Design of a Kernel Module for providing a Feasible Solution to ARP Spoofing Problem

[1]Vijaya saraswathi.R, [2]LakshmiKumari.CH, [3]P. Radhika

[1,3]Assistant Professor, VNRVJIET, Bachupally, Hyderabad
[2]Assistant Professor, MGIT, Gandipet,Hyderabad.
Email : [1]vijayasaraswathir@gmail.com, [2]lakshmi.itsmine@gmail.com, [3]radhika.pathi123@gmail.com

Abstract - ARP cache poisoning, a serious threat affects LAN Community by spoofing the sender's original Identity. The Identity here refers to the sender's MAC Address. ARP Cache Poisoning otherwise known as ARP spoofing is a technique in which attacker fakes Address Resolution Protocol (ARP) messages onto a Local Area Network intended to associate the attacker's MAC address with the IP address of another host diverting any traffic meant for that IP address sent to the attacker instead. This is a longstanding problem that is seriously affecting the host's Cache .The problem is serious to the extent that any neophyte with basic knowledge on networking can attack the system. ARP Spoofing results in other attacks such as Man In The Middle Attack(MITM), Denial of Service attacks(DoS). In spite of many solutions proposed, the problem still remains alive, counterfeiting the original sender. To address this problem, in our paper we have proposed a Linux based solution ARP Spoof Protection System, a software that manages authentication of sender within LAN Community. Our solution is based on a loadable kernel module that detects and defeats ARP Poisoning attacks maintaining efficient system performance all the time.

Keywords: ARP Spoofing, ARP cache poisoning, MITM, DoS

I. INTRODUCTION

ARP(Address Resolution Protocol) , a member of TCP/IP protocol suite is used to translate between IP addresses and MAC Addresses. The Address Resolution Protocol (ARP) is used to associate known IP addresses to unknown physical hardware, MAC, addresses. However, it is well-known that ARP control message can be spoofed, and ARP cache can be poisoned [14]. ARP spoof (or ARP cache poisoning) is a serious security problem on Local Area Network (LAN) leading to Denial of Service (DoS) or Man in the Middle (MITM) attacks. Recently, there have been several solutions, proposed to solve the ARP spoof problem. In this paper, we point out in details the drawbacks of the previous proposed solutions (such as low compatibility, infeasibility, inefficiency, too high cost, and unmanageability). Furthermore, we have proposed a new system of a better solution. A prototyped program corresponding to our design has been implemented and experimented. The experimental results show impressive features.

The remainder of this paper is organized as follows. We begin in Section II by discussing literature survey. In Section III, we set up criteria for our system model. The details of design and algorithms are discussed in Section IV. Section V discusses the details of experimental environment and results on the prototyped program. Finally, the conclusions and the direction of future work are given.

II. LITERATURE SURVEY

A. Address Resolution Protocol

Address Resolution Protocol (ARP) [2] is specified in RFC 826. It is important to convert IP address into Media Access Control (MAC) address in order to communicate in a LAN. When a PC wants to send an IP datagram as an Ethernet frame to another PC whose MAC address is not yet known, it broadcasts an ARP request message to ask for the MAC address associated with the destination IP address. Every PC on the same LAN receives the ARP request and checks whether the requested IP address is its. The PC with the requested IP then sends a unicast ARP reply message to the sender with the IP-MAC address pair. Each PC dynamically keeps a list of IP-MAC address pairs in its ARP cache according to the received ARP request and reply messages. This dynamic ARP entries is to minimized the number of ARP request and reply messages. However, the weakness is that ARP is stateless protocol. It updates entries in the cache after receiving ARP reply even though the corresponding ARP request was never been sent. Even worst, ARP has no scheme to validate the IP-MAC address pairs from ARP reply and ARP request messages. This weakness causes an attack, called ARP Spoof.

B. ARP Spoof

ARP Spoof (also known as, ARP cache poisoning) is a well known technique of hackers to forge ARP request or ARP reply messages. The main two purposes are MITM and DoS attacks. For the MITM purpose, the attacker may spoof two PCs at the same time. The attacker then can listen to the traffic between those two PCs. For the DoS attack purpose, an attacker may poison ARP table of a duped so that every packet that the duped sends is sent to the wrong MAC address. So, the duped is blocked from communication. In addition, ARP Spoof is an initial hacking technique used to start other serious attacks, such as sidejacking, HTTPS hacking, Pharming and so on.

C. Previous Solutions and Drawbacks

Recently, there have been several solutions proposed to solve the ARP spoof problem. However, most of them have some critical drawbacks. The previous solutions may be grouped as follows:

1) Modifying ARP using some cryptographic techniques

S-ARP [1] and TARP [3] are some sampled proposals of this kind of solutions. They all give a suggestion to change the design of ARP implementation using some cryptographic techniques. However, until now, ARP has never been really changed in almost all Operating Systems (OS). So, it is infeasible solution. Their new ARP protocol is hardly in the real world and is not compatible with the standard ARP. Furthermore, the add-on cryptographic features have caused some serious performance penalty in some proposals (such as S-ARP).

2) Kernel-based patch

Anticap [6] and Antidote [10] are some examples of solutions to ARP spoof that suggest a patch to some specific OS to protect against ARP spoof. However, their patch can be used only with some specific kernel. ARP in that kernel after patched may not be compatible and interoperable with the ARP mechanisms in other un-patched kernels.

3) Port security on switch

This group of solutions suggests using an expensive switch that can support port security (such as Dynamic ARP Inspection (DAI) [3]). This kind of switch can help detect ARP spoof easily. However, the main problem of this solution is cost. For most of organization, it would not be possible to change all LAN switches to the high-end ones (in particular at the access level of network).

4) Manually configuration of static ARP entries

The most common way to protect against ARP spoof is manually configuring static ARP entries at everyPC. However, this solution is not manageable for network administrators of a rather large organization. Also, it would be rather difficult to monitor all end users of any organization to configure static ARP entries properly.

5) ARP spoof detection & protection software

There have been several programs, proposed to detect and protect against ARP spoof. Yet, most of them work inefficient. Some of them can only detect but not protect, such as XArp [15], ARPWatch [5]. Several programs (such as Anti Netcut [12], NoCut [16], and AntiARP [11]) thse are tested in our lab [14] and found an inefficient of protection. AVASS [13] is ARP spoof detection/protection software, designed and implemented. From our testing, it demonstrates effectiveness. However, we have also found that the design of AVASS is still not very efficient and easy to manage.

III. THE CRITERIA OF SYSTEM MODEL

We define some key criteria in designing new ARP spoof detection protection software as follows:

1) Feasibility: The solution must be feasible to implement (in both small and medium size of networks). Also, the solution should be ready to utilize immediately without waiting for network protocols **Effectiveness**: The solution must be effective in both detect and protect against ARP spoof. Also, it must be effective for both hosts and gateways. Furthermore, it must be effective for different kinds of ARP spoof attacks (such as using ARP request message or ARP reply message, DoS or MITM).

4) Performance: The solution should have a minimal overhead, and minimize the performance penalty.

5) Cost: The solution should not be too expensive. So, it must not need any high-priced switches with port security.

6)Manageability: The solution should be manageable. It should be ease both network administrators and users in fighting against ARP spoof.

IV. DESIGN AND ALGORITHMS

A. ARP Spoof Antivenin

We compare "ARP Spoof' attack of any IP address as "germ" infection. So, we use a valid "static ARP entry" (of IPMAC mapping) as a "antivenin" to protect against the germ of that IP address. By specifying valid static ARP entries into our ARP cache for all IP addresses that our hosts want to communicate, we therefore have all needed antivenins against the ARP spoof hacking. However, as mentioned previously, managing valid static ARP entries (antivenins) for all hosts in the organization is a tedious job. So, we design a system to automate the "antivenin" provision and injection jobs. The details are in the following sections.

B. Dynamic ARP spoof Protection System (DAPS)

Our new design is called "Dynamic ARP spoof Protection and Scrutiny(DAPS)" system. As shown in Figure 1, it consists of four components as follows:

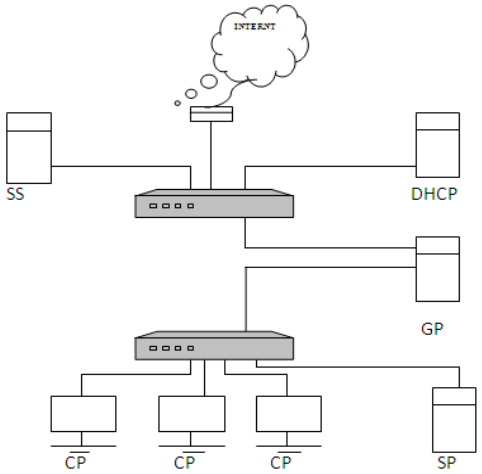

Figure 1 DAPS Components

1) Gateway Protection (GP)

ARP spoof hacking at gateway is very popular and very effective. The hackers can easily denie of service all the PCs in the LAN from accessing the Internet by ARP spoof hacking to gateway. Also, by MITM attack the gateway and any specific PC, the hacker can disclose important information transmitted between that PC and the outside-LAN service (such as web mail, online banking and so on). From our study, we have found that most of ARP spoof hacks (both DoS and MITM purposes) generally target the gateway. Therefore, protecting gateway from ARP spoof is very important. GP is a component to protect the gateway from ARP spoof attack. GP is actually a PC router (setting up from a Linux box) enhanced with ARP spoof protection software. It can be used as a gateway of any small organization or as a "gateway ARP spoof wall" for the organization that already has a commercial router. The details of GP are similar to our design of Vaccine Gateway (VG) in AVASS [14]. The difference is that GP is more proactive than VG. Unlike VG, GP is not relied on network administrators to specify all vaccines. Yet, GP is Equipped with a module to snoop and learn IP-MAC pairs from DHCP packets coming through gateway and stored them as vaccines for all dynamic IPs automatically. For the static IP address, a network administrator has to manually configure the vaccine into GP. These vaccines are not only used to protect the gateway from ARP spoof attacks but also protect clients and servers in the LAN from ARP spoof as well.

2) Client Protection (CP) / Server Protection (SP)

CP is a DHCP client that has installed a CP program. SP is a server, which has static IP address (manually configured by network administrators), and installed an SP program. IP-MAC pairs on the clients and servers that installed CP or SP are always valid because they receive the vaccines from GP and add them as static ARP entries in their ARP cache.

SP has also a job to provide its vaccine to store at GP. After that, GP can distribute the vaccine to all CPs in the same LAN with SP. The communication between CP/SP and GP is authenticated by using digital signatures. Due to the DHCP snooping module at GP, DAPS is more proactive than AVASS, and requires less manual configuration by network administrators.

3) Scrutiny Server (SS)

SS is a machine equipped with ARP spoof surveillance system program. The main function of SS is receiving warning messages from SP, CP and GP when they detect ARP spoof attacks on their LAN.SS in a small LAN environment can be integrated into GP. However, for MAN (Metropolitan Area Network) such as campus network, SS would be installed in a separated server.

C. Simple ARP Spoof Detection Algorithms (SADA)

As shown in Figure 2, SADA is an algorithm for detect ARP spoof. SADA is run at CP, SP and GP. Due to the vaccines (provided by GP DHCP snooping module, SP and manual configuration by administrators), all hosts in the LAN are now safe from ARP spoof. Furthermore, it is very easy to detect ARP spoof by comparing IP-MAC pairs from ARP request message ({qM, qIP}) and IP-MAC pairs from ARP reply message ({rM, rIP}) with the local IP-MAC pairs in antivenin stores of CP/GP/SP ({cM, cIP}). If the arriving ARP request or reply messages give IP-MAC pairs different from the ones in antivenin stores, CP/SP/GP will expect ARP spoof attack and send warning messages to.

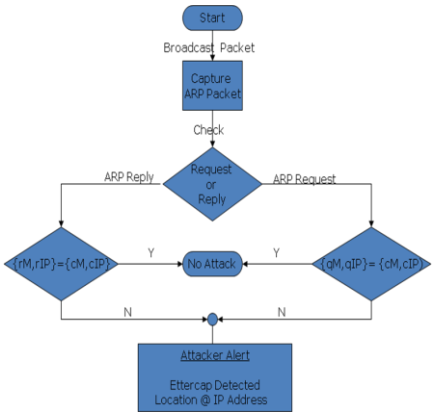

Figure 2: Simple ARP Spoof Detection Algorithm(SADA)

V. EXPERIMENTS

A. Tools

We have implemented a prototyped system of DAPS using Linux. To experiment on our solution, we test the implemented DAPS by running on CentOS Linux 5. ARP spoof software, namely Cain Abel [9], Packet Builder [8], Acai Netcut [17] and Ettercap are used as ARP spoof attack tools. We used Ettercap.

B. Experimental Environment

GP is set on a PC router using CentOS Linux. CP is set on CentOS Linux boxes. We used VMware to create virtual workstations. VMware software provides a completely virtualized set of hardware to the guest

operating system. In this way, VMware virtual machines become highly portable between computers, because every host looks nearly identical to the guest. For attacking simulation, we use ARP spoof tool Ettercap to attack.

C. Experimental Results

We have run extensive experiments to test DAPS in many aspects. due to the limitation of space, here we demonstrate some results.The compiled code of DAPS can also be requested from the authors to further test. new solution an successfully detect and protect against ARP spoof attacks (both MITM and DoS attacks). They can also tolerate to a heavy attacks .DAPS has lower overhead and easier to manage than.Before ARP Spoofing attack the host1 i.e. 192.168.1.45 ARP Cache contents are shown in the Figure 3.

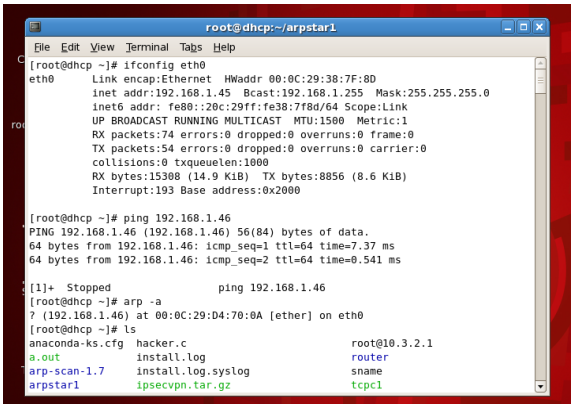

Figure 3 : Before ARP Spoofing attack the host1

After generating arpstar.ko file that has to be inserted into a kernel of specified host to prevent ARP poisoning attack. After attacking host1, The ARP Cache contents of it which are not modified are as shown in the Figure 4.

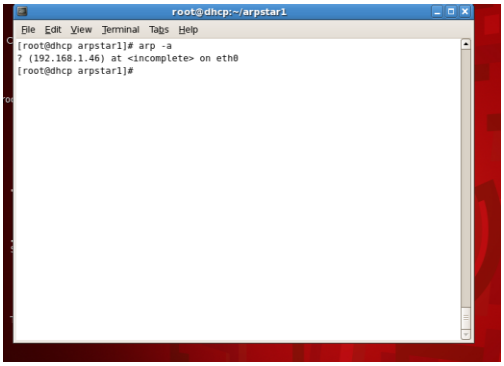

Figure 4: After attacking host1, The ARP Cache contents of it.

In comparison with other solutions, we found that our solution is very successful in detection after attacking host1 the arp cache contents are shown above

VI. CONCLUSIONS

ARP spoof is a serious problem of LAN security. Although there have been several solutions recently proposed to solve the problem, we have proposed a new design of ARP spoof detection and protection solution using Linux. Implementation of this technique requires a thorough understanding of the networking system in Linux to protect important hosts in LAN. This paper deals with the changes to an existing algorithm for ARP Cache poisoning Prevention and detection for a host running in Linux. The experimental results show the favorable features of our solution, and demonstrate that it is a good solution. In the future, we plan to extensively test our software in real sites such as, campus network, and company network .

REFERENCES

[1] D. Bruschi, A. Omaghi and E. Rosti. "S-ARP: A secure address resolution protocol", In Proceedings of the Annual Computer Security Applications Conference, December 2003

[2] D. C. Plummer. "An ethemet address resolution protocol", IETF RFC 826, November 1982.

[3] W. Lootah, W. Enck and P. McDaniel. "TARP: Ticketbased address resolution protocol", In Proceedings of the Annual Computer Security Applications Conference, December 2005.

[4] Cisco Systems. "Configuring Dynamic ARP Inspection", chapter 39, pp. 39: 1-39:22. Catalyst 6500 Series Switch Cisco lOS Sofware Configuration Guide, ReleaseI2.2SX.

[5] L. N. R. Group. "Arpwatch, the ethemet monitor program; for keeping track of ethemet/ip address pairings", http://ee.1bl.gov/arpwatch.tar.gz

[6] M. Bamaba. "Anticap" http://www.antifork.org/svn/trunk/anticap

[7] V. Ramachandran and S.Nandi. "Detecting ARP Spoofing: An Active Technique", In Proceedings of ICISS, December 2005.

[8] Packet Builder. http://www.colasoft.com/packet_builder

[9] M. Montoro. "Cain&Abel version 4.9.19", http://www.oxid.it/cain.html

[10] I.Teterin, "Antidote" http://online.securityfocus.com/archive/l/299929

[11] ColorSoft. "AntiARP", http://www.antiarp.com/English/e_index.htm

[12] "Anti netcut version 2.0", http://www.tools4free.net.

[13] P. Casaby, T. Chuachan, S. Puangpronpitag, "ARP Spoof Vaccination And Surveillance System", In Proceedings of NCSEC, November 2008.

◈◈◈