Interested in learning more about security?

SANS Institute InfoSec Reading Room

Understanding Wireless Attacks and Detection

This paper introduces wireless attacks from a OSI layer 2 perspective and attempts to understand how wireless attacks can be detected by looking at wireless frames at these layers.

Understanding Wireless attacks & detection

GIAC Security Essentials
Certification (GSEC)
Practical Assignment
Version 1.4c

Option 1 - Research on Topics
in Information Security

Submitted by: Christopher Low 13 April 2005

Paper Abstract: This paper introduces wireless attacks from a OSI layer 2 perspective and attempts to understand how wireless attacks can be detected by looking at wireless frames at these layers.

Current wireless IDS detection techniques and technologies are then introduced and evaluated for its effectiveness in light of these attacks.

Table of Contents

List of Figures

Executive Summary

Productivity gain, ease of use and deployment, work mobility are all reasons why wireless technologies have not only infiltrated the homes of many but also enterprises. Enterprise wireless deployments have become very prevalent.

The functionalities and convenience brought about by wireless technologies have also introduced risks into these traditionally “closed” networks to allow for access from anywhere where the RF signals could reach.

This paper will first start by introducing the topic of wireless technology (specifically 802.11 technology) from the wireless frame layer (OSI layer 2) perspective. We’ll be looking at wireless frames and understanding how they can be captured, dissected and analyzed.

We’ll then proceed to look at different types of wireless attacks and how those attacks can be recognized at the wireless frame layer.

Wireless IDS technologies are then introduced and we’ll be looking briefly at how effective and comprehensive they are when it comes to detecting for such forms of wireless attack.

Introduction to 802.11 wireless frames

The best way to understand how any type of network protocol work is to look at it from the packet level. I'll first be describing how we can capture wireless frames that are flying through the air and then talk about dissecting and analyzing these frames for purposes of understanding how these packets look like under normal traffic conditions.

Capturing wireless packets

Unlike traditional wired networks where network packets are transmitted along physical wires, wireless technologies use the air as the physical media when sending and receiving data packets.

Air being the physical media where wireless packets traverse, opens up whole new opportunities for anyone within the vicinity (either of the wireless client or the wireless access point) with the right devices and software to be able to capture those packets.

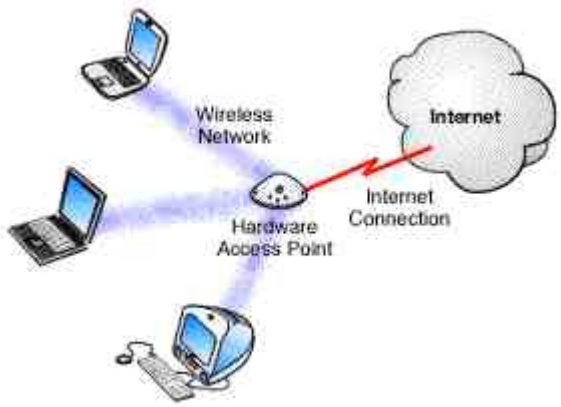

Figure 1 - Wireless packets sniffing

In order to capture wireless packets, the sniffing station will have to be equipped with the appropriate hardware and software.

The hardware required will depend on the wireless network being targeted and they come in the form of a wireless network interface card (CF, PCI, USB, PCMCIA based & on-board wireless NICs/chipsets are available). Refer to http://www.linux-wlan.org/docs/wlan_adapters.html.gz for a list of wireless NICs as well as their corresponding chipsets.

Depending on the device drivers as well as the capability of each chipset, they might differ in support when it comes to the software that we're about to introduce for purposes of sniffing these traffic.

The software, which we'll be using for purposes of wireless traffic sniffing in this research, is Kismet (http://www.kismetwireless.net). And to assist in our analysis of the sniffed packets, we'll be using ethereal (http://www.ethereal.com).

Kismet is an 802.11 layer 2 wireless network detector, sniffer, and intrusion detection system. Kismet will work with any wireless card that supports raw monitoring (rfmon) mode. It can sniff 802.11b, 802.11a, and 802.11g traffic.

Kismet can be installed very easily on any unix based operating system and can also be executed in Windows running CYGWIN (http://www.cygwin.com).

Under normal circumstances when your wireless NIC are sending and receiving wirelessly, they're placed into a mode called managed mode, and in this mode, the wireless NIC will not pick up any wireless packets which are not destined for it, thus defeating the purposes of being a sniffing station. In order for the wireless NIC to pick up all packets regardless of who the packet is for, the NIC will have to be placed in rfmon mode. The problem with running Kismet in Windows boils down to the difficulty in getting the wireless NIC into rfmon mode as most publicly available Win32 drivers don't support the rfmon mode by default.

Packets captured by Kismet can be saved into pcap files, which are then analyzed by ethereal by opening those files in an offline mode.

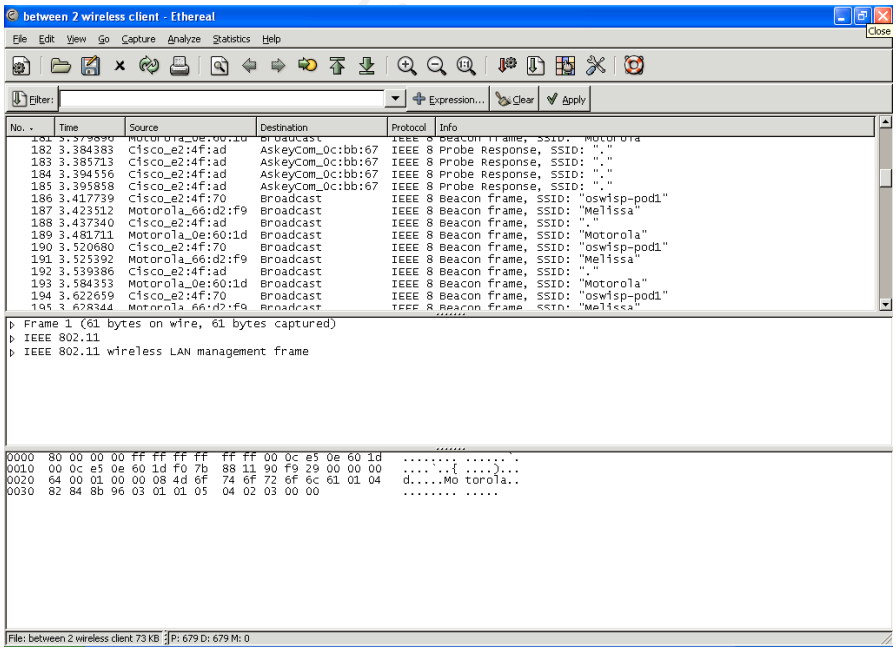

Figure 2 - Opening Kismet capture using ethereal

Dissecting wireless packets

Let's now extract some common wireless packet types and observe how wireless communication is made possible.

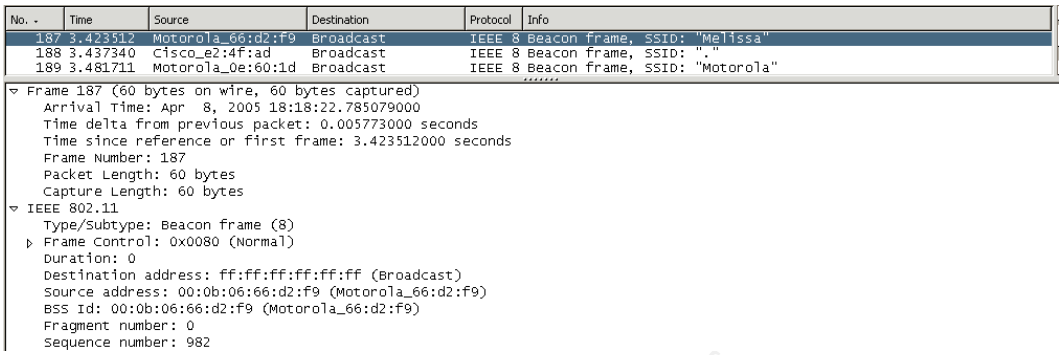

Figure 3 - Beacon frame

One of the most common wireless frames you'll see whenever you do any kind of wireless sniffing would be beacon frames. Figure 3 shows an example of a captured beacon frame.

A beacon frame is a packet send by a wireless access point (on a regular basis) in a wireless infrastructure mode operation to allow wireless clients within the vicinity to detect the Station Set Identifier (SSID) of the wireless networks. SSID defines the name of the wireless network that all the wireless clients associate with.

The first address field in the beacon frame is the **destination address**. This field has a value of "ff:ff:ff:ff:ff:ff", this value indicates that the packet is to be send to all stations. The third address field is the **BSS ID** (Basic Station System ID) field which contains the MAC address for the wireless side of the access point.

Using the filter mechanism in ethereal, we can effectively select packets belonging to this wireless network by simply keying in the following filter string "wlan.bssid==***bssid value***" in the filter field in ethereal (see figure 4 below).

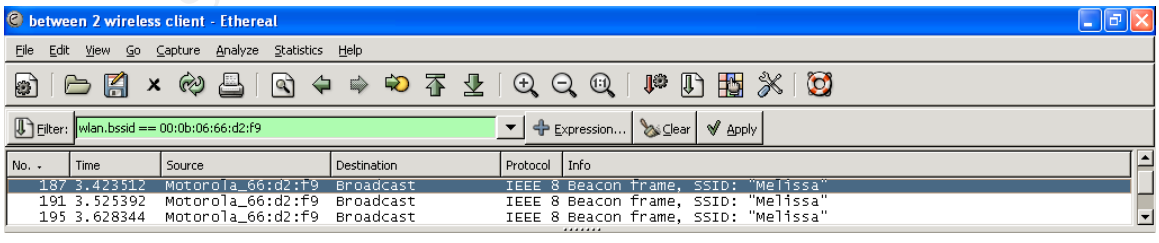

Figure 4 - Select all packets from a particular BSSID

Another field to note in this packet is the **sequence number** field; this field is incremented by one every time the wireless station emits a packet. In figure 5

below, we see that the sequence number for this beacon frame packet is 982. Therefore the next packet that will be emitted by this wireless station would then be 983.

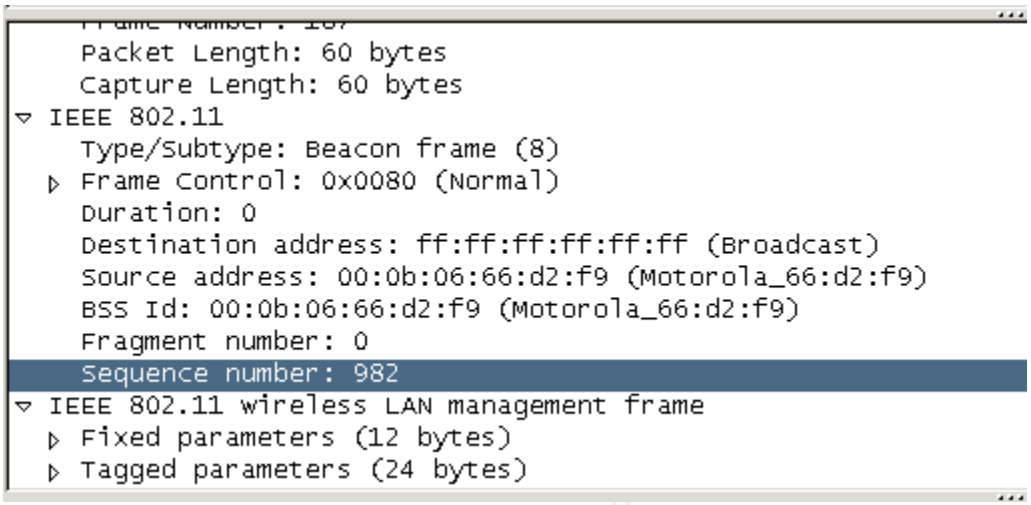

Figure 5 - Frame sequence number

Under the tagged parameters, the SSID of this particular wireless LAN can then be found, “Melissa” in this example. See figure 6 below.

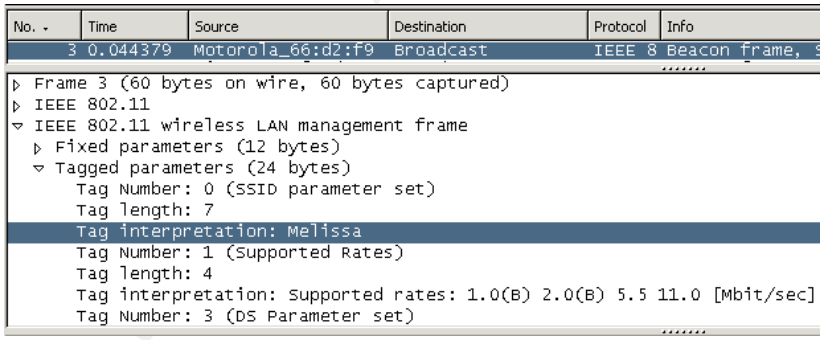

Figure 6 - Capturing the SSID value

In figure 7 below, we observer that the SSID value of this particular wireless LAN is cloaked. This can be setup in various Access points to prevent the SSID from being broadcast. This is usually recommended as a best practice so as to prevent casual wardrivers from picking up this network using war driving software.

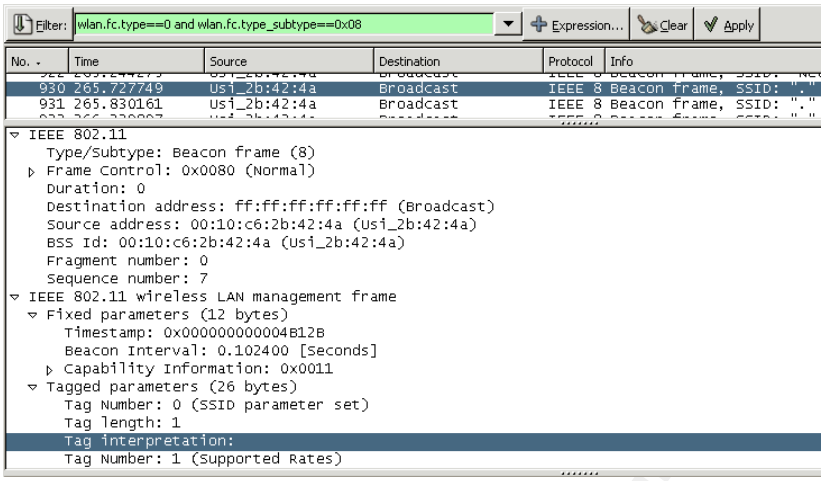

Figure 7 - SSID is cloaked

Let's now look at how a typical wireless client would connect to the wireless AP. In this example below, a wireless client is trying to connect to a predefined SSID configured and 6 packets that are shown below includes a probe request by the wireless client followed by a probe response, 2 authentication packets (Open Authentication is engaged) as well as 2 association packets. After these 6 packets are exchanged, the wireless client can start sending and receiving packets over this wireless network.

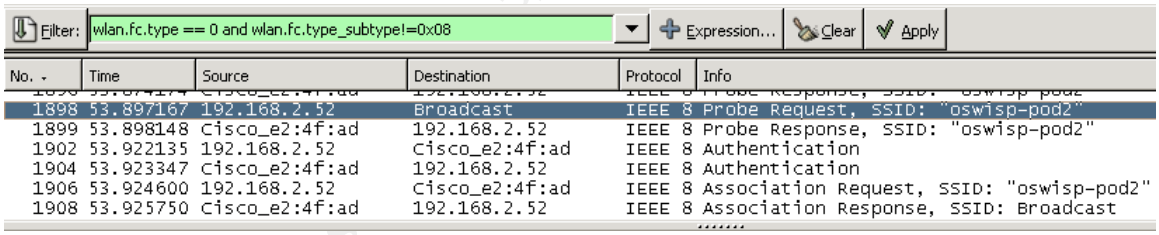

Figure 8 - Normal wireless client association traffic

The wireless client can subsequently send and receive packets over the wireless network as depicted in the following diagram.

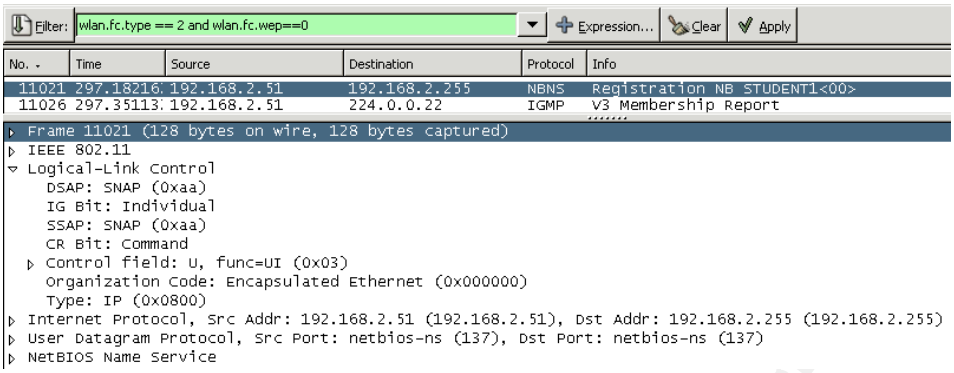

Figure 9 - A wireless data packet

We can see from the above diagram that the wireless client is sending a netbios name service broadcast message over UDP/IP. The Logical Link Control section of the frame contains information about the encapsulated higher protocol (in this example IP protocol).

After looking at a normal packet without any form of encryption, let's look at how a WEP encrypted packet would look like.

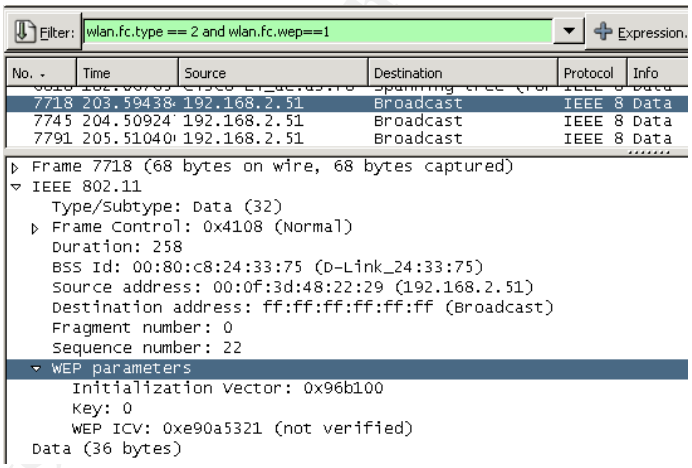

Figure 10 - A WEP encrypted packet

You'll notice that the information beyond the IEEE 802.11 header is decoded by ethereal as Data. If you open up the IEEE 802.11 header, you'll notice that there is a WEP parameter section where we can find the 24 bit Initialization vector (IV) used for this particular packet encryption as well as the ICV (used for ensuring data integrity). One other piece of information that is not encrypted and can be inspected by all is the MAC addresses of both communicating parties.

Wireless Attacks

Probing & Network Discovery

Before an attacker is able to attempt any kind of wireless mischief, one of the first activities would be for him to identify the various wireless targets in range. Probing and network discovery type attacks described in this section are amongst the first activities engaged by any attacker

There are primarily 2 main types of probing, active and passive probing. Active probing involves the attacker actively sending probe requests with no SSID configured (very much like a normal wireless client would do) in order to solicit a probe response with SSID information and other information from any access points in range. Active probing cannot detect for access points that are cloaked (configured not to respond to probe requests with no SSID set) or out of range of the attacker's wireless transmission range.

When an attacker engages in passive probing, he is listening on all channels for all wireless packets send and receive without sending even a single packet, thus the detection capability is not limited by his transmission power. But, cloaked APs with no wireless activities during the period of the probe would not be detected.

A good example of a tool that uses active probing is NetStumbler. Kismet (shown in Figure 11) on the other hand is an example of a passive probing tool.

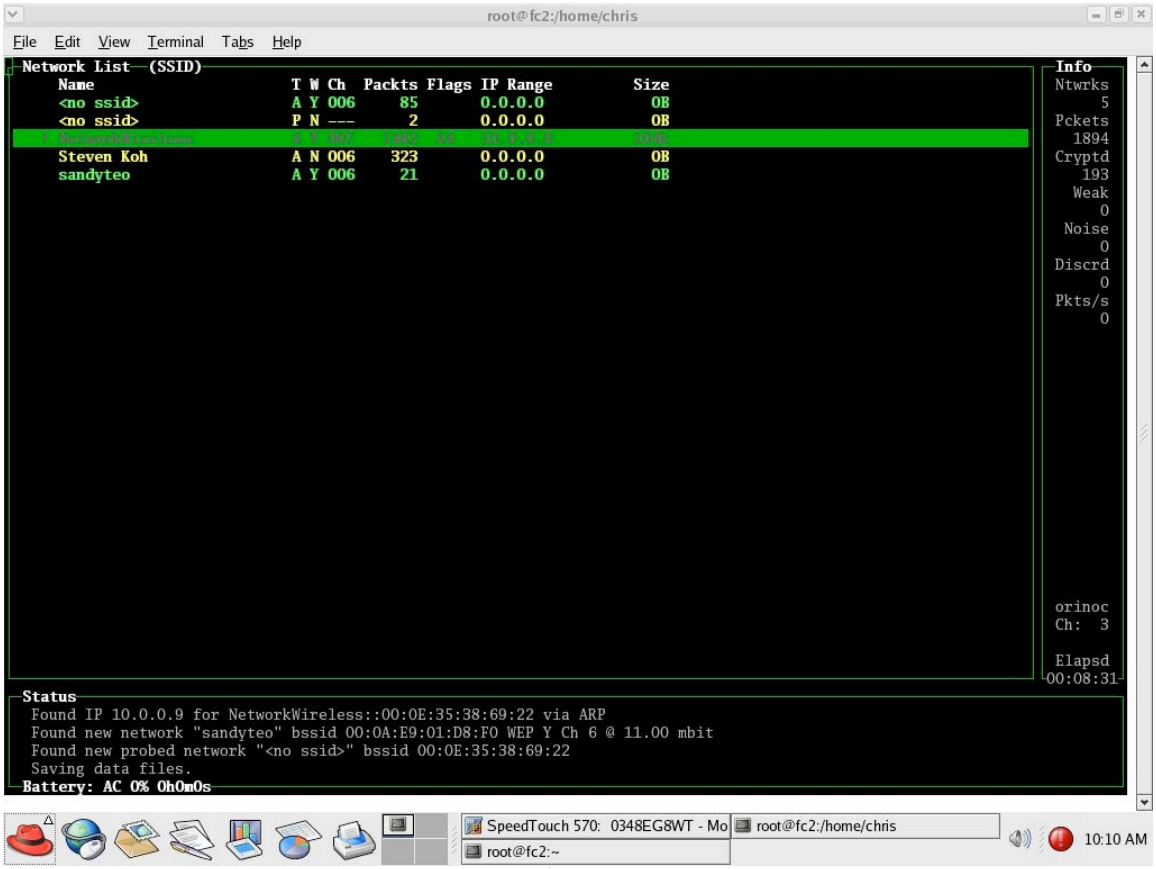

Figure 11 - Kismet

Surveillance

Once the wireless target has been identified, the attacker can proceed to gather information about the network using tools like kismet or airodump (comes with aircrack package at http://www.cr0.net:8040/code/network/). Data gathered can be saved into pcap format for subsequent offline analysis.

If the traffic stream is not encrypted, immediately the attacker could look at the traffic stream and identify the network parameters (e.g. MAC address, IP address range, gateway etc) from the traffic.

If the traffic stream is WEP encrypted, there are also WEP crackers which are available for him. In this case, airodump is used to gather all the encrypted packets transmitted and aircrack (see Figure 12) is then used to try to crack the WEP key given enough WEP IVs that are gathered.

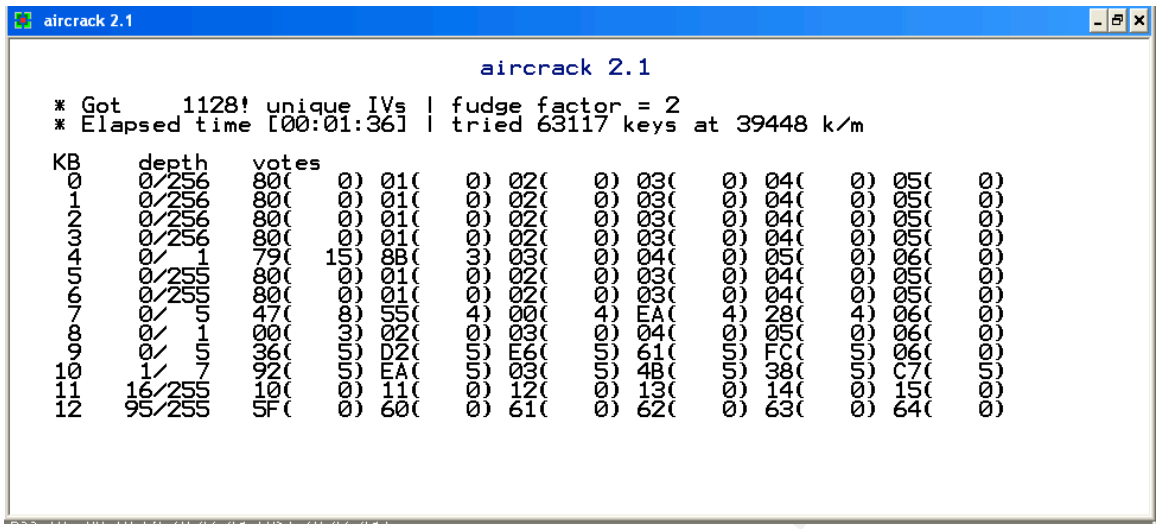

Figure 12 - Aircrack used to crack WEP key

In cases where there isn't sufficient traffic on the network, packet injection tools like WEPWedgie (http://sourceforge.net/projects/wepwedgie/) can be employed to insert arbitrary traffic into the WEP encrypted network. This will solicit responses from the network, which can then be collected and send for WEP key cracking. This is made possible because in WEP implementation, as long as one obtains the keystream used for the XOR operation using a single IV value, one can effectively reuse the same IV for all subsequent communications. To obtain a single keystream that corresponds to a particular IV, one has to look for a known plaintext and a corresponding ciphertext in the network. Using that, one can then perform an XOR operation to obtain the keystream used to encrypt the packet. One example where you'll find the plaintext and its corresponding ciphertext is when a wireless client is authenticating to the access point using shared key authentication.

Even when you are unable to get the shared key authentication traffic on the network, tools like chopchop (http://www.netstumbler.org/showthread.php?t=12489) which makes use of the access point to help it decrypt 1 WEP encrypted packet at a time without knowing the WEP key is available.

DOS attacks

DOS type attacks at layer 1 as well as layer 2 are easily executed in a wireless network. Emitting a very strong RF interference on the channel in which the wireless network is operating on will increase the noise on that channel and thus causing interference to all wireless networks that are operating at / near that channel.

Layer 2 type DOS attacks come in the form of packet injection, where the attacker will flood wireless clients who are already attached to the wireless networks with disassociate or deauthenticate packets. Example of such a tool is Void11 (http://www.wi-foo.com/index-3.html).

Impersonation

Another category of attacks that can be easily executed in a wireless network is the impersonation attack. In such an attack, the attacker changes his MAC address to a MAC address which he found earlier during the surveillance stage. This MAC address would most definitely belong to an authorized client in the network. This is usually done to defeat the MAC filtering capabilities of access points where only a list of authorized MAC addresses are allowed to use the wireless network.

As we have earlier described, even if the wireless network is WEP encrypted, the MAC address of the sending and receiving party is still viewable by a wireless sniffing tool.

Changing the MAC address of your wireless NIC card is made easier by the fact that some client software that comes with the NIC card actually allows the user enter their desired MAC address.

To change the MAC address manually in linux, use the ifconfig command with hw ether as shown below.

```
[root]# ifconfig eth0 hw ether 01:02:03:04:05:06
[root]# ifconfig eth0
eth0 Link encap:Ethernet HWaddr 01:02:03:04:05:06
inet addr:192.168.0.1 Bcast:192.168.0.255 Mask:255.255.255.0
UP BROADCAST MULTICAST MTU:1500 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:4 dropped:0 overruns:0 carrier:4
collisions:0 txqueuelen:100
RX bytes:0 (0.0 b) TX bytes:168 (168.0 b)
Interrupt:11 Base address:0xdf00 Memory:df9ff000-df9ff038
```

Figure 13 - MAC spoofing in linux

To change the MAC address manually in Windows, locate the registry settings for your wireless NIC and add a new string call NetworkAddress with the new MAC address information to it. (See figure 14 & http://www.nthelp.com/NT6/change_mac_w2k.htm for details)

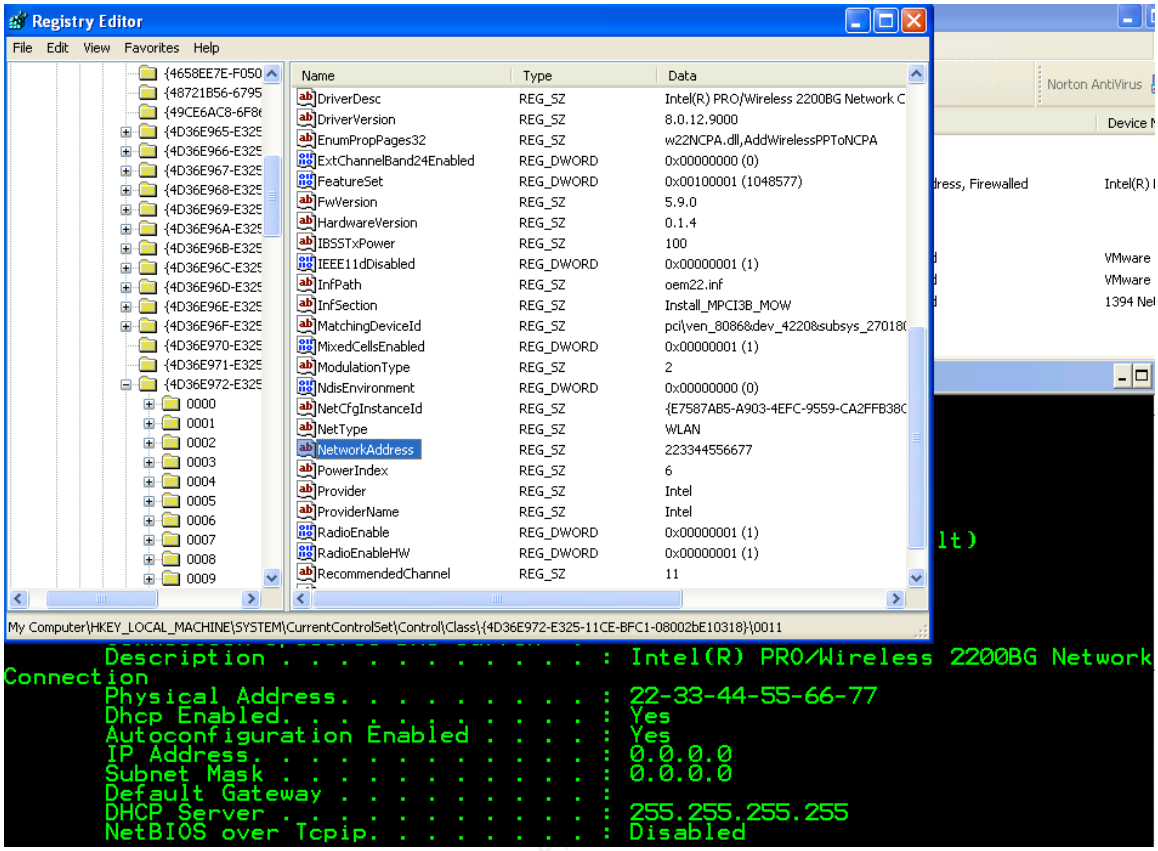

Figure 14 - MAC spoofing in Windows

Man in the middle and Rouge AP

In this type of attack, the attacker attempts to insert himself in the middle of a communication for purposes of intercepting client's data and could potentially modify them before discarding them or sending them out to the real destination.

In order to insert oneself in the middle of the communication, one has to accomplish 2 tasks, first, the legitimate AP serving the client must first be brought down or made "very busy" so as to create a "difficult to connect" scenario for the wireless client, secondly, the attacker must setup an alternate rouge AP with the same credentials as the original for purposes of allowing the client to connect to it.

Either RF interference or layer2 packet flooding as described above in DOS attack section can accomplish the first task.

The second task can be accomplished by setting up a rouge AP that will take over the tasks of the failed AP. Tools like monkey_jack (part of airjack tool found on http://www.wi-foo.com/index-3.html) could be employed.

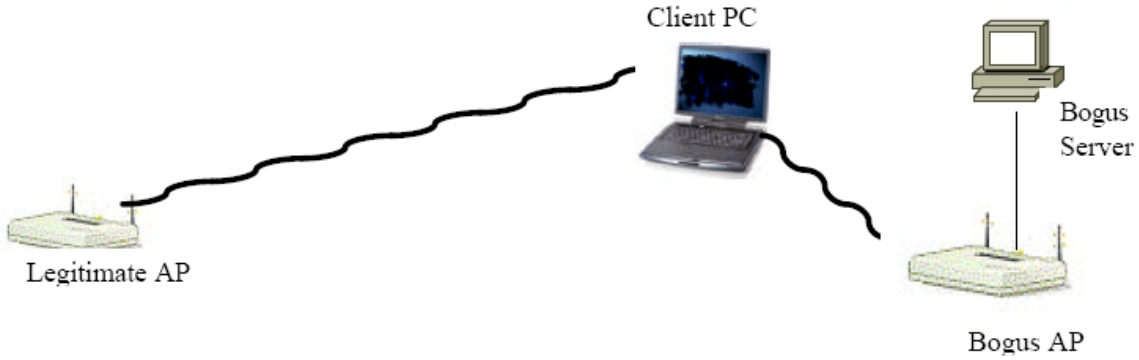

Figure 15 - Man in the middle

Wireless Attacks Detection Techniques

Let's now turn our focus towards wireless attacks detection techniques and look at what should be in place in order to sufficiently detect all of the above mentioned types of attacks.

Access Point Monitoring

This would typically entail the owner of the wireless network having a list of authorized AP equipment with their respective SSID, MAC address, channel information recorded down as a baseline. The monitoring component would then listen to wireless frames (beacons, probe response and authentication / association frames etc) send out by all its AP and compare these information to the pre-recorded information. The monitoring device should listen to all possible channels and record all packets for this technique to be effective.

In the case of a Man-in-the-middle attack, such a component would detect that there is a sudden introduction of an AP on another channel previously not present. Though the SSID, MAC address might be spoofed by the attacker in the process of setting up the rouge AP, the channel in which the genuine AP was operating from has been changed provides an alert on a possible MITM attack.

Wireless Client Monitoring

Unlike APs, it would not be possible to have a list of "allowed" client information baseline without incurring a whole lot of administrative overheads, nevertheless several aspects of the wireless clients can be monitored

Firstly, the owner could keep a "blacklist" of wireless clients that would be checked against all connecting clients, any client within this list trying to access the network would be automatically denied and an alert send off.

Secondly, all wireless clients with an "illegal" MAC address (MAC address ranges which have not been allocated out yet) be automatically denied access and an alert send off.

Thirdly, wireless client that just sends out probe requests or wireless clients that send out special distinguishable data packets after the initial probe request (E.g. Netstumbler sends out a special data packet to solicit AP's nickname after it receives a probe response from its AP, see http://www.kismetwireless.net/cgi-bin/ezmlm-cgi?mss:366:eafojgdoalggkiopbclf) but does not associate / authenticate within a certain period of time can be flagged out as potential network discovery activities.

One more area where monitoring might apply is when WEP traffic is being send/receive, no station should be reusing the same IV over and over again within a very short period of time (we've seen this earlier when we describe the use of WepWedgie to generate traffic on your WEP enabled network for purposes of cracking your WEP key)

Lastly, for wireless clients that have been authenticated and associated, the sequence number field within the IEEE 802.11 header can be tracked for sudden changes. Usually when impersonation attacks are underway, the attacker will take on the MAC / IP address of the victim, but it will not be able to continue with the sequence number used previously by the victim, thus by monitoring the sequence number in these client generated packets, potential impersonators could be identified.

General Wireless Traffic Monitoring

Wireless traffic can be monitored for attempts to flood the network using de-authentication, de-association, authentication, association, erroneous authentication (as implemented by Fatajack).

Frequency and Signal-To-Noise Ratio monitoring could help signal an oncoming RF based DOS attack on your wireless network.

Failures in authentication as well as association can also be monitored and reported.

Wireless IDS

We'll look at some examples of open source wireless IDS and evaluate their effectiveness in employing the above mentioned detection techniques.

snort-wireless

Snort-wireless is a wireless intrusion detection system adapted from the snort IDS engine. Hosted at http://www.snort-wireless.org/, it adopts the similar syntax when it comes to writing snort-wireless rules as the famous snort IDS.

Replacing the source and destination IP addresses in the normal snort rules with source and destination MAC addresses, one can write snort-wireless rules for detecting wireless traffic like one would detect for IP layer attacks.

As at this point, there are quite a bit of to-do items under the future development. These items are required (as highlighted in the previous section) to effectively be able to address some of the common threats in the wireless world.

WIDZ

This wireless IDS is build by Loud Fat Bloke (Mark Osborne). Hosted at http://www.loud-fat-bloke.co.uk/tools.html. The version at the time of this writing is at 1.8. It has the following modules :

a. Unauthorized AP monitor – responsible for detecting bogus & rouge APs by checking an AP scan result with a baseline file of all authorized APs.
b. 802.11 Traffic monitor – includes probe / flood monitoring as well as MAC and ESSID blacklist and whitelist.

Though addresses more areas, still there are multiple areas of wireless intrusion detection techniques not addressed at this version.

AirIDS

AirIDS is a wireless intrusion Detection System which is hosted at http://www.internetcomealive.com/clients/airids/general.php. It presents a number of interesting aspects to wireless IDS. First of all, like any other IDS, a robust and powerful rules file controls filtering, which is user definable. Also, it is able to forge frames so as to provide not just detection but active defenses against malicious 802.11(b) activities.

Conclusion

To sum it all up, wireless technologies have matured to a stage where it has become very common deployment. Wireless attacks are also evolving as the security standards evolved. Current intrusion detection tools have not matured to a stage where detection is sufficiently reliable.

This paper looks primarily at Layer 1 and Layer 2 type attacks for wireless LAN, but once that is breached, the attacker will then employ traditional types of attack strategies to attack higher layer protocols and applications. Thus defenders ought to look at defense in depth strategies other than just concentrating their efforts in the Layer 1 & 2 type defenses.

As the 802.11i standard is being finalized and rolled out to vendor products, we can expect more attacks to move towards the authentication as well as the encryption technology deployed in these new standards.

In the meantime, enterprises seeking to deploy wireless technologies for whatever reason should stay aware of current standards, software and hardware releases so as to better mitigate the risks brought about by these wireless deployments.

References

Andrew A. Vladimirov, Konstantin V. Gavrilenko, and Andrei A. Mikhailovsky. Wi-Foo: Addison Wesley, 2004

Rob Flickenger. Wireless Hacks: O'Reilly, 2003

Chris Hurley, Michael Puchol, et al. WarDriving: Drive, Detect, Defend: A Guide to Wireless Security: Syngress Publishing, 2004

Wright, Joshua."Layer 2 Analysis of WLAN Discovery Applications for Intrusion Detection".13 Apr.2005.< http://home.jwu.edu/jwright/papers/l2-wlan-ids.pdf >

Brenner, Pablo."A Technical Tutorial on IEEE 802.11 Protocol".13 Apr 2005.< http://www.sss-mag.com/pdf/802_11tut.pdf >

Loud Fat Blokes. 802.11 Home Page. 13 Apr. 2005 < http://www.loud-fat-bloke.co.uk/w80211.html >.

Ossmann ,Michael. " WEP: Dead Again, Part 2" Securityfocus.8 Mar.2005.13 Apr.2005 < http://www.securityfocus.com/infocus/1824 >.

Wireless LAN Security (802.11) Wardriving & Warchalking. 13 Apr. 2005 < http://802.11-security.com/security/tools >

AbsoluteValue Systems, Inc. "WLAN Adapter Chipset Directory".2 Feb.2004. 13 Apr 2005. < http://www.linux-wlan.org/docs/wlan_adapters.html.gz >

Kismet. Home Page. 2 Apr.2005.13 Apr.2005.< http://www.kismetwireless.net >

Ethereal. Home Page. 11 Mar.2005. 13 Apr.2005.< http://www.ethereal.com >

CYGWIN. Home Page. 2 Apr.2005. 13 Apr.2005.< http://www.cygwin.com >

Networking Tools. Aircrack. 13 Apr.2005.< http://www.cr0.net:8040/code/network/ >

WebWedgie. Home Page. 13 Apri.2005.< http://sourceforge.net/projects/wepwedgie/ >

Korek."chopchop experimental WEP attacks". Netstumbler Forum. 13 Apr.2005. < http://www.netstumbler.org/showthread.php?t=12489 >

Lai, Kyle.” Changing the MAC address in W2K and XP”. 13 Arp.2005.< http://www.nthelp.com/NT6/change_mac_w2k.htm >

WI-FOO Tools list. Home Page. 13.Apr.2005.< http://www.wi-foo.com/index-3.html >

Craik, Mike.”All Your 802.11b Are Belong To Us (Netstumbler Signature).”.Kismet Forum. 13 Apr.2005.< http://www.kismetwireless.net/cgi-bin/ezmlm-cgi?mss:366:eafojgdoalggkiopbclf >

Snort-wireless.Home Page.13 Apr.2005.< http://www.snort-wireless.org/ >

AirIDS. Home Page. 13 Apr.2005.< http://www.internetcomealive.com/clients/airids/general.php >

Last Updated: February 22nd, 2018

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

CyberThreat Summit 2018	London, GB	Feb 27, 2018 - Feb 28, 2018	Live Event
SANS London March 2018	London, GB	Mar 05, 2018 - Mar 10, 2018	Live Event
SANS Secure Osaka 2018	Osaka, JP	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS San Francisco Spring 2018	San Francisco, CAUS	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS Paris March 2018	Paris, FR	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS Secure Singapore 2018	Singapore, SG	Mar 12, 2018 - Mar 24, 2018	Live Event
SANS Northern VA Spring - Tysons 2018	McLean, VAUS	Mar 17, 2018 - Mar 24, 2018	Live Event
ICS Security Summit & Training 2018	Orlando, FLUS	Mar 18, 2018 - Mar 26, 2018	Live Event
SANS Munich March 2018	Munich, DE	Mar 19, 2018 - Mar 24, 2018	Live Event
SEC487: Open-Source Intel Beta One	McLean, VAUS	Mar 19, 2018 - Mar 24, 2018	Live Event
SANS Pen Test Austin 2018	Austin, TXUS	Mar 19, 2018 - Mar 24, 2018	Live Event
SANS Secure Canberra 2018	Canberra, AU	Mar 19, 2018 - Mar 24, 2018	Live Event
SANS Boston Spring 2018	Boston, MAUS	Mar 25, 2018 - Mar 30, 2018	Live Event
SANS 2018	Orlando, FLUS	Apr 03, 2018 - Apr 10, 2018	Live Event
SANS Abu Dhabi 2018	Abu Dhabi, AE	Apr 07, 2018 - Apr 12, 2018	Live Event
Pre-RSA® Conference Training	San Francisco, CAUS	Apr 11, 2018 - Apr 16, 2018	Live Event
SANS Zurich 2018	Zurich, CH	Apr 16, 2018 - Apr 21, 2018	Live Event
SANS London April 2018	London, GB	Apr 16, 2018 - Apr 21, 2018	Live Event
SANS Baltimore Spring 2018	Baltimore, MDUS	Apr 21, 2018 - Apr 28, 2018	Live Event
SANS Seattle Spring 2018	Seattle, WAUS	Apr 23, 2018 - Apr 28, 2018	Live Event
Blue Team Summit & Training 2018	Louisville, KYUS	Apr 23, 2018 - Apr 30, 2018	Live Event
SANS Riyadh April 2018	Riyadh, SA	Apr 28, 2018 - May 03, 2018	Live Event
SANS Doha 2018	Doha, QA	Apr 28, 2018 - May 03, 2018	Live Event
SANS SEC460: Enterprise Threat Beta Two	Crystal City, VAUS	Apr 30, 2018 - May 05, 2018	Live Event
Automotive Cybersecurity Summit & Training 2018	Chicago, ILUS	May 01, 2018 - May 08, 2018	Live Event
SANS SEC504 in Thai 2018	Bangkok, TH	May 07, 2018 - May 12, 2018	Live Event
SANS Security West 2018	San Diego, CAUS	May 11, 2018 - May 18, 2018	Live Event
SANS Melbourne 2018	Melbourne, AU	May 14, 2018 - May 26, 2018	Live Event
SANS Northern VA Reston Spring 2018	Reston, VAUS	May 20, 2018 - May 25, 2018	Live Event
SANS New York City Winter 2018	OnlineNYUS	Feb 26, 2018 - Mar 03, 2018	Live Event
SANS OnDemand	Books & MP3s OnlyUS	Anytime	Self Paced