



Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Hacker Tools, Techniques, and Incident Handling (Security 504)"
at <http://www.giac.org/registration/gcih>

**Employing Packet Replay
Techniques against the WEP
Encryption Scheme**

GCIH

Practical Assignment
Version 4.0
Option 1

Ng Zee Howe, Alex
SANS LMP: Track 4
Singapore, 22nd March, 2005

Table ofContents

<u>Abstract</u>	3
<u>Document Conventions</u>	4
<u>Statement of Purpose</u>	5
<u>The Exploit</u>	6
<u>Name</u>	6
<u>Operating System</u>	6
<u>Protocols/Services/Applications</u>	6
<u>Affected Systems</u>	7
<u>Description</u>	7
<u>Exploit/Attack Signatures</u>	7
<u>Stages of the Attack Process</u>	7
<u>Platforms/Environments</u>	8
<u>Reconnaissance</u>	10
<u>Scanning</u>	12
<u>Exploiting the System</u>	14
<u>Network Diagram</u>	19
<u>Keeping Access</u>	20
<u>Covering Tracks</u>	21
<u>The Incident Handling Process</u>	22
<u>Incident Context</u>	22
<u>Preparation</u>	22
<u>Identification</u>	23
<u>Timeline</u>	24
<u>Containment</u>	25
<u>Eradication</u>	25
<u>Recovery</u>	25
<u>Lessons Learnt</u>	26
<u>List of References</u>	27
<u>Works Cited</u>	27

Abstract

This paper will demonstrate the effectiveness of using a packet replay attack against a Wired Equivalent Privacy (WEP) secured 802.11b/g network. The majority of existing wireless networks deployed today implement an encryption scheme called WEP. This scheme was designed to ensure that data transmitted across the wireless medium would enjoy an equivalent amount of security as using an Ethernet cable, hence the name. The WEP scheme describes a method in which packets can be encrypted using either a 40 bit or 104 bit key before transmission. Only clients in possession of the key are able to participate in the wireless network.

For many years the weaknesses of the WEP Encryption scheme against statistical attacks have been a much publicized fact. This has led to WEP cracking with exploit tools such as Aircrack¹ and Wepcrack². Using these tools was a trivial but extremely time consuming process. It typically took anywhere from 24 to 72 hours to gather the requisite number of data packets for a successful attack. Hardware vendors were eventually able to convincingly defeat such attacks by introducing firmware that could filter packets with weak Initialization Vectors (IV), as well as offering dynamic encryption key rotation schemes, whereby the WEP keys¹ for a given network were changed before an attacker would have had time to gather a large enough data set to perform an attack.

However several new developments have made WEP cracking feasible again. They will be explained in section titled "The Exploit". Suffice to say that these developments enable the attacker to defeat the obstacles listed above in a matter of minutes as opposed to hours using so called "older generation" attacks.

A scenario will be used to better illustrate the risks that small to medium size companies face when deploying a wireless network. Larger organizations typically have the resources and knowledge to deploy additional protection measures in addition to WEP. Bob Black is an IT savvy chemical salesman for an ambitious and unscrupulous company named ToxChem. Their rival across town is named Mercury Chemicals. Andy White is their IT administrator who has just deployed a wireless network and secured it with WEP. This paper will explain step by step how Bob gains access to the Mercury Chemicals network.

The perspective then switches to that of the incident handler, in this case Andy White. His response to this sort of attack is then discussed in the context of the 6 Steps to Incident Handling. It is hoped the reader will gain a practical understanding of how these attacks are conducted and more importantly, how they may be averted.

¹ <http://aircrack.shmoo.com/>

² <http://wepcrack.sourceforge.net/>

Document Conventions

When you read this practical assignment, you will see that certain words are represented in different fonts and typefaces. The types of words that are represented this way include the following:

Operating system commands are represented in this font style. This style indicates a command that is entered at a command prompt or shell.

Filenames, paths and directory names are represented in this style.

The results of a command and other computer output are in this style.

URL

Web URL's are shown in this style.

Quotation

A citation or quotation from a book or web site is in this style.

© SANS Institute 2000 - 2005, Author retains full rights.

Statement of Purpose

The relative difficulty in performing a WEP cracking attempt in recent years has led to the perception that WEP is safe enough in the face of current generation attacks. It simply took too long for an attacker to gather enough viable data packets to perform an attack.

However with the introduction of several advancements, this borrowed time is now over and the latest attacks make the WEP encryption scheme truly obsolete.

In summer 2004, a hacker named KoreK first discussed a tool called Chopper³ on the Netstumbler forums. It reduced the number of packets that was required to reliably crack a WEP key from millions of packets to just a few hundred thousand. The tool is no longer maintained and a better implementation can be found in the latest versions of Aircrack. A further discussion of the different types of wireless packets is presented later in this paper.

However it still takes some time to passively gather a few hundred thousand data packets. This data gathering process can be accelerated even further with an arp replay attack such as the one implemented in Aireplay⁴.

The idea of replaying wireless packets to speed up the data packet generation was first discussed on the Netstumbler forums⁵ by Christophe Devine. He demonstrated this concept by building a toolkit called Aircrack.⁶ This toolkit combines packet sniffing, packet replay and the new Korek optimized WEP key cracking. Successful use of this toolkit will allow the user to derive a WEP key within minutes as opposed to hours if a passive method was used.

The purpose of this document is also to illustrate how easy it could be for an attacker to gain access to a wireless network without physical access to your network. It will also provide a primer on which tools to look out for, so that security administrators can be aware of their function.

Bob Black shall be attempting to access the internal networks of Mercury Chemicals by using steps that a real life attacker would take. His actions will ultimately result in him stealing sensitive corporate data.

This paper concludes with a section on the 6 steps Andy White would take to handle this incident.

³ <http://www.netstumbler.org/showpost.php?p=89692&postcount=22>

⁴ Part of the Aircrack toolkit, see footnote 6

⁵ <http://www.netstumbler.org/showthread.php?t=11878>

⁶ <http://www.cr0.net:8040/code/network/aircrack/>

The Exploit

Name

Since this attack does not have a CVE or CERT listing, there is also no formal name for it. However it is referred to on the various wireless interest forums as ARP packet spoofing or wireless packet replay attack.

The only publicly available tool that implements this attack is Aireplay.

Operating System

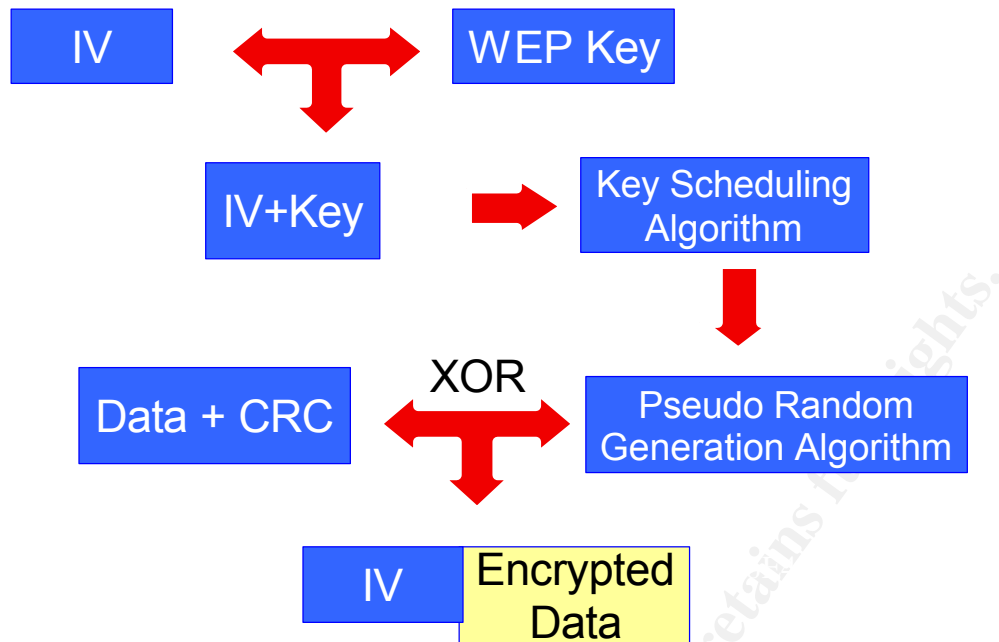
This attack simply is a component in the WEP cracking process because it accelerates the rate of data collection. After which a separate tool is still required to perform the statistical analysis on the collected data. As such it does not affect operating systems directly. This attack is restricted to the network layer, thus any operating systems that can understand 802.11 traffic could be indirectly affected once the WEP key is obtained.

Protocols/Services/Applications

To gain an in depth understanding of how this replay attack works it is necessary to provide a quick primer on the WEP encryption scheme. The WEP standard defines a method by which a 24 bit Initialization Vector (IV) is concatenated with a 40 bit or 104 bit secret shared key which is used as the RC4 seed. RC4 is a stream cipher developed by the RSA Corporation.⁷

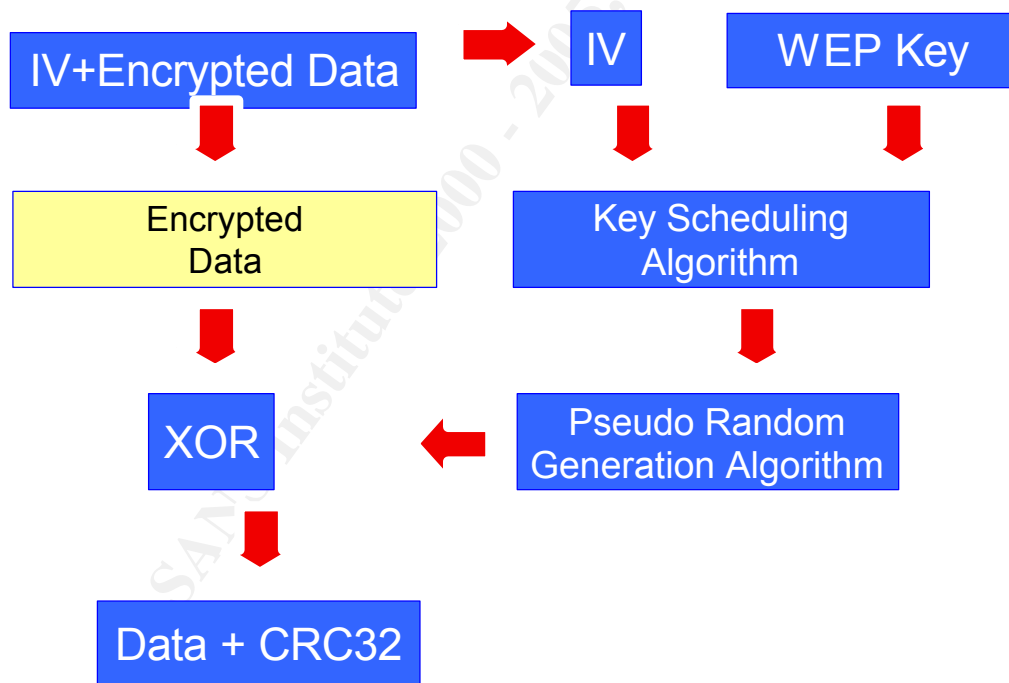
Most wireless cards and Access Points (AP) generate the IV using a pseudo random number generator (prng). The data payload plus a 32bit checksum is then encrypted and sent out together with the IV in plaintext. See the figure below.

⁷ www.rsasecurity.com/



WEPA Encryption Process

On the receiving end, the recipient simply concatenates their received IV with their secret key to decrypt the payload. If the checksum is verified then the data is valid.



WEPA Decryption Process

This scheme however is inherently flawed and deemed unfixable. A detailed

explanation of the cryptanalysis behind this assertion is beyond the scope of this paper.

However the end result is that when the new KoreK inspired statistical optimizations are used, all it takes is 1 million packets to dependably guess a 104 bit secret key and about 500,000 IV encrypted packets to guess a 40bit secret key in a trivial amount of time.

Affected Systems

All wireless networks which implement the WEP encryption scheme are susceptible to this attack. However some brands and models of AP are immune to this attack because they either implement frame counters or

Description

The Aircrack toolkit is available for both Windows and Linux environments; however it is much easier to get the wireless drivers working in Linux. Aircrack is usually used in conjunction with Kismet⁸, which in turn only works on Linux.

The process of deriving the WEP key from statistical analysis begins by collecting a data set which contains enough WEP encrypted packets to infer a solution. This is accomplished through passive listening using tools such as Kismet and Airodump. However this passive process can take hours. To create more traffic, an active solution such as ARP packet replaying has to be implemented.

Wireless packets can either be unencrypted or encrypted. Unencrypted packets are useless for WEP key cracking and can be ignored. They are usually AP beacon frames and client probe requests.

On the other hand, ARP traffic is encrypted and thus can be useful when cracking the data set. Since it is encrypted, choosing the correct 802.11 frame to replay can be a time consuming experience. The Aireplay tool can either accept live input from your wireless interface, or a saved packet file. It will only attempt to replay packets that conform to a certain length and are encrypted. This represents the tool's best guess as to whether the 802.11 frame is in reality an ARP packet.

It is usually better to go with traffic analysis using tcpdump or Ethereal to select a likely ARP packet. I have described the Ethereal packet filter string used in the section titled "Stages of the Attack".

If your assumption is correct and the packet is in reality an ARP packet, what will happen is that the wireless AP will broadcast this request to all hosts on the wireless segment; prompting a flood of useful WEP encrypted ARP replies. In this manner the data collection phase is massively accelerated, allowing the attacker to quickly break into the wireless network.

⁸ See note 13.

Exploit/Attack Signatures

Wireless hacking techniques uses jargon akin to submarine warfare. Just like the passive and active sonar's that submarines can deploy to locate targets, the wireless attacker also has a range of passive and active attack tools. For example Kismet and airodump are examples of passive listening tools. They do not generate an attack signature at all and are practically impossible to detect electronically.

On the other hand, ARP packet replay spoofing as implemented by Aireplay is an active packet generation attack. This means that the attack signature generated is very recognizable. The attack will generate a massive amount of identical 802.11 frames. By monitoring the traffic on a wireless LAN, one will be able to see results as shown in figure 17. A wireless IDS such as AirDefense Guard⁹ or Snort-Wireless¹⁰ will pick up on this activity quite easily, as the wireless medium will suddenly be flooded with identical ARP requests.

Another unfortunate side effect of using Aireplay is that sometimes the flood of ARP packets will bring the Wireless AP down. This is entirely dependent on the brand and model of the AP. This will immediately prompt the IT administrator to power cycle the Wireless AP. If the Wireless AP crashes consecutively within the space of a few minutes, it might be a good idea to check if a replay attack is being conducted against it.

Stages of the Attack Process

Bob Black was an ambitious young executive at the newest chemical company in town, ToxChem. In order to grow his sales, he hit upon the idea to steal sensitive pricing and customer data from a rival company around town. In this way he would be able to easily exceed his sales targets!

After reading about the strength of the new WEP attacks, he decides to unobtrusively drive by each of his competitor's workplaces to discover if they are running wireless networks.

He knew that risk of being traced after performing a wireless attack was very low. Furthermore it was extremely unlikely that he would be caught while

⁹ AirDefense, AirDefense Guard, www.airdefense.net/products/airdefense_ids.shtml

¹⁰ Snort Wireless, snort-wireless.org

accessing the wireless network from outside the target compound.

After surreptitiously scanning each of his competitors, he determined that only Mercury Chemicals was running a wireless network. Armed with this knowledge, he returns to his home to prepare his attack.

Platforms/Environments

To successfully carry out his attack, Bob needs to perform an activity called Wardriving¹¹. This involves driving around target locations with a laptop loaded with a wireless card, wardriving software and an external antenna. The platform that Bob Black uses is the Auditors Toolkit¹². This can be downloaded as a bootable CD-ROM and is based on a Linux distribution called Kanotix. It comes preloaded with all the various wireless attack tools that Bob will need. Any Linux distribution can be used; however it is essential that the wlan-ng¹³ wireless drivers are installed on them.

These tools will be introduced as the attack progresses. Another good reason is that the new attacks that implement the KoreK optimizations do not work on the Windows environment at this time.

The Access Point (AP) that this attack was carried out on was a D-Link DI-624.

The choice of wireless card is important because the wireless chipset used must be compatible with certain Linux based wireless card drivers. The two most common chipsets are the Prism and the OriNoco chipsets. It is important that the card chosen belongs to either of these types. Another important consideration is that the card must contain a port to plug an external antenna into. For the purposes of this paper, Bob used a Compex WL-54G PCMCIA card.

The external antenna extends the effective range of his wireless card. This is important because not only does it allow Bob to detect more wireless networks, it also affords him a measure of physical obscurity when he performs an attack. A detailed discussion of what type of antenna to use is beyond the scope of this paper, however as a rule of thumb, a directional YAGI style antenna (see fig. 1) is recommended for targeted attacks because it allows for the greatest signal gain when pointed at the specific network.

¹¹ For more information on Wardriving, please visit www.wardriving.com

¹² Moser, Max, Auditor's Security Toolkit, 2005, www.remote-exploit.org

¹³ Linux WLAN Project, <http://www.linux-wlan.org/>



1: Sample Directional Antenna.¹⁴

The next useful piece of equipment is a Global Positioning System (GPS) receiver. This device will allow a geographical map of the wireless networks in the area to be created. This greatly enhances reconnaissance phase of the attack because it allows the attacker to quickly pinpoint the location of specific wireless networks.



Figure 2: Sample GPS Receiver¹⁵

¹⁴ Source: Cantenna website, www.cantenna.com

¹⁵ Source: Garmin website, GPS18 OEM, www.garmin.com/products/gps18oem/

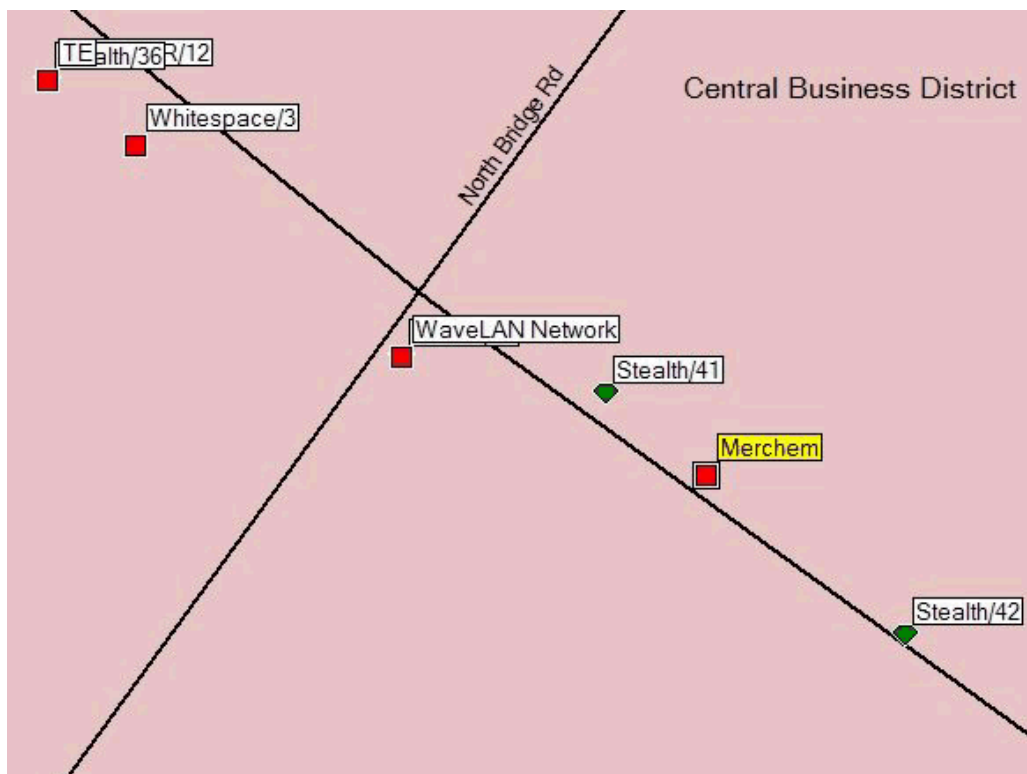


Figure 3: Sample Wardrive Map

Bob managed to use his GPS and wardriving software to create the above map and home in on the Mercury Chemical network.

Reconnaissance

Bob begins by driving past each of his competitor's compounds at noon, searching for a wireless network to penetrate. The two most common programs that can do perform this task are Netstumbler¹⁶ and Kismet¹⁷ by Mike Kershaw.

Both these programs work by hopping between the eleven 802.11b/g defined channels and detecting wireless beacon or data packets. The main difference is in the manner that they elicit responses from APs. Netstumbler actively sends out probe requests once a second to ping APs. This behavior is noisy and easy to detect, in fact Kismet can be deployed to detect Netstumbler activity.

Kismet on the other hand is a completely passive tool; it does not generate

¹⁶ Milner, Marius, Netstumbler, 2005, www.netstumbler.com

¹⁷ Kershaw, Mike, Kismet, 2004, www.kismetwireless.net

any 802.11 frames. It simply listens for any activity, thus it is the tool of choice when conducting wireless reconnaissance for extended periods.

Netstumbler is invoked by simply double clicking on the program icon. It only works on the PocketPC and Windows platforms.

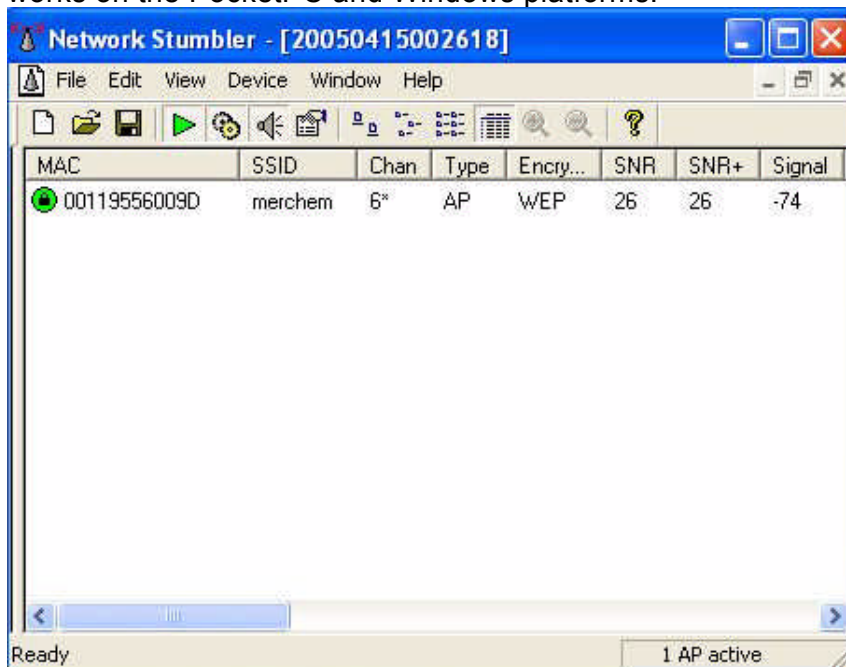


Figure 4: Netstumbler Output

Kismet is invoked by:

To configure your particular preferences (eg. Network interface, logfile name) use vi to edit

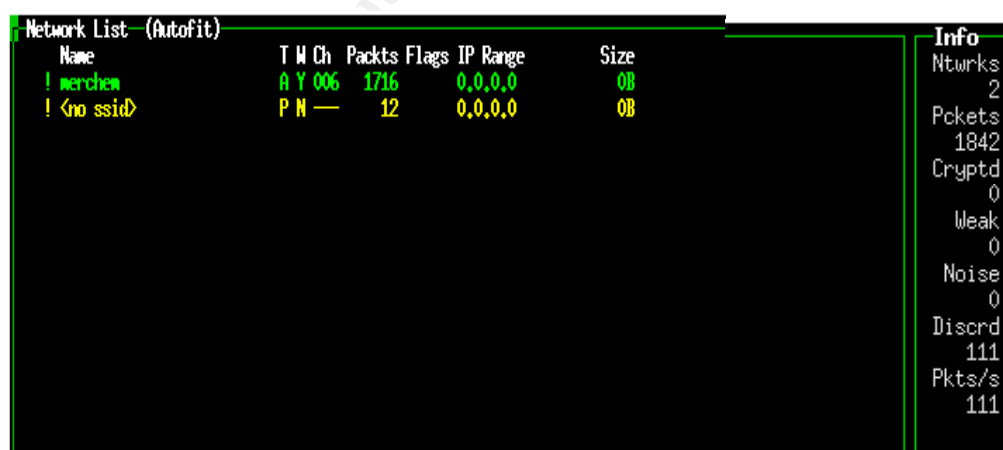


Figure 5: Kismet Output

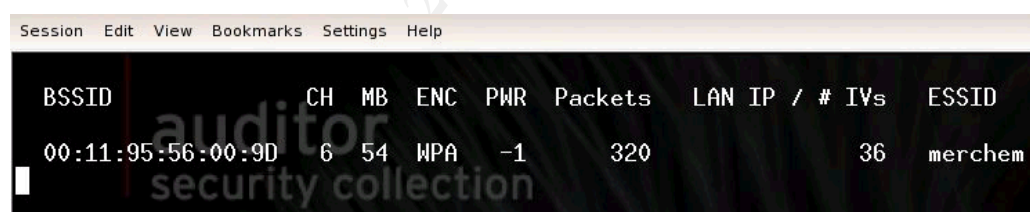
In Bob's case, he decides to use Netstumbler because of its ease of use. It detects the Mercury Chemicals wireless AP called "merchem" (fig.4). It is WEP encrypted and broadcasting on channel 6. He then proceeds to find a nearby spot to park his car and move on to the next stage of the attack.

Scanning

Bob first uses a packet sniffer to collect some sample packets for further analysis. Kismet automatically generates a logfile when detecting networks, however Airodump¹⁸ is a tool which is easier to run. Please note that Airodump will not channel hop unless Kismet or some other channel hopping software is active. Certain wireless cards can be put into a mode called "monitor", where all 802.11 frames are sniffed regardless of the destination address. This is akin to putting an Ethernet card into "promiscuous" mode, only that the card has to sniff frames on all channels.

As explained earlier, since the WEP scheme works on the network layer the packets received are actually 802.11 frames. Only when the WEP key is used to decrypt 802.11 frames, then the actual transport layer TCP/IP packets are visible. For the rest of this discussion packets and frames shall be used interchangeably unless they are referred to as TCP/IP packets.

In this case we simply want to monitor on channel 6, thus we need to initialize some settings on the wireless card. The commands below will initialize the card on eth1 to monitor mode on channel 6. Next airodump will start collecting data restricted to BSSID 00:11:95:56:00:9D. The output will be saved into a file called sniff.cap.



The screenshot shows the Airodump application window with a menu bar (Session, Edit, View, Bookmarks, Settings, Help) and a table of detected networks. The table has columns for BSSID, CH, MB, ENC, PWR, Packets, LAN IP / # IVs, and ESSID. One network is listed: BSSID 00:11:95:56:00:9D, CH 6, MB 54, ENC WPA, PWR -1, Packets 320, LAN IP / # IVs 36, and ESSID merchem.

BSSID	CH	MB	ENC	PWR	Packets	LAN IP / # IVs	ESSID
00:11:95:56:00:9D	6	54	WPA	-1	320	36	merchem

Figure 6: Airodump Output

The figure above shows that the "merchem" AP is running at 54Mbps on channel 6. The encryption scheme shows up as Wireless Protected Access (WPA), however this only means that the AP supports WPA, not necessarily that it is running it. Finally the output also shows of 320 packets collected, 36 are encrypted packets. His target is to collect 500,000+ encrypted packets. How will Bob achieve this figure in a short amount of time? After letting airodump run for a few minutes longer Bob decides to look into the output.

¹⁸ Devine, Christophe, Airodump, 2004, www.cr0.net:8040/code/network/aircrack/

Ethereal¹⁹ is a perfect tool for this purpose. It can understand the output from both Kismet and airodump.

No.	Time	Source	Destination	Protocol	Info
1	0.000000	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
2	0.006648	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
3	0.013565	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
4	0.020908	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
5	0.027852	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
6	0.035152	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
7	0.046993	Cisco-Li_a5:66:db	Broadcast	IEEE 802.11	Beacon frame
8	0.053180	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
9	0.069647	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
10	0.076095	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
11	0.083132	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
12	0.089740	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
13	0.100888	LibitSig_12:00:00	Broadcast	IEEE 802.11	Beacon frame
14	0.149300	Cisco-Li_a5:66:db	Broadcast	IEEE 802.11	Beacon frame
15	0.150005		PhilipsC_42:8e:3d (R)	IEEE 802.11	Acknowledgement
16	0.234639	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
17	0.241840	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
18	0.251973	Cisco-Li_a5:66:db	Broadcast	IEEE 802.11	Beacon frame
19	0.257653	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
20	0.264283	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
21	0.281375	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
22	0.288110	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
23	0.295280	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
24	0.302265	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
25	0.305097	LibitSig_12:00:00	Broadcast	IEEE 802.11	Beacon frame
26	0.311555	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
27	0.318378	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
28	0.325531	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
29	0.354234	Cisco-Li_a5:66:db	Broadcast	IEEE 802.11	Beacon frame
30	0.382339	Cisco-Li_a5:66:db	Spanning-tree-(for-b)	IEEE 802.11	Data
31	0.456676	Cisco-Li_a5:66:db	Broadcast	IEEE 802.11	Beacon frame
32	0.457411	Cisco-Li_a5:66:db	Spanning-tree-(for-b)	IEEE 802.11	Data
33	0.461509	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
34	0.464994		Netgear_e6:6b:48 (RA)	IEEE 802.11	Acknowledgement
35	0.472075	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
36	0.478703	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
37	0.485895	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
38	0.493044	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
39	0.499735	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
40	0.507254	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
41	0.509891	LibitSig_12:00:00	Broadcast	IEEE 802.11	Beacon frame
42	0.516660	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
43	0.523951	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request
44	0.530786	Intel_58:b9:2f	Broadcast	IEEE 802.11	Probe Request

Frame 30 (84 bytes on wire, 84 bytes captured)
 IEEE 802.11
 Data (46 bytes)

```

0000 08 43 3a 01 00 0c 41 67 23 38 00 0f 66 a5 66 db  .C...Ag #8..f.f.
0010 01 80 c2 00 00 00 c0 79 00 0f 66 a5 66 db 6a c2  ....y ..f.f.j.
0020 1d 00 b4 7c 3b 99 30 68 e9 98 86 93 ab 2a 2d 0f  ...;..0h ....*-l
0030 bf e4 95 70 f8 05 a3 12 f7 59 6c 4f f2 74 9f a8  ...p....Yl0.t...
0040 4c 6c 0b 35 27 47 52 e4 b8 b2 2f 63 cf 31 b5 19  LL.S'GR. ./c.l..
  
```

Figure 7: Unfiltered Ethereal Output

Ethereal will automatically resolve MAC addresses to manufacturer names, thus we see names like Cisco, Intel and Netgear above. These MAC addresses all correspond to different wireless clients and APs.

¹⁹ Ethereal, www.ethereal.com

Bob's task is to now identify a suitable packet for replay. The easiest packet to replay is the ARP packet. However the difficulty is in finding an ARP packet amongst the jumble of packets encountered above. Fortunately Ethereal is able to filter out ARP packets for us. The trick is to realize that ARP packets have the following characteristics:

- 1) They are WEP encrypted packets.
- 2) They have a destination address of FF:FF:FF:FF:FF:FF (broadcast)
- 3) They have a length of 68 bytes.

By translating the above 3 rules into an ethereal packet filter we get:

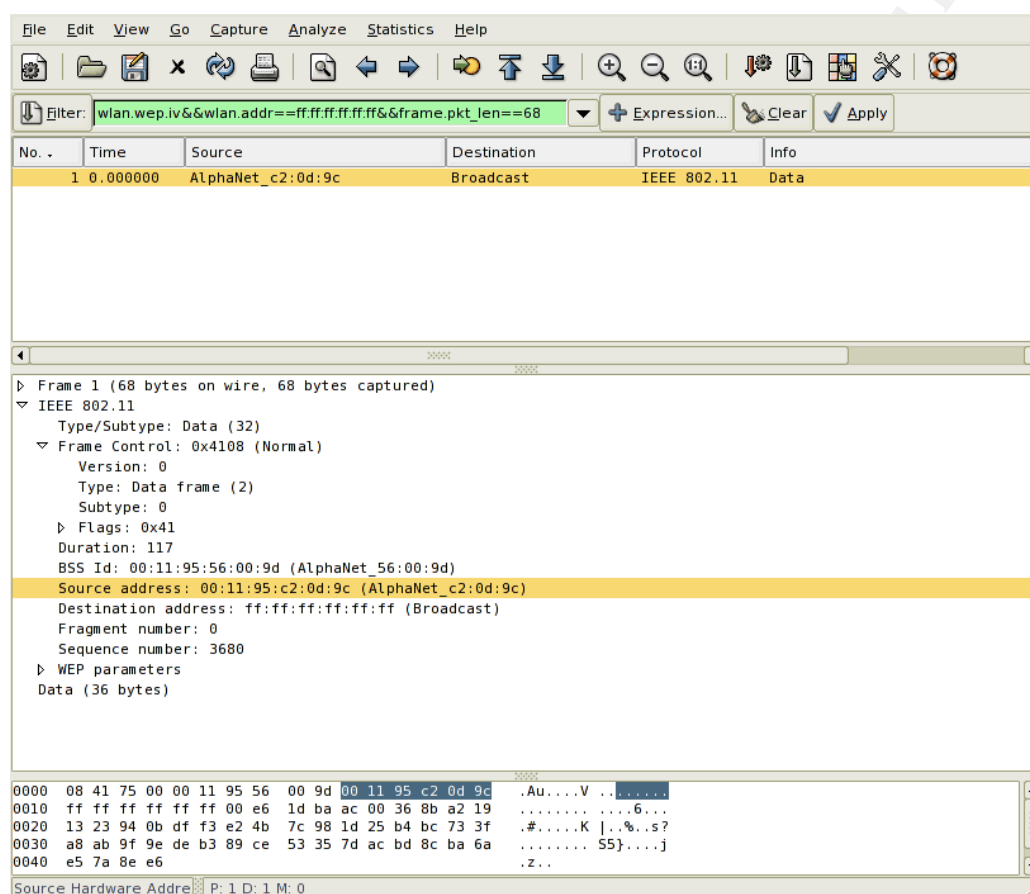


Figure 8: Filtered Ethereal Output

This above packet looks very promising. It is most likely an ARP packet from the MAC address 00:11:95:c2:0d:9c. Bob saves this packet in a separate file called `magic_pkt.cap`. He now feels confident of moving into the next stage of the attack.

Exploiting the System

To speed up the collection of encrypted packets, Bob now has to replay the above packet over and over again. This will fool the “merchem” AP into replying with thousands of ARP response packets. Bob's packet sniffer will then be able to collect the requisite 500,000+ encrypted packets.

To accomplish this he uses the second part of Aircrack, called Aireplay.

The above command gets aireplay to replay our spoofed ARP packet on wireless interface eth1. Sometimes it is necessary to limit the rate of packets sent in order to avoid crashing the wireless AP. This is done by adding by -x <packet rate> as an argument. If the wireless AP is crashed this will bring the wireless attack to a premature ending.

```
root@4[dumpfiles]# aireplay -r magic_pkt.cap eth1

Found one usable WEP data packet:

    From DS = 0, To DS = 1
    BSSID   = 00:11:95:56:00:9D
    Src. MAC = 00:11:95:C2:0D:9C
    Dst. MAC = FF:FF:FF:FF:FF:FF

    0x0000: 0841 7500 0011 9556 009d 0011 95c2 0d9c .Au....V.....
    0x0010: ffff ffff ffff 00e6 1dba ac00 368b a219 .....6...
    0x0020: 1323 940b dff3 e24b 7c98 1d25 b4bc 733f .#.....Kl..%.s?
    0x0030: a8ab 9f9e deb3 89ce 5335 7dac bd8c ba6a .....S5}....j
    0x0040: e57a 8ee6                      .z..

Replay this packet ? y
Saving replayed packet in replay-20050313_2014.cap
Open airodump in another console to capture replies.
Sent 16991 packets at 969 pkt/s
```

Figure 9: Aireplay Output

Bob then opens airodump in another console to record the results of his attack.

His attack has worked! Very rapidly he has collected over 18,000 encrypted packets in just 90 seconds.

Session Edit View Bookmarks Settings Help								
BSSID	CH	MB	ENC	PWR	Packets	LAN IP / #	IVs	ESSID
00:11:95:56:00:9D	6	54	WPA	-1	69135		18474	merchem

Figure 10: Output after 90 seconds

Bob finally collects 500,000+ packets at just over 30 minutes. Using Aireplay resulted in a packet generation rate of approximately 300 encrypted packets/second.

Session Edit View Bookmarks Settings Help								
BSSID	CH	MB	ENC	PWR	Packets	LAN IP / #	IVs	ESSID
00:11:95:56:00:9D	6	54	WPA	-1	1730627		510509	merchem

```
root@3[GIAC]#
```

Figure 11: Output after 34 minutes

After this breakthrough, Bob excitedly employs the final component of Aircrack to derive the WEP key.

```
root@3[GIAC]# aircrack sniffer.cap
Opening pcap file sniffer.cap
Choosing first WEP-encrypted BSSID = 00:11:95:56:00:9D
Reading packets: total = 963118, usable = 343224
```

```
KEY FOUND! [ 872D7525F1 ]
```

```
root@3[GIAC]#
```

Figure 12: Aircrack Output

The result is a 10 digit key, corresponding to a 40 bit WEP key. Bob now has all the information needed to associate with the “merchem” AP. It has taken only a total of 50mins to finish completing this attack. Using so called “first generation” attacks would have typically taken up to 24 hours to reach this same point.

In fact Bob can now decrypt the contents of any wireless traffic that is collected from the Ethereal packet collector. There is a tab in the Preferences Menu of Ethereal to enter in a WEP key to decrypt wireless packets on the fly. (See fig. 13)

© SANS Institute 2000 - 2005

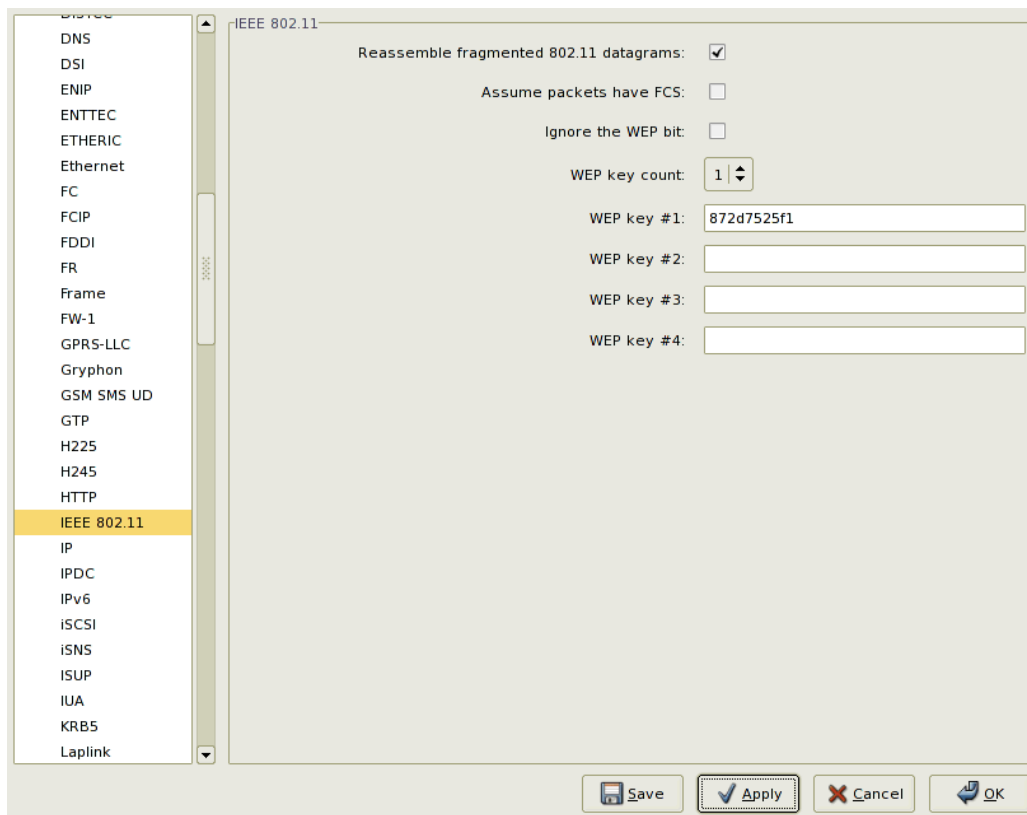


Figure 13: Decrypting 802.11 traffic using Ethereal

Once the WEP key has been input, Bob is able to see the 802.11 frames miraculously reassembled to become various higher level protocol packets such as TCP/IP, ARP and SSL traffic.

© SANS Institute 2000-2005

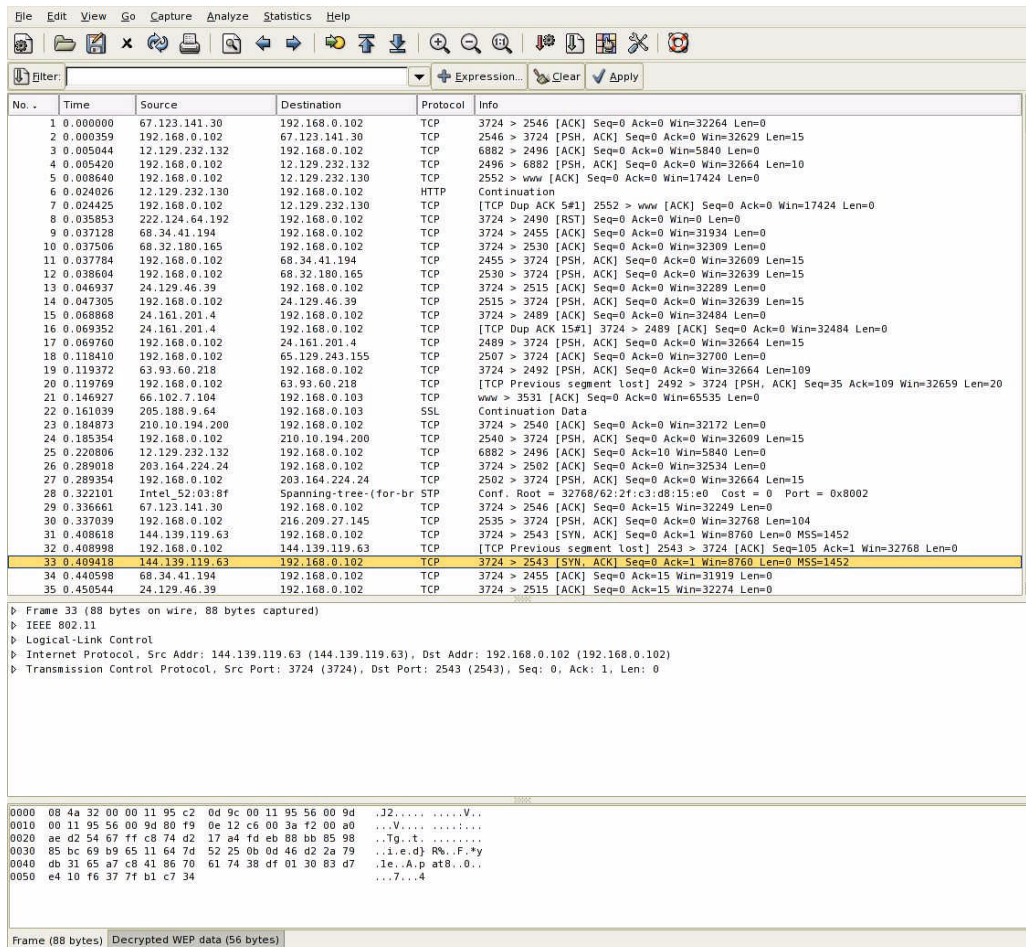
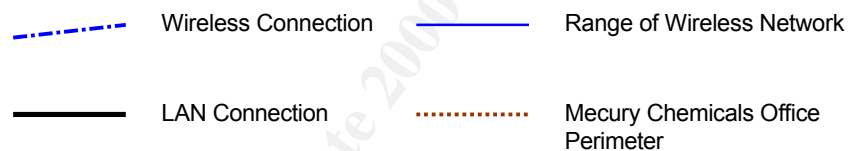
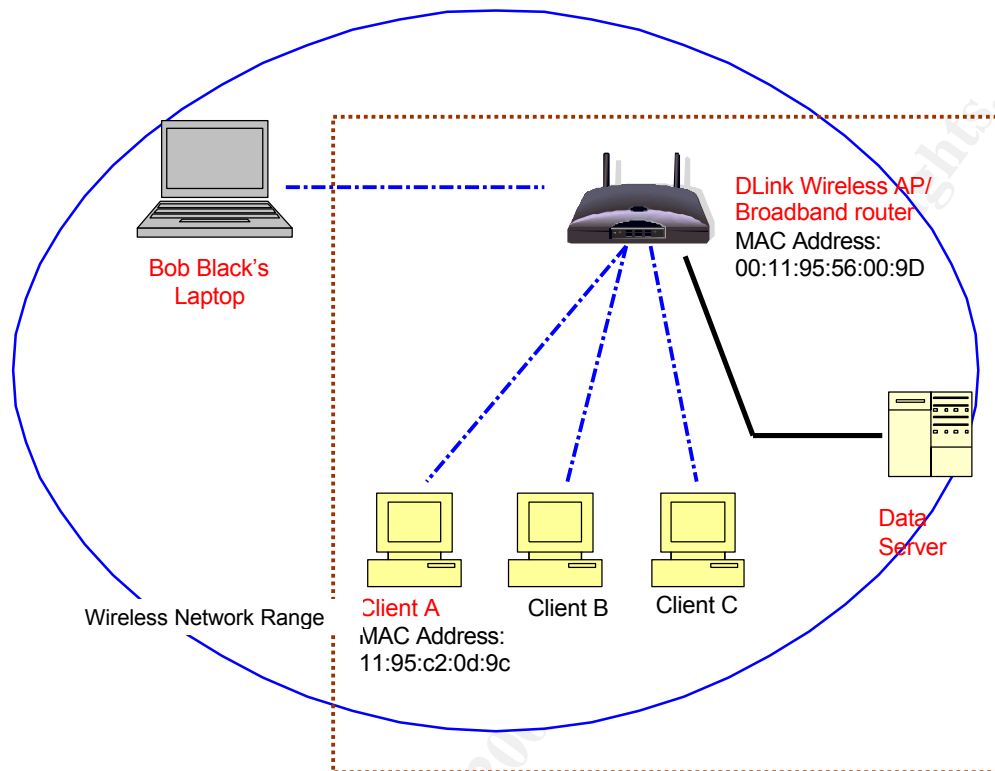


Figure 14: Decrypted packet output

Network Diagram



Computers in red
were actually
deployed in the lab.

Keeping Access

Now that Bob has obtained the WEP key, he can join the Mercury Chemicals network. If there are no further security measures in place, then he can utilize the same tools he would use on a wired LAN. The following commands will associate his wireless interface with the Mercury Chemical's network.

If the wireless access point has implemented MAC address table filtering, whereby only MAC addresses which have an entry in the table are allowed access, Bob still has another trick to play. He can change his MAC address to match some of the MAC addresses revealed in the earlier part of the attack. This is accomplished through the change-mac script provided by the Auditor's toolkit.

Bob takes a calculated gamble that Mercury Chemicals is running a Windows environment. This is based on his assessment of their network size and sophistication. He reboots his machine in Windows mode and associates his wireless card with the "merchem" network. Next he starts up his favorite integrated scanning tool called Cain & Abel²⁰. This tool is powerful because it combines ARP Poison Routing (APR), APR-DNS poisoning, password sniffing across multiple protocols, file sharing detection and a wireless scanner thrown in for good measure. APR poisoning is used to conduct man in the middle attacks and to listen to switched traffic. It works by listening to broadcast ARP traffic so that the correct IP-MAC mappings can be learned. It then proceeds to change the direction of the normal flow of traffic between 2 hosts by poisoning their ARP caches. The poisoning is carried out by broadcasting an erroneous ARP update packet. This will force the targeted hosts to route all traffic to the Cain host, whereby the attacker can inspect the traffic before passing it on to its correct destination.

He runs Cain and is immediately rewarded with a list of open file shares. The folders named "Sales Figures", "Product List" and "Customer Files" look promising. Bob decides to copy the contents of each directory into his laptop hard disk. This can be accomplished by right clicking on the desired shared folder and selecting Map Network Drive.

²⁰ Montoro, Massimiliano , Cain & Abel, 2005, www.oxid.it/cain.html

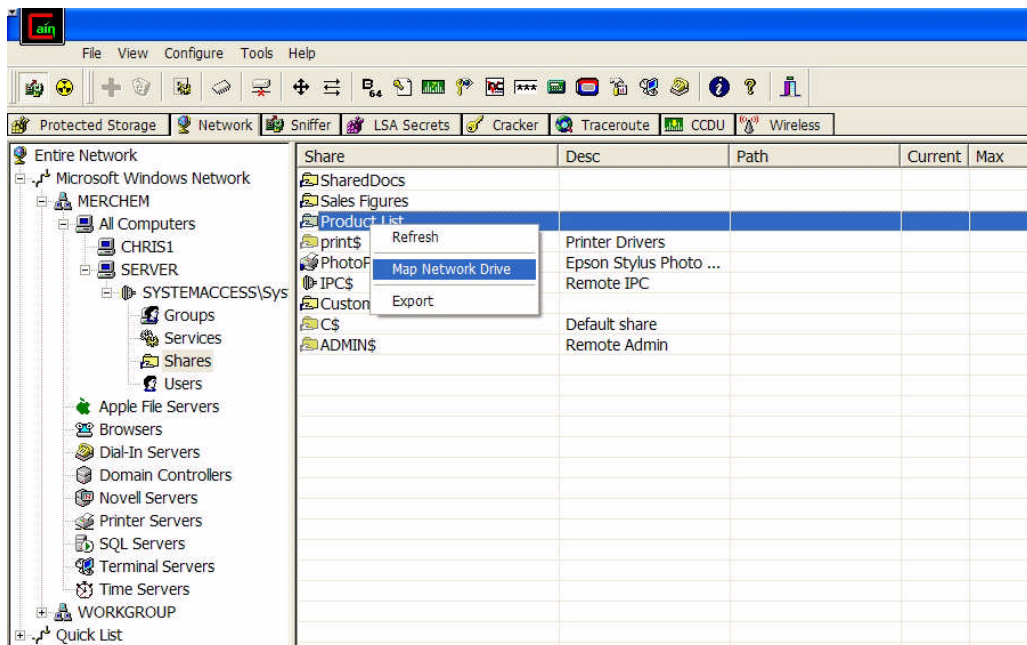


Figure 15: Cain Output showing open file shares

Covering Tracks

The dangerous thing about wireless penetration is that it usually enables the attacker to bypass the Internet facing firewalls and IDS systems, allowing direct access to the internal network.

However one of the risks faced by the attacker is being apprehended while in the vicinity of the targeted wireless network. Unlike attacks that take place remotely through the Internet, if an alert security administrator detects an unauthorized wireless client he can very quickly do a physical check to see if there are any suspicious vehicles or people in the immediate vicinity.

Added to the fact is that an attacker peering into a laptop with a large antenna hanging out of it is extremely conspicuous on a busy street. Unfortunately for the ARP replay attack to work there must be some initial wireless activity on the network to begin with. Thus attempting this attack at night will actually reduce the chances of success.

Bob must take into account factors such as the time of attack and also his level of physical concealment when planning a wireless penetration. By its very nature, active ARP replay attacks are very noisy. It is very difficult to cover your tracks once you have begun the replay attack. Thus the longer the replay attack goes on for, the higher the chance of detection. The good or bad (depending on perspective) news is that once the attacker drives away from the wireless network, there is very little that can tie him to the crime. On that premise, Bob decides to make tracks by driving off with the stolen data.

The Incident Handling Process²¹

²¹ SANS Track 4 GIAC Courseware

Incident Context

Andy White had just been instructed by the boss of the company to set up a wireless network. His boss wanted to be able to receive emails from the conference room without having to fumble with Ethernet cables. Andy had been informed about the dangers of deploying unsecured wireless LANs, thus he ensured that WEP was switched on. He also planned on manually changing the WEP key used once in a while. His wireless router supported the newer WPA standard, unfortunately the wireless clients installed on his network only supported the older WEP standard.

Since Andy worked for a small company, they did not have the manpower and hardware to deploy an IDS or an automated event log analyzer. Andy made the most of his resources by deploying a firewall to protect his broadband router and depended on his antivirus software to protect his clients from malware, virii and Trojan programs. This setup is very common in Singaporean enterprises, whereby computer security is not taken seriously until a major incident occurs.

Preparation

Andy had thought he had configured the company wireless LAN in a secure fashion. His wireless traffic was WEP encrypted with a 40 bit key. As an additional insurance, Andy also deployed a “poor mans” version of a wireless IDS. Kismet can also be used to monitor for suspicious wireless activity. Because of its client/server architecture, it can be run in drone mode.

*Remote Kismet drones are designed to turn Kismet into a stationary, distributed IDS system.*²²

Andy runs the command below on a spare computer with a compatible wireless card.

The drone configuration options are found in:

Important options to set are the correct wireless interface source and the number of users that can access the drone output.

Andy can connect to the drone server at any time by modifying the kismet.conf file to include

And running kismet.

Andy has worked out a simple procedure to follow when an actual incident occurs. He will get his team of junior technicians to report to him to whenever an incident occurs. He would then assess the severity, risk and expected impact of the incident. If further action is necessary then the server is taken offline and replaced with a backup preinstalled with the same OS and

²² Kismet Homepage, www.kismetwireless.net/documentation.shtml, Section 13

applications. The affected server is then taken to the lab for forensic analysis.

Identification

Around 12:30pm, the kismet screen on Andy's workstation starts constantly beeping with a high pitched sound. Unfortunately for Andy, he is out at lunch and unable to respond to this event immediately.

```
Status
ALERT: Suspicious client 00:0C:F1:58:B9:2F - probing networks but never participating.
ALERT: Suspicious client 00:0C:F1:58:B9:2F - probing networks but never participating.
ALERT: Suspicious client 00:0C:F1:58:B9:2F - probing networks but never participating.
Found new probed network "<no ssid>" bssid 00:11:95:C2:0D:9C
Battery: 90% 3:53:31s
```

Figure 16: Kismet ALERT warning of active probes

Andy returns at the end of his lunch hour at 1:30pm to discover that his wireless network has been constantly probed by the MAC address 00:0C:F1:58:B9:2F for some time. He is not worried at this stage because there are plenty of Wardrivers who are only looking for unencrypted wireless networks. They usually are not interested in WEP protected networks. Andy decides to investigate further by viewing his DHCP (Dynamic Host Configuration Protocol) lease information.

Host Name	IP Address	MAC Address	Expired Time
systemaccess	192.168.0.102	00-0c-f1-58-b9-2f	Apr/23/2005 02:10:38
SERVER	192.168.0.101	00-11-95-16-63-cc	Apr/22/2005 21:56:59
CHRIS1	192.168.0.100	00-11-95-c2-0d-9c	Apr/22/2005 22:55:18

To his horror he realizes that the suspicious MAC address with hostname: "systemaccess" has been granted a DHCP lease. This can only mean that an attacker has been successful in bypassing his WEP encryption scheme.

Fortunately Andy remembers that his Kismet drone server is keeping a log of all wireless traffic. He uses ethereal to view the Kismet log and determines the attacker flooded his wireless AP with ARP packets to gain access.

No. -	Time	Source	Destination	Protocol	Info
1495	134.585948	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1496	134.595287	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1497	134.596777	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1498	134.597841	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1499	134.602593	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1500	134.608306	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1501	134.609729	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1502	134.611126	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1503	134.615782	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1504	134.616879	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1505	134.618215	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1506	134.619649	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1507	134.622160	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1508	134.623347	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1509	134.624699	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1510	134.630980	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1511	134.634593	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1512	134.635803	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1513	134.638845	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1514	134.645357	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1515	134.651094	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1516	134.653632	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1517	134.654832	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1518	134.656167	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1520	134.661609	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1521	134.662817	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1522	134.667272	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1523	134.668600	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1524	134.672120	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1525	134.673293	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1526	134.677742	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1527	134.684255	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1528	134.685685	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1529	134.691133	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1530	134.692275	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1531	134.694746	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1532	134.697089	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1533	134.700463	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1534	134.702676	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1535	134.710374	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP
1536	134.716900	192.168.0.102	Broadcast	ARP	Who has 192.168.0.102? Gratuitous ARP

Figure 17: ARP Replay attack signature

Timeline

12:30pm Bob parks his car near the Mercury Chemicals warehouse and starts using Netstumbler.

12:30pm Andy's kismet drone server detects suspicious activity, but the entire IT team is out for lunch.

12:40pm Bob is sifting through captured 802.11 frames for a suitable ARP packet.

12:45pm Bob finds a suitable packet and begins the ARP replay packet generation technique.

1:18 pm Bob cuts airodump short when it has collected 500,000+ packets.

1:19 pm Aircrack successfully gives Bob the WEP key.

1:25 pm Bob reboots into windows on his laptop to run Cain & Abel.

1:30pm Andy comes back from lunch to discover the wireless intrusion.

1:35pm Andy pings Bob's machine and determines he is still connected.

1:40pm Bob finds the open shares on SERVER and dumps them into his hard disk.

1:45pm Bob finishes copying the files and immediately drives off.

1:50pm Andy finally receives written confirmation from Alan to take down the wireless lan.

2:00pm Andy moves into the containment phase of the Incident Handling Process.

Containment

At this point Andy has realized his network has been compromised but he is not sure exactly what has been lost or stolen. He is able to ping the address 192.168.0.102.

This suggests the attacker is still connected to the “merchem” AP. After a very brief discussion with his junior technicians, he decides the best way to contain the situation would be to shut down the Wireless Access point before the attack can do even more damage.

The first person on Andy’s contact list is his immediate supervisor, Alan. He asks for permission to shut down the “merchem” wireless AP. Alan realizes the severity of this incident and immediately authorizes the shutdown of the “merchem” AP in writing.

Now that the attacker’s initial entry vector has been shut down, Andy can concentrate on finding out what was changed or stolen.

By filtering all packets with a source and destination of 192.168.0.102, he is able to reconstruct which hosts the attacker accessed. Andy is certain that the attacker spent the most time accessing the Mercury Chemicals host named SERVER. He also looks at the SMB traffic to see which file shares were accessed.

Eradication

Andy however rescans each of his hosts for evidence of tampering or backdoors. He restores SERVER from the last full backup image to ensure that the host integrity is restored. Unfortunately at this stage it is looking like Andy is closing the barn door after the horse has bolted. He can do nothing to recover the stolen data. Andy has to ensure that open network shares are not allowed on his network.

Recovery

There are many things that Andy has to implement to ensure his wireless network is not penetrated in the same manner. Firstly if possible to drop the WEP encryption scheme in favor of WPA. This scheme is much more secure and there exist no foolproof attacks against it at the moment. Andy will harden his wireless network by placing his access point behind a firewall.

Lessons Learnt

- Don't name your SSID after your company.
- Keep up to date on the latest attacks.
- Do not keep open network shares and assume the internal network is safe.
- Assume that all wireless traffic is insecure by placing the Access Point inside the DMZ.
- A further layer of security can be added by implementing a VPN authentication for all wireless clients.

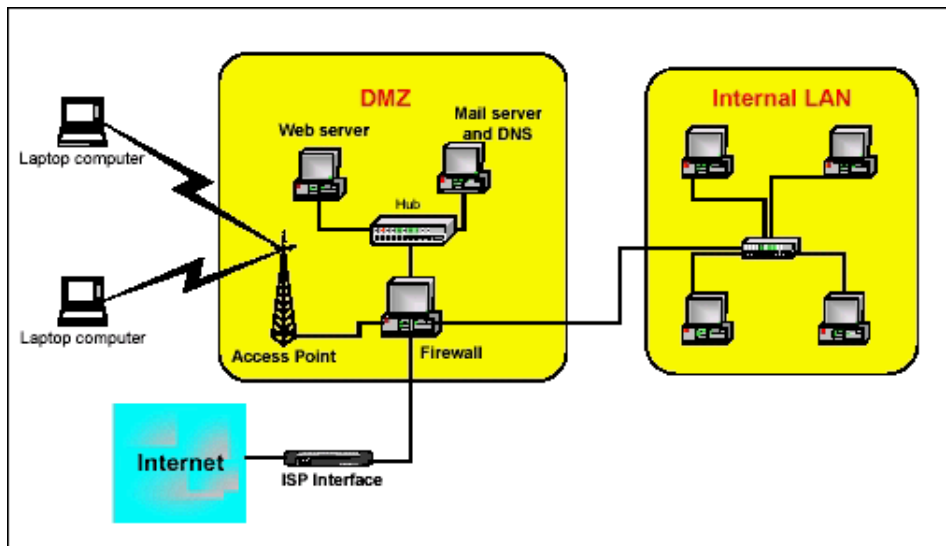


Figure 18: Correct Placement of the Wireless AP in the DMZ²³

²³ Source: Ernst & Young, Extreme Hacking Coursenotes , August 2004

List of References

Hulton, David, Practical Exploitation of RC4 Weaknesses in WEP Environments, February 2002, <http://www.dachb0den.com/projects/bsd-airtools/wepexp.txt>

Lam, Steve, Wireless Hacking, August 2004, Ernst & Young Extreme Hacking Courseware

Ossman, Michael, WEP: Dead Again Part I, Securityfocus article, Sept 2004, <http://www.securityfocus.com/infocus/1814>

Ossman, Michael, WEP: Dead Again Part II, Securityfocus article, Mar 2005, <http://www.securityfocus.com/infocus/1824>

Works Cited

¹ Cisco Systems Inc., A Comprehensive Review of 802.11 Wireless LAN Security and the Cisco Wireless Security System, White Paper, 2002, Pg 21

<http://airsnort.shmoo.com/>

<http://wepcrack.sourceforge.net/>

<http://www.netstumbler.org/showpost.php?p=89692&postcount=22>

<http://www.netstumbler.org/showthread.php?t=11878>

<http://www.cr0.net:8040/code/network/aircrack/>