Volume 4, Issue 4, April 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Design and Development of an Enhanced Address Resolution Protocol to Overcome Security Threats

Santanu Kr. Sen, Debraj Roy, Sinthia Roy, Shirsankar Basu, Poojarini Mitra

CSE, GNIT, India

Abstract— Address Resolution Protocol (ARP) is a very simple networking roll call protocol that is used for dynamic address resolution between network layer address and its corresponding data link layer address. Unfortunately, the simplicity of ARP protocol leads to major insecurity like ARP Cache Poisoning, Man-in-the-Middle, Women-in-the-Middle and most vulnerable Denial of Service attack. We have, in this paper, proposed a modified ARP (MARP) that avoids the above mentioned low-level attacks and also reduces the total number of ARP transactions required to form individual ARP table for individual machine in a LAN. The MARP requires the ARP Reply to be broadcast instead of point to point transmission. This particular feature makes ARP secured and also reduces the number of ARP transactions. Two special ARP message types viz., ARP Withdrawal and ARP Popularization have been incorporated to make the ARP operation more intelligent and realistic. Some enhancement is done in ARP cache replacement policy by inclusion of check-update, binary search and least recently used algorithm. The study of MARP has demonstrated better bandwidth utilization, reduced ARP transactions and increased performance along with removal of ARP vulnerabilities.

Keywords— ARP, MARP, NLA, DLL, PNL, AREQ, ARPLY, LRU, Lucky Node

I. INTRODUCTION

Address Resolution Protocol (ARP) that resides in the network layer interface and hence a part of physical network system rather than the IP layer, is a popular protocol for dynamic address resolution between Network Layer Address (NLA) and its corresponding Data Link Layer Address (DLLA) of a machine in a local area network (LAN) like Ethernet LAN or FDDI LAN that possess large, fixed-sized physical address for each individual machine, in the same physical network, where direct mapping between NLA and DLLA addresses is not convenient [2].

It is known fact that two machines on a given Physical Network Link (PNL) can communicate with each other only if they are aware of each other's physical network addresses or Media Access Control Layer addresses (MACLA) or DLLA.

When a node N wishes to transmit a packet to a NLA 'D' in the same PNL 'L' , N first of all, transmit an ARP Request (AREQ) to the DLL broadcast address, with data indicating that N seeks D's DLLA. All nodes on L are bothered with the packet since the AREQ is transmitted to the broadcast address and all but D throws it away after examining that data and discovering it is not their NLA that is being queried. If D receives the AREQ and sees D inside, D responds with an ARP Reply (ARPLY), transmitted to the DLL source address in the received ARPR. Now, when N, the originator of the AREQ receives the ARPLY, it then stores the IP-MAC mapping or NLA-DLLA mapping in a cache, called ARP Cache, so that the IP layer (NL) of N can now construct the full Ethernet frame with the proper source and destination MACA or DLLA and also to reduce communication costs for further communication with the same destination in near future. Thus, broadly there are two types of ARP messages viz. ARP Request message and ARP Reply message.

i) **ARP Request message:** This is a message sent to the DLL broadcast address indicating a NLA for which the transmitter seeks a corresponding DLLA. The transmitter itself includes its own IP-MAC mapping in the ARP Source header during AREQ and broadcasts the request to all. The optimization/refinement obtained from such inclusion of self IP-MAC address mapping by the transmitting machine while requesting for IP-MAC mapping for a destination machine and the Broadcast nature of the AREQ is that during its transmission, all machines including the destination machine on the Ethernet LAN can enter the IP-MAC mapping of the transmitter into their respective ARP cache in advance to reduce the no. of ARP transactions to construct their respective ARP Cache [3]

ii) **ARP Reply message:** This is a simple unicast message transmitted from the destination machine to the enquirer machine in response to the enquirer's ARP Request. It contains the desired DLLA [7]

II. SECURITY THREATS ON ARP

The trick with ARP is that it is not encapsulated within an IP packet, instead, ARP has its own packet format. The Ethernet layer sends AREQ and replies directly to the ARP service without going through IP. This means that ARP is always running, awaiting for possible requests or replies. This very nature of ARP creates some serious security threats or hazards like ARP Cache Poisoning, Man in the Middle, Women in the Middle and Denial of Service attack which enables local hackers to cause general networking mayhem.

The founders of networking probably simplified the communication process for APR so that it would function efficiently. Unfortunately, this simplicity mainly leads to above mentioned major insecurity.

ARP Cache Poisoning: The largest problem with ARP is due to its stateless nature of transaction, which means that the process that listens for ARP replies is unaware of the process that sends the request. This means that if an ARP reply comes to a machine, there is no way to tell if the machine ever sent an AREQ.

Imagine a situation in which a machine M_V (Victim machine) receives a false ARPLY sent by a machine M_H (Hacker machine) that maps the DLLA of machine M_H to the NLA of the only Internet Router (M_R) connected to this LAN to communicate with the external world - all on the same PNL L. This updates the ARP Cache on machine M_V with the new incorrect IP-MAC mapping and thus poisoning its ARP Cache.

More importantly, the Router is not the only machine worth for such impersonation. A smart Hacker might choose to impersonate a File Server or a Database Server as well. With some knowledgeable customization, the hacker could intercept database files or queries. A more complex approach might involve two deceptions- the database user’s machine would think that the Hacker’s machine was the database. Likewise, the database itself would think that the Hacker’s machine was the user’s machine. Thus, there is a lot of power in such type of low-level attack. However, ARP Cache does expires after a short period of time, but a simple piece of software can constantly send the ARP Reply, thus ensuring that the cache entry stays valid.

Some implementations of ARP attempt to update their cache by sending out AREQs to each entry in the cache and thus creating a problem for the Hacker since the machine with the actual IP address will respond with its actual hardware address and in that case if attacker tries to impersonate the requester, he will be caught red handed. But, once again, the stateless nature of ARP comes to the rescue of the Hacker. Sending replies to the requests before the requests are sent, that is, sending reply to a machine before giving it any chance of putting any request, will actually prevent the requests from being sent in the first place. Thus, an attacker can successfully have one machine impersonation another with relatively little effort due to the flaws existing in the present ARP technique.

Man in the Middle Attack: A hacker can exploit ARP Cache Poisoning to intercept network traffic between two machines in the same PNL L. For instance, let’s say hacker wants to see all the traffic between M_V and M_R. The hacker begins by sending a malicious ARPLY (for which there was no previous request) to M_R, associating the MAC address of M_H with M_V. Now M_R thinks that the hacker's computer is the Victim’s computer. Next, the M_H sends a malicious ARPLY to M_V, associating MAC address of M_H with M_R. Now, M_V thinks that the hacker’s computer is the required Internet Router. Finally, the hacker turns on an operating system feature called IP Forwarding. This feature enables M_H to forward any network traffic it receives from M_V to M_R. Now, whenever M_V tries to go to the Internet, it actually sends the network traffic to the hacker’s machine, which is then forwards to the real router M_R. Since, the hacker is still forwarding the network traffic of M_V to M_R, M_V remains unaware that the hacker is intercepting all its traffic and perhaps also sniffing the clear text passwords or hijacking its (M_V’s) secures Internet session.

Denial of Service Attack: Another vulnerable low-level attack on ARP is Denial of Service (DoS) attack. A little effort could easily warp the ARP Caches on the entire LAN, rendering the entire LAN worthless for quite a good while.

Here, a hacker can easily associate an operationally significant IP address to a false MAC address. For instance, a hacker can send an ARPLY associating the Router’s IP address with a MAC address that does not exists at all. In this case, a Victim’s computer believes that he knows where his default gateway is, but in reality, he is sending any packet whose destination is not on the local segment, into the Great Bit Bucket in the sky. In one move, the hacker has cut off the Victim’s network from the Internet. Thus, ARP gives a creative Hacker a wonderful palette of vibrant colors with which to redecorate a network.

III. **ARP REQUEST & REPLY- BOTH BROADCAST**

A machine uses ARP to find the hardware address of another machine by broadcasting an ARP Request i.e., the ARP Request procedure is a broadcast procedure whereas the ARP reply procedure is a unicast procedure that is directed to one particular machine, truly speaking, to the requester. Thus, ARP replies are not broadcast.

The proposed modified ARP scheme suggests to covert the ARP Reply from Unicast nature to Broadcast nature. That leads to convert both ARP Request and Reply to become broadcast transmission. The said modification (to the existing ARP Reply) has two major advantages. The first advantage is to increase the performance of existing ARP protocol by reducing the number of ARP transactions during ARP Cache construction by each individual machine on the same LAN. The second advantage leads to overcome the low-level ARP security threats like ARP Cache Poisoning, Man in the Middle, Women in the Middle and Denial of Service attacks that occurred by fake advertisement by an attacker taking the advantage of point-to-point communication during ARP Reply and the inherent stateless nature of ARP service

It is to be noted that poisoning the ARP Cache of a particular machine during ARP Request is not possible since the AREQ is broadcast. Therefore, if an attacker tries to impersonate a machine M_V in the same PNL L, supplying a fake IP-MAC mapping, it is simply not possible because the actual machine whose MAC address the attacker wants to impersonate, will also see the AREQ and if it finds that its own MAC address is being broadcast wrongly with another IP address by another machine, the machine M_R will immediately raise its objection and will complain to the system manager about the conflict. Keeping this advantage of broadcast message in mind, we have thought of making the ARP Reply also broadcast so that everything should be open in nature - nothing hidden, nothing secret. Therefore, if an attacker with or without getting any ARP request, sends an ARP Reply with wrong IP-MAC mapping, it will be brought to the notice of all machines immediately and the machine M_R will definitely raise its objection and lodge a complain of address conflict to the system or network manager. Thus, ARP Cache Poisoning will not be possible.

However, at a first glance, it seems that if ARPLY is made broadcast, there could be a bandwidth overhead, but the study shows it is not so. Once ARP Caches are built up, no further BW is required and since the ARP Cache timers operate in terms of minute(s) and not in terms of millisecond or second, bandwidth (BW) is not at all a problem. Results are shown in Section V.

On the other hand, those who thinks that converting the transmitting property of ARP Reply from point-to-point style to Broadcast style, it may create a security vulnerability as because the MAC address of a machine is getting known to all during ARPLY, it is to be noted that the intrinsic meaning of a true network is nothing but to make contacts with each other in more effective and simplified way. To be added, making ARPLY broadcast, the security threats are removed instead.

IV. ARP Cache Searching and Replacement Policy

Conceptually, ARP software can be divided into three modules viz.

1) Output Module (OM)
2) Input Module (IM) and
3) Cache Manager (CM)

To reduce communication costs, computers that use ARP maintain a cache of recently acquired IP-MAC address bindings [Section 1].

The CM, the third module of the ARP software usually deals with ARP Cache Replacement Policy. It examines entries in the cache using *Sequential searching* technique and removes them when they reach a specified age decided by the TTL counter [2]. In addition, whenever an ARP request or Reply comes, the Input Module (IM) uses arpfind subroutine to locate the entry in the existing cache and if found, updates the entry in the cache by *directly overwriting* the corresponding entry else it creates a new entry for the incomer. The arpalloc subroutine of CM decides which entry to eliminate from a full cache while finding a space for a new entry. Usually, arpalloc deletes entries in a Round Robin fashion with a global pointer [2]

The proposed paper suggests the following modifications:

Binary Search: Use of Binary Search (BS) instead of Sequential Search reduces the searching time. Although, the use of BS technique requires the ARP Cache to be sorted, which adds some extra complexity; but the this computational overhead cannot be a criterion since the TTL timer works in terms of minutes and not in terms of millisecond or second. On the other hand, for Class B or Class A network or even private Class C network (192.168.0.0), where the number of nodes in the same PNL could be huge compared to public/real Class C network, BS would yield an efficient result.

Check and Update: When an entry in the form of AREQ or ARPLY arrives to a machine M_V, the arpfind subroutine is used to check whether the entry already exists in the current ARP Cache or not.

If found then instead of directly overwriting the cache (as it happens in present scheme of ARP), the arpalloc subroutine checks if there is any change in the IP-MAC mapping and no overwrite or updation is done if no change is reflected. The same entry is overwritten if any change is found. However, in both the cases, the TTL timer is reset to mark the entry as the latest one.

Auto Timeout: When the TTL timer of an entry reaches to zero, its terminating value, the entry is deleted from the ARP Cache since the lifespan of the entry has expired. No further request will be sent to the deleted machine to minimize the BW overhead. However, a fresh AREQ is required for further communication with deleted machine. This scheme already exists in CM.

LRU: When the ARP Cache is full and a new entry (AREQ/ARPLY) arrives, the CM needs to allocate space in the ARP Cache for the newcomer if its entry is not currently present in the ARP Cache. In such situation, an existing entry needs to be deleted from the ARP Cache to make room for the new. The removal can be done using the LRU algorithm based on TTL timer. An entry with the lowest TTL value is an entry which is the oldest one indicating its least usage. Therefore, this LRU entry could be removed to provide space to the newcomer. However, if more than one entries are found having same TTL value, FCFS algorithm will be applied as the sub criterion under the LRU algorithm.

V. ARP Refinements: ARP Withdrawal and Popularization

Operationally, ARP message is of two types viz. ARP Request (OPERATION filed value=1) and ARP Reply (OPERATION filed value=2).

However, two more optimization is possible by inclusion of two more ARP message types viz.

1) ARP Withdrawal
2) ARP Popularization

The above two optimization or refinements are obviously possible since the "OPERATION" field of the ARP message format is 16-bit whose only a few bits are practically in use for ARP and RARP purpose. So, easily two more bit sequence can be used to include the above-mentioned two additional ARP message types.

1) ARP Withdrawal message: When a machine goes for shut down or when a machine wants to be disconnected from the network it is currently on or when a machine wants to withdraw itself for any other reason, the machine will broadcast an ARP Withdrawal message setting the OPERATION FIELD to a particular value, say X in this case, where X is a positive integer number and not used in present ARP message format. This message is broadcast to notify everybody in the LAN that it no more available. Receiving this message, all machines, except the owner of the message, will immediately delete the entry from their respective ARP Cache. None will give any reply to this message.

2) ARP Popularization message: In the present case, when a machine boots, it broadcasts its own IP-MAC mapping in the form of an ARP request. There should not be a response. Strictly speaking, the machine, which sends this AREQ, will reply to itself. However, the side effect of this broadcast is to make an entry in everyone's ARP Cache. If a response does arrive, two machines must have been assigned the same IP address. The new one should inform the system manager and not boot [3].

The proposed scheme suggests that instead of doing such Self ARP Request, a new type of ARP broadcast message, called ARP Popularization, could be incorporated by taking the advantage of huge no. of unused bit sequence in the OPERATION FIELD of the ARP message format. This new type of ARP message type is necessary because, since in this proposed scheme, the ARP Reply message is broadcast, therefore, doing Self ARP Request may create unwanted traffic in the network thrown by the Requester. However, the scheme can be made more intelligent adapting the concept that a machine doing Self ARP Request will never reply in response to the request made by itself.

This popularization message used by a machine when booted or rebooted, will help the machine to popularize its presence (newly) in the network and everybody in the network will extract the IP-MAC mapping and cache the entry in their respective ARP cache without giving any reply. Impersonation in this case is not possible due to the broadcast nature of the popularization message. The second advantage of this type of message is to gain latest knowledge about the status of a newcomer, which was not possible in earlier scheme.

VI. RESULTS

The experiment is done considering the present ARP scheme and the proposed Modified ARP (MARP) scheme. Following is an example of an bus based Ethernet LAN that consists of 4 nodes N1,…..,N4. The study considers both the Best Case and the Worst Case phenomena for both the present and the proposed modified schemes of ARP for comparison.

However, a general study considering 'n' no. of nodes is henceforth.

PRESENT ARP SCHEME

ARP WORST CASE:

TABLE I
ARP REQUEST/REPLY TABLE:

ARP Tr. No.	AREQ/ ARPLY	Source	Destination	Broadcast/ Unicast (B/U)	Popularization	Redundant (R)	Remarks
1	AREQ	N1	N2	B	N1 is known to ALL	-	Total No. of CARPTr.=6 **N4 is "Lucky Node"**
	ARPLY	N2	N1	U	N2 is know to N1 Only	-	
2	AREQ	N1	N3	B	N1 is known to ALL	R	
	ARPLY	N3	N1	U	N3 is know to N1 Only	-	
3	AREQ	N1	N4	B	N1 is known to ALL	R	
	ARPLY	N4	N1	U	N4 is know to N1 Only	-	
4	AREQ	N2	N3	B	N2 is known to ALL	-	
	ARPLY	N3	N2	U	N3 is know to N2 Only	-	
5	AREQ	N2	N4	B	N2 is known to ALL	R	
	ARPLY	N4	N2	U	N4 is know to N2 Only	-	
6	AREQ	N3	N4	B	N3 is known to ALL	-	
	ARPLY	N4	N3	U	N4 is know to N2 Only	-	

MODIFIED ARP SCHEME

MARP WORST CASE:

TABLE II
ARP REQUEST/REPLY TABLE:

ARP Tr.No.	**AREQ/ ARPLY**	**Source**	**Destination**	**Broadcast/ Unicast (B/U)**	**Popularization**	**Redundant (R)**	**Remarks**
1	AREQ	N1	N2	B	N1 is known to ALL	-	Total No. of CARPTr.=3 **N2,N3 and N4 are "Lucky Node"**
	ARPLY	N2	N1	B	N2 is know to ALL	-	
2	AREQ	N1	N3	B	N1 is known to ALL	R	
	ARPLY	N3	N1	B	N3 is know to ALL	-	
3	AREQ	N1	N4	B	N1 is known to ALL	R	
	ARPLY	N4	N1	B	N4 is know to ALL	-	

PRESENT ARP SCHEME

ARP BEST CASE:

TABLE III
ARP REQUEST/REPLY TABLE:

ARP Tr.No.	**AREQ/ ARPLY**	**Source**	**Destination**	**Broadcast/ Unicast (B/U)**	**Popularization**	**Redundant (R)**	**ARP Cache formation**
1	AREQ	N1	N4	B	N1 is known to ALL	-	Total No. of CARPTr.=3 **N4 is "Lucky Node"**
	ARPLY	N4	N1	U	N4 is know to N1 Only	-	
2	AREQ	N2	N4	B	N2 is known to ALL	-	
	ARPLY	N4	N2	U	N4 is know to N2 Only	-	
3	AREQ	N3	N4	B	N3 is known to ALL	-	
	ARPLY	N4	N3	U	N4 is know to N3 Only	-	

MODIFIED ARP SCHEME

MARP BEST CASE:

TABLE IV
ARP REQUEST/REPLY TABLE:

ARP Tr.No.	**AREQ/ ARPLY**	**Source**	**Destination**	**Broadcast/ Unicast (B/U)**	**Popularization**	**Redundant (R)**	**ARP Cache formation**
1	AREQ	N1	N4	B	N1 is known to ALL	-	Total No. of CARPTr.=2 **N3 and N4 are "Lucky Node"**
	ARPLY	N4	N1	B	N4 is know to ALL	-	
2	AREQ	N2	N3	B	N2 is known to ALL	-	
	ARPLY	N3	N2	B	N3 is know to ALL	-	

COMPARATIVE STUDY RESULT

TABLE V

Sl.No.	**Index**	**ARP**		**MARP**	
		Worst Case	**Best Case**	**Worst Case**	**Best Case**
1	No. of Complete ARP Transaction	N*(N-1)/2	N-1	N-1	Integer{(N+1)/2}
2	Total no. of Broadcast messages (AREQ/ARPLY)	N*(N-1)/2	**N-1**	N	N
3	Redundant Broadcast messages(AREQ/ARPLY)	N*(N-1)/2 -1	0	N-1	0
4	Total no. of AREQ	N*(N-1)/2	N-1	N-1	Integer{(N+1)/2}
5	Total no. of ARPLY	N*(N-1)/2	N-1	N-1	Integer{(N+1)/2}
6	Total no. of Unicast messages	N*(N-1)/2	N-1	0	0

The above study shows that the no. of Complete ARPTr (CARPTr). in the best case of present scheme is equal to that of the worst case of the proposed modified scheme. Also, the above tables clearly indicate that making ARP Reply broadcast; we are not only making our network more secured from ARP vulnerabilities but also gaining better performance.

VII. CONCLUSION

The proposed MARP has several advantages. Several modifications to the existing ARP system are suggested to increase performance and to make ARP free from low-level security threat like ARP cache poisoning, Man in the Middle, Women in the Middle and Denial of Service attacks. However, the somewhat increased processing and complexity and by the inclusion of binary search and ARP Withdrawal and Popularization message types in the MARP, is only a small price to be paid when viewed against the backdrop of the vast performance improvement and security enhancement. The comparative study between ARP and MARP shows that there is also a BW gain in all cases only except in one case- row no. 2 in Table V above.

REFERENCES

[1] Douglas E. Comer, *Internetworking with TCP/IP Vol.-I*, 4th Ed., Pearson Education (Singapore), LPE, 2005.

[2] Douglas E. Comer, David L. Stevens, *Internetworking with TCP/IP Vol-II*, 3rd Ed., Pearson Education (Singapore), LPE, 2005.

[3] A. S. Tanenbaum, *Computer Networks*, 3rd Ed., PHI, 2000.

[4] Albeto Leon-Garcia and Indra Widjaja, *Communication Networks*, Tata McGraw Hill, 2000.

[5] Wes Sonnenreich and Tom Yates, *Building Linux and OpenBSD Firewalls*, Wiley Computer Publishing (USA).

[6] M Beck, H Bome, M Dziadzka, U Kunitz, R Magnus, and D Verworner, *Linux Kernel Internals*, 2nd Ed., Pearson Education Asia, LPE, 2001.

[7] Radia Perlman, *Interconnections: Bridges and Routers*, Addison Wesley, 1994.