On investigating ARP spoofing security solutions

Zouheir Trabelsi* and Wassim El-Hajj

Information Security Department, College of Information Technology,
UAE University, P.O. Box 17551, Al-Ain, UAE
E-mail: trabelsi@uaeu.ac.ae
E-mail: welhajj@uaeu.ac.ae
*Corresponding author

Abstract: The address resolution protocol (ARP) has proven to work well under regular circumstances, but it was not designed to cope with malicious hosts. By performing ARP spoofing attacks, a malicious host can either impersonate another host [man-in-the-middle attack (MiM)] and gain access to sensitive information, or perform denial of service attack (DoS) on target hosts. Several security solutions, such as high-cost LAN switches and intrusion detection or prevention systems (IDS/IPS), are currently used to detect and prevent these attacks. In this paper, we evaluate, through extensive practical experiments, how effective these security solutions are in detecting ARP spoofing. We clearly show that ARP spoofing has not been given enough attention by most common security solutions which lack efficient detection and prevention mechanisms. We then propose an optimal algorithm that is capable of detecting all various ARP spoofing attacks; especially those not detected using the current mechanisms. The suggested algorithm can be easily integrated in any available security solution with very minimal overhead.

Keywords: intrusion detection system; ARP spoofing; man-in-the-middle attack; MiM; denial of services attack; DoS.

Reference to this paper should be made as follows: Trabelsi, Z. and El-hajj, W. (2010) 'On investigating ARP spoofing security solutions', *Int. J. Internet Protocol Technology*, Vol. 5, Nos. 1/2, pp.92–100.

Biographical notes: Zouheir Trabelsi is an Associate Professor at the College of Information Technology, the United Arab Emirates University, and the Head of the Information Security Program. He received his PhD from Tokyo University of Technology and Agriculture, Japan, in the field of Computer Science. He was a Computer Science Researcher at the Central Research Laboratory of Hitachi in Tokyo, Japan. He was also a Visiting Assistant Professor at Pace University, New York, USA. His research areas are mainly networking security, intrusion detection and prevention, firewalls, and TCP/IP covert channels.

Wassim El-Hajj received his BS degree from the American University of Beirut in 2000 and MS and PhD degrees in 2002 and 2006, respectively, from Western Michigan University (WMU), all in Computer Science. He is currently an Assistant Professor in the College of Information Technology at UAE University. He is the recipient of numerous recognitions, most notably, the WMU Excellence in Research Award for three years in a row, Honourable mention for Graduate Research and Creative Scholar, the Outstanding Graduate Student Award for two consecutive years, and the Teaching Effectiveness Award which is considered the highest teaching award at Western Michigan University. His research interests include security, network planning, and bioinformatics.

1 Introduction

In a LAN network, malicious hosts can perform various types of attacks, such as DoS attack, MiM attack, ARP spoofing, buffer overflow, and malicious sniffing. ARP spoofing is a hacking method that spoofs the contents of the ARP table of a remote computer on a LAN network. Using ARP spoofing, malicious users can corrupt the ARP caches of target hosts in order to perform MiM or DoS attacks. ARP spoofing is an easy attack to conduct, very harmful, and presents a very serious threat. This attack can be performed by novices or *script kiddies*, using widely available and easy to use tools specially designed for that purpose. Examples of tools are ARP Spoof Tool (2009), Winarp (2009), SwitchSniffer (2009), WinArpSpoof (2009), WinArpAttacker (2006), and Cain and Abel (2009). Skilful malicious users can use packet generators, such as Frameip Packet Generator (2009) to build the appropriate ARP packets that allow performing ARP spoofing attack.

Due to the importance of this problem, several security solutions, ranging from highly-cost LAN switches, IDS/IPS hardware appliances and software tools, to unified threat management (UTM[1]) appliances, integrate mechanisms to cope with ARP spoofing, but each has its limitations.

In this paper, we evaluate common security solutions regarding their ability to detect ARP spoofing. The paper provides analysis based on heavy practical experiments. It shows clearly that ARP spoofing has not been given enough attention by most common security solutions, even though this attack presents a serious threat, is very harmful and more dangerously is easy to conduct. In fact, despite the fact that some highly-cost security solutions claim to deal with most common network intrusions, they are still unable of fully and efficiently detect ARP spoofing attack. As a solution, we propose an optimal algorithm that can be implemented in security solutions to effectively detect and prevent ARP spoofing attacks.

This paper is an extension of an early published work (Trabelsi and El-Hajj, 2009) which lacks appropriate description and analysis of the proposed algorithm. The rest of the paper is organised as follows. Section 2 presents a brief description of the ARP protocol and its components. Section 3 highlights the ARP DoS and MiM attacks. Section 4 lists and illustrates all possible abnormal ARP packets. Section 5 depicts the experiments carried out on various security solutions designed to deal with network intrusions. Section 6 discusses and analyses the experiments' results. Section 7 explains our proposed algorithm and Section 8 concludes the paper and presents the future work.

2 ARP protocol and ARP cache

In LAN networks, ARP protocol (Plummer, 1982) messages are exchanged when one host knows the IP address of a remote host and wants to discover the remote host's MAC address. The ARP protocol specifies no rules to maintain consistency between the ARP header and the Ethernet header. This means that one can provide uncorrelated addresses between these two headers. For example, the source MAC address in the Ethernet header can be different from the source MAC address in the ARP message header.

Each host in a network segment has a table, called ARP cache table, which maps IP addresses with their corresponding MAC addresses. New entries in an ARP cache can be created or already existing entries can be updated by ARP request or reply messages. For more details on the process of creating and updating ARP caches entries in common operating systems, the reader can refer to the work presented in (Trabelsi and Shuaib, 2008).

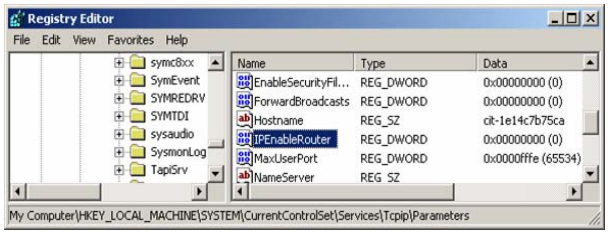
By storing the MAC addresses in the ARP table, a potential weakness arises. A remote hacker can change MAC to IP address entries, which could cause traffic to be redirected from the correct target to a target of the hacker's choice. This attack is called ARP spoofing, or ARP Cache poisoning. It is the malicious act of introducing a spurious IP address to MAC address mapping in the ARP cache of a target host. This can be done by manipulating directly the ARP cache of a target host, independently of the ARP messages sent by the target host. To do that, the malicious host can either add a new fake entry in a target host's ARP cache, or update an already existing entry by fake IP and MAC addresses.

3 MiM and DoS attacks

MiM and DoS are very common attacks that can be easily performed in a network segment. These attacks use usually spoofed ARP packets to corrupt the ARP caches of target hosts. The classic MiM attack relies upon convincing two hosts that the computer in the middle is the other host. The attack consists of re-routing (redirecting) network traffic between two target hosts to a malicious host (usually the attacker host). Then, the malicious host will forward the received packets to its real destination, so that the communication between the two target hosts will not be interrupted and the two hosts' users will not notice that their traffic is being redirected and sniffed by a malicious user.

In such attack, the malicious user should enables the host's IP packet routing option in his/her host, in order to act as a router and be able to forward the redirected packets (Figure 1). Then, using ARP spoofing, the malicious user corrupts the ARP caches of the two target hosts, in order to force the two hosts to forward all their packets to the malicious host.

Figure 1 A screen shot showing how to enable IP routing in Windows XP host (see online version for colours)



It is important to notice that if the malicious host corrupts the ARP caches of the two target hosts without enabling its IP packet routing, then the two hosts will not be able to exchange packets and it will be a DoS attack. In this case, the malicious host does not forward the received packets to their legitimate destination. This is extremely potent when we consider that not only can hosts be poisoned, but routers/gateways as well. All internet traffic for a host could be intercepted by performing a MiM attack on the host and the LAN's router.

We assume that hosts A and B are the two target hosts and host C is the malicious host. To perform MiM attack, host C enables its IP packet routing and corrupts the ARP caches of hosts A and B. Figure 2 shows the initial entries in the ARP caches of hosts A and B, before the ARP spoofing attack.

Host C sends a fake ARP request packet to host A in order to corrupt its ARP cache with the fake entry IP_Host_B/MAC_Host_C (Figure 3).

Also, host C sends a fake ARP request packet to host B in order to corrupt its ARP cache with the fake entry IP_Host_A/MAC_Host_C (Figure 4).

Figure 2 The entries of the ARP caches of hosts A and B before the ARP cache poisoning attack

Host A's ARP Cache	
IP address	**MAC address**
IP_of_host_B	MAC_address_of_host_B
IP_of_host_C	MAC_address_of_host_C

Host B's ARP Cache	
IP address	**MAC address**
IP_of_host_A	MAC_address_of_host_A
IP_of_host_C	MAC_address_of_host_C

Figure 3 Fake ARP request sent to host A by the malicious host C

ARP header
Operation code =1 (Request)
Source IP address = **IP_Host_B**
Source MAC address = **MAC_Host_C**
Destination IP address = IP_B
Destination MAC address = 00-00-00-00-00-00
Ethernet header
Source MAC address = Any
Destination MAC address = **MAC_A**
Ethernet Type (=0x0806 for ARP message)

Figure 4 Fake ARP request sent to host B by the malicious host C

ARP header
Operation code =1 (Request)
Source IP address = **IP_Host_A**
Source MAC address = **MAC_Host_C**
Destination IP address = IP_B
Destination MAC address = 00-00-00-00-00-00
Ethernet header
Source MAC address = Any
Destination MAC address = **MAC_B**
Ethernet Type (=0x0806 for ARP message)

Figure 5 The entries of the ARP caches of hosts A and B after the ARP cache poisoning attack

Host A's ARP Cache	
IP address	**MAC address**
IP_of_host_B	MAC_address_of_host_C
IP_of_host_C	MAC_address_of_host_C

Host B's ARP Cache	
IP address	**MAC address**
IP_of_host_A	MAC_address_of_host_C
IP_of_host_C	MAC_address_of_host_C

After the attack, host A associates host B's IP with host C's MAC, and host B associates host A's IP with host C's MAC (Figure 5). All packets sent by host A to host B will first go to host C. Then, host C forwards them to host B, since IP packet routing in host C is enabled. Moreover, all packets sent by host B to host A will first go to host C, then, host C forwards them to host A.

However, in DoS attack, target hosts are denied from communicating with each other, or with the internet. This is done simply by corrupting their ARP caches with fake entries including non-existent MAC addresses, or by disabling the IP packet routing option in the malicious host, so that the received and redirected traffic will not be forwarded to its real destination.

4 Abnormal ARP packets

ARP spoofing uses abnormal ARP packets to corrupt the ARP caches of target hosts. There are many security solutions claiming to be able to cope with ARP spoofing. These solutions are found usually in highly-cost switches, network IDS/IPS and UTM hardware appliances, and IDS/IPS software tools.

Table 1 and Table 2 list all possible abnormal ARP request and reply packets, respectively. Four possible types of abnormal ARP request packets and six possible types of abnormal ARP reply packets have been identified, namely:

- *P#1, P#5, and P#7*: Security devices should keep track of IP-to-MAC address mappings. Every ARP packet contains a mapping of IP-to-MAC address. ARP requests contain the IP-MAC mapping of the sender. ARP replies contain the IP-MAC mapping of the machine resolved. Every mapping is inserted into a database. If a monitored mapping breaks current mappings, an alert is generated. IP-to-MAC mappings database can fill either automatically or manually.
- *P#2, P#6, and P#8:* ARP packets have special restrictions. In an ARP request and reply packet, the Ethernet source MAC address has to match the ARP source MAC address. In ARP reply, the Ethernet destination MAC address has to match the ARP destination MAC address.
- *P#3:* A normal ARP request needs to be sent to the broadcast MAC address, and not to a unicast MAC address. Such packets are used by ARP spoofing software to spoof only a specific machine and not all machines on a network.
- *P#9:* A normal ARP replies needs to be sent to unicast MAC address, and not to the broadcast MAC address. Such packets are used by ARP spoofing software to spoof only a specific machine and not all machines on a network.
- *P#4 and P#10:* There are fields in the ARP packet that have restrictions regarding the values they can adopt. These values should be checked for correctness. ARP mappings may not contain certain IP addresses. These include broadcast and multicast as well as null addresses.

Moreover, some MAC addresses in ARP packets are highly suspicious. No IP-to-MAC mapping should, for example, have the MAC broadcast, multicast or null address assigned. Every ARP packets IP addresses need to be in the same subnet. An ARP packet with IP addresses that are not in the network interfaces configured subnet are suspicious and will be alerted.

Table 1 and Table 2 show that only abnormal packets P#1 and P#5 can corrupt ARP caches of target hosts with fake IP-MAC entries. The remaining abnormal ARP packets do not corrupt ARP caches. However, they may still be harmful and can produce DoS situations. Also, unknown attacks can also use these abnormal ARP packets. Therefore, the need to create an efficient security solution that detects all kind of abnormal ARP traffic becomes a must.

5 Experiments

Through extensive practical experiments, we evaluated how effective common security solutions are in detecting ARP spoofing. The selected security solutions are classified into four categories, namely:

- LAN switches
  - a Cisco switch 3560 Series
  - b Juniper Switches EX3200 Series
- Software IDS/IPS
  - a Snort IDS
  - b XArp 2 tool
  - c Sax2 NIDS
- IDS/IPS hardware appliances
  - a Cisco IPS 4255 Series
  - b TopLayer Model 5000
  - c IBM ISS Proventia Model GX4004C
  - d SourceFire
  - e TippingPoint 50
- UTM devices
  - a Juniper Netscreen 50

Table 1 List of possible abnormal ARP request packets

Packet number	*P#1*	*P#2*	*P#3 (Unicast ARP request)*	*P#4 (Unexpected IP or MAC address in ARP request packets*)*
		ARP header		
ARP operation	1	1	1	1
Source IP	IP_A*	IP_A		0.0.0.0
				255.255.255.255
				Multicast
				Not in the network subnet
Source MAC	MAC_X*	MAC_A*		00-00-00-00-00-00
				ff-ff-ff-ff-ff-ff
				Multicast
Destination IP				0.0.0.0
				255.255.255.255
				Multicast
				Not in the network subnet
Destination MAC				
		Ethernet header		
Source MAC		MAC_X		00-00-00-00-00-00
				ff-ff-ff-ff-ff-ff
				Multicast MAC
Destination MAC			Unicast	00-00-00-00-00-00
				Unicast or multicast
Does the packet corrupt the ARP cache?	Yes	No	No	No

Notes: IP_A*: is the IP address of a host A.
MAC_A*: is the MAC address of a host A.
MAC_X*: is a MAC address of a non-existent host.
Unexpected IP or MAC address in ARP request packets*: These addresses are considered unexpected and consequently ARP request packets should not have such addresses.

Table 3 shows the identified security solutions that can perform ARP inspection on ARP packets regardless of the type of inspection.

In the upcoming experiments, we excluded from the above list, the IPS TippingPoint 50 since it includes ARP inspection that is not concerned with the detection of ARP spoofing attack. TippingPoint 50 uses three ARP signatures to check whether or not the hardware type and protocol type fields in the Ethernet header contain valid values. This type of inspection does not allow detecting ARP spoofing.

Among the security solutions that include ARP inspection mechanisms (Table 3), Table 4 shows the ones that can totally or partially detect the abnormal ARP packets listed in Table 1 and Table 2.

Using the data presented in Table 4, we can easily notice that no system offers an ideal solution for the problem of ARP spoofing detection. Out of the detection systems, the XArp 2 tool seems ideal in terms of the number of detected abnormal ARP packets. Snort IDS seems to be a good alternative, but both of them perform only detection and are not enable to prevent ARP spoofing attack. The prevention/blocking systems, such as Cisco switches 3560 Series (2009) or Juniper switches EX3200 Series (2009), are the most ambitious ones, but require usually complex installations. In addition, the high costs of these switches make this solution prohibitive for many companies (Abad and Bonilla, 2007). Cisco IPS is a prevention system and is a limited alternative solution since it can deal with few types of abnormal ARP packets (P1 and P5). Nevertheless, it is important to remember that the packets P#1 and P#5 are the most used ARP packets during ARP spoofing, since they are the only packets that can corrupt the ARP caches of target hosts.

Table 2 List of possible abnormal ARP reply packets

	P#5	*P#6*	*P#7*	*P#8*	*P#9 (Broadcast ARP reply)*	*P#10 (Unexpected IP or MAC address*)*
			ARP header			
Operation	2	2	2	2	2	2
Source IP	IP_A	IP_A				0.0.0.0 255.255.255.255 Multicast Not in the network subnet
Source MAC	MAC_X	MAC_A				00-00-00-00-00-00 ff-ff-ff-ff-ff-ff Multicast
Destination IP			IP_B*	IP_B		0.0.0.0 255.255.255.255 Multicast Not in the network subnet
Destination MAC			MAC_X	MAC_B*		00-00-00-00-00-00 ff-ff-ff-ff-ff-ff Multicast
			Ethernet header			
Source MAC		MAC_X				00-00-00-00-00-00 ff-ff-ff-ff-ff-ff Multicast
Destination MAC				MAC_X	ff-ff-ff-ff-ff-ff	00-00-00-00-00-00 ff-ff-ff-ff-ff-ff Multicast
Does the packet corrupt the ARP cache?	Yes	No	No	No	No	No

Notes: IP_B*: is the IP address of a host B.
MAC_B*: is the MAC address of a host B.
Unexpected IP or MAC address in ARP reply packets*: These addresses are considered unexpected and consequently ARP reply packets should not have such addresses.

Table 3 Security solutions performing ARP inspection

	Type	*Performing ARP inspection (Yes or No)?*	*Detection or prevention solution?*
Cisco Switch 3560 Series	Switch	Yes	Prevention
Juniper Switches EX3200 Series	Switch	Yes	Prevention
Snort IDS	IDS software tool	Yes	Detection
XArp 2 tool	IDS software tool	Yes	Detection
Sax2 NIDS	IDS software tool	Yes	Detection
Cisco IPS 4425 Series	IPS appliance	Yes	Detection
TopLayer Model 5000	IPS appliance	No	Detection
IBM ISS Proventia Model GX4004C	IPS appliance	No	Detection
SourceFire	IPS appliance	No	Detection
TippingPoint 50	IPS appliance	Yes	Detection
Juniper Netscreen 50	UTM	No	Detection

Sax2 NIDS cannot detect any abnormal packet described in Tables 1 and 2. However, it can detect ARP request storm traffic and ARP scanning traffic. This type of traffic uses normal ARP packets and it will be described in Section 5.3.

In Table 4, *partially detected* means that the device detects all or some ARP request or reply packets that have unexpected IP source or destination addresses, and/or unexpected MAC source or destination addresses.

5.1 Cross-layers ARP inspection

In order to be able to detect the abnormal ARP packets P#2, P#6, and P #8 described in Table 1 and Table 2, a security solution requires including an ARP inspection mechanism that can perform cross-layers ARP inspection between the Ethernet and ARP headers. In an ARP request and reply packet, the Ethernet source MAC address has to match the ARP source MAC address. However, in ARP reply, the Ethernet destination MAC address has to match the ARP destination MAC address. Table 5 shows the security solutions that include cross-layers ARP inspection mechanism.

5.2 ARP stateful inspection

ARP replies should normally follow ARP requests. A stateful detection process should remember all ARP requests originating and match them to ARP replies. Many ARP spoofing tools send ARP replies that are not requested. Table 6 shows the list of security solutions that perform ARP stateful inspection on ARP requests against ARP replies. ARP inspection mechanism might give false positives in some cases as machines want to distribute their IP-to-MAC mapping to other machines that did not request it. Among the above tested security solutions, XArp 2 tool and Sax2 IDS are the only solutions that perform ARP stateful inspection

Table 4 Detection of abnormal ARP request and reply packets

	P#1	*P#2*	*P#3*	*P#4*	*P#5*	*P#6*	*P#7*	*P#8*	*P#9*	*P#10*
Cisco switch 3560 series	Detected	Detected	Not detected	Not detected	Detected	Detected	Detected	Detected	Not detected	Not detected
Juniper switches EX3200 series	Detected	Detected	Not detected	Not detected	Detected	Detected	Detected	Detected	Not detected	Not detected
Snort IDS	Detected	Detected	Detected	Not detected	Detected	Detected	Detected	Detected	Not detected	Not detected
XArp 2 tool	Detected	Detected	Detected	Partially detected	Detected	Detected	Detected	Detected	Detected	Partially detected
Sax2 NIDS	Not detected	Not detected	Not detected	Not detected	Not detected	Not detected	Not detected	Not detected	Not detected	Not detected
Cisco IPS series 4255	Detected	Not detected	Not detected	Partially detected	Detected	Not detected	Detected	Not detected	Not detected	Partially detected

Table 5 Security solutions performing cross-layers ARP inspection

	Performing cross-layers ARP inspection?
Cisco switch 3560 series	Yes
Juniper switches EX3200 series	Yes
Snort IDS	Yes
XArp 2 tool	Yes
Sax2 NIDS	No
Cisco IPS 4425 series	No

Table 6 Security solutions including ARP stateful inspection

	Performing cross-layers ARP inspection?
Cisco switch 3560 series	No
Juniper switches EX3200 series	No
Snort IDS	No
XArp 2 tool	Yes
Sax2 NIDS	Yes
Cisco IPS 4425 series	No

5.3 ARP request storm and ARP scan

ARP request storm: dynamic ARP entries remain in the ARP cache for few minutes then they are removed if they are referenced. Consequently, to keep the ARP cache of a target host corrupted with fake entries, malicious users may storm the target host with ARP request packets. In other words, the malicious host keeps sending continuously fake ARP request packets to the target host. If the number of ARP request packets per second exceeds the ARP request threshold, then this is an indication that an ARP request storm is taking place. Table 7 shows the security solutions that include mechanisms to detect ARP request storm and/or ARP scanning. Among the above tested security solutions, Sax2 IDS is the only solution that is able to detect ARP request storm and ARP scanning.

ARP scan: the possible reason of ARP scanning in LAN networks is surveillance software running, host infected with virus, or the virus is doing ARP scanning.

Table 7 Security solutions including ARP request storm and/or ARP scan detection mechanisms

	Detect ARP request storm?	*Detect ARP scan?*
Cisco switch 3560 series	No	No
Juniper switches EX3200 series	No	No
Snort IDS	No	No
XArp 2 tool	No	No
Sax2 NIDS	Yes	Yes
Cisco IPS 4425 series	No	No

6 Experiments' results analysis

The experiments in this work show clearly that ARP spoofing is not fully detected by most common security solutions. This is because of the absence of an efficient ARP spoofing detection algorithm.

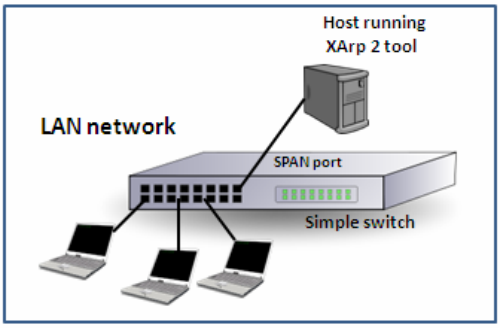
In addition, to detect some abnormal ARP packets, such as P#2, P#6 and P#8, cross-layer ARP inspection is required. Among the tested security solutions, only Cisco switch 3560 Series, Juniper switch EX3200 Series, Snort IDS, and XArp 2 tool perform cross-layers ARP inspection.

On the other hand, security solutions should be able to remember all ARP request originating and match them to ARP replies. This can be achieved by using ARP stateful inspection. XArp 2 tool and Sax2 IDS are the only security solutions that perform ARP stateful inspection.

Security solutions should also be able to cope with ARP request storm traffic and ARP scanning traffic. This type of traffic is used usually to keep target hosts' ARP caches corrupted or produce DoS attack. Sax2 IDS is the only security solution that is able to detect ARP request storm and ARP scanning.

According to the conducted experimental results, XArp 2 tool is the most efficient available security solutions to cope with ARP spoofing. However, it needs minor improvement, compared to the other tested security solutions, by adding mechanisms to detect ARP request storm and ARP scanning. Figure 6 shows a LAN network that uses a simple switch without any security features and a host running XArp 2 tool to detect ARP spoofing attack. The host running XArp 2 tool is connected to a SPAN port (mirroring port) in order to be able to receive and analyse all the LAN network traffic. This network architecture is considered ideal in terms of its low cost and its efficiency regarding the detection of ARP spoofing. However, this network architecture cannot prevent ARP spoofing, unless the simple switch is replaced by a more costly switch that integrates advanced security features. Cisco switch 3560 series (2009) and Juniper switch EX3200 series (2009) are examples of highly cost switches that can prevent ARP spoofing using a feature called dynamic ARP inspection (DAI).

Figure 6 A LAN network with simple switch and XArp 2 tool (see online version for colours)



7 Optimal ARP spoofing detection algorithm

Based on the experiments results, our work concludes that any security system claiming to cope with ARP spoofing, should use an efficient algorithm. We compiled the following six requirements that any security analyst should follow in order to get an ideal algorithm that deals with ARP spoofing on switched LANs:

1 perform a cross-layer ARP inspection between the Ethernet and ARP layers

2 perform ARP stateful inspection

3 detect non-expected IP and MAC addresses

4 detect ARP storm

5 detect ARP scanning

6 build manually (in case of non-DHCP environment) or automatically (in case of DHCP environment) IP-MAC mapping table, in order to be able to detect invalid IP-MAC pairs.

Algorithms 1 and 2 have been developed in order to detect ARP Spoofing. In Algorithms 1 and 2 some preprocessing is needed to extract the contents of the ARP header and the Ethernet header. A constant number of steps is needed to extract this information (O(1)) since the number of fields in both headers is a constant. In Algorithm 1, line 4 requires searching the IP_to_MAC Mapping Table in order to check whether (ARP IP Source, ARP MAC Source) is found in the table or not. This step requires a sequential search of the table. Since the table has size n, the time complexity of this step is O(n). Lines 7 and 10 of the algorithm require one comparison each (O(1)). Lines 13, 15, 17, 19, and 21 require 4, 3, 4, 3, and 3 comparisons respectively; 17 comparisons in total (O(1)). Therefore, the overall time complexity of the algorithm is O(n). Usually, n is equal to the number of ports in the switch (12, 24, 48, or more). In all cases, n is bounded by a constant leading to an overall complexity of O(1) for Algorithm 1. Using similar arguments, the time complexity of Algorithm 2 is also O(1). This fact indicates that very little overhead is added to the communication when our method is used.

8 Conclusions

In this study, we conducted extensive experiments to identify which security solutions are able to detect a very dangerous MAC layer attack called ARP spoofing. It is to be noted that ARP spoofing constitutes the beginning of many attacks, one of which is, the destructive MiM attack. We were able to show throw testing and experimentation that the current security solutions have many shortcomings and defects when it comes to detecting ARP spoofing. XArp 2 tool was the most efficient available security solution that can cope with ARP spoofing attacks. However, it needs minor improvement, compared to the other security solutions, by adding mechanisms to detect ARP request storm and ARP scanning. As a conclusion of our study, we suggested six basic and crucial requirements that any algorithm should follow in order to detect efficiently ARP spoofing on switched LAN networks.

Algorithm 1 Abnormal ARP packet detection when ARP operation is '*request*'

```
    Data: Ethernet header, ARP header, IP_to_MAC Mapping Table
    Result: Abnormal ARP Packets P#1, P#2, P#3, P#4
1   begin
2     Let the size of the IP_to_MAC Mapping Table = n
3     if ARP Operation = request then
4       if ((ARP IP Source, ARP MAC Source) ∉ IP_to_MAC Mapping
        Table ∀ 0 ≤ i < n) then
5         Abnormal ARP Packet (P#1)
6       else
7         if (Ethernet MAC Source ≠ ARP MAC Source) then
8           Abnormal ARP Packet (P#2)
9         else
10          if (Ethernet MAC Destination = Unicast)
11            Abnormal ARP Packet (P#3)
12          else
13            if (ARP IP Source =
              [0.0.0.0 || 255.255.255.255 || Multicast ||
              (∉ network subnet) ]
              ) then
14              Abnormal ARP Packet (P#4)
15            if (ARP MAC Source =
              [00 – 00 – 00 – 00 – 00 – 00 || ff – ff – ff –
              ff – ff – ff || Multicast]
              ) then
16              Abnormal ARP Packet (P#4)
17            if (ARP IP Destination =
              [0.0.0.0 || 255.255.255.255 || Multicast ||
              (∉ network subnet) ]
              ) then
18              Abnormal ARP Packet (P#4)
19            if (Ethernet MAC Source =
              [00 – 00 – 00 – 00 – 00 – 00 || ff – ff – ff –
              ff – ff – ff || Multicast]
              ) then
20              Abnormal ARP Packet (P#4)
21            if (Ethernet MAC Destination =
              [00 – 00 – 00 – 00 – 00 – 00 || Unicast ||
              Multicast])
              then
22              Abnormal ARP Packet (P#4)
23  end
```

Algorithm 2 Abnormal ARP packet detection when ARP operation is '*reply*'

Data: Ethernet header, ARP header, *IP_to_MAC* Mapping Table

Result: Abnormal ARP Packets P#5, P#6, P#7, P#8, P#9, P#10

```
1  begin
2    Let the size of the IP_to_MAC Mapping Table = n
3    if ARP Operation = request then
4      if ((ARP IP Source, ARP MAC Source) ∉ IP_to_MAC Mapping
       Table ∀ 0 ≤ i < n) then
5        Abnormal ARP Packet (P#5)
6      if ((ARP IP Destination, ARP MAC Destination) ∉ IP_to_MAC
       Mapping Table ∀ 0 ≤ i < n) then
7        Abnormal ARP Packet (P#7)
8      else
9        if (Ethernet MAC Source ≠ ARP MAC Source) then
10         Abnormal ARP Packet (P#6)
11       if (Ethernet MAC Destination ≠ ARP MAC Destination)
         then
12         Abnormal ARP Packet (P#8)
13       else
14         if (Ethernet MAC Destination = ff-ff-ff-ff-ff-ff) then
15           Abnormal ARP Packet (P#9)
16         else
17           if (ARP IP Source =
             [0.0.0.0 || 255.255.255.255 || Multicast ||
             (∉ network subnet) ]
             ) then
18             Abnormal ARP Packet (P#10)
19           if (ARP MAC Source =)
             [00 – 00 – 00 – 00 – 00 – 00 || ff – ff – ff –
             ff – ff – ff || Multicast]
             ) then
20             Abnormal ARP Packet (P#10)
21           if (ARP IP Destination =
             [0.0.0.0 || 255.255.255.255 || Multicast ||
             (∉ network subnet) ]
             ) then
22             Abnormal ARP Packet (P#10)
23           if (ARP MAC Destination =
             [00 – 00 – 00 – 00 – 00 – 00 || ff – ff – ff –
             ff – ff – ff || Multicast]
             ) then
24             Abnormal ARP Packet (P#10)
```

Algorithm 2 Abnormal ARP packet detection when ARP operation is '*reply*' (continued)

```
25           if (Ethernet MAC Source =
             [00 – 00 – 00 – 00 – 00 – 00 || ff – ff – ff –
             ff – ff – ff || Multicast]
             ) then
26             Abnormal ARP Packet (P#10)
27           if (Ethernet MAC Destination =
             [00 – 00 – 00 – 00 – 00 – 00 || ff – ff – ff –
             ff – ff – ff || Multicast]
             ) then
28             Abnormal ARP Packet (P#10)
29 end
```

References

ARP Spoof Tool (2009) available at http://www.imfirewall.com/en/arp-spoof.htm.

Cain and Abel (2009) available at http://www.oxid.it/cain.html.

Cisco Catalyst 3560 Series Switches (2009) available at http://www.cisco.com.

Abad, C. and Bonilla, R. (2007) 'An analysis on the schemes for detecting and preventing ARP cache poisoning attacks', *Proceedings of the 27th International Conference on Distributed Computing Systems Workshops (ICDCSW'07)*, 22–29 June.

Plummer, D. (1982) *An Ethernet Address Resolution Protocol*, RFC 826.

Frameip Packet Generator (2009) available at http://www.frameip.com.

Juniper Switches EX3200 Series (2009) available at http://www.juniper.net.

SwitchSniffer (2009) available at http://www.nextsecurity.net/software/SwitchSniffer.html.

Trabelsi, Z. and Shuaib, K. (2008) 'A novel man-in-the-middle intrusion detection scheme for switched LANs', *The International Journal of Computers and Application*, ACTA Press, Vol. 3, No. 3.

Trabelsi, Z. and El-Hajj, W. (2009) 'ARP spoofing: a comparative study for education purposes', *Information Security Curriculum Development Conference 2009, InfoSecCD09*, 25–26 September, Kennesaw State University, Kennesaw GA, USA.

Winarp (2009) available at http://www.arp-sk.org.

WinArpSpoof (2009) available at http://www.nextsecurity.net/software/Windows_ARP_Spoofer.html.

WinArpAttacker (2006) available at http://www.xfocus.net/tools/200606/WinArpAttacker3.50.rar.

Notes

1 *Unified threat management (UTM):* is used to describe a security device that has many features in one box, including a firewall, an intrusion detection (or prevention) system (IDS or IPS), e-mail spam filtering, anti-virus capability, and World Wide Web content filtering.