Securing Unlicensed WLAN Data Communications

There is no lack of evidence that present 802.11 WLAN networks use unsecure or minimally secure links. Even secured links can be easily breached. A system design will not bea success unless it supports the needed security

By Bob Cromwell

There is a growing realization that wireless networking has some serious security problems. The non-RF community is slowly realizing, to their surprise, that RF signals sometimes propagate through structures and out of windows.

Information security specialists have demonstrated that some protocols intended to secure wireless traffic also have serious security problems.

RF engineers need to understand the major issues of wireless network security. A system design will not be a success unless it supports the needed security. This must include data integrity and confidentiality, plus device authentication.

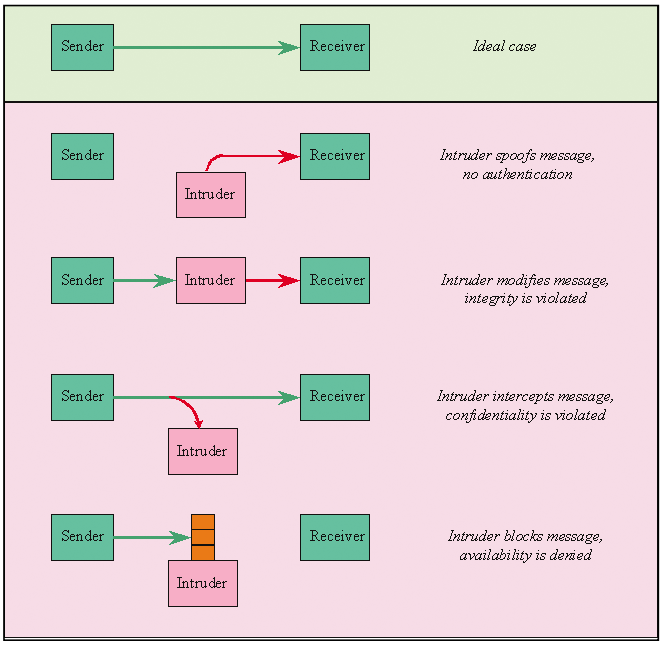

Figure 1. 4-threat COMSEC model

The bad news is that current wireless security mechanisms are deeply flawed. The good news is that there is technology available today to secure wireless local area networks (WLANs). This article will make sense of the alphabet soup of wireless LAN modes, and it will explain security mechanisms and security protocols; which ones work, and which do not. It will then show how strong cryptographic communications security can be provided in the operating system of a networked computer, in the network interface device itself, or in both. This can provide host authentication, data confidentiality, and key management through X.509 certificates provided by a trusted public-key infrastructure (PKI).

COMSEC and Networking Models

A common communications security model considers four possible threats to a flow of information from a sender to a receiver, as shown in figure 1. In an ideal situation, both parties could have a high confidence in these four properties:

- Authentication, meaning that the sender is provably the source of the information.
- Data integrity, meaning that the message was not modified along the way, either maliciously or accidentally.
- Data confidentiality, meaning that the receiver is the only party able to receive the information.
- Availability of the communications channel, meaning that sender could always send a message to receiver.

Cryptographic techniques allow the detection of attempted spoofing and modification. Other cryptographic techniques can provide confidentiality, although we cannot detect attempts, successful or not, at defeating them. Unfortunately, there is no cryptographic defense for availability, so we cannot discuss probabilities of risks or strengths of defenses in any formal way.

The open systems interconnection (OSI) networking model (see figure 2), shows the placement of various cryptographic defenses and the traditional division between hardware and software. It will be shown that modern designs are pushing the boundary upwards — more security capabilities are being built into the hardware.

Wireless Networking Technologies

While WLAN is today's hot topic, the specific areas of interest are the wireless network links, in general. The techniques described in this article apply to any wireless network links, LANs or point-to-point.

Table 1 shows common WLAN technologies: IEEE designation, frequency, and bit rate. The IEEE uses "802.11" to refer to wireless networking in general, with a code letter appended for each IEEE task group. Some (e.g., 802.11a) refer

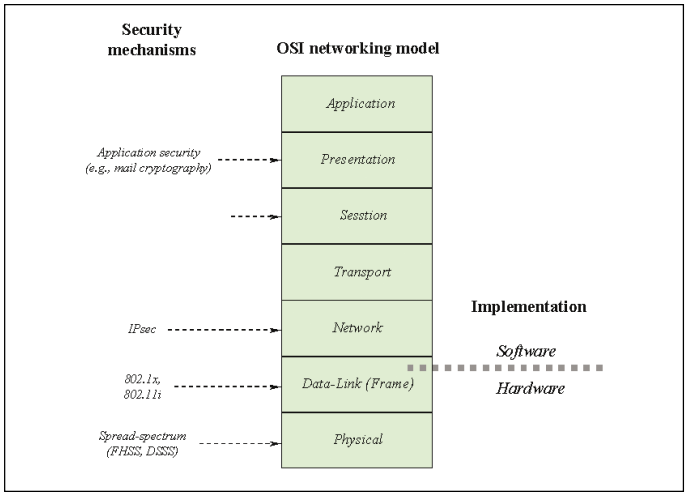

Figure 2. The open systems interconnection (OSI) networking model

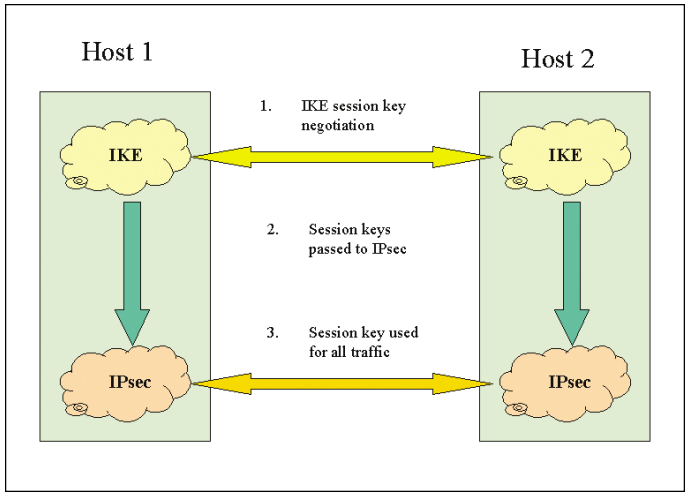

Figure 3. IKE key exchange model

to a specific frequency-modulation combination. Other task groups work on techniques applicable to any frequency-modulation mode, such as 802.11e (quality of service — QoS) or 802.11i (security). All modes listed use direct-sequence spread spectrum (DSSS) for channel sharing, although not in a way that increases security.

802.11 Alphabet Soup

Mode 802.11h is 802.11a modified to include "listen before transmit" capability, called dynamic frequency selection (DFS), and transmit power control (TPC) for minimum power use. Both of these modifications are required to use the 5 GHz band in the European Union, and they could appear in later versions of 802.11b and 802.11g. IEEE 802.11g devices are required to be backwards compatible, supporting 802.11b. IEEE 802.11b, called "Wi-Fi", is today's most-used mode. Note that the bit rates are simply that — the expected throughput is less than half the bit rate under the best of conditions, and the devices will reduce the bit rate further to reduce the probability of single-bit error with decreasing signal strength.

Known WLAN Risks

All observations indicate that the majority of organizations do not enable even the most trivial defenses. So-called "war-driving," or even "war-flying," shows that vast numbers of unintentionally public networks are found in urban centers and industrial parks. This has led the U.S. Department of Defense (DoD) and other organizations to consider WLANs as tempting but dangerous technologies[2]. Risks include:

- Compromise of information (theft or modification),
- Theft of service, and
- Liability (your WLANs as launching points for attacks).

WLAN access points must broadcast "beacon frames" in the clear. These frames include a service set identifier (SSID), also called a network name. Many organizations have not even changed the SSID from its factory default. Even if changed, the SSID is easily captured by a receiver. Attacks are quite possible, and studies have shown that they are not necessarily detectable[1].

Another risk is the rogue access point. Consider security in the other direction for a moment. Are you willing to blindly trust an access point, or do you want some assurance of its identity before transmitting your sensitive information?

Attempts to Secure WLAN Links — OSI Layers 1 and 2

Physical layer security could be done with device-specific spread-spectrum techniques — basically CDMA via DSSS. However, security would require much longer code sequences than for mere spectrum sharing, and the hardware cost and key management complexity make this unlikely for consumer devices.

IEEE Designation	Frequency	Bit rate
802.11a	5 GHz	54 Mbps
802.11b	2.4 GHz	11 Mbps
802.11g	2.4 GHz	54 Mbps
802.11h	5 GHz	54 Mbps

Table 1. Common WLAN technologies

Some organizations have attempted to secure WLAN access by filtering frames at the access point based on their 48-bit hardware addresses. Only the frames from "registered" devices are forwarded. This would not scale well — can you really maintain an accurate corporate-wide database of all WLAN hardware addresses? But even if it were practical, this approach still provides little security. The right combination of operating system and WLAN device allows an attacker to modify the hardware address and make its frame appear legitimate.

The original 802.11b security work was promising, but flawed. The wired equivalent privacy (WEP) protocol has been known to be quite insecure since 2001. It was originally limited to 40-bit keys due to U.S. export control laws, but even with longer keys it has been called "unsafe at any key size." The cryptographic analysis is daunting, but the problem summarizes to using secure algorithms in insecure ways. The RC4 encryption algorithm is used poorly, a bad method is used to expand the key into a keystream, and there is a flaw in the key scheduling algorithm.

A few hundred megabytes of traffic is adequate to build a table from which the WEP key is extracted. On today's networks, that may not take

long. A tool like *AirSnort* can automate WEP key recovery, allowing an intruder to decrypt the poorly-protected data, or masquerade as a legitimate host.

Note that the latest firmware releases from most vendors have eliminated these risks, but are you really certain that you are using the most recent firmware? The computer industry generally fails to apply needed software patches even for well-publicized problems, so an assumption that the hardware upgrades have been scrupulously followed is a risky one.

Hope for the Future — 802.1x Into 802.11i

The initial work used to secure WLAN traffic is IEEE task group 802.1x (and not 802.11x as sometimes reported).

The 802.1x spec provides "per-port user authentication," which requires user authentication before granting network access. But 802.1x was designed for wired networks with a fixed physical topology and very limited opportunity to inject or capture signals.

An additional patch is temporal key integrity protocol (TKIP). This starts with a 128-bit "temporal key" shared among clients and the access point, and requires periodic re-keying.

Some vendors have implemented TKIP-like solutions called simple secure networks (SSNs). These also use time-varying keys.

Yet another piece of the puzzle is Wi-Fi protected access (WPA). This requires a user password to initiate encryption, and includes TKIP. Released in October 2002, it has been shown to be susceptible to a number of denial-of-service attacks. Many analysts recommend skipping WPA and stand-alone TKIP or SSN, and waiting for the integrated solution of 802.11i, which could be used with 802.11b or any other physical protocol.

The 802.11i subset starts with 802.1x, but adds two crucial improvements. First, it mandates the use of a much stronger encryption algorithm, the advanced encryption standard (AES), with a 256-bit key (see sidebar). Initial releases of the 802.11i standard are expected to allow use of the weaker TKIP algorithm as a temporary fix until AES chipsets become more common. Some AES chipsets are already available, meaning that new designs can skip the stop-gap TKIP and move directly to AES support.

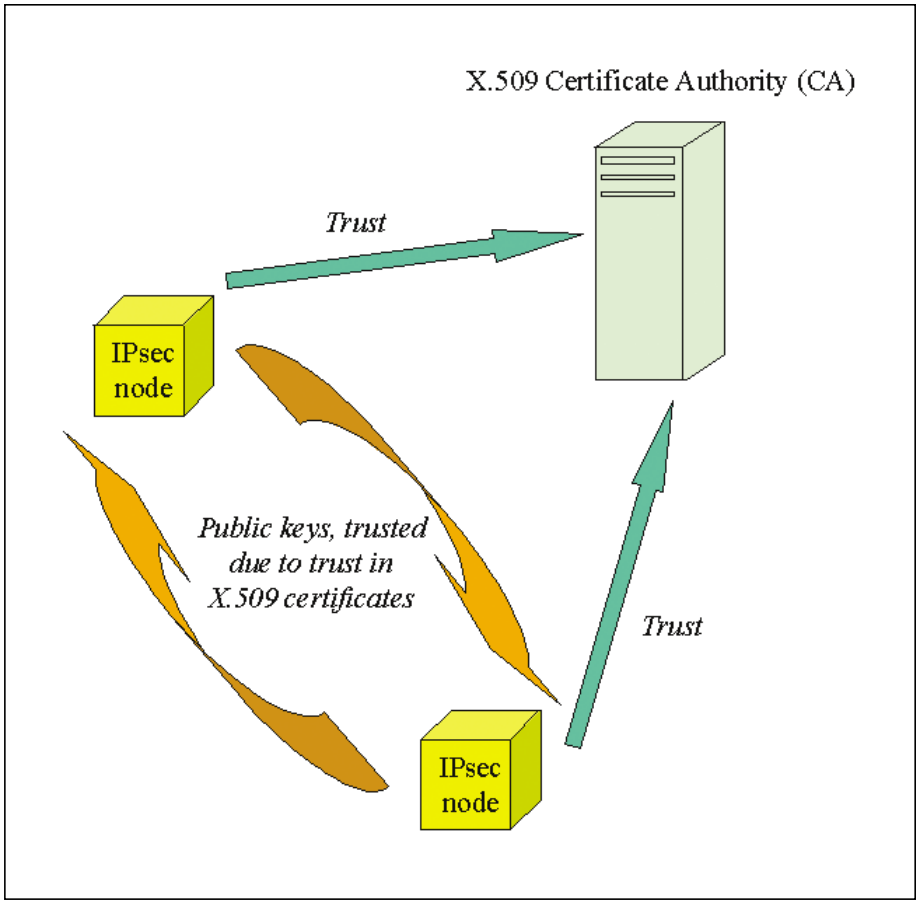

Figure 4. Key exchange using X.509 certificates

Second, 802.11i adds a key distribution framework. This includes extensible authentication protocol (EAP), using a central authentication server (such as a RADIUS server), and transport layer security (TLS). These provide "mutual authentication," authenticating the client to the server, and the access point to the client.

OSI Layer 3 COMSEC With IPsec

A full security solution requires cryptographic assurance of authentication, integrity, and confidentiality on every packet. The network layer (OSI Layer 3) protocol in the transmission control protocol/Internet protocol (TCP/IP) stack is the Internet protocol, or IP. The Internet backbone switched over to IP 20 years ago; it is very mature technology as networking protocols go. An extension that has been around for some time is Internet protocol security, or IPsec. IPsec provides this full security.

IPsec introduces its own set of terms and acronyms. Two hosts wanting to communicate securely describe their relationship as a security association (SA). A host may be communicating securely with several other hosts, so a security parameter index (SPI) is an index, or code, indicating which SA applies to a given data packet. The SA specifies a combination of cryptographic authentication, integrity checking, and confidentiality.

These operations require cryptographic keys, and they must be exchanged in a highly insecure environment.

Internet key exchange (IKE) is a high-level protocol that allows two devices to exchange information in an insecure environment, independently deriving the same shared secret key while preventing any listener from deriving that same key. This is ultimately based on the Diffie-Hellman key exchange, a well-known and

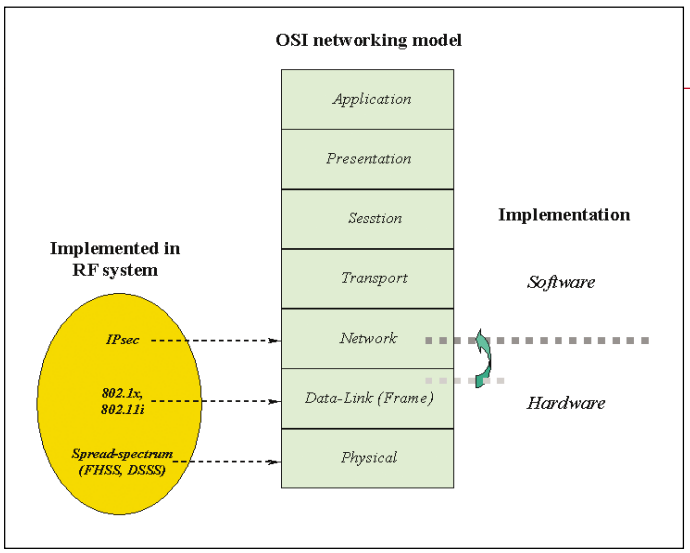

Figure 5. The future of the hardware/software relationship

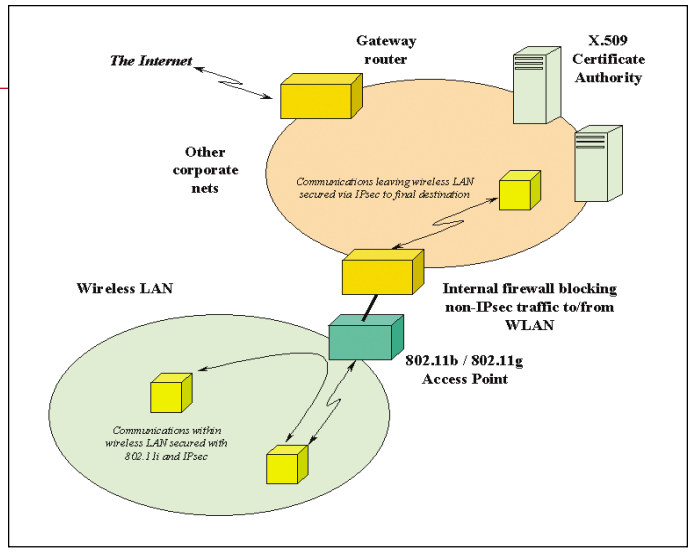

Figure 6. A shapshot of next generation WLAN security

trusted algorithm.

If a host wants to exchange a packet with another host, it first examines its set of SAs to see if they share an SA. If so, a quick flurry of IKE traffic generates session keys, which are handed to the IP modules of the operating systems. The keys are only used for a limited amount of time or traffic before one end initiates a new IKE key exchange (see figure 3).

The public keys used for the asymmetric cryptography within IKE must somehow be trusted. In other words, any IPsec node must have some valid reason to believe that it truly has the other host's public key, and not the key of some would-be interloper. A number of solutions have been implemented.

The first was to manually configure each SA on each IPsec node. This obviously does not scale to large organizations. The second solution was to distribute key information through some network service like the domain name system (DNS) server. But this just displaces inappropriate trust from one system to another.

A third solution, popular among many personal IPsec enthusiasts, has been termed "opportunistic encryption." The emphasis here is on encrypting for confidentiality, regardless of the complete lack of trust in the other node's identity. This has been described as providing a mechanism hereby two complete strangers can whisper secrets to each other in a darkened room. An interesting application in some ways, but definitely not what is needed for industrial or government use.

The fourth solution is to utilize chains of trust back to a known public-key infrastructure (PKI) server. A PKI server can issue X.509 certificates.These are cryptographic messages that can prove the identity of a host, but cannot be practically spoofed. Your trust in the identity of a host offering X.509 credentials is the combination of your trust in the cryptographic algorithms and in the PKI certificate server. Each host in an SA can offer X.509 credentials to the other, to establish identity and initiate an IKE key exchange. This provides quantifiable trust in node identity with the ease of "opportunistic encryption." The trust in the communications security is based on the trust in the keys, which in turn is traceable back to the trust in the certificate server.

IPsec Today

IPsec is available today, and has been available for some time, under a variety of operating systems: Windows 2000, MacOS X, Linux, BSD, Solaris, and other versions of Unix. Also, many routers, including Cisco, can serve as one end of an IPsec connection.

The best documentation on interoperability of various IPsec implementations is probably found on the FreeS/WAN site (**www.freeswan.org**). Although FreeS/WAN is the IPsec implementation for Linux, its interoperability document very nicely spells out hints for getting various operating systems all running IPsec together. The most general solution is either to rely on manual keying, or to use an X.509 certificate server trusted by all hosts.

A nice attribute of IPsec is that it is not a wireless-only solution. It can provide very strong communications security between any two IP nodes, whether they are adjacent on the same LAN or on opposite sides of the world. It can be used as a part of defense in depth: use 802.11i to secure the wireless links themselves, and use IPsec on all nodes regardless of their connection point.

IPsec capability is already appearing in chipsets from various manufacturers. This effectively moves the boundary between hardware and software a little higher up the OSI protocol stack. More of the networking, including the crucial security, can be moved from the operating system into the wireless device.

IPsec Tomorrow

There are changes underway with IPsec. The nature of cryptography means that encryption also provides authentication and integrity verification. Any attempt to spoof an encrypted message without the needed key will result in gibberish after the application of what would appear to be the appropriate decryption. Any single-bit change in the encrypted data (the ciphertext) will utterly corrupt the entirely of the decrypted data (the plain text).

Most IPsec users are primarily concerned with confidentiality, and since encryption also provides data integrity and host authentication, encryption alone is adequate. In IPsec terms, encapsulating security payload (ESP) processing is adequate, the additional AH (Authentication Header) is not needed. A future IPsec specification may call for ESP (encryption) only.

Future releases of IPsec will definitely require stronger cryptographic algorithms. The 56-bit data encryption standard (DES) is currently dep-

recated, and unsupported by many, if not most, IPsec implementations. The 112-bit triple-DES (3DES) standard will be the minimum acceptable security, but expect most implementations to prefer, if not require, 256-bit AES. Also, IKE will be slightly modified, with a new name of just fast keying (JFK).

Putting It All Together

What could a near-future WLAN look like when it is integrated with a corporate wired LAN? Perhaps something like figure 6.

The WLAN segment is protected with 802.11i security overlaid on the 802.11b or 802.11g physical layer. A trusted authentication server provides mutual authentication of the client and access point, and data is encrypted at OSI Layer 2.

There would be a firewall between the WLAN and the rest of the corporate intranet, blocking non-IPsec traffic. With an SA between a wireless node and the access point, IPsec could protect the traffic as it traversed the WLAN. If an SA could be negotiated between the wireless node and an internal host, IPsec could protect the data end-to-end. This could even include traveling out through the corporate firewall and across the Internet to a remote site. This could all be supported by a trusted PKI service.

The firewall could be a conventional router coupled with a WLAN access device, or it could be an integrated wireless security gateway.

One thing that recent "war-driving" tests have clearly shown is that WLAN must be secure by default, as users cannot be trusted to do anything to enable even the simplest of security. Chipsets supporting 802.11i and IPsec will help, as they move the hardware/software boundary up into the network stack.

The future for WLAN security is promising. There are many possible solutions, and with high security provided by default, the market for RF designs will increase.

RF

Article References

1. Joshua Wright, "Layer 2 Analysis of WLAN Discovery Applications for Intrusion Detection," November 8, 2002, **http//home.jwu.edu/jwright/papers/ l2-wlan-ids.pdf**.

2. Don Caterinicchia, "Military Pushes for Wireless Security," *Federal Computer Week*, November 20, 2002.

Related Links

Nikita Borisov, Ian Goldberg, and David Wagner, "Intercepting Mobile Communications: The Insecurity of 802.11," Proceedings of 7th Annual International Conference on Mobile Computing and Networking, July 16-22 2003, **www.isaac.cs.berkeley.edu/isaac/mobicom.pdf**.

William A. Arbaugh, Narendar Shankar, and Y.C. Justin Wan, "Your 802.11 Wireless Network has No Clothes," March 30, 2001, **www.cs.umd. edu/~waa/wireless.pdf**.

Jesse R. Walker,"Unsafe at any key size; an analysis of the WEP encapsulation," IEEE Document 802.11-00/362, October 2000, **http://grouper.ieee.org/groups/802/11/documents/index.html**, year 2000 document number 362.

Scott Fluhrer, Itsik Mantin, and Adi Shamir, "Weaknesses in the Key Scheduling Algorithm of RC4," Eighth Annual Workshop on Selected Areas in Cryptography, August 2001, **http://downloads.securityfocus.com/library/rc4_ksaproc.pdf**.

Adam Stubblefield, John Ioannidis, and Aviel D. Rubin "Using the Fluhrer, Mantin, and Shamir Attack to Break WEP"; AT&T Labs Technical Report TD-4ZCPZZ, *Revision 2*, August 2001; **http://www.cs.rice.edu/~stubble/wep**.

About the author

Bob Cromwell has a Ph.D. in computer engineering from Purdue University. His consulting clients include Cummins Engine Company, TASC, the Department of Defense, SITA, the American International Health Alliance, and Kicon. He also teaches and writes networking and information security courses for Learning Tree International. He holds extra-class Amateur Radio license KC9RG. He can be reached at or at **bob@cromwell-intl.com.**