Detection of ARP Spoofing-Based Attack on a LAN

Yan Naung Soe
University of Computer Studies,Yangon,Myanmar
yannaung83@gmail.com

Abstract

This paper presents the network monitoring method for ARP spoofing attack. ARP spoofing is also called man-in-the-middle (MITM) attack. The key to ARP spoofing attacks lies in modifying the cache MAC and IP addresses pair information maintained by each system. The technique utilized to perform an ARP spoofing attack is sending false ARP broadcast to all devices on LAN.[10] Then the attacker can know the communication of viaticum system and other LAN users connect with each other and can monitor any plain text authentication information and communications. So, it can be detect ARP spoofing attack. To detect ARP spoofing, there are some ways. But, the best approach is to monitor the communications in LAN. Detection spoofing system is to monitor the IP and MAC pairs in cache when the system received updated information. When spoofing event occurs this system will appear alert and then user not only can check their network but also can avoid spoofing attack.

Keywords : ARP (address resolution protocol), MITM (man-in-the-middle attack), MAC spoofing, MAC flooding, ARP spoofing

1. Introduction

Now a day, network computer systems are more widely used in anywhere and anytime. One computer connects with other computers as in network to improve productivity by using and sharing data or program.

A computer connected to an IP/ Ethernet LAN has two addresses. One is the address of the network card, called the MAC address. The MAC, in theory, is a globally unique and unchangeable address which is stored on the network card itself. MAC addresses are necessary so that the Ethernet protocol can send data back and forth, independent of whatever application protocols are used on top of it. Ethernet builds "frames" of data, consisting of 1500 byte blocks. [7] Each frame has an Ethernet header, containing the MAC address of the source and destination computer.

The second address is the IP address. IP is a protocol used by applications, independent of whatever network technology operates underneath it. Each computer on a network must have a unique IP address to communicate. IP addresses are virtual and are assigned via software.

IP and Ethernet must work together. IP communicates by constructing "packets" which are similar to frames, but have a different structure. These packets cannot be delivered without the data link layer. In our case they are delivered by the Ethernet, which splits the packets into frames, adds an Ethernet header for deliver, and sends them down the cable to the switch. The switch then decides which port to send the frame to, by comparing the destination address of the frame to an internal table which maps port numbers to MAC addresses.

When an Ethernet frame is constructed, it must be build from an IP packet. However, at the time of construction, Ethernet has no idea what the MAC address of the destination machine is, which it needs to create an Ethernet header. The only information it has available is the destination IP from the packet's header. There must be a way for the Ethernet protocol to find the MAC address of the destination machine, given a destination IP.

This is where ARP, the address resolution protocol, comes in. ARP protocol was published in November 1984 by David C.Plummer. As IT security was not an important factor back in 1982, the aim was simply to provide functionality. [4]

But, attacking events increased more and more in later year. Curiosity, revenge, industrial espionage are all reasons why attack systems on their own network. Static's show that 70 to 80 percent of all attacks originate on the internal network. [9] One of the most formidable forms of the internal attack is known as ARP spoofing attack. The level of interconnection where ARP spoofing attacks occur is known as Layer 2, or the data link layer in the OSI network model. [10]

2. ARP operation

Layer 2 communication is the MAC address. Every network interface in an Ethernet network is assigned a MAC, or Medium Access Control address, at the time the device is manufactured. [10] The MAC address is used to unique identify very interface connected to an Ethernet network/ Every Ethernet card manufactured has a unique address so that cards from any vender can be interconnected on an

Ethernet based network without having to worry about address conflicts.

MAC addresses are used by network equipment such as switches to route information to the correct port on which a destination machine resides. This MAC address-based routing eliminates the need to broadcast traffic on all ports, as a hub does. Devices with connected interfaces on an Ethernet LAN use methods for discovering other connected interfaces on the LAN: Address Resolution Protocol, ARP, respectively. Without this protocol to perform this interface discovery, it would be necessary to manually input the MAC addressed and associated IP addresses into every machine for every interface on a LAN. This would be a daunting task considering the size and dynamic nature of most modern networks.

ARP automates this process through a series of Ethernet frame broadcasts to detect other locally connected machines. This information is then stored so that traffic sent between systems on the LAN can be properly routed by interconnecting network devices.

2.1 ARP Request and Reply

ARP operates by sending out "ARP request" packets and by receiving "ARP reply" packets.[1] The first part maps an IP address to a physical address when sending a packet, and the second part answers requests from other machine. Request asks the question "who has the IP address a.b.c.d?", and then "send me back MAC address". These packets are broadcast to all computers on the LAN, even on a switched network. Each computer examines the ARP request, checks if it is currently assigned the specified IP, and sends an ARP reply containing its MAC address. Broadcasting an ARP request to find an address mapping can become complex. The target machine can be down or just too busy to accept the request. If so, the sender may not receive a reply or the reply may be delayed. Because is a best-effort delivery system, the initial

ARP broadcast request can also be lost. In this case, the sender should retransmit, at least once. Meanwhile, the host must store the original outgoing packet so it can be sent once the address has been resolved. In fact, the host must decide whether to allow other application programs to proceed while it processes an ARP request. If so, the software must be handle the case where an application generates additional requests for the same address without broadcasting multiple requests for a given target.

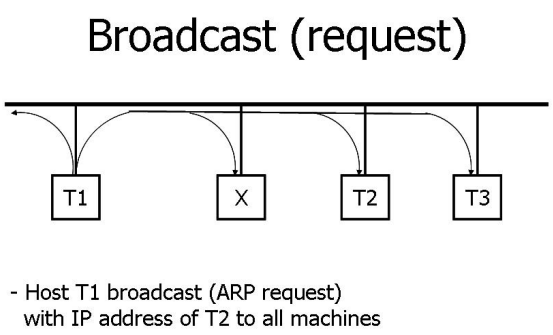

Figure-1: ARP request operation

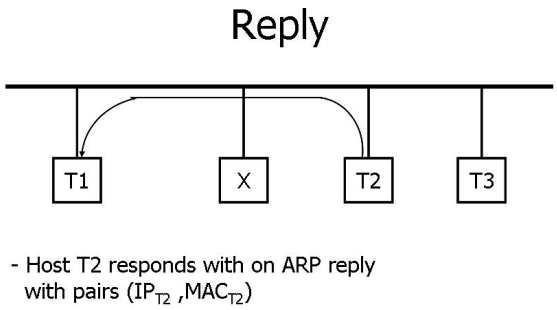

Figure-2: ARP reply operation

2.2 Packet Format

To communicate mappings from <protocol, address> pairs to 48.bit Ethernet addresses, a packet format that embodies the Address Resolution protocol is needed. The format of the packet follows. Ethernet transmission layer. [4]

48.bit : Ethernet address of destination
48.bit : Ethernet address of sender
16.bit : Protocol type = ether_type $ADDRESS_RESOLUTION

Ethernet packet data:

16.bit : (ar$hrd) Hardware address space (e.g., Ethernet, Packet Radio Net.)
16.bit : (ar$pro) Protocol address space. For Ethernet hardware, this is from the set of type fields ether_typ$<protocol>.
8.bit : (ar$hln) byte length of each hardware address
8.bit : (ar$pln) byte length of each protocol address
16.bit : (ar$op) opcode (ares_op$REQ_UEST | ares_op$REPLY)
nbythes : (ar$sha) Hardware address of sender of this packet, n from the ar$hln field.

mbytes : (ar$spa) Protocol address of sender of this packet, m from the ar$hln field.
nbytes : (ar$tha) Hardware address of target of this packet.
mbytes : (ar$tpa) Protocol address of target.

+	Bits 0 - 7	8 - 15	16 - 31
0	Hardware type = 1		Protocol type = 0x0800
32	Hardware length = 6	Protocol length = 4	Operation
64	SHA (first 32 bits) = 0x000958D8		
96	SHA (last 16 bits) = 0x33AA		SPA (first 16 bits) = 0x0A0A
128	SPA (last 16 bits) = 0x0A8C		THA (first 16 bits) = 0x0009
160	THA (last 32 bits) = 0x58D81122		
192	TPA = 0x0A0A0A7B		

Figure-3: Frame format of ARP packet
Note that: Operation 1 for request packet
Operation 2 for reply packet

2.3 Cache

To minimize the number of ARP requests being broadcast, operation systems keep a cache of ARP replies. [1] When a computer receives an ARP reply, it will update its ARP cache with the new IP/MAC association. Most operation systems will update their cache if a reply is received. Thus, when two computers on a network communicate, they began with an ARP request and reply, and then repeatedly transfer packets without using ARP for each one. Experience shows that because most network communication involves more than one packet transfer, even a small cache is worthwhile. An ARP cache entry 2 minutes under Windows 2000, and can be renewed up to a 10 minute maximum while referenced in outgoing frames. Linux expires entries after 1 minute. These parameters are defaults and are configurable per client. [2]

2.4 Time-out Cache

The ARP cache provides a problem as soft state [1], a technique commonly used in network protocols. If the one of the network interface card will change or remove, the information in the cache has become incorrect. There are many computers in a network, and host T1 want to send packets host T2. Computer T1 may be send ARP request and receive the reply, and then store the cache. When the next time connection of these two hosts, computer T1 do not need to broadcast request packets because it already has address binding information for computer T2 in its ARP cache, compute T1 will continue to send packets to T2. If the network interface card of T2 is changed, T1 has no way of knowing when information in its ARP cache has become incorrect. To accommodate soft state, it is required to update cache with a timer. [1] When the time is expires, the information is removed. And request and reply processes are repeatedly provided and update the information in cache.

2.5 Experiments

For generating request packet,

1. The IP module sends a packet, destined for another host in the network, to the ARP module.
2. The ARP module looks up the ARP table (cache) to resolve the IP address.
3. If the supplied IP address is present in the ARP cache, it is resolved into its Ethernet address.
4. If the ARP module is not able to find an entry for this IP address in the ARP cache, then it sends an ARP request packet to the Ethernet driver, to resolve the IP address to the Ethernet address.
5. After the IP address is resolved by the ARP module, the packet is sent to the Ethernet driver for transmission.

For receiving the reply packet,

1. If the IP address to be resolved is for this host, then the ARP module sends an ARP reply packet with its Ethernet MAC address.
2. If the IP address to be resolved is for this host, then the ARP module updates its ARP cache with the source Ethernet MAC address to source IP address mapping present in the ARP request packet. If the entry is already present in the cache, it is overwritten. If it is not present, it is added.
3. If the IP address to be resolved is not for this host, then the ARP module discards the ARP request packet.

3. Spoofing ARP

As ARP makes no attempt to protect itself against spoofed packets, it is vulnerable to a series of attacks. The most common types are MAC spoofing, MAC flooding, and ARP spoofing. [9]

MAC spoofing involves the attacker using

a spoofed MAC source address. This Technique makes sense if privileges are linked to MAC address. MAC spoofing is useful for attackers who want to protect their identity. There is a good way to preventing this wired network: many switches enable port security. The switch only learns each MAC address once, and then stores this address permanently. From this point onward, the switch will not accept any other source MAC address on the mapped port. This mechanism is effective against MAC spoofing attacks.

MAC folding attacks are designed to take down a switch's port security mappings. In contrast to hubs, switches use CAM (Content Addressable Memory) tables, which specify the port behind each active MAC address on switch. The switch will only send packets via the port that leads to the target machine. Attackers can disable this function by flooding the switch with addresses.

The third attack is not as easy to detect, and there are no simple countermeasures. The attack is based on ARP spoofing, where the attacker deliberately transmits fake ARP packets. ARP poisoning is specific type of ARP spoofing that aims to manipulate (poison) the ARP tables on other machines.[9]

As operating systems tend not to check if an ARP reply is the answer to an ARP request sent previously, the address information from the reply is cached. On Windows systems attackers can even modify entries explicitly declared as static by users.

During so allows an attacker to monitor the dialog between a client and a server, as the man in the middle, to manipulate that dialog. The man in the middle manipulates the server entry in the client's ARP cache, making the client believe that the attacker's own MAC address is actually the server address. The same trick is also played on the server.

3.1 Spoof Attack

The Key to ARP spoofing attacks lies in modifying the cached MAC and IP address pair information maintained by each system. The technique utilized to perform an ARP spoofing attack is sending false ARP broadcast notification to devices on the local network. These false ARP spoofing massages trick network devices into delivering network data to incorrect switch ports, allowing the attacker to have information destined for a victim system on the LAN sent to the attacker's port on the network device.

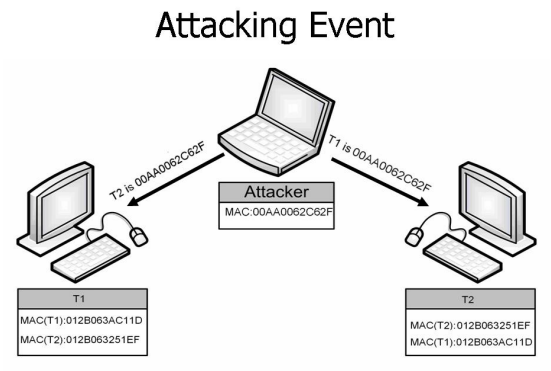

Figure-4 : Spoofing attack to the network

ARP spoofing, man-in-the middle attack (MITM) attack places the attacking system between the victim's system and the local gateway for egress traffic, allowing the attacking system to "sniff" everything the victim sends and receives. [10]

The attacker tricks the victim's system into incorrectly addressing Ethernet frames of its packets, and tricks the switch into sending the victim's data to the attacker's switch port. The attacker does with a series of spoofed ARP massages, after which it is possible for the attacker to monitor the victim's egress connections. The victim to continue to send and receive data to remote systems, the attacker sends another set of false ARP broadcast frames. This set of forged massages incorrect tells the victim's system that information destined for remote systems should be sent to the attacker's MAC rather than to the gateway.

As ARP is a stateless protocol, most operating systems will update their cache if a reply is received, regardless of whether they have sent out an actual request. ARP spoofing involves constructing forged ARP replies. When attacking is performed, attacker inserts his computer between the communications path of two target computers.

The attacker is performed as follows..

- X poisons the ARP cache of T1 and T2
- T1 associates T2's IP with X's MAC
- T2 associates T1's IP with X's MAC
- All of T1 and T2's IP traffic will then go to X fast, instead of directly to each other

3.2 Attacker's Work

The implications to security of being interposed between the victim and the gateway are severe. Any plain-text authentication information or communication, such as network passwords or e-mail, can be monitor by the attacker. [10]

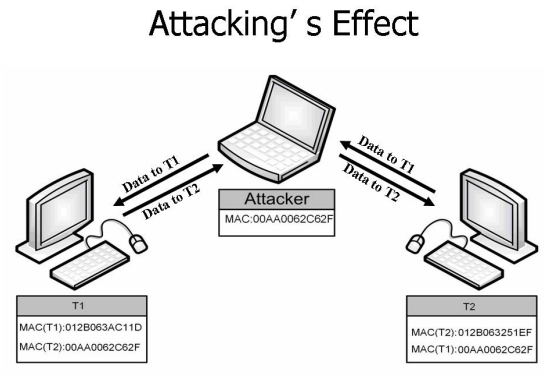

Figure-5 : Spoofing effect due to attack

4. Detection System

ARP spoofing based attack is highly effective in a local area network (LAN). So, it is required to detect from ARP spoofing based attack.

Due to dynamic nature of the modern LAN and the automatic configuration of the ARP and IP pair information, detecting an ARP spoofing-based attack is difficult. The best approach is to enable software that monitors the ARP/IP pair combination for machines on a given LAN.

The feature of this detection system is followed.

- broadcast the ARP requests and receives the replies
- show the routing table with the pairs of MAC and IP
- monitor each of the MACs in table
- if spoofing attack occurs, it can give alert
- update the IP and MAC pairs in require seconds.

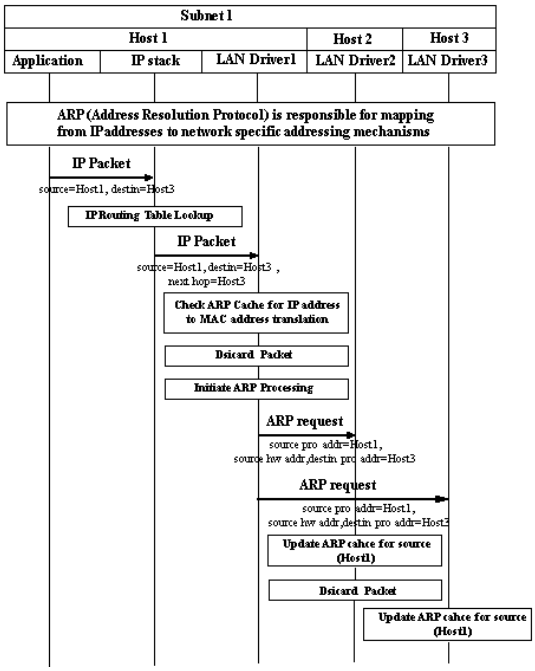

Figure-6 : Send ARP request packet

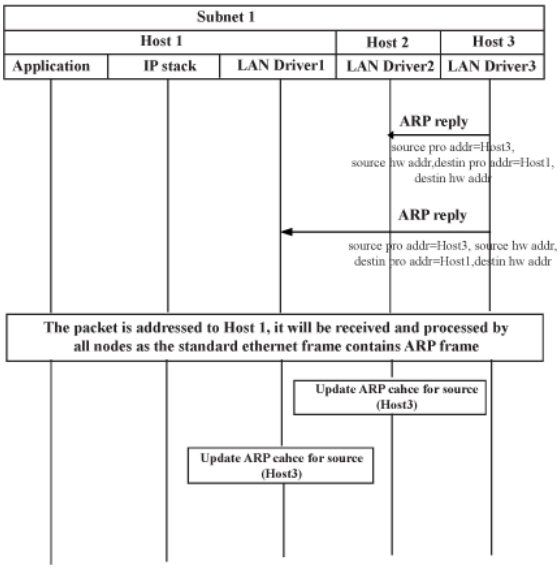

Figure-7 : Receive ARP reply packet

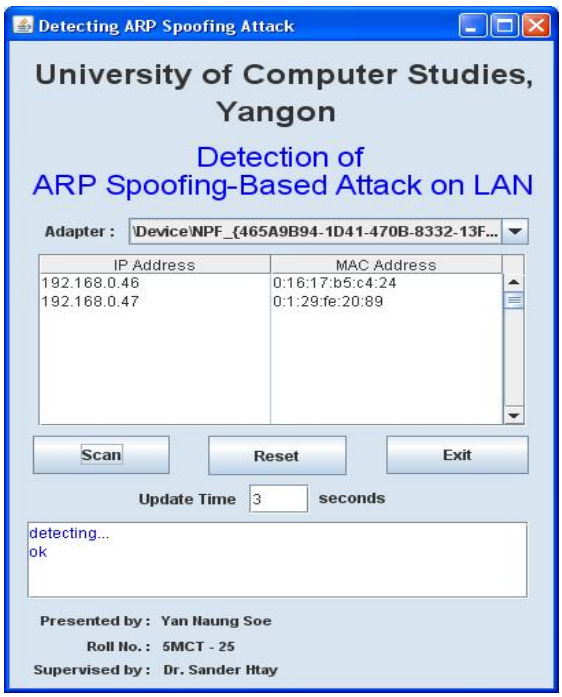

Figure-8 : Detection system for spoofing attack

Above these step, first process is to scan the user's network adapter. If user uses two network adapters, it can display these two lists in adapter list. (Figure-8) Then user select adapter that used and scan IP/MAC pairs all network devices that these adapter is connected. And then user can set scan processes' update time. This system can update IP/MAC list in network required update time and check IP/MAC list in table (cache). If spoofing event occurs, alert massage will appear. If so, user can check their network and can reset the system. If the system is not spoof, detecting OK massage is displayed. Update process requires because one or more of the network adapter can change or damage and to know spoofing process when the attack is processed. Figure-9 is expressed when spoofing event occurs.

When Spoofing occurs,

IP Address	MAC Address
192.168.0.5	00-00-0c-94-36-ab
192.168.0.2	00-00-0c-94-36-ab
192.168.0.1	00-00-0c-94-36-ab
192.168.0.4	00-00-0c-94-36-bb

Figure-9 : Spoofing effect in cache

5. Conclusion

This system is useful for all users that use in local area network (LAN). If an attacker will attack in the network, this system can give users alert massage. When user gets this massage, user can check their network and can protect from effects of attacker's attack. The future extension for the system is to protect from spoof attack by using these two ways. The first is to segment network method, can be accomplished by separating out subnets, using virtual LAN and router-based broadcast access control to limit the exposure of hosts to ARP spoofing-based attacks. The second is to hard-core the important MAC/IP pairs into mission critical machines, so that an attacker cannot modify them.

References

1. Douglas E. Comer: "I*nternetworking with TCP/IP*", 4th Edition
2. Sean Whalen, Matt Bishop: "*Layer 2 Authentication*", February 25, 2005
3. T. Melsen Erisson :"*MAC Force Forwarding*" ,January 14, 2004
4. David C. Plummer: RFC 826, "*An Ethernet Address Resolution Protocol (or) Converting Network Protocol Address to 48 bit Ethernet Address for transmission on Ethernet Hardware*", November 1984
5. Charles Hornig, RFC 894, "*A Standard for Transmission of IP Datagrams over Ethernet Networks*", April 1984
6. Ross Finlayson, Timothy Mann, Jeffrey Mogul, Marvin Theimer, RFC 903,"*A Reverse Address Resolution Protocol*", June 1984
7. Sean Whalen, "*An introduction to ARP spoofing*", April, 2001Revision 18
8. http://www.keelog.com
9. http://www.Linux-magazine.com,(issue : 56 July 2005)
10. http://www.foundstone.com
11. http://www.oxid.it/downloads/arp-intro.swf
12. http://en.Wikipedia.org/wiki/Proxy_ARP.htm