. The main problem is that WPA might not be backward-compatible with some legacy devices and operating systems.
- **Long-Term Solution (CCMP/RSN).** A long-range solution that replaces WEP instead of adapting it is CCMP (Counter-Mode-CBC-MAC Protocol), also known as RSN (Robust Security Network). This long-range solution uses the Advanced Encryption System (AES) that provides a much stronger encryption and integrity for users. AES uses 128-, 192-, and 256-bit keys and thus is hard to break. It also uses dynamic negotiation of authentication and encryption algorithms between access points and mobile devices, thus making it more secure. The bad news is that AES requires much more processing power – in some cases a separate processor is needed for AES processing.

12.3.1.4 IEEE 802.1X “Network Port Authentication”

IEEE 802.1X “Network Port Authentication” is an IEEE standard (approved in June 2001) that enables authentication and key management for IEEE 802 Local Area Networks, including Ethernet, Token Ring, FDDI, and 802.11. It basically brings the authentication/key management technologies of dial-up networks to the wired and wireless LANs. It is important to mention this development because 802.11i uses the 802.1X port-based authentication for user and device authentication. The link www.drizzle.com/~aboba/IEEE/802-1x-d11.pdf gives the IEEE 802.1X specification (IEEE Standard, as of June 2001), and the Wireless World 2001 and BAWUG Presentations on IEEE 802.1X can be found at www.drizzle.com/~aboba/IEEE/BAWUG.ppt.

IEEE 802.1X is not a cipher, so it is not an alternative to WEP. However, it can be used to derive authentication and encryption keys for use with any cipher, and can also be used to periodically refresh keys. IEEE 802.1X is not a single authentication method; rather it utilizes Extensible Authentication Protocol (EAP) as its authentication framework. EAP is an IETF standard for extensible authentication in network access. It is supported within PPP, IEEE 802.1X, and VPNs (L2TP/IPsec and PIC). The EAP (Proposed Standard, RFC 2284) is discussed in www.ietf.org/internet-drafts/draft-ietf-pppext-rfc2284bis-01.txt. Due to EAP support, 802.1X-enabled switches and access points can support a wide variety of authentication methods, including certificate-based authentication, smartcards, token cards, one-time passwords, etc. Switches and access points act as a “pass through” for EAP, so new authentication methods can be added without the need to upgrade the switch or access point, by adding software on the host and back-end authentication server.

IEEE 802.1X was designed to be scalable – it adds no per-packet overhead because it does not involve encapsulation (unlike PPPOE or VPN). This means that it can be implemented on existing switches and access points with no performance impact. IEEE 802.1X can scale from speeds of 11 Mbps (802.11) to 10+ Gbps, and can be enabled on existing switches with a firmware upgrade, without the need to buy new hardware. IEEE 802.1X also integrates well with AAA (authentication, authorization and accounting) standards such as RADIUS and LDAP. Thus VPNs and RADIUS servers (including Windows 2000 IAS) that support EAP can be used to manage IEEE 802.1X-based network access. Through RADIUS, IEEE 802.1X permits management of authorization on a per-user basis. Information about using RADIUS with IEEE 802.1X can be found at www.ietf.org/internet-drafts/draft-congdon-radius-8021x-17.txt

12.3.1.5 802.11 Wireless Roaming

Roaming, an important aspect of wireless networks, is the ability to connect to multiple ISPs while maintaining an account with only one. To keep the same IP address while roaming, there are approaches at layer 2 as well as layer 3. Mobile IP is the layer 3 approach and dynamic VLANs and tunneling are layer 2 approaches (both are enabled by RADIUS tunneling attributes). “Shared use” access points (APs) need to be enabled to support roaming within 802.11. Shared use APs are important for wireless because they save cost (it costs more to deploy multiple APs in the same location) and also reduce interference (the limited radio channels in 802.11 makes radio interference a potential problem). Standards are in progress that describe, for example, how to do seamless, authenticated fast handoff between 802.11 access points. The link (www.ietf.org/rfc/rfc2194.txt) gives roaming implementations survey and (www.ietf.org/rfc/rfc2477.txt) discusses roaming architecture and requirements.

12.3.1.6 Approaches to Secure WLANs

As discussed above, the Wired Equivalent Privacy (WEP) algorithm, part of the IEEE802.11 standard for wireless LANs, is designed to protect wireless communication from eavesdropping. However, several weaknesses of WEP have been demonstrated. Several other problems also exist as discussed previously.

Given all these problems with equipment built to 802.11 standards, how to safely use it? Here are some thoughts and suggestions (see the example at the beginning of this chapter):

- Make sure that WLANs do not bypass the corporate firewalls. It is best to place the WLANs outside instead of inside the corporate firewalls. This way, any WLAN -- internal or external -- will have to go through the firewall to access any corporate resources. When corporate wired LAN users connect through a LAN switch or hub, there

is an assumption that they already are trusted users because they are inside the firewall. Thus they typically do not go through a firewall. This assumption is not true for WLANs.

- It is best to encrypt the wireless LAN traffic by using solutions such as wireless VPN. In this case, WLAN traffic is authenticated and encrypted in a fashion similar to the dial-up traffic.
- Heavily protect the resources accessed through wireless LANs. For example, higher-level security offered by SSL, PGP, and SET (discussed later in this chapter) can be used to protect Web resources, email, and financial transactions.
- It should be also recognized that some users do not want a great deal of security because it limits their ability to communicate freely with others. For example, the free and open WLAN communities continue to exchange unencrypted information and send IDs and passwords as clear text despite warnings [Schmidt 2003]. In such cases, the current WLAN security is quite adequate.

12.4 Cellular Wireless Network Security

12.4.1.1 Overview

Cellular networks, as explained in a previous chapter, are wireless WANs that establish a connection between cellular users. Figure 12-5 shows a high-level view of a cellular communication network used in wide areas. This cellular network shows the various “cells,” the Base Transceiver Stations (BTSS), and the Mobile Telephone Switching Center (MTSC). Keep in mind that the communication is wireless within a cell only. The bulk of cell-to-cell communication is carried through regular telephone lines. The MTSC typically uses two databases, called Home Location Register (HLR) and Visitor location Register (VLR), to locate the mobile users.

The following security concerns are unique to the cellular networks:

- The call setup information that includes the user ID and other information should be protected.
- The speech and data transmitted during a cellular conversation should be kept private and confidential.
- Privacy of user location should be maintained. The location (cell ID) from where the user is calling should be private, as should the VLR/HLR records that trace where the user has been visiting.
- The calling patterns (e.g., calling home every day at 5 PM to inform your family about when you will be home) should be private.
- The user ID in the cellular networks should be kept private.

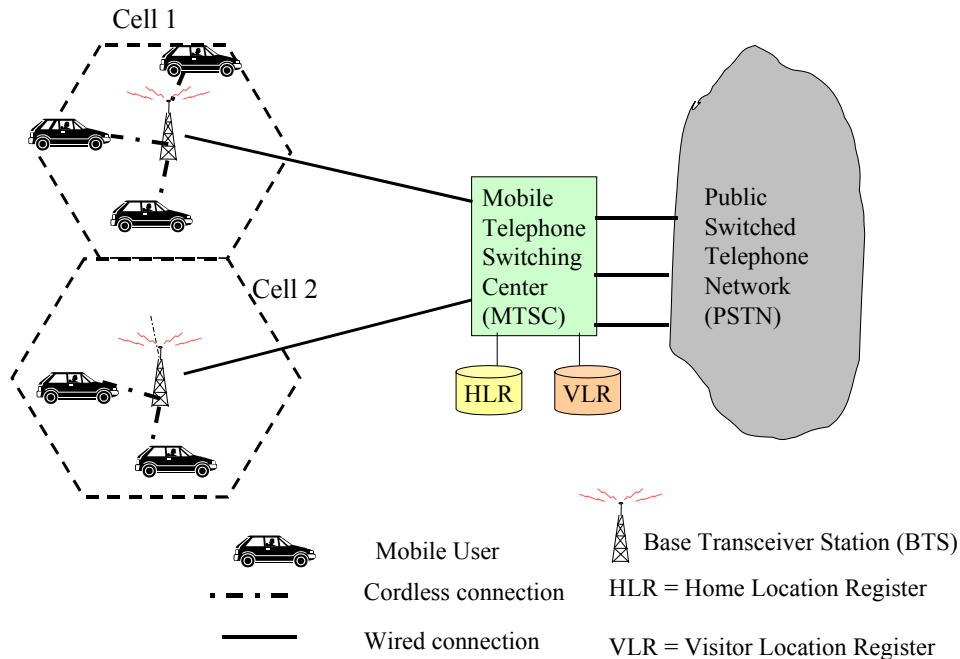


Figure 12-5: A Cellular Communication Network

12.4.1.2 Cellular Network Security Solution Approaches

The security approaches of cellular networks can be discussed in terms of the various generations of cellular networks: :

1G: First-generation wireless cellular: These systems, introduced in the early 1980s, use analog transmission, and are primarily intended for speech over very slow lines (less than 1 kilobits per second). The security for these networks was virtually non-existent. Several hackers were able to capture large amounts of cellular data by just driving around in the neighborhood with a car antenna.

2G: Second-generation wireless cellular: Introduced in the late 1980s, these systems use digital transmission and are also intended primarily for speech. However, they do support low bit-rate data transmissions. The high-tier 2G systems use GSM and the low-tier ones are intended for low-cost, low-power, low-mobility PCS. These systems, most prevalent at present, operate at 9.6 kbps. GSM systems have improved security by introducing three elements: a SIM (subscriber information module) that contains a unique user ID that can be used for authentication, the GSM handset that includes an encryption algorithm, and the GSM network itself that supports encryption. GSM security is described extensively in the GSM recommendations.

2.5G Systems are essentially 2G systems that have evolved to handle medium-rate (around 100kbps) data. As part of the 2.5G initiative, GSM is being extended by the **General Packet Radio System (GPRS)** to support data rates of 112 kilobits per second. Generally, 2.5G technologies have been developed for third-generation (3G) networks, but they are applied incrementally to existing networks. GPRS uses encryption in its core network to avoid eavesdropping. In addition, since GPRS uses packet-switching services, the IPsec services described previously can be used in GPRS. IPsec, as you recall, encrypts the packets before transmission.

3G Systems represent the future broadband multimedia applications and can operate at 2 million bits per second. 3G systems will be based on evolution from 2G – they build on the success of GSM, and dual-mode terminals to ease migration from 2G to 3G are commercially available. 3G system specifications include extensive security features in user equipment and the underlying network.

In essence, the security of cellular networks is improving as the next generation of cellular networks are being introduced.

12.5 Satellite Communication Security

12.5.1.1 Overview

There are several issues that keep satellites from becoming the ultimate wireless WAN. The most important is the turnaround propagation delay (can be about a second) that affects performance of many applications. Also, security questions abound. Satellite security is a major issue in commercial as well as government settings. In particular, security is an important issue in IP over satellite, since an attacker can easily intercept such communication and can even corrupt the transmitted data [Noubir 1998]. Consequently, commercial satellite services have largely been excluded from national initiatives to tighten up the US communications infrastructure (see Wrexler [2002]).

The problem is particularly severe for government agencies. For example, the US General Accounting Office (GAO) released a report warning that the nation's commercial satellites have been largely ignored in discussions of critical infrastructure protection and are vulnerable to attack from hackers [Roberts 2002]. The report, posted on the GAO's website, found critical vulnerabilities in the nation's commercial satellite network. It further suggests that federal agencies using commercial satellites may be exposing sensitive data to unauthorized snooping. Although the government uses encryption to protect satellite communications and employs physical security to protect ground stations, many federal agencies rely on commercial satellite service providers to provide security for tracking satellites and satellite control stations.

The commercial satellite providers fall short of the security standards the government uses to protect its own satellite networks. In addition, government agencies cannot impose specific security requirements on commercial satellite service providers because existing federal laws governing satellite system security apply only to satellites used for national security. According to current policy, federal agencies only have power to secure those satellites that they own. At the same time, the dependence of the federal government on commercial satellites is increasing. Traffic from federal agencies makes up 10% of all traffic handled by commercial satellites. In addition, up to 45% of all federal government traffic between the Persian Gulf region and the US is carried over commercial satellite networks. The GAO report recommends expanding the current federal policy governing satellite security to cover commercial satellites used by government agencies.

Besides the government issues, the privacy of sensitive commercial information that traverses satellites is a concern. For example, point-of-sale transactions and credit card authorizations, Internet traffic, telemedicine information, backhaul mobile traffic, and location-tracking information are all carried over satellites. Satellites also transmit global positioning information to aircraft and air traffic controllers. The Federal Aviation Administration (FAA) plans to use satellite systems to broadcast live cockpit communications from domestic

airliners for national security purposes (see the sidebar “Use of Aircraft Satellite Security Systems”). Security of satellite communications is obviously important.

FAA Use of Aircraft Satellite Security Systems

The Federal Aviation Administration (FAA) plans to use satellite systems to broadcast live cockpit communications from domestic airliners for security purposes. Qualcomm demonstrated a system that uses a global satellite constellation to beam real-time cockpit conversations and potentially live video streams from commercial aircraft to controllers on the ground. This is intended to provide controllers real-time information to help avert a potential hijacking. The task, however, is complex because the satellite equipment would have to be “type certified” for each kind of aircraft, ranging from small commuter planes to jumbo jets. The FAA will also have to rigorously test the new systems to ensure that they do not interfere with the operations of key systems, such as airborne navigation systems. In addition, management of live voice and video data streams from large number of aircrafts would be equally difficult because there are between 35,000 to 40,000 flights a day in the U.S.

Besides cockpit information, other efforts for using satellites for commercial flights are underway. Boeing, for example, is planning broadband Internet service for passengers. The offering, Connexion by Boeing, could be adapted to broadcast real-time images from aircraft cabins and the cockpit to enhance security.

Source: Brewin [2001].

12.5.1.2 Approaches to Secure Satellite Communications

Satellite communications are typically secured through scrambling of satellite signals by using cryptography or spread-spectrum techniques. Spread spectrum, as discussed in a previous chapter, is most widely used in wireless LANs but was developed for military and intelligence operations. The message is “spread” over a range of frequencies to make it jam-resistant – it basically transmits different data bits on different signals, based on a secret scheme, for secure communications. The receiver must know the parameters of the spread-spectrum signal being broadcast to understand the signal. If a receiver is not tuned to the right frequency, a spread-spectrum signal looks like background noise.

For additional security, the messages can be encrypted before transmission and decrypted on reception – a technique used in VPNs (Virtual Private Networks). VPNs, as stated previously, set up a private network over a public network by using encryption. VPNs use IETF IPsec (RFC 2401) and related standards to transport encrypted messages over shared networks. IPsec provides security at the packet level, instead of at the application layer. It encrypts and signs Headers and/or Data parts of IP Header. In addition to spread spectrum and encryption, the security can be also improved by using attack-resistant satellite components and employing better physical security on ground stations.

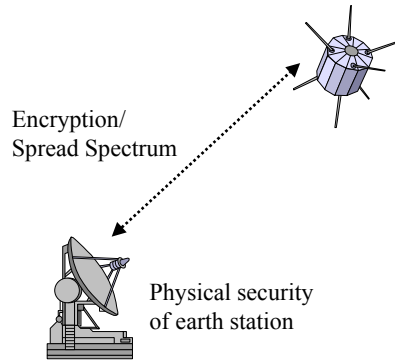


Figure 12-6: Satellite Security Approaches

The main challenge in achieving satellite security is to strike a balance between performance and security [Wexler 2003]. For example, satellite VPNs (Virtual Private Networks) that encrypt messages by using IPsec are good solution candidates. This would allow secure IP VPN services for satellite users and support highly distributed sites. However, TCP was developed for low-delay lines and is very slow when the IP packets are transmitted over satellite because of the high round-trip time of satellite links. To overcome this limitation, several modifications to TCP and other IP protocols have been suggested. For example, some very small-aperture terminal (VSAT) and satellite modem vendors have enhanced their implementations of TCP and HTTP to accelerate throughput. But many IP VPN security solutions do not interoperate with these protocol modifications. Thus, the users have to choose between performance and security.

Some companies, such as Encore Networks, have built satellite VPN appliances that interface to the various enhanced versions of TCP and HTTP. Development of IPsec that does not interfere with TCP is an interesting area of research. For example, a set of rules for optimizing TCP without interfering with IPsec have been proposed [Naubir 1999]. Secure IP over satellite VPNs can be of benefit to many users. For example, a broadband satellite VPN with accompanying service-level agreements (SLA) would be very beneficial for businesses with widely distributed offices.

12.6 WLL and Cordless Security

12.6.1 Wireless Local Loop (WLL) Security

WLLs are examples of wireless metropolitan area networks and offer broadband wireless data rates between 10 to 50 Mbps. WLLs are *fixed wireless networks* where the devices being connected are stationary. Thus there is no need for location services and security does not involve mobility support. Figure 12-7 shows a possible configuration for WLL. A base station antenna, mounted on top of a tall building, serves each WLL cell that consists of residential and business subscribers. A WLL provider can serve one or many WLL cells from its switching center.

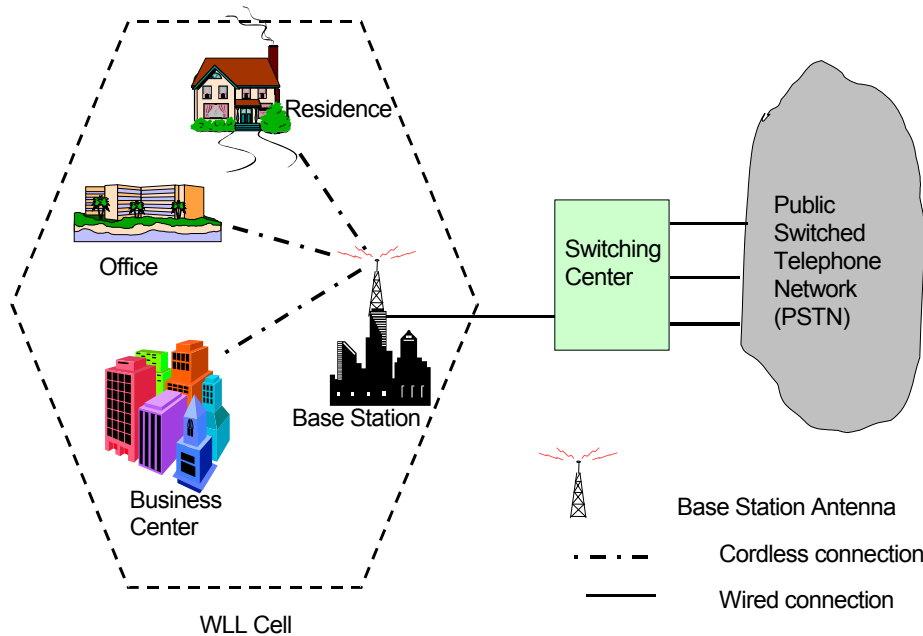


Figure 12-7: WLL Configuration

Several wireless local loops are in operation at present. The best known examples are MMDS and LMDS. Multichannel multipoint distribution service (MMDS) is an older service that operates in the 2.15 GHz to 2.68 GHz frequency ranges and can offer 27 Mbps over 50 km. Local multipoint distribution service (LMDS) is a newer service for 30 GHz (US) and 40 GHz (Europe) frequency ranges and can deliver up to 37 Mbps within 2 to 4 km distances. The security of WLL is at lower layers (layer 1 to 3) and is addressed by the IEEE 802.16 standard. Figure 12-8 shows the abstract reference model that is at the foundation of the IEEE 802.16 specification.

802.16 security is specified through a security manager (SM) that protects against unauthorized access to data transport services by enforcing encryption of the associated service flows across the network. SM employs an authenticated client/server key management protocol in which the SM, the server, controls distribution of keying material to client devices. Security is based on two component protocols:

- An encapsulation protocol for encrypting packet data across the fixed broadband wireless access network. This protocol defines a) a set of supported cryptographic suites, i.e., pairings of data encryption and authentication algorithms, and b) the rules for applying those algorithms to a MAC PDU payload. Encryption services are defined as a set of capabilities within the MAC security sublayer. Encryption is always applied to the MAC PDU payload; the Generic MAC Header is not encrypted.
- A key-management protocol (Privacy Key Management, or “PKM”) providing the secure distribution of keying data from SM to devices. Through this key management protocol, devices and SM synchronize keying data. In addition, the SM uses the protocol to enforce conditional access to network services. A DEV uses the Privacy Key Management protocol to obtain authorization and traffic keying material from the SM, and to support key refresh.

It should be borne in mind that this security is only intended for lower layers (layer 1 to 3) because 802.16 concentrates on lower layers. Intricate details of 802.16 security can be found in the specification on the IEEE 802 website (<http://standards.ieee.org/getieee802/>).

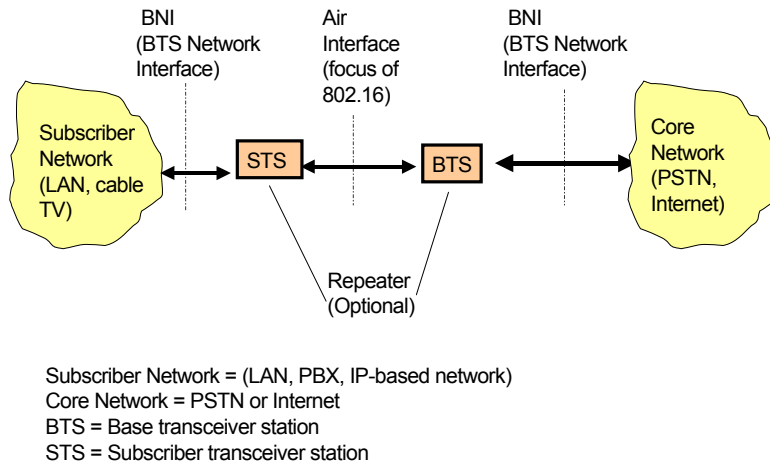


Figure 12-8: IEEE 806.16 Reference Architecture

12.6.2 Cordless Phone Security

Cordless phones are a special class of cellular networks in which the cell sizes are very small (less than 100 meters, typically) and there is no need for location and roaming support. DECT (Digital Enhanced Cordless Telecommunications), developed in Europe, is the most commonly used standard and provides for cordless security. DECT architecture, shown in Figure 12-9, consists of several layers based on the ISO-OSI model. The security is handled at a higher level. In particular, mobility management is responsible for security of DECT communications. It is organized into the following groups of services:

- Identity procedures that are used for the mobile unit to identify itself to the base station
- Authentication procedure that establishes that the mobile unit is a valid network user
- Location procedure that is used in systems with multiple base stations to track location of the mobile unit
- Access rights procedure that establishes that the mobile unit has the right to gain access to a specific type of local or global network
- Key allocation procedure that distributes encryption keys for protecting network control information and user information.
- Parameter retrieval procedure that is used to exchange information about the parameters of the mobile unit and network operation.
- Cipherring-related procedure for encryption and decryption operations.

In general, DECT security is based on GSM security. It added features such as an enhanced key management support and the possibility of the mobile terminal to authenticate the network. A related standard, from a DECT security point of view, is the Terrestrial Trunked Radio (TETRA) standard that has been built on the DECT security. TETRA added features which are relevant for Professional Mobile Radio users, such as end-to-end encryption, encryption for closed user groups and secure enabling and disabling of mobile terminals. A full description of the DECT security model can be found in the formal ETSI standards via <http://www.etsi.org>.

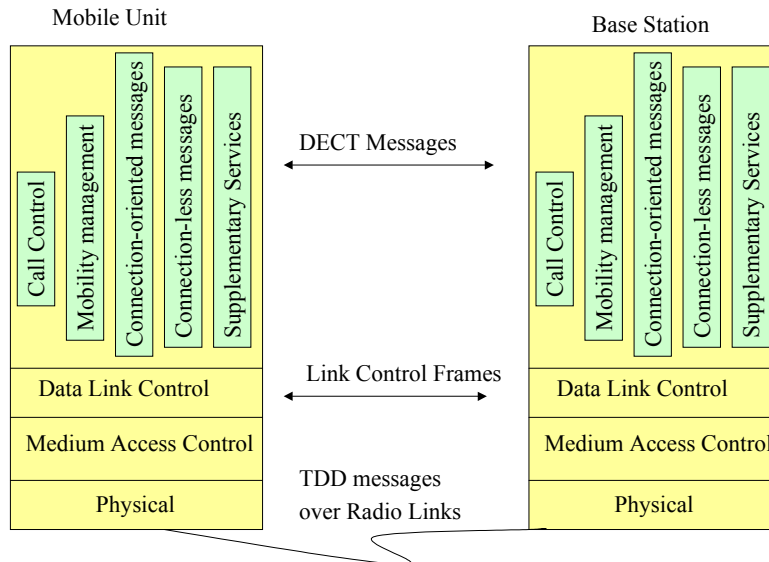


Figure 12-9: DECT Architecture

12.7 Emerging Wireless Network Security

Many wireless networks, as discussed in Chapter 10, are emerging at present. This is a brief discussion of the unique issues and possible security solutions.

12.7.1 Free Space Optics (FSO) Security

As discussed in a previous chapter, Free-Space Optics (FSO) uses high-intensity optical waves (lasers) to transmit information. FSO has emerged as a solution option in the deployment of next-generation wireless networks because of its high availability, bandwidth scalability, and deployment simplicity. As a result of its worldwide license-free operation, combined with a multitude of applications, FSO is providing an attractive and cost-effective option for high speed connectivity between LANs in metropolitan settings.

The main advantage of FSO transmission is that it is among the most secure connectivity solutions. It is virtually impossible to intercept FSO networks at the physical layer. Eavesdropping and physical interception are extraordinarily difficult, and the chance of an attempted intercept being discovered is very high. For these reasons, organizations with serious security requirements (e.g., government and military) adopt free-space laser communication systems for voice, video and broadband data communications. Specifically, there are a number of factors that make intercepting FSO links virtually impossible:

- **Detection Considerations.** FSO laser beams cannot be detected with spectrum analyzers or RF meters. Thus the typical wireless detection and interception systems do not work.
- **Physical Considerations.** FSO laser transmissions travel along a line of sight path that cannot be intercepted easily. An adversary needs to intercept a portion of the transmitted beam to intercept an FSO link, without exposing himself and his equipment. This is not easy because the optical intercept equipment must be carefully placed in the very narrow beam and pointed at the originating transceiver. Because the FSO transceivers are typically installed high above street level, such efforts are quite difficult and the chance of discovery are very high.

- **Signal Considerations.** It is difficult to intercept the laser beam without altering the signal reception. Although it is possible to intercept the beam without bringing down the link, any attempt could be detected as an anomalous power loss at the receiver, which could be used to send an alarm to appropriate network management software.
- **Overshoot Considerations.** An interception could theoretically occur by placing an adversary receiver directly behind the FSO receiver to intercept the energy that overshoots. This intrusion is easily foiled by placing the FSO equipment indoors, behind a window, or by placing a wall behind the FSO receiver.
- **Encryption Considerations.** Even if a determined intruder overcomes the aforementioned challenges, you can still encrypt the FSO messages.

While there is no wireless communication system that can guarantee transmission security, FSO offers an excellent wireless transmission solution for the highest possible level of physical layer security.

12.7.2 Mobile Ad Hoc Network Security

Mobile ad hoc networks (MANETs), as discussed in a previous chapter, provide a different wireless networking paradigm for mobile hosts. Unlike traditional mobile wireless networks, ad hoc networks do not rely on any fixed wireless infrastructure such as access points – ad hoc networking is basically communication between stations without an access point. Instead, hosts rely on each other to keep the network connected. Thus your mobile host can communicate with other mobile hosts just by being in their vicinity. This ad hoc formation of networks without a pre-existing wireless infrastructure is highly desirable in military situations (e.g., a battlefield) or emergency situations (e.g., a building that has been just demolished). However, the principal challenge in design of these networks is their vulnerability to security attacks. The main problem is that two mobile devices in a MANET can start communicating by just being in the vicinity of each other. In particular, MANETS present the following security challenges [Zhou 1999, Ramanathan 2002]:

- **Availability Concerns.** A denial-of-service attack could be launched at any layer of an ad hoc network. On the physical and media access control layers, an adversary could employ jamming to interfere with communication on physical channels. On the network layer, an adversary could disrupt the routing protocol and disconnect the network. On the higher layers, an adversary could bring down high-level services. One such target is the key management service, an essential service for any security framework.
- **Privacy Concerns.** Network transmission of sensitive information, such as strategic or tactical military information, requires confidentiality. Leakage of such information to enemies could have devastating consequences. Routing information must also remain confidential in certain cases, because the information might be valuable for enemies to identify and to locate their targets (other participating nodes) in a battlefield.
- **Integrity Concerns.** A message could be corrupted because of failures, such as radio propagation impairment, or because of malicious attacks on the network.
- **Authentication Concerns.** Due to lack of central control, an adversary could masquerade as a node, thus gaining unauthorized access to resources and sensitive information and interfering with the operation of other nodes.
- **Non-repudiation (NR) Concerns.** It is difficult to define and enforce NR in MANET because the partnering hosts can change positions and roles in a dynamic manner.

Due to these challenges, MANETs are subject to attacks that can lead to impersonations, unauthorized access to secret information, deletion/modification of messages, and injection of erroneous messages. In addition, nodes roaming in a hostile environment (e.g., a battlefield) with relatively poor physical protection can be easily compromised and used to launch

internal attacks. It is not advisable to have a central security authority in a MANET because if this centralized entity is compromised, then the entire network is compromised. Trust relationships among nodes are also difficult to maintain because the topology and membership of MANETs change frequently. For example, nodes frequently join and leave the network, so it is difficult to keep track of nodes that have been compromised.

MANET security is an active area of research [Castro 1999, Desmedt 1997, Sun 2001, Haas 1999, Ramanathan 2002]. Approaches to secure MANETs rely on traditional security mechanisms, such as authentication protocols, digital signatures, and encryption to achieve privacy, integrity, authentication, and non-repudiation of communication. Additional measures are, however, needed. Examples of these measures [Zhou 1999, Haas 1999, Ramanathan 2002, Sun 2001] include:

- Redundancies in the network topology (i.e., multiple routes between nodes) can be exploited to achieve availability.
- Trust needs to be distributed so that no single node is trustworthy – the trust can be distributed to an aggregation of nodes. We can require consensus of at least $n + 1$, assuming that any $n + 1$ nodes are unlikely to be all compromised.
- Nodes can protect routing information through the use of cryptographic schemes such as digital signatures. Routing information needs to be protected because adversaries can inject erroneous routing information or distorting routing information to starve some nodes from getting any information.
- To defend against compromised nodes, redundant routing information is transmitted in the network. Thus, as long as some routing information is correct, it is used to find alternate routes and make the compromised nodes ineffective. This assumes that there are many correct nodes, so that the routing protocol could find routes that go around the compromised nodes.
- Certificate authorities (CAs) are protected rigorously because MANETs rely heavily on encryption for protecting data plus routing information. CAs are important because they are the trusted parties that keep the public/private key pairs for public key encryption (see Section 12.2) – a commonly used encryption approach for MANETs. To avoid compromise of a central CA, the CA functionality is distributed to multiple nodes.

Special measures are needed to handle other special security and availability threats. For example, a “black hole attack” can be launched inside a MANET by a malicious node that advertises itself as always having the shortest path but then swallows the messages so that they are not sent anywhere. To overcome this threat, other nodes can hold “grudges” by spying on suspicious nodes and keeping track of whether they route packets that they receive [Buuchegger 2002].

The main complicating problem in MANET security is that very sophisticated security procedures cannot be employed because the mobile devices have low processing power and low battery lives. Thus only very few of the available approaches can be actually implemented.

Ad Hoc Networking with 802.11

Ad hoc networking is supported by 802.11. Recent IETF work in progress enables hosts to automatically assign IPv4 addresses without a DHCP server, and to resolve names without a DNS server (IPv4 or IPv6). The stations can be linked into a coherent network by either acting as bridges (layer 2 approach) or routers (layer 3 approach). A problem with bridging is that convergence times are large. For example, a 802.11D spanning tree does not converge very quickly to be viable in an ad hoc network where hosts are constantly moving,

associating and disassociating with each other. IEEE work on “rapid spanning tree convergence” is intended to address this problem. This could enable ad hoc networks with dozens or even hundreds of users that could stretch over a substantial geographic distance. The link (www.drizzle.com/~aboba/IEEE/802-1w-d10.pdf) describes rapid spanning tree convergence for ad hoc networks.

12.7.3 Wireless Sensor Network Security

As discussed in previous chapters, wireless sensor networks (WSNs) consist of small, low-powered devices (sensors) that allow the physical environment to be monitored at high resolution. Sensors can be developed to measure temperature, humidity, motion, color changes in a painting, or any other measurable thing. WSNs provide thousands of tiny low-powered devices that collaboratively monitor the environment, predict potential faults in buildings and bridges, and record vehicle movements. The sensors can be “sprayed” in a room or in a vast field. A WSN becomes a powerful component of a corporate Intranet when it connects to the Internet through an IP router, leading to many new possible applications never thought of before.

While these networks provide many new opportunities, they also raise some unique security issues. Why worry about security of wireless sensor networks? Here are some reasons.


- Many sensor applications record and report sensitive data, especially in military and surveillance applications.
- Sensors containing important data can be captured and tampered with, especially in battlefield situations.
- Wireless sensor networks broadcast the information that is not encrypted. Thus others can possibly listen to data by using antennas.
- The WSNs are vulnerable to active attacks which include replay attacks and denial of service.

Thus WSNs need proper inclusion of privacy, integrity, authentication, and authorization controls. However, it is difficult to build a sophisticated security system around WSNs due to the limitations of battery power, computational capabilities, and communication aspects of WSNs. Another special factor is the mobility of sensor nodes – they are quite dynamic in making and breaking connections with other sensors on an ad hoc basis. Existing security solutions such as cryptography and key management are not suitable for WSNs because they were developed for relatively powerful machines with enough resources for intensive calculations. However, since most WSNs use MANET model, the MANET security issues discussed above are applicable.

Many open questions exist in WSN security that need extensive research. For example, Perrig, Stankovic and Wagner [Perrig 2004] list several issues in key distribution and trust setup, secrecy and authentication, robust communication, secure routing, resilience to node capture, and secure group management. They also give a quick survey of promising research approaches. Due to space limitations, it is beyond the scope of this book to discuss these approaches in detail. A few promising developments are (see [Perrig 2004, Zhao 2004, Slijepcevic 2002, Zhou 1999] for details):

- Random key distribution mechanisms in which a large pool of symmetric keys is chosen and a random subset of the pool is distributed to each random node. Two nodes willing to communicate search their pools to find a common key. If such a key is not available, then possibly one of the nodes has been tampered with.

- SNEP (Sensor Network Encryption Protocol) provides data confidentiality, two-party data authentication, integrity, and freshness. SNEP has low communication overhead, has a counter value that is kept at communicating parties, and also provides data authentication.
- μ TESLA provides authentication for data broadcast. μ TESLA overcomes the problem of needing asymmetric encryption (which requires a large computational overhead) by introducing asymmetry through a delayed disclosure of symmetric keys, which results in an efficient broadcast authentication scheme.
- Use of redundant views of the environment and cross check to detect if some nodes have been captured. For example, a network may ask for the same data from different nodes and eliminate the nodes that repeatedly produce fundamentally different data.



Time to Take a Break

- ✓ • Security Principles
- ✓ • Wireless Network Security
- Higher Layer Security and Security Design

12.8 Internet Layer Security Protocols – VPNs and IPSec

Virtual Private Networks (VPN), as indicated above, are private networks (e.g. networks internal to corporations) that use public communication infrastructure. In other words, you set up a *private* network over a *public* network by using encryption. The main idea is that if your messages are encrypted, then the intruder cannot understand them even if he/she looks at them. Transportation of encrypted messages over a public network that spans a multitude of physical networks requires agreements and standards to avoid chaos. Currently available VPNs use IETF IPSec (RFC 2401) and related standards to transport encrypted messages over shared networks. VPNs and IPSec operate at a higher layer (layer 3) as compared to the network access (layer 2) because they encrypt packets that can be routed anywhere. An overview of VPNs and IPSec follows.

12.8.1 Virtual Private Networks (VPNs)

Simply stated, a VPN provides dedicated, secure paths, or tunnels, over a network that is shared by other users. VPN networks consist of authenticated and encrypted tunnels over a shared data network (typically, an IP network). The tunnels are set up between a point of presence (POP), also called a network access point (NAP), and a tunnel terminating device on the destination network. Shiva Corporation's LanRover Access Switch® is an example of a VPN POP. A POP encapsulates packets sent by the user so that the data travels securely over the shared network. A sample VPN is shown in Figure 12-10.

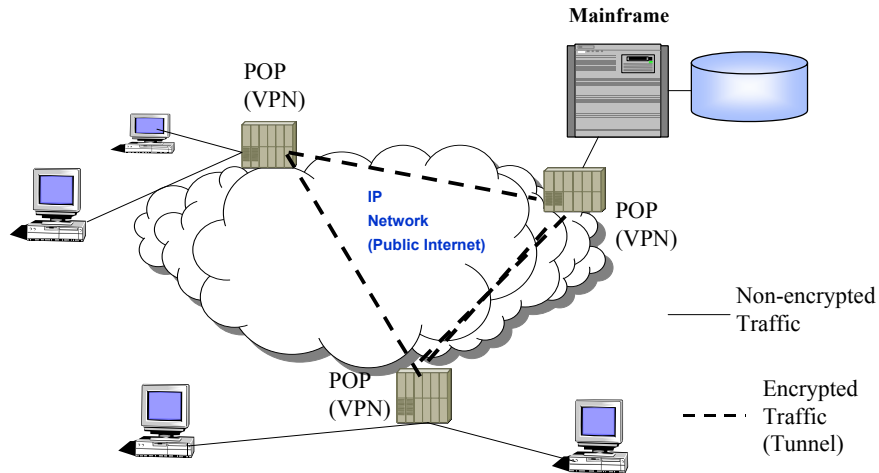


Figure 12-10: A VPN

Early attempts to provide VPN remote access involved simply encrypting every packet. They employed encryption hardware that encrypted and compressed data before it traveled on a shared data network. Current typical VPN configurations establish a secure tunnel between the POP server and a tunnel-terminating device on the local network. The POP server allows you to make a local call. An ISP or a network service provider may own a POP and add encryption/decryption service to provide VPN support. A user initiates a dial-up session to a local POP, where a server authenticates the user and then establishes a tunnel through its Internet “cloud,” which terminates at the edge of the user’s corporate network. The IP packets are encapsulated in a tunneling protocol such as PPTP or L2F (see below), and these packets are, in turn, packaged by an IP packet containing the address of the corporate network – the packet’s ultimate destination. Note that in this case the POP assigns the user an IP address. The encapsulated packets can be encrypted end-to-end by using IPSec or an equivalent protocol. All packaging/unwrapping and encryption/decryption is transparent to the end user.

VPN users have basically two choices: install VPN software at their machine site or use VPN capabilities of an ISP. With a VPN-enabled client, the users install software on their laptops and basically develop an end-to-end tunnel. The advantage of this *Internet service provider-independent* configuration is that mobile users can dial into any traditional POP to establish a VPN tunnel to a corporate network, independent of their contracted service provider. If the software is not embedded in the client – an *ISP-dependent model* – the participating ISPs are required to support VPN technology in the NAP server. The choice between the service provider-dependent and -independent models depend on port availability, backbone performance and client deployment. These considerations are beyond the scope of this book. Visit the VPN Consortium website (www.vpnc.org) for a detailed discussion of the tradeoffs.

VPN POPs use protocols such as Point-to-Point Tunneling Protocol (PPTP) and Layer Two Forwarding (L2F) to encapsulate the data for Internet travel. PPTP is geared toward ISPs (Internet Service Providers) and has provisions for call origination and flow control, while L2F has less overhead and is suited for managed networks. The best features of both protocols have been combined into a new protocol called Layer Two Tunneling Protocol (L2TP). L2TP has provisions for flow control, call origination and secure tunnels across the Internet. The current protocols such as L2F and PPTP, and future ones such as L2TP, do not preclude the use of a Point-to-Point Protocol (PPP) client from having the tunnel-originating functionality embedded in it directly.

Currently, a large number of companies offer VPN services. Examples are Shiva, telecommunication companies (e.g., Southwestern Bell and Nortel), and network service providers such as UUNET. Additional information about VPN can be found at the VPN Consortium website (www.vpnc.org). Although VPN is a favorite choice for physical network security, some issues with VPN security still exist mainly because of multiple VPN implementations in the marketplace. In addition, many IPSec products in the marketplace are proprietary, with poor interoperability. In addition, many of the proprietary extensions have security flaws. A list of Web links to the security analyses of VPN protocols as well as to the IETF standard of IPSec is given in the sidebar, “VPN and IPSec Information Sources”

12.8.2 IPSec

Most of the currently available VPNs are based on the IETF IPSec (RFC 2401) and related standards to transport encrypted messages over shared networks. IPSec is not, however, restricted to VPNs – a corporate LAN within a building can use IPSec by installing IPSec-compliant software on various routers. IPSec-compliant software encrypts and signs Headers and/or Data parts of an IP Packet and specifies security at the packet level, instead of the application level. It thus provides security without requiring changes to applications and is especially suitable for VPNs (see Figure 12-11). IPSec was developed for the next generation of IP (IPv6) but is flexible enough that it is being used in the current versions of IP (IPv4).

The principal feature of IPSec is that it can encrypt and/or authenticate all traffic at the IP level. Thus all applications that use IP (email, Web access, file transfer, etc.) can be secured. IPSec encompasses the following functional areas at the IP level:

- Authentication: Ensure that the received packet was, in fact, transmitted by the party identified as the source in the packet header. In addition, this mechanism ensures that the packet was not modified in transit.
- Integrity: Ensure that the data is not modified in transit.
- Confidentiality: Encrypt messages to prevent eavesdropping by third parties.
- Key management: Ensure secure exchange of keys.

To provide privacy and authentication services at the IP layer level, IPSec is typically implemented at the network router level or in a “firewall” that serves as the main entry point into a system. When implemented in a firewall, IPSec provides strong security that applies to all traffic crossing the firewall. If the firewall is the *only* way to enter the system, then you have very strong protection by making the firewall IPSec-enabled. In addition, since IPSec runs below the TCP/UDP layer, no change is needed on the application software for added security. In large-scale systems this is very valuable because *all* applications can be secured without any changes. This does not address the different security needs of different applications. For example, email may not need the same level of security as a corporate retirement system. Those special needs have to be addressed at the application level.

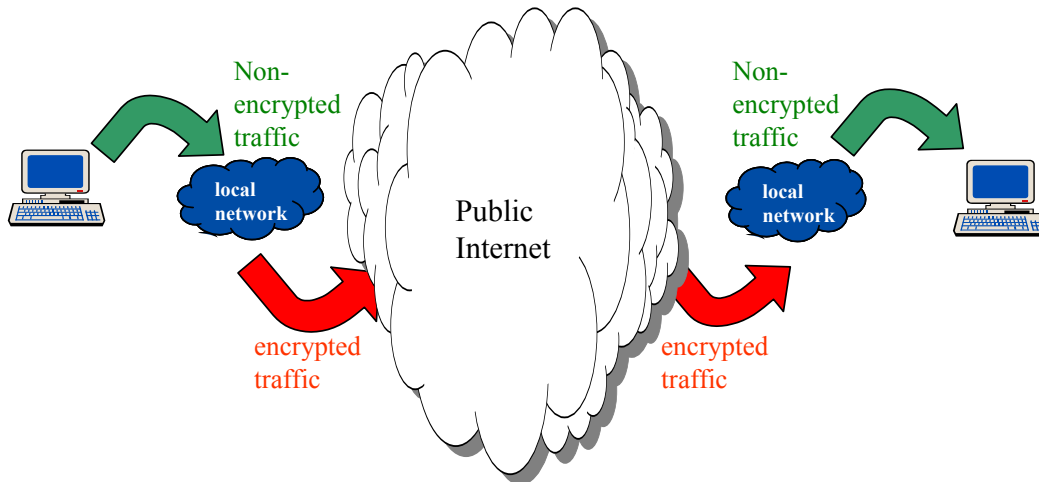


Figure 12-11: IPSec Conceptual View

Figure 12-12 shows a more detailed view of how IPSec can be used in enterprise networks. The IPSec-compliant software is installed in a set of network devices (routers). In addition this software can be directly installed in a user workstation. The main purpose of this software is to generate and process the encrypted packets that have the following format:

- IP Header – this indicates the regular IPv4 or IPv6 header that shows the origin and destination addresses.
- IPSec Header – this header is generated by IPSec software and itself can consist of two headers: an Authentication Header (AH) used to describe the authentication to be used, and an Encapsulating Security Payload (ESP) to describe the encrypted payload. AH and ESP headers will be described later.
- Secure IP Payload – this is the actual data that has now been encrypted.

LAN1 and LAN2 (in the following figure) generate the regular IP traffic with packets that contain the IP Header and the the IP Payload. These packets are transformed by the IPSec-enabled devices (routers) into IPsec packets by adding a new header (IPSec Header) and encrypting the payload. These packets are then sent over the public Internet (treated as a VPN) or over a private network. The IPSec-enabled workstations directly generate and interpret the IPsec packets.

IPSec Versus SSL

A commonly asked question is: how does IPSec differ from SSL? Although we will discuss SSL later, it is good to note here that IPSec differs from SSL in that it creates a secure channel between two TCP/IP *hosts* over which *multiple* TCP/IP connections can be established. Each TCP/IP session itself may or may not use SSL. This also implies that IPSec can authenticate machines but not users because it is too low-level.

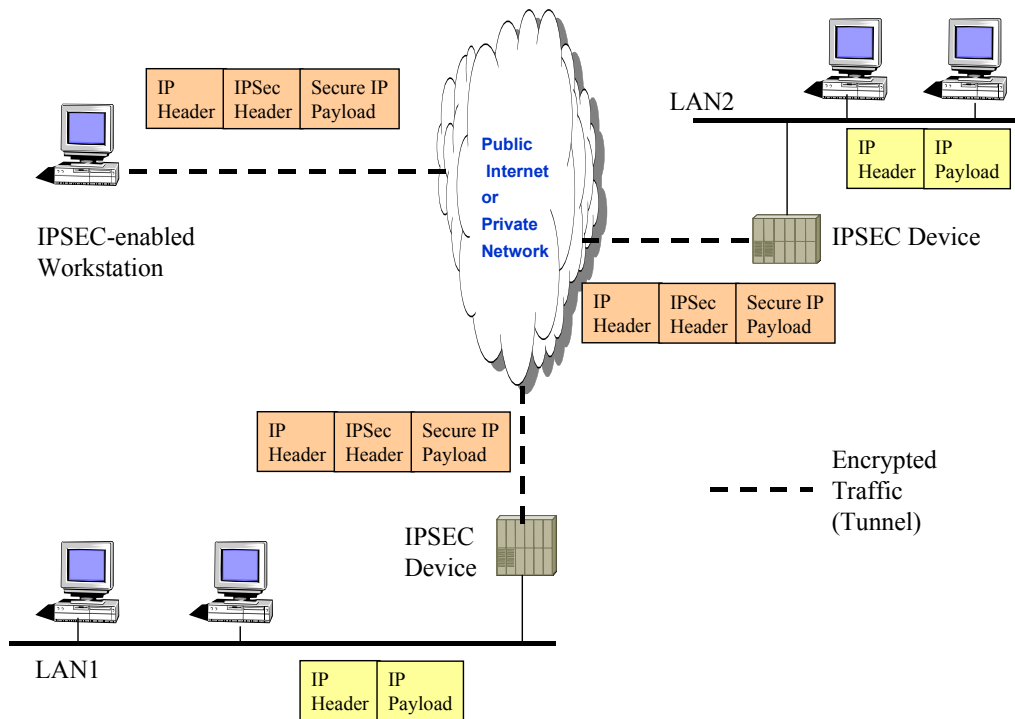


Figure 12-12: A More Detailed View of IPsec

12.9 Wireless Middleware Security

12.9.1 Overview

Wireless middleware, as discussed in previous chapters, is the set of software routines that reside above the network and below the applications to provide connectivity of mobile users to Web content, databases, and applications. Security is the main concern of wireless middleware. However, different wireless middleware packages such as WAP and i-mode provide different security approaches in terms of authentication, data integrity, and data privacy. We discuss these approaches in this section. A review of SSL is presented first because SSL is used directly by some wireless middleware services such as i-mode, has been extended by WAP, and also fills in the gaps where necessary (e.g., between WAP gateways and Web servers).

12.9.2 Secure Socket Layer (SSL) for Wireless Web Security

12.9.2.1 Overview

Secure Socket Layer (SSL), also known as Transport Layer Security (TLS), is by far the most heavily used security technology for the World Wide Web. It is also used in wireless systems such as i-mode. At present, SSL is being packaged with almost all Web browsers (Netscape Navigator, Microsoft Internet Explorer) and servers (Apache, IIS). SSL runs on top of TCP/IP and manages secure messaging on the network (see Figure 12-13). The SSL protocol provides data encryption, server authentication, message integrity, and optional client authentication for a TCP/IP connection. SSL consists of software installed in browsers and on servers. All

major browsers and servers today are “SSL capable.” If needed, SSL software can be obtained by subscribing to a Secured Service Provider such as www.ssl.com or by obtaining a Server Certificate from www.ssl.com and installing it on an existing secured server. The SSL protocol provides, as we will see, a wide range of encryption and authentication choices to ensure that communications between a client and a server remain private based on user requirements. The cryptographic choices are known as “cipher suites.” A user can select a cipher suite when establishing an SSL session.

From an end-user point of view, the screen appearance of your browser with SSL is very similar to the one without SSL. To use SSL, you just need to type “https” instead of “http.” For example, the link “<https://www.fedex.com>” connects you to the Federal Express website over SSL. If an SSL connection is successful, a lock appears in the bottom left part of your browser – the rest of your screen looks just about the same. Once an SSL session is established, all Web server-to-client traffic (both ways) is encrypted. This includes:

- URL of the requested document
- Contents of the requested document
- Contents of any filled-out forms
- Cookies sent from client to server
- Cookies sent from server to client
- Contents of the HTTP header

Thus SSL provides a great deal of confidentiality. However, you cannot hide that a particular browser is talking to a particular server. If this type of privacy is needed, then you should use a proxy server for anonymity.

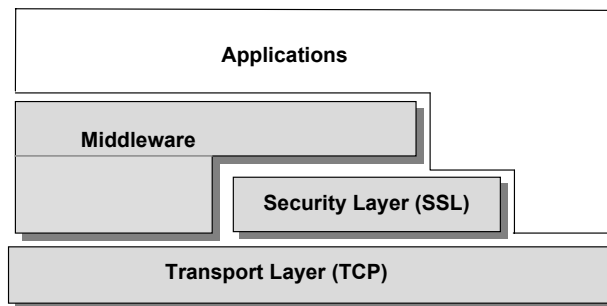


Figure 12-13: SSL

12.9.2.2 How SSL Works

A secured server uses Secure Sockets Layer (SSL) technology to provide a safe way to transmit sensitive information, such as credit card numbers, online banking, email messages, surveys and other personal information. SSL client and server negotiate the encryption scheme and key size. SSL uses RSA (Rivest, Shamir, and Adleman) Public encryption for key session negotiation and DSA (Digital Signature Algorithm) for session encryption. In reality, SSL gives users many cryptographic choices, such as the following:

- Symmetric key (for encryption) can be DES, triple DES, or others.
- Asymmetric key (for authentication) can use the RSA public key and certificates.
- Message digest (for integrity) can use the MD5 or SHA algorithms.
- Various key lengths are supported for conformance to different (especially overseas) secure websites.

These choices are known as “*cipher suites*” – each cipher suite has a different security strength. For example, the cipher suite (DES-RSA-MD5) in SSL 3.0 represents a security

option with very high strength. Each Web browser and server supports several cipher suites. When an SSL client connects to a server, they both negotiate a cipher suite that is strongest but available on both sides. A common problem is that international websites have smaller key lengths (e.g., 40-bit). Thus the SSL session uses 40-bit keys even though higher key lengths are available on the Web browser.

Let us go through the information flow between a client and a server when a client clicks on, for example, <https://www.fedex.com>. Figure 12-14 shows the exchange of messages between the two parties to establish an SSL session and to display the lock at the bottom of the browser.

- 1. Send Client Hello.** The client (Web browser) opens a connection and sends its capabilities, i.e., the cipher suites it supports.
- 2. Respond with Server Hello.** The secure server responds to the client after determining the most suitable cipher suite. The server selects the highest cipher suite that is supported by the client and the server. The server sends the cipher suite selected to the browser. The server also sends a session ID to be used. If a mutually agreeable cipher suite is not found, then the server sends the “handshake failure” message and disconnects.
- 3. Server sends certificate.** The server sends a signed X.509 site certificate to the client to identify itself. Almost all servers at present have signed certificates.
- 4. Server requests client certificates (optional).** This optional step is used if the client also has a signed certificate. Client-side certificates are gaining popularity slowly.
- 5. Send client certificate (optional).** This optional step sends the client-signed certificate to the server.
- 6. Send client key exchange message.** The client selects a suitable symmetric key for encryption. This key is used to encrypt/decrypt the messages. This key is encrypted by using the server public key (recovered from the server certificate) and is sent to the server.
- 7. Send a client “certificate verify” message (optional).** The client sends its certificate to acknowledge that it knows the symmetric key.
- 8. Change cipherspec message.** The client as well as the server exchange this simple message to indicate that now they are ready to start communication.
- 9. Send finished message.** The client and server send the MD5 and SHA hashes of all messages exchanged so far. This confirms that no messages have been compromised in this conversation.
- 10. Exchange traffic.** The famous lock appears now and the two sides now start communication.

After step 9, an SSL session is established, and all Web server-to-client traffic (both ways) is encrypted by using the encryption key chosen in step 6. You also have the SSL icon (the famous lock) on the bottom of your screen. As indicated previously, all SSL communications are encrypted, including URLs of the requested document, contents of the requested document, contents of any filled-out forms, cookies sent from client to server, cookies sent from server to client, and contents of the HTTP header. Thus you can have secure Web communications.

It should be noted that SSL is not flawless. Several defects were found in the earlier versions of SSL and it was cracked a few times when it used only 40-bit keys. Most of these defects *appear* to have been fixed, and longer key sizes are widely supported at present. I have not seen a recent story about SSL being cracked – so far, so good.

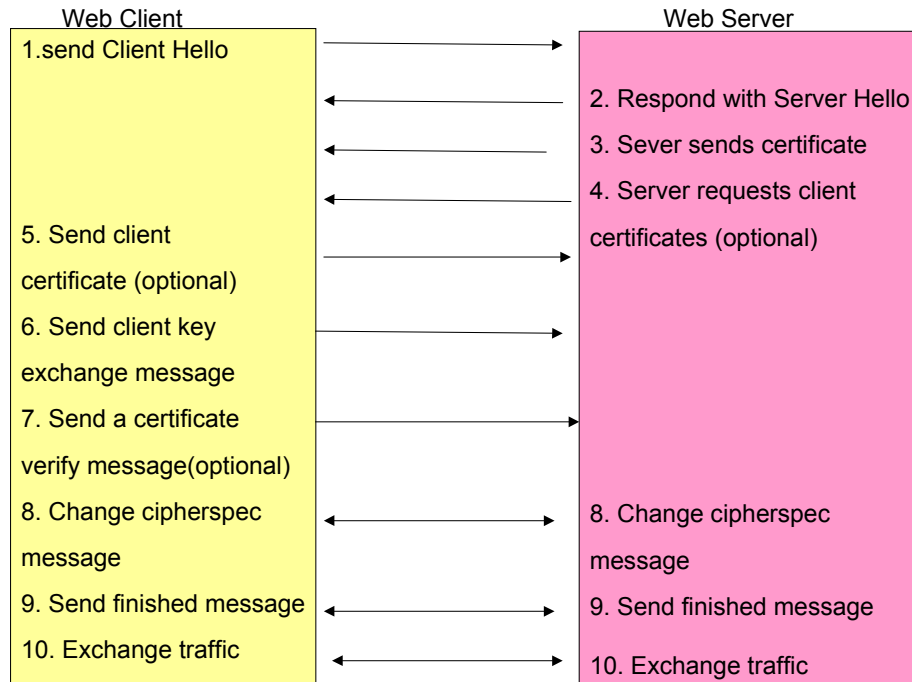


Figure 12-14: Flow of an SSL Session

12.9.3 WAP Security and WTLS

12.9.3.1 WAP (Wireless Application Protocol) Security

WAP is a set of protocols to enable the presentation and delivery of wireless information and telephony services on mobile phones and other wireless devices. Three main constraints make this market different from the wireline market. First, the wireless links are typically constrained by low bandwidth, high latency, and high error rates. Second, the wireless devices are constrained due to limited CPU power, limited memory and battery life, and the need for a simple user interface. Third, wireless networks introduce challenging security issues, as discussed in previous sections.

WAP specifications address these issues by using the existing standards where possible, with or without modifications, and also by developing new standards that are optimized for the wireless environment where needed. The WAP specification has been designed such that it is independent of the air interface used, or of any particular device. A WAP gateway serves as the “middleman” for WAP by translating the WAP to non-WAP (Internet-HTTP) protocols through adapters; it also enforces WAP security (see Figure 12-15). A detailed discussion of WAP was given in a previous chapter.

WAP should be analyzed for potential intrusion threats due to the weaknesses of the wireless security model. The WAP specification ensures that a secure protocol is available for transactions on a wireless handset. The Wireless Transport Layer Security (WTLS) protocol is based on the industry-standard Transport Layer Security (TLS) protocol, more popularly known as Secure Sockets Layer (SSL). WTLS is intended for use with the WAP transport protocols and has been optimized for use over narrow-band communication channels.

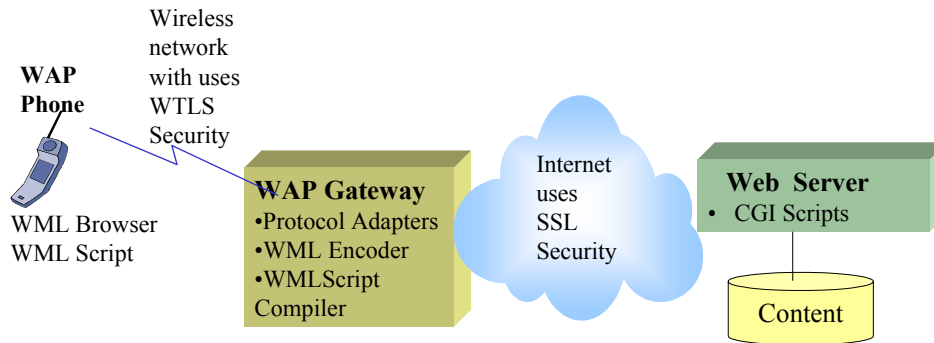


Figure 12-15: Conceptual View of WAP Security (WAP handset to Gateway uses WTLS, Gateway to Web Server uses SSL)

WTLS is not SSL, so it cannot directly communicate with SSL-enabled Web servers. As shown in Figure 12-15, WTLS works between the WAP client and the WAP gateway. The traffic from the WAP gateway to the Web server is typically protected by using SSL. Some implementations of WAP have a client-proxy-server model architecture where the proxy can be used to present a simplified view of familiar websites. An important security function performed by a proxy is that it unwraps the WAP WTLS secure data from the client and then rewraps it into SSL/TLS before passing it to a Web server. For Web applications that employ standard Internet security techniques with TLS, the WAP gateway automatically and transparently manages wireless security with minimal overhead.

WTLS can provide end-to-end security between WAP protocol endpoints. End-to-end security is achieved through two approaches: a) browser and origin server directly communicate using the WAP protocol, or b) a WAP proxy is trusted – by being, for example, located at the same physical secure place as the secure origin server.

WAP components can be attacked at several levels. Examples of the components that can be attacked are WAP clients and servers, the WAP gateway, and WAP messages. For example, intrusion of the WAP gateway can have a very high impact on WAP users. It is important to secure the WAP gateway through high levels of security. It should also be replicated. In addition, intrusion of WAP clients, servers and messages could have high impact. It is important to use authorization, authentication, and encryption by using WTLS. The implementation of WTLS by vendors needs to be watched.

12.9.3.2 A Closer Look at WTLS

WTLS ensures data integrity, privacy, authentication and denial-of-service protection – it does not support non-repudiation. The WTLS specification is designed to work even if packets are dropped or delivered out of sequence – a more common phenomenon in some wireless networks. Another issue is that some WTLS messages can be sent without authentication of origin. WTLS provides for client or server authentication and allows for encryption based on negotiated parameters between the handheld device and the WAP gateway. Users can implement any of the following three classes of authentication types:

- Class 1 (anonymous authentication). The client forms an encrypted connection with an unknown server. This has limited use (mainly for testing purposes) because end users have no way of determining the identity of those to whom they are talking.
- Class 2 (server authentication). Once clients are assured they are talking securely to the correct server, they can authenticate using alternative means, such as a user name/password. This is a very common model for WTLS usage. Keep in mind that

WTLS certificates are not the same as X.509 certificates, and they cannot be used interchangeably.

- Class 3 (server-and-client authentication). The server and the client authenticate each other's WTLS certificate. This is the strongest class of authentication. Client certificates required for Class 3 authentication pose special management problems because the key pairs must be generated and managed on the handheld device (see the sidebar, "Maintaining WTLS Certificates on Mobile Devices").

The WTLS specification does specify cryptographic algorithms that may be supported by WAP devices, but does not require this feature. For example, the WTLS specification provides support for the RSA and Diffie-Hellman key exchanges, but most vendors are supporting RSA because of its widespread use. Similarly, several bulk encryption ciphers are specified; however, DES and 3DES are used most widely. In addition, WTLS supports various key lengths used with the bulk encryption algorithms, so that the security parameters can be negotiated based on user needs. The main consideration in WTLS security is to make low CPU-powered wireless devices secure by making the cryptography efficient. Because PDA and cell phone CPUs are typically slow, using SSL from end to end can take more than a minute, depending on the key size used to negotiate an SSL connection. Specialized cryptographic algorithms such as Elliptic Curve (EC) cryptography are more promising than RSA for CPU-starved PDAs and cell phones because they require far fewer resources.

Unlike SSL, WTLS does not provide for end-to-end security between WAP clients and Web servers. End-to-end security means the client and server have a secure session, without any intervening servers. When your Web browser sets up an SSL session with a Web server, the browser and Web server are communicating directly. As discussed previously, when you send your credit-card number over SSL, in effect, only the receiving Web server can receive it. WTLS works between the WAP client and the WAP gateway, as shown in Figure 12-15. The WAP gateway terminates the WTLS sessions and initiates SSL sessions to the destination Web server. The potential problem exists at the WAP gateway. Between the time the data is decrypted and "decapsulated" from WTLS and WAP and re-encapsulated and re-encrypted in SSL, the protected data is exposed – albeit for only a very short time (a fraction of a millisecond). For most applications and users, this should not be a problem because to access this data, someone has to break into the WAP gateway.

Maintaining WTLS Certificates on Mobile Devices

Client certificates required for WTLS Class 3 authentication pose special management problems because the key pair associated with the client certificate resides only on the client. First, the key pairs must be generated on the mobile device (or generated and loaded onto the mobile devices). Second, the client certificate has to be safeguarded and managed until the certificate expires. This creates several additional problems. The client certificates can be retained on the handheld device, raising concerns about theft. Alternatively, the client may refer the WAP gateway to a directory to retrieve the client certificate from a directory. This saves the communication bandwidth needed to send the client certificate over the air; however, the WAP gateway must trust the directory the client refers to in order to assure authentication. The certificate directory also must be available at all times to allow users to retrieve the certificate when requested.

12.9.4 i-mode Security

i-mode, from NTT DoCoMo (<http://www.nttdocomo.co.jp/>), is a competitor to WAP. I-mode security is important because it has millions of users. In particular, mobile commerce (m-commerce) is conducted on i-mode, including mobile banking and security trading. From a security point of view, i-mode consists of the following features (see Figure 12-16):

- Security of the radio link between the i-mode handset and the cellular base station is provided through proprietary protocols and encoding controlled by NTT DoCoMo. Digital radio packets sent between handsets and radio towers are encoded via this proprietary NTT DoCoMo scheme.
- Encryption and authentication of the data between i-mode handsets and Docomo websites is supported by SSL. To support 128-bit encryption in SSL, Docomo has embedded digital certificates into its cellular phones. Digital certificates supplied by companies such as VeriSign and Baltimore Technologies are built into the i-mode phones such as the 503 cell phones and others. The SSL-based authentication and encryption make i-mode sessions secure when users are accessing websites that offer digital certificates.
- Security of private network links between the i-mode center and special service providers such as banks is sent via 128-bit SSL-encrypted dedicated lines, typically over wired networks. This transmission is between the i-mode server and the bank server that does not use radio packets through the air.

I-mode network and i-mode handsets are equipped for SSL (secure socket layer) encrypted transmission, and i-mode handsets have unique identifiers allowing similar security to be implemented as on the wired Internet. Mobile banking on i-mode and corporate networks usually uses SSL-encrypted communication.

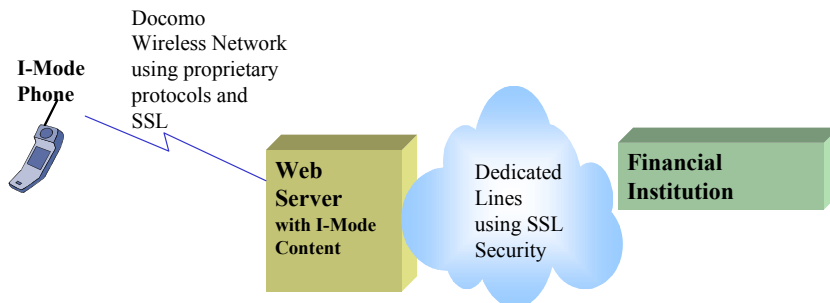


Figure 12-16: i-mode Security – Conceptual View

12.9.5 Wireless VPN Versus WAP Security

Wireless VPNs work well in situations where you do not have a WAP gateway or if you have to support mobile users who do not have a WAP/WTLS microbrowser. On the other hand, wireless VPN is mostly restricted to handheld devices such as Palm Pilots because cell phones do not have the processing power or memory to run VPN software. The success of PDA-based VPN clients largely depends on the ease of use and VPN efficiencies that can be achieved on low-powered PDAs. VPN client software for the Palm and Palm Pilot is commercially available from companies such as Geritome and Top Gun.

12.10 Wireless Application Security

12.10.1 Overview

Mobile applications allow handheld devices to access and use a wide range of databases and applications for e-banking, retail payment, brokerage, and e-business. Examples of the mobile applications are:

- *Mobile Enterprise Business Applications (MEBAs)* add the mobility dimension to EB applications such as ERPs, SCMs, CRMs, etc. This allows employees, partners, and customers to use mobile devices such as laptop computers, personal digital assistants (PDAs), and digital telephones to conduct business.
- *Mobile Commerce (M-Commerce)* applications allow cellular phones and PDAs to search the Internet, access data and information, and conduct purchasing or business transactions.
- *Voice Commerce (V-Commerce)* is gaining importance to support users who want to use telephones and other voice-driven devices for conducting e-commerce. This includes “Voice Portals”.
- *Positional Commerce (P-Commerce)* is becoming popular to provide support to the customers based on their geographic position (e.g., to give you information about deals in the Atlanta area when you are in Atlanta). The systems use a GPS (Geographical Positional System) to locate the position of the customers.
- *Mobile agents* roam around the network looking for information and bargains on behalf of the customers. Most mobile agents are Java applets that go from one computing system to another.

Many of these applications raise security and privacy concerns. In particular, online transactions from handsets can result in fraud and theft, with huge financial losses. An important area of concern is the mobile agents because they move from site to site and thus introduce numerous security risks of viruses, time bombs, Trojan horses, and the like. Due to these concerns, the use of mobile agents in enterprise computing is greatly restricted. In addition, the positional commerce raises some privacy issues (you may not want everyone to know that you are visiting Atlanta). Additional concerns about mobile users (e.g., increased chance of eavesdropping) also exist.

To discuss mobile application security risks and approaches, let us review Figure 12-17, which shows a conceptual view of mobile applications. The main difference is at the front end (i.e., the wireless network and the front-end integration layer). Thus the network and the front-end layer must be able to handle all mobile security concerns. We have already discussed the wireless network issues in a previous section, so let us focus on the front-end layer that must support mobility-specific software in a secure manner.

The mobility-specific software is responsible for “roaming support” (e.g., the GIS Map for GPS), voice support for voice commerce, and uniform access to the CRM, ERP and proprietary or custom-developed business/commerce applications. The security features vary widely. For example, if WAP was used, then WAP security as discussed previously could be used; but if other technologies such as i-mode are used, that lack security support, then the deficiencies need to be compensated for. The best approach in this situation is to build a separate “mobility” proxy that performs the front-end integration task and also provides the necessary security checks for integrity, privacy, and other requirements. The mobility proxy could also serve as a firewall (an application-level gateway) that verifies and filters the traffic before attaching to the back-end systems.

Let us discuss wireless security in terms of mobile client security, Web-tier security, and the back-end transaction security.

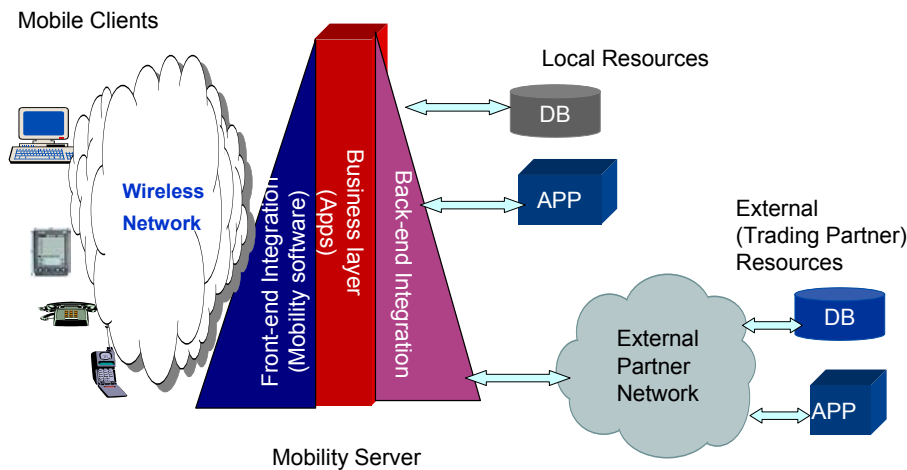


Figure 12-17: Conceptual Architecture for Mobile Applications

12.10.2 Mobile Client Security

Mobile client software residing on handheld devices is responsible for identifying users with passwords, or with biometrics such as voice recognition. In addition, the encryption/decryption software as well as digital certificates, if employed, reside on the handheld devices and are the responsibility of client software. Typical threats to client-side security are based on active content that appears as:

- Helper applications and plug-ins that are used to handle special programs (e.g., Powerpoint) that are not shipped with Web browsers
- Java script and VB script that are used to embed executable code in HTML pages
- XML processing on the client side that uses Extensible Stylesheet Language (XSL) to convert XML to HTML
- Java applets that can do a variety of tasks on the browser (e.g., show video clips, draw graphs and charts)
- ActiveX control applets that control the display of MS Desktop Services
- XML/SOAP clients that access remote resources in the MS Dot Net environment
- Browser-side cookies that keep track of the activities at browser sites

Any of these programs can be compromised, replaced, and/or modified to show undesirable behavior. Due to space limitations, it is beyond the scope of this document to discuss this content in detail. We mention a few for completeness.

Client certificates required for client-side authentication pose special management problems because the key pair associated with the client certificate resides only on the client. Many systems such as WAP and i-mode produce certificates that are stored on the clients (PDAs, cellular phones). The main problem is that the client certificate, once stored on the handset, has to be safeguarded and managed until the certificate expires. This creates several additional problems because the client certificates go with the device if the device is stolen. Another problem is that the key pairs must be generated on the mobile device. These certificates can also be generated and loaded onto the mobile devices. Instead of certificates on mobile devices, the client may refer the wireless gateway to a directory to retrieve the client

certificate from a directory. This saves the communication bandwidth needed to send the client certificate over the air; however, the wireless gateway must trust the directory the client refers to in order to ensure authentication. The certificate directory also must be available at all times to allow users to retrieve the certificate when requested.

Java security is an important aspect of client-side protection for mobile devices. Security of Java code has been an area of concern for a while and is important for handheld devices because some phones, such as i-mode phones, include Java code. In addition, wireless Java, part of the Sun J2ME (Java 2 Micro Edition) is used in many handheld devices. Current Java security is defined at the following levels:

- Java 1.1's security management system. All local code is trusted. All remote code is untrusted, unless it is digitally signed by a trusted source. Untrusted code runs in a "sandbox," and has limited access to local system resources.
- Java 2's security management system. Local and remote code are checked by the same security management system. It enforces fine-grained, flexible and easy-to-specify security and permission policies.

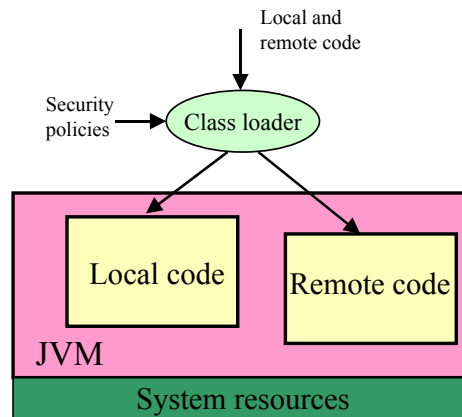


Figure 12-18: Java Security Model

12.10.3 Web Server Tier (Middle Tier) Security

There are many serious issues on the Web server (middle tier) sites when accessed from mobile devices. Examples of the issues are:

- HTML content that may be accessed and modified by unauthorized users.
- XML content that may represent customer data may be accessed and modified by unauthorized users.
- CGI scripts, Servlets, JSPs (Java Server Pages) and ODBC/JDBC drivers may be modified to access different databases and applications.
- EJBs (Enterprise Java Beans) containers can be contaminated to disable large applications.
- Server logs can be accessed to review confidential information and can even be modified if not protected properly.

When data must travel outside of a secure system environment, it needs to be protected so that the policies governing its use cannot be violated. Secure communications, ensuring data privacy, data integrity, and origin authentication are an important aspect of information protection.

Web Services Security and SAML (Security Assertion Markup Language). Web Services (WS) Security is an area of considerable activity at present, and the interest in this activity is growing with time. For example, IBM and Microsoft have published a joint white paper on “Security in a Web Services World: A Proposed Architecture and Roadmap” (msdn.microsoft.com/library/en-us/dnwssecur/html/securitywhitepaper.asp). This document defines a Web Service security model that supports, integrates and unifies several popular security models, mechanisms, and technologies (including both symmetric and public key technologies) in a platform-neutral manner. The specifications build upon foundational technologies such as SOAP, WSDL, XML Digital Signatures, Kerberos, XML Encryption and SSL/TLS. This allows Web service providers and requesters to develop solutions that meet the individual security requirements of their applications.

Security Assertion Markup Language (SAML) is at the core of the WS Security architecture. SAML is an open security specification developed by the Organization for the Advancement of Structured Information Standards (OASIS – www.oasis-open.org). It uses XML to exchange security information in the form of assertions. SAML sponsors include companies such as Boeing, Citrix, HP, IBM, Microsoft, Netscape, and Sun. SAML’s stated purpose by OASIS is “to define an XML framework for exchanging authentication and authorization information.” SAML concentrates on authentication and authorization, not message integrity and confidentiality. In practical terms, SAML provides standardized architecture for secure communications, helps secure Web Services, and provides Single Sign On (SSO) in WS environments. Since SAML does not directly provide message integrity or confidentiality, it relies on XML Signature to protect integrity and on SSL (Secure Sockets Layer) and TLS (Transport Layer Security) for confidentiality.

SAML is designed to exchange security information in the form of assertions, stated in XML, about subjects (a person or computer that has an identity in the security domain and needs access to a protected resource). Three types of assertions are supported: a) an authentication assertion indicates how and when the entity’s identity was proved, b) an authorization assertion declares that the entity is (or is not) authorized to access specific resources, and c) an attribute assertion provides other information about the entity, such as membership in a group. Assertions in the SAML token are digitally signed, so the entity can reuse the token for further requests without reauthenticating. This gives end users the benefit of single sign-on (SSO) so that an application that uses numerous Web services has to provide authentication only once, rather than once for each interaction. The following is a sample SAML-compliant request sent from a relying party (“Umar”) requesting password authentication by the issuing party (“nge.com”):

```
<samlp: Request ...>
  <samlp: AttributeQuery>
    <saml: Subject>
      <saml: NameIdentifier
        SecurityDomain="nge.com"
        Name="umar"/>
    </ saml: Subject>
    <saml: AttributeDesignator
      AttributeName="Employee_ID"
      AttributeNamespace="nge.com">
    </ saml: AttributeDesignator>
  </ samlp: AttributeQuery>
</ samlp: Request>
```

A great deal of information about SAML is becoming available at the time of this writing. The following principal references are suggested for additional information: [SAML](#)

[specification documents \(www.oasis-open.org\)](http://www.oasis-open.org) and [“Security Assertion Markup Language \(xml.coverpages.com/saml.html\)](http://xml.coverpages.com/saml.html).

12.10.4 Back-end System and Transaction Security Through SET

Information must be protected at the back-end sites where it exists in large databases. Access control (allowing authorized users to access needed data) protects data at various sites. Most database managers have security features that allow only authorized users to access needed data. In some cases, data is encrypted and stored for additional security. The topic of back-end system protection is well discussed under the general heading of “host security” [Oppliger 2000] and typically includes discussion of operating system and database security. For our purpose, it is important to note that back-end systems should be typically behind DMZ walls.

Most new issues related to e-business/e-commerce protection are related to secure electronic payments. **SET (Secure Electronic Transaction)** was developed jointly by Visa, MasterCard, IBM, and other technology providers for secure credit card processing. SET is used to protect the transfer of bankcard payment information over open networks like the Internet. This is an application-layer security protocol that is used primarily for credit card processing. Unlike SSL that encrypts all communications between a client and server using TCP/IP, SET is highly specific to credit card processing and contains logic that is based on the dance (“choreography”) between four players: consumer, merchant, merchant bank, and consumer bank (Figure 12-19). Basically, a consumer places an order with the merchant by giving her credit card number that also includes the consumer bank (CB) information. The payment information is sent to the merchant bank (MB), which checks with CB for credit authorization (i.e., does this payment go over the consumer credit limit?). If the CB accepts the credit, then the merchant completes the order. Although the actual SET processing is quite intricate, from an end-user’s privacy point of view, the purchase transaction is separated into two parts by SET:

- Purchase information that is sent to the merchant
- Credit card information that is only handled for credit card verification by the CB. This information is encrypted so that the merchant does not see and cannot store the credit card information.

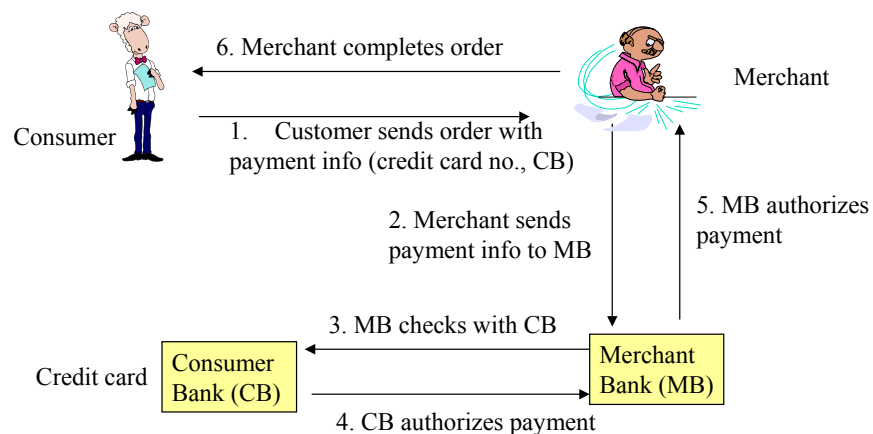


Figure 12-19: SET Processing

The consumer’s credit card information is thus kept private even from the merchant by SET – the merchant only sees what goods he is selling, how much money he will be paid, and if the credit was authorized. This level of processing cannot be done by SSL; i.e., SSL cannot be

used to check a credit card for validity, to check for credit authorization, or to process the payment transaction. An organization called SETCo manages the SET specification and promotes/supports the use of SET Secure Electronic Transaction on the Internet. See their site (www.setco.org) for additional information about SET.

While SET is quite popular, other payment systems such as cybercash (www.cybercash.com) and DigiCash (www.digicash.com) are also available.

12.11 Short Examples and Case Studies

12.11.1 Wireless in Government Services

Government agencies see great potential in wireless technology, for police services, satellite offices, temporarily relocated employees, and training through virtual classrooms. While many of the government-issued handheld devices are still used primarily to send and receive email, an increasing number are being loaded with job-specific software – as is the case with OnPatrol in Canada for police patrols. The US military is using BlackBerry handhelds for wireless management of inventory, and to help track medical records of armed-forces personnel. Sky marshalls employed at airports since 9/11 are now using the technology to communicate both with one another and with other law-enforcement officers. And, there is a widespread use of handheld devices among members of the US and Canadian federal legislatures.

Schools in remote communities have been interfacing satellite communications with wireless local-area networks (WLANs) to provide local high-speed Internet service. This opens the door to a world of educational opportunity. WLANs can enable Internet-based communication in places where wires either do not exist or cannot be installed.

Several police force offices across Canada are using a wireless solution, called OnPatrol, that involves software developed specifically for police agencies. The software is loaded onto Research in Motion (RIM) Wireless Handheld™ devices. This solution enables foot-patrol officers to securely query police databases, as well as to send and receive messages. Officers do not have to stop and write out details of a complaint because all the information is contained in their handheld. OnPatrol is being built by xwave, a services provider that has worked with public sector organizations.

Several wireless applications are also being developed for Environment Canada (EC). The department has developed a wireless network to allow visiting EC executives from across Canada to connect easily with their regional offices. EC will also use the wireless network for presentations and training.

Security is of critical importance to government organizations using wireless technology. For example, the Canadian police officers are keen to try the handhelds but recognize the obvious security concerns associated with such small, portable devices. The EC wireless network has been developed with security as a high priority. Different measures are being used to secure these wireless networks. The idea is end-to-end encryption: from the moment the message leaves one device to the moment it is received by another. The messages are encrypted on transit and remain so while stored on the devices.

For example, a VPN is being used for securing the EC wireless access to add an extra layer of encryption, which makes the network much more difficult to break into. In addition, the

OnPatrol has been developed with close attention to security. Along with incorporating 128-bit encryption, OnPatrol makes use of a log-in screen, through which users sign on with a password. As well, the device includes both application-lock-out and remote-device-erasure capabilities: application lock-out essentially shuts the device down if the wrong person repeatedly and incorrectly tries to log onto it; remote-device-erasure allows a user to disable a lost or stolen device by erasing the information contained on it.

For wireless LANs, a mixture of techniques are used. Specifically, the Wired Equivalent Privacy (WEP) encryption is turned on, the Media Access Control (MAC) address transmission is turned off, and the Service Set ID (SSID) is changed. This selective enablement/disablement makes the network harder to break into. In addition, the power-transmission levels are lowered to suit a smaller space and avoid undue eavesdropping. It is also a good idea to make sure that the vendors are FIPS-compliant (FIPS- 140-1 certification is granted to vendors that meet the security standards established by the US National Institute of Standards and Technology [NIST]).

Source – http://www.xwave.com/ebrochures/wireless_security_cs_frames.htm

12.11.2 Wireless Security in the Health Sector

Wireless communication is at the forefront of technologies being considered by the healthcare sector. There are a number of reasons why wireless communication holds such an appeal for medical practitioners and institutions, but security concerns are the major deterrents.

A major benefit of wireless communications is in the area of doctor/patient relationships. Medical practitioners must keep a large amount of collected data on all their patients. This is generally done by using a database or a paper-based filing system. A paper-based system has several obvious flaws (patient medical records can be easily lost or damaged). Even when stored on electronic databases, the doctors often have the patient's records printed to hard copy before seeing the patient. In addition, manual updates are normally performed by an office assistant based on written notes provided by the doctor – an error-prone process. By providing a medical practitioner with a Personal Data Assistant (PDA) connected to a wireless network, these problems can be readily overcome. All patient records can be updated in real-time by the doctor without the risk of them being lost or damaged.

Wireless communications can also be of great value to doctor/institution relationships. It is not easy for doctors to keep track of medication prescriptions and billing information. For example, hospital residents commonly carry around scraps of paper containing such information for entire shifts. This can often lead to the loss or damage of such information – an outcome that is unsatisfactory and potentially financially damaging to the institution. Wireless billing improves the accurate maintenance of financial records of this type and is also considerably faster and more efficient.

But wireless networks are vulnerable to attacks as discussed in this chapter. For example, a hacker parked in the street outside an institution with no more than a laptop and a \$200 wireless network card could read all of the information being passed between the doctor's PDA and the medical record server, without even leaving the vehicle. Letting a hacker access this information is a violation of doctor/patient privilege, and failing to adequately secure the information is a violation of the US Health Insurance Portability and Accountability Act (HIPAA 1996). A malicious hacker could also potentially cause financial damage to the institution by misusing the information. All are obviously big problems with serious consequences.

A wireless network can be secured in a variety of ways discussed in this chapter. For example, SSL can be used to encrypt commercial transactions over the Internet. SSL is the ideal solution to server-side security problems since most web/application servers like IBM's WebSphere support the use of SSL for communications. The process of setting up server-side SSL is normally no more difficult than generating a new key-pair and directing the web/application server to use it for encrypting the communications. Unfortunately, SSL support is not common on wireless client devices.

Without SSL support on the client side, SSL cannot be supported across the communication channel. Even those clients that do have SSL support often have widely varying implementations that must all be supported in different ways. This can become a major issue in networks with many different types of client devices, as it leads to longer rollouts and higher maintenance costs. The only real solution to this problem is to use a Java-based SSL implementation which provides SSL in a completely platform-independent way. The problem is that these Java implementations typically are too resource-hungry for the limited resources of the average wireless client.

What is really needed is a Java-based SSL solution especially designed for the limited hardware requirements of the average PDA. Companies such as Wedgetail Communications provide a number of small-footprint security implementations designed especially for wireless network devices. Of particular importance to this problem is the JCSI® Micro Edition (Java Cryptography and Security Implementation®, Micro Edition) by Wedgetail Communications. JCSI Micro Edition provides an SSL/TLS library that supports RSA encryption and is also compatible with all Java™ 2 Micro Edition (J2ME™) implementations. This means that client applications only need to be written once and can run over a wide variety of wireless clients in the network.

Source: http://www.wedgetail.com/datasheets/ehealth_case_study_us.pdf

12.11.3 Wireless LANs at Texas A&M University

It is well known that many universities are going wireless. There are several reasons for this. Wireless access is a natural fit for the highly mobile university population. In addition, wireless access is a strong recruitment tool at universities to attract top-notch students, faculty and prestigious conferences. However, the rate of wireless adoption on campuses raises several security issues, because universities are populated with bright and highly energetic students who like to probe and investigate. In addition, university networks tend to be more open than corporate environments. Intrusions into proprietary or exclusive research data, as well as into course grades and student information, need to be detected and avoided.

Texas A&M has implemented a large wireless network that is protected by a wireless VPN. The university felt comfortable with this choice because it already had experience with a wired VPN. Wireless users are authenticated through a RADIUS server – all users of a laptop or handheld device in an area with a wireless access point (AP) get routed to a VPN server and RADIUS authentication. The school has adopted, like many other campuses, the 802.11b standard for its wireless transmission. The campus network includes a wired network with Ethernet ports and wireless access points in dorm rooms, campus libraries, and conference rooms in the Bush Presidential Center. The network administrators create short-term users in the school's RADIUS authentication server database, giving guests access only while they are visiting.

Texas A&M took advantage of its student population, especially the telecommunications engineering undergraduate students, to design and test the system. After examining several

vendors, Texas A&M decided to use Cisco and Enterasys access points because of their flexibility with multiple cards. The university has no plans to entirely go wireless – the wired network still will be used to interconnect buildings.

Source: A. Saita, “The Wild Wireless West – Texas A&M brings law and order with WLAN,” *Information Security Magazine* (January 2002).

12.12 Analysis and Design of Wireless Security

12.12.1 Analysis of Security Tradeoffs and Design Guidelines

As can be seen from the discussion above, wireless security can be supported at various different layers (wireless network, TCP/IP, middleware, application) by using a wide range of technologies (see Figure 12-20). We have also seen that the solution approach at each layer employs the various cryptographic and other security technologies to support the privacy, integrity, authorization, authentication, accounting, and availability (PIA4) requirements. For example, IPsec, SSL, and SET use the same type of cryptographic technologies (private-public key encryption, digital certificates, hashes, etc) to support PIA4, albeit at different layers of the system.

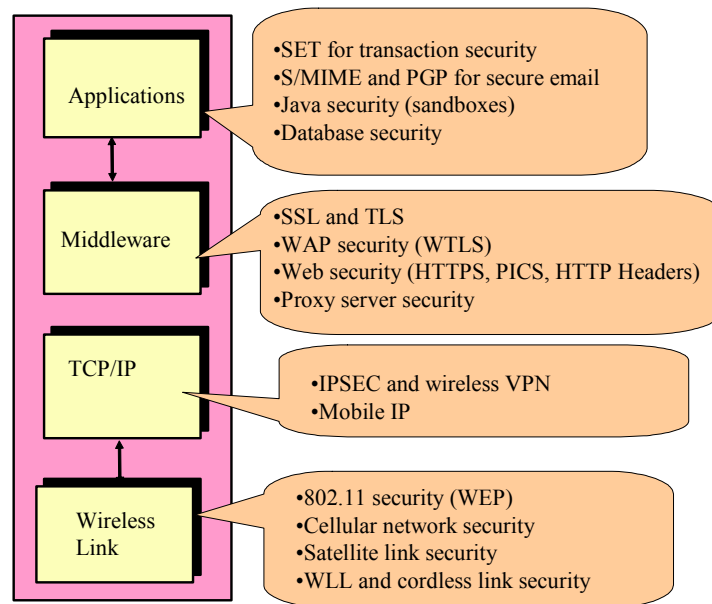


Figure 12-20: Security Technologies at Different Layers – A Starter Checklist

Security is needed at these different levels since security at each level fulfills different requirements. Let us briefly review the strengths and weaknesses of security at various levels and suggest some guidelines (see Table 12-2):

1. Wireless link security protects communications at the physical wireless link (layer 1 and 2) levels. This type of security is at the heart of wireless networks (WLANs, WLLs, cellular networks, satellites) and is especially important in wireless because of eavesdropping and other threats we have discussed previously. Although this security does not give complete

end-to-end security for a network (all links of a network need to be secure), it attempts to protect at least the segments of wireless links that need to be secured. Some guidelines are:

- Turn on security at wireless links to avoid eavesdropping even if it is deficient. For example, use WEP because it does provide some security – and make up for the shortcomings of WEP security by also providing higher layers of security (e.g., SSL).
- Make sure that all access points are themselves monitored and controlled so that no one sets up rogue access points.
- Treat wireless networks as untrusted networks. Thus put the internal WLANs outside the firewall so that they are treated as outsiders. In addition, minimize placing critical applications and databases on wireless networks – instead move them to wired networks behind firewalls.
- Make sure that the passwords on wireless networks are different than those used on the wired networks. Hackers usually capture passwords from wireless networks and then use them to gain access over wired networks.

2. TCP/IP network security encrypts TCP/IP packets. This can be done by using VPNs employing IPSec. VPNs for wireless networks provide many benefits, because all traffic spanning many links can be secured. Many “firewalls” and “gateways” are also erected to regulate the IP traffic and operate at this level. VPNs are commonly used to overcome wireless link security weaknesses. However, there are some drawbacks of VPNs:

- Roaming between VPNs is not completely transparent.
- VPNs have to overcome firewall barriers.
- VPNs do not support multicasting.
- VPNs do introduce excessive overhead. Everything is encrypted, even if you are surfing the web.

3. Middleware-level security can secure point-to-point communication between specific clients and servers. For example, SSL (Secure Socket Layer) secures most Web clients and servers. Similarly, WTLS secures communications between WAP clients and WAP servers (WAP gateway). This is important especially if lower-level security was not employed or was weak due to wireless. Specific issues with middleware-level security are:

- Middleware security only applies to the applications that operate on top of the said middleware. For example, CORBA security can only protect CORBA applications and WAP security can only secure WAP applications.
- Gaps may exist between different types of middleware security. For example, WTLS protects between the WAP device and the WAP gateway. A gap exists when the WAP gateway has to translate to the final website using SSL.

4. Application-level security is provided by database managers, Java security, SET (Secure Electronic Transactions), PGP, S-MIME, and several other application-specific security packages. A variety of security approaches exist at the application level, in which case authorization controls are used within applications to regulate access to specific data, and cryptographic infrastructures are built to strongly authenticate users and provide confidentiality. In particular, applications themselves provide access control and strong user authentication. Specific considerations for this level of security are:

- Different applications use different security features. You need to turn on security for each application to make it secure. For example, SET makes transactions secure, while S/MIME and PGP make email secure.
- Very strong application-level security should be used for applications and databases that are accessed over wireless networks.
- Application-level firewalls that filter application traffic provide a much finer level of security as compared to IP-level firewalls.

Security must be considered at all levels. However, very strong security at every level (e.g., encryption at all levels) can add significant overhead. Tradeoffs are essential. But you have to decide what level to emphasize. Specifically, application security protects application data (e.g., database security mechanisms allow the data to be stored on the hosts in a protected manner) and system resources (e.g., Java Security), while SSL, IPSec, and wireless link security protects data while being transferred on the network.

Table 12-2: Security Levels

Security Level	Example of Security	Why Needed?	Why Not Enough?
Application-level security	SET, PGP, S-MIME	Provide security specific to an application	Only protection of application-specific data
Client/Server Security	SSL and WTLS Security	Assures secure communication over an unsecure link	Only middleware-level security
IP Level	IPSec, VPN	Protects the IP path	Does not protect databases
Network Link Level	Wireless LAN Security, 3G and Satellite Security	Deters breaking in at physical link level	Protects only one link. Does not cover other links in a large network

12.12.2 Simple Design Procedure for Wireless Security

Figure 12-21 shows a general security design procedure that has been customized for wireless security. This procedure consists of several steps, using the cryptographic and other techniques to build solutions at different layers (wireless network links, TCP/IP-level traffic, middleware, applications), and then putting the pieces together. The focus is on developing circumventions based on the available solutions in the marketplace.

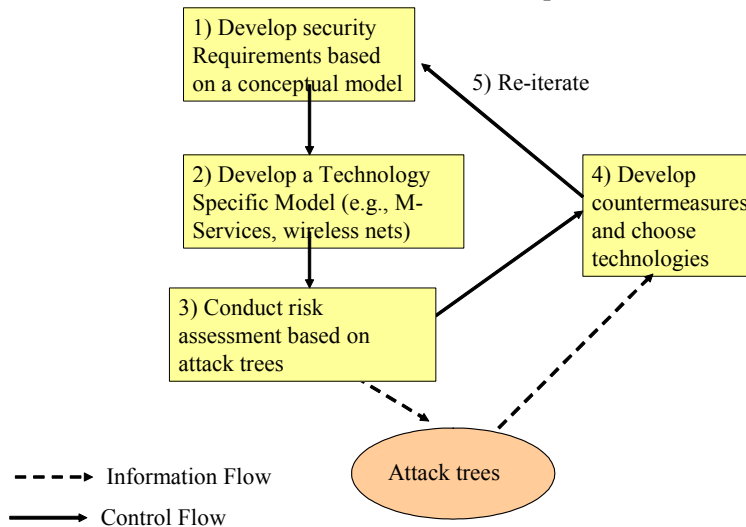


Figure 12-21: Security Design Procedure

12.12.2.1 Step 1: Determine Security Requirements

Develop a conceptual model of the business problem and determine the type of security that is needed by the customer in terms of PIA4 (privacy, integrity, authentication, authorization,

accounting, availability). Specifically, the security requirements should specify PIA4 for the following assets to be secured:

- Information submitted by the users such as IDs, passwords, credit cards, emails, etc.
- Data stores (e.g., files and databases) where the information is stored.
- Information exchanged between programs and databases such as electronic fund transfer.
- IT infrastructure such as computers, middleware, network devices (e.g., access points), etc.

These requirements drive the wireless security design and can be stated in a variety of ways (e.g., a business model of the system that can be used to infer security requirements). Although everyone wants all attributes of PIA4, the level of security is determined by knowing the risk involved. For example, privacy and integrity requirements of a chat group about extra-terrestrial life are different than those of a military command and control system. In addition, the value and volume of transactions can influence the security requirements. For example, online purchasing of books has lower security requirements than does electronic fund transfer between banks.

12.12.2.2 Step 2: Develop a System Architecture

Develop a system architecture that shows the network as well as the applications that will address the business problem. For example, for a wireless office, an architecture of the office is developed with networks, access points, etc. Similarly, for online purchasing, an architecture that shows the users, the cellular network, and the purchasing system is constructed. The architecture should show the type of network (cellular, W-LAN, Wireless local loop, Bluetooth, ad hoc network, etc.). Figure 12-22 shows a sample architecture of an online purchasing system that is accessed from a wireless network. In this architecture, the wireless network can be a cellular phone or a hotspot Wi-Fi LAN connected through a wireless local loop to the online purchasing system. A wireless gateway converts the wireless communications to regular HTTP traffic. A WAP gateway is such an example. The purchasing system is also connected to an internal back end as well as to external supplier systems.

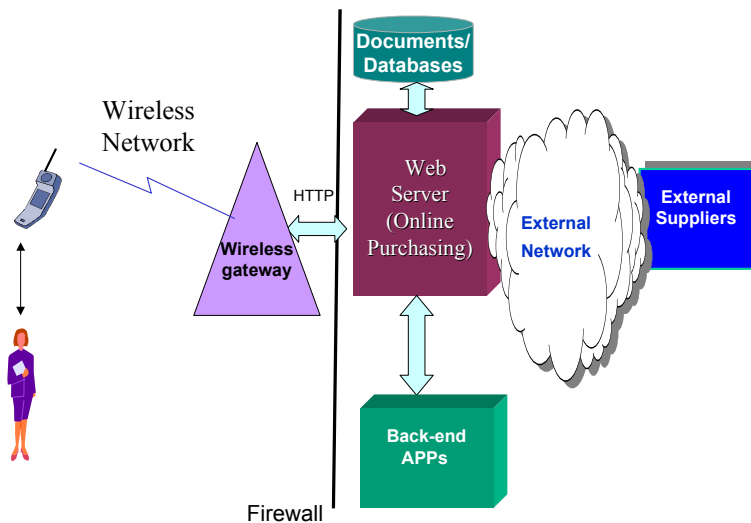


Figure 12-22: Mobile Online Purchasing System

12.12.2.3 Step 3: Conduct Risk Analysis

It is important to conduct risk/vulnerability analysis of the proposed wireless network at this stage. The main idea of detailed risk analysis is not only to identify the threats but also to develop countermeasures to combat them. For example, the key length for encryption can be chosen based on this analysis. To determine the possible attacks on the system, attack trees can be used. Let us review the design of the online purchasing system shown above, and then develop “attack trees” to understand system vulnerabilities.

An attack tree is a convenient way to explore potential attacks and thoroughly examine the “attack space” in a given situation [Schneier 1999]. An attack tree is simply a tree that is similar to a logical decision tree used to perform a systematic analysis of the attack space. It may be represented through a graph or some other means such as the extended outline mode of Microsoft Word. Attack trees are built by considering the “what,” “where,” “when,” and “how” of attacking the system. For “what,” an attacker can try to compromise system and data integrity, data confidentiality, or system availability. For “where,” an attacker could attempt to do this inside a firewall (an internal attacker), at the firewall that separates the internal system from the public Internet, or on the public Internet. We will focus on wireless attack points. For “when,” an attacker could mount the attack at any point in the life cycle of the system: during system startup, during system integration, during system operation, or after the system has exceeded its useful life and is being discarded. The “how” of an attack deals with the mechanism used to execute the attack, such as eavesdropping.

Attack trees can be built for each security concern: privacy, integrity, and availability. An example of building a privacy attack tree for confidentiality of data over the wireless network of the mobile purchasing system shown in Figure 12-22 is:

C What: Confidentiality of data in the mobile purchasing system

C1 Where: Inside the firewall (not considered)

C2 Where: Between the Web server and the customer over wireless Internet connection

C2.2 When: During system operation

C2.2.1 How: Deliberate software attack

C2.2.1.1 How: Tapping radio link of a wireless network

This tree focuses on confidentiality attacks and can be expanded. In addition to confidentiality, there may be concerns about other security properties such as denial of service and availability. The main advantage of building attack trees is that they provide a systematic way of isolating vulnerabilities due to different technologies (wireless, in our case) from a total system point of view. To be useful, attack trees are heavily pruned during construction to avoid explosion of leaves. This can be done by assigning a “possible” or “impossible” attribute to the leaves. For example, the “where” branch of the availability tree that examines attacks occurring inside the firewall can be ignored for wireless because the inside network is wired (we assume). This may or may not be a prudent decision. By pruning this branch early in the analysis and excluding any such attacks, the remainder of the analysis will not consider this option and countermeasures will not be chosen against this type of attack.

12.12.2.4 Step 4: Develop Countermeasures

Develop countermeasures from a knowledge of the possible attacks and the security technologies and approaches that can help you to survive the attacks. For wireless, there can be many countermeasures for a given attack, so a countermeasure is chosen based on cost, functionality, performance measures, ease of use, and effectiveness in dealing with the corresponding attack. The countermeasures are highly dependent on the type of wireless network chosen; e.g., a cellular network needs a different type of countermeasures than an 802.11 wireless LAN. For comparisons among countermeasures, it is desirable to choose the countermeasures that increase the costs and capabilities needed by the adversary and also increase the risk of being detected. Comparative analysis should lead to the most appropriate countermeasures to be implemented for risk mitigation.

The main idea is to select the appropriate security technologies and a security architecture based on the countermeasures. The security tradeoffs and design guidelines discussed previously (Section 12.12.1) are of direct relevance here. Let us reconsider the wireless online purchasing system shown in Figure 12-22. If the wireless network is a cellular network and the online purchasing system supports WAP, then we can make the following choices:

- The customer uses a pin to access the telephone.
- A WTLS session with encryption is established between the microbrowser and the WAP gateway.
- Communication between the gateway and the Web server is encrypted by using SSL.

Additional security can be added for accessing back-end and external applications.

It is also possible to provide a higher level of security to compensate for lower-level security weaknesses. The following figure, repeated from a previous section, shows different levels of security technologies. Basically if lower-level systems are weak, then higher-level services such as databases and applications need to be protected. For example, if a database can be only accessed through a wired local network inside a corporate firewall, then it may not need heavy encryption and authentication. However, if the same database is accessed over a wireless network, then it must have heavy-duty encryption and authentication support.

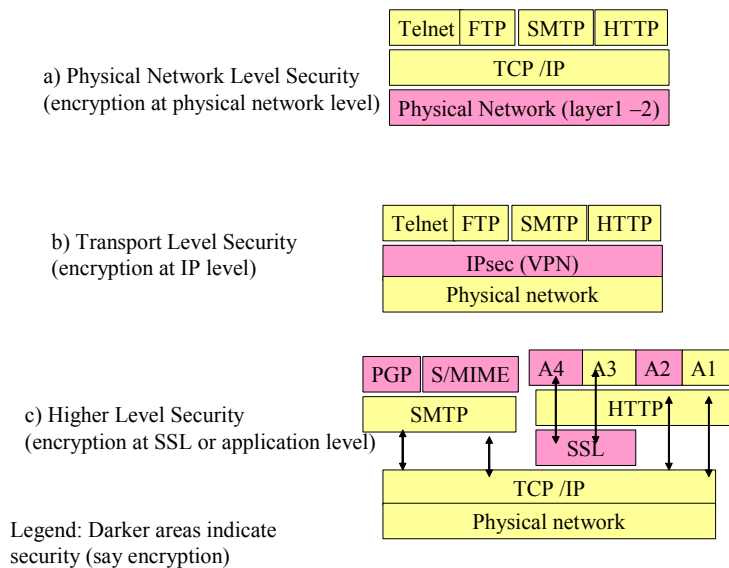


Figure 12-23: Approaches to Wireless Security

12.13 Summary and Conclusions

Security concerns are growing more serious due to the widespread use of wireless networks. Although wireless networks face the same type of security issues (e.g., privacy, integrity, authentication) as the wired networks, the main difference is that wireless network traffic is transmitted over the air and thus is easier to tap into than wired networks. This chapter has provided enough details about various wireless security issues so that a sound solution can be developed. The design procedure includes all levels of security: enterprise applications and corporate databases, middleware security (Web and WAP security), and network-level security (VPNs and wireless link security).

12.14 Review Questions and Exercises

- 1) What are the issues that are unique to wireless communications?
- 2) Create a table that shows different types of wireless security issues at various levels and the key technologies that are used to address these issues
- 3) What are the main problems in Wi-Fi security? Why is it a major concern and what specifically can be done to address these problems?
- 4) Why should anybody care about wireless PAN security? Are there some unique issues and solution approaches in this area?
- 5) What are the unique issues in cellular networks? What exactly do 3G networks offer in this area?
- 6) What are the main concerns in satellite security and what are the key practical approaches?
- 7) Is wireless local loop security a major concern for the everyday consumer? Who needs to worry about these issues?
- 8) How can FSO and UWB provide better security? What type of wireless networks can benefit from these emerging networks?
- 9) What are the most serious problems in MANET security? What appears to be the most promising approach?
- 10) Where does Mobile IP fit in the wireless security landscape? Be specific.
- 11) What are the wireless middleware issues in WAP and i-mode? Why do they need to be discussed separately?

12.15 References

- Aron, M. "Better Security Needed for B2B." *Australasian Business Intelligence* (Nov. 4, 2002).
- Arbaugh, W., et al. *Your 802.11 Wireless Network has No Clothes*. Department of Computer Science, University of Maryland, College Park, Maryland 20742 (March 30, 2001).

- Borisov, N., Goldberg, I., and Wagner, D. "Intercepting Mobile Communications – The Insecurity of 802.11." Available at: <http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html>
- Brewin, R. "FAA Views Aircraft Satellite Security Systems as 'Complex Undertaking.'" *Computerworld* (Oct. 20, 2001).
- Buchegger, S. and Le Boudec, J. "Nodes Bearing Grudges: Towards Routing Security, Fairness, and Robustness in Mobile Ad Hoc Networks." *Proc. of 10th European Workshop on Parallel, Distributed and Network-based Processing*. 2002.
- Byers, S. and Kormann, D. "802.11b Access Point Mapping." *Communications of ACM* (May 2003).
- Castro, M. and Liskov, B. "Practical Byzantine Fault Tolerance." *Proceedings of the 3rd Symposium on Operating System Design and Implementation*. New Orleans, February 1999.
- Cam-Winget, N., et al. "Security Flaws in 802.11 Data Link Protocols." *Communications of ACM* (May 2003).
- Desmedt, Y. and Jajodia, J. "Redistributing Secret Shares to New Access Structures and its Applications." *Technical Report ISSE TR-97-01*. George Mason University, July 1997.
- Fratto, M. "Tutorial: Wireless Security." *Network Computing Magazine* (January 22, 2001).
- Haas, Z. and Liang, B. "Ad hoc Mobility Management using Quorum Systems," *IEEE/ACM Transactions on Networking* (1999).
- Herzberg, A. "Payments and Banking With Personal Devices," *Communications of ACM* (May 2003).
- Housley, R. and Arbough, W. "Security Problems in 802.11-based Networks." *Communications of ACM* (May 2003).
- LAN97, "LAN MAN Standards of the IEEE Computer Society. Wireless LAN medium access control (MAC) and physical layer(PHY) specification." *IEEE Standard 802.11*. 1997 Edition, 1997.
- Oppliger, R. *Security Technologies for the World Wide Web*. Artech, 2000.
- Noubir, G. "Optimizing Multicast Security over Satellite Links." *European Space Agency Project*. Work package 20 report, version 0.1, April 1998.
- Noubir, G. and Allmen, L. "Security Issues in Internet Protocols over Satellite Links." *Proc. of the IEEE VTC*. 1999.
- Perrig, A., Stankovic, J., and Wagner, D., "Security in Wireless Sensor Networks", *Comm. ACM*, June 2004, pp. 53-57.
- Ramanathan, R., and Redi, J. "A Brief Overview of Ad Hoc Networks: Challenges and Directions." *IEEE Communications Magazine* (May 2002).
- Roberts, P. "Government Report Finds Satellite Security Lax." *IDG News Service* (Oct. 04, 2002).
- Schmidt, T. and Kormann, D. "Why Wi-Fi Wants To Be Free." *Communications of ACM* (May 2003).
- Schneier, B., "Attack trees", *Dr. Dobb's Journal*, Dec.,1999.

- Schneier, B., "Applied cryptography: Protocols, algorithms and source code in C". Second edition. New York: Wiley, 1996, p. 2.
- Sheldon, T. "General Firewall White Paper." Available at: <http://secinf.net/info/nt/fw/firewall.html> (Nov. 1996).
- Slijepcevic, S., et al. "On Communication Security in Wireless Ad-Hoc Sensor Networks." IEEE Eleventh International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE-2002), Pittsburgh, Pennsylvania, USA. June 10-12, 2002.
- Stallings, W. *Network Security Essentials*. Prentice Hall, 2000.
- Stein, L. *Web Security: A step-by-step Reference Guide*. Addison Wesley, 1998.
- Sun, J. "Mobile and Ad Hoc Networking: An Essential Technology for Pervasive Computing." *Info-tech and Info-net International Conference*, Beijing, 2001.
- Umar, A. *Information Security and Auditing in the Digital Age*. 2nd ed. NGE Solutions, June 2004.
- Walker, J. "Unsafe at any key size: an analysis of the WEP encapsulation." *Tech. Rep. 03628E*. IEEE 802.11 committee, March 2000. Available at: <http://grouper.ieee.org/groups/802/11/Documents/DocumentHolder/0-362.zip>
- Walker, J. "Overview of 802.11 security." Available at: http://grouper.ieee.org/groups/802/15/pub/2001/Mar01/01154r0P802-15_TG3
- Wexler, J. "Satellite VPNs to Address Performance and Security Issues." *Network World Wireless in the Enterprise Newsletter* (March 17, 2003).
- Wexler, J. "Satellite: What's at Risk?" *Network World Wireless in the Enterprise Newsletter*, (Oct. 16, 2002).
- Zhao, F. and Guibas, L. *Wireless Sensor Networks : An Information Processing Approach*. Morgan Kaufmann, May 2004.
- Zhou, L., and Haas, Z. "Securing Ad Hoc Networks." *IEEE Network* (1999). Available at: <http://citeseer.nj.nec.com/zhou99securing.html>
- Zhou, L. and Haas, Z. "Securing Ad Hoc Networks." *IEEE Networks Special Issue on Network Security*, November/December, 1999.