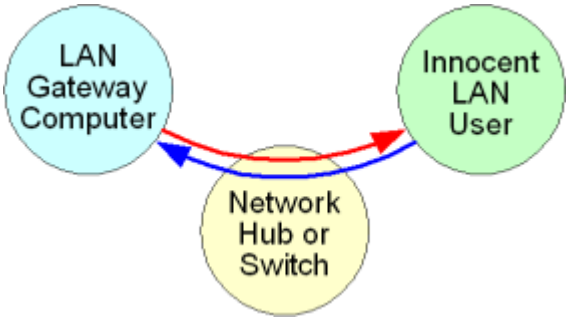
Address Resolution Protocol

- Computers on an Ethernet-based Local Area Network, or LAN, communicate by sending data to each other in the form of packets.
- These packets have headers that identify the sender and the recipient of each one.
- In an Ethernet LAN, traffic is addressed to a target Media Access Control address, or MAC address, as it's called.
- Internet traffic is sent using IP addresses to specify the sender and the recipient of the packets.
- This creates problems for traffic being sent to and from the Internet.
- To translate a LAN IP address to its MAC address owner, we have something called the **Address Resolution Protocol**, or **ARP**.

ARP is used to translate packets between IP address and MAC address.

Let's consider an illustration:



(graphic from http://www.grc.com/nat/arp.htm)

A user on computer A wants to go to http://www.suseblog.com/

After the initial DNS lookup, the computer makes the HTTP request to the remote server. The data comes back in the form of packets, which have to be sent to the proper computer on the LAN. The gateway must find out which machine to send the returning data to. If the gateway already knows this, it just translates the packets, and sends them along. If not, it has to send out an ARP request to find the MAC address associated with the IP address. Once the gateway has this information, it sends the packets to the appropriate internal computer. It then saves the MAC address and IP address in a temporary lookup table for future use.

Such a table may look like this:

```
[1736][root@suse-linux:/home/scott]$ arp -a
```

```
laptop (192.168.0.138) at 00:18:8B:BF:31:5B [ether] on eth0
bigben.truenorth.local (192.168.0.2) at 00:14:22:75:8B:CA [ether] on eth0
```

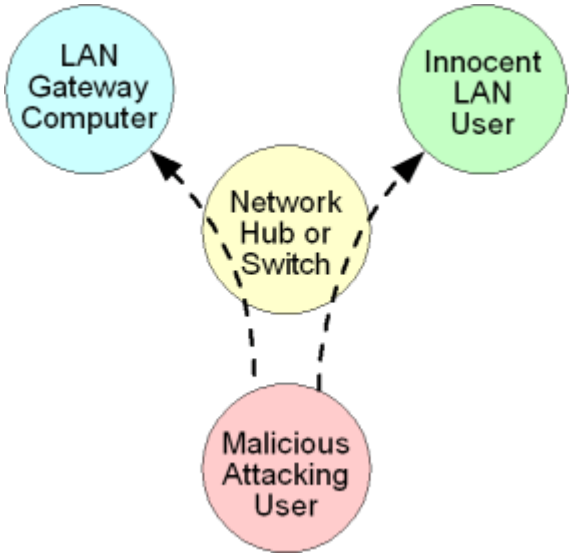
Each computer on the LAN has a cache or table of which MAC address belongs to each IP address.

Fascinating as that is, so what?

Poisoning

Vulnerability #1: There is absolutely no authentication used in this protocol
Vulnerability #2: It is possible to send unsolicited ARP replies

This Means: Any computer on the LAN can appear to any other computer on that LAN as any other computer on it.

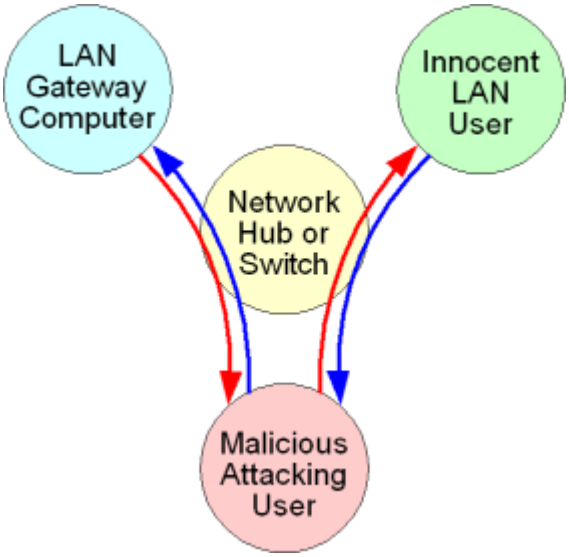


(graphic from http://www.grc.com/nat/arp.htm)

In other words, you can tell the gateway that computer A is computer B, and you can tell computer B that computer A is the gateway. This can be done with 2 packets.

Packet 1: You just pair computer A's MAC with computer B's IP in the gateway's ARP table. This is effectively telling the gateway that data it wants to send to computer B's IP address should be sent to computer A's MAC address.

Packet 2: You can then tell computer B that computer A's MAC address belongs to the gateway's IP address.



(graphic from http://www.grc.com/nat/arp.htm)

Uses

Why would you want to do this?

You own a business and you want to set up a temporary content monitor or filter. Rather than change any configuration on the gateway, router, or firewall rules, you can put a machine with 2 NICs just inside the network connection, set up ARP poisoning, and you are watching all packets. You can choose to block packets, drop them, change them, or monitor them.

As the admin of a network, you suspect unauthorized use of instant messenger software. You can quickly and easily set up temporary ARP poisoning with a regular desktop machine, watch for the traffic, and log the packets. You can then take appropriate corrective action against the offending computer and/or users.

The Windows Fanboy
Windows admin who had a little bit of a superiority complex about Windows.
I set up a temporary ARP poison just on his machine.
I watched the traffic for a minute and found his email password: s4ndy@1996m
I put together a rule so that any information coming through my computer would change that password to something else: sandy!l995z
I activated the filter, and all of a sudden, he had no access to check his email.
No more Windows fanboy.

Couldn't you just sniff packets in promiscuous mode?
Yes, but this only works when you are using a hub. If you are using switches, this will not work. A

switched network will not send all packets to all machines on the LAN. It sends packets only to the port that has the correct machine attached to it. Thus, even in promiscuous mode, you will only capture the packets destined for your own machine. ARP poisoning, as it's called, works whether you have hubs or switches.

Dangers

What other dangers exist from ARP poisoning?

It is possible to intercept, decrypt, and log even SSL-encrypted data. Won't that put up a flag saying that the certificate has changed? Yes, but who really pays any attention to those? I've seen seasoned system administrators dismiss such warnings without a second thought.

Anything transmitted from your computer to the LAN or the Internet can be intercepted and viewed by anyone else on the LAN. Even if it is encrypted. Luckily, you have at least a notification pop up when the SSL cert is changed.

People can grab your instant messenger passwords, credit card information, account information, email passwords, and Web site login information without touching your computer in any way. This is all possible with absolutely no required physical access to your computer. No software installation needed.

Is it traceable? Yes, if you get a snapshot of the ARP table showing the correct information before, and another snapshot after the ARP tables have been poisoned. You will then be able to tell the MAC address of the offending computer. This is only good if you own the offending machine and know which machine the offending MAC belongs to. If people bring in laptops, you will not have a record of those MAC addresses, and will not know which machine it is.

The other part with all this is that on Linux machines, you can fake a MAC address, or use one associated with a known machine (provided that the other machine is off).

Conclusion

ARP Poisoning can be a useful tool if you legitimately own a network. Malicious users can also cause mayhem with it. ARP Poisoning is a design flaw, but only because the original Ethernet design was conceptualized to work with trusted computers on the network. There is no way the people who came up with the architecture could have known that their protocols would be in such wide ranges of use as they are today. Just be careful.