Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 5, May 2014, pg.1055 – 1060

RESEARCH ARTICLE

Detection of ARP Spoofing Attack Using ICMP Protocol

Vinay K.R[1], B.K. Gudur[2]

[1]M.Tech Scholar, Department of Electronics and Communication Engineering, BLDEA's college of Engineering &Technology, Bijapur, Karnataka, India

[2]Assistant Professor, Department of Electronics and Communication Engineering, BLDEA's college of Engineering &Technology, Bijapur, Karnataka, India

[1] vinaykr49@yahoo.com; [2] bmeshg@gmail.com

Abstract—Address Resolution Protocol (ARP) is a protocol for mapping an Internet Protocol address (IP address) to a physical machine address (MAC) that is recognized in the local network. ARP spoofing is the act of vindictively changing the IP-MAC associations stored in ARP cache of any network host. This paper discusses ARP spoofing attack and some related works about it first. On these bases, the paper proposed an efficient algorithm based on ICMP protocol to detect malicious hosts that are performing ARP spoofing attack. The technique includes collecting and analysing the ARP packets, and then injecting ICMP echo request packets to probe for malicious host according to its response packets.

Keywords— ARP cache; ARP protocol; ARP spoofing; Winpcap; Cain & Abel; ICMP protocol

I. INTRODUCTION

Address Resolution Protocol (ARP) is a protocol having simple architecture and have been in use since the advent of Open System Interconnection (OSI) network architecture. It's been working at network layer for the important dynamic conversion of network address i.e. Internet Protocol (IP) address to physical address or Media Access Control (MAC) address. When a host wants to communicate with another host whose hardware address it does not know, it broadcasts an ARP request for the hardware address associated with the protocol address of the destination. And only the host with corresponding protocol address sends a unicast reply to the sender with its < protocol address, hardware address,> pair. Obviously, ARP protocol plays a key role in local area network communication, but due to its own loopholes, it is often used as part of other serious attacks such as Man-in-the-Middle (MiM) attack, Denial of Service (DoS) attack. With a MiM attack, the attacker can sniff the traffic between two victim hosts. With a DoS attack, the attacker makes a victim host deny communicating with others. So ARP spoofing attack is becoming the most dangerous attack in the LAN [1].

Earlier it was sufficiently providing its services but in today's complex and more sophisticated unreliable network, security being one major issue, standard ARP protocol is vulnerable to many different kinds of attacks. These attacks lead to loss of important information. With certain loopholes it has become easy to be attacked and with not so reliable security mechanism, confidentiality of data is being compromised. In order to increase the efficiency of the network and not tie up bandwidth doing ARP broadcasting, each computer keeps a table of IP addresses and matching Ethernet addresses in memory. This is called ARP cache. The host who issued the request will cache the address pair taken by the reply. There are two types of entries in an ARP cache: Static

entries which remain in the ARP cache until the system reboots Dynamic entries which only remain in the ARP cache for few minutes. Before sending a broadcast, the sending computer will check to see if the information is in its ARP cache. If it is then it will complete the Ethernet data packet without an ARP broadcast. All operation systems allow update an old entry by an ARP request or reply packet. ARP is a stateless protocol designed without security in mind, which makes it an ideal means for launching DoS and MitM attacks on a LAN. By sending spoofed MAC addresses in ARP reply packets, a malicious host can poison the ARP cache of other hosts on the local network and thereby easily redirect network traffic [2].

There are two techniques for detecting ARP spoofing one is Passive technique and other is Active technique. In Passive Detection we sniff the ARP requests/responses on the network and construct a MAC address to IP address mapping database. If we notice a change in any of these mappings in future ARP traffic then we raise an alarm and conclude that an ARP spoofing attack is underway. The main drawback of the passive method is a time lag between learning the address mappings and subsequent attack detection. In active method we are purposely injecting the packet to the network. The proposed technique actively interacts with the network to gauge the presence of ARP spoofing attacks.

II. BACKGROUND

The most commonly employed network frame architecture is based on OSI model. Each layers service depends on the varying protocols working at layers. But working of these protocols to ensure complete security to the applications running has really become a challenging task for the developers and security professionals. There have been different efforts made in the field of network security to provide effective and reliable defensive mechanisms to control attacks being made of these protocols due to their lack of ability to work and prevent themselves in complex and insecure network. One such protocol is ARP working on network layer. Due to its architecture it cannot prevent itself from being attacked over LAN.

As ARP is a stateless protocol and its reply packets are not authenticated, all hosts blindly cache the ARP replies they receive from the network. This mechanism provides convenient for malicious host. Attackers send forging ARP packets to the victim periodically to perform ARP spoofing attack. In an ARP spoofing attack, the attacker sends ARP request or reply packets with fake <IP, MAC> mappings. For example, if a malicious host wants to sniff traffic sent from X to Y, he could send X an ARP packet with the address mapping <IP of Y, MAC of attacker>. Host X will cache the wrong address mapping and send data destined to Y to the attacker instead. If the attacker also wants to know the information sent from Y to X (MiM attack), the attacker just needs to send Y an ARP packet with address mapping <IP of X, MAC of attacker>. In order not to interrupt normal communication between host X and Y, the attacker need to enable IP packet routing to redirect the packet to the original destination host. If the attacker wants to perform a DoS attack, the attacker can poison the ARP cache of a host in the same way. Every packet the host sends is sent to the attacker. Once the attacker receives the packet, he simply drops it, and therefore, blocks the communication of the victim host [3].

A. Functionality of ARP

When an incoming packet destined for a host machine on a particular local area network arrives at a gateway, the gateway asks the ARP program to find a physical host or MAC address that matches the IP address. The ARP program looks in the ARP cache and, if it finds the address, provides it so that the packet can be converted to the right packet length and format and sent to the machine. If no entry is found for the IP address, ARP broadcasts a request packet in a special format to all the machines on the LAN to see if one machine knows that it has that IP address associated with it. A machine that recognizes the IP address as its own returns a reply so indicating. ARP updates the ARP cache for future reference and then sends the packet to the MAC address that replied. Here is an example of a simple ARP Communication. Jessica, the receptionist, tells Word to print the latest company contact list. This is her first print job today. Her computer (IP address 192.168.0.16) wants to send the print job to the office's HP LaserJet printer (IP address 192.168.0.45). So Jessica's computer broadcasts an ARP Request to the entire local network asking, "Who has the IP address, 192.168.0.45?" as seen in Fig.1.

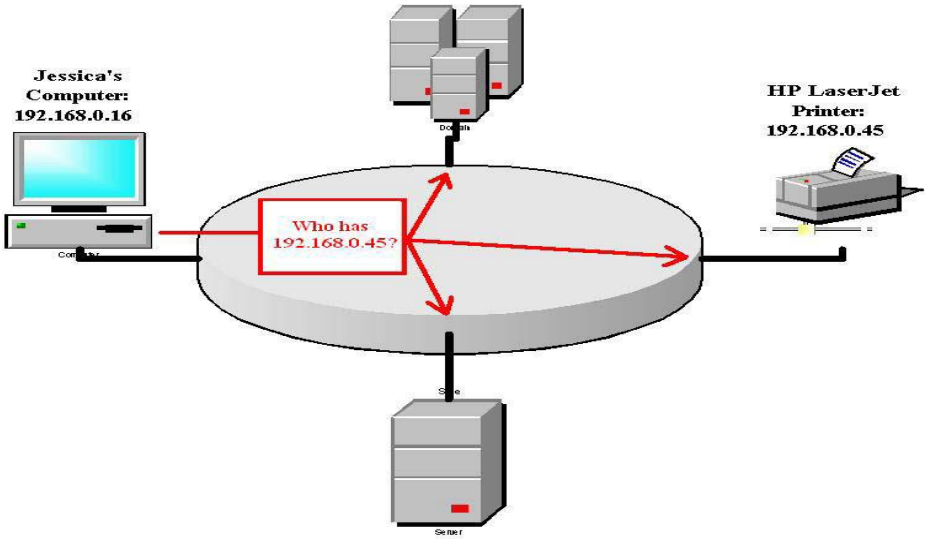

Fig.1. ARP Functionality

All the devices of the network ignore this ARP Request, except for the HP LaserJet printer. The printer recognizes its own IP in the request and sends a unicast ARP Reply: Hey, my IP address is 192.168.0.45. Here is my MAC address: 00:90:7F:12: DE: 7F as in Fig.2. Now Jessica's computer knows the printer's MAC address. It sends the print job to the correct device, and it also associates the printer's MAC address of 00:90:7F:12: DE: 7F with the printer's IP address of 192.168.0.45 in its ARP table.

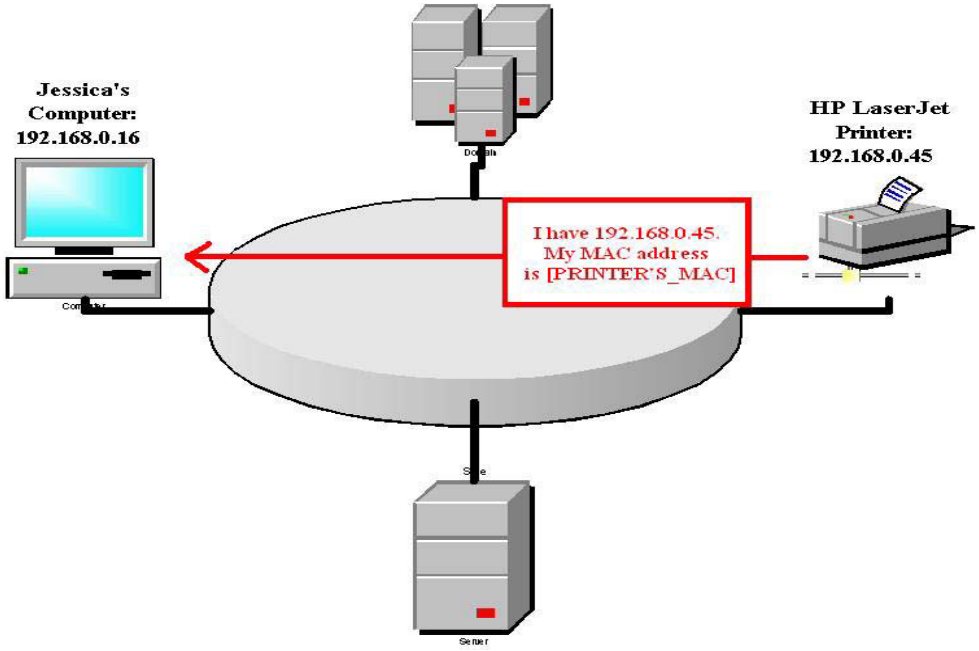

Fig.2. ARP Functionality

This is the case when the receiving host is on the same network. If the receiving host is on another network, the sending computer will go through its route table and determine the correct router (A router should be between two or more networks) to send to, and it will substitute the ethernet address of the router in the ethernet message. The encased IP address will still have the intended IP address. When the router gets the message, it looks at the IP data to tell where to send the data next. If the recipient is on a network the router is connected to, it will do the ARP resolution either using its ARP buffer cache or broadcasting [4].

III. RELATED WORK

Many efforts have been made and different methods have also been applied to prevent ARP spoofing attacks, but none has been able to give satisfactory results. Different tools and architectures have been proposed but each have their own feasibility issues. One simple but effective way to prevent ARP attacks is using static entries in the ARP cache. The drawbacks of this solution are its low scalability. It does not work well in dynamic environment and it would be a really heavy work for the network administrator to deploy and update these tables throughout the network especially when the network is big.

Gouda and C.-T. Huang [5] proposed the architecture to resolve IP addresses into MAC addresses over an Ethernet. The proposed architecture consists of a secure server connected to the Ethernet and two protocols: an invite–accept protocol and a request–reply protocol. Each computer connected to the Ethernet can use the invite–accept protocol to periodically record its IP address and its hardware address in the database of the secure server. Each computer can later use the request–reply protocol to obtain the hardware address of any other computer connected to the Ethernet from the database of the secure server. This solution is not practical because it requires changing the ARP protocol implementation of every host with this new address resolution protocol.

D. Bruschi, A. Ornaghi, and E. Rosti. Secure ARP protocol (S-ARP) [6] is a backward compatible extension to ARP. Beginning with an ARP request, Cryptographic Link Layer (CLL) applies public key cryptography to perform an initial handshake between two hosts with the aim to establish a security association. The two hosts prove their identity to each other and exchange keying material. Here upon, secured IP data packets may be sent. This solution involves cryptography to authenticate the origin of ARP packets. To implement this solution in a LAN, every host has to be modified to use S-ARP instead of ARP. This is not scalable to update a stack across all available operating systems. Another disadvantage of this method is that it has the additional overhead of cryptographic calculations as S-ARP uses Digital Signature Algorithm (DSA).

M.Barnaba, Anticap [7] is a kernel patch for UNIX-based operating systems. It prevents ARP poisoning attacks by rejecting ARP updates that contain a different MAC address from the current table entry for the same IP address. This solution works only in static environment, and is available for a limited number of operating systems.

Some high-end Cisco switches have a new feature which allows the switch to drop ARP packets with invalid <IP, MAC)>address bindings [8]. One disadvantage of this feature is its high cost; another one is that it might not be able to verify some ARP packets on all switches in the VLAN.

IV. THE PROPOSED METHOD FOR DETECTING ARP SPOOFING

The following sections describe the details of the architecture we propose and tools which are used.

A. The composition of the experiment is listed here:

1) Winpcap used as a packet capture tool.
2) Cain & Abel used as a spoofing tool.

Winpcap is an architecture for packet capture and network analysis for the Win32 platforms. It includes a kernel-level packet filter, a low-level dynamic link library (packet.dll), and a high-level and system independent library [9].

Cain & Abel is a two part program where Cain is the GUI of the program, and Abel is windows service that provides a remote console on the target machine. An interesting feature of Cain & Abel is APR (ARP Poison Routing) which allows sniffing packets of various protocols on switched LAN's by hijacking IP traffic of multiple hosts concurrently [10].

B. Architecture

As shown in Fig.3, we adopt a modularized approach and divide our ARP spoofing detection system into the following modules:

1) ARP Packet Sniffer Module: This module sniffs all ARP packets from the Ethernet.
2) IP-MAC Mapping Database: Initially first IP-MAC entry is stored in this database and second entry is compared with database, If both entry matches it indicates no sign of ARP spoofing but if new MAC entry for same IP entry it assumed to be spoofing hence we send that IP entry to ARP spoofing detection module, and if new IP entry comes then it stored in database.
3) ARP Spoofing Detection Module: This is the main Detection module. We feed the IP entry for which new MAC entry came into it as input. This module send ICMP packet to that IP address with known identifier and sequence number in the ICMP packet and if reply comes from that host then it compares the identifier and sequence number field of the reply ICMP packet with the sent (known) identifier and sequence number, if both are matches then it decides the host is legitimate else if reply doesn't come then it decides the host is fake, with reply it obtains real MAC and the database is updated with real MAC. The details of the module will be discussed in Section C.
4) Response Module: This module is used to alert the network administrator the happening of ARP spoofing attack using alert message and this message contains information of malicious host.

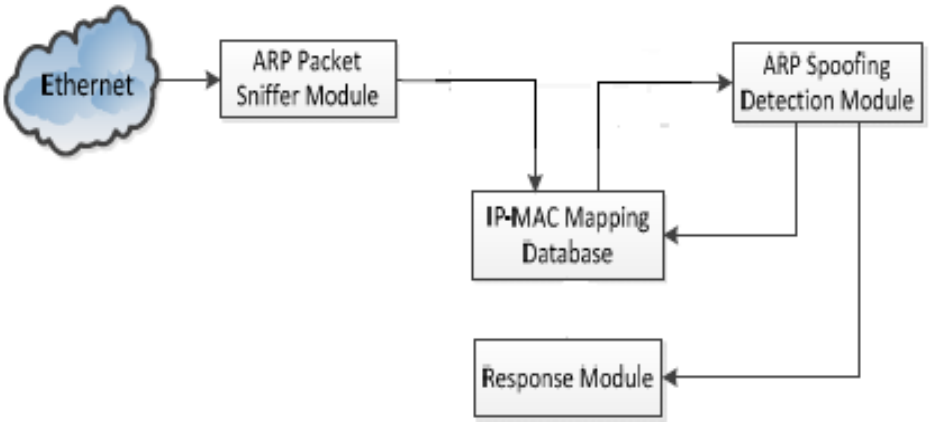

Fig.3. ARP spoofing detection system.

C. ARP Spoofing Detection Module

It works based on two rules as follows.

Rule 1: The NIC of a host only accepts packets with its own hardware address, broadcast address, and subscribed multicast addresses. The network layer only accepts IP packets addressed to its IP address and will drop the other packets silently. For example, there is a host with MAC address X and IP address Y, it would accept packet with destination MAC address X and destination IP address Z as the destination MAC address matches, but still discard the packet as the destination IP address doesn't match, without sending any error messages back to the source host.

Rule 2: Hosts with enabled IP packet routing will forward the packet to the destination host. All legitimate hosts in the network do not enable IP packet routing and will response back after it receives an ICMP echo request packet. Based on the two rules, we can verify the ARP packets we've got whether they are real or fake packets

A ping packet is an ICMP packet usually used to detect the connectivity between two hosts. When a host A wants to ping another host B in a network, it will sent B an ICMP echo request packet and wait for an ICMP echo reply packet sent by B. It is worth mentioning that the identifier and sequence number fields of the echo reply packet are set to the same with the received echo request packet in order to facilitate the sender to match the echo reply with the request packets. When the detection host gets an ARP packet with source MAC address X and source IP address Y which does not have an entry in the mapping database, we will regard the source addresses <MAC address = MAC-X, IP address = IP-Y> as the addresses of the suspicious host. And then it will construct a trap ICMP ping packet with them as the destination addresses. The value of the source addresses is set to the addresses of the detection host, assume IP-MAC pair of detection host is (222.199.193.62, DC: 0E: A1: DD: CC: 41). Table 1 shows the value of the main fields of the trap ICMP ping packet. When an ICMP packet as constructed below is sent to the source of the ARP packet, the host's response will be based on Rule1 and Rule2.

TABLE I. TRAP ICMP PING PACKET

Ethernet Header	
Destination MAC address =	MAC-X
Source MAC address =	DC: 0E: A1: DD: CC: 41
Ethernet Type =	0x0800
IP Header	
Destination IP address =	IP-Y
Source IP address =	222.199.193.62
ICMP Header	
Type =	8 (echo request)
Code =	0
Identifier =	-
Sequence Number =	-

V. DETECTION ALGORITHM

```
1: If packet type is 0X806 then ARP packet
2: Store First IP-MAC entry into database.
3: If (stored first entry matches to next obtained entry) {
        Capture next packet}
   Else if (only IP matches) {
        Send IP to ARP spoofing detection module}
   Else (new IP entry) {
        Store into database}
4: ARP spoofing detection ( ) {
        Send ICMP packet to obtained IP with known identifier and sequence number.
        If (reply comes) {
                If (identifier and sequence number matches) {
                        Host is legitimate}
        }
        Else
                Malicious host and call response module ( )
    }
5: Response module ( ) {
        Send Alert message to network administrator.
   }
```

VI. CONCLUSIONS

In this paper we study the theory of ARP spoofing attack and various existing techniques proposed to defend against this attack, and then, we proposed a comprehensive method to deal with ARP spoofing problem.

As the method we proposed to probe the authenticity of every ARP packet is very active, the time delay between capturing the packets and detecting spoofing attack is minimum. We send one trap ICMP ping packet for each spoofed ARP packet on the network and then infer its authenticity according to the response to our packet. This method can also detect correct IP-MAC address mappings of both the true host and the malicious host during an actual attack.

REFERENCES

[1] Stevens, TCP/IP Illustrated: vol. 1 (2001).

[2] Zouheir Trabelsi and Khaled Shuaib, "Spoofed ARP Packets Detection in Switched LAN Networks". pp. 81–91, 2008.

[3] T. Demuth and A. Leitner, "ARP spoofing and poisoning: Traffic tricks" Linux Magazine, 56: 26–31, July 2005.

[4] http://node99.org/projects/arpspoof/arpspoof.pdf

[5] M. Gouda and C -T. Huang, "A secure address resolution protocol", Computer Networks, 41(1):57–71, Jan 2003.

[6] D Bruschi, A. Ornaghi, and E. Rosti, "S-ARP: A secure address resolution protocol", In Proceedings of the19th Annual Computer Security Applications Conference (ACSAC '03), Dec. 2003.

[7] M.Barnaba.anticap, <http://www.antifork.org/viewcvs/trunk/anticap>.

[8] Cisco Systems. Configuring Dynamic ARP Inspection, chapter 39, pages 39:1–39:22, 2006.

[9] Mihai Dorobanţu and Mihai L. Mocanu, "A simple way to capture network traffic: the windows packet captures (Winpcap) architecture".

[10] http://www.chmag.in/article/feb2012/cain-and-abel-black-art-arp-poisoning.

