ARP Cache Poisoning with Ettercap

Mohammad Shabbir Bashir

GIAC GSEC Security Essentials Certification
Practical Assignment
Version 1.4b
Option 1

August 4th, 2003

Abstract:
This paper will provide a review and analysis of an open source sniffing and ARP Cache Poisoning tool called Ettercap. Ettercap uses the insecure ARP protocol to conduct man in the middle attacks on one or more than one targets by poisoning their ARP cache. This feature enables it to sniff passwords, instant messages, e-mails and much more on a switched local area network.

The main objective of this paper is to warn administrators of packet sniffing methods on switched networks so they can be prepared against such tools. This paper discusses installing Ettercap, its basic functionality, plug-ins that do specific functions and finally, solutions to mitigate the risks presented will be discussed.

Introduction

ARP poisoning methods can go undetected on a network if proper detection methods are not in place. Network administrators are usually aware that sniffing is possible on network segments that are connected via network hubs but not all administrators know that there are tools freely available that allow attackers to sniff on switched networks as well.
Learning about ARP cache poisoning is important for every system administrator since this is the type of attack that can go completely unnoticed for a long time if the attacker is savvy and skilled.

This type of attack does not pass the boundaries of the network because network routers block ARP Broadcasts. Firewalls block unauthorized traffic from coming inside a network thus internal network is usually considered a safe zone and generally not monitored for attack patterns.

Topics in this discussion will include:
- ARP and ARP cache poisoning;
- Differences between Hubbed Networks versus switched networks;
- Ettercap, as a sniffing tool; and
- Detection and Prevention against ARP Cache Poisoning attacks.

The references cited at the end of the article will point the reader to further resources if they want to research more about this topic.

Hubbed versus Switched LAN

To understand Ettercap's capabilities, we first need to differentiate between network hubs and switches. Hubs and switches both forward incoming network signal intended for Ethernet hosts. Hubs can be called modern repeaters, they forward all traffic to all of their ports weather the traffic is intended for hosts connected to those ports or not. It is up to the hosts to determine if the packet is

intended for them or not. So if Host A wants to send a packet to Host B, the hub will broadcast the packet to all its physical ports (even though the data is not intended for all the ports) Host C and Host D will read the packet headers and realize that the packet was intended for B and discard it.
Host B receives the frame, realizes that the packet is destined for it by reading the frame headers and passes it up its protocol stack to the corresponding application.

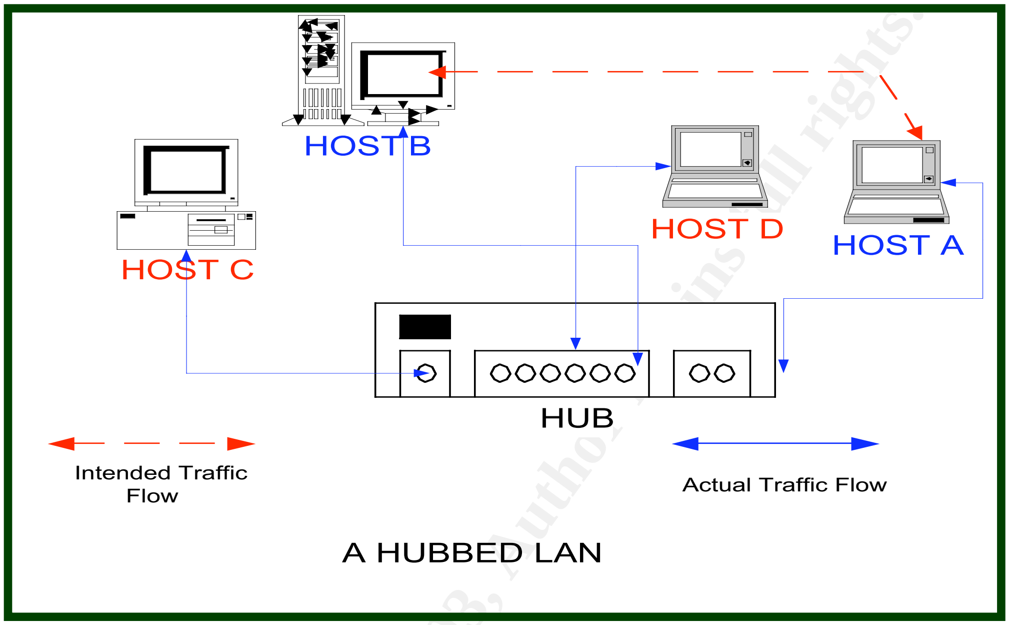
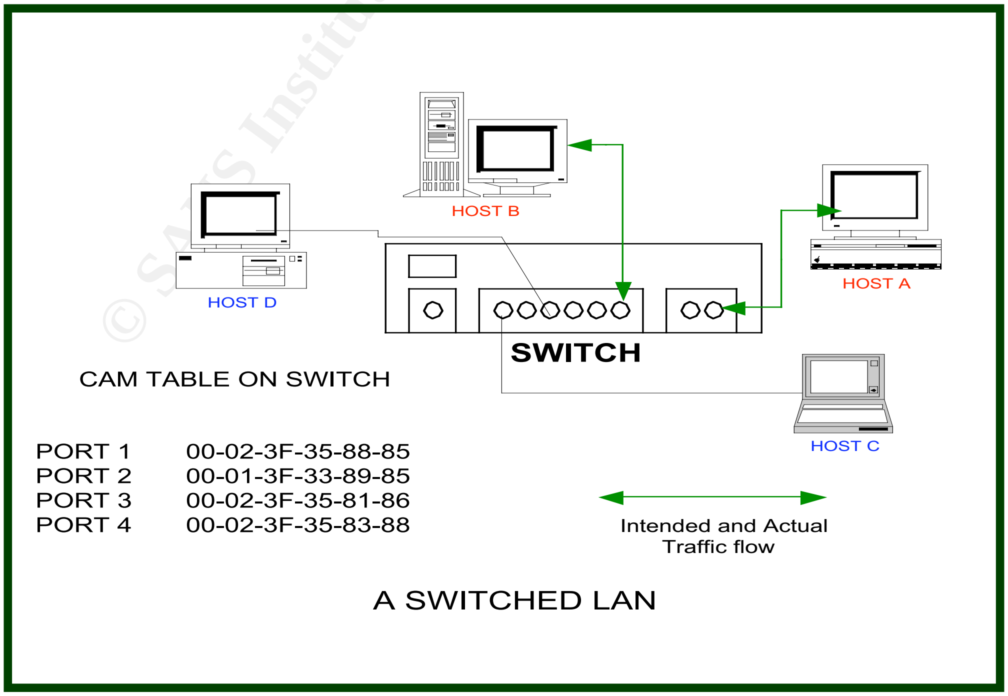


Figure 1



PORT 1	00-02-3F-35-88-85
PORT 2	00-01-3F-33-89-85
PORT 3	00-02-3F-35-81-86
PORT 4	00-02-3F-35-83-88

Figure 2

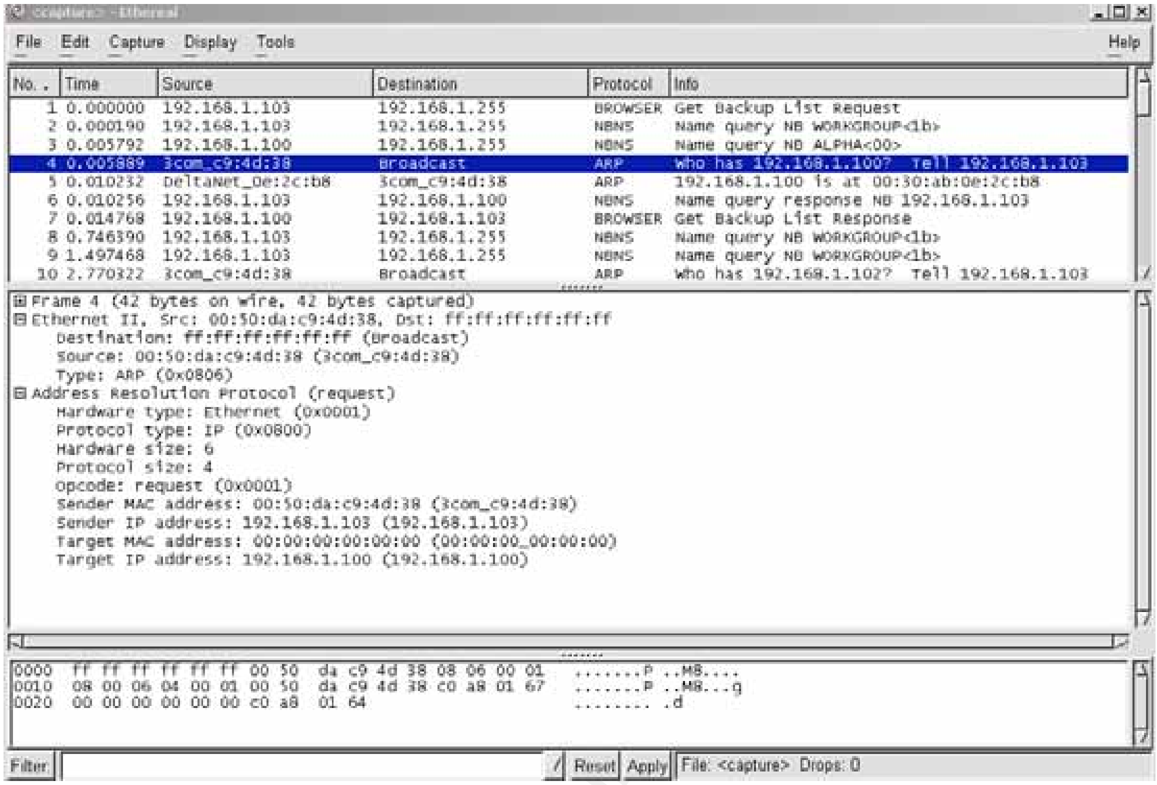
Switches however make smart decisions based on MAC addresses. They keep a CAM (Content Addressable Memory) table that lists all Ethernet addresses that are bound to each port of the switch and forward data packets based on packet headers which include source and destination MAC addresses. If Host A wants to send a packet to Host B on a switched network only host B will receive that packet, because the switch will only forward that packet to the port that Host B is connected. There are however a few exceptions to this rule, network broadcast traffic such as requests for an IP address from a DHCP server and ARP requests are passed to all hosts on all ports even on a switch. Network routers do not pass any broadcasts.

What is ARP and ARP cache?

The hardware address, or MAC address, of a machine is a unique 48-bit address that is hard coded on the Ethernet interface. It is also referred to as physical address. A Logical address such as IP address is configured via software thus is not hard coded and is subject to frequent changes.
ARP (Address Resolution Protocol) is a protocol that was designed to help Ethernet hosts resolve an Internet Address (IP) to a Media Access Control Address (MAC).
The whole process of ARP request and ARP reply goes as follows;
A network application (for example an ICMP ping request is initiated on host A, user on host A types these commands on her terminal.

```
C:\>ping 192.168.1.100
Pinging 192.168.1.100 with 32 bytes of data:
```

At this point, host A knows the IP Address of host B and recognizes that host B is on its local subnet (Host A concludes this by performing some calculations on its own IP address and compares them with Host B’s IP address scheme, this process is called the logical ANDing process) thus it does not pass the request to its default gateway, instead it sends a ARP request broadcast to the whole subnet asking “Who has 192.168.1.100, tell 192.168.1.103 (Host A)”
All hosts on the 192.168.0.X subnet will receive this broadcast, read the header, compare it to their own IP address and discard it because it is not intended for them.
Host B will realize that this broadcast is intended for it and send an ARP reply to 192.168.1.103, stating that the MAC address for 192.168.1.100 is 00:30:ab:oe: 2c:b8.



```
No. . Time      Source              Destination        Protocol Info
    1 0.000000  192.168.1.103       192.168.1.255      BROWSER  Get Backup List Request
    2 0.000190  192.168.1.103       192.168.1.255      NBNS     Name query NB WORKGROUP<1b>
    3 0.005792  192.168.1.100       192.168.1.255      NBNS     Name query NB ALPHA<00>
    4 0.005889  3com_c9:4d:38       Broadcast          ARP      Who has 192.168.1.100?  Tell 192.168.1.103
    5 0.010232  DeltaNet_0e:2c:b8   3com_c9:4d:38      ARP      192.168.1.100 is at 00:30:ab:0e:2c:b8
    6 0.010256  192.168.1.103       192.168.1.100      NBNS     Name query response NB 192.168.1.103
    7 0.014768  192.168.1.100       192.168.1.103      BROWSER  Get Backup List Response
    8 0.746390  192.168.1.103       192.168.1.255      NBNS     Name query NB WORKGROUP<1b>
    9 1.497468  192.168.1.103       192.168.1.255      NBNS     Name query NB WORKGROUP<1b>
   10 2.770322  3com_c9:4d:38       Broadcast          ARP      Who has 192.168.1.102?  Tell 192.168.1.103
```

```
⊞ Frame 4 (42 bytes on wire, 42 bytes captured)
⊟ Ethernet II, Src: 00:50:da:c9:4d:38, Dst: ff:ff:ff:ff:ff:ff
    Destination: ff:ff:ff:ff:ff:ff (Broadcast)
    Source: 00:50:da:c9:4d:38 (3com_c9:4d:38)
    Type: ARP (0x0806)
⊟ Address Resolution Protocol (request)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (0x0001)
    Sender MAC address: 00:50:da:c9:4d:38 (3com_c9:4d:38)
    Sender IP address: 192.168.1.103 (192.168.1.103)
    Target MAC address: 00:00:00:00:00:00 (00:00:00_00:00:00)
    Target IP address: 192.168.1.100 (192.168.1.100)
```

```
0000  ff ff ff ff ff ff 00 50  da c9 4d 38 08 06 00 01   .......P ..M8....
0010  08 00 06 04 00 01 00 50  da c9 4d 38 c0 a8 01 67   .......P ..M8...g
0020  00 00 00 00 00 00 c0 a8  01 64                     ........ .d
```

Figure 3: The ARP request sent via Broadcast and the Unicast ARP directly underneath it.

At his point Host A will add this IP to MAC resolution in its "ARP Cache", a table that maintains MAC addresses and their corresponding IP addresses. The ARP request was sent as a broadcast to every host on the local LAN because the MAC address for host B was not in Host A's ARP Cache. Lets take a look at host A's ARP cache before and after the ping request.

C:\>arp -a
No ARP Entries Found

At this point host A issues the ping command

C:\>ping 192.168.1.100

ARP request Broadcast was sent out
ARP reply is received

Pinging 192.168.1.100 with 32 bytes of data:
Reply from 192.168.1.100: bytes=32 time=10ms TTL=128

Ping is completed Lets see the ARP Cache of Host A now.

```
C:\>arp -a
Interface: 192.168.1.103 on Interface 0x1000003
Internet Address      Physical Address      Type
192.168.1.100         00-30-ab-0e-2c-b8     dynamic
```

We notice that both the IP and physical address of host B are now stored as an entry in the host A's ARP cache and the type of this entry is dynamic.
Now If host A needs to communicate with host B again, it will check its own ARP cache table and determine that there is no need for an IP to MAC resolution for host B and thus no time is wasted in another ARP broadcast.

There are two types of ARP cache entries: Dynamic and Static.
Dynamic entries are added and removed while normal TCP/IP connections are established, such as the ping example above. These entries have certain time limits before they expire. Different operating systems have different timeout value for ARP cache, for example the default value for NetWare 6 is 5 minutes; FreeBSD is 20 minutes; Windows NT and 2000 is 2 minutes. If these dynamic entries are used again within a certain time, they usually renew and age to a maximum life, again it all depends on the operating system.
When the IP to MAC entries are not found in the ARP cache, an ARP request broadcast is sent out.
Static ARP cache entries on the other hand are added manually by using the ARP command with the -s option. Static entries remain in the ARP cache until the computer is restarted.

<u>What is ARP Cache Poisoning?</u>

ARP Cache on a host is vulnerable to false gratuitous ARP replies. It allows an attacker to create and/or modify victims ARP cache entries. Since ARP is a stateless protocol, every time a host gets an ARP reply from another host, it thinks that at some point in the past it must have sent an ARP request and now that reply has arrived, thus accepting that ARP entry and placing it in its ARP cache.
Because of this insecure nature of the ARP protocol, all traffic can be redirected from the intended host to the unintended (attackers) host.
The process of ARP Cache poisoning goes as follows:

Host A	192.168.1.100	00-40-ab-0e-2c-b8
Host B	192.168.1.101	00-01-05-2a-1b-5a
Attacker X	192.168.1.102	00-55-02-2g-4b-6a

1, X wants to poison the ARP cache of A and B.

2, X sends an ARP reply to A that looks like this
"I am B (192.168.1.101) and my Mac Address is 00-55-02-2g-4b-6a"

3, X also sends an ARP reply to B that looks like this
"I am A (192.168.1.102) and my Mac Address is 00-55-02-2g-4b-6a"

4, A and B both accept this information and add it to their ARP Cache.

5, Now all communication between A and B will go to X, which sniffs all traffic and then forwards it to the intended hosts.

This is also referred to as the "man in the middle attack"
An Attacker can also poison these hosts by sending them completely non existing Mac addresses, which will stop them from communicating on the network completely and thus disable them to properly function as network clients, this type of attack is also referred to as a DOS, (Denial of Service) attack.

What is Ettercap

Ettercap is a very powerful packet sniffer and ARP cache poisoning tool for Unix based systems. It can perform MAC and IP based sniffing, intercept and modify packets, decrypt passwords and launch a denial of service attack against other Ethernet hosts.
Ettercap has also been ported to Windows platforms, but is highly unstable and not supported on windows. At the time of the writing of this paper, Ettercap 0.6.b is the current version of Ettercap.
Sniffing traffic on a hubbed network can be accomplished by any regular packet sniffer such as Microsoft network monitor, ethereal and tcpdump, however sniffing non broadcast packets on switched networks can be accomplished via ARP cache poisoning. Ettercap can perform the classic "man in the middle attack" by sending fake ARP replies. Ettercap is capable of capturing/decoding ssh1, HTTP, FTP, POP, SMTP and SSL passwords.

Installing Ettercap

To Install Ettercap, download the latest version from
http://ettercap.sourceforge.net
Installation is simple, switch to the directory where you downloaded Ettercap, Type

```
# tar –zxvf ettercap-0.6.b.tar.gz
```

This will create a directory in the current working directory called ettercap-0.6.b
Now type

```
# cd ettercap-0.6.b
# ./configure
```

Once configuration has been done via the automatic configure script, type

```
# make
```

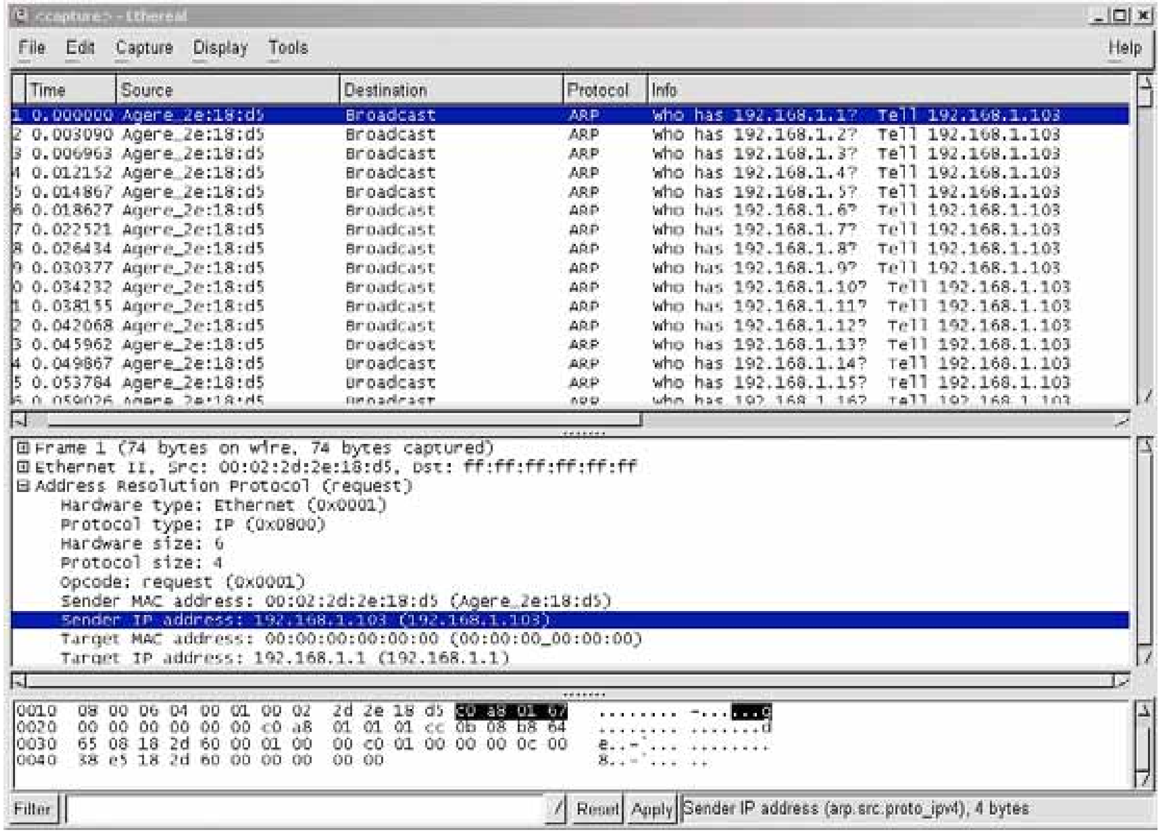
once make is done, switch to root user and type

```
# make install
```

once make install is done, type
# make plug-ins
then type
# make plug-ins_install
If you are having problems installing, refer to the forums on Ettercap website.
The default location for the Ettercap binary is */usr/local/sbin*

Running Ettercap

Ettercap can be run in a non-interactive simple mode or a colored ncurses based Interactive mode. If launched without any options, by typing ettercap at the command line, Ettercap will first determine the IP address of the machine it is launched from, then launch a broadcast of ARP requests to each host in its default subnet (this creates a MAC to IP address table for its reference) for example if its IP address is 192.168.1.1 and its default subnet mask is 255.255.255.0 then all hosts starting from 192.168.1.2 through 192.168.1.255 will be requested for their Mac addresses. (see Figure 3)



```
  Time     Source            Destination   Protocol  Info
1 0.000000 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.1?  Tell 192.168.1.103
2 0.003090 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.2?  Tell 192.168.1.103
3 0.006963 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.3?  Tell 192.168.1.103
4 0.012152 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.4?  Tell 192.168.1.103
5 0.014867 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.5?  Tell 192.168.1.103
6 0.018627 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.6?  Tell 192.168.1.103
7 0.022521 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.7?  Tell 192.168.1.103
8 0.026434 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.8?  Tell 192.168.1.103
9 0.030377 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.9?  Tell 192.168.1.103
0 0.034232 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.10?  Tell 192.168.1.103
1 0.038155 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.11?  Tell 192.168.1.103
2 0.042068 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.12?  Tell 192.168.1.103
3 0.045962 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.13?  Tell 192.168.1.103
4 0.049867 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.14?  Tell 192.168.1.103
5 0.053784 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.15?  Tell 192.168.1.103
6 0.059026 Agere_2e:18:d5    Broadcast     ARP       Who has 192.168.1.16?  Tell 192.168.1.103
```

```
⊞ Frame 1 (74 bytes on wire, 74 bytes captured)
⊞ Ethernet II, Src: 00:02:2d:2e:18:d5, Dst: ff:ff:ff:ff:ff:ff
⊟ Address Resolution Protocol (request)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (0x0001)
    Sender MAC address: 00:02:2d:2e:18:d5 (Agere_2e:18:d5)
    Sender IP address: 192.168.1.103 (192.168.1.103)
    Target MAC address: 00:00:00:00:00:00 (00:00:00_00:00:00)
    Target IP address: 192.168.1.1 (192.168.1.1)
```

```
0010  08 00 06 04 00 01 00 02  2d 2e 18 d5 c0 a8 01 67   ........ -......g
0020  00 00 00 00 00 00 c0 a8  01 01 01 cc 0b 08 b8 64   ........ .......d
0030  65 08 18 2d 60 00 01 00  00 c0 01 00 00 00 0c 00   e..-`... ........
0040  38 e5 18 2d 60 00 00 00  00 00                     8..-`... ..
```

Sender IP address (arp.src.proto_ipv4), 4 bytes

Figure 4: ARP Broadcast Storm at startup

Ettercap General Options

Some of the most common options are for running Ettercap are listed below, for more options, type man Ettercap.

- -z By giving Ettercap the –z option it wont start an ARP broadcast at startup, many Network based Intrusion Detection Systems raise red flags when an ARP Broadcast such as the one shown above in the screen shot is detected on the network.
- -b This method will start a broad ping upon startup instead of an ARP Broadcast upon startup.
- -Z <n sec> A delay of n amount of seconds when sending an ARP storm to avoid detection by IDS and smart switches, if given this option at startup, Ettercap will wait n seconds before sending each ARP requests.
- -D The delay in seconds between the ARP replies to a poisoned or victim host since different operating systems have different ARP Cache timeout values.
- -S This option is used to spoof the attackers IP address when scanning the network with ARP requests.
- -H This option, when given the arguments of IP1; IP2; IP3 will only send ARP requests to these three hosts. This is a less invasive method.
- -J This option will take a filename of the hosts as an argument that were created as a result of typing k during an interactive session so when an ARP request broadcast is not desired, Ettercap is launched with the –j <filename> option.
- -k This option is used to save the current host list which includes host names, IP's and Mac addresses.

The Non Interactive (Simple) Way

By giving Ettercap the –N argument, the tool can be launched in a simple non GUI mode, used for passive or scripted sniffing. Features such as character injecting which require user interaction are not possible in this simple mode. Some of Ettercap's options can be used only in the simple mode, below is a list of some options. Refer to the man page for more information on options, syntax and arguments.

- -t This option with the argument tcp or udp will only sniff the given protocol.
- -J This option will only poison the victims and not sniff any traffic, this option can be used if an attacker plans to sniff data via another packet sniffer such as tcp dump or ethereal.
- -R This option will sniff all the connections except the selected one, useful if you are connected to the machine running Ettercap via ssh or telnet.
- -O This option will put Ettercap in a Passive or semi Interactive state.
- -p This option with a name of the plugin as the argument will run the plug-in.

- -l This option will dump a list of all known hosts and their Mac addresses in a file named in a format similar to 20030721-Passive_Local_Report.log
- -C This option will collect user names and passwords from the hosts that were specified as arguments when Ettercap was launched.
- -f This option with the argument <host> will try to guess the Operating system of <host>
- -1 This option will dump the data in hex mode.
- -2 This option will dump the data in text mode.
- -3 This option will dump the data in ebcdic mode.
- -L This option will log all data such as host info, open tcp/udp ports, mac address, host names and if used with the –C option, even collected passwords to a file format
- -q This option will set Ettercap as a daemon, disconnected from any terminal. It must be combined with the –n –l and -c options.

The Interactive (Ncurses) Way

Ettercap can be run in an Interactive way by typing Ettercap at a shell prompt. This will start Ettercap in an Ncurses mode. This interactive mode allows almost a "point and click" based sniffing and poisoning opportunity, which makes it an attractive feature for novice attackers.
One can select and deselect victims by using the arrow keys on the keyboard and then hitting the enter key to mark them as source and destination hosts. At any point typing h on the keyboard will present a help menu which shows options that can be typed as input. Some common options are…

- Tab- Switch between source and destination.
- A – Arp Poisoning based sniffing.
- S- IP based sniffing.
- M- Mac based sniffing.
- J – Only poisoning, no sniffing.

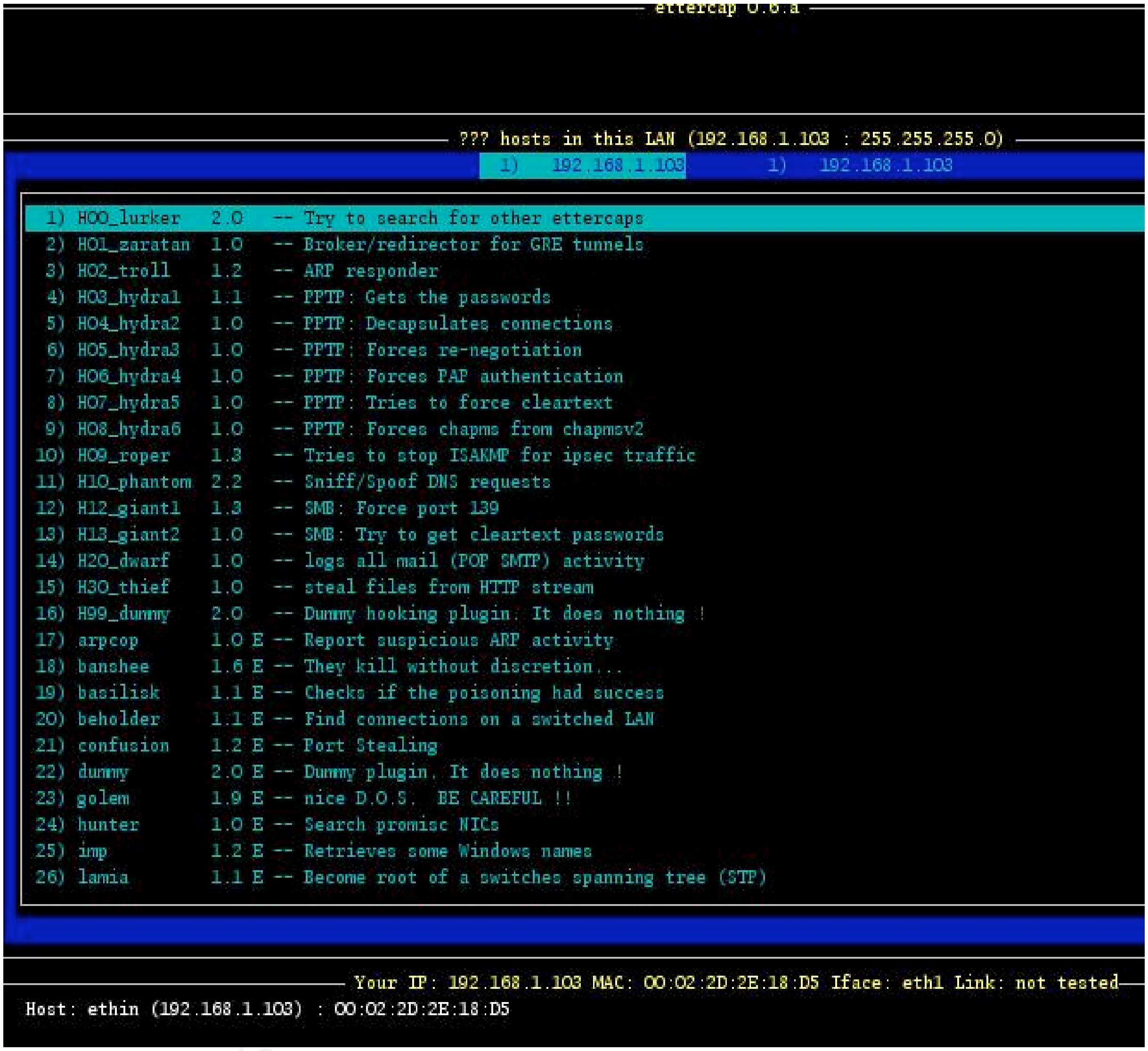
There are a few other options that can also be selected. (See Screen shot below)

Figure 5: Options in the Interactive Way

Ettercap Plug-ins

Once Ettercap is running in n-curses mode, you can type "p" to see a list of plug-ins available. If you installed the plug-ins by typing
# make plug-ins_install
during the install process, you should see a total of 32 plug-ins. Each of these has a different use. Some of the plug-ins and their uses are listed below. Plug-ins can be selected by using the up and down keys and selecting by hitting Enter on

the keyboard, an A will appear between the version of the plug-in and the description, this means that the plug-in is now active.



```
                                                          ettercap 0.6.a

                                       ??? hosts in this LAN (192.168.1.103 : 255.255.255.0)
                                         1)    192.168.1.103          1)    192.168.1.103

  1) H00_lurker   2.0   -- Try to search for other ettercaps
  2) H01_zaratan  1.0   -- Broker/redirector for GRE tunnels
  3) H02_troll    1.2   -- ARP responder
  4) H03_hydra1   1.1   -- PPTP: Gets the passwords
  5) H04_hydra2   1.0   -- PPTP: Decapsulates connections
  6) H05_hydra3   1.0   -- PPTP: Forces re-negotiation
  7) H06_hydra4   1.0   -- PPTP: Forces PAP authentication
  8) H07_hydra5   1.0   -- PPTP: Tries to force cleartext
  9) H08_hydra6   1.0   -- PPTP: Forces chapms from chapmsv2
 10) H09_roper    1.3   -- Tries to stop ISAKMP for ipsec traffic
 11) H10_phantom  2.2   -- Sniff/Spoof DNS requests
 12) H12_giant1   1.3   -- SMB: Force port 139
 13) H13_giant2   1.0   -- SMB: Try to get cleartext passwords
 14) H20_dwarf    1.0   -- logs all mail (POP SMTP) activity
 15) H30_thief    1.0   -- steal files from HTTP stream
 16) H99_dummy    2.0   -- Dummy hooking plugin: It does nothing !
 17) arpcop       1.0 E -- Report suspicious ARP activity
 18) banshee      1.6 E -- They kill without discretion...
 19) basilisk     1.1 E -- Checks if the poisoning had success
 20) beholder     1.1 E -- Find connections on a switched LAN
 21) confusion    1.2 E -- Port Stealing
 22) dummy        2.0 E -- Dummy plugin. It does nothing !
 23) golem        1.9 E -- nice D.O.S.  BE CAREFUL !!
 24) hunter       1.0 E -- Search promisc NICs
 25) imp          1.2 E -- Retrieves some Windows names
 26) lamia        1.1 E -- Become root of a switches spanning tree (STP)

                          Your IP: 192.168.1.103 MAC: 00:02:2D:2E:18:D5 Iface: eth1 Link: not tested
 Host: ethin (192.168.1.103) : 00:02:2D:2E:18:D5
```

Figure 6: List of Plug-ins shown in Interactive mode.

Detection and Prevention

Bruce Schneier, Founder and CTO of Counterpane Internet Security, Inc, in his May 2000 monthly newsletter said, "Security is a process, not a product."
ARP cache poisoning on your network can go completely undetected if appropriate measures are not taken. A skilled attacker only needs one live network jack from within the network boundaries to successfully poison any host or hosts and collect all kinds of data and passwords and leave the premises without anyone ever finding out about it.

Imagine a scenario in which an non IT employee comes to work one day with a freely available CD based Linux distribution such as Knoppix STD (Security Tool Distribution which includes Ettercap that boots and operates from a cdrom drive on any computer) and turn any machine into an ARP cache poisoning and sniffing machine. Once the attacker is done using that machine, she can eject the cd based Linux distribution, reboot the computer back into its original operating system and be on her way out with all sorts of data and passwords.

Unless there are countermeasures present, an attack like this can go unnoticed. Network administrators can minimize the risk of ARP cache poisoning on their network by using the following methods.

Monitoring Traffic

The need to monitor and analyze live traffic on the network is perplexed but crucial. Network based Intrusion Detection Systems can be set to trigger alarms if an enormous number of ARP requests such as a broadcast is initiated from any host. This should take care of the novice attacker who launches Ettercap with no options and thus sends a massive ARP broadcast upon startup.

ARP Watch and Remote ARP Watch

ARP watch is a freely available Unix based utility that monitors local ARP cache and if it discovers two hosts with the same MAC address on the network, it sends an e-mail to the root account and logs an entry into */var/log/messages*, see example below.

```
From: arpwatch@localhost.localdomain (Arpwatch)
To: root@localhost.localdomain
Subject: changed Ethernet address (bravo)

Hostname: bravo
IP address: 192.168.1.100
Ethernet address: 0:2:2d:2e:18:d5
Ethernet vendor: <unknown>
Old Ethernet address: 0:30:ab:e:2c:b8
Old Ethernet vendor: <unknown>
Timestamp: Saturday, July 26, 2003 13:47:47 -0400
Previous timestamp: Saturday, July 26, 2003 13:47:36 -0400
Delta: 11 seconds
```

```
root@localhost log]# tail -f messages
Jul 26 13:51:47 localhost arpwatch: Ethernet mismatch 192.168.1.1
0:2:2d:2e:18:d5 (0:6:25:7e:4a:7b)
Jul 26 13:51:47 localhost arpwatch: Ethernet mismatch 192.168.1.1
0:2:2d:2e:18:d5 (0:6:25:7e:4a:7b)
Jul 26 13:51:47 localhost arpwatch: Ethernet mismatch 192.168.1.100
0:2:2d:2e:18:d5 (0:30:ab:e:2c:b8)
```

```
Jul 26 13:51:47 localhost arpwatch: Ethernet mismatch 192.168.1.1
0:2:2d:2e:18:d5 (0:6:25:7e:4a:7b)
Jul 26 13:51:47 localhost arpwatch: Ethernet mismatch 192.168.1.1
0:2:2d:2e:18:d5 (0:6:25:7e:4a:7b)
Jul 26 13:51:47 localhost arpwatch: Ethernet mismatch 192.168.1.100
0:2:2d:2e:18:d5 (0:30:ab:e:2c:b8)
Jul 26 13:51:47 localhost arpwatch: Ethernet mismatch 192.168.1.1
0:2:2d:2e:18:d5 (0:6:25:7e:4a:7b)
Jul 26 13:51:47 localhost arpwatch: Ethernet mismatch 192.168.1.1
0:2:2d:2e:18:d5 (0:6:25:7e:4a:7b)
```

Remote ARP Watch on the other hand collects and compares ARP cache entries of remote devices capable of communicating via SNMP.

Ettercap

Ettercap itself can be used as an ARP cache poisoning detection tool by using the plug-in called arpcop which detects ARP Cache poisoning on the network. Since Ettercap is an open source tool, the plugin can be modified to send e-mail to a designated account upon detection of suspicious activity.

VPN

A partial solution is to deploy a Virtual Private Network (VPN) between critical devices to provide a encrypted authentication and data transfer method, however an attacker can still redirect and passively monitor the traffic by using ARP cache Poisoning, except that she wont be able to decipher the captured traffic, since all VPN based communication is encrypted. A denial of service can still be launched if bogus ARP entries are fed to victims.

Static ARP Entries

One can setup static ARP entries on hosts by giving the ARP command the –s argument so that no gratuitous ARP replies are accepted, only statically entered entries are used for communicating within the LAN. However this approach is not very practical on large networks where machines are constantly added, renamed and IP addresses are assigned by DHCP. Also not all operating systems will work flawlessly with static ARP entries, some versions of Windows operating systems will update their ARP cache entries when they receive a gratuitous ARP reply from another host even though static ARP entries were entered for that host.

Hardened Switches

Some Network switch manufacturers such as Cisco Systems allow their layer two switches to be hardened against MAC Address Spoofing or poisoning attacks, for example switches can be configured to use Mac Limiting and Mac Hard coding. Mac Limiting allows one Mac address on one port, so for example If Host X, the attacker sends Host B, a false ARP reply stating “I am IP aa.aa.aa.aa and my Mac is xx-xx-xx-xx-xx-xx” and also retain its original Mac address which is yy-yy-

yy-yy-yy-yy. A hardened switch will block all traffic to that port until the block is manually removed by the administrator.
Mac Hard coding on the other hand means that MAC address of key devices such as gateway routers are hard coded into the switches configuration so if any attacker tries to send a packet stating it is the default gateway, that port on the switch where the attackers host is connected will be disabled as a result because of the security violation.

Keeping up with security related sites and lists
Administrators are encouraged to subscribe to security related mailing lists and Ettercap's website for new detection and prevention methods. New tools come out every day that all help attackers avoid detection. On the other hand the security community discovers new prevention and detection methods regularly as well.

Conclusion

ARP spoofing and poisoning is a threat that can take network administrators by surprise if they are not aware of ARP cache poisoning and unprepared to detect unauthorized activity on their network. Tools like Ettercap can be launched on a network without detection if the attacker is smart enough to run them silently. Once the attacker has a list of IP to MAC addresses, there is nothing to stop them from sniffing passwords, sensitive information, corporate secrets, instant messages, e-mails and any or all traffic that they want to sniff. A carefully executed attack can go unnoticed for a long time.

References:

[1] Plumme, David C. "RFC 826. An Ethernet Address Resolution Protocol" Nov. 1982.
URL: http://www.ietf.org/rfc/rfc0826.txt?number=826

[2] Volobuev, Yuri "ARP and ICMP redirection games." September 1997
URL: http://www.insecure.org/sploits/arp.games.html

[3] Fleck, Bob; Dimov, Jordan "Wireless Access Points and ARP poisoning" August 2002
URL: http://www.cigitallabs.com/resources/papers/download/arppoison.pdf

[4] Fairhurst, Gorry "Address Resolution Protocol (arp)" January 2001

URL: http://www.erg.abdn.ac.uk/users/gorry/course/inet-pages/arp.html

[5] "Ettercap"
URL: http://ettercap.sourceforge.net

[6] "dsniff"
URL: http://www.monkey.org/~dugsong/dsniff/

[7] Whalen, Sean "Introduction to ARP spoofing" April, 2001
URL: http://packetstormsecurity.nl/papers/protocols/intro_to_arp_spoofing.pdf

[8] GTS Learning "The TCP/IP Transport Tutorial" April 2002
URL: http://tutorials.findtutorials.com/read/category/97/id/381/p/3

[9] Granneman, Scott " What's the Difference Between a Hub and a Switch"
URL: http://www.granneman.com/techinfo/networki/hubsswit

[10] " Sniffing Administrator's Password in Symantec Firewall/VPN Appliance V. 200R" Oct 2002
URL: http://www.securityfocus.com/archive/1/296659/2003-04-09/2003-04-15/0

[11] "Arp Watch"
URL: http://www.securityfocus.com/tools/142

[12] "Remote ARP Watch"
URL: http://www.raccoon.kiev.ua/projects/remarp/

[13] " Hack Proofing Your Network"
Russel, Ryan; Cunningham, Stace, "Hack Proofing Your Network," Syngress Publishing Inc, Copyright 2000

[14] Beekey, Mike "ARP Vulnerabilities" June 2001
URL: http://www.blackhat.com/presentations/bh-usa-01/MikeBeekey/bh-usa-01-Mike-Beekey.ppt

[15] "Knoppix STD"
URL: http://www.knoppix-std.org/tools.html

[16] Gill, Stephen "Hardening Cisco Switches" Nov 2002
URL: http://www.qorbit.net/documents/catalyst-secure-template.htm#_Toc24971116