

A Survey on ARP Cache Poisoning And techniques for detection and mitigation

Jitta Sai Meghana
Department of Electronics
Engineering
Madras Institute of Technology
Anna University, Chennai, India
Email: saimeghanajitta@gmail.com

Dr. T. Subashri
Assistant Professor, Department of
Electronics Engineering
Madras Institute of Technology
Anna University, Chennai, India
Email: tsubashri@annauniv.edu

K.R. Vimal
Senior Technical Assistant (SG),
Department of Electronics
Engineering
Madras Institute of Technology
Anna University, Chennai, India
Email: vimalkrme@gmail.com

Abstract—Network security has become a major concern in any industry. Address resolution protocol (ARP) is a basic protocol used by all hosts in a network. When ARP is compromised, it leads to several security attacks. In this survey, we present a deep investigation of existing ARP cache poisoning or ARP spoofing solutions. We first introduce Address resolution protocol and its operation. We then dwell on ARP spoofing, with an emphasis on its impact on network security. Then we provide an overview on existing solutions to detect and mitigate ARP spoofing in both traditional networking and software-defined networking (SDN) platforms. Finally, we compare the techniques discussed and conclude that SDN based solutions are more effective in detecting and eliminating any kind of ARP spoofing attack.

Keywords—Address resolution protocol, ARP, ARP cache poisoning, attacks, security, Software-defined networks, SDN

I. INTRODUCTION

In today's era, internet has become a basic necessity with digitalization of many systems. Organizations, Government, educational institutions, small offices need an efficient network in order to function well. Such networks contain sensitive, confidential information which attracts the attackers. If network security is ignored, an attacker can do all sort of attacks to steal the data. Research groups and organizations are developing variety of solutions for network security. In this paper we focus on ARP spoofing, a security attack which leads to several attacks such as Denial of service (DOS), Man in the Middle (MITM) attack etc. Various researchers have proposed solutions to mitigate this problem. We conduct an extensive survey on techniques proposed by various researchers. We also focus on the Software-defined networks (SDN), an emerging network technology and discuss how SDN can help in mitigating ARP attacks.

II. ADDRESS RESOLUTION PROTOCOL

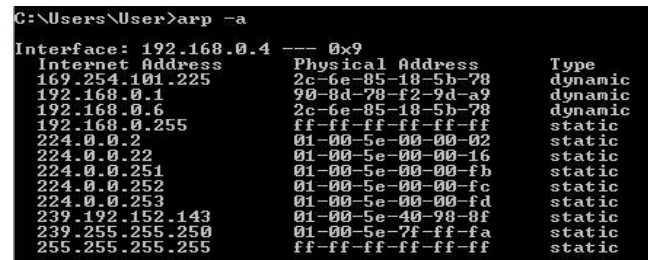
Every device on an IP network has both MAC and IP addresses. When a node needs to send data to another node, it needs both these addresses of the destination device. The IP address of the destination node is provided by some protocols of higher OSI layers. But to find the MAC address, we need Address resolution protocol [1]. Thus ARP is an OSI layer two protocol which helps to find a MAC address with IP

address known. It also maintains ARP table with IP-MAC mappings.

A. ARP cache and it's maintenance

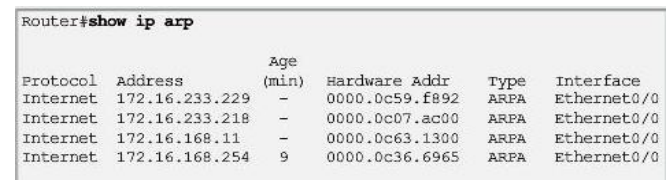
It is a table stored in the RAM of a device. Each row in ARP cache binds an IP address to a MAC address. Entry of an ARP cache expires after certain duration if the device doesn't Receive any frame from the device in the entry within that duration.

Fig. 1 shows an example of ARP cache in windows 8.1 PC displayed on command prompt. The command "arp -a" is used to display ARP cache on command prompt of windows operating system. Fig.2 shows an example of ARP cache in a Cisco router displayed on Cisco packet tracer using Cisco IOS command "show ip arp".



```
C:\Users\User>arp -a
Interface: 192.168.0.4 --- 0x9
Internet Address      Physical Address      Type
169.254.101.225       2c-6e-85-18-5b-78    dynamic
192.168.0.1           90-8d-78-f2-9d-a9    dynamic
192.168.0.6           2c-6e-85-18-5b-78    dynamic
192.168.0.255         ff-ff-ff-ff-ff-ff    static
224.0.0.2             01-00-5e-00-00-02    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.251           01-00-5e-00-00-fb    static
224.0.0.252           01-00-5e-00-00-fc    static
224.0.0.253           01-00-5e-00-00-fd    static
239.192.152.143       01-00-5e-40-98-8f    static
239.255.255.250       01-00-5e-7f-ff-fa    static
255.255.255.255       ff-ff-ff-ff-ff-ff    static
```

Fig. 1. Example of ARP cache in windows 8.1 PC



```
Router#show ip arp
Protocol Address      Age (min)  Hardware Addr  Type   Interface
Internet 172.16.233.229 -          0000.0c59.f892 ARPA   Ethernet0/0
Internet 172.16.233.218 -          0000.0c07.ac00 ARPA   Ethernet0/0
Internet 172.16.168.11  -          0000.0c63.1300 ARPA   Ethernet0/0
Internet 172.16.168.254 9          0000.0c36.6965 ARPA   Ethernet0/0
```

Fig. 2. Example of ARP cache on a Cisco router

ARP cache is maintained dynamically in two ways.

- Traffic monitoring: When a node receives any frame from another node, it creates an entry in its ARP cache with the source MAC and IP present in the frame.
- ARP request/reply packets: A node makes use of ARP request packet to request MAC address

corresponding to a particular IP address and uses ARP reply packet to record the requested MAC address in its ARP cache. Fig. 3 shows the format of an ARP request/reply frame on Ethernet.

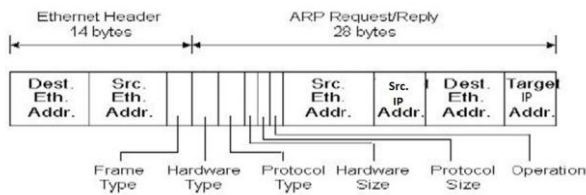


Fig. 3. Format of ARP request/reply frame on Ethernet.

B. ARP operation

When a node wants to send data to another node, it needs to create a frame which requires MAC addresses of both the source and the destination. It checks its ARP cache to find the destination MAC address. If no entry is found with the desired MAC address, it broadcasts ARP request to all the nodes in the LAN with the destination MAC address field in the ARP request as FFFF.FFFF.FFFF. All the nodes in the LAN receive ARP request and check whether the IP address in the ARP request match with their own IP addresses. The node whose IP address matches will send ARP reply to the requested node. Fig. 4, 5 and 6 demonstrate ARP process.

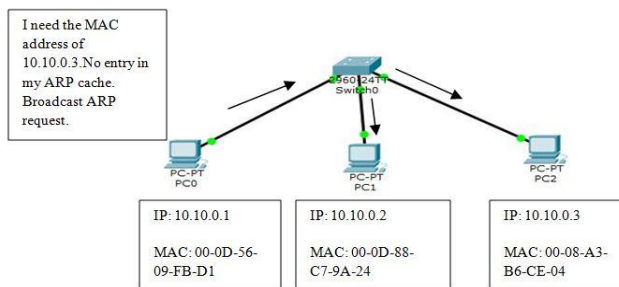


Fig. 4. Broadcast ARP request to all the nodes in the LAN.

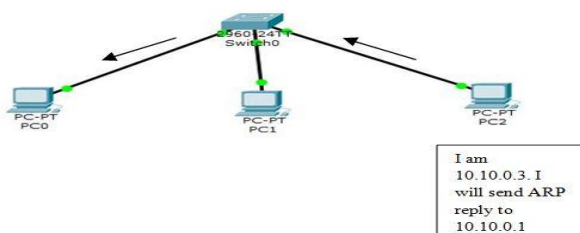


Fig. 5. Destination node replies its MAC address to the requested node.

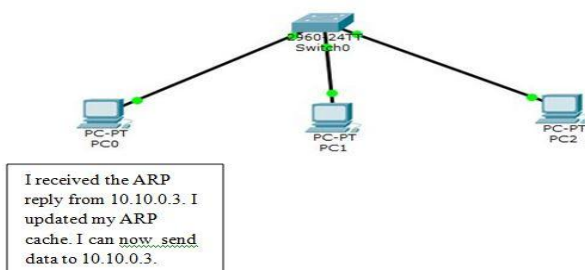


Fig. 6. The requested node updates its ARP cache.

III. ARP CACHE POISONING

ARP cache poisoning or ARP spoofing is a technique used by an attacker to poison the ARP tables of hosts in a LAN by injecting wrong IP-MAC address mapping.

ARP spoofing is often used as a form of following attacks:

- 1) Denial of service (DOS) attack: If the ARP cache of a host is poisoned with a wrong IP-MAC mapping, the host being attacked cannot reach the real destination. If the MAC address of the default gateway is a fake one in the ARP cache, the host cannot reach the internet thus resulting in the Denial of Service (DOS) to the host being attacked as illustrated in Fig. 7.

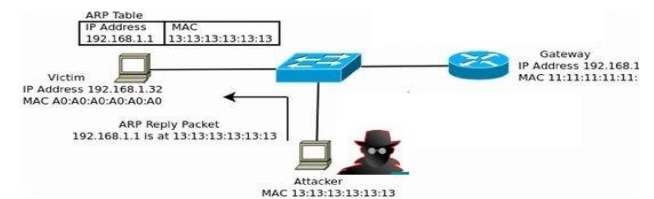


Fig. 7. Denial of service attack

- 2) Host Impersonation: Instead of just dropping the packets received from the host being attacked, the attacker can reply. Thus, the attacker can impersonate any host in the network.
- 3) Man in the middle (MITM) attack: An attacker can spoof two hosts in a network at same time as shown in the Fig. 8. By doing so, the attacker can silently listen to the traffic between two hosts. In this way the attacker can gain access to confidential data or he/she can modify the data being sent.

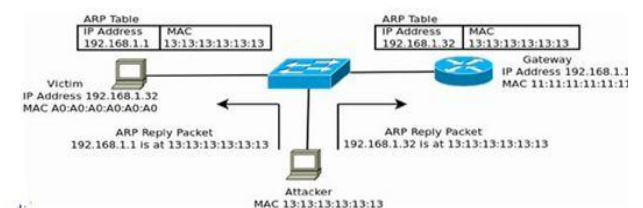


Fig. 8. Man in the middle attack.

Tripunitara *et al.* identified the ways in which ARP table can be poisoned [2] and those are as follows:

- 1) Unsolicited Response: A response that is not associated with any request is accepted by ARP process. An attacker only has to send an ARP response with a fake mapping to poison the ARP cache of the host being attacked. This response can be broadcasted to poison ARP tables of all the hosts in the LAN.

- 2) Request: When a host broadcasts ARP request, all the other hosts in the LAN record the IP, MAC addresses of the sender host in their ARP tables. By pretending to be sending a legitimate ARP request, the attacker can poison the ARP tables of all the hosts in the LAN.
- 3) Response to a request: Instead of sending unsolicited request or response, the attacker waits till a request is received and sends a fake reply to the requested host. The requested host receives one reply from the legitimate host and one from the attacker. Since a host accepts the latest reply received, there may be a chance that attacker wins.
- 4) Request and response: A malicious host can send not only a fake request to all the hosts in the LAN but also a fake reply to any legitimate request received thus poisoning ARP caches of many hosts in the LAN.

IV. STATELESS NATURE OF ARP

ARP cache poisoning by unsolicited response is possible due to the fact that ARP is a stateless protocol. Network hosts will automatically cache any ARP replies they receive, regardless of whether network hosts requested them. Even ARP entries which have not yet expired will be overwritten when a new ARP reply packet is received.

V. SOLUTIONS TO MITIGATE ARP SPOOFING

A. Static ARP

One basic approach to eliminate ARP spoofing is to configure static ARP mappings on each host in the LAN. Legitimate MAC addresses can be configured on the network devices to restrict network access to only those devices listed. The major disadvantages of this scheme are:

- When network is large it becomes very difficult for the network administrator to configure it manually on each device.
- It doesn't work in a dynamic environment (e.g. when DHCP is enabled in the LAN).
- It does not scale well, as it would be very cumbersome for the network administrator to deploy and update these tables throughout the network.

B. Stateful ARP

Zouheir Trabelsi *et al.* proposed a Stateful ARP cache [3] and a fuzzy logic controller.

Stateful ARP tries to make ARP secure by addressing its basic vulnerability 'statelessness'. Each host maintains separate queues for ARP requests and ARP replies. When a reply is received, the host checks whether there is any request associated with it. If there is no request corresponding to the reply, the host assumes it as the one sent by the attacker and doesn't update its cache. If source host receives two similar ARP reply packets that look like coming from destination host, the two packets will be discarded by fuzzy logic controller and host will not update its ARP cache. The major drawbacks of this scheme are:

- Changing of stateless ARP to stateful ARP requires modification of the existing protocol. This makes ARP more complex since it is designed to be a simple protocol.
- Cannot detect request based attacks.

Tripunitara *et al.* proposed a similar procedure [2] which maintains separate queues for ARP requests sent and replies received.

C. Secure ARP (S-ARP)

D.Bruschi *et al.* proposed a Secure Address Resolution Protocol (S-ARP) [4]. In this scheme each host has a public and private key pair distributed by Authoritative Key Distributor (AKD) which acts as certification authority. AKD also distributes clock value to all the hosts with which all the hosts must synchronize. Messages are digitally signed by the sender thus preventing ARP cache from being poisoned.

D. T-ARP

Lootah *et al.* proposed Ticket based ARP [5]. TARP provides security by distributing attestations generated by a Local Ticket Agent (LTA). These attestations, called tickets, also include an expiration time. The host, whose IP address matches with the one in the ARP request, sends a reply by attaching previously obtained ticket. Upon receiving the reply, the host sees the signature of LTA on the ticket and considers it as valid.

The drawbacks of the above two schemes are:

- Addition of cryptographic features in ARP requires more computational resources due to signature generation, verification and key management.
- Need for up-gradation of network stacks of all the hosts in the LAN, because all hosts need to understand the ARP with cryptographic features.

Sumit kumar *et al.* proposed a centralized system and ARP Central Server (ACS) to manage ARP table entries in all hosts [6]. The ARP cache of ACS has IP-MAC bindings for all the nodes in the LAN. The ARP reply sent by ACS is stored in a secondary long term cache and replies from other hosts are stored in main ARP cache. Entry for a particular IP-MAC mapping in main ARP cache is checked against the secondary long term cache and if the entry doesn't match, the host can broadcast an ARP request. The major drawbacks of this scheme are:

- Non compatibility with the existing ARP since each node needs to maintain two ARP caches.
- Maintenance of a dedicated server (ACS).
- ACS becomes single point of failure because if ACS is compromised, the entire system fails. We may need to configure the ARP cache of ACS manually in order to be more efficient. But it would be cumbersome for the network administrator and also fails in a dynamic environment (when DHCP is enabled).

RaviyaRupal *et al.* proposed a utility which provides authentication to the users as well as detects and prevents ARP poisoning in dynamic IP configuration [7]. The proposed utility provides a mechanism which is based on internet control management protocol (ICMP) which uses a secondary cache for checking IP-MAC pairs of hosts in the network. The major drawbacks of this scheme are:

- Need for up-gradation of network stacks of all the hosts to support cryptographic features.
- Non compatibility with the existing ARP process since each host needs to maintain two ARP tables.
- Need more computational resources.

E. Active detection techniques

These are the techniques which are compatible with the existing ARP. That is, these techniques do not need any modification to the existing protocol.

Yeob Nam *et al.* proposed a voting based mechanism [8]. In this scheme when a host A receives an ARP reply with MAC details of the desired host B, it broadcasts a voting request with IP address of host B as target address. It calculates a polling score for each MAC address received based on early N voting replies obtained. If a MAC has got more than 0.5N votes, that MAC is accepted as MAC of host B. The drawbacks of this scheme are:

- Voting replies from all the hosts in the network generates significant amount of traffic resulting in congestion.
- There can be a situation where an attacker sends N voting replies with a fake MAC pretending as many hosts.

Rahmani *et al.* proposed a technique which functions in three phases [9]. In the first phase, the test host creates an ARP request packet with a fake source IP address and a fake source MAC address. It then sends this packet with a fake broadcast address FF-FF-FF-FF-FF-FE. A legitimate host will drop this packet due to incorrect broadcast address. But the host running in the promiscuous mode will accept this packet and makes an entry in its ARP cache. In the second phase, the test host attempts to establish a TCP connection by sending a TCP SYN packet to all the hosts. In the third phase, the test host tries to find the attacker. If it receives a TCP SYN/ACK packet, the test host assumes it as the attacker because the legitimate hosts send ARP request packets to the test host since they do not have an entry for the fake MAC address used by the test host.

Gao Jinhua *et al.* proposed an active detection scheme which uses Internet Control Message Protocol (ICMP) echo request and echo reply packets [10]. When a host sends an ARP packet, it sends an ICMP echo request to the device with the IP-MAC pair as the one in the ARP packet received and finds the attacker by analysing the ICMP echo reply received.

Biswas *et al.* proposed a similar method called Discrete Events Systems (DES) [11] which makes use of probe packets to distinguish between normal and abnormal conditions.

Sukumar Nandi *et al.* proposed an active detection technique which makes use of a spoof detection engine [12]. In this scheme, TCP SYN and TCP SYN/ACK packets are used instead of ICMP echo request and reply packets used by Gao Jinhua *et al.*

The techniques proposed by Gao Jinhua, Biswas and Sukumar Nandi *et al.* can mitigate the ARP spoofing when a weak attacker is present in the LAN. Their techniques cannot prevent strong attackers from poisoning the ARP cache of a host.

Pandey *et al.* identified three types of attackers [13]. He categorized the attackers as follows:

- 1) Weak attacker: A weak attacker can generate fake ARP packets but he/she doesn't have a compromised protocol stack.
- 2) Intermediate attacker: An intermediate attacker can generate fake ARP packets and he/she can modify his/her own system's protocol stack such that it can generate a response to any request received.
- 3) Strong attacker: A strong attacker is the one who targets a specific host. He/she can have a customized protocol stack which makes him/her indistinguishable from the legitimate host by detection techniques.

Pandey *et al.* further enhanced the ICMP based technique proposed by Gao Jinhua *et al.* in order to prevent all types of attackers from creating ARP problems.

The major drawbacks of the aforementioned active detection schemes are:

- Need for the installation of an additional software on all the hosts in the LAN
- High implementation costs.
- For a single ARP request received, all the hosts in the LAN perform detection process which result in additional network overheads.

Cisco developed Dynamic ARP inspection (DAI)[14] for Cisco catalyst 6500 switches. It is a security feature that validates the ARP packets. The switch maintains IP-MAC address bindings in a trusted database. The switch compares the IP-MAC mappings in the ARP packets with the IP-MAC mappings in the database in order to detect ARP spoofing attacks in the network. The switch forwards an ARP packet only upon validation.

Cristina L. Abad gave requirements [15] for an ideal detection technique and those are as follows:

1. Should be backward compatible with existing ARP process.
2. Should not require installation of an additional software on each hosts as this incurs additional costs.
3. Should consume low network resources.
4. Should be easy to deploy.
5. Should be capable of detecting all possible ARP attacks.
6. Should prevent all types of ARP attacks.
7. Shouldn't slow the ARP process significantly.

VI. SOLUTIONS TO MITIGATE ARP SPOOFING BY LEVERAGING SDN

A. Introduction to Software-defined networks (SDN)

SDN is an emerging technology which separates the control plane from the data plane. It is very dynamic, easily manageable and cost effective which make it ideal for the deployment of a dynamic network. The heart of the SDN technology is the SDN controller which takes care of the control plane. It is basically a server where network professionals can directly program the control plane according to the application [16]. The data plane also called forwarding plane, is handled by the underlying network devices like switches, routers etc. The architecture of SDN is shown in the Fig. 9.

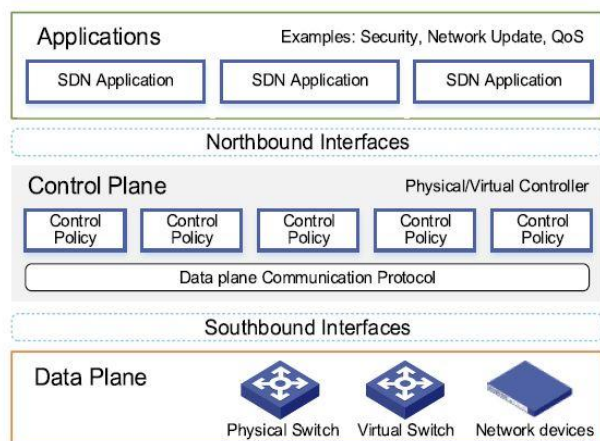


Fig. 9. Architecture of SDN

Ability to collect network status of SDN allows better analysis of traffic for potential security threats. Attacks such as low rate burst attacks, distributed denial of service (DDOS) attacks can be detected just by analyzing traffic patterns [17]. Since SDN offers programmatic control over forwarding plane, traffic of interest can be directed to the Intrusion detection system for deep packet inspection. If attacks are detected, SDN can apply forwarding rules to underlying switching or routing devices to block the traffic from entering the network.

Because of these features of SDN, an SDN based ARP detection techniques are more advantageous than others as shown in the table.1.

TABLE I. Comparison of various detection techniques

Requirement	Static arp	Stateful arp	Crypto-graphic solution	Active solution	SDN solution
1	Yes	No	No	Yes	Yes
2	Yes	No	No	No	Yes
3	Yes	No	No	No	Yes
4	No	No	No	No	Yes
5	Yes	No	Yes	Yes	Yes

6	Yes	No	Yes	Yes	Yes
7	Yes	No	No	No	Yes

B. SDN based solutions to detect and mitigate ARP spoofing

Wanqing *et al.* proposed ARP spoofing defence application [18] in SDN. It manages a buffer which stores the IP-MAC address pairs of all the network devices in the network. It also collects every ARP reply and compares the IP-MAC pair in the reply packet with the IP-MAC pairs in the buffer. If any two pairs have same IP address but different MAC addresses, the application prompts an alarm. Otherwise, the new IP-MAC pair is added to the buffer. This method is not effective in mitigating ARP spoofing attacks since the new IP-MAC pair is not verified.

Alharbi *et al.* proposed SDN based technique [19] for detecting ARP spoofing. In this scheme, ARP packets are forwarded via SDN controller where the controller uses safe dummy IP, mac address values which are not present on the local LAN. This approach is effective in eliminating ARP request based attacks. A few drawbacks of this scheme are:

- For ARP reply based attacks, this approach doesn't provide efficient solution. It just prevents the host from communicating to the destination host until a valid ARP reply is received without proper analysis of the replies received.
- Care must be taken to ensure that the dummy values used in the scheme are not present on the local LAN. This requires the maintenance of a database of all the hosts in the LAN with MAC details which is tedious when network is large and scalable.

Huan Ma *et al.* [20] proposed a Bayes-based algorithm to detect ARP cache poisoning in SDN-based cloud centres. He defined the basic features of an attacker. The algorithm calculates the probability of a host being an attacker by analyzing the ARP packets and based on the probability value calculated, the SDN controller takes decision whether to forward the packet or discard it.

Ashraf *et al.* [21] proposed a technique in SDN which identifies the conditions to distinguish fake ARP packets from the legitimate packets. ARP packets are forwarded to the controller and if the packet matches any of the stated conditions, the packet is dropped. If the attacker knows that the ARP packets are sent to the controller, he may flood the controller with several ARP packets to perform DOS attack. So the proposed mechanism sets a threshold on each port of the switch for ARP packets. If ARP count exceeds the threshold on a switch port, the packets are stopped on that port for a particular amount of time. This method prevents attack packets from entering the network and also protects the SDN controller.

Masoud *et al.* [22] proposed a technique in SDN which consists of two scenarios differing by how IP assignment is done to the hosts. The first scenario works in DHCP environment and the second scenario works in static

environment. A table of IP-MAC mappings is maintained in the SDN controller. In the first scenario, the table is populated by analyzing DHCP replies from a DHCP server whereas in the second scenario, the table is loaded manually with static IP-MAC entries.

Jacob *et al.* proposed an SDN based technique which makes use of a Network Flow Guard (NFG) module [23] in SDN controller. NFG provides security by analyzing DHCP and ARP packets. The controller maintains a dynamic table with IP:MAC:port:state details. When a DHCP packet is received, the NFG updates the dynamic table. ARP request and reply from any host are sent to the controller. The controller validates the ARP packets against its dynamic table and prevent attacker to poison the ARP cache of the requested host with his/her fake reply. This method can detect any type of attacker and prevent the attacker from poisoning the ARP cache of the hosts thus satisfying all the requirements in the table. 1.

VII. CONCLUSION

In this paper, we provided an overview of ARP and how ARP cache can be poisoned in various methods by an attacker. We investigated several ARP cache poisoning detection and mitigation techniques proposed by various authors. We also gave an introduction to SDN and discussed how leveraging SDN helps in effective mitigation of ARP spoofing attacks.

REFERENCES

- [1] D.C.Plummer, "An Ethernet address resolution protocol", Rfc 826,InterNet Network Working Group, 1982.
- [2] Mahesh V. Tripunitara and Partha Dutta, "A Middleware Approach to Asynchronous and Backward Compatible Detection and Elimination of ARP Cache Poisoning," in Computer Security Applications Conference, (ACSA99) Proceedings, 15th Annual, Phoenix, UK, pp. 303-309, 1999.
- [3] Zouheir Trabelsi and Wassim El-Hajji, "Preventing ARP attacks using a Fuzzy-based Stateful ARP Cache," in IEEE Conference on Communications (ICC) Proceedings,2007.
- [4] D.Bruschi, A. Ornaghi and E.Rosti, "S-Arp, A secure Address Resolution Protocol," in Computers Society Applications Conference, Proceedings, 19th Annual, IEEE, pp. 66-74, 2003.
- [5] W.Lootah, W.Enck and P. Mc Daniel, "Tarp:Ticket based address resolution protocol," Computer networks, Vol.51, no. 15, pp. 4322-4337,2007.
- [6] S. Kumar and S. Tapaswi, "A centralized detection and prevention technique against arp poisoning," in Cyber Security, Cyber Warfare and Digital Forensic (CyberSec), International Conference in, Kuala Lumpur, Malaysia, pp. 259-264, 2012.
- [7]RaviyaRupal.D,DhavalSatasiya, H.Kumar, A.Agrawal, "Detection and Prevention of ARP Poisoning in Dynamic IP configuration", IEEE International Conference On Recent Trends In Electronics Information Communication Technology in, India, May 20-21, 2016.
- [8] S.Y. Nam, D.Kim and J.Kim, "Enhanced arp:Preventing arp poisoning-based man-in-the-middle attacks," Communications Letters, IEEE, vol.14, no. 2, pp.187-189, 2010.
- [9] Zouheir Trabelsi and Hamza Rahmani, "Detection of Sniffers in an Ethernet network," in ISC 2004, pp.170-182,2004.
- [10] J.Gao and K.Xia, "Arp spoofing detection algorithm using ICMP protocol," in Computer Communication and Informatics (ICCCI), 2013 International Conference in Coimbatore, India, pp. 1-6, 2013.
- [11] F.H.Barbhuiyah, S.Hubballi, S.Biswas and S.Nandi, "A host based DES approach for detecting arp spoofing," in Control and Automation (MED), 2010 18th Mediterranean Conference in Marrakech, Morocco, pp.695-700, 2010.
- [12] V.Ramachandran and S.Nandi, "Detecting arp spoofing: An active technique," in Information Systems Security, in Springer , pp.239-250, 2005.
- [13] P.Pandey, "Prevention of arp spoofing:A probe packet based technique," in Advance Computing Conference (IACC), 2013 IEEE 3rd international, Ghaziabad, India, pp. 147-153, 2013.
- [14] Cisco Systems, "Configuring Dynamic ARP Inspection in Catalyst 6500 Series Switch Cisco IOS Software Configuration Guide, Release 12.2SX", ch. 39, pp. 39:1-39:22, 2006.
- [15] C.L.Abad and R.Bonilla, "An analysis on the schemes for detecting and preventing ARP cache poisoning attacks," in Distributed Computing Systems Workshops, ICDCSW07, 27th International Conference in Toronto, Canada, p.60, 2007.
- [16]F.A.Lopes,M.Santos,R.Fidalgo,S.Fernandes, "A software engineering perspective on SDN programmability", in IEEE communications surveys & tutorials, Vol. 18, No. 2, second quarter 2016.
- [17] W. Xia, Y. Wen, C.H. Foh, D. Niyato and H. Xie, "A survey on Software defined networks," IEEE Communications Surveys and Tutorials, Vol.17, issue:1, FirstQuarter,2015.
- [18] W.You, K.Qian, Xi He, Y.Qian,"Int. J. Advanced Networking and Applications", in ISSN :0975-0290,Vol. 6 Issue: 3,pp. 2347-2351,2014.
- [19] T.Alharbi, D.Durando, F.Pakzad and M.Portmann, "Securing ARP in software defined networks", IEEE 41st Conference on Local Computer Networks, 2016.
- [20] Huan Ma, H.Ding, Y.Yang, Z.Mi, J.Y.Yang, and Z.Xion, "Bayes-Based ARP Attack Detection Algorithm for Cloud Centers", Tsinghua Science and Technology, in ISSN 1007-0214 02/10,Vol. 21, No.1, pp.17-28, in February 2016.
- [21] A.M.Abdelsalam, A.el-Sisi, Vamshi reddy, "Mitigating arp spoofing attacks in software-defined networks",in ICCTA ,at alexandria,Egypt,2015.
- [22] M.J. Masoud, Y.Zaradat and I.Jannoud, "On preventing ARP poisoning attack utilizing software defined network paradigm," in Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT),IEEE 2015.
- [23] J.H.Cox, R.J.Clark and H.L.Owen, "Leveraging SDN for ARP security," in Southeast Conference, IEEE, 2016.