



iALERT White Paper

Hacking the Invisible Network

Insecurities in 802.11x

By Michael Sutton
iDEFENSE Labs
msutton@idefense.com

July 10, 2002



iDEFENSE Inc.
14151 Newbrook Drive
Suite 100
Chantilly, VA 20151
Main: 703-961-1070
Fax: 703-961-1071
<http://www.idefense.com>

Copyright © 2002, iDEFENSE Inc.
"The Power of Intelligence" is trademarked by iDEFENSE Inc.
iDEFENSE and iALERT are Service Marks of iDEFENSE Inc.

TABLE OF CONTENTS

Executive Summary	4
WEP Insecurities	5
What is 802.11x?	5
What is WEP?	6
Issues.....	6
<i>Initialization Vector</i>	6
<i>Cyclical Redundancy Check</i>	8
Attacks	10
IEEE 802.11 Chair Response	12
Auditing WLANs.....	13
Finding WLANs ("What's the Frequency, Kenneth?").....	13
Cracking WEP Keys (Keys to the Kingdom)	15
<i>AirSnort</i>	15
<i>WEPCrack</i>	18
Sniffing Traffic (Something Smells Fishy)	20
<i>Malicious Attackers</i>	21
<i>Denial-of-Service Attacks</i>	21
Securing WLANs.....	23
WLAN Hardening Checklist	23
<i>Do Not Rely on Wep for Encryption</i>	23
<i>Segregate Wireless Networks</i>	23
<i>Do Not Use a Descriptive Name for SSID Or Access Point</i>	23
<i>Hard Code MAC Addresses that Can Use the AP</i>	23
<i>Change Encryption Keys</i>	24
<i>Disable Beacon Packets</i>	24
<i>Locate APs Centrally</i>	24
<i>Change Default Passwords/IP Addresses</i>	24
<i>Avoid WEP Weak Keys</i>	24
<i>Do Not Use DHCP on WLANs</i>	25
<i>Identify Rogue Access Points</i>	25
The Future of 802.11x Security	25
<i>TKIP</i>	25
<i>AES</i>	26
<i>802.1x</i>	26
<i>Too Little Too Late</i>	26
Other Security Concerns	26
<i>Physical Security</i>	26
End-User Awareness.....	27
Conclusion.....	28
Acknowledgements.....	29
Appendix A: Auditing Tools.....	30
WLAN Scanners.....	30
WLAN Sniffers	30
WEP Key Crackers	30
Other.....	31



Appendix B: Statistics..... 32
 War Driving and Walking 32
Appendix C: References 34
Appendix D: IEEE Task Groups..... 35



EXECUTIVE SUMMARY

Wireless networking technology is becoming increasingly popular but, at the same time, has introduced many security issues. The popularity in wireless technology is driven by two primary factors — convenience and cost. A wireless local area network (WLAN) allows workers to access digital resources without being tethered to their desks. Laptops could be carried into meetings or even out to the front lawn on a nice day. This convenience has become affordable. Vendors have begun to produce compatible hardware at a reasonable price with standards such as the Institute of Electrical and Electronics Engineers Inc.'s (IEEE's) 802.11x.

However, the convenience of WLANs also introduces security concerns that do not exist in a wired world. Connecting to a network no longer requires an Ethernet cable. Instead, data packets are airborne and available to anyone with the ability to intercept and decode them. Traditional physical security measures like walls and security guards are useless in this new domain.

Several reports have discussed weaknesses in the Wired Equivalent Privacy (WEP) algorithm employed by the 802.11x standard to encrypt wireless data. This has led to the development of automated tools, such as AirSnort and WEPCrack, that automate the recovery of encryption keys. The IEEE has organized the 802.11i Task Group to address 802.11x security, and hardware vendors are racing to implement proprietary solutions. Still, securing vulnerable networks could take some time. Beyond this, research has shown that that majority of networks use no encryption at all. WEP is far from perfect, but it does at least provide a deterrent to attackers.

WLANs introduce security risks that must be understood and mitigated. If not, vulnerable WLANs can compromise overall network security by allowing the following attack scenarios:

- Vulnerable WLANs provide attackers with the ability to passively obtain confidential network data and leave no trace of the attack.
- Vulnerable WLANs, positioned behind perimeter firewalls and considered to be trusted networks, may provide attackers with a backdoor into a network. This access may lead to attacks on machines elsewhere on the wired LAN.
- Vulnerable WLANs could serve as a launching pad for attacks on unrelated networks. WLANs provide convenient cover, as identifying the originator of an attack is difficult if not impossible.

Tools to identify WLANs, break WEP encryption keys and capture network traffic are freely available. To protect against attacks, understand both the vulnerabilities that exist and how attackers employ these tools to exploit the vulnerabilities. Identify compensating controls and determine if the risks can be mitigated to an acceptable level to justify the introduction of wireless network technology.

This paper addresses how to find the vulnerabilities inherent in the WEP algorithm, how to determine if a WLAN is vulnerable using freeware tools and, most importantly, how to best secure WLANs.



WEP INSECURITIES

Two researchers from the University of California at Berkeley and one from Zero Knowledge Systems Inc. published a report identifying security weaknesses within the Wired Equivalency Privacy (WEP) algorithm in 2001.¹ Based on their research, WEP was found to be insecure due to improper implementation of the RC4 encryption algorithm and the use of a 32-bit cyclical redundancy check (CRC-32) checksum for data integrity. These vulnerabilities create the potential for active and passive attacks that could allow attackers to decrypt traffic or inject unauthorized data into a network. Furthermore, the researchers hypothesized that the attacks would not require specialized equipment but could be conducted using readily available hardware sold at consumer electronics stores.² (At the risk of losing reader suspense, the prediction was very accurate indeed.) Hackers began automating the exploits once the vulnerabilities were made public.

What is 802.11x?

Wireless LAN standards are defined by the IEEE's 802.11 working group. WLANs come in three flavors, namely 802.11b, 802.11a and 802.11g.³ 802.11b-networking equipment first became available in 1999 and quickly gained popularity. 802.11b operates in the 2.4000-GHz to 2.4835-GHz frequency range and can operate at up to 11 megabits per second, although it can also reduce throughput to 5.5 Mbps, 2 Mbps or 1 Mbps when interference degrades signal quality.⁴ The 802.11a standard increases throughput to a theoretical maximum of 54 Mbps and operates in the 5.15- to 5.35-GHz through 5.725- to 5.825-GHz frequency range. 802.11a hardware first became available in late 2001. Due to operation at different frequencies, 802.11a is not compatible with 802.11b hardware. Finally, the 802.11g standard has not yet been approved but promises compatibility with 802.11b hardware as it too will operate at the 2.4-GHz frequency. The major advantage that will be offered by the 802.11g standard will be increased bandwidth comparable to 802.11a at 54 Mbps.⁵

Confused? For the purposes of this paper, keep in mind that WEP is defined in the 802.11 standard, not the individual standards for the 802.11b, 802.11a or 802.11g task groups. As a consequence, WEP vulnerabilities have the potential to affect all flavors of 802.11 networks; therefore, this paper frequently refers to WLANs as 802.11x networks.

When setting up a WLAN, the channel and service set identifier (SSID) must be configured in addition to traditional network settings such as an IP address and a subnet mask. The channel is a number between one and 11 (one and 13 in Europe) and designates the frequency on which the

¹ Nikita Borisov, Ian Goldberg and David Wagner, "Intercepting Mobile Communications: The Insecurity of 802.11," March 3, 2001. Available at <http://www.isaac.cs.berkeley.edu/isaac/wep-draft.pdf>.

² See the section Auditing WLANs on page 13 for more on the topic.

³ See Appendix D: IEEE Task Groups on page 35 for a listing of all 802.11 task groups

⁴ Rob Schenk, Andrew Garcia and Russ Iwanchuk, "Wireless LAN Deployment and Security Basics," Aug. 29, 2001. Available at http://www.extremetech.com/article/0,3396,s=1034&a=13521_00.asp.

⁵ Bruce Brown, "Wireless Standards Up in the Air," Dec. 3, 2001. Available at <http://www.extremetech.com/article2/0,3973,9164,00.asp>.



network will operate (see **Figure 1: 802.11b channels**). The SSID is an alphanumeric string that differentiates networks operating on the same channel. It is essentially a configurable name that identifies an individual network. These settings are important factors when identifying WLANs and sniffing traffic, which is discussed later.

Channel	Frequency (GHz)
1	2.412
2	2.417
3	2.422
4	2.427
5	2.432
6	2.437
7	2.442
8	2.447
9	2.452
10	2.457
11	2.462

Figure 1: 802.11b channels

What is WEP?

WEP is a component of the IEEE 802.11 WLAN standards. Its primary purpose is to provide for confidentiality of data on wireless networks at a level equivalent to that of wired LANs. Wired LANs typically employ physical controls to prevent unauthorized users from connecting to the network and thereby viewing data. In a wireless LAN, the network can be accessed without physically connecting to the LAN; therefore, the IEEE chose to employ encryption at the datalink layer to prevent unauthorized eavesdropping on a network. This is accomplished by encrypting data with the RC4 encryption algorithm. WEP employs an integrity check field in each data packet to ensure that data is not modified during transmission. A CRC-32 checksum is used for this purpose.

Issues

INITIALIZATION VECTOR

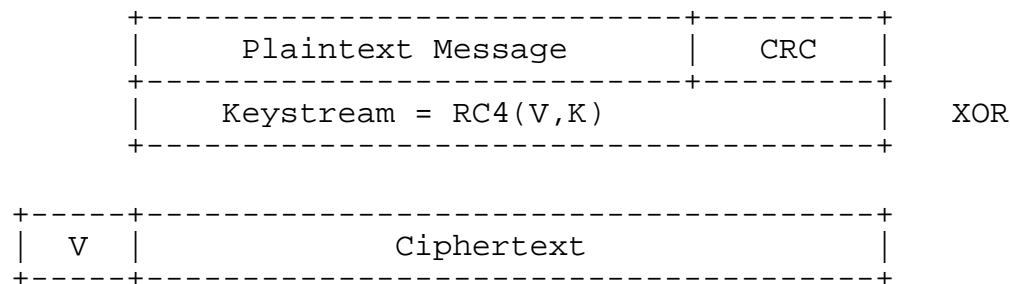
RC4 is a stream cipher designed by Ron Rivest for RSA Security. A stream cipher expands a fixed-length key into an infinite pseudo-random key stream for the purpose of encrypting data. In WEP, plain-text data is exclusive or'd with the key stream to produce the cipher text. Exclusive or (XOR) is a Boolean operator that compares two numbers and determines if they are the same or different. If the numbers are the same, a value of "0" is returned; if they are different, a value of "1" is returned. The following example shows the binary equivalent of the letter "b" being XOR'd with the binary equivalent of the letter "n":

```
01100010 The letter b, in binary
01101110 The letter n, in binary
00001100 The XOR'd value.
```



WEP requires that each wireless network connection share a secret key for encryption purposes. WEP does not define key management techniques such as the number of different keys used within a network or the frequency to change keys. In practice, networks use one or only a few keys among access points and change keys infrequently, as most vendor implementations of WEP require that keys be changed manually. The key stream produced by the WEP algorithm depends upon both the secret key and an initialization vector (IV). The IV is used to ensure that subsequent data packets are encrypted with different key streams, despite using the same secret key. The IV is a 24-bit field that is unencrypted within the header of the data packet, as shown below:

V = Initialization Vector
 K = Secret Key



According to the Berkeley report, the use of a 24-bit IV is inadequate because the same IV, and therefore the same key stream, must be reused within a relatively short period of time. A 24-bit field can contain 2^{24} or 16,777,216 possible values. Given a network running at 11 Mbps and constantly transmitting 1,500-byte packets, an IV would be repeated (referred to as an IV collision) about every 5 hours as the following calculations detail:

11 Mbps ÷ (1,500 bytes per packet × 8 bits per byte) = 916.67 packets transmitted each second
 16,777,216 IVs ÷ 916.67 packets per second = 18,302.41745 seconds to use all IVs
 18,302.41745 seconds × 60 seconds per minute × 60 minutes per hours = 5.0840048 hours to use all IVs

This time could be reduced under various circumstances. The aforementioned scenario assumes only one device on the network transmitting data and incrementing IVs by “1” for each packet transmitted. Each additional device using the same secret key would reduce this time. Devices that use random IVs would also reduce the time required for an IV collision to occur. Once an IV collision occurs and an attacker has two different plain-text messages encrypted with the same key stream, it is possible to obtain the XOR of the two plain-text messages by XORing the two cipher text messages. The XOR that results can then be used to decrypt traffic.⁶ The following calculation shows how XORing two ciphertexts cancels out the key stream:

⁶ As explained in the Attacks section on page 10.



C_1 = Ciphertext 1
 C_2 = Ciphertext 2
 P_1 = Plaintext 1
 P_2 = Ciphertext 2
 V = initialization vector
 K = secret key
 \oplus = XOR

If $C_1 = P_1 \oplus RC4(V,K)$
And $C_2 = P_2 \oplus RC4(V,K)$
Then $C_1 \oplus C_2 = (P_1 \oplus RC4(V,K)) \oplus (P_2 \oplus RC4(V,K))$
 $= P_1 \oplus P_2$

Let's test this theory with the following example.

	Data
Letter "a" plain-text	01100001
Letter "n" – secret key	01101110
XOR – "a"	00001111

	Data
Letter "b" plain-text	01100010
Letter "n" – secret key	01101110
XOR – "b"	00001100

	Data
XOR – "a"	00001100
XOR – "b"	00001111
XOR – "a" & "b"	00000011

	Data
Letter "a" plain-text	01100001
Letter "b" plain-text	01100010
XOR – "a" & "b"	00000011

Therefore, when using the same secret key, the XOR'd value of the plain-text messages ("a" and "b") is equivalent to the XOR'd value of the encrypted messages. Thus, if an attacker has knowledge of the contents of one plain-text message when an IV collision occurs, the attacker could then decipher the contents of the other plain-text message without any knowledge of the key stream used for encryption.

CYCLICAL REDUNDANCY CHECK

WEP uses CRC-32 to ensure the integrity of data transmitted over the wireless network. Cyclical redundancy checking (CRC) enhances the integrity of transmissions by calculating a checksum that is included with each data packet. The recipient calculates the same checksum for each data packet. If the checksums are equivalent, WEP provides assurance that the data has not been changed during transmission. Transmitted messages are divided into predetermined lengths and are divided by a fixed divisor. The remainder is one bit smaller than the divisor and serves as the



checksum. In the case of CRC-32, the remainder is a 32-bit number and this checksum is then appended onto the message sent. In the following example, a CRC-32 checksum (101001010010011111111011111001) for the letter “b” (01100010) is calculated:

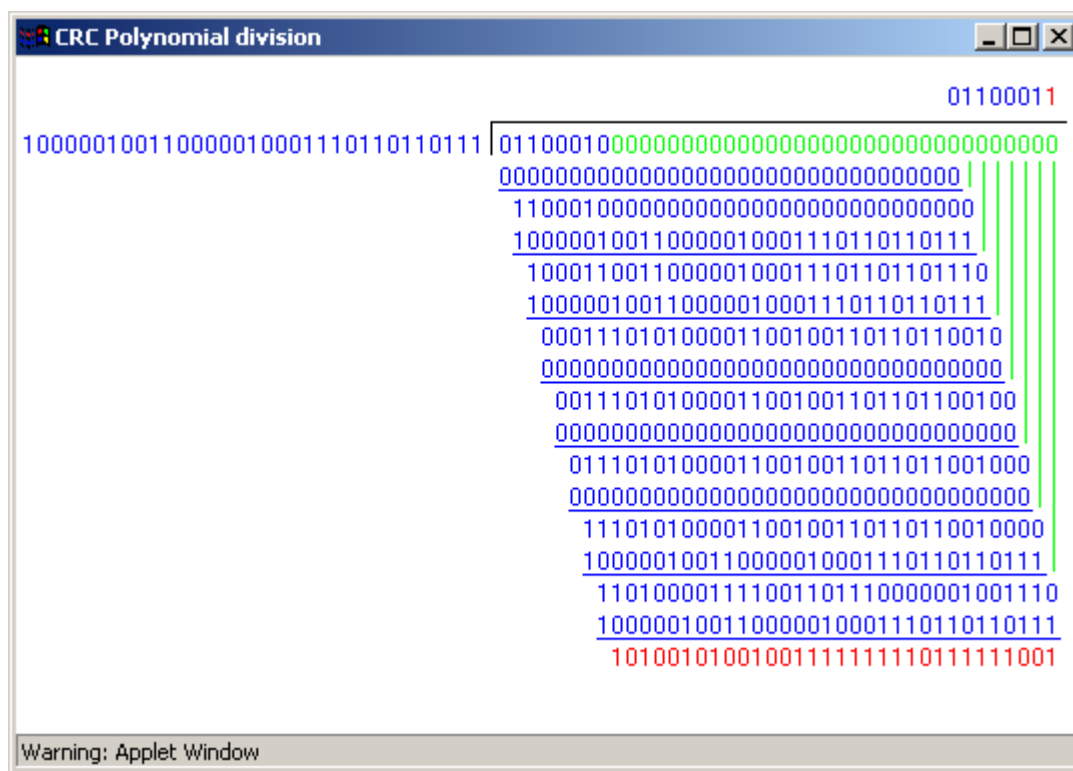


Figure 2: CRC-32 checksum for the letter “b”

According to the Berkeley report, CRC-32 is not an appropriate integrity check for WEP as it is a linear checksum. Therefore, modifications could be made to the ciphertext, and the bit difference between the original and modified checksums could be calculated. An attacker may adjust the checksum appropriately, and a recipient would not be aware that the data has been altered.

Let’s assume the following scenario. The letter “b” is being encrypted using a secret key of letter “n.” To ensure data integrity, a CRC-8 checksum is used and encrypted in the data packet. An attacker wants to alter the message by flipping bits in the encrypted data packet. If the attacker were to simply flip the appropriate bits in the ciphertext, the decrypted checksum would no longer match and WEP would reveal that the data was altered. Therefore, the attacker must also determine the appropriate bits to flip in the encrypted checksum. Prior to any alteration, the encrypted data packet is calculated as follows:

	Data	CRC-8
Letter “b” plain-text	01100010	00101001
Letter “n” – secret key	01101110	01101110
XOR encryption	00001100	01000111

The attacker could determine the bits that need to be flipped in the checksum by XORing the change to the data and its corresponding CRC-8 checksum against the original data and its



checksum, as follows:

	Data	CRC-8
XOR encryption	00001100	01000111
Change	00000011	00001001
Altered XOR encryption	00001111	01001110

To see if the altered checksum was calculated correctly, first decrypt the data and its checksum.

	Data	CRC-8
Altered XOR encryption	00001111	01001110
Letter 'n' – secret key	01101110	01101110
Decrypted data – letter 'a'	01100001	00100000

The decrypted data (01100001) turns out to be the letter “a.” Next, let’s calculate the CRC-8 checksum for the letter “a.”

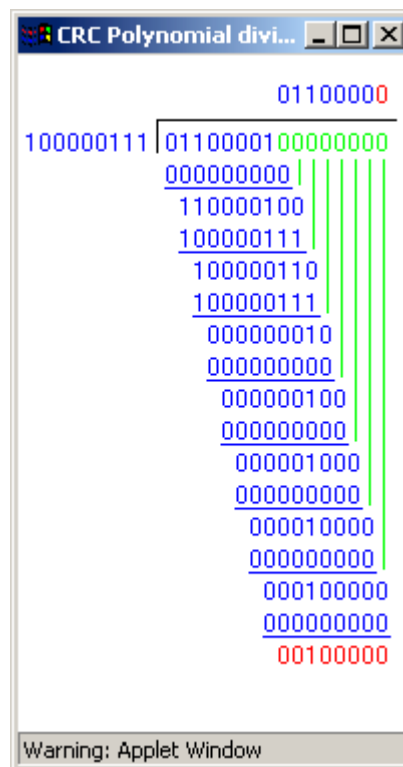


Figure 3: CRC-8 checksum for the letter “a”

The CRC-8 checksum (00100000) was calculated correctly; therefore, the altered packet would not appear to have been intercepted. Note that the attacker does not need to have complete knowledge of the original plain-text message. The attacker only requires knowledge of the bits to be changed.

Attacks

Collisions of IVs make WEP susceptible to having cipher text decrypted. Once the XOR of two plain-text messages is obtained, at least partial knowledge of one of the plain-text messages can



be used to decrypt the other plain-text message. Moreover, research done by Fluhrer, Mantin and Shamir revealed that prior knowledge of only the first byte of plain-text, rather than the entire message is required to derive information about the key bytes.⁷ Messages being transmitted on a network often use sufficient repetition that they lend themselves to prediction. For example, login sequences tend to follow the same text format consistently.

Another means of determining the contents of one of the two plain-text messages is for the attacker to implement a known plain-text attack by creating messages and injecting them into the network. Consider the following scenario. An attacker could send an e-mail message to a recipient who is using a wireless network. When the user retrieves the e-mail message, it would be transmitted from the e-mail server to the wireless access point, where it would be encrypted with the WEP algorithm. The encrypted message would then be transmitted to the user. Simultaneously, the attacker could sniff the network traffic and grab the packets containing the encrypted e-mail. Once an IV collision occurs and the attacker captures a subsequent message encrypted with the same key stream, decryption of the new plain-text message would be possible. With the two plain-text messages and their encrypted XOR values, the key stream could then be calculated.

Given sufficient time, an attacker could develop a dictionary of key streams and ultimately decrypt all traffic on the network.

Stubblefield, Ioannidis and Rubin have demonstrated that predicting the plain-text content of encrypted messages is even easier than the aforementioned scenarios demonstrate.⁸ The 802.11 header encapsulates and encrypts the headers of higher-level protocols such as ARP and IP. Therefore, the first plain-text byte of the encrypted message becomes easier to predict as the structure of headers follows documented standards. If the attacker can determine the type of packet being sent, the attacker could then drastically narrow the possibilities for the plain-text contents of the first byte in the encrypted message. Depending upon factors such as packet size or when during transmission packets are sent, predicting packet types becomes a possibility.

However, Stubblefield, Ioannidis and Rubin also determined that even this might not be necessary. They discovered that, on an 802.11x network, an additional 802.2 (Logical Link Control) Subnetwork Access Protocol (SNAP) header is added for all IP and ARP traffic. This discovery revealed that all IP and ARP traffic has the same first plain-text byte (0xAA), thereby eliminating the need for devising a known plain-text attack or attempting to determine packet types to predict the first byte in the encrypted packet. WEP key crackers such as WEPCrack take advantage of this fact when deciphering the WEP key.⁹

The reliance on CRC-32 checksums for integrity checking leaves WEP networks vulnerable to the injection of unauthorized and unnoticed data. This can obviously lead to numerous exploitation techniques and ultimately endanger the overall security of the network. Note the

⁷ Scott Fluhrer, Itsik Mantin and Adi Shamir, "Weaknesses in the Key Scheduling Algorithm of RC4," August 2001. Available at http://online.securityfocus.com/data/library/rc4_ksaproc.pdf.

⁸ Adam Stubblefield, John Ioannidis and Aviel D. Rubin, "Using the Fluhrer, Mantin and Shamir Attack to Break WEP," Aug. 21, 2001. Available at http://www.cs.rice.edu/~astubble/wep/wep_attack.pdf.

⁹ See Auditing WLANs on page 13, Cracking WEP Keys (Keys to the Kingdom) on page 15 and WEPCrack on page 18.



Berkeley paper only discusses such attacks at a theoretical level and does not attempt a proof of concept. However, these forms of active attacks are overshadowed by the IV attacks. If attackers could crack the encryption keys being used, they could then connect to the network and send traffic that appears to be legitimate. This would eliminate the need to inject packets by taking advantage of CRC-32 weaknesses.

IEEE 802.11 Chair Response¹⁰

Stuart J. Kerry, the chair for the IEEE 802.11 standards group, responded to the Berkeley report by acknowledging the shortcomings of WEP but also offered justifications. Kerry pointed out that the goals for WEP never included absolute security. Like all security mechanisms, the goal is to achieve a level of security that requires attackers to expend effort to obtain protected data that exceeds the value of the data itself. He agreed that WEP could be made more secure but felt that it had achieved its specified goals. However, he also indicated that the subcommittee planned to add WEP enhancements to the 802.11b standard that would address the weaknesses detailed in the Berkeley report. The effort to add such enhancements began with the formation of the 802.11i Task Group.¹¹

¹⁰ Stuart J. Kerry, "Chair of IEEE 802.11 Responds to WEP Security Flaws," Feb. 15, 2001. Available at <http://slashdot.org/articles/01/02/15/1745204.shtml>.

¹¹ See The Future of 802.11x Security on page 25.

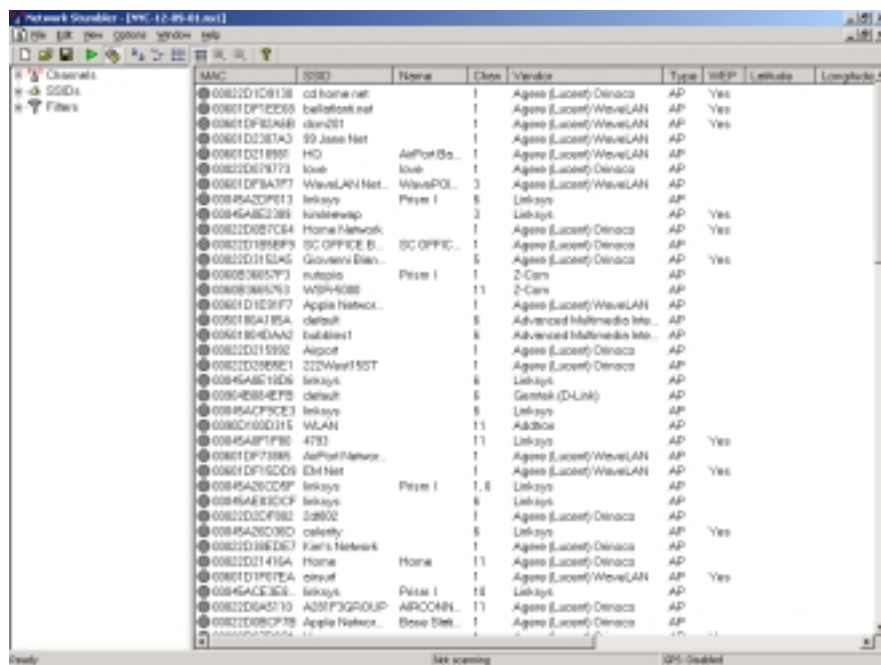


AUDITING WLANS

Finding WLANs (“What’s the Frequency, Kenneth?”)

By design, 802.11x WLANs make the process of identifying wireless networks relatively straightforward. To find one another, wireless access points (APs) and clients send beacons and broadcasts (aka probes) respectively.¹² Beacons are sent by APs at predefined intervals. They are essentially invitations and driving directions that enable the client to find the AP and configure the appropriate settings to communicate. A beacon announces the SSID and the channel that the network is using. The SSID is simply a text string that differentiates an 802.11x network from others operating on the same channel. The channel is a number between 1 and 11 (US) or 1 and 13 (Europe) that identifies the frequency on which the network is operating.

While this system allows simple configuration of networks and minimizes hassle when moving between networks, it is a significant security weakness. Fortunately, some APs allow for beacon packets to be disabled. This action would not, however, prevent WLAN scanners such as NetStumbler from identifying WLANs, as some scanners operate by sending a steady stream of broadcast packets on all possible channels.¹³ APs respond to broadcast packets to verify their existence, even if beacons have been disabled.



MAC	SSID	Name	Chan	Vendor	Type	WEP	Latitude	Longitude
00822D1C9138	cd home net		1	Agere (Lucent)/Dinaca	AP	Yes		
00601DF1E255	ballfoote net		1	Agere (Lucent)/WaveLAN	AP	Yes		
00601DF1E2A8	slam201		1	Agere (Lucent)/WaveLAN	AP	Yes		
00601D2381A3	59 Jane Net		1	Agere (Lucent)/WaveLAN	AP			
00601D218901	HD	AirPort Ex...	1	Agere (Lucent)/WaveLAN	AP			
00822D178172	boob	boob	1	Agere (Lucent)/Dinaca	AP			
00601DF19A3F7	WaveLAN Net...	WaveLAN	3	Agere (Lucent)/WaveLAN	AP			
00845A2C3F013	lekays	Prism I	8	Leksys	AP			
00845A2C2389	knickerwap		3	Leksys	AP	Yes		
00822D087C54	Home Network		1	Agere (Lucent)/Dinaca	AP	Yes		
00822D1898F9	SC OFFICE B...	SC OFFICE B...	1	Agere (Lucent)/Dinaca	AP			
00822D115245	Giovanni Dian...		5	Agere (Lucent)/Dinaca	AP	Yes		
0060836857P3	nutopia	Prism I	1	2-Cars	AP			
0060836857P3	WSP4088		11	2-Cars	AP			
00601D1E3F7	Apple Netwo...		1	Agere (Lucent)/WaveLAN	AP			
0050185A185A	default		5	Advanced InfMedia Info...	AP			
00601864DA2	ballblat1		6	Advanced InfMedia Info...	AP			
00822D115992	Airport		1	Agere (Lucent)/Dinaca	AP			
00822D1389E1	222Wave155T		1	Agere (Lucent)/Dinaca	AP			
00845A2C1626	lekays		8	Leksys	AP			
00904B94E7P2	default		5	Genex (D-Link)	AP			
00845A2C3CE3	lekays		8	Leksys	AP			
00822D16211E	WLAN		11	AdiTech	AP			
00845A2C1785	4783		11	Leksys	AP	Yes		
00601DF17386	AirPort Netwo...		1	Agere (Lucent)/WaveLAN	AP			
00601DF15CDE9	Ext Net		1	Agere (Lucent)/WaveLAN	AP	Yes		
00845A2C2C9F	lekays	Prism I	1, 8	Leksys	AP			
00845A2C3C9F	lekays		8	Leksys	AP			
00822D1C3F82	SMB02		5	Agere (Lucent)/Dinaca	AP			
00845A2C20C0	cafethy		5	Leksys	AP	Yes		
00822D188DE3	Karl's Network		1	Agere (Lucent)/Dinaca	AP			
00822D114164	Home	Home	11	Agere (Lucent)/Dinaca	AP			
00601D1FETE4	cinuf		1	Agere (Lucent)/WaveLAN	AP	Yes		
00845A2C3C3E	lekays	Prism I	10	Leksys	AP			
00822D0A4510	ASSTP3GROUP	ASSTP3GROUP	11	Agere (Lucent)/Dinaca	AP			
00822D09CF79	Apple Netwo...	Base Del...	1	Agere (Lucent)/Dinaca	AP			

Figure 4: NetStumbler in action

¹² William Arbaugh, Narendar Shankar and Justin Wan, “Your 802.11 Wireless Network has No Clothes,” March 30, 2001. Available at <http://downloads.securityfocus.com/library/wireless.pdf>.

¹³ NetStumbler is available from <http://www.NetStumbler.org>.



WLAN scanners are the modern equivalent of the police scanner. WLAN scanners allow users to identify WLANs through the use of a wireless network interface card (NIC) running in promiscuous mode and software that will probe for APs. While a handful of WLAN scanners are available, NetStumbler is likely the most popular on the Windows platform. Not only is it free, but it also provides an easy-to-use graphical interface with features such as the ability to incorporate GPS to identify the longitude and latitude of an identified AP.¹⁴ This is convenient for an attacker who wants to return at a later time for sniffing traffic or cracking WEP keys. NetStumbler was created by Marius Milner and has developed a bit of a cult following. NetStumbler.org has an ongoing project that allows individuals to upload their war-driving results to the website. Due to the GPS functionality of NetStumbler, the site has built a repository of AP locations throughout the US. Results are displayed graphically on maps and users can even select individual APs and see where they reside.

Think about it — a website that identifies a company’s insecure network for the entire world to see. Imagine a section in the newspaper where you could look up companies that choose to leave their doors unlocked at night; this website provides a similar service. Fortunately, the administrators of NetStumbler.org allow organizations to request removal of their AP information, but security through obscurity is no substitute for the real thing.

Linux aficionados will appreciate Kismet.¹⁵ Kismet is not graphical and not as user friendly as NetStumbler, but it provides superior functionality. Kismet is not only a WLAN scanner, but combines the features of a WLAN sniffer. While scanning for APs, packets can also be logged for later analysis. Logging features allow for captured packets to be stored in separate buckets, depending upon the type of traffic captured. Kismet can store encrypted packets that use “weak keys” separately to run them through a WEP key cracker.¹⁶

In late 2001, iDEFENSE Labs joined the NetStumbler bandwagon. Equipped with a laptop running Microsoft Windows 2000 Professional, NetStumbler v0.3.23 and a Lucent Orinoco Gold 802.11b PC card, iDEFENSE Labs set out to explore local WLANs. The Labs initially had no specialized antenna to boost signal strength.¹⁷ iDEFENSE Labs used only basic hardware and software available at any local computer store. The experiment began with the launching of NetStumbler running on a laptop placed in the passenger seat of an automobile.

The initial foray into the world of war driving took iDEFENSE Labs into the technology corridor in Northern Virginia. At first the laptop received no responses, prompting concerns over its proper configuration. However, within a few minutes, the chime croaked by NetStumbler to indicate the presence of a WLAN sounded. After about 45 minutes of war driving, iDEFENSE Labs identified about 40 WLANs. The Labs conducted follow-up drives.¹⁸

¹⁴ See Appendix A: Auditing Tools on page 30.

¹⁵ See <http://www.kismetwireless.net>.

¹⁶ See AirSnort on page 15.

¹⁷ Although such things are available. See http://www.hyperlinktech.com/web/antennas_2400.html, <http://www.telexwireless.com/24ghzantennas.htm> or <http://www.antennasystems.com/broadband.html>

¹⁸ The results of some of these journeys are shown in Appendix B: Statistics found on page 32.



iDEFENSE Labs decided to follow up its drives through northern Virginia with drives through Manhattan. Due to the large number of people crammed onto the tiny island, the Labs expected it to be a hotbed of WLAN traffic. The results were impressive beyond imagining. The first war driving expedition into Manhattan, a 15-minute cab ride from the Upper East Side to the Meat Packing district, allowed NetStumbler to record 106 WLANs, 77 of which used no encryption whatsoever.

The most astonishing discovery to result from the war driving has to be the lack of encryption used by wireless networks. iDEFENSE Labs does not claim the results in Appendix B: Statistics portray a proper scientific study, but the findings represent a significant problem.¹⁹ Seventy-five percent of Manhattan networks did not possess any encryption; about 72 percent of the northern Virginia networks did not. WEP has its flaws, but at least it does provide some degree of security. If an attacker living in a populated area could access dozens — if not hundreds — of WLANs to hack, the attacker would not likely bother to attack one using WEP because many WLANs would offer no security challenge at all.

In a best-case scenario, several hours would be necessary to obtain a WEP key, but an attacker needs only a few minutes to identify a wide-open network. Once a non-WEP-enabled WLAN is identified, the attacker could begin sniffing plain-text traffic immediately. If free Internet access is the goal, the attacker only needs to obtain a valid IP address, a challenge made trivial by the use of DHCP on WLANs. Even without DHCP, only a limited number of private IP address ranges are available.²⁰ Therefore, a determined attacker would ultimately be able to steal resources.

Cracking WEP Keys (Keys to the Kingdom)

The automating of attack tools by hackers was inevitable following the release of white papers such as “Using the Fluhrer, Mantin and Shamir Attack to Break WEP” and “Intercepting Mobile Communications: The Insecurity of 802.11,” both of which discussed attacks on the WEP algorithm. A wide range of tools may be available for download, but WEPCrack and AirSnort are two of the most popular.²¹ WEPCrack is a series of Perl scripts designed to crack WEP keys using data captured by a sniffer. AirSnort, on the other hand, is more all encompassing. AirSnort obtains the traffic necessary for breaking the encryption keys itself without the need for a separate sniffer.

AIRSNORT

AirSnort is a Linux-based tool written by Jeremy Bruestle and Blake Hegerle. It exploits WEP vulnerabilities discussed in the Stubblefield, Ioannidis and Rubin paper and requires a version of Linux using the 2.2 or 2.4 kernel, wlan-ng drivers and a network card that uses the Prism2 chipset.²² Not all tools are compatible with the same wireless network cards, resulting in one of the difficulties in auditing WLANs using the tools discussed in this paper. This is due to a lack of

¹⁹ See page 32.

²⁰ See RFC 1918.

²¹ WEPCrack is available at <http://sourceforge.net/projects/wepcrack> and AirSnort can be found at <http://www.be-secure.com/airsnort.html>.

²² wlan-ng drivers are available from <http://www.linux-wlan.com>.



readily available drivers for the cards. The lack of drivers is likely to be a moot point over time, but one may need to buy at least two separate network cards if planning to use freeware tools for now.

NetStumbler and most Windows-based tools require a NIC using the Hermes chipset, while AirSnort and most Linux-based tools are only compatible with cards using the Prism2 chipset (although AirSnort v2.0 claims to support ORiNOCO cards with appropriate patches to the orinoco_cs driver). **Figure 6: Wireless PCMCIA network cards** lists specific cards that use the two different chipsets.

Hermes Chipset	Prism2 Chipset
ORiNOCO (Lucent PC) Card	Addtron AWP-100
Dell TrueMobile 1150	Ambicom WL100B-PC
Avaya Wireless PC Card	Bromax Freeport
Compaq WL110	Compaq WL100
Enterasys Roamabout	D-Link DWL-650
Elsa Airlancer MC-11	GemTek WL-211
ARtem CC-W11	Intalk/Nokia WL201
IBM High Rate Wireless LAN	Linksys WPC11
Buffalo WLI-PCM-L11	Samsung SWL2000-N
1stWave 1ST-PC-DSS11IS, DSS11IG, DSS11ES, DSS11EG	SMC 2632W
	Teletronics WL1000
	YDI Diamond
	Z-Com XI300
	Zoom Telephonics ZoomAir 4100

Figure 6: Wireless PCMCIA network cards

AirSnort is a very useful tool once it is up and running, but it can be challenging to compile. It may take a fair bit of experimentation before the discovery of the right combination of Linux kernel, PCMCIA card services, wlan-ng drivers and AirSnort versions that are willing to work together. iDEFENSE Labs found that RedHat Linux 7.1 running the 2.4.2-2 kernel, PCMCIA Card Services 3.1.22 and AirSnort 0.0.9 cooperate nicely.

Once AirSnort is running, the NIC must be in promiscuous mode and set to listen on the appropriate channel for the targeted WLAN. Obtain the channel from the WLAN scanner used to locate the WLAN in the first place. AirSnort comes with a shell script (dopromisc.sh) that will automatically launch the NIC in promiscuous mode with the appropriate channel setting, but the channel has to be hard-coded into the script if the default of channel 6 is not appropriate. AirSnort itself is comprised of two separate applications – capture and crack. Once the NIC is in promiscuous mode, launch the capture application using the following command:

```
capture -c <filename>
```

The -c flag displays the progress of the capture. You would know immediately if the application is working properly because the Encrypted Packets counter would begin to increment. **Figure 7: AirSnort capture** shows a screenshot of AirSnort in action capturing packets.



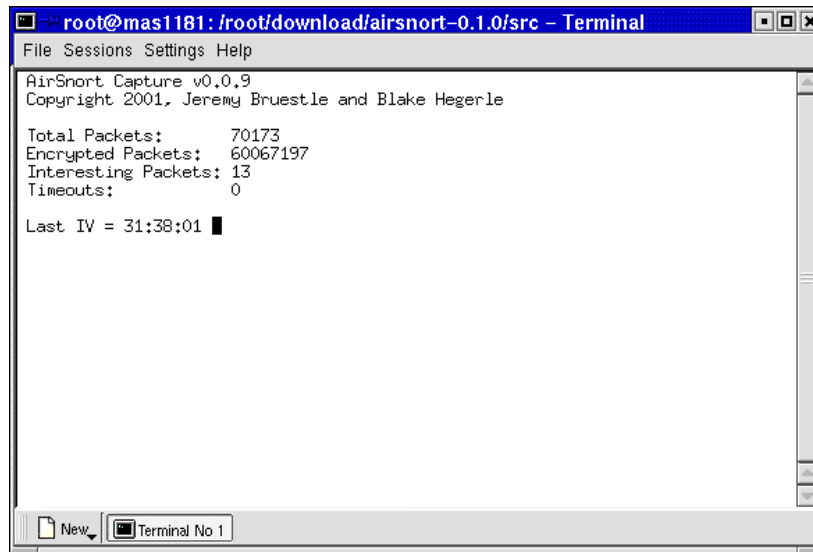


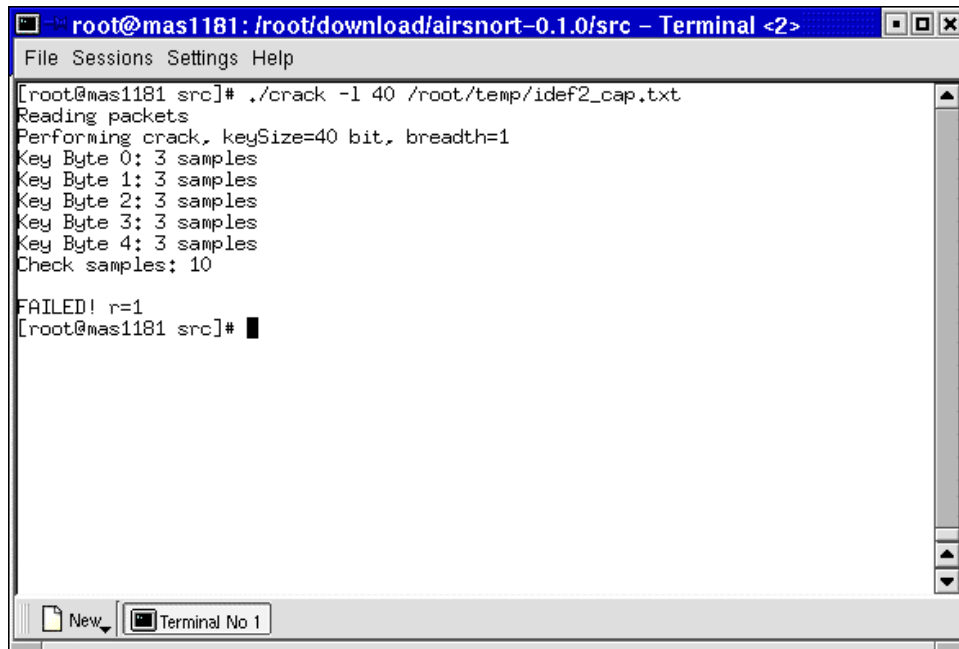
Figure 7: AirSnort capture

AirSnort will also display the number of “Interesting Packets” (aka weak keys) that have been captured. AirSnort is efficient because it does not capture all encrypted packets but rather only those that would be used to crack the WEP encryption key. Interesting packets are those where the second byte of the IV is 0xFF. Once a sufficient number of interesting packets have been captured, attempt to crack the WEP key by launching the crack application in a separate console window using the following command:

```
crack -c -l <keysize> <filename>
```

If a sufficient number of interesting packets have been obtained, the WEP shared key will be returned. If not, the message in **Figure 8: Unsuccessful attempt to crack a 40-bit key using AirSnort** would be shown. Unsuccessful cracking attempts do not affect the capture process. Therefore, if at first you don’t crack, try, try, again. According to the AirSnort ReadMe file, about 1,500 interesting packets are required to successfully crack a 128-bit key. In practice, it actually requires a fair bit more.





```
root@mas1181: /root/download/airsnort-0.1.0/src - Terminal <2>
File Sessions Settings Help
[root@mas1181 src]# ./crack -l 40 /root/temp/idef2_cap.txt
Reading packets
Performing crack, keySize=40 bit, breadth=1
Key Byte 0: 3 samples
Key Byte 1: 3 samples
Key Byte 2: 3 samples
Key Byte 3: 3 samples
Key Byte 4: 3 samples
Check samples: 10
FAILED! r=1
[root@mas1181 src]#
```

Figure 8: Unsuccessful attempt to crack a 40-bit key using AirSnort

To test the application, set up the test environment shown in **Figure 10: Network diagram of test environment**. Artificially generate network traffic using a UDP flooder by sending streams of UDP packets to port 80 on the AP (used to administer the AP) from the wireless client.²³ This simulates network traffic and decreases the amount of time required to crack the key. When network traffic is near the capacity of 11 Mbps, cracking a 40-bit WEP key may take three to four hours. Cracking time is dependent upon both the key size and the amount of traffic on the network. These cracking times represent optimal conditions, but they certainly prove that WEP can be cracked and an attacker needs only patience and time to access data.

WEPCrack

WEPCrack is a SourceForge project that is administered by Paul Danckaert and Anton Rager. It is easier to use than AirSnort. WEPCrack is simply a set of Perl scripts and does not therefore require any configuration. However, WEPCrack must be used in conjunction with a separate sniffer because it does not have the ability to capture traffic. It is comprised of the following four scripts:

- **prisim-decode.pl**: Used to decode data packets once the WEP key has been cracked.
- **prisim-getIV.pl**: Extracts weak IVs and the first byte of encrypted data from a prisdump capture.
- **WeakIVGen.pl**: Creates a list of weak IVs and one byte of encrypted data when provided with a specific encryption key. This script can be used to test the program in the absence of captured data.
- **WEPCrack.pl**: Used to crack WEP keys given data generated by prisim-getIV.pl.

²³ UDP flooder can be found at http://www.foundstone.com/knowledge/free_tools.html.



Data capturing must be complete before using WEPCrack. A sniffer such as prismdump must capture the data.²⁴ prismdump is a very basic command line sniffer that takes no arguments and simply captures all traffic. prismdump recognizes 802.11x headers, which is obviously crucial to capture WEP traffic. prismdump uses the wiretap libraries that are included with Ethereal.²⁵ Therefore, Ethereal must be compiled successfully before prismdump can be installed.

By default, prismdump sends captured data to STDOUT. Data must be redirected to a text file. The following command can be used to capture traffic:

```
./prismdump > textfile
```

Once sufficient encrypted data has been captured, the weak IVs and the first byte of encrypted data must be extracted to a separate file. The following command will produce an IVFile.log file that contains the extracted data:

```
./prism-getIV.pl textfile
```

Run WEPCrack when the relevant data has been extracted. **Figure 9: Successful attempt to crack a 128-bit key using WEPCrack** illustrates the process. WEPCrack is less efficient than AirSnort. WEPCrack captures unnecessary data, requiring the user to extract relevant data. This could quickly consume hard drive space. AirSnort, on the other hand, only captures what it needs for cracking purposes. However, WEPCrack could utilize the prism-decode.pl script to decode all previously captured data if hard drive space is available while AirSnort only decodes new data once the WEP key has been cracked.

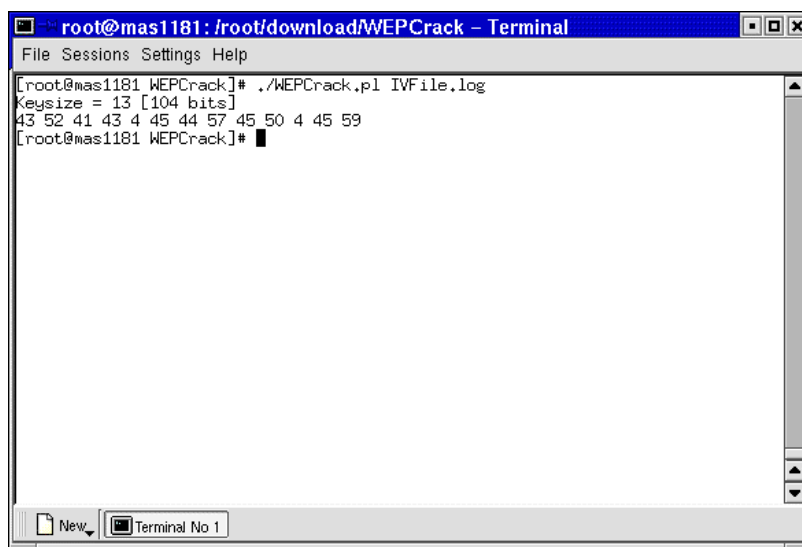


Figure 9: Successful attempt to crack a 128-bit key using WEPCrack

²⁴ prismdump is available from <http://developer.axis.com/download/tools/>.

²⁵ Ethereal is available from <http://www.ethereal.com>.



Sniffing Traffic (Something Smells Fishy)

Once the WEP key has been cracked, the process for sniffing decoded traffic is not very different than it is for a wired LAN; however, not all sniffers work with 802.11b network cards.²⁶ Some may need patches before they will work properly.

If a WLAN does not employ encryption, the only configuration detail necessary to begin sniffing traffic is the network's operating channel, which can be determined using WLAN scanning tools such as NetStumbler. Once configured, the network card needs to be placed into promiscuous mode. The mode is usually set with tools bundled with the card's drivers. The wlanctl-ng tool can change most configuration settings on 802.11b network cards, including setting the channel that the card uses and placing the card in promiscuous mode. The tool is installed along with the wlan-ng Linux drivers required for AirSnort. To actually participate on the network, the SSID (also provided by WLAN scanning tools) and an unused IP address would also need to be configured. When wireless networks use DHCP, obtaining an IP address becomes trivial as well.

Once the network card has been properly configured, launch the sniffer, sit back and relax. If interested in viewing the web pages a neighbor likes to surf or reading private e-mail messages, a sniffer capable of packet reassembly would make things a lot easier. iDEFENSE Labs is unaware of an 802.11x-aware sniffer capable of packet reassembly. However, the Labs managed to convince eEye Digital Security's Iris and a Windows based version of Ethereal to work with the Lucent ORiNOCO card when used in conjunction with the Lucent ORiNOCO drivers provided with Wildpackets AiroPeek or AiroPeek NX.²⁷ First install a demo copy of AiroPeek or AiroPeek NX. Then upgrade to the Lucent ORiNOCO drivers contained in the \Diver\Lucent directory to allow Iris or Ethereal to use the Lucent card.

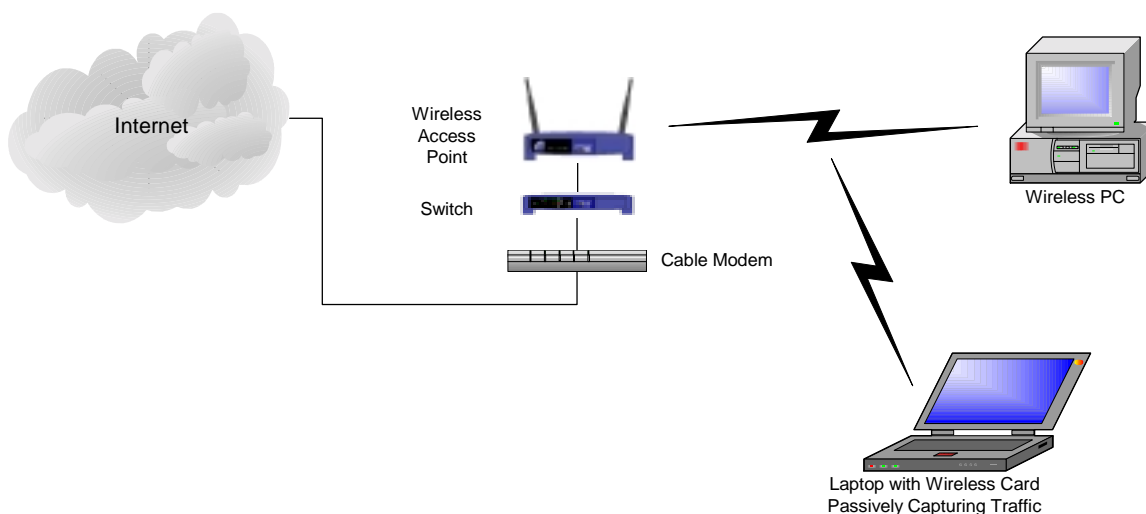


Figure 10: Network diagram of test environment

²⁶ See Appendix A: Auditing Tools on page 30 for a list of sniffers that will work on WLANs.

²⁷ eEye Digital Security's Iris is available from <http://www.eeye.com/html/Products/Iris/index.html>; Windows-based version of Ethereal can be found at <http://www.ethereal.com>; Wildpackets AiroPeek can be found at <http://www.wildpackets.com/products/airopeek/> and AiroPeek NX at http://www.wildpackets.com/products/airopeek_nx/.



MALICIOUS ATTACKERS

Beyond privacy concerns, insecure WLANs provide an easy target for attackers who want to damage a network. Once connected to a WLAN and armed with publicly available information about default settings, an attacker could deny network access to legitimate users or redirect traffic by changing the configuration on the AP. APs are generally configured by connecting a computer to the AP using a USB cable or a hard-wired Ethernet connection, or by accessing an administrative web server running on the AP itself. The last method allows the greatest security risk. Configuring the AP by connecting to a web page is convenient, but without proper precautions, a wireless hacker could also gain access to the configuration console.

The configuration consoles are generally installed with default authentication credentials and IP addresses. Such information is easily obtained by downloading vendor documentation or by viewing papers such as those offered by WI2600.org that summarize the default settings of various vendors.²⁸ This process is made even easier because APs broadcast their MAC address in beacon packets. An attacker could identify vendor hardware by looking up MAC addresses obtained using WLAN scanners by referencing the IEEE Organizationally Unique Identifier (OUI) assignment database.²⁹

Tools such as NetStumbler cross-reference this information automatically, as seen in the Vendor column in **Figure 4: NetStumbler in action**. Once access to the configuration console is obtained, an attacker has free reign to administer the AP. An attacker could trigger a denial of service by changing the channel or SSID used by the WLAN. An attacker could also implement more malicious and less noticeable changes. Depending upon the capabilities of the hardware, an attacker may redirect specified traffic. A hacker could set up a rogue AP, allow wireless clients to connect to it, and then redirect the traffic to another destination. A man-in-the-middle attack such as this could channel users to a fake server set up by the attacker or collect authentication credentials.

DENIAL-OF-SERVICE ATTACKS

WLANs are susceptible to the same protocol-based attacks that plague wired LANs but to perpetrate such attacks on WLANs, an individual would first need to connect to the network. WLANs are also susceptible to a unique form of denial-of-service (DoS) attack. WLANs send information via radio waves on public frequencies, thus they are susceptible to inadvertent or deliberate interference from traffic using the same radio band.³⁰ To demonstrate this vulnerability, place a laptop with an 802.11b NIC next to a microwave oven.³¹ As both devices generally use the 2.4-GHz band, signal degradation on the 802.11b network is likely to occur any time the microwave is in operation. An attacker could use the same principle to disable or degrade an 802.11b network by broadcasting traffic on the same frequency as the network.

²⁸ Found at http://www.wi2600.org/mediawhore/nf0/wireless/ssid_defaults/.

²⁹ See <http://standards.ieee.org/regauth/oui/index.shtml>.

³⁰ Richard A. Stanley, "Wireless LAN Risks and Vulnerabilities," Information Systems Control Journal, Volume 2, 2002. Available at <http://www.isaca.org/wirelesswhitepaper.pdf>.

³¹ Ad Kameran and Nedim Erkoçevic, "Microwave Oven Interference on Wireless LANs Operating in the 2.4GHz ISM Band," Accessed on June 8, 2002, at <http://www.devx.com/wireless/articles/Bluetooth/whitepapers/1a6900.pdf>.



All 802.11x networks operate at 2.4- or 5.0-GHz, depending upon the 802.11x hardware being used. Unfortunately, practical defenses against such an attack are limited, other than to avoid using 802.11x networks for critical components of the network infrastructure. Use wireless access as a convenient means of connecting to the network, but also have the option of using a hard-wired connection if the WLAN goes down or is compromised.



SECURING WLANS

WLAN Hardening Checklist

DO NOT RELY ON WEP FOR ENCRYPTION

WEP is insecure. This is not a revelation but a reality. It was not designed to provide a complete security solution for wireless networks, only a level of privacy equivalent to wired LANs.³² Do not view WEP as a security solution. Instead, use it in combination with encryption standards for other insecure networks such as virtual private networks. Use application-level security (i.e., PGP) for sensitive data.

SEGREGATE WIRELESS NETWORKS

WLANs present different security challenges than wired LANs. WLANs are generally not as secure. Do not allow traffic between the two environments to exist in a trusted environment. Place internal firewalls between LANs and WLANs, and require authentication before traffic passes between the two.

DO NOT USE A DESCRIPTIVE NAME FOR SSID OR ACCESS POINT

The SSID and optional AP names are not encrypted in the header of 802.11x data packets. Even when WEP is enabled, WLAN scanners could easily obtain these items.³³ Providing descriptive names, such as the company name, makes a hacker's job much easier because identifying the source of the signal becomes trivial. iDEFENSE Labs encountered one situation while war driving where a company had used its web site address as the name of its AP. Two clicks into its website yielded not only its address but driving directions to its office as well! To top it off, the company was not using WEP. Talk about handing over the keys to the kingdom.

HARD CODE MAC ADDRESSES THAT CAN USE THE AP

Many manufacturers of APs provide the ability to identify the MAC addresses of network cards permitted to use the AP. An inventory of authorized cards must be maintained, but the maintenance effort provides a reasonable security enhancement. While a hacker could still identify APs and passively sniff traffic, they would not be able to connect to hosts on the network without spoofing a legitimate MAC address.

³² Stuart J. Kerry, "Chair of IEEE 802.11 Responds to WEP Security Flaws," Feb. 15, 2001. Available at <http://slashdot.org/articles/01/02/15/1745204.shtm>.

³³ See Appendix A: Auditing Tools on page 30.



CHANGE ENCRYPTION KEYS

Changing the encryption keys periodically would not prevent the compromise of WEP keys because an attacker could crack the keys within a matter of hours. However, changing the encryption keys would ensure that a compromised network does not remain compromised indefinitely. A hacker could always crack the encryption key a second time, but changing keys provides some disincentive to the hacker. Unfortunately, changing keys could be time consuming as each AP and every wireless NIC using the AP would require manual updates. Implementing this recommendation depends upon finding a balance between security and convenience — a common issue in the security world. Fortunately, vendors are already introducing proprietary solutions to automate key management and the 802.11i Task Group is working to establish standards.³⁴

DISABLE BEACON PACKETS

Some APs provide an option that prevents the AP from advertising its presence via periodic beacon packets. These APs require the wireless network cards to use the same SSID before they respond to traffic.³⁵ This feature prevents hackers from using some of the WLAN scanning tools listed in Appendix A: Auditing Tools.

LOCATE APs CENTRALLY

When creating the layout of APs within an office, factor in their broadcast range. Ensure adequate signals reach all necessary areas within the building, but do not unnecessarily broadcast traffic into the parking lot or a neighbor's office.

CHANGE DEFAULT PASSWORDS/IP ADDRESSES

Most APs have a built in web server that provides a console for administration. While convenient, this could also allow an attacker on either a wireless or hard-wired network to access the AP administration console by opening a web browser and pointing it to the IP address assigned to the AP. Change the IP address and authentication credentials for the AP. Obtaining the default IP address and authentication credentials is as simple as downloading support documentation from the vendor web site. WLAN scanning tools, such as NetStumbler, identify hardware vendors by comparing broadcast MAC address to listings published by the IEEE.³⁶ If an attacker could access the AP administration console and the default password had not been changed, the attacker could then disable any security settings or cause a denial of service by changing settings such as the channel or SSID. This would prevent wireless clients from using the access point.

AVOID WEP WEAK KEYS

Vendors are beginning to provide firmware upgrades for 802.11b products that avoid the use of IVs that result in the so-called interesting packets (aka weak keys) targeted by

³⁴ For an interesting paper on how automated key management can be piggy-backed on DHCP, read "A Transparent Key Management Scheme for Wireless LANs Using DHCP" by Shankar, Arbaugh and Zhang. Available at <http://www.hpl.hp.com/techreports/2001/HPL-2001-227.html>.

³⁵ "Cisco Aironet Access Point Broadcast SSID," Dec. 6, 2001. Available at <http://xforce.iss.net/static/6287.php>.

³⁶ See <http://standards.ieee.org/regauth/oui/index.shtml>.



tools such as AirSnort. This workaround will only be effective if all wireless products on the network are upgraded as the transmitting station always determines the IV that is used. The firmware update for ORiNOCO PC Cards v8.10 – Winter 2002 release is an example of such an upgrade.³⁷

DO NOT USE DHCP ON WLANS

To access hosts at a targeted site, a hacker would need to obtain a valid IP address and subnet mask on the WLAN. Identifying valid IP addresses on a network does not require significant effort, but why make the hacker's job easier than it needs to be? Without DHCP, identifying IP addresses requires passively sniffing traffic and reviewing the captured packets. A hacker could also use brute force, as the number of private address ranges is limited. In short, a hacker could identify valid addresses and subnet masks whether DHCP is present or not, but static IP addresses are one more deterrent that may cause a hacker to go next door and find a less secure network.

IDENTIFY ROGUE ACCESS POINTS

In large companies, end users may cause concern by deploying their own hardware or software. Just as an enterprising employee may install a modem to allow for remote access from home, the employee may also want to add a wireless network for web surfing convenience. The low cost of the necessary hardware and relative ease of installation make this a significant concern for network administrators. The only surefire way to identify rogue access points is to look for them. Grab a laptop, a wireless NIC, and a WLAN scanner and start war walking.

The Future of 802.11x Security

The IEEE has established the 802.11i task group to develop standards to address the security issues in 802.11x. As of this writing, no standards have been approved, but proposals include a number of enhancements to address the issues of weak encryption, key management authentication and access control.

TKIP

The 802.11i draft promotes the use of Temporal Key Integrity Protocol (TKIP) to strengthen the weak keys used by WEP. TKIP is an effort by the IEEE to engineer a solution to strengthen the security of 802.11x networks while remaining backward compatible with existing hardware. The IEEE would accomplish this with the distribution of software/firmware upgrades that would add the following new algorithms to the WEP protocol:³⁸

- Message Integrity Code (MIC) – to prevent forged packets
- New IV sequencing discipline – to prevent replay attacks
- Per-packets key mixing function – to add complexity to the correlation between IVs and the per-packet keys with which they are used

³⁷ Found at <http://www.orinocowireless.com/template.html?section=m52&envelope=90&page=3267>.

³⁸ Jesse Walker, "802.11 Security Series – Part II: The Temporal Key Integrity Protocol (TKIP)," accessed on July 4, 2002, at http://cedar.intel.com/media/pdf/security/80211_part2.pdf.



- Re-keying mechanism – to prevent key reuse

AES

TKIP is only an intermediate solution. Ultimately, WEP will need to be replaced entirely. Due to the architectural constraints of existing 802.11x hardware, replacing WEP with a new encryption algorithm would require replacing the hardware — a costly venture for companies that have recently deployed 802.11x networks. The 802.11i task group is considering the use of the Advanced Encryption Standard (AES) as WEP's replacement. In December 2001, the US federal government selected AES to replace the Data Encryption Standard (DES) as the encryption standard used by federal agencies.³⁹

802.1X

The 802.11i task group is attempting to leverage the 802.1X standard to add authentication controls to wireless networks. 802.1X defines Extensible Authentication Protocol (EAP) over LANs (EAPOL), which is used to authenticate clients as they join the network.⁴⁰ The inclusion on 802.1X would prevent hackers from connecting to 802.11x networks simply by determining the channel and SSID used by the network and identifying a legitimate IP address by passively sniffing network traffic.

TOO LITTLE TOO LATE

While the enhancements proposed by the 802.11i task group could significantly enhance the security of 802.11x networks, the task group has been slow to approve the new standard. Several delays have caused impatient vendors to implement proprietary solutions. Vendors may not be willing to replace these proprietary fixes with the 802.11i standard once it becomes available.

Other Security Concerns

No matter how much money is invested in technology to make WLANs secure, they will never be impenetrable. WLANs are like hard-wired networks in that they were designed to facilitate communication. Access to a WLAN would remain possible whether it is authorized or not with the appropriate credentials and a connection to the network. Security planning cannot cease once the architecture is in place. Physical security and end-user awareness need to be revisited whenever WLANs are implemented.

PHYSICAL SECURITY

WLANs require a fresh look at physical security controls. Best practices for securing networks in the wired world no longer apply. Adequate physical security once involved restricting entry to a building and requiring that guests be escorted throughout the premises. Hackers no longer need access to a building to access a network. Depending upon the strength of the signal broadcast and the hacker's hardware, the hacker may be able sniff traffic several hundred yards away from the access point. Someone in a car in the parking lot could sniff traffic. The hacker may not need to be on company premises at all.

³⁹ iDEFENSE Intelligence Report ID# 106452, Dec. 6, 2002.

⁴⁰ Paul Goransson, "802.1X provides user authentication," March 25, 2002. Available at <http://www.nwfusion.com/news/tech/2002/0325tech.html>.



While war driving, a hacker could travel at 50 mph down a major road and still detect APs. This fact must be taken into consideration when deciding on the implementation of wireless technologies. A company with a campus-like environment needs to ensure that security guards and employees are on the lookout for individuals that seem to be loitering on the premises. A company in Manhattan would not have this luxury. No matter how big the office in a crowded urban area, someone outside the property could obtain access to a signal. WLANs may not be a viable option without adequate security of the signal through encryption.

Physical security does not only apply to unauthorized persons on company property. Escorting a visitor on the premises would do nothing to prevent someone from identifying the presence of APs and passively sniffing traffic. An escorted individual could carry a laptop computer or handheld silently auditing the company network. Add GPS to the equation, and someone could walk away with a detailed map of exactly where different APs are located throughout the building. Armed with this knowledge, the visitor could return at a later time and set up shop in a public location in the building or in the parking lot and continue hacking into the network.

Sound far-fetched? Tools such as NetStumbler are freely available, run on a laptop and support inexpensive GPS receivers. Better yet, MiniStumbler is a scaled down version of NetStumbler for the PocketPC platform.⁴¹ While MiniStumbler does not provide all of the functionality of its older brother, it is a WLAN auditing tool that fits in a pocket. Berkeley Varitronics Systems Inc. produces a handheld appliance, known as Grasshopper, designed for troubleshooting/auditing WLANs.⁴² Now that such tools are available, requiring individuals to leave their bags at the front desk does not prevent security breaches because someone could easily conceal these appliances in a jacket pocket. Does that mean that everyone needs to be thoroughly searched before entering the building? That is for security managers to decide.

End-User Awareness

The passive sniffing of traffic on a network may be undetectable without human diligence. Anti-sniffer software could detect LAN-based network sniffers, but no method exists to identify a passive WLAN card running in promiscuous mode. Therefore, make employees aware of the risks of deploying WLANs. During security briefings, instruct employees to be mindful of suspicious individuals loitering on the premises. Encourage employees to look out not only for suspicious individuals, but also computer hardware. Capturing traffic and cracking keys takes time. An attacker may plant a laptop computer with a wireless network card and return to pick it up after the damage has been done. If adequate inventory records exist, employees may determine if hardware belongs where it is found.

⁴¹ Found at <http://www.NetStumbler.org/download.php?op=getit&lid=21>.

⁴² See <http://www.bvsystems.com/Products/WLAN/Grasshopper/grasshopper.htm> for information.



CONCLUSION

Wireless networks offer the convenience once only dreamed of but could become a security nightmare if not used appropriately. In the future, enhanced encryption schemes may make wireless networking more secure than its wired counterpart; however, that is not yet the case. Only use WLANs in situations where sensitive information does not traverse the network. WLANs are generally insecure and must be treated as such. Companies use firewalls for the same reason that people do not leave their front doors open; no one trusts individuals they do not know. If unknown individuals could access a wireless network, it should not be a trusted network. Use perimeter security controls to restrict traffic between a WLAN and the rest of the corporate network.

iDEFENSE Labs does not, however, suggest that WLANs should not be deployed. What could be better than going outside to complete a report on a hectic workday and maintaining Internet access to do research?

WLANs certainly have their place; the author has one in his home and does not want to contemplate life without it. Is it possible that someone could invade his privacy? Yes. Is that a risk that he is willing to take? Yes, but someone responsible for security at a bank may not feel the same.

ComputerWorld reported that airlines were using 802.11b networks at several airports for bag matching and curbside check-in without using WEP.⁴³ This is truly a scary situation. What happens if someone is able to place a bag on a flight and delete any record of its existence? The decision to implement a WLAN in a corporate setting must involve some form of risk analysis.

If after assessing the risks you decide that a wireless network is appropriate, be sure and employ the security features made available. WEP may not be perfect but it does provide a reasonable deterrent. If an attacker is simply looking for free Internet access or a testing ground for his new wireless card he'll go down the street and find an easier target. The greatest weakness with wireless security is not the technical shortcomings but out of the box insecure installations. Once again, the human factor is the weakest link.

⁴³ Bob Brewin, Dan Verton and Jennifer DiSabatino. "Wireless LANs: Trouble in the Air," *ComputerWorld*, Jan. 14, 2002. Available at <http://www.computerworld.com/securitytopics/security/story/0,10801,67344,00.html>.



ACKNOWLEDGEMENTS

Thanks to the following individuals for their efforts:

- David Endler, iDEFENSE Inc.
- Yong-Gon Chon, TruSecure Corp.
- Narendar Shankar, University of Maryland
- Michael Cheek and Andrew Schmidt, iDEFENSE Inc.
- Mickey McCarter, Newspoint



APPENDIX A: AUDITING TOOLS

WLAN Scanners

Name	Platform	Vendor website
NetStumbler	Windows	http://www.NetStumbler.org/
Dstumbler	BSD	http://www.dachb0den.com/projects/dstumbler.html
MacStumbler	Macintosh	http://homepage.mac.com/macstumbler/
MiniStumbler	Pocket PC	http://www.NetStumbler.org/download.php?op=getit&lid=21
SSIDSniff	Unix	http://www.bastard.net/~kos/wifi/
Airosniff	Unix	http://gravitino.net/~bind/code/airosniff/
AP Scanner	Macintosh	http://homepage.mac.com/typexi/Personal1.html
wavemon	Linux	http://www.jm-music.de/projects.html
WLAN Expert	Windows	http://www.vector.kharkov.ua/download/WLAN/wlanexpert.zip
wavelan-tools	Linux	http://sourceforge.net/projects/wavelan-tools/
Kismet	Linux, iPaq, Zaurus	http://www.kismetwireless.net/
AiroPeek	Windows	http://www.wildpackets.com/products/airopeek/
Sniffer Wireless	Windows	http://www.sniffer.com/products/sniffer-wireless/
THC-WarDrive	Linux	http://www.thehackerschoice.com/download.php?t=r&d=wardrive-2.3.tar.gz
APSniff	Windows	http://www.bretmounet.com/ApSniff/
Wellenreiter	Linux	http://www.remote-exploit.org/
PrismStumbler	Linux	http://prismstumbler.sourceforge.net/
AirTraf	Linux	http://airtraf.sourceforge.net/

WLAN Sniffers

Name	Platform	Vendor website
Mognet	Java VM	http://chocobospore.org/mognet/
Kismet	Linux, iPaq, Zaurus	http://www.kismetwireless.net/
Ethereal	Unix, Windows	http://www.ethereal.com/
TCPDump	Unix	http://www.tcpdump.org/
Prismdump	Unix	http://developer.axis.com/download/tools/
prism2dump	BSD	http://www.dachb0den.com/projects/prism2dump.html
AiroPeek	Windows	http://www.wildpackets.com/products/airopeek/
Sniffer Wireless	Windows	http://www.sniffer.com/products/sniffer-wireless/

WEP Key Crackers

Name	Platform	Vendor website
WEPCracker	Perl	http://sourceforge.net/projects/wepcrack/
AirSnort	Linux	http://www.be-secure.com/airsnort.html
AirSnort for BSD	BSD	http://www.dachb0den.com/projects/bsd-airsnort.html



Other

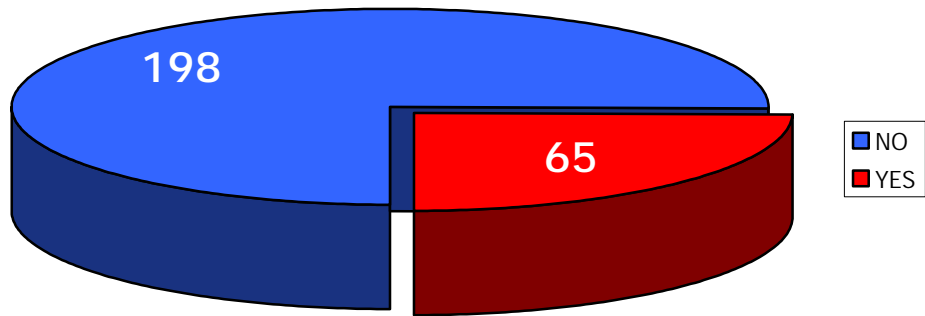
Name	Platform	Vendor website
APTtools	Windows, Unix	http://aptools.sourceforge.net/
Note: Identify APs based on MAC addresses by querying routers and switches		
Wireless Tools	Linux	http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html
Note: Tools allowing for the manipulation of Wireless Extensions		
THC-Rut	Unix	http://www.thehackerschoice.com/download.php?t=r&d=thcrut-0.1.tar.gz
Note: Local network discovery tool developed to brute force its way into WLAN access points		
AirMagnet	PocketPC	http://www.airmagnet.com/products.htm
Note: This commercial product is a wireless vulnerability scanner that attempts to identify rogue access points, denial of service attacks, unencrypted traffic, default SSIDs and MAC address spoofing, along with functionality to troubleshoot connectivity issues.		



APPENDIX B: STATISTICS

War Driving and Walking

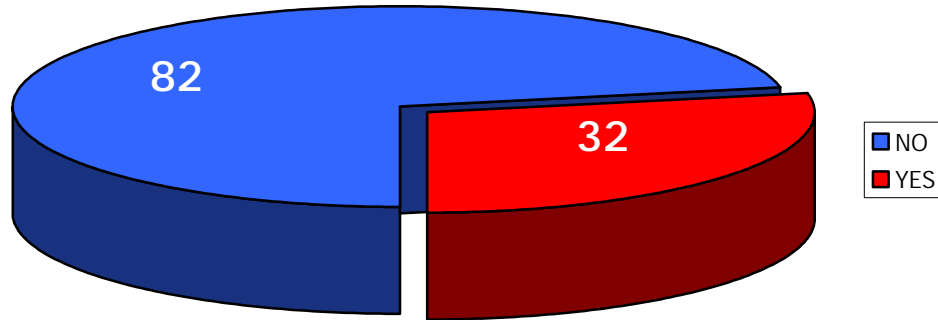
WEP-Enabled (Manhattan)



APs	Number	Percentage
WEP not enabled	198	75%
WEP enabled	65	25%
Total	263	100%



WEP-Enabled (Northern Virginia)



APs	Number	Percentage
WEP not enabled	82	72%
WEP enabled	32	28%
Total	114	100%



APPENDIX C: REFERENCES

- Arbaugh, William, Narendar Shankar and Justin Wan. "Your 802.11 Wireless Network has No Clothes." Available at <http://downloads.securityfocus.com/library/wireless.pdf>.
- Azrael's Dominion, XOR Encryption. Available at <http://www.angelfire.com/ct/azraelsdominion/xorencryption.html>.
- Borisov, Nikita, Ian Goldberg and David Wagner. "Intercepting Mobile Communications: The Insecurity of 802.11." Available at <http://www.isaac.cs.berkeley.edu/isaac/wep-draft.pdf>.
- Brewin, Bob, Dan Verton and Jennifer DiSabatino. "Wireless LANs: Trouble in the Air." ComputerWorld. Available at <http://www.computerworld.com/securitytopics/security/story/0,10801,67344,00.html>.
- Brown, Bruce. "Wireless Standards Up in the Air." Available at <http://www.extremetech.com/article2/0,3973,9164,00.asp>.
- "Cisco Aironet Access Point Broadcast SSID." Available at <http://xforce.iss.net/static/6287.php>.
- Cyclic Redundancy Check Polynomials Tutorial. Available at <http://www.cee.hw.ac.uk/~pjbk/nets/crcutorial.html>.
- Fluhrer, Scott, Itsik Mantin and Adi Shamir. "Weaknesses in the Key Scheduling Algorithm of RC4." Available at http://online.securityfocus.com/data/library/rc4_ksaproc.pdf.
- Goransson, Paul. "802.1X provides user authentication." Available at <http://www.nwfusion.com/news/tech/2002/0325tech.html>.
- Kamerman, Ad and Nedim Erkoçevic. "Microwave Oven Interference on Wireless LANs Operating in the 2.4 GHz ISM Band." Available at <http://www.devx.com/wireless/articles/Bluetooth/whitepapers/1a6900.pdf>.
- Kerry, Stuart J. "Chair of IEEE 802.11 Responds to WEP Security Flaws." Available at <http://slashdot.org/articles/01/02/15/1745204.shtml>.
- Noble, Ivan. "Wireless networks wide open." Available at http://news.bbc.co.uk/1/hi/english/sci/tech/newsid_1638000/1638920.stm.
- Schenk, Rob, Andrew Garcia and Russ Iwanchuk. "Wireless LAN Deployment and Security Basics." Available at <http://www.extremetech.com/article/0,3396,s=1034&a=13521,00.asp>.
- Stanley, Richard A. "Wireless LAN Risks and Vulnerabilities." Information Systems Control Journal, Volume 2 (2002). Available at <http://www.isaca.org/wirelesswhitepaper.pdf>.
- Stubblefield, Adam, John Ioannidis and Aviel D. Rubin. "Using the Fluhrer, Mantin and Shamir Attack to Break WEP." Available at http://www.cs.rice.edu/~astubble/wep/wep_attack.pdf.
- University of Berkeley FAQ. Available at <http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html>.
- Walker, Jesse, "802.11 Security Series – Part II: The Temporal Key Integrity Protocol (TKIP)." Available at http://cedar.intel.com/media/pdf/security/80211_part2.pdf.



APPENDIX D: IEEE TASK GROUPS

802.11 Task Group	Name
802.11a	High-speed Physical Layer in the 5GHz Band
IEEE description	The family of specifications for wireless, Ethernet local area networks in 5-gigahertz bandwidth space.
URL	http://standards.ieee.org/getieee802/download/802.11a-1999.pdf
802.11b	Higher-Speed Physical Layer Extension in the 2.4GHz Band
IEEE description	The family of specifications for wireless, Ethernet local area networks in 2.4-gigahertz bandwidth space.
URL	http://standards.ieee.org/getieee802/download/802.11b-1999.pdf
802.11d	Specification for Operation in Additional Regulatory Domains
IEEE description	Define the physical layer requirements (channelization, hopping patterns, new values for current MIB attributes, and other requirements to extend the operation of 802.11 WLANs to new regulatory domains, or countries).
URL	http://standards.ieee.org/getieee802/download/802.11d-2001.pdf
802.11e	MAC Enhancements for Quality of Service
IEEE description	Enhance the current 802.11 MAC to expand support for applications with Quality of Service requirements, and in the capabilities and efficiency of the protocol.
URL	http://grouper.ieee.org/groups/802/11/Reports/tge_update.htm
802.11f	Recommended Practice for Inter Access Point Protocol
IEEE description	Develop recommended practices for an Inter-Access Point Protocol (IAPP), which provides the necessary capabilities to achieve multi-vendor Access Point interoperability across a Distribution System supporting IEEE P802.11 Wireless LAN Links.
URL	http://grouper.ieee.org/groups/802/11/Reports/tgf_update.htm
802.11g	Standard for Higher Rate (20+ Mbps) Extensions in the 2.4GHz Band
IEEE description	Develop a higher speed(s) physical layer extension to the 802.11b standard. The new standard shall be compatible with the IEEE 802.11 MAC. The maximum physical layer data rate targeted by this project shall be at least 20 Mbit/s. The new extension shall implement all mandatory portions of the IEEE 802.11b physical layer standard.
URL	http://grouper.ieee.org/groups/802/11/Reports/tgg_update.htm
802.11h	SMA - Spectrum Managed 802.11a
IEEE description	Enhance the 802.11 Medium Access Control (MAC) standard and 802.11a High Speed Physical Layer (PHY) in the 5GHz Band supplement to the standard; to add indoor and outdoor channel selection for 5GHz license exempt bands in Europe; and to enhance channel energy measurement and reporting mechanisms to improve spectrum and transmit power management (per CEPT and subsequent EU committee or body ruling incorporating CEPT Recommendation ERC 99/23).
URL	http://grouper.ieee.org/groups/802/11/Reports/tgh_update.htm
802.11i	MAC Enhancements for Enhanced Security
IEEE description	Enhance the current 802.11 MAC to provide improvements in security.
URL	http://grouper.ieee.org/groups/802/11/Reports/tgi_update.htm

