A Proposed Technique to Prevent ARP spoofing in Local Area Networks

Mohamad Fakher Karzon
Postgraduate Student (MSc)
Telecommunications Engineering Department
Aleppo University, Syria

Mohammad Samir Modabbes
Professor
Telecommunications Engineering Department
Aleppo University, Syria

ABSTRACT
Address Resolution Protocol (ARP) is a network protocol used to identify MAC address through mapping IP. When exchanging information between two computers on a Local Area Network LAN, the source computer must obtain the physical address (MAC) that is mapped to the logical address (IP) of the destination computer, where this process is the basic role of Address Resolution Protocol (ARP) as defined in (RFC826) by Internet Engineering Task Force (IETF). But the nature of this protocol makes it vulnerable to some threats that allow any host on the LAN to have access to unauthorized information or cause the service to stop.

In this paper, we present a technique that prevents ARP spoofing attack using a computer application (Server - Client). The application utilizes the static feature of the ARP protocol to build a trusted ARP table that contains correct IP-MAC mapping for each computer on the local area network subnet, then the trusted ARP table is sent to all computers on the subnet. To evaluate the performance of proposed technique, a simulation was implemented using GNS and Virtual Box, where the results showed the efficiency of the proposed technique in terms of attack prevention, and in terms of time, cost and scalability compared to other related studies

General Terms
Network Security Application, Information Security

Keywords
Network security, MITM, DoS, ARP Spoofing, Local Area Network

1. INTRODUCTION
Security is the major concern for many companies in today's world. However, one area that is usually left untouched is hardening Data link layer and this can open the network to a variety of attacks and admittances [1].

Networking is the primary need for today's communication channel management and business. The growth of networks from ARPANET to today's internet is a makeable journey and a changeover from the idealization of the network to practical practices [2]. The technical improvisation of networking infrastructure with respect to time has also led to the misuse and wrong interpretation of the networking protocol for ill-beneficial reasons such as hacking, spoofing etc. Thus, the trivial networking infrastructure needs to be protected from malicious attackers at any given point of time [2].

In Ethernet, when both sides of communication are sending messages, they will need not only the network logical address (IP) but also the network physical address (MAC). So it presents an issue how to get the MAC address based on the IP address. ARP protocol is used for resolving this issue [3]. The ARP table will keep the reflection between the IP and MAC address and is updated unceasingly. The network attacker uses several spoofing methods to attack the network by the disadvantage of the ARP protocol (stateless nature) which is seriously threatening the network security [4].

This paper introduces an effective technique to prevent one of the ARP attacks which is ARP Spoofing. It also presents a comparison with other related studies. The basic idea of the presented architecture is to design and install a client/server software on each computer in the LAN subnet where we assume installing the first part (server) on one trusted computer and the second part (client) allocated to the hosts, whom we plan to protect. However, the proposed technique is characterized with flexibility, since we could install any part (client/server) on any host. This solution does not need another software or hardware, therefore, it is backward compatible to communicate with the standards of current ARP. On the other hand, this scheme is not costly and easy to implement. The rest of this paper is organized as follows: Briefly description about ARP and ARP Attacks is explained in section 2. Section 3 explores some related works. In section 4, the design of the proposed technique is presented. Simulation and results are found in section 5. Finally, Section 6 concludes the paper.

2. ADDRESS RESOLUTION PROTOCOL (ARP)
Address resolution protocol (ARP) is one of the most important protocols in the Transmission Control Protocol (TCP)/Internet protocol (IP) model and defined in [RFC826]. ARP is used to associate the IP address of the network layer to the MAC address of the data link layer, which is stored in ARP cache of each client machine [5], [6]. This protocol can identify how many hosts are connected in the LAN, and discover the MAC address associated to given IP address in a LAN subnet [7].

2.1 How ARP works
ARP is done by sending message request called broadcast, to find out the MAC address for the IP address. Subsequently, each device in the network receives the message and compares it with its IP address. If there is a match between IP's, then the generated ARP reply is called unicast. When other devices are identified which do not match the IP's, the packet will be dropped. After that, the IP-MAC addresses mapping is saved in the ARP cache table [8].

When a device wants to communicate with another device on the same LAN subnet, the sending device sends an ARP request message containing the IP address of the receiving device. All devices on a local network segment see the message, but only the device that has that IP address responds with the ARP reply message containing its MAC address. The

sending device now has enough information to send the packet to the receiving device. An example of ARP Request and ARP Reply is as follows: [9]

1. Machine A wants to send a packet to D, but A only knows the IP address of D.

2. Machine A broadcasts an ARP Request with the IP address of D as shown in Figure (1).

3. All machines on the local network receive the broadcasted ARP Request.

4. Machine D replies with its MAC address by unicast of ARP Reply as shown in Figure (2) and update its ARP table cache with the MAC of A.

5. Machine A adds the MAC address of D to its ARP cache.

6. Now machine A can deliver packets directly to machine D.

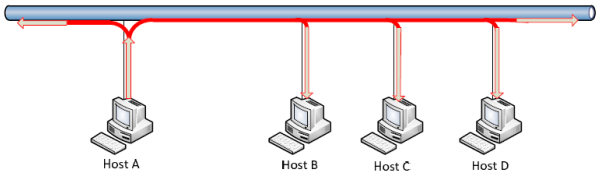

Fig. 1 Host A broadcast request for Host D

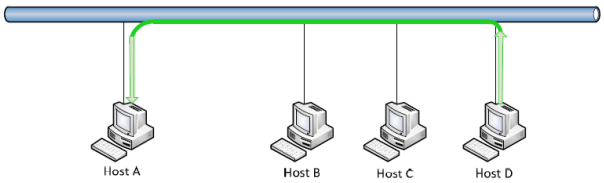

Fig. 2 Host D Replies to Host A (Unicast)

ARP was designed without security features, so ARP doesn't support the authentication or integrity scheme, and thus can be easily spoofed. Therefore, ARP is highly susceptible to spoof and poison attacks [10].

2.2 ARP attacks

While considering about network security, it ought to be underscored that the total network is secure, instead of considering only the computers at the end of the communication chain to avoid attacks in hosts [11]. An attacker will focus on the communication channel, get the information, and decode it and reinsert a copy message. While building up a protected network, the essentials should be considered are confidentiality and integrity. However, the variety of attacks is still increasing by passing time [12]. The fundamental class of attacks is categorized as active and passive attacks which include spoofing, modification, wormhole, fabrication, denial of services, sinkhole, Sybil, eavesdropping, black hole, rushing attacks, etc. ARP spoofing or ARP cache poisoning is a procedure used by attackers to spoof ARP packets in a LAN [6]. Figure (3) demonstrates the ARP spoofing or ARP cache poisoning attack [5].

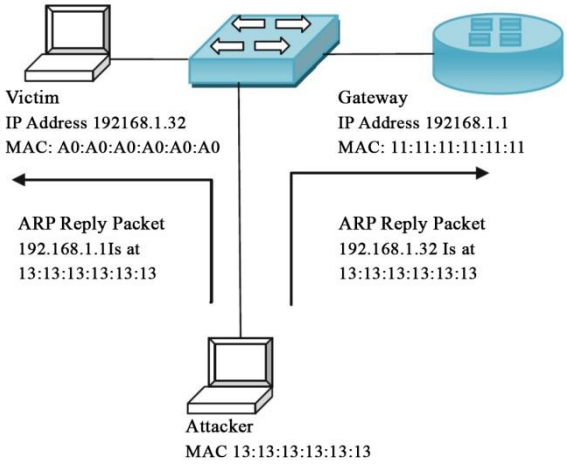

Fig. 3 Arp Spoofing/ARP Cache Attack

The fundamental norm of ARP spoofing is to misuse the absence of validation in the ARP protocols by directing spoofed ARP communications onto the LAN [13]. ARP spoofing attacks are generally started from a long used host in LAN or from an attacker system connected in a LAN without the admin knowledge. The attacker in most cases copy the MAC address of a host and utilizes it to identify itself as the host. This allows the attackers to track the packets from the LAN presenting itself as the host [13], [11]. In most scenarios, this is used as an opening for diverse attacks. The attacker assesses the packets (spying), while at the same time sending the data packets to the genuine destination to evade detection, alter information in advance; sending MITM attack, or dispatch a DOS attack by triggering a few or the majority of the packets on the system to be dropped out [11]. The attack utilized on systems that employ ARP, and are bound to require the intruder to increase guide admittance to the LAN to be criticized. There are a few methodologies to mitigate ARP spoofing like utilizing ARP entries and prevention software or securing the operating system [14], [11]. However, the current strategies fail when the attack gets stronger and hence prominent methods are sought to be developed [11].

3. RELATED WORKS

Many researches were presented in the domain of defending and preventing ARP spoofing, some of them using static ARP while others were based on dynamic ARP. According to [11] a new network security system based on routing trace that can protect the internal network against ARP spoofing attacks (RTNSS) was designed and developed as a recognition model which uses monitoring agents to detect the ARP spoofing. The installed monitoring agent observes the modifications in the ARP cache table made by the adversary. These changes are made to spoof the host address with a fake address including the IP, MAC address pair. Detection of these spoofing can be performed by the agent through a routing trace and it alerts the main server. Though this method seems to be similar to few existing detection methods, the ARP cache table changes are more accurately sensed in RTNSS and thus the attack detection is significant. Another important aspect is that, unlike existing methodologies, RTNSS incorporates the mechanisms to tackle the effects of structure changes and increased traffic changes due to encryption complexity. However, this model fails to handle packet forwarding relay based attack strategy and also the lack of encryption mechanisms increases the security vulnerability of ARP [11].

Another solution was Securing ARP and DHCP for Mitigating Link Layer Attacks, which is a security model that applies cryptography based protocols to the communications at the data link layer to improve the authentication and data integrity [20]. This model utilizes IPsec and Transport Layer Security (TLS) to offer authentication and data integrity using cryptography for communications at the network and transport layer. This model of intrusion detection can mitigate the link layer attacks including the rogue Dynamic Host Configuration Protocol (DHCP) server, DHCP exhaustion, mischievous customer, host impersonation, ARP Spoofing, MITM, and DOS attacks. Unlike existing models, this model mitigates the DHCP starvation attack using the symmetric key cryptography. However, the limitation of this model is that it fails to prevent DoS attacks using flooding and hence care should be taken [11].

According to [6] the NIDPS (Network Intrusion Detection and Prevention System) technique is suggested to have a server that collects IP-MAC mappings from users using small agents. These mappings will be then used as static ARP entries to correct any wrong mapping detected. However, agents aren't authenticated to the server. Moreover, the server examines every packet going in or out the LAN segment. Finally, it waits for the attack to occur and then try to solve it [6].

Another method is suggested to solve ARP spoofing problem using snort IDS and static ARP entries [6]. Yet, it still needs the administrator to add the static mappings manually.

The researcher in [15] proposed a hybrid technique to prevent and detect ARP Spoofing attack using a centralized ARP server ACS with the assistance of ARPWATCH software after correcting its detection system to increase detection efficiency. As for the process of preventing the attack, it is the responsibility of the central server which analyzes all ARP data exchanged over the network. Figure (4) illustrates the general structure of the proposed system. However, the architecture of this technique is complicated and time consuming.

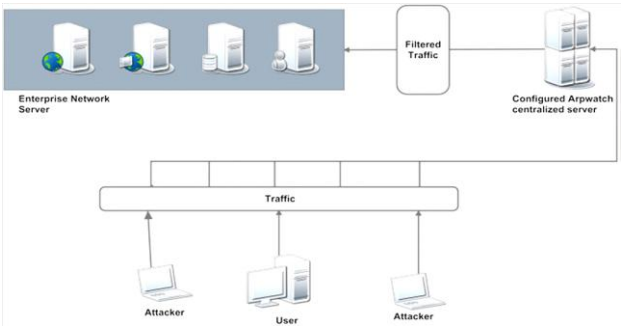

Fig.4 General architecture of ARP protection system using a centralized server

While in the research [16] a mechanism was proposed to control ARP messages through an ACL, in addition to a MAC Address Filter and a computer application that responds to ARP Request messages. Figure (5) demonstrates the flow chart of the proposed mechanism.

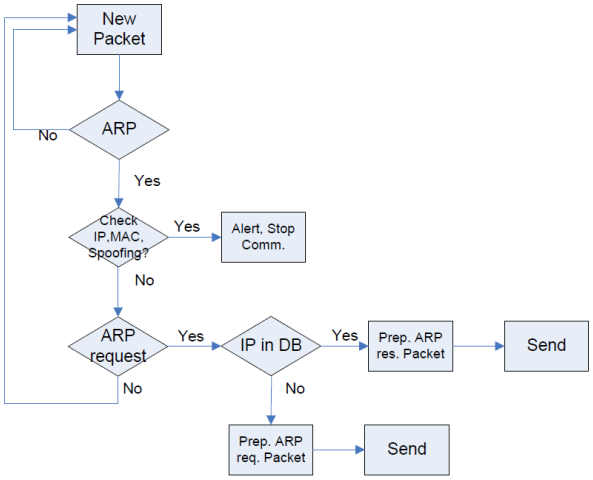

Fig.5 Flow chart for the work of the computer application

The ACL is applied to the network switch that closes the physical port connected to the attacker's computer, but this technology is also expensive as it requires Cisco network switches and it requires an amendment to the mechanism of the network adapter's interaction with exchanged data packets Across the network. In addition, <IP, MAC> pairings database keeps at most 254 entries which is the maximum limit for a C-class network.

In [17], a mechanism has been proposed to prevent ARP poisoning attack in networks that operate with a dynamic addressing system, by using a software tool based on the ICMP protocol that uses a secondary table to verify the validity of logical addresses and their associated physical addresses. Figure (6) shows the proposed architecture that uses a DHCP server that is responsible for the dynamic addressing system, the Radius server is responsible for user authentication, as well as the MySQL database server.

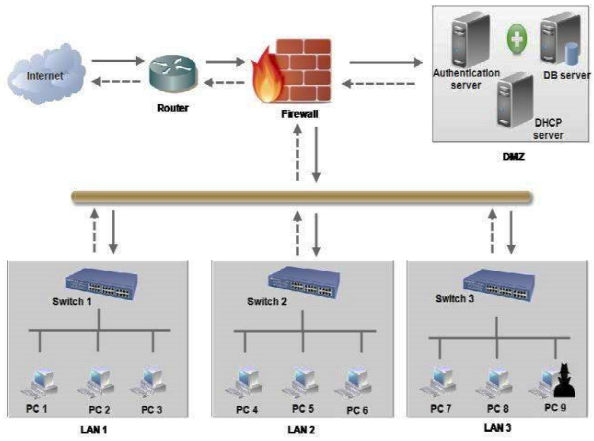

Fig.6 The proposed architecture for attack prevention in networks running a dynamic addressing system

Figure (7) shows the results of using the proposed system, as the number of ARP packets exchanged across the network decreases in both the normal state and the attack state. However, this technology is complex and expensive from a design point of view.

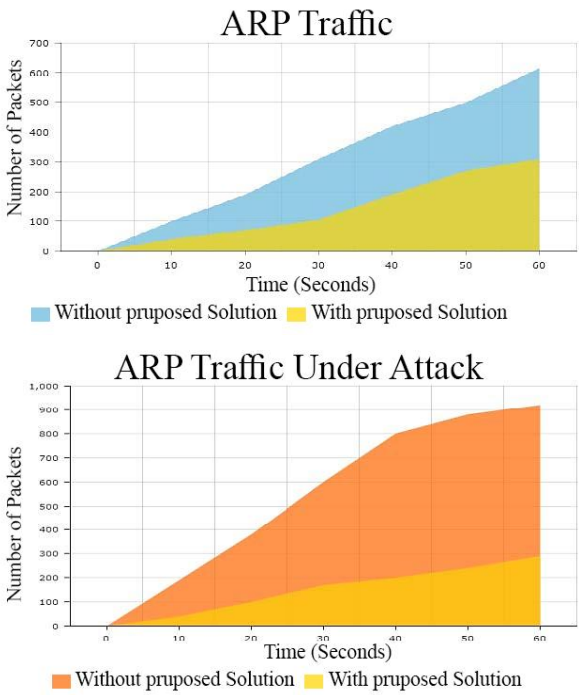

Fig. 7 The rate of ARP packets exchanged across the network in the normal state and the attack state

Another solution to ARP spoofing using A client-server protocol was introduced in [6] by automatically configuring static ARP entries. The protocol works in both static and DHCP networks. Moreover, it can work in large-scale networks without any overhead on the administrator. In addition, the technique doesn't require any special hardware to be deployed, as any host can work as an ARP server, but the disadvantages of this method is the complicated way to build the trusted ARP table using hashed messages between client and server and attempting 3 tries to figure out the correct MAC of the sending host which in turn requires more time to prevent the attack compared to the proposed technique in this paper.

Table (1) compares the proposed technique with previous techniques where the comparison criteria includes if it depends on static ARP entries, works in DHCP networks, ability to prevent ARP spoofing attacks, scalability, if it has manual or automatic configuration or implementation, uses key authentication, depends on cryptography algorithms, consumes time and if it needs special or expensive hardware.

Table .1 Comparison among different techniques using 9 comparative attributes

Technique	Static ARP	DHCP	Full Prevention of ARP spoofing	Scalability	Automation	Key Authentication	Cryptographic	Cost Effective	Time effective
Proposed	✓	✓	✓	✓	✓	✓	×	✓	✓
Client-server protocol [6]	✓	✓	✓	✓	✓	✓	✓	✓	×
NIDPS [18]	✓	✓	×	✓	✓	×	×	×	×
Snort IDS [19], [6]	✓	×	✓	✓	×	✓	✓	✓	×
S-ARP and DHCP for Mitigating LL Attacks [20]	×	✓	✓	✓	✓	✓	✓	✓	×
RTNSS [21], [11]	✓	✓	✓	✓	✓	×	×	✓	×
Centralized ARP server ACS [15]	×	✓	✓	✓	✓	✓	×	✓	×
Layer-2 MAC and protocol filtering and ARP server [16]	×	✓	✓	×	×	✓	×	✓	×
Prevention in dynamic IP configuration [17]	×	✓	✓	✓	✓	✓	×	×	×

4. THE PROPOSED TECHNIQUE

The importance of this research lies in improving security in local networks by preventing ARP Spoofing attack.

A computer application was designed to work in a (server - client) environment where the server builds a trusted ARP table, and distributes that table to all client computers on the network.

Client computers receive trusted ARP table and add static ARP entries automatically, which ensures that computers do not respond to fraudulent response messages because the nature of ARP prioritizes static entries, thus preventing ARP Spoofing attack. The proposed technique has the following advantages:

- Proficiency in working in wired and wireless local networks.
- The ability to work in static or dynamic local area networks.
- Low cost due to the absence of the need for special equipment, as the server can be any of the devices on the network. In addition, the technique is compatible with all types of switches and network routers.

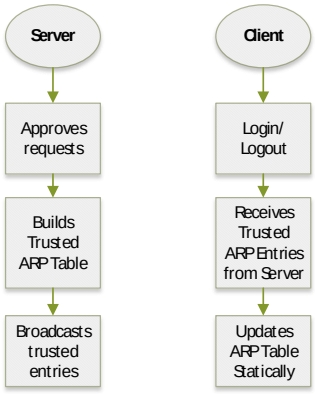

Fig. 8 Basic functions of both the server application and the client application

Figure (8) shows the basic functions of both the server application and the client application where the server is mainly responsible for the following:

1. Verification and approval of requests for logging in and logging out of the server.
2. Building the trusted ARP table.
3. Distributing trusted ARP table entries to all computers on the network.

Whereas, the client application is responsible for:

1. Submitting login and logout requests.
2. Receiving the trusted ARP entries.
3. Automatically updating ARP tables on computers using static mode.

Computer applications were designed using the C# programming language, which is one of the high-level programming languages within the .NET framework. These applications are characterized by their small size and their ability to work on computers without the need for expensive servers.

4.1 General Architecture of the Proposed Technique:

1) Server side:

Figure (9) shows the flow chart of the server application. The user database is consisting of username and password fields which will be created by the system administrator. The IP field represents the logical address of the computer that will login to the server. The MAC field represents the physical address of the computer that sends the login request to the server, where the status field represents the current state of the user which can be (idle, logged in, logged out).

The server listens to the requests coming from the client application and checks for correct username and password. If the username and password are correct, the server checks the flag field to determine the type of the request (1 for login request or 0 logout request).

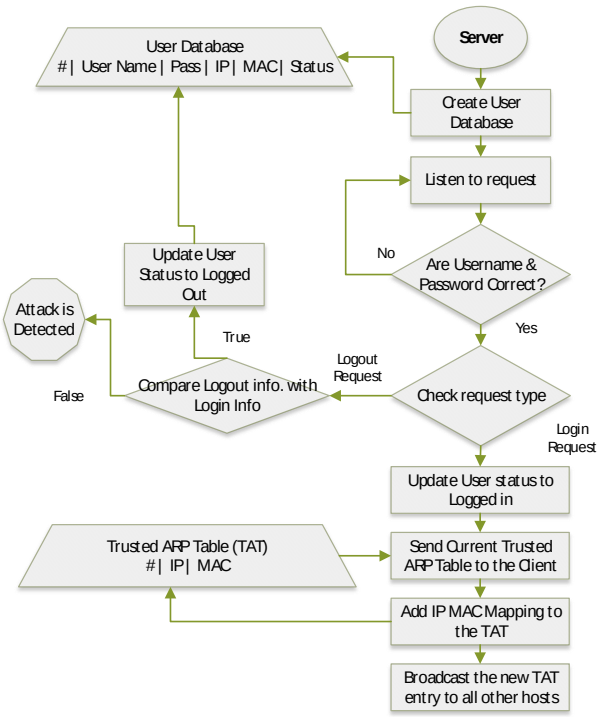

Fig.9 Flow Chart of the Server Application

Requests received by the server are handled according to the following:

Login Status:

1. The server application updates the status field of the user record in the user database to Logged in

2. The current trusted ARP table is sent to the computer that successfully logged in.

3. The IP-MAC Pair of the newly Logged-in computer is added to the trusted ARP table on the server.

4. The new Trusted ARP Table TAT is broadcasted to all other clients on the subnet, which ensures a correct static ARP Table at each of those clients.

Log-out status:

1. The data of the Logout request sender is compared with the data already exist in the User database.

2. In case of match, the user status in the users table is updated to Logged out, thus giving the opportunity to the same logical address to be granted to another user, which enhances the system's operation in the dynamic addressing environment using DHCP.

3. In case of mismatch, the request will be considered as a threat, informing the network administrator with a detected attack, as this indicates that one of the users has sent a forged registration message to the server in which it is:

Logical address = Logical address of the victim

Physical address = the attacker's physical address

2) Client side:

Figure (10) shows the flow chart of the client application where the user initially adds the logical address and the physical address of the server, given that there is one server on the network, which is a computer approved by the rest of the computers and managed by the system administrator. Later on, the application works according to the following:

1. The user logs in according to the registration data provided by the system administrator.

2. The application starts listening to updates from the server.

3. When a new update is received, the client application makes sure that the physical address of the update source is the physical address of the server. This is done by verifying that the incoming address matches the address previously registered with the application.

4. Upon ensuring the validity of the server's physical address, the update is approved and the application updates the computer's ARP table in static mode according to the entries in the update message.

5. When the protected client computer wants to logout, the Logout request will be received and processed by the server according to the steps explained in the server application flow chart.

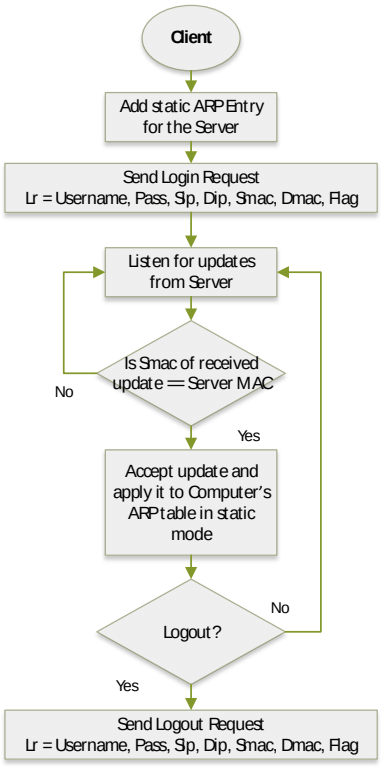

Fig.10 Flow Chart of the Client Application

5. SIMULATION AND RESULTS:

The proposed technique was designed using C # programming language according to the flow charts for both the server and client application mentioned previously. As for the simulation, it was done using GNS software and the results were analyzed using Wireshark. Table (2) shows the simulation component and specifications for each component:

Table 2. Simulation component and specifications

Components	Computers	Router	Switch
Type	Intel Core i5	Cisco C3600	Default GNS
Processor	2.50 GHz	Default GNS	Default GNS
RAM	1 GB	Default GNS	Default GNS
Operating System	Windows 7 (32 bit)	Cisco IOS 7	Default GNS

Figure (11) shows the network diagram of the simulation and its components, which consist of 4 computers, one of which is chosen as a trusted computer and works as a server, in addition to a switch and a router.

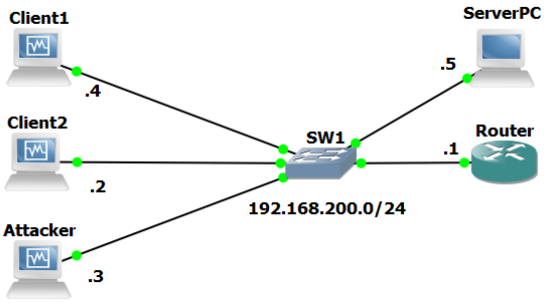

Fig.11 Network diagram of the simulation and its components

DHCP protocol was activated on the router and the computers obtained IP addresses as shown in Figure (11). The Attacker also started an ARP Posioning attack on both Client1 and Client2 computers using Cain & Able software, as shown in Figure (12).

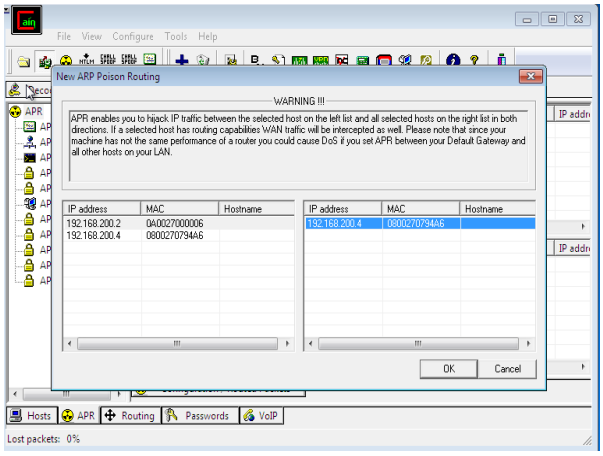

Fig.12 Cain and Abel ARP Poisoning Attack Window

Figure (12) shows an ARP Spoofing attack by the computer with the logical address (192.168.200.3), where the attacker begins sending fake response messages to both client1 and client2 stating that each of the other computers' logical addresses are related to the physical address of the attacking computer, so that both computers will exchange information with the attacker instead of communicating with each other's.

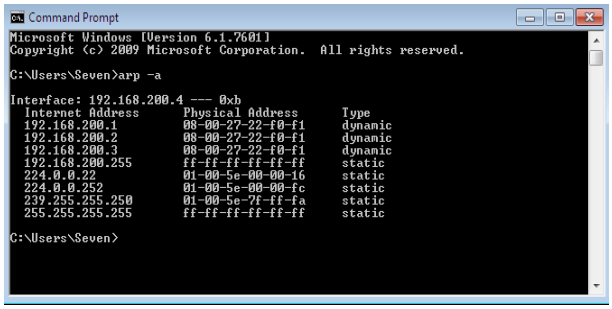

Fig.13 An ARP table on the victim's computer before applying the proposed technique

As for Figure (13), it shows the ARP table at the victim's computer (client1) with the logical address (192.168.200.4), where all the logical addresses obtained dynamically are linked to one physical address (08-00-27-22-f0-f1), which is the physical address of the attacker.

Later on, the client application was run on the victim's computer, and Figure (14) shows the user interface, where the user enters the logical address and the physical address of the server computer as provided by the system administrator, since they are static and do not change. Moreover, the user can enter the logical address and the physical address of the default gateway to the application; protecting those addresses from attacks. After the login process is successful, the trusted ARP table is exchanged with the server and the ARP table on the victim's computer is corrected with right physical address for each computer as shown in Figure (15).

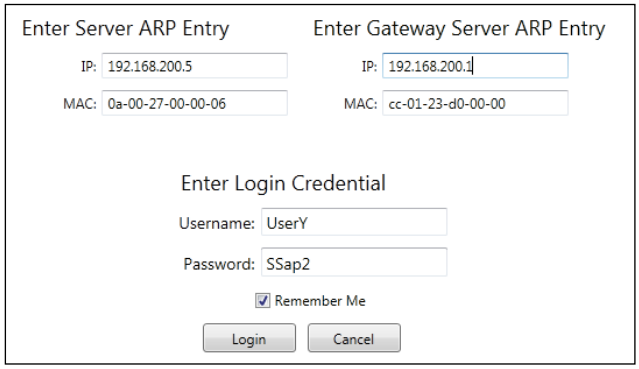

Fig.14 Client application interface

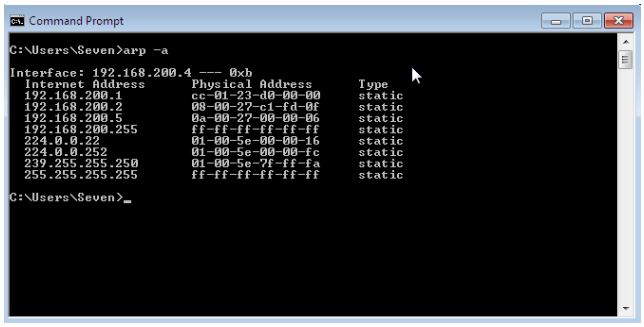

Fig.15 ARP table on the victim's computer after applying the proposed technique

As a result, the ARP Spoofing attack stops at all computers running the client application. Figure (16) shows the information obtained by the attacker from the moment 0 seconds to the moment 600 seconds of the attack, while from the moment 600 seconds (when the technique was applied) and beyond, the flow of The information to the attacker is stopped due to applying the proposed technique and the victim's computer is protected.

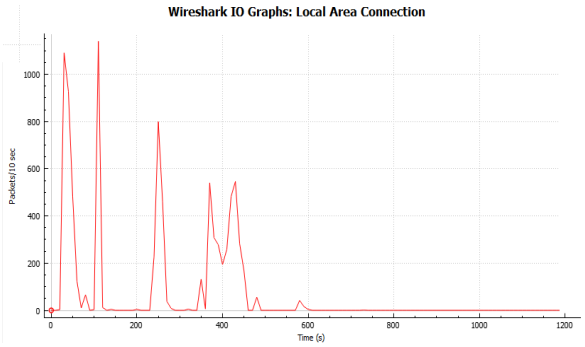

Fig. 16 The amount of information the attacker's computer has

6. CONCLUSION

In this paper, an effective technique was proposed to prevent ARP Spoofing attack in local area networks (LANs) by relying on the static characteristic of the ARP protocol. Two computer applications were designed (server - client) where the server application builds a trusted ARP table and distributes that table to client computers on the subnet. The client application is then automatically updates the ARP table on the computer in a static mode. The results showed the technique's efficiency in stopping the attack, as it prevented the attacker from obtaining any information although the attacker continued to send fraudulent response messages while the proposed technique was working. We also found that, in comparison with previous studies, the process of preventing the attack was done without the need for expensive special equipment or any modifications to ready-made programs or pre-prepared systems.

7. REFERENCES

[1] Rajwinder Kaur, Er. Gurjot Singh, Suman Khurana, 2015, "A Security Approach to Prevent ARP Poisoning and Defensive tools", International Journal of Computer and Communication System Engineering (IJCCSE), Vol. 2 (3), 431-437, ISSN: 2312-7694.

[2] M. Anathi , K. Vijayakumar , 2020, "An Intelligent Approach For Dynamic Network Traffic Restriction Using MAC Address Verification", Published by Elsevier B.V. Computer Communications 154, 559–564.

[3] J. Lach, 2003," Sniffing local network and its detecting", Studia Infor-matica, Vol.2, No.24, pp. 289-296.

[4] Yang Liu, Kaikun Dong, Lan Dong, Bin Li, 2008," Research of the ARP Spoofing Principle and a Defensive Algorithm", Wseas Transactions On Communications, Issue 5, Volume 7, May 2008, ISSN: 1109-2742.

[5] Samvedi A, Owlak S, Chaurasia V. 2014, "Improved secure address resolution protocol", fifth international conference of communications security and information assurance.

[6] Abdel Salam AM, Elkilani WS, Amin KM, 2014, "An automated approach for preventing ARP spoofing attack using static ARP entries", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 5, No. 1.

[7] Raval N, Chaudhary P, 2016, "Detection and prevention of ARP poisoning attacks based on scripts", Int J Adv Res Innovat Ideas Educ-ISSN(O)-2395-4396;2(3):367–374.

[8] Sherin Hijazi, Mohammad S. Obaidat, December 2018, "Address resolution protocol spoofing attacks and security approaches: A survey", John Wiley & Sons, Ltd.

[9] Md. Ataullah, Naveen Chauhan, 2012, "An Efficient and Secure Solution for the Problems of ARP Cache Poisoning Attacks", World Academy of Science, Engineering and Technology, International Journal of Information and Communication Engineering, Vol:6, No:8.

[10] Younes O, 2017, "Modeling and performance analysis of a new secure address resolution protocol", Wiley Int J Commun Syst;31(1): e 3433. https://doi.org/10.1002/dac.3433.

[11] D. Francis Xavier Christopher and C. Divya, 2020, "Address Resolution Protocol Based Attacks: Prevention and Detection Schemes", Springer Nature Switzerland AG 2020 A. P. Pandian et al. (Eds.): ICCBI 2018, LNDECT 31, pp. 247–256, https://doi.org/10.1007/978-3-030-24643-3_30.

[12] Conti, M., Dragoni, N., Lesyk, V, 2016, "A survey of man in the middle attacks", IEEE Commun. Surv. Tutor. 18(3), 2027–2051.

[13] Trabelsi, Z., El-Hajj, W, 2010, "On investigating ARP spoofing security solutions", Int. J. Internet Protoc. Technol. 5(1–2), 92–100.

[14] Singh, J., Kaur, G., Malhotra, J, 2015, "A comprehensive survey of current trends and challenges to mitigate ARP attacks", In: Proceedings of the International Conference on Electrical, Electronics, Signals, Communication and Optimization (EESCO), pp. 1–6. IEEE.

[15] Sudhakar and Aggarwal, R. K, 2017 "A Security Approach and Prevention Technique against ARP Poisoning", Information and Communication Technology for Intelligent Systems (ICTIS) - Volume 1, Springer International Publishing.

[16] Arslan, Yuksel, 2017, "A solution for ARP spoofing: Layer-2 MAC and protocol filtering and ARP server", Research gate.

[17] D. R. Rupal, D. Satasiya, H. Kumar and A. Agrawal, 2016, "Detection and prevention of ARP poisoning in dynamic IP configuration", IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, pp. 1240-1244.

[18] Dr. S. G. Bhirud and Vijay Katkar, 2011, "Light Weight Approach for IP-ARP Spoofing Detection and Prevention", Second Asian Himalayas International Conference on Internet (AH-ICI), page(s):1-5.

[19] Boughrara, A.; Mammar, S, 2012, "Implementation of a SNORT's Output Plug-in in Reaction to ARP Spoofing's Attack", 6th International Conference on Sciences of Electronics Technologies of Information and Telecommunications (SETIT), pp.643,647.

[20] Younes, O.S, 2017, "Securing ARP and DHCP for Mitigating Link Layer Attacks", Sādhanā 42(12),2041–2053.

[21] Moon, D., Lee, J.D., Jeong, Y.S., Park, J.H, 2016, "RTNSS: a routing trace-based network security system for preventing ARP spoofing attacks". J. Supercomput. 72(5), 1740–1756.