Global Information Assurance Certification Paper

Insecurities of WEP and Securing the Wireless Networks

Nuruddin Mohd. Alamgir

Prepared by: Nuruddin Alamgir
Version: 1.3
Certification: Security Essentials, GCSE
Date: 5th June 2002

Table of Content

Abstract

Wireless LANs (WLAN) have grown tremendously since its introduction in 1997. Cheap equipment cost and easy maintainability have attracted many organizations and home users to implement WLANs. Unfortunately WLANs are inherently insecure. WLANs do not provide strong security mechanisms to protect data availability, confidentiality and integrity. WEP (Wired Equivalent Privacy) is the only security mechanism available in current WLAN standard. WEP uses symmetric RC4 cipher algorithm to encrypt data transmitted through the WLAN.

It has been proven that WEP can be easily cracked with-in a short time. Moreover, most network administrators do not enable WEP on the WLANs, which leaves wireless networks highly insecure and susceptible to eavesdropping and vulnerable to unauthorized access. This paper discusses the insecurities in WEP and how a WLAN can be secured by using IPSec. Step by step instructions are provided on how to configure IPSec on an OpenBSD machine which will act as a gateway to the WLAN. A complete list of measures that can be taken to secure a WLAN is also available at the end of the paper.

Introduction to WEP

Wired Equivalent Privacy (WEP) is the encryption protocol provided by the 802.11 standard. Security goals of WEP are confidentiality, data integrity, access control. The main objective of WEP is to protect data transmitted within a WLAN from eavesdropping. WEP uses the RC4 encryption algorithm to encrypt the data. RC4 is a symmetric algorithm. A symmetric algorithm relies on a single shared key that is used at one end to encrypt plain text into cipher text, and decrypt it at the other end. The sender and the receiver use the same secret key.

RC4 is a variable key size stream cipher consisting of two parts:

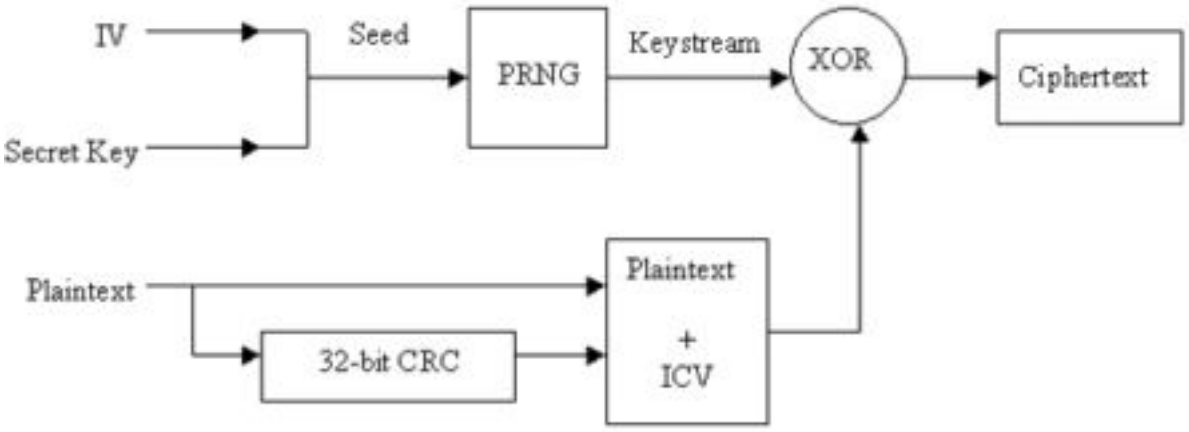
1. A key scheduling algorithm (KSA),
2. Psuedo Random generation algorithm

RC4 operates by expanding a short key into an infinite pseudo-random key stream. WEP uses the RC4 encryption with either a 64bit or 128bit 'unique' per-frame key. The per-frame key is a combination of a pre-shared key either 40bits or 104bits and a pseudo random 24bit initialization vector. The IV is randomized for every packet to ensure that every packet has a different RC4 key. The IV is transmitted in clear text.

The following steps are used to encrypt a WLAN frame using WEP.

- A 32bit CRC is done on the plaintext to derive the Integrity Check Value (ICV). The ICV is then concatinated to the plain text.
- The per packet IV is combined with the shared key and is passed through a pseudo-random number generator (PRNG) to produce a key stream. The length of the key stream is equal to the text to be encrypted.
- The key stream is then XORed (Exclusive OR) with the plain text frame to generate a ciphertext frame.

The following diagram shows how the IV, secret key, ICV and the plain text (data) is used to create the ciphertext. The IV is then pre-pended (unencrypted) to the ciphertext frame as the WEP data frame before transmission.



WEP Encryption Process, source: [7]

Every transmitted frame contains the IV in plaintext. The receiver uses the IV with the shared key to generate the key stream used to encrypt the data. The receiver then XORs the ciphertext with the key stream to generate the plain text.

WEP Insecurities

WEP has defenses against data integrity and confidentiality. It uses an integrity check value (ICV) to ensure that the data has not been modified during transmission. An Initialization Vector (IV) is used to augment the shared key and produce a different RC4 key stream for each packet. This ensures that two packets are 'never' encrypted with the same key stream. Unfortunately both of these critical measures are implemented incorrectly, resulting in serious security vulnerabilities in WEP.

The integrity check value is a CRC32 checksum. WEP uses the Integrity Check Value to verify that the decoded message matches the one that was sent. It has been demonstrated [1] how an intercepted message could be manipulated in such a way that the resulting checksum would be same even though the decrypted message would say something completely different from what was originally intended [2]. The attacker this way can inject rogue packets to the network. For example, the attacker can modify a legitimate command (i.e. shell command) sent from one host to another to a malicious command. The receiving host will accept the packet with the malicious command and will execute it.

The IV in WEP is only 24bits long which guarantees the reuse of the same IV hence reuse of the same key stream (unless the shared key is changed very frequently, which is rare in current implementations of WLANs due to poor key management mechanism). As the IV is sent in clear text, the attacker can actually check if two packets are encrypted with the same key stream. If an attacker flips a bit in the cipher text the corresponding plain text also gets flipped. By doing statistical analysis on two cipher text encrypted with the same key stream, the attacker can recover the plain text, and hence the key stream that was used to encrypt the data.

The key stream can also be obtained by doing a XOR operation between the cipher text and the plain text. If the attacker is able to inject traffic into the WLAN and then able to sniff the encrypted version of the traffic, he will have the plain text and the equivalent cipher text. He can then use the following formula to extract the key stream that was used to encrypt the traffic injected by him.

Cipher text XOR Plain text = key stream

Besides being susceptible to eavesdropping, most WEP networks have no authentication mechanism. This leaves endless possibilities for the attackers. Most of the current available wireless cards can sniff network traffic. All of the prism2 chipset cards allow sniffing of the wireless networks. All the attacker needs is a laptop, a PCMCIA wireless card and an antenna for scanning and he could penetrate into most (if not all) private networks. By having a WLAN directly connected to the internal Wired LAN one is exposing the whole internal network to such attacks.

Attack Scenarios

In a practical attack scenario all an attacker needs is a handy device with a wireless Card, preferably a laptop with a PCMCIA wireless card that can be put into promiscuous mode for network traffic sniffing. Most of the time the attacker makes an external antenna for more efficient scanning of access points, the PCMCIA cards like Dlink DWL 650 allows an external antenna to be added. There are some efficient external antennas made out of 'Pringles Can' which are

much more efficient than most commercially available antennas [3]. Please refer to reference material [3] for step by step instructions on how to create and external antenna using Pringles Can.

Pringles Antenna to scan Wireless LANS, source: [3]

Equipped with these tools the attacker uses to scan for access points once the network is identified the attacker then runs a DHCP (Dynamic Host Configuration Protocol) client program requesting the access point to allocate an IP address and most of the DHCP servers do not authenticate and assign an IP address to the requesting host. Once the attacker has an internal IP he becomes part of the internal wireless network. If the wireless access point does not have WEP enabled, the attacker can, most of the time, take control of the internal network. He can scan the internal network for vulnerable hosts, get unauthorized access to sensitive corporate data or sniff traffic; the possibilities are endless. But if the access point has WEP enabled the attacker can expose insecurities in RC4 to crack the shared secret key by passively sniffing the network and once the keys are retrieved the attacker can decrypt the network traffic.

WLAN Security Recommendations

Due to inherent flaw in the WLAN implementation additional measures are needed to make it secure. The following measures can be adopted to make a WLAN more secure and resistant to eavesdropping and attack. Each of these recommendations provide additional layer of security for WLAN. One single measure can not completely secure the WLAN; hence one should not rely solely on one of the recommendations for complete WLAN security but should try to accommodate as many of these measures as possible to provide additional layer of security.

Enable WEP

WEP cannot protect against eavesdropping from dedicated attackers, but can certainly discourage unskilled or impatient hackers. Majority of attackers are people just poking around. They do not have the time or patience to crack WEP. And some of them might not be skilled enough to crack WEP even if they have the time. A WLAN without WEP is more susceptible to attack then a WLAN with WEP. If an attacker finds two WLANs, one with WEP and one without WEP, he will certainly attack the WLAN without WEP first. Hence it is highly recommended that WEP must be enabled in all WLANs.

Implement VPN Based Solution

The only real way to secure WLAN traffic is to tunnel it through VPN. Any traffic in the WLAN must be tunneled through the VPN. Once the data is encrypted using IPSec, it will be of no use to the attacker who is sniffing traffic on the WLAN. Even though IPSec is recommended in this paper, other technologies exist that can secure the WLAN traffic, for example FreesWAN for Linux, Microsoft PPTP server, SSH etc. [14]. The next section describes how an OpenBSD machine can be setup with IPSec to act as a Gateway to the WLAN.

SSID – Service Set Identifier

It is highly recommended to change the default SSID of the Access Point. Most access points broadcasts their SSID and does not allow this feature to be turned off. Cisco access points allows SSID broadcast to be turned off. The default SSID must be changed and the access points must be configured not to broadcast their SSID. Changing the default SSID keeps other users in the area with the default SSID on their equipment from getting on ones network by accident.

Disable DHCP

Dynamic Host Configuration Protocol (DHCP) assigns IP addresses to clients that requests for one. DHCP servers do not usually authenticate the client and

leases an IP address to who ever ask for it. This makes it so much easier for an attacker to get an internal IP address. If his wireless device is within the signal range, he can just run a DHCP client on the device and get an internal IP address.

It is recommended to disable DHCP servers on the WLAN and rather assign static IP addresses to the devices in the WLAN. The attacker then have to determine the IP range for the WLAN and assign an IP address to his wireless device. Moreover, it is also recommended not to use default IP ranges in the WLAN. This will make it more difficult for the attacker to assign his device an IP address, and could discourage him from attacking the network.

Enable MAC Filtering

Access Points allow MAC address filtering. They can maintain a list of MAC addresses allowed to associate with the access point. Network cards, whose MAC address is not there in the list are not allowed to use the access point. It is true that MAC addresses can be changed or spoofed, but the attacker must know a valid MAC address before he can use the network. MAC address filtering must be enabled in the access point to discourage attackers.

Authentication

Many access points have the ability to authenticate via RADIUS or similar services [14]. The use of authentication server must be considered to control who can use the access point by maintaining a user list on a server.

Reduce Access Point Signals

The location of access point must be carefully chosen to minimize signal leakage. Access points should not be placed near the window as the signal from the access point can travel the other side of the window where an attacker can place a wireless device and sniff the network traffic or try to get unauthorized access to the network. The access points should be placed towards the center of the room to minimize the chance of signal leaking out of the room. Directional antennas can be used to focus the signal towards a particular area. Again, this will only discourage a casual attacker. A more determined attacker can boost the signal of his device by use of antennas (e.g. Pringles antennas) to connect to the access point.

Location of Access Point

The WLAN should be treated as an un-trusted domain and a firewall must be placed between the WLAN and the internal wired LAN. If a firewall already exists, then the WLAN must be placed in a separate DMZ (De-Militarized Zone). Filters must be applied to the firewall to deny any traffic originating from the WLAN and

destined towards the internal network unless explicitly allowed by a firewall rule. Firewall logs must be regularly checked to analyzed traffic coming from the WLAN.

Location of Resources

Resources that do not require wireless access must be placed in the wired LAN and the firewall should block any request coming for these resources from the WLAN. Only resources that require wireless access should be placed in the WLAN or should be accessible to the WLAN.

Periodically Audit WLAN

Once implemented and secured, the WLAN must be periodically audited to identify any vulnerability i.e. rogue access point, signal leakage etc. and take necessary action to eliminate the vulnerability immediately. New vulnerabilities are discovered all the time; hence it's important that the WLAN is audited regularly for the newly discovered vulnerabilities.

Monitoring the Network

The network must be constantly monitored to detect attacks and strange behaviors. An Intrusion Detection System (IDS) should be deployed in the WLAN segment to monitor the network for intrusions. In case of any incident, the IDS logs will be very handy for forensics purposes. Log files created by various monitoring systems must be regularly reviewed to detect any possible attacks.

IPSec

Virtual Private Networks (VPN) makes it possible for a user on an un-trusted network to connect to a private network in a secure fashion. Internet Protocol Security (IPSec) is the most widely used mechanism for securing VPN traffic. IPSec is a pair of protocols, ESP (Encapsulating Security Payload) and AH (Authentication Header), which provide security services for IP datagrams. IPSec can use multiple algorithms for encrypting data, keyed hash algorithms for authenticating packets, and digital certificates for validating public keys [13]. Three principal security elements of IPSec are Authentication Header (AH), Encapsulation Security Payload (ESP) and Internet Key Exchange (IKE).

Authentication Header (AH)

AH adds authentication information to the IP data to provide authentication and data integrity. This ensures that the data is available to only authorized users. AH also ensures that the transmitted data is not altered on the route (data integrity). Message Digest Algorithm 5 (MD5) and Secure Hashing Algorithm (SHA) are used as authentication techniques.

Encapsulation Security Payload (ESP)

Data confidentiality is achieved through ESP. ESP encrypts all data and part of ESP header using encryption algorithms such as Triple-DES (3DES), blowfish etc.

Internet Key Exchange (IKE)

IKE are the management protocols that are used to negotiate the cryptographic algorithm choices to be employed by the AH and ESP. Keys are maintained, exchanged, and verified using these protocols [13].

OpenBSD and IPSec as WLAN Gateway

OpenBSD is one the most secure operating systems which never had a remote hole in the default install for the past 4 years and IPSec is very well implemented in OpenBSD.

The following sections describe how IPSec can be enabled on OpenBSD. Please note that to be able to run the configuration commands, the user must logon as root on the OpenBSD machine.

Turning on IP forwarding through sysctl

IP forwarding must be enabled in the kernel so that packets can be routed through the interfaces. The following `sysctl` command enables IP forwarding in the kernel.

```
root@ferrari#sysctl net.inet.ip.forwarding=1
```

Turning on the IPSec protocol inside the kernel through sysctl

The next step is to turn on IPSec protocol inside the kernel. This can be achieved in two ways. First by using the `sysctl` command as shown below

```
root@ferrari#sysctl -w net.inet.esp.enable=1

root@ferrari#sysctl -w net.inet.ah.enable=1
```

And secondly by modifying the following lines in /etc/sysctl.conf to look as shown below.

```
net.inet.esp.enable=1      # 1=Enable the ESP IPsec protocol

net.inet.ah.enable=1       # 1=Enable the AH IPsec protocol
```

This will enable ESP and AH on the OpenBSD machine. Please note that as of OpenBSD 3.0 ESP and AH is enabled by default, hence the users of OpenBSD 3.0 can skip this step.

Configuration of network interface to set its IP

The default network is made using 10.0.0.x series. The following command is used to assign the IP address 10.0.0.1 to the network interface card of the OpenBSD machine.

```
root@ferrari#ifconfig lx0 10.0.0.1
```

Key Generation

Two keys are required to be shared between the server and client, one for encryption and one for authentication. Two random 160 bit keys must be generated. The key for encryption is generated using blowfish and the key for authentication is generated using SHA1. Different algorithms and key lengths can also be used. The following command is used to create a random 160-bit key.

```
root@ferrari#openssl rand 20 | hexdump -e '20/1 "%02x"'
```

Flush all current IPSec rules

```
root@ferrari#ipsecadm flush
```

Establishing security association on the Server

The following command establishes the security association 2000 from server 10.0.0.1 to the client 10.0.0.2. As mentioned above, the encryption is done using blowfish and the authentication key is SHA1

```
root@ferrari#ipsecadm new esp -spi 2000 -src 10.0.0.1 – \
dst 10.0.0.2 –forcetunnel -enc blf -key <encryption key> -\
auth sha1 –authkey  <authentication key>
```

The following command establishes the security association 2001 from the client 10.0.0.2 to the server 10.0.0.1

```
root@ferrari#ipsecadm new esp -spi 2001 -src 10.0.0.2 -dst\
10.0.0.1 -forcetunnel –enc blf -key <encryption key> -auth\
sha1 -authkey <authentication key>
```

Creating the Flow on the Server

The following command creates the flow of the data going from the client to outside

```
root@ferrari#ipsecadm flow -in -src 10.0.0.1 -dst \
10.0.0.2 -proto esp -addr 10.0.0.2 255.255.255.255 0.0.0.0\
0.0.0.0 –dontacq
```

The following command creates the flow of the data going from outside to the client

```
root@ferrari#ipsecadm flow -out -src 10.0.0.1 –dst \
10.0.0.2 -proto esp -addr 0.0.0.0 0.0.0.0 10.0.0.2 \
255.255.255.255 –dontacq
```

Client Configuration

The client configuration is identical to the server configuration, except for the IP address which should be 10.0.0.2 and the flows.

Creating the Flow on the Client

The following command creates the flow for the data going from the client to outside

```
root@ferrari#ipsecadm flow -in -src 10.0.0.2 -dst \
10.0.0.1 -proto esp -addr 10.0.0.2 255.255.255.255 0.0.0.0\
0.0.0.0 –dontacq
```

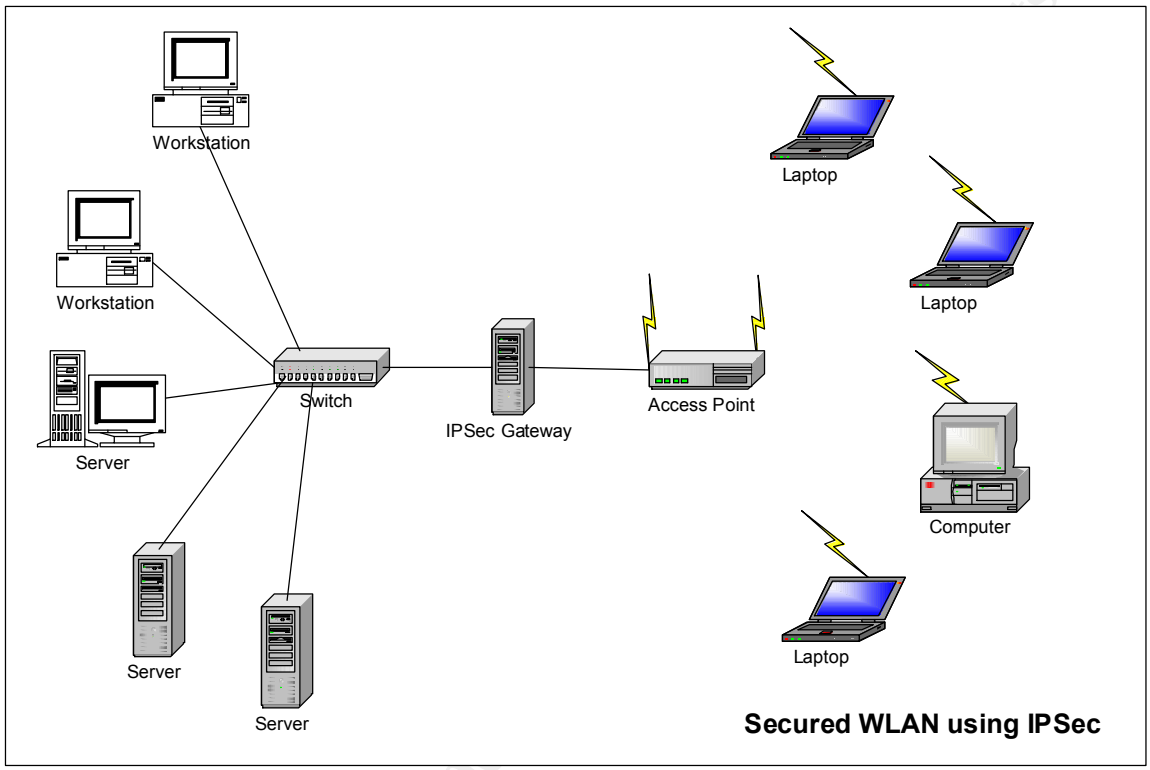
The following command creates the flow for the data going from outside to the client

```
root@ferrari#ipsecadm flow -out -src 10.0.0.2 –dst \
10.0.0.1 -proto esp -addr 0.0.0.0 0.0.0.0 10.0.0.2 \
255.255.255.255 –dontacq
```

The above configuration enables an OpenBSD machine as an IPSec gateway. Once IPSec is configured properly, an rc.ipsec file should be created with the configuration commands to make them run each time the machine is rebooted. The IP addresses shown here are the IP addresses used for the test machine. The IP addresses must be changed to suit the user's environment. The machine can be further used to prevent all kinds of IP spoofing by using strict packet filtering rule sets using ipf [11].

The IPSec clients can be anything from Windows 2000 clients to other OpenBSD machines. The MTU size has to be taken care of as it should not be more that

1500. The following diagram shows a WLAN architecture using an OpenBSD machine as IPSec Gateway. All traffic in the WLAN is encrypted with IPSec.



Secure WLAN using IPSec Gateway

If WEP is enabled with IPSec, the proposed solution provides two layer of encryption to protect the transmitted data from eavesdropping. The transmitted data is then encrypted at both layer 2 (Data Link) and layer 3 (Network) of OSI model. As the data encrypted with WEP is also encrypted with IPSec, there is not much left for the attacker to base their attack against. Sniffing this traffic is of no use to the attacker as the encryption used is not yet crackable.

Conclusion

WLANs are inherently insecure. Weaknesses in WEP are well known and tool that can be used to crack WEP within minutes are widely available. One should not depend on WEP as the only security mechanism for WLAN. Until a more viable security mechanism is available, various precautions must be taken to secure a WLAN. This paper makes several recommendations that must be adopted to provide additional security layers to WLAN. Logs are of no use if they are not analyzed regularly. Logs on the WLAN must be regularly reviewed and regular security audits must be carried out to find and patch new vulnerabilities in the WLAN.

References

1. Borisov, Goldberg, Wagner "Security of the WEP algorithm." URL: http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html
2. Westoby, Kevin, "Security Issues Surrounding Wired Equivalent Privacy", March 2002, URL: http://www.cas.mcmaster.ca/~wmfarmer/SE-4C03-02/projects/student_work/westobkj.html
3. Flickenger, Rob, "Antenna on the Cheap (er, Chip)" July 2001, URL: http://www.oreillynet.com/cs/weblog/view/wlg/448
4. Cox Johd, "Serious security weakness in 802.11b wireless LANs exposed", June 2001. URL: http://www.nwfusion.com/news/2001/0806ieee.html
5. Newsham, Tim "Cracking WEP Keys" June 2001, URL:
6. Hulton, David "Practical Exploitation of RC4 Weaknesses in WEP Environments", 22 February 2002, URL: http://www.dachb0den.com/projects/bsd-airtools/wepexp.txt
7. Tiong Ow, "IEEE 802.11b Wireless LAN: Security Risks", 20 September 2001, URL: http://rr.sans.org/wireless/IEEE.php
8. Owen, Daniel, "Wireless Networking Security: As Part of Your Perimeter Defense Strategy", January 23, 2002, URL: http://rr.sans.org/wireless/netsec.php
9. Ellison, Craig,"Exploiting and Protecting 802.11b Wireless Networks", 4 September 2001, URL: http://www.extremetech.com/print_article/0,3428,a=13880,00.asp
10. OpenBSD Web Site URL: http://www.openbsd.org
11. Stein , Joshua, "Replacing WEP With IPSec", 20th May 2002, URL: http://rt.fm/~jcs/ipsec_wep.html
12. "Using IPSec Clients with OpenBSD", URL: http://www.allard.nu/openbsd/
13. "VPN and Wireless Security", URL: http://www.linksys.com/edu/vpnwireless.asp
14. Lewis, Jason, "Wireless LAN Security", March 30, 2002, URL: http://www.packetnexus.com/docs/packetnexus/Wireless%20LAN%20security.pdf
15. AirSnort URL: http://airsnort.shmoo.com/
16. WepCrack URL: http://wepcrack.sourceforge.net/

Appendix A

List of cards prism 2 chip set cards which allows sniffing

Addtron AWP-100

Bromax Freeport

Compaq WL100

D-Link DWL-650

GemTek (Taiwan) WL-211

Linksys WPC11

Samsung SWL2000-N

SMC 2632W

Z-Com XI300

Zoom Telephonics ZoomAir 4100

LeArtery Solutions SyncbyAir LN101

Appendix B

Some of the popular tools used for WEP cracking:

AirSnort

Wepcrack

NetStumbler