Address Resolution protocol (ARP) is used to map logical addresses into its corresponding physical addresses and is widely used protocol in TCP/IP network. ARP protocol doesn't provide any mechanism for authentication amongst hosts in the network. On other hand it is a stateless protocol. These limitations make ARP protocol vulnerable to attacks. ARP poisoning is a type of Man-In-The-Middle (MITM) in which attacker poisons the ARP cache of two hosts and place itself between legitimate traffic. In this work we implemented ARP poisoning using Ettercap and Cain and Abel tool. The work gives the systematic literature reviews of techniques for mitigation of ARP poisoning and bitcoin based model is proposed. The proposed model is based on the bitcoin networks and bitcoin cryptography. Our proposed scheme can efficiently mitigate the ARP poisoning attack as authentication is achieved through digital signature which can only be generated by legitimate host with their private key. Also, other host in the same network can check whether the communication between two hosts is authentic or not.

Mehtab Alam

Investigating ARP poisoning

ARP in great detail

PhD Scholar in Computer Science and Engineering, with Master of Technology in Information Security and Cyber Forensics and Bachelor of Technology in Information Technology. Residing in New Delhi, India.
My areas of interest are Internet of Things (IoT), Smart Cities, Edge Computing, Fog Computing and other similar technologies.

9 786203 911008

LAP LAMBERT Academic Publishing

Mehtab Alam

Investigating ARP poisoning

Mehtab Alam

Investigating ARP poisoning

ARP in great detail

LAP LAMBERT Academic Publishing

Imprint
Any brand names and product names mentioned in this book are subject to trademark, brand or patent protection and are trademarks or registered trademarks of their respective holders. The use of brand names, product names, common names, trade names, product descriptions etc. even without a particular marking in this work is in no way to be construed to mean that such names may be regarded as unrestricted in respect of trademark and brand protection legislation and could thus be used by anyone.

Cover image: www.ingimage.com

Publisher:
LAP LAMBERT Academic Publishing
is a trademark of
Dodo Books Indian Ocean Ltd., member of the OmniScriptum S.R.L Publishing group
str. A.Russo 15, of. 61, Chisinau-2068, Republic of Moldova Europe
Printed at: see last page
ISBN: 978-620-3-91100-8

ABSTRACT

Address Resolution protocol (ARP) is used to map logical addresses into its corresponding physical addresses and is widely used protocol in TCP/IP network. ARP protocol doesn't provide any mechanism for authentication amongst hosts in the network. On other hand it is a stateless protocol. These limitations make ARP protocol vulnerable to attacks. ARP poisoning is a type of Man-In-The-Middle (MITM) in which attacker poisons the ARP cache of two hosts and place itself between legitimate traffic. In this work we implemented ARP poisoning using Ettercap and Cain and Abel tool. The work gives the systematic literature reviews of techniques for mitigation of ARP poisoning and bitcoin based model is proposed. The proposed model is based on the bitcoin networks and bitcoin cryptography. Our proposed scheme can efficiently mitigate the ARP poisoning attack as authentication is achieved through digital signature which can only be generated by legitimate host with their private key. Also, other host in the same network can check whether the communication between two hosts is authentic or not.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ARP	Address Resolution Protocol
DHCP	Dynamic Host Configuration Protocol
S-ARP	Secure-ARP
S-UARP	Secure Unicast Address Resolution Protocol
TARP	Ticket based Address Resolution Protocol
DAPS	Dynamic ARP-spoof Protection & Surveillance
E-SDE	Enhanced Spoof Detection System
GDPS	Gratuitous decision Packet System
SDN	Software Defined Network
PARP-S	Secure Piggybacking based ARP
KARP	Kerberos-ARP
BB-ARP	Bitcoin Based-ARP
ES-ARP	Efficient and Secure ARP
MR-ARP	MITM Resistant-ARP

CHAPTER 1 INTRODUCTION

The network security is a prime concern for the companies in today's scenario. Many companies adopt strict security policies to ensure security within an organization. Attacks on an organization can be inside attacks and outside attacks. The primary goal of the security policies is to achieve CIA (Confidentiality, Integrity and Authenticity) model. Security mechanism is implemented at every layer of the OSI reference model. However, security of the upper layer protocols depends upon the security of the lower layer protocols. Therefore, hardening lower layer protocol is essential for robust network as compromise of lower layer protocol leads to serious inside attacks.

ARP protocol is used to map logical address into its corresponding physical address. The purpose of ARP protocol is address resolution. ARP protocol is a stateless protocol. It doesn't provide any authentication amongst hosts in network thus making it susceptible to attacks. ARP spoofing, or ARP poisoning is an attack in which an attacker poisons the ARP cache of the target hosts and placed itself between legitimate traffic leading to attacks like MITM, sniffing, connection hijacking, connection spoofing and DoS. Thus, makes it necessary to secure ARP protocol.

In this work, we investigated the ARP packet using wireshark protocol analyzer. ARP poisoning attack is implemented using Ettercap tool. The work gives the systematic literature review of techniques that are proposed in the literature. Also, a bitcoin based model for mitigation of ARP poisoning is proposed. The Proposed idea is based on the concept of bitcon network and bitcoin cryptography.

CHAPTER.2 ADDRESS RESOLUTION PROTOCOL

2.1. ARP (ADDRESS RESOLUTION PROTOCOL)

The packet in the computer networks passes through many physical networks and interconnected devices before reaching its destination. In TCP/IP network, every host has two addresses namely- logical addresses and physical addresses. The logical address is a 32-bit network layer address well known as IP address. It is assigned by the DHCP or DNS server. The layer two addresses are called physical address. It is imprinted in the hardware NIC. It is well known as MAC address. The jurisdiction of the MAC address is local network and is unique locally.

Anytime the host sends the IP datagram to another host in the network, it has IP address of the receiver at network layer. However, this IP address should be resolved so that it can be encapsulated within the data link layer frame and pass the physical layer.

ARP protocol maps the IP address to its corresponding MAC address [1]. It is defined in RFC 826 [2]. The mapping can be done is two ways either statically or dynamically. In static mapping, ARP cache table are static which is stored at every host's machine in the network on other hand dynamic mapping finds one of the two address with the use of address resolution. Static mapping has various limitations as physical address can be changed in many ways. Also, for resolving MAC address to its corresponding IP address RARP (Reverse Address Resolution Protocol) is used. Whenever the host communicates with another host in the same network it sends the broadcast request containing its IP address, MAC address and receiver's IP address and asks for receiver's MAC address (Figure.1).

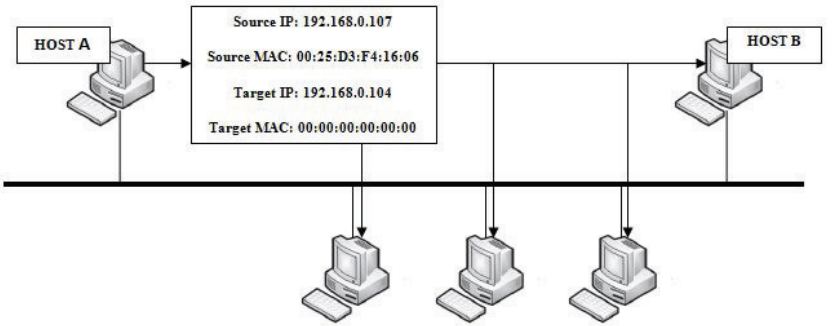

Figure.1 ARP Request

Every host on the network receives the broadcast request but only intended receiver replies with its MAC Address. Other host on the network discards this request. ARP request is broadcast and reply is unicast (Figure.2).

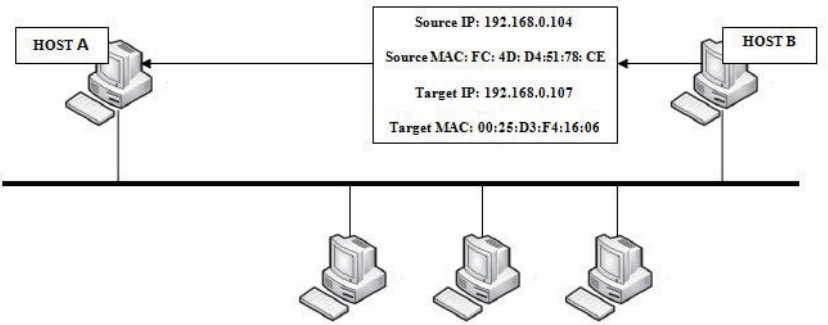

Figure.2 ARP Reply

In an order to minimize the broadcast requests within the network. ARP reply is cached and kept in a cache memory for a while (20-30 minutes). Every host maintains its ARP cache table and before sending the ARP broadcast request it first check its cache. Cache table is record of IP addresses and their corresponding IP addresses [1].

(Figure. 3) depict the format of the ARP header and the fields are as follows [3]: -

Hardware Type- This 16-bit field defines the type of network on which ARP is running.

Protocol Type- A 16-bit field defining protocol.

Hardware Address Length- A 8-bit field defines length of physical address in bytes or octets. For Ethernet the length is 6.

Protocol Address Length- It defines the length of the logical address in bytes or octets.

Operation- It's a function of ARP packet- 1 for request and 2 for reply.

Sender Hardware Address- It's a Sender's hardware address.

Sender Protocol Address- It's an Address of sender's upper layer protocol.

Target Hardware Address- The recipient hardware address. Always zero in request.

Target Protocol Address- It's an Address of receiver's upper layer protocol.

Address Resolution Protocol

Bit Offset	0–7	8–15
0	Hardware Type	
16	Protocol Type	
32	Hardware Address Length	Protocol Address Length
48	Operation	
64	Sender Hardware Address (1st 16 Bits)	
80	Sender Hardware Address (2nd 16 Bits)	
96	Sender Hardware Address (3rd 16 Bits)	
112	Sender Protocol Address (1st 16 Bits)	
128	Sender Protocol Address (2nd 16 Bits)	
144	Target Hardware Address (1st 16 Bits)	
160	Target Hardware Address (2nd 16 Bits)	
176	Target Hardware Address (3rd 16 Bits)	
192	Target Protocol Address (1st 16 Bits)	
208	Target Protocol Address (2nd 16 Bits)	

Figure.3 ARP Packet Structure [3]

2.2. VIEWING ARP CACHE ENTRIES

Static and dynamic ARP cache entries can be easily viewed on windows operating system by command prompt. The steps are as follows: -

1. Open the Command Prompt.
2. ARP cache can easily be viewed by arp – a command in command prompt (Figure.4).

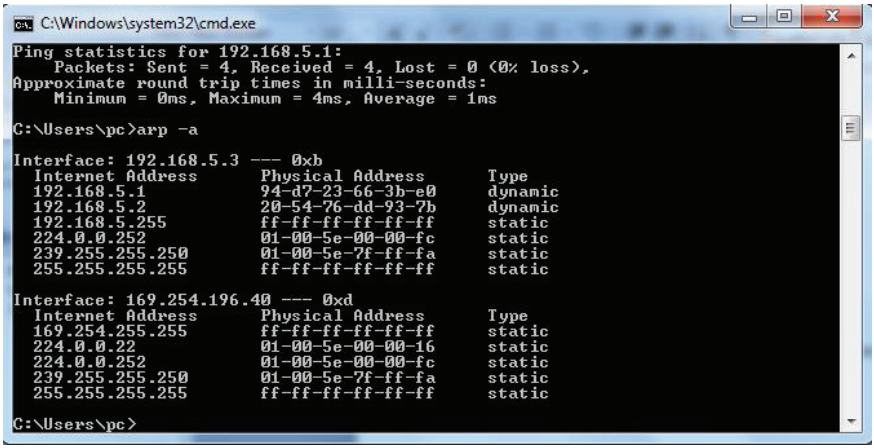

Figure.4 Using arp –a command

3. Addition in cache entries can be done manually by arp – s command (Figure.5)

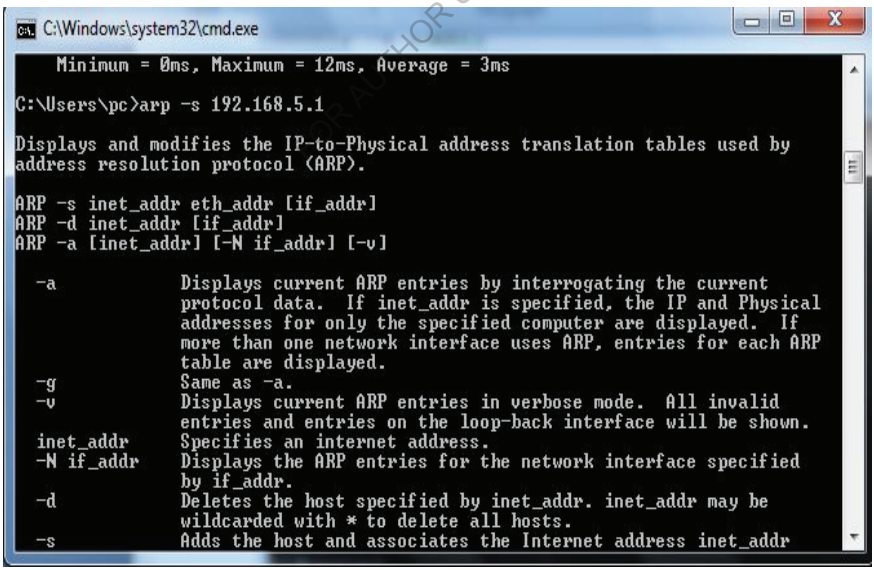

Figure.5 Using arp –s command

4. Also to delete the ARP cache manually arp –d command is used (Figure.6)

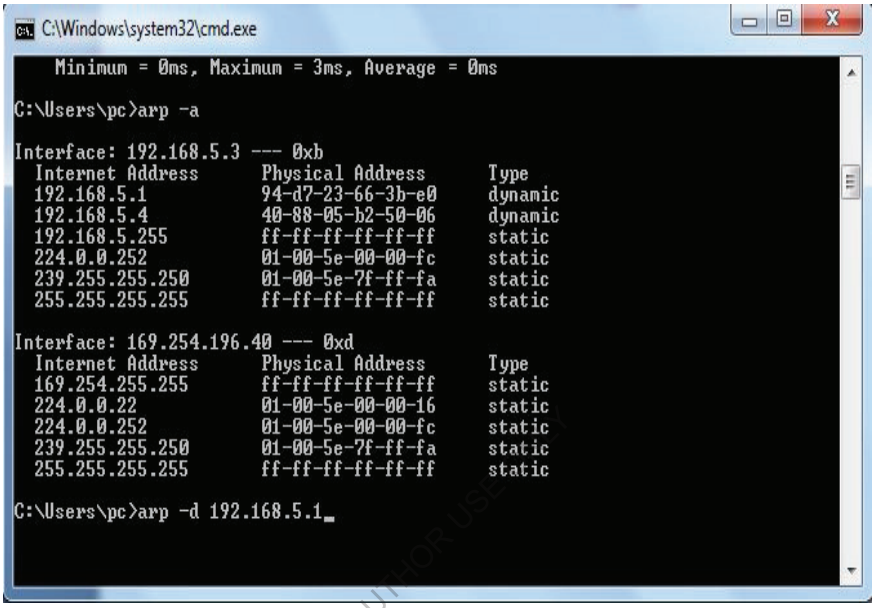

Figure.6 Using arp –d command

2.3. ANALYZING ARP PACKETS

ARP packets can be analyzed using wireshark network protocol analyzer. We analyzed the two packets used by the ARP protocol for dynamic mapping for IP address and MAC address namely- ARP request packet and ARP reply Packet.

1. *ARP request*- The first packet is ARP request packet which is a broadcast request containing source's IP address, MAC address and target's IP address (Figure.7). As shown in Figure. The receiver MAC address is 00:00:00: 00:00:00 as it is marked zero in ARP request. Opcode (1) depict it's an ARP request packet. The hardware address for Ethernet is 6.

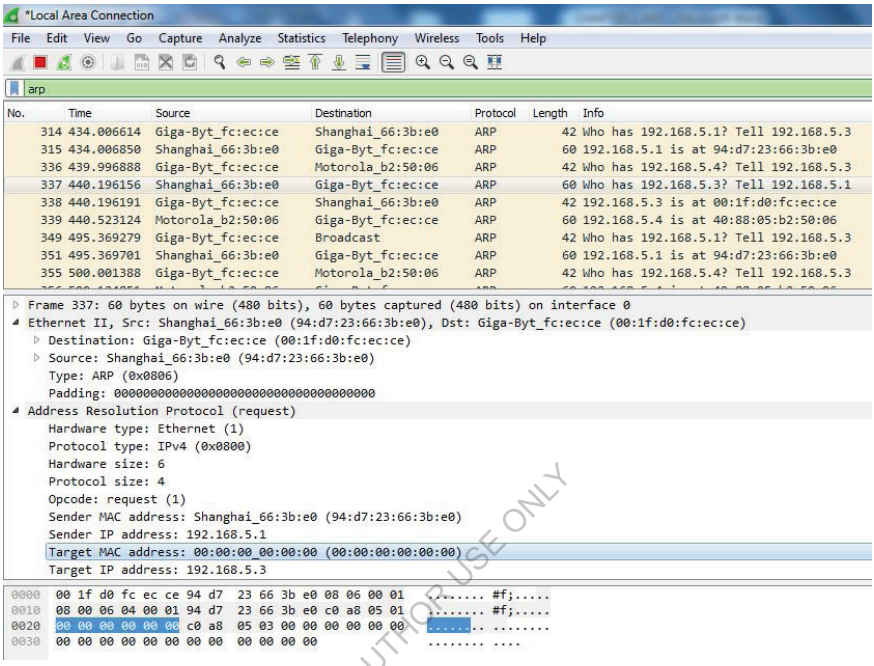

Figure.7 Investigating ARP Request packet in wireshark

2. *ARP reply*: On response of the broadcast request the receiver sends it MAC address. The receiver sends its MAC address. The Opcode here is (2) depict it's a ARP reply request as shown in (Figure.8).

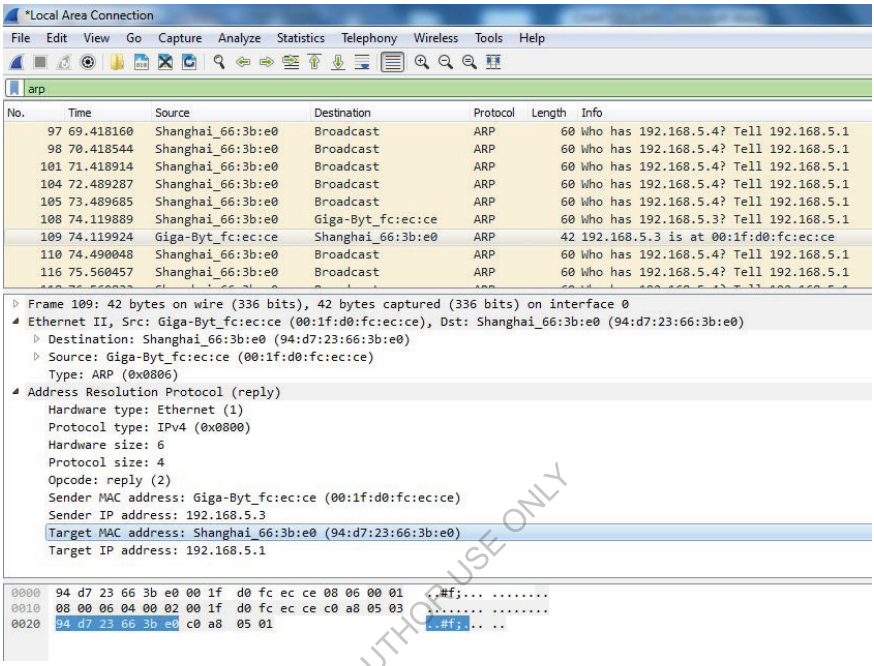

Figure.8 Investigating ARP reply packet in wireshark

2.4. ARP POISONING (MITM) ATTACK

ARP protocol is a stateless protocol, it doesn't keep track of request coming and going out. Also ARP protocol doesn't guarantee any authentication amongst host in the network. These factors make ARP protocol susceptible to attacks [4].

ARP cache poisoning, ARP spoofing, or ARP poison routing is the Man-In-The-Middle attack in which attacker placed itself in between two legitimate hosts and poison their ARP cache table [6] (Figure.10). This is done by forge IP addresses, sending fake ARP replies, sending fake IP address and so on [4].

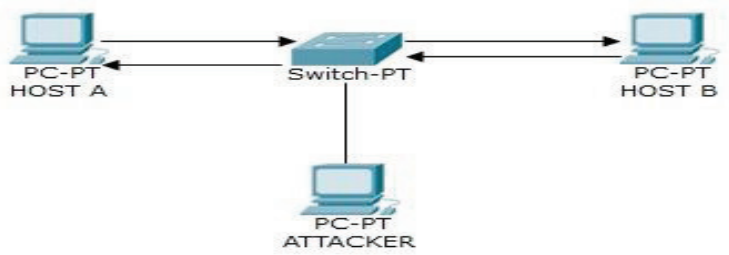

Figure.9 Legitimate traffic between Host A and Host B

(Figure.9) shows the legitimate traffic between Host A and Host B. After the successful ARP poisoning attack, attacker place itself between the traffic (Figure.10). The attacker can now intercept and view the information. The objective of ARP poisoning is to take over the session. ARP poisoning attack can cause sniffing, connection hijacking, connection spoofing and DoS [5].

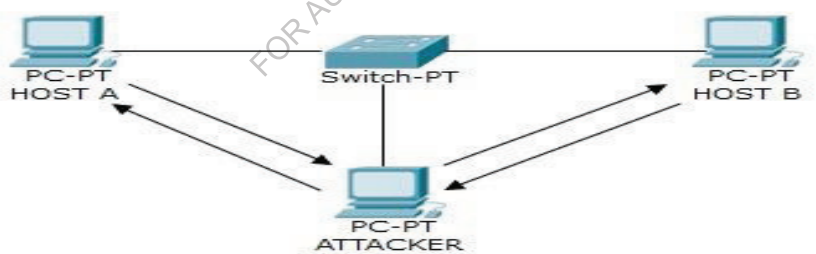

Figure.10 Traffic after ARP poisoning

Consider the scenario where three Hosts are connected via Ethernet. Host A wants to communicate with Host B. Attacker can implement the ARP poison and poison the cache through various ways [4]: -

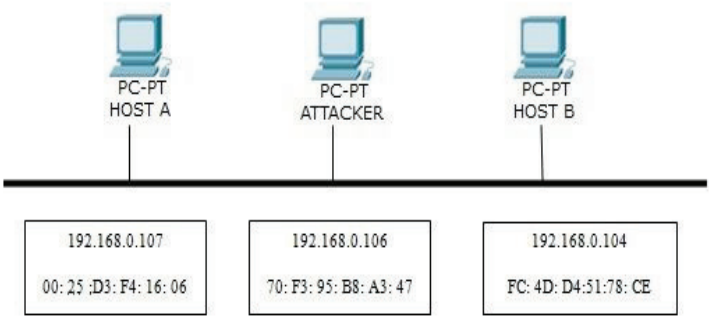

Figure.11 ARP poisoning setup

Scenario one: Broadcast request

Host A sends the broadcast ARP request. This ARP request is received by every host connected via Ethernet. Attacker in this scenario uses the IP address of the Host A and sends the ARP broadcast request. This results in the two hosts with same IP address. This request is stored by the Host B's ARP cache table resulting in the ARP cache poisoning [4].

Scenario two: Multiple responses

In this scenario, Host A receives the response from the attacker and Host C. There exists a race condition that attacker's response is received first by the Host A [4].

Scenario three: Unsolicited response

Since ARP protocol is a stateless protocol. It doesn't keep track of requests going out. Therefore, an unsolicited response sometimes causes ARP cache poisoning [4].

2.5. IMPLEMENTATION OF ARP POISONING USING ETTERCAP

There are various tools available in the market through which ARP cache poisoning can be implemented such as Ettercap, Subterfuge, Arpoison, Arpspoof, ARP-FILLUP-v0.1, Arp-sk-

v0.3.2, arping, Cain and Abel, SwitchSniffer and many more. The implementation is done on windows operating system. Ettercap tool was used to launch ARP poisoning.

It should be noted that implementation is done in a controlled environment and three laptops was connected to the LAN. Host A IP address is 192.168.0.107, Attacker IP address is 192.168.0.106 and Host C IP address is 192.168.0.104. Host A and Host C ran on windows operating system. Attacker ran on Kali Linux and used Ettercap to implement ARP poisoning attack. The steps for attack are as follows-

1. Start the Ettercap on attacker machine and configure it for "Promisc Mode" via options drop down menu (Figure.12).

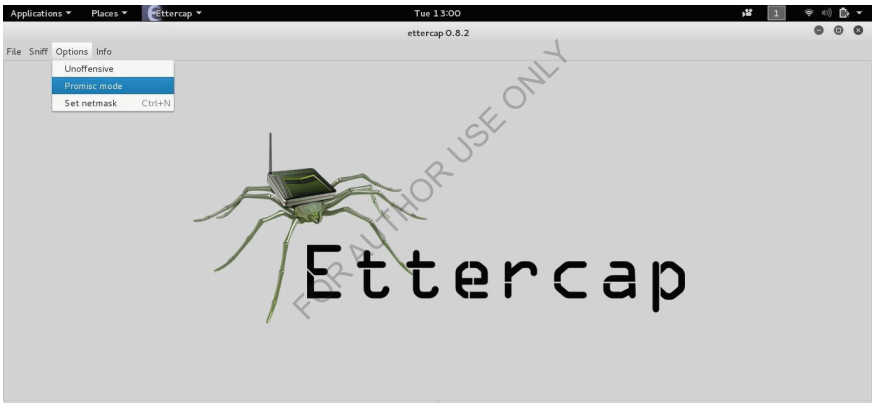

Figure.12 Promisc mode option

2. Start the "Unified Sniffing" through sniff drop down menu (Figure.13).

Figure.13 Unified sniffing option

3. Select the interface (wlan0) as Hosts are connected on the LAN (Figure.14).

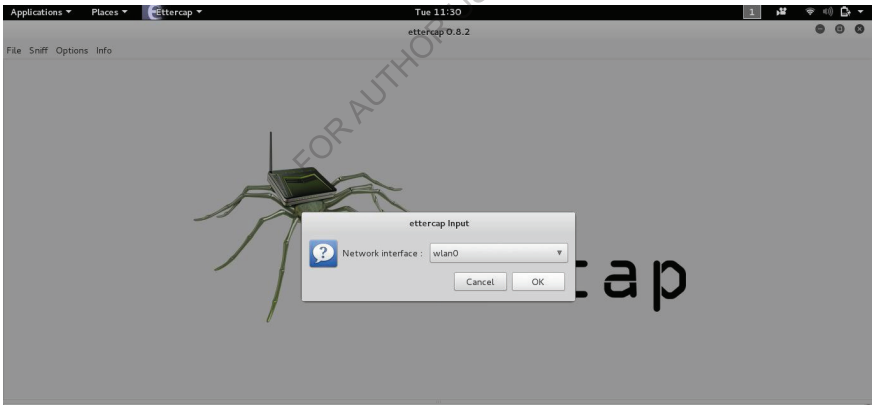

Figure.14 Network interface selection

4. Click on the “Hosts” drop down menu and select “Scan for hosts” option for scanning the hosts that are connected to the LAN (Figure.15). Once the scanning is completed, the list of hosts was viewed (Figure.16) by selecting Host list option in Hosts drop down menu (Figure.17).

Figure.15 Selecting “Scan for host” option

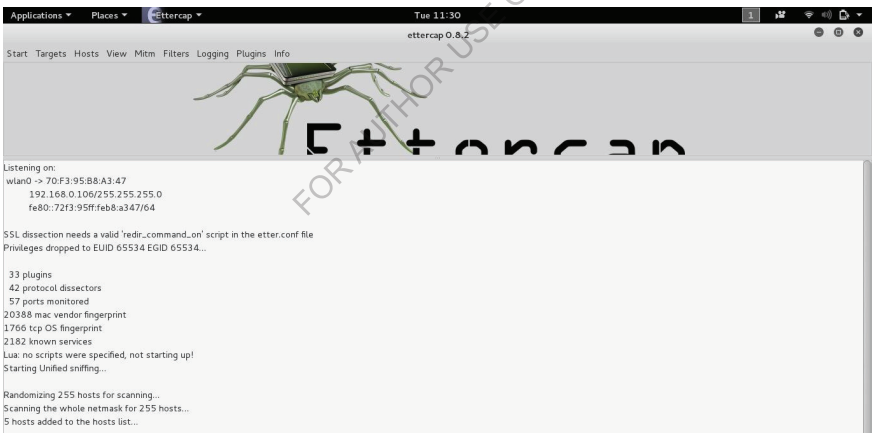

Figure.16 Hosts scanned in the network

Figure.17 Selecting"hosts list" option

5. From the list of hosts connected on the LAN, Attacker selects the two targets for MITM attack (Figure.18) (Figure.19).

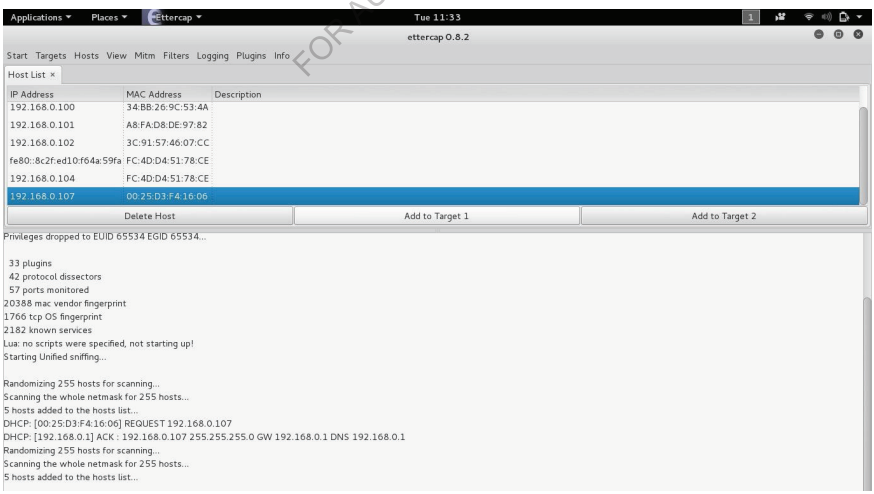

Figure.18 Selecting target 1 for ARP poisoning

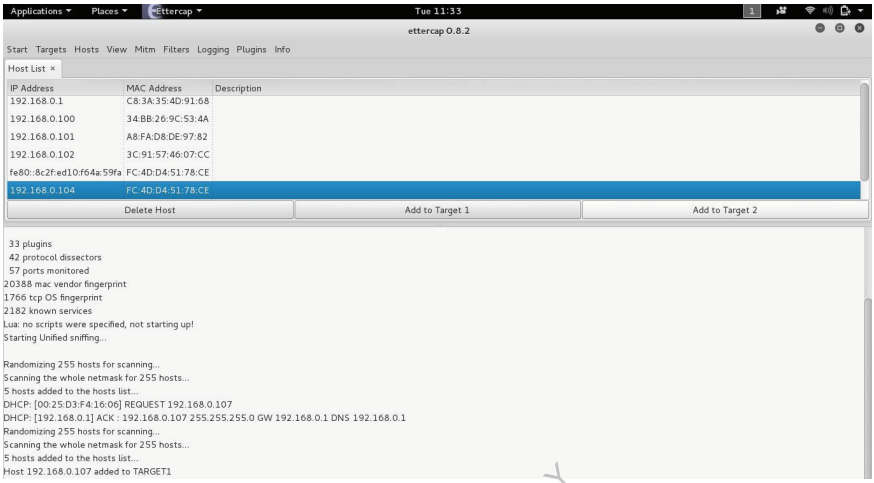

Figure.19 Selecting target 2 for ARP poisoning

6. After selecting the two targets, click on the "Start" drop down menu and select "Start sniffing" (Figure.20).

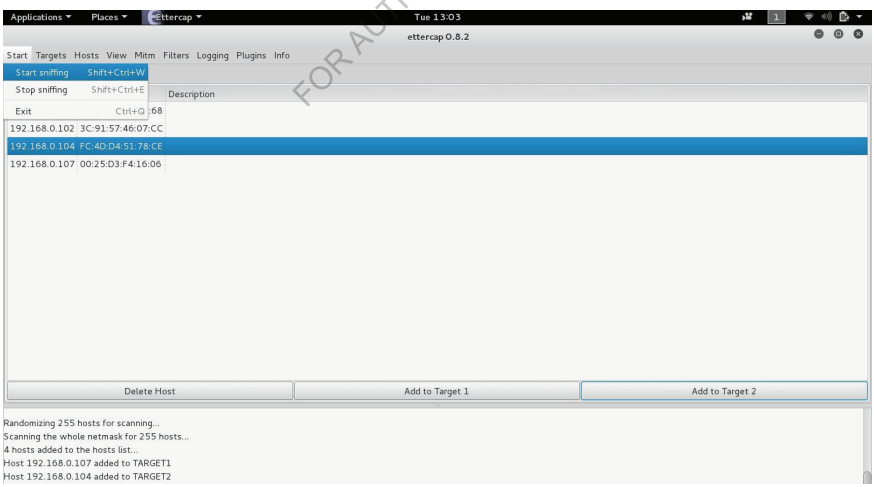

Figure.20Selecting "Start sniffing" option in ettercap

7. Once the sniffing started, select the “Arp poisoning” option in the “Mitm” drop down menu and select “Sniff remote connection” (Figure.21) (Figure.22).

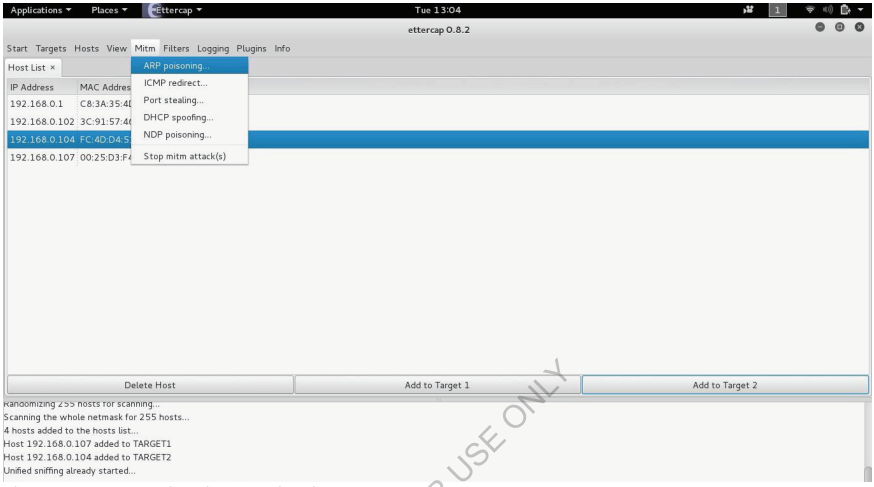

Figure.21“ARP poisoning” option in ettercap

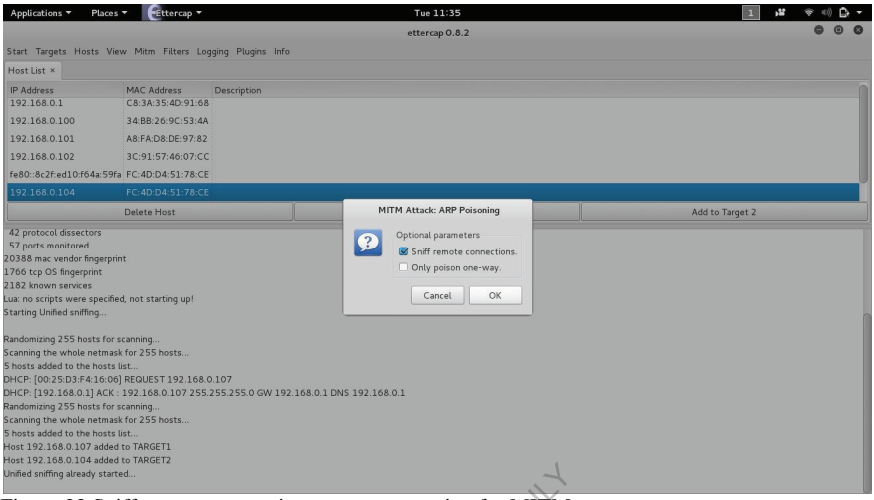

Figure.22 Sniff remote connections parameter option for MITM

8. Under the “Plugins” drop down menu select “Manage the plugins” option (Figure.23). Select “repoison_arp” and “remote_browser” parameters. The “remote_browser” parameter allow attacker’s web browser to display the web pages of the victims (Figure.24) (Figure.25).

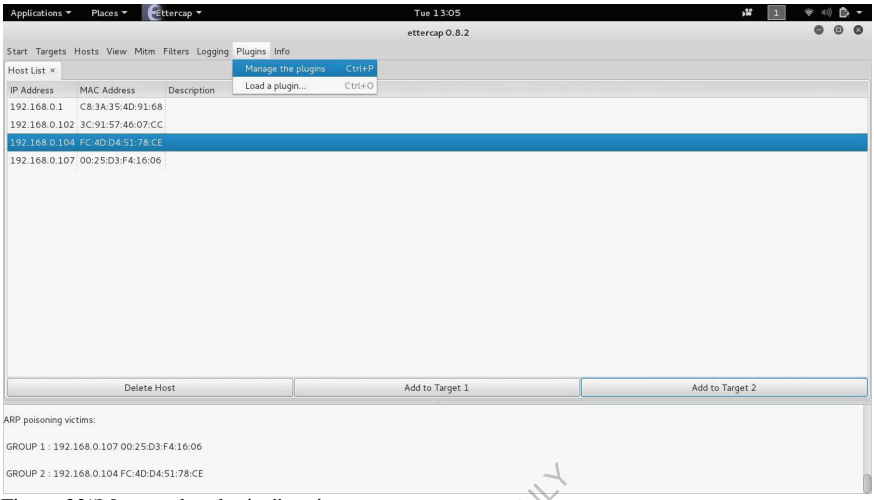

Figure.23"Manage the plugins" option

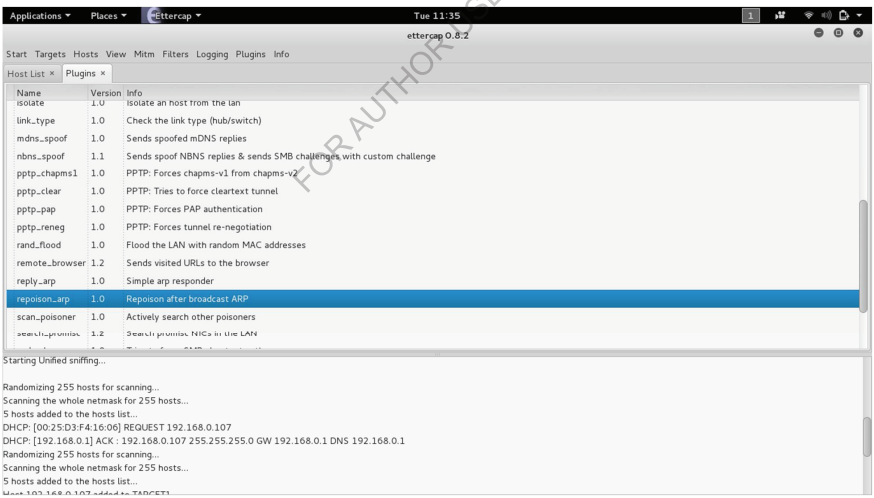

Figure.24 Selecting "repoison_arp" parameter

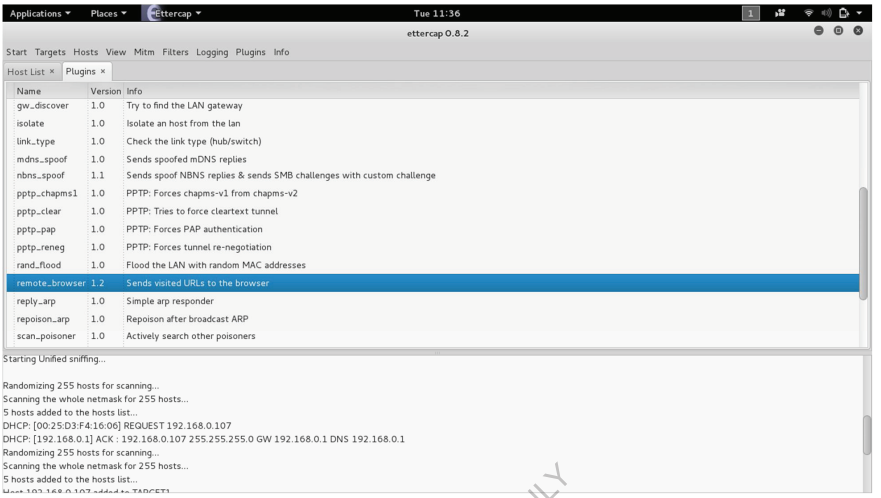

Figure.25 Selecting “remote_browser” parameter

9. After selecting the parameters, go to “View” drop down menu and select “Connections”. This shows the active and idle connection with source IP address and destination IP address (Figure.26) (Figure.27).

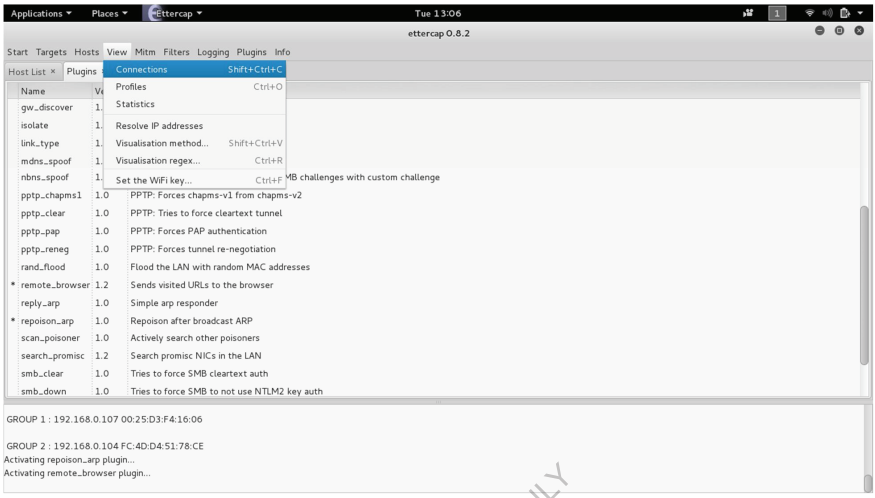

Figure.26 viewing connections by selecting "Connections" option

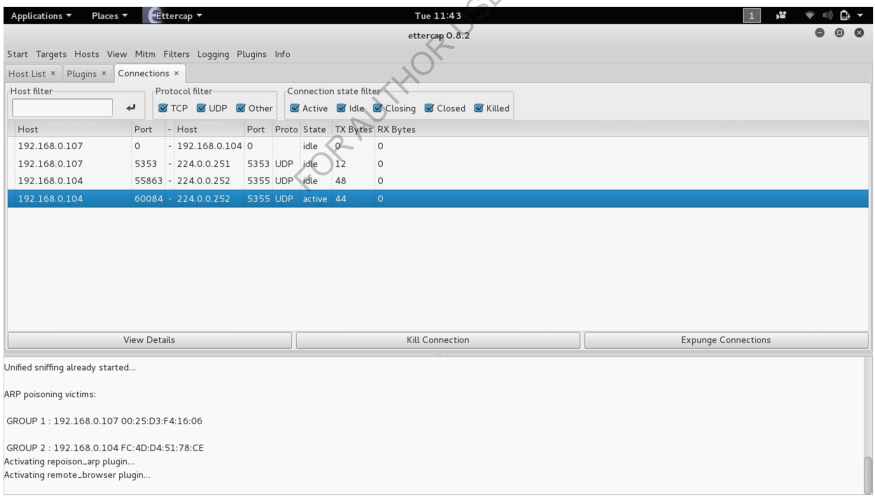

Figure.27 Active and idle connections

10. Now Attacker can easily view the connection detail by right clicking the mouse and choosing "View details" option (Figure.28). This will reveal the protocol, bytes

transferred, source IP address and MAC address. Also destination IP address and MAC address. By double clicking on the connection, screen is spiltted into two parts. Attacker can choose the targets and can inject characters (Figure.29) (Figure.30).

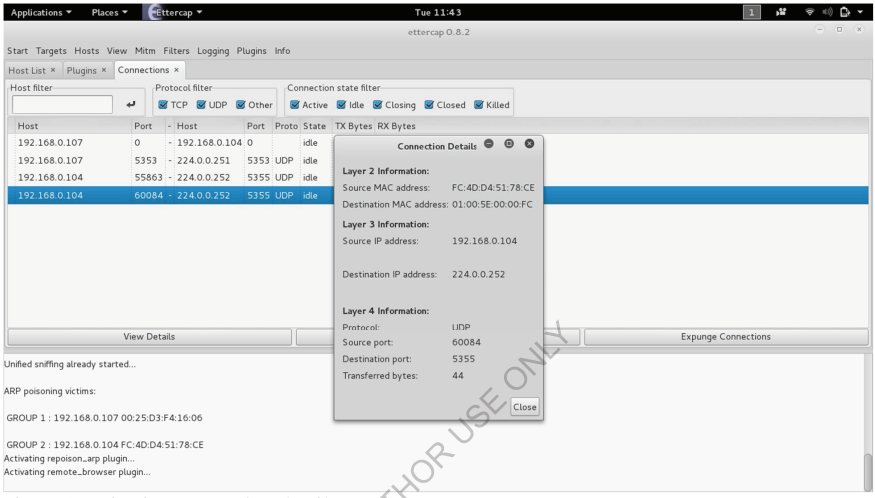

Figure.28 Viewing connection details

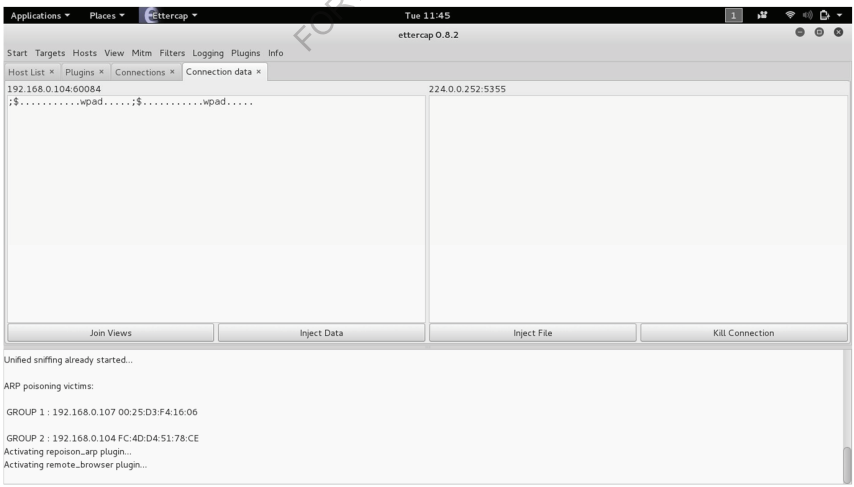

Figure.29 double clicking on connection to split screen

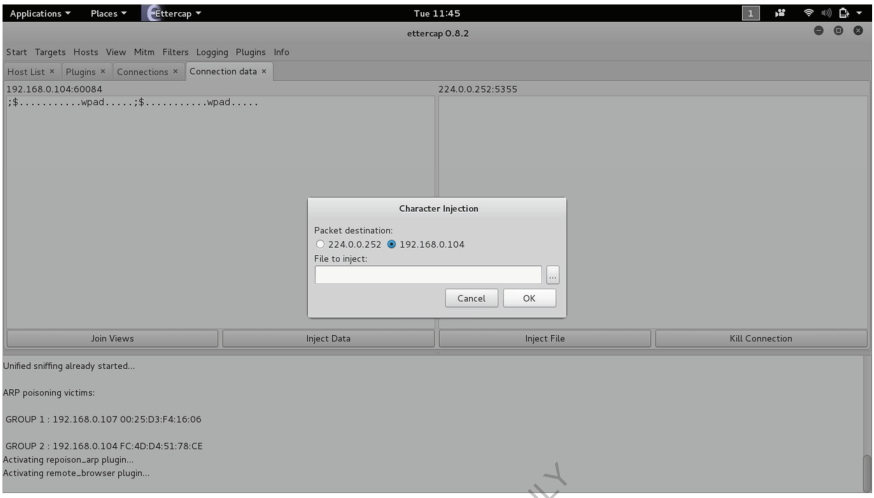

Figure.30 Character injection

The Attack can easily be investigated in the network by wireshark protocol analyzer. Selecting "wlan0" as an interface and click on the "Start" option (Figure.31). After starting the capturing, the protocol, apply "arp" to the filter and apply (Figure.32). The poisoned ARP packet is easily rectified (Figure.33).

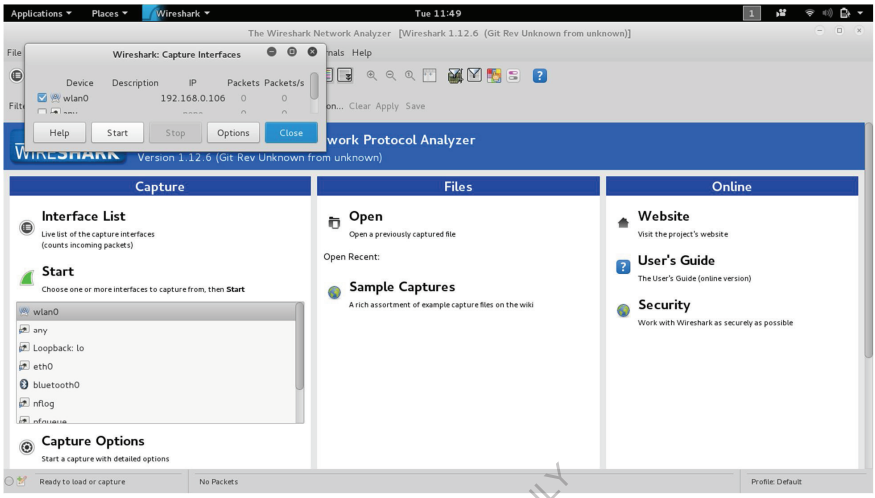

Figure.31 Selecting interface on wireshark

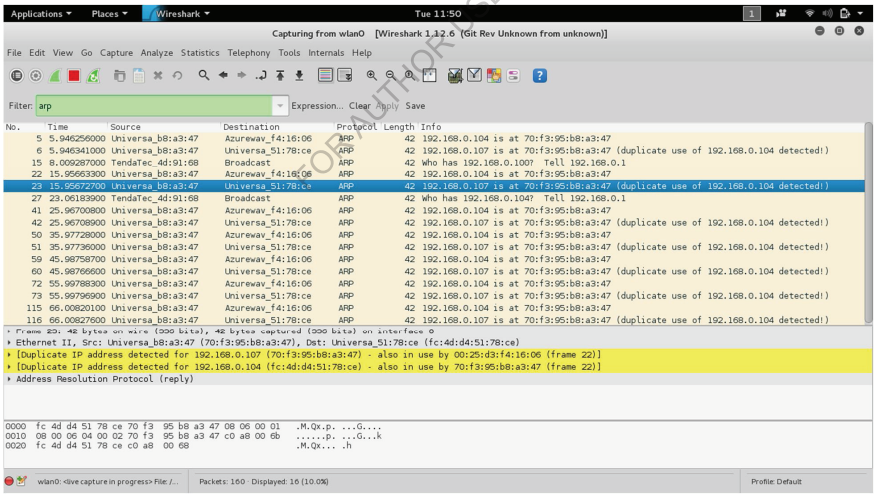

Figure.32 Capturing ARP protocol by using “arp” keyword in filter option

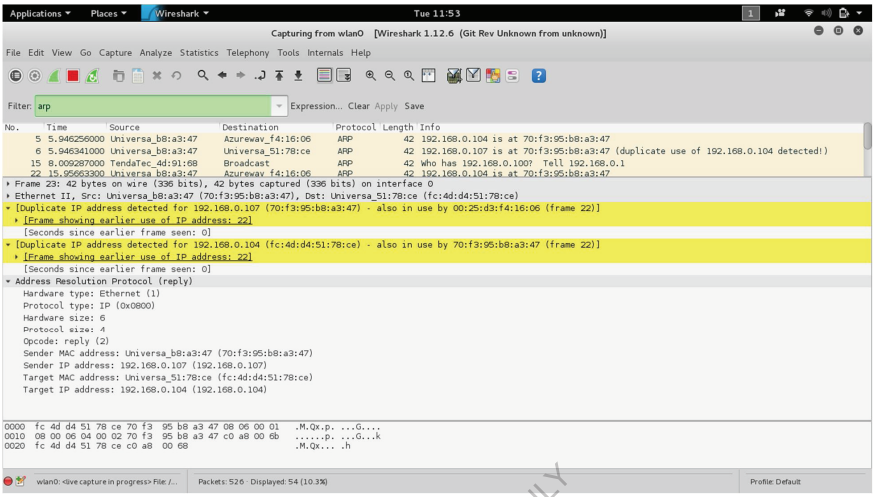

Figure.33 Viewing ARP reply packet in wireshark

2.6. IMPLEMETATION OF ARP POISONING USING CAIN AND ABEL

Cain and Abel is a password recovery tool for windows operating system. This is done by sniffing the network. However, ARP poisoning can be done using Cain and Abel by following the below steps: -

1. Start the Cain and Abel and click on the Start/Stop sniffer button (Figure.34)

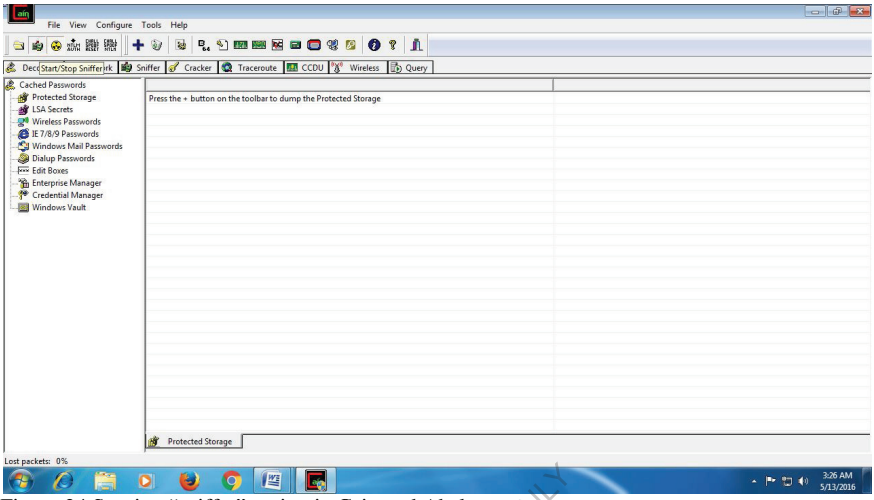

Figure.34 Starting "sniffer" option in Cain and Abel.

2. Choose the "Sniffer" option and click on the Add (+) button to scan all hosts in the subnet (Figure.35).

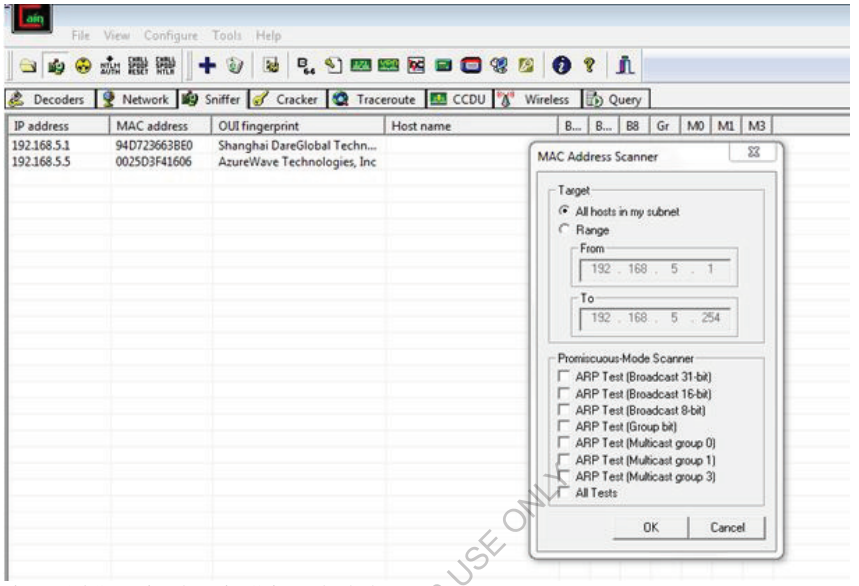

Figure.35 Scanning host in Cain and Abel

3. Start "APR" option and select the target for ARP poisoning by Add (+), this will add the hosts in the list. After selecting the two target if the status is not idle passwords and HTTP packets can be easily recovered. (Figure.36) (Figure.37) (Figure.38) (Figure.39).

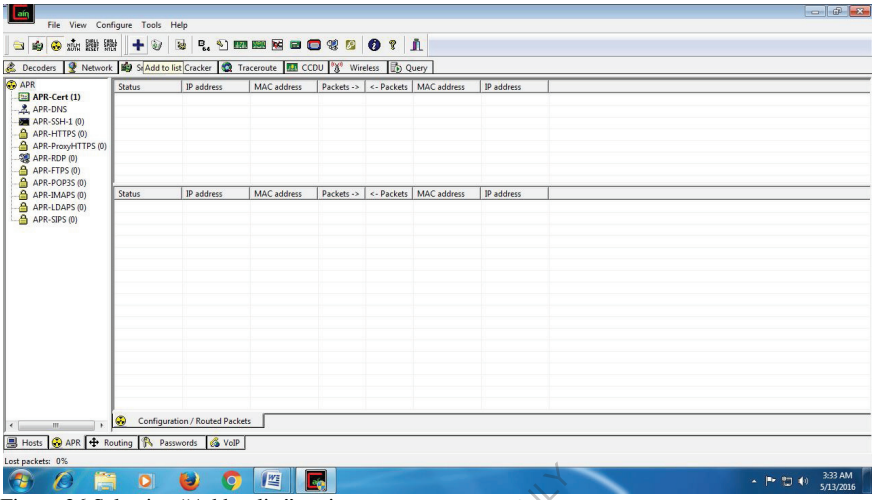

Figure.36 Selecting "Add to list" option

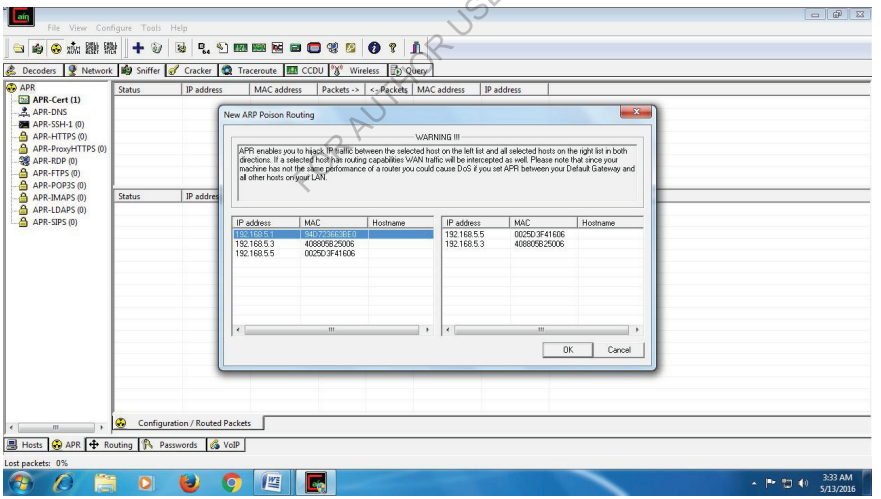

Figure.37. Selecting target IP address for ARP poisoning

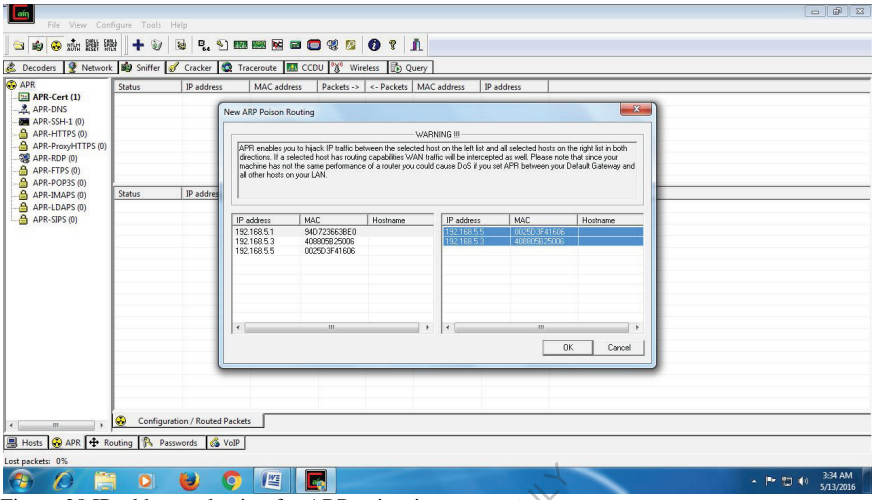

Figure.38 IP address selection for ARP poisoning

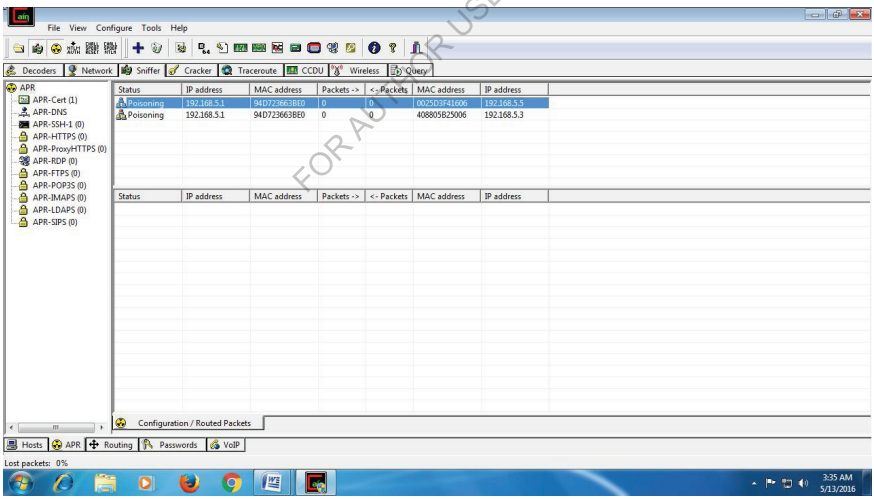

Figure.39 ARP poisoning process initiation

2.6. DETECTING AND PREVENTING ARP POISONING

The ARP poisoning can be detected in the network by using the IDS (Intrusion Detection System) and use of Firewalls. There are many tools available which can be used to rectify ARP poisoning within an organization. Some of them are Snort, Colasoft, Arpalert, Arpwatch, anti-spoof, Antidote, Xarp etc. These are GUI based tools which is easy to use and deploy. We have used Xarp for detecting the ARP poisoning within the network. (Figure.40).

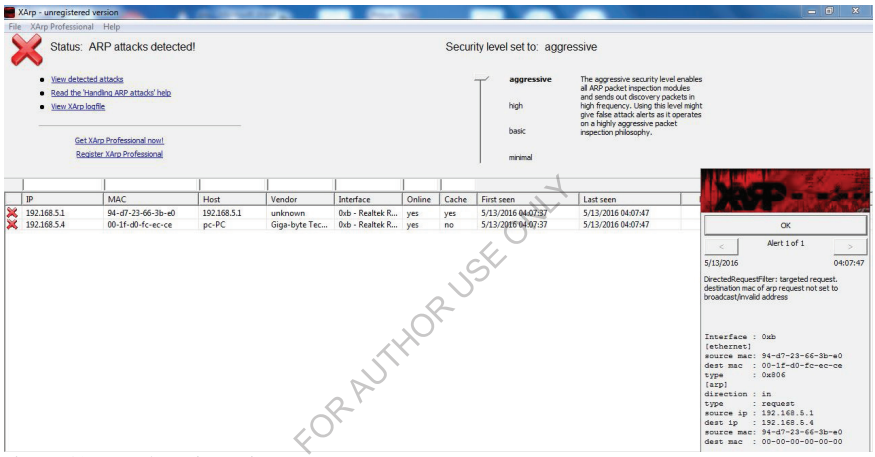

Figure.40 ARP detection using Xarp

CHAPTER 3 LITERATURE REVIEW

The extensive literature review was carried as per the guidelines proposed by Kitchenham [7]. The objective of carrying literature review was to gain deeper understanding of mitigation techniques that exists in literature and to find gap in the study. The literature has been carried out in following journals: -

1. Springer
2. ACM digital library
3. Wiley online library
4. Science Direct
5. IEEE Xplore

The search term was "ARP poisoning" in the search field. The papers in journals and conference proceedings were taken into consideration for the literature survey. The result of the search is shown in (Table.1).

S.NO.	JOURNAL	SEARCH RESULTS	RELEVANT PAPERS
1.	IEEE Xplore	41	16
2.	Springer	118	1
3.	Science Direct	720	2
4.	Wiley Online Library	421	0
5.	ACM	145	3

Table.1 Search results and relevant papers

Total 22 relevant papers were selected for review. The papers in journals and conferences are taken into consideration.

One of the earliest paper in literature was published by Tripunitra and Dutta in 1999 [8], proposes a middleware approach that utilizes design constraints. The implementation was done on stream based networking subsystem. ARP poisoning can be detected and prevented but limitation was compatibility of existing network architecture with proposed one.

The paper by Bruschi. Ornaghi and Rosti in 2003 [9] presents S-ARP in which public and private key are distributed to every hosts in the network by Central Server that act as a Certificate Authority. PKI based authentication mechanism is used. S-ARP has a limitation of single point of failure as if the Central Server fails; the whole network will not work. Also the cost of manual configuration to give keys to new hosts.

In 2005, Isaac and Mohammed [10] proposed S-UARP that uses ARP unicast request mechanism rather than broadcast request to Central Server. Thus, minimizes the broadcast requests within the network. As technique depends on Central Server it has a limitation of single point of failure.

In 2007, two papers were published. The work by Trabelsi and El-Hajj [11] introduced a technique that uses ARP stateful cache instead of traditional ARP stateless cache via Fuzzy Logic Controller. The paper by Loorah et al [12] proposes a TARP where tickets and IP/MAC mapping are centrally distributed to the hosts by DHCP server. Since it is a centralize approach, it has a limitation of single point of failure.

In 2009, three paperswere published. Ortega et al [13] introduced a scheme that uses SOHOs by using low end networking equipment running on the OpenWrt firmware. The paper by Hammouda and trabelsi [14], proposes scheme where modification of switch is done to act as a "Trusted Authority" and authenticate hosts while communication. The scheme has dependability on switch configuration. The work done by Puangpronpitag and Masusas [15] presents dynamic ARP spoof protection and surveillance (DAPS) for mitigation of ARP poisoning.

In 2010, Nam et al [16] invented the voting based resolution mechanism MR-ARP, where concept of voting is done by the hosts in an order to prevent ARP poisoning. It has some limitations like it is not valid in 802.11 networks.

In 2011, Dangol et al [17] proposes GARP. In this protocol the ARP reply is broadcast not unicast and Central Authority monitors ARP table.

In 2012, four papers were published. Kumar and Tapaswi [18] present a centralize mechanism where every host check and validates their ARP entries through ARP Central Server (ACS). ACS corrects and detects ARP poisoning. It is susceptible to single point of failure. Again, Nam et al [19] published a paper that improves voting based mechanism of MR-ARP [16] through puzzle based computational method. In an order to gain fair voting RSA algorithm is used. Salim et al [20] introduced GDPS that detects the doubtful packets and legitimate host can be recognized by sending modified request of gratuitous packets. Ataullah and Chauhan [21] launched ES-ARP that is a stateful protocol and cache is updated after each communication.

In 2013, three papers were published. The paper by Tripathi and Mehtre [22], suggested an approach where ICMP based secondary cache is maintained by every host in the network. According to ICMP responses, the cache is updated. It has an advantage as there is no single point of failure. Nam et al [23] in their paper proposes a scheme to enhance the voting amongst host by adding some parameters like filtering, key parameters and early response. Pandey [24] presents E-SDE in which ICMP and ARP packets are used as a probe packets and algorithm for this is introduced.

In 2015, five papers were published. Masoud et al [25] make use of SDN (Software Defined Network) for mitigating ARP poisoning. Tian [26] et al introduces arpsec, a secure ARP/RARP that utilizes TPM commodity as attestation base, verify the identity of the tar-get machine and doesn't require protocol modification. Saputro and Akkaya [27], presents PARP-S but it is suited for 802.11s based smart grid networks. Bakhache and Rostom [28], launched a method that use Kerberos protocol for authentication. This technique was named KARP. It is low cost but has a limitation of single point of failure. The paper by Arote and Arya [29] detects and prevent ARP poisoning by voting among host and modified ICMP.

CHAPTER 4 PROPOSED WORK

This work proposed a Bitcoin network based approach for the mitigation of ARP poisoning attacks. Bitcoin is a transaction system where every host in the network maintains a digital file called as Ledger [30]. The various host on the network exchange money by exchanging ledger. Bitcoin starts with the creation of bitcoin account which contains the account number, public key and the private key of the host. Unlike ARP cache table, bitcoin ledger table contains the host's name and its balance (Figure.41) (Figure.42).

Ledger

Alice	12.5$
Bob	10$
Dave	18.5$

Figure.41 Bitcoin Ledger

ARP cache table

192.168.0.1	00:25:D3:F4:16:06
192.168.0.2	70:F3:95:B8:A3:47
192.168.0.3	FC:4D:D4:51:78:CE

Figure.42 ARP cache table

Bitcoin is based on decentralized network. There is no centralizing authority in bitcoin transaction like bank. Suppose Host A wants to send 5$ to Host B, Host A sends the broadcast request containing digital signature and transaction message. Host B receives the request update its balance with plus 5$ resulting in balance of Host A minus 5%. Now this updated ledger is broadcasted to the network so that everyone updates their ledger. In bitcoin, everyone can see everyone else balances and transactions [30]. Now the following questions arise: -

Who maintains the ledger?

Every host in the network maintains its own copy of ledger there is no centralize authority which owns the ledger (Figure.45).

How authentication is done?

The authentication in bitcoin is done by digital signature by make use of host's private key (Figure.43).

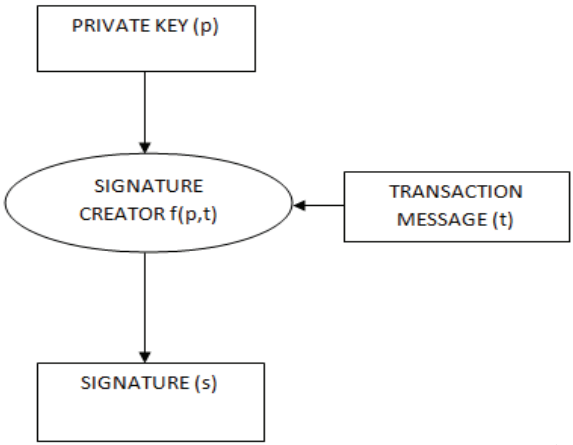

Figure.43 Digital signature generation

How other Hosts make verify that request is legitimate?

The broadcast requests are easily verified by the other hosts in the network as another function is created and it allows other people to check whether the transaction is legitimate or not (Figure.44).

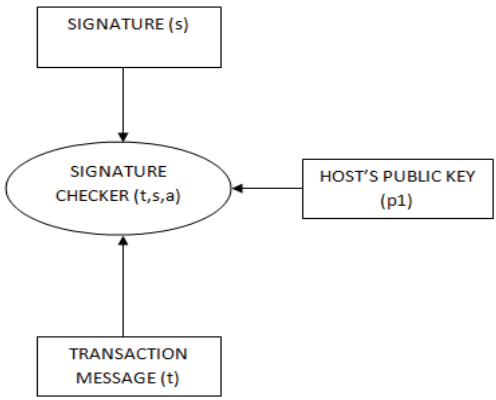

Figure.44. Signature verification process

In bitcoin transactions, the cryptographic functions and mathematics is used to achieve authentication amongst the host. However, the concept if used in ARP protocol can enhance its security and makes it less susceptible to attacks. The processing of broadcast request is more in bitcoin as compare to ARP protocol, but if the security is the prime concern. It can be used with ARP protocol. In this work we propose a Bitcoin Based ARP (BB-ARP).

4.1 BITCOIN BASED ADDRESS RESOLUTION PROTOCOL (BB-ARP)

In bitcoin based ARP protocol every host maintains the ledger containing host's account number, public key and private key. Also the list of corresponding MAC addresses and its IP addresses.At the administrator level each host on the network will be assigned a pair of public key and private key. Public key will have distributed in such a way that every host in the network will have everyone's public key. Now if host want to resolve its IP address into its corresponding MAC address it broadcast the message encrypted with the public key of destined host so that it can only decrypted by it.

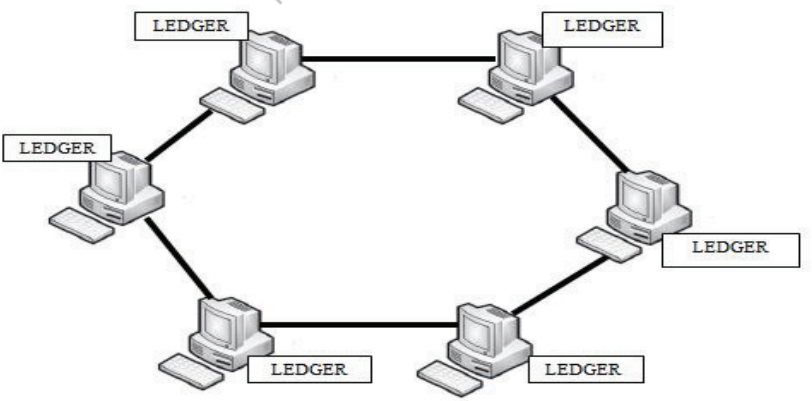

Figure.45 BB-ARP hosts in the network

When Host A communicates with Host B in the network. Host A sends the broadcast request containing digital signature of host and a transaction message (Figure 46). Host B encrypt the message by using Host A's public key and its private key. Other Hosts on network can verify whether the transaction between Host A and Host B is legitimate or not (Figure 44). After the successful communication or transaction Host A and Host B broadcast the updated transaction message containing IP address and MAC address mapping. Therefore, other host on the network updates their ledger containing updated information. In BB-ARP every host maintains its copy of ledger and mechanism is decentralized (Figure.47) (Figure.45).

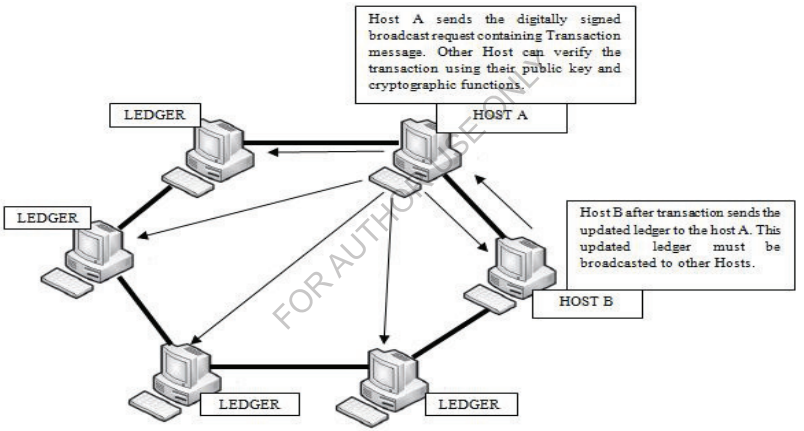

Figure.46 Communication via bitcoin based ARP

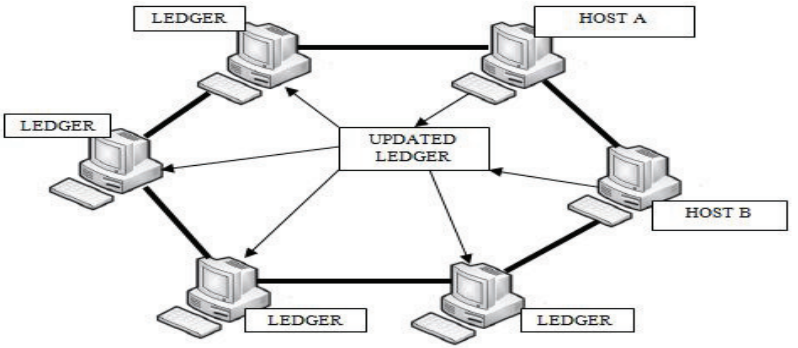

Figure.47 Broadcast of updated ledger

CHAPTER 5 CONCLUSION AND FUTURE SCOPE

The systematic literature review has been carried out in an order to find techniques that were proposed for mitigation of ARP poisoning. There are various techniques that exists in literature but has limitations and constraints. Therefore, need for proposing new technique arises.In this work the bitcoin based model is proposed for the mitigation of ARP poisoning. The authentication is done by the digital signature and other hosts can verify whether the communication between two host are legitimate or not. Since the approach is decentralized it is not prone to single point of failure problem. The proposed scheme has a limitation of cost of manual configuration as public key and private key is distributed to the new host manually every time. Apart from this, it provides robust security and strong authentication.

REFERENCES

1. B. Forouzan and S. Fegan, *Data communications and networking*. New York: McGraw-Hill Higher Education, 2007.
2. 1982 [Online]. Available: https://tools.ietf.org/html/rfc826. [Accessed: 10-Apr-2016].
3. C. Sanders, *Practical Packet Analysis: Using WIRESHARK to Solve Real-World Network Problems; Second Edition*. No Starch Press, 2011.
4. "ARP Cache Poisoning Detection and Prevention", San Jose University, 2003.
5. 2007 [Online] Available:http://www.harmonysecurity.com/files/HS-P004_ARPPoisoning.pdf
6. 2014 [Online]. Available:http://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-switches/white_paper_c11_603839.html.
7. B. Kitchenham, O. Pearl Brereton, D. Budgen, M. Turner, J. Bailey and S. Linkman, "Systematic literature reviews in software engineering – A systematic literature review", *Information and Software Technology*, vol. 51, no. 1, pp. 7-15, 2009.
8. M. Tripunitara and P. Dutta, "A middleware approach to asynchronous and backward compatible detection and prevention of ARP cache poisoning", Proceedings 15th Annual Computer Security Applications Conference (ACSAC'99).
9. D. Bruschi, A. Ornaghi and E. Rosti, "S-ARP: a secure address resolution protocol", 19th Annual Computer Security Applications Conference, 2003. Proceedings.
10. L. Biju Issac, "Secure Unicast Address Resolution Protocol (S-UARP) by Extending DHCP", 2005 13th IEEE International Conference on Networks Jointly held with the 2005 IEEE 7th Malaysia International Conf on Communic.
11. Z. Trabelsi and W. El-Hajj, "Preventing ARP Attacks Using a Fuzzy-Based Stateful ARP Cache", 2007 IEEE International Conference on Communications, 2007.
12. W. Lootah, W. Enck and P. McDaniel, "TARP: Ticket-based address resolution protocol", Computer Networks, vol. 51, no. 15, pp. 4322-4337, 2007.
13. A. Ortega, X. Marcos, L. Chiang and C. Abad, "Preventing ARP cache poisoning attacks: A proof of concept using OpenWrt", 2009 Latin American Network Operations and Management Symposium, 2009.
14. S. Hammouda and Z. Trabelsi, "An enhanced secure ARP protocol and LAN switch for preveting ARP based attacks", Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing Connecting the World Wirelessly - IWCMC '09, 2009.
15. S. Puangpronpitag and N. Masusai, "An efficient and feasible solution to ARP Spoof problem", 2009 6th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, 2009
16. S. Nam, D. Kim and J. Kim, "Enhanced ARP: preventing ARP poisoning-based man-in-the-middle attacks", *IEEE Communications Letters*, vol. 14, no. 2, pp. 187-189, 2010.
17. S. Dangol, S. Selvakumar and M. Brindha, "Genuine ARP (GARP)", *SIGSOFT Softw. Eng. Notes*, vol. 36, no. 4, p. 1, 2011.

18. S. Kumar and S. Tapaswi, "A centralized detection and prevention technique against ARP poisoning", Proceedings Title: 2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensic (CyberSec), 2012.
19. S. Nam, S. Jurayev, S. Kim, K. Choi and G. Choi, "Mitigating ARP poisoning-based man-in-the-middle attacks in wired or wireless LAN", EURASIP J Wirel Commun Netw, vol. 2012, no. 1, p. 89, 2012.
20. H. Salim, Z. Li, H. Tu and Z. Guo, "Preventing ARP Spoofing Attacks through Gratuitous Decision Packet", 2012 11th International Symposium on Distributed Computing and Applications to Business, Engineering & Science, 2012.
21. M. Ataullah and N. Chauhan, "ES-ARP: An efficient and secure Address Resolution Protocol", 2012 IEEE Students' Conference on Electrical, Electronics and Computer Science, 2012.
22. N. Tripathi and B. Mehtre, "An ICMP based secondary cache approach for the detection and prevention of ARP poisoning", 2013 IEEE International Conference on Computational Intelligence and Computing Research, 2013.
23. S. Nam, S. Djuraev and M. Park, "Collaborative approach to mitigating ARP poisoning-based Man-in-the-Middle attacks", Computer Networks, vol. 57, no. 18, pp. 3866-3884, 2013.
24. P. Pandey, "Prevention of ARP spoofing: A probe packet based technique", 2013 3rd IEEE International Advance Computing Conference (IACC), 2013.
25. M. Masoud, Y. Jaradat and I. Jannoud, "On preventing ARP poisoning attack utilizing Software Defined Network (SDN) paradigm", 2015 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT), 2015.
26. J. Tian, K. Butler, P. McDaniel and P. Krishnaswamy, "Securing ARP From the Ground Up", Proceedings of the 5th ACM Conference on Data and Application Security and Privacy - CODASPY '15, 2015.
27. N. Saputro and K. Akkaya, "PARP-S: A secure piggybacking-based ARP for IEEE 802.11s-based Smart Grid AMI networks", *Computer Communications*, vol. 58, pp. 16-28, 2015.
28. B. Bakhache and R. Rostom, "Kerberos secured Address Resolution Protocol (KARP)", 2015 Fifth International Conference on Digital Information and Communication Technology and its Applications (DICTAP), 2015.
29. P. Arote and K. Arya, "Detection and Prevention against ARP Poisoning Attack Using Modified ICMP and Voting", *2015 International Conference on Computational Intelligence and Networks*, 2015.
30. C. Wilmer and C. Barski, *Bitcoin for the Befuddled*. No Starch Press, 2015.
31. Alam M, Khan AH, Khan IR (2016). Swarm intelligence in MANETS: a survey. Int J Emerg Res Manag Technol 5(5):141–150. DOI: 10.6084/m9.figshare.14309384.
32. Alam, M. (2011). Online Banking (1st ed.). New Delhi. LAP LAMBERT Academic Publishing, ISBN: 978-620-3-86302-4, DOI: 10.6084/m9.figshare.14612127.
33. Alam, M. (2012). Electronic Ticket Machine (1st ed.). New Delhi. LAP LAMBERT Academic Publishing, ISBN: 978-620-3-86332-1, DOI: 10.6084/m9.figshare.14661354.
34. Alam, M. (2013). Just Shop-Shopping (1st ed.). New Delhi. LAP LAMBERT Academic Publishing, ISBN: 978-620-3-58124-9, DOI: 10.6084/m9.figshare.14431382.
35. Alam, M. (2013). Core ePortal (1st ed.). New Delhi. Glasstree Bookstore, ISBN: 978-1-6671-9827-9, DOI: 10.20850/9781667198279.

36. Alam, M., & Khan, M. (2013). E-Cops (1st ed.). New Delhi. LAP LAMBERT Academic Publishing, ISBN: 978-620-3-86368-0, DOI: 10.6084/m9.figshare.14662479.
37. Alam, M. (2014). Stegnography (1st ed.). New Delhi. LAP LAMBERT Academic Publishing, ISBN: 978-620-3-86944-6, DOI: 10.6084/m9.figshare.14662680.
38. Alam, M. (2016). Applicability of Swarm Intelligence in Mobile Ad Hoc Network (1st ed.). New Delhi. LAP LAMBERT Academic Publishing, ISBN: 978-620-3-57426-5, DOI: 10.6084/m9.figshare.14313548.
39. Alam, M., & Ahmed, I. (2017). Payroll Management System (1st ed.). New Delhi. LAP LAMBERT Academic Publishing, ISBN: 978-620-3-86260-7, DOI: 10.6084/m9.figshare.14662860.

40.

41.

42.