ARP Poisoning : An investigation into spoofing the Address Resolution Protocol
By Steve Fewer

Table Of Contents

1 Introduction

In this paper I will discuss the techniques of ARP spoofing. I will detail the basics of the protocol itself before going on to talk about the different types of attack methodologies involved with ARP spoofing. Finally I will outline protective measures against any such malicious activity. In this paper I will be speaking of the ARP protocol in terms of the IPv4 network model and Ethernet so a general understanding or these technologies shall be assumed.

2 Protocol Overview

A node on an IP/Ethernet network maintains two addresses. The first is the address of the network interface card (NIC). This hardware address is known as the Media Access Controller (MAC) address. The MAC address in theory is globally unique, as it is comprised of the manufacturers serial number and model number. It is also supposed to be static, as it is stored in the firmware of the card itself. This MAC address is required to send frames of data out on an ethernet network. When a node sends a frame to another node the source and destination MAC addresses are held in the ethernet header of the frame.

The second address a node on an IP/Ethernet network has is its IP address. An IP address is a unique virtual address that is assigned via software and bound to a hardware address, e.g. a NIC. IP communicates by constructing packets of data, encapsulating any higher protocols (e.g. TCP or UDP) and transmitting them via the network layer, in our case ethernet frames.

When the ethernet frame is built to send an IP packet the ethernet header must be filled in, however ethernet will not know the destination MAC address that is bound to the destination IP address. A method of resolving this address must be available to ethernet and it comes in the form of the Address Resolution Protocol. ARP sole purpose is to resolve 32-bit logical IP addresses into their associated 48-bit ethernet hardware address.

ARP communicates via four messages. An ARP Request is a message asking to resolve a given IP address into its bound MAC address. This message is usually broadcast to all hosts on a network via the ethernet broadcast address. Every receiving host will examine the request to see if it is assigned the specified IP address and if so will respond with an ARP Reply telling the requesting host its MAC address. Two more messages exist, a Reverse ARP (RARP) Request and a RARP Reply. A RARP Request asks to resolve a given MAC address into its associated IP address. A RARP Reply is the response to a RARP Request giving the IP address of the associated MAC address. Often hosts maintain a cache of ARP replies to minimise the amount of ARP requests being broadcast. When a host receives an ARP reply it will update this cache with the new IP address to MAC address association. This will happen regardless of whether the host initially sent out an ARP Request, due to ARP being a stateless protocol.

3 Protocol Attacks

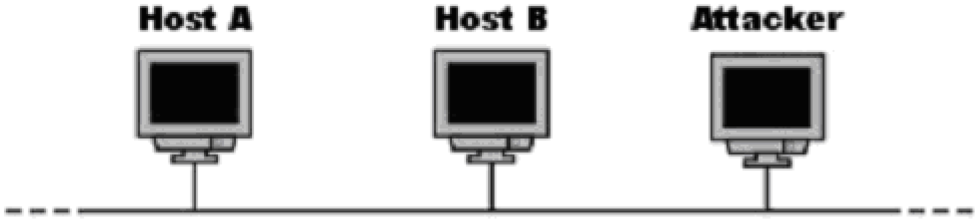
ARP spoofing is the technique of forging fake ARP messages on a network. It is possible to update a host's ARP cache with false information via spoofed ARP Replies. This technique is known as 'ARP Poisoning' and is the basis of more complex attacks.

3.1 Sniffing

Sniffing is the term used to describe the reading of all packets on a network segment. This is relatively easy on a network connected via a hub as ethernet is a broadcast medium and the attacker would only have to place his NIC in promiscuous mode to 'sniff' all traffic on that network segment. In a switched network this is not possible. This is because a switch builds a table of MAC addresses and their associated ports when the switch is powered on. When a host transmits an ethernet frame the switch examines the destination MAC address and routes the frame to the associated port as given in the switch table. Therefor it is not possible to sniff any traffic on the network.

There are two methods to sniff traffic in a switched environment using ARP Poisoning. The first is for the attacker to send multiple spoofed ARP Replies to the switch. The switch will process these replies, updating its table. If this is done at a rapid rate the switches table will overflow and the switch will default to broadcasting all traffic to all ports. The attacker can now 'sniff' all network traffic. Not all switches are vulnerable to this method as it depends on the manufacturer.

The second method involves a 'man in the middle' style attack. I will explain this technique in detail by aid of an example. Below is a simple network segment with three hosts in a switched environment.



The attacker wishes to sniff all traffic that A sends to B and visa versa. This is currently not possible as the attacker is connected to the network via a switch. The correct IP addresses and MAC addresses for each host are as follows:

Host	IP Address	MAC Address
Host A	192.168.0.2	00:00:00:00:00:02
Host B	192.168.0.3	00:00:00:00:00:03
Attacker	192.168.0.4	00:00:00:00:00:04

We can also assume that the above is true for all the hosts ARP caches.

Firstly the Attacker will poison A's ARP cache with a spoofed ARP Reply. The ARP reply will tell A that the IP address of B now has a MAC address of 00:00:00:00:00:04. Once A has processed the ARP Reply its ARP cache will look like this:

Host	IP Address	MAC Address
Host A	192.168.0.2	00:00:00:00:00:02
Host B	192.168.0.3	00:00:00:00:00:04
Attacker	192.168.0.4	00:00:00:00:00:04

Secondly the Attacker will poison B's ARP cache with a spoofed ARP Reply. The ARP reply will tell B that the IP address of A now has a MAC address of 00:00:00:00:00:04. Once B has processed the ARP Reply its ARP cache will look like this:

Host	IP Address	MAC Address
Host A	192.168.0.2	00:00:00:00:00:04
Host B	192.168.0.3	00:00:00:00:00:03
Attacker	192.168.0.4	00:00:00:00:00:04

Now whenever A sends B an ethernet frame the switch will route it to the Attackers port, this will also be the case whenever B sends A an ethernet frame. The attacker may now 'sniff' the traffic whilst forwarding it on to its originally desired host.

3.2 Connection Hijacking

Connection hijacking lets an attacker take control of two host's connection over a network. This type of attack is used against connection oriented protocols such as TCP/IP. An attacker could hijack a client's Telnet connection to a server and begin executing commands. ARP 'man in the middle' attacks (as detailed above) play an important part in hijacking a connection. A discussion of the TCP elements of the attack is beyond the scope of this paper. For further detailed reading on the mater please read my paper on 'Security weaknesses inherent in the design of TCP over IP'.

3.3 Connection Spoofing

Connection spoofing, in term of TCP/IP, lets an attacker open a full bi-directional TCP connection to a host masquerading as another legitimate host. This type of attack is common against trusted services such a 'rlogin' where authentication comes from the client's IP address. The attacker will need to poison the targets ARP cache so as the host being spoofed will never receive any packets ensuing from the spoofed connection. If the host being spoofed did receive a TCP packet from a connection it did not initially create it would automatically try to close that connection down.

3.4 Denial of Service

A Denial of Service (DoS) attack can be performed with an attacker altering a hosts ARP cache (via ARP Poisoning) with non-existent entries. Any frames sent to these non-existent entries will be dropped.

4 Protection

To protect a hosts ARP cache from being poisoned it is possible to make it static. If an ARP cache has been made static it will not process any ARP Replies received unlike a dynamic ARP cache. This is not practical for large networks as the correct IP address to MAC address association of every host would have to be present in the cache of every host before it is made static. If one host changed its MAC address (e.g. after replacing a NIC) all hosts ARP caches would need to be updated manually. On windows a login script could automate this process however it has been reported Windows will still accept and process ARP Replies even when the ARP cache has been made static.

It is also possible to use Intrusion Detection Systems (IDS) to detect ARP Poisoning attacks. Arpwatch is a tool that will monitor a network for any changes in MAC address to IP address association, e-mailing the administrator should any such offence occur.

5 Conclusion

Many higher level protocols such as IP, TCP and even SSL depend on solid foundation protocols. The inherent weakness in the ARP protocol directly effects the security of these higher level protocols. As we have seen these attacks are relatively simple to employ, as there are a wide variety of automated tools available, while any defence against them is minimal. Security measures such as switched networks and hard coded ARP tables do not offer great protection against ARP poisoning attacks.

6 Bibliography

- Altering ARP Tables - Data Wizard
- ARP Spoofing and Man-in-the-Middle Attacks - Robert Wagner
- An Introduction to ARP Spoofing - Sean Whalen
- Arpwatch
- Security weaknesses inherent in the design of TCP over IP - Steve Fewer