Security Analysis of Address Resolution Protocol and Its Exploitable Vulnerabilities

Ritik Kalal[1]*,Harsh Mandaliya[2],Dhruv Patel[3],Deep Solanki[4],Jigar Gajjar[5]

Lok Jagruti Kendra University, Ahmedabad GJIndia[1, 2, 3]
TechDefenceCybersecurity Limited, Ahmedabad GJ India[4,5]

ritikkalal1121@gmail.com[1], harshmandaliya4114@gmail.com[2], pateldhruv190725@gmail.com[3], deepsolanki5799@gmail.com[4], cyberian.jg@gmail.com[5]

Abstract: "Address Resolution Protocol" (ARP) is a simple network protocol that enables devices on a network to discover each other based on their assigned IP addresses and their actual physical addresses referred to as MAC addresses. ARP is a fundamental protocol under IPv4 that has laid a solid foundation however, it was never designed with security considerations. Consequent to its lack of security, there is a chance of various security attacks such as ARP spoofing, man-in-the-middle attack, and denial of service attacks. These security attacks may result in data theft, data modification, and denial of services. In the paper, various issues regarding ARP, its security flaws, research on various kinds of attacks associated with ARP, and prevention are discussed. For instance, various security protocols such as IPv6 Neighbor Discovery, Software Defined Network, and many more have become alternatives to ARP.

Keywords:Address Resolution Protocol (ARP), ARP Cache Poisoning, Network Security Vulnerabilities, Man-in-the-Middle Attacks, Denial-of-Service Attacks, ARP Defense Mechanisms, IPv6 Neighbor Discovery, Software-Defined Networking (SDN).

I. INTRODUCTION

A. ARP and its Need

Modern networks separate physical and logical addressing (MAC and IP respectively), enabling wide-area routing and local link delivery. The Address Resolution Protocol (ARP) enables the dynamic mapping of IP addresses to their corresponding MAC addresses thus, ensuring delivery of data frames to the correct hardware destination on networks such as Ethernet (RFC 826) [9]. A device wishing to communicate with another device locally will send a broadcast ARP request to the network, asking "Who has this IP address?" The device that possesses the matching IP address will send a unicast reply containing its MAC address back to the requester, who will cache this information for future use, in an efficient manner [2].

The automation of this process eliminates the need for manual configuration, works well in dynamic environments, and provides an easy way to plug-and-play in LAN environments. ARP is necessary, as all higher-layer protocols will assume IP-level abstraction (logical addressing), but will send physical delivery to the specific MAC address of the hardware being communicated with without ARP, devices would have no means to determine where to send data locally, nor would they be able to do so reliably, if static mappings of IP address to MAC address were used [5] **.** ARP uses a broadcast method of operation, as this was the simplest and easiest to implement in early trusted networks [6] .

ARP was designed for trusted local networks, so it prioritizes simplicity over authentication. Because the protocol accepts address mappings with little verification, an attacker on the same segment can exploit that trust by sending forged replies and redirect traffic to an unintended host. This design choice was acceptable in early LAN environments, but it becomes a serious

http://doi.org/10.63766/spujstmr.26.000095

limitation in modern networks where mobility, virtualization, and untrusted endpoints are common [5], [6], [9].

B. Different attacks possible on ARP and its impact

The ARP "trust model" that allows for unsolicited ARP replies to be accepted without verification, enables "cache poisoning", where an attacker can forge an ARP reply associating their MAC address with a legitimate IP address such as a default gateway [2]. Cache poisoning is part of a larger "man in the middle" (MITM) attack mechanism that facilitates methods to eavesdrop, hijack a session or modify traffic [13]. Gratuitous ARP floods can further exacerbate the impact of cache poisoning by allowing multiple caches to be poisoned quickly [11]. Attackers will take advantage of similar methods in SDN "contexts" and can disrupt a controller at the same time as expanding their reach [7]. The DoS variations can thin valid IP traffic to non-existent MACs (resulting in "blackholing" packets) or generate broadcast storms, overwhelming a switch and CPU [10].

The impacts to the CIA triad are substantial: confidentiality is diminished by means of obtaining a victim's credentials or stealing sensitive data traversing the wire integrity is compromised through attacking or modifying an arbitrary packet (e.g., adding malicious code) and availability is eliminated through DoS, resulting in service outages to critical systems such as medical or financial networks [10]. There are also risks of escalation such as entering a network with ransom ware or other broader breaches, which demonstrate ARP to be a persistent point of entry into legacy infrastructures [11].

TABLE I

ATTACKS POSSIBLE ON ARP

Attack Name	What the Attacker Does	Main Effect / What Happens to the Network
ARP Spoofing / Poisoning	Sends fake ARP messages saying "My MAC is for this IP"	Victim's device sends traffic to the attacker instead
Man-in-the-Middle (MITM)	Uses spoofing to sit between two devices	Attacker can spy on, steal, or change messages secretly
Denial-of-Service (DoS)	Sends fake mappings or floods with bad ARP messages	Network slows down, packets get lost, devices can't connect
Session Hijacking	Grabs login cookies/sessions during MITM	Attacker takes over your logged-in account
Gratuitous ARP Flood	Sends lots of fake "I'm this IP" messages to everyone	Many devices get wrong addresses → network confusion/DoS

C. Prevention and mitigation strategies to secure ARP

Mitigating ARP's lack of verification requires additional layers of verification. Static ARP entries enforce manual locks on the IP-MAC bindings of users, which can help to mitigate against spoofed ARP entries, however, they are not suited for dynamic environments [6]. Dynamic ARP Inspection (DAI) on switches will inspect packets against DHCP-snooped ARP entries to ensure packets are valid and drop the invalid ones [12]. In SDN, centralized policy layers can leverage adaptive algorithms, such as game theory, to anticipate the actions of an attacker and optimize verification [7].

Cryptographic mechanisms, such as S-ARP, use signatures to allow the server to verify that it is responding to the correct client when sending a response [4]. Other protocols, such as ticket-based protocols (e.g., TARP), require that the server validate the token before responding [6]. Network segmentation through the use of VLANs can shrink the size of broadcast domains that can carry attacks [4]. Intrusion Detection Systems (IDS) can monitor for ARP/spoofed attacks (e.g., MAC

flapping) and allow for rapid response through Anomaly Detection [10]. Combining host-level filtering (e.g., ignore unsolicited ARP Requests) with the aforementioned entries provides a multi-layered defense and this approach towards a verified (zero-trust) local resolution will provide usable access [12].

A solid strategy for defense, therefore, has to incorporate not just protection against ARP spoofing, but also feasibility. Static ARP is a good choice, but it needs regular maintenance, which can be very demanding in a dynamic network. Dynamic ARP Inspection further secures enterprise switches when properly implemented along with DHCP snooping and sufficient hardware. However, the cryptographic approach of S-ARP enhances security but involves higher overheads. Conversely, SDN-based approaches offer centralized management at the cost of increased complexity[4], [6], [7], [12] .

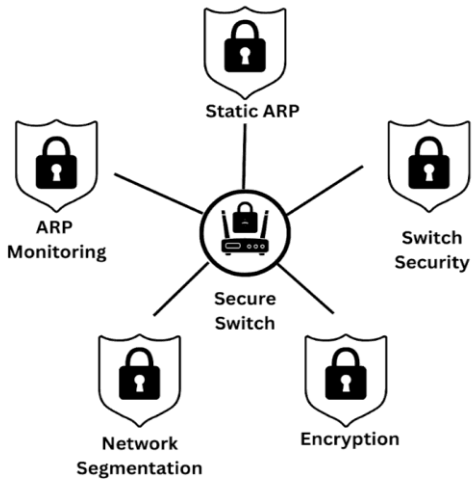

Fig.1 Prevention and mitigation strategies to secure ARP

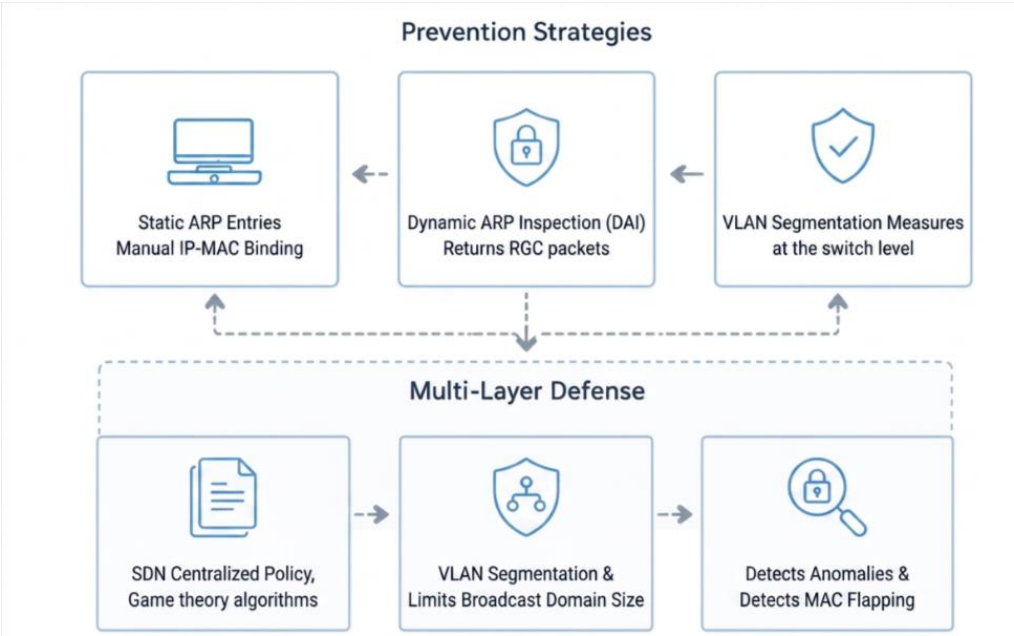

Fig.2 Prevention and Mitigation Strategies to Secure ARP Protocol

D. Limitations of ARP protocol and its alternatives

ARP uses broadcast flooding to waste bandwidth and create storm conditions in large environments or data centers [5]. ARP has no security built-in and is easily spoofed without additional external measures, also limited to IPv4 networks [12]. The scalability of ARP breaks down when you add mobility and virtualization.

The Neighbor Discovery Protocol (NDP) in IPv6 replaces ARP with more efficient multicast and offers Secure Neighbor Discovery (SEND) to provide cryptographic validation. SDN solutions use proxy ARP via a controller that queries a centralized database to avoid broadcast flooding while improving security [12]. All of these alternatives are consistent with the modern-day's need for an authenticated and scalable method of address resolution.

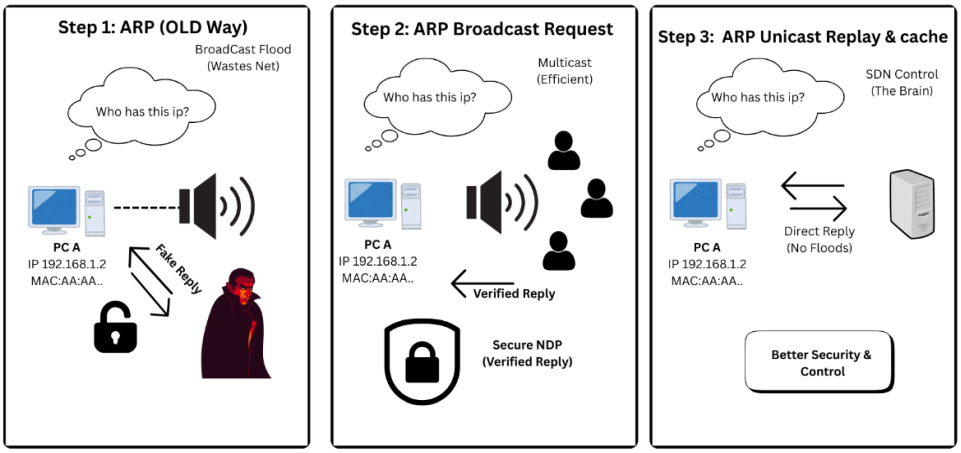

Fig.3 ARP Limitations vs Modern Solutions

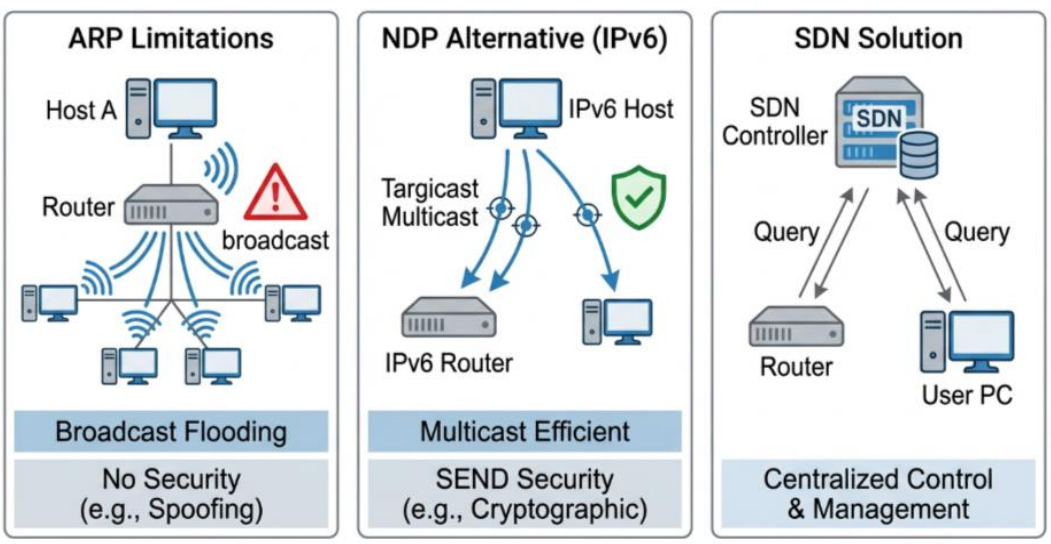

Fig.4 Comparison of Network Address Resolution and Security Approaches

II. LITERATURE SURVEY

ARP has been around since 1982 it’s an early TCP/IP protocol designed to provide mapping between logical (IP) and physical (MAC) addresses on a shared local area network (like Ethernet) [9]. David C Plummer authored RFC 826 defining it as a simple request/reply mechanism using broadcasts, which allowed ARP to dynamically resolve address pairs without having to create

manual mappings. By the end of the 1980's and into the 1990's, the number of LANs were growing rapidly, and ARP was able to accommodate this growth [15]. Over the years, ARP has only received a few minor updates with some RFCs adding enhancements to how ARP detects address conflicts. However, the functionality of ARP itself has remained the same: broadcast-based and unauthenticated. While these characteristics of ARP work well in small trusted networks, the evolution of new types of threats has revealed the weaknesses of ARP and caused researchers to begin looking for ways to:

1. Securely extend ARP
2. Replace ARP with an IPv6 (Internet Protocol Version 6) alternative.

A. Security Challenges

1) ARP Spoofing / Cache Poisoning

On a local network, an attacker can leverage a vulnerability in ARP to poison the ARP cache of the victim's system by broadcasting fake ARP replies that map the attacker's MAC address to the IP address of a legitimate device, such as the default gateway. Because ARP is a stateless protocol, the victim's device blindly accepts these updates, causing their traffic to be redirected to the attacker[11].

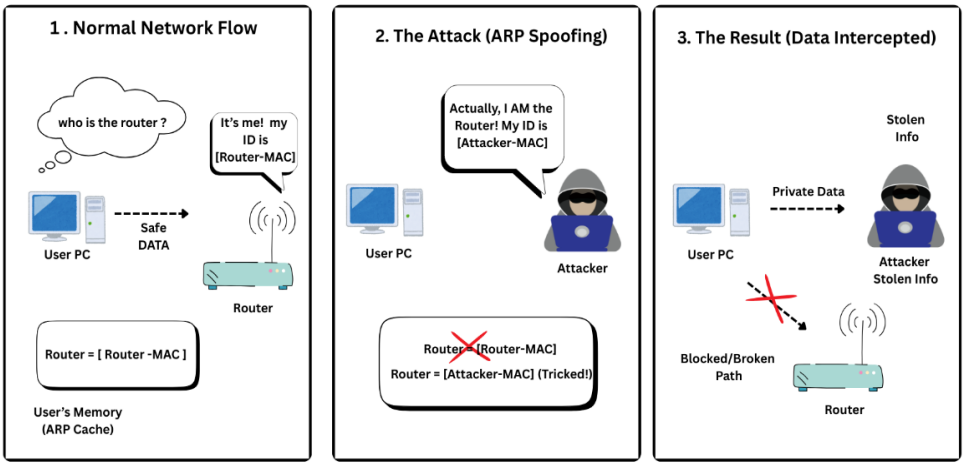

Fig.5 How ARP Spoofing Works

2) Man-in-the-Middle

The attacker (MITM) places himself in between two endpoints by taking over the source and destination's ARP caches. By doing this, they will intercept, inspect, or alter the contents of packets sent between those two endpoints (e.g., stealing credentials or injecting malware) while still passing packets (traffic) between the endpoints as if they had not been altered or modified [7].

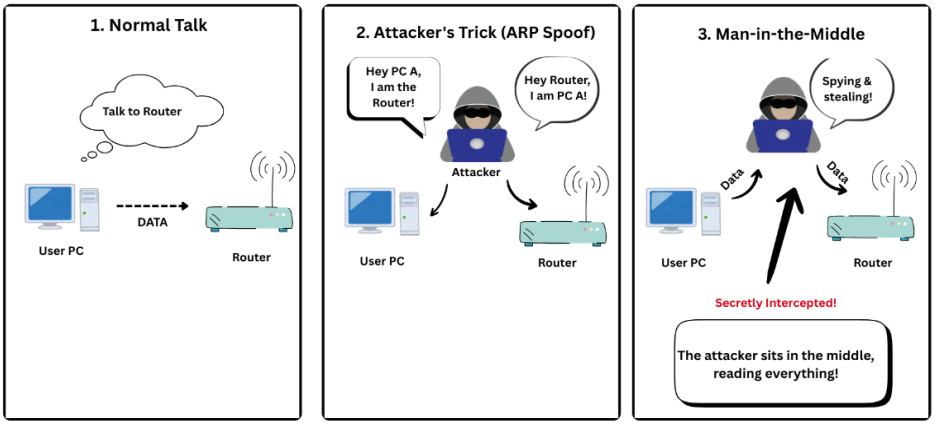

Fig. 6 Man-in-the-Middle Attack

3) Denial-of-Service (DoS) Attack

Attackers may produce a high number of ARP request/reply pairs or link valid IP addresses (that correspond to a real machine) to non-existent MAC addresses with the intent of creating broadcast storms to deplete both bandwidth resources and CPU resources, or to create packet blackholing which cause a loss of connectivity [12].

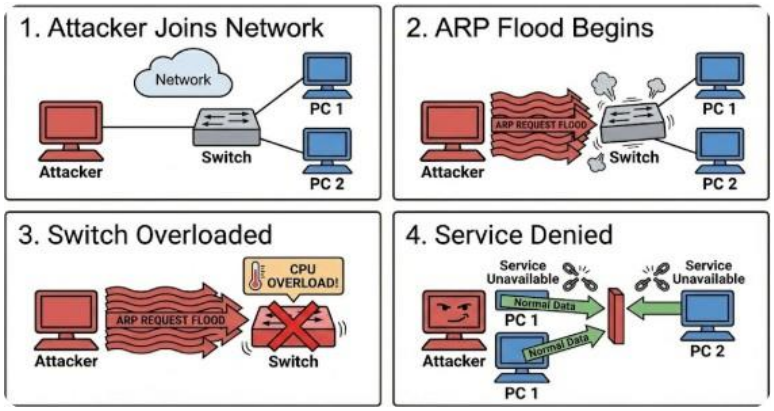

Fig.7. Denial-of-Service (DoS) Attack

B. Limitations of ARP

1) Broadcast Flooding

Broadcasts for address resolution queries made by ARP produce excessive broadcasts across networks which do not scale well. For example, large LANs or data centres can have a high rate of broadcasts leading to possible congestion from broadcast storms which could lead to an overall performance hit [14].

2) No Built-in Authentication

The ARP protocol has no method for confirming the authenticity of a reply or the identity of the sender, which allows virtually any device to create fake replies and fill invalid entries in any ARP cache much too easily [4].

3) IPv4-Specific and Poor Scalability

The Address Resolution Protocol (ARP) was developed primarily for Internet Protocol Version 4 (IPv4). ARP has problems in dynamic, virtualized, and/or highly mobile networks because of the broadcast way in which it sends information, how it manages its cache, and how it cannot support the transition to Internet Protocol Version 6 (IPv6) [6].

C. Alternatives to ARP

1) Neighbor Discovery Protocol (NDP) in IPv6

Instead of ARP, NDP uses ICMPv6 messages to resolve addresses in an IPv6 environment using targeted multicast instead of flooding the whole segment with broadcasts. By reducing unnecessary network traffic, this approach improves efficiency for larger networks. SEND, or Secure Neighbor Discovery, extends NDP with cryptography to help prevent spoofing of local address mappings, making them more secure and dynamic to current configurations. [8] [3].

2) Proxy ARP / Centralized Resolution in SDN

With SDN, address resolution is centralized. Instead of letting devices broadcast their address queries everywhere throughout a network, the controller (proxy) retrieves the address mappings from its comprehensive view of the network and responds directly to the querying device with that information. This reduces the amount of broadcast overhead (traffic), enables rapid policy application, and allows for better containment of potential threats in virtual (or dynamic) network infrastructures [12][1].

D. Gap

Though there are papers which have mentioned approaches that could detect or prevent ARP attack, yet only a handful of papers mention the practicality of such approaches in terms of administrative burden, backward compatibility issues, and computational overhead. The reason for this is that sometimes an approach may be effective yet still remain impractical. Consequently, an approach's feasibility should be evaluated both from security efficiency and implementability perspectives[2], [7], [12], [15] .

III. ANALYSIS

A further assessment of the strengths of ARP and the solutions available should focus not just on the security benefits of a possible solution, but also on how realistically feasible the deployment of such solutions may be in practical situations. Some solutions may provide better security measures but at the same time, be more complex, whereas other solutions may be simpler but fail to completely protect networks from spoofing attacks and poisoned information.

TABLE II

ANALYSIS TABLE: ARP, ITS PROBLEMS, AND SOLUTIONS

Feature	ARP (IPv4)	ARP Attacks	Security Solutions	Better Alternatives
What it does	Helps a device find the MAC address of an IP on a local network	Takes advantage of ARP to send fake IP-MAC information	Tries to prevent or reduce fake ARP messages	Uses newer and safer ways instead of ARP
How it works	Sends a broadcast message asking who owns a certain IP	Sends fake ARP replies even when no request is made	Verifies ARP messages before trusting them	Relies on multicast, encryption, or centralized control

Security	Has no built-in security at all	Very easy for attackers to exploit	Improves security using rules or cryptographic methods	Security is designed into the protocol itself
Trust model	Trusts every ARP reply it receives	Attackers misuse this blind trust	Only trusted or verified replies are accepted	Replies are checked before being accepted
Main attacks	—	ARP spoofing, Man-in-the-Middle, DoS	Detects or blocks these attacks	Prevents such attacks by design
Confidentiality	Does not protect data	Data can be intercepted or stolen	IDS and traffic monitoring tools help	Communication is encrypted and authenticated
Integrity	Does not check if data is altered	Packets can be changed by attackers	Static bindings reduce the risk	Strong identity and message verification is used
Availability	Works fine in small networks	Can slow down or even shut down networks	Rate limiting and VLANs improve stability	Less flooding and better network reliability
Network traffic	Generates a lot of broadcast traffic	Attackers can greatly increase traffic	Broadcast traffic is controlled	Very little unnecessary traffic
Scalability	Does not scale well in large networks	Problems get worse as the network grows	Some improvement, but still limited	Designed to work well in large networks
IPv6 support	Not supported	—	—	Fully supported
Examples	ARP cache	ARP poisoning	Dynamic ARP Inspection (DAI), IDS, S-ARP	IPv6 NDP, SEND, SDN, Proxy ARP

Source: Based on RFC 826 and prior studies on ARP security, spoofing attacks, mitigation techniques, and secure alternatives [1], [2], [4], [7], [10], [11].

The comparison table in the following provides a structured analysis of the functionalities of the Address Resolution Protocol, its vulnerabilities in terms of security, mitigation strategies, and modern alternatives. ARP originally was defined in RFC 826 [1], introducing IP-to-MAC address mapping into IPv4 local area networks without any kind of built-in authentication. This design weakness turns networks into potential targets of various malicious activities: poisoning of the ARP cache and MITM attacks [2], [4], and an inability to ensure DoS conditions that implicate network availability [10]. The table further refers to the following common countermeasures that were typically implemented: dynamic ARP inspection, intrusion detection systems, and cryptographic extensions such as Secure ARP (S-ARP) [7]. In turn, it outlines the differences of ARP from more secure and scalable modern alternatives, among which are NDP - Neighbor Discovery Protocol upon shifting to IPv6 [11] - and a centralized resolution upon utilizing SDNs [10]. Overall, the table summarizes some operational limitations of the protocol and underlines the urgent need for enhanced security mechanisms in contemporary environments.

From an analytical point of view, this means that ARP-related security cannot be fully addressed through a single universal defense. A practical and secure approach should combine preventive controls, detection mechanisms, and, where possible, migration toward more secure resolution models that reduce broadcast dependence and strengthen authenticity.

IV. CONCLUSION

ARP has always played an important role as a protocol for communication between devices in local network environments. However, since this protocol does not have any built-in security features, it is considered unsafe for use in modern network environments. The trust-based feature of this protocol allows attackers to easily compromise this protocol for malicious activities, thereby creating significant risks for all aspects of computer security confidentiality, integrity, and availability.

However, several security controls such as Dynamic ARP Inspection, IDS, and cryptography have been developed to address ARP-related threats. Yet such controls cannot thoroughly address the issue since modern network technologies have introduced alternative concepts for secure network communications such as IPv6 Neighbor Discovery protocol along with its enhanced security features and SDN's centralized address resolution techniques.

The research presented is deliberately a survey-based approach rather than an experiment performed on an active testbed. Future research may expand the analysis to include the simulation of ARP spoofing attacks in order to measure how well each method detects the attacks, how much latency overhead and how expensive its implementation is for small networks, enterprises, and software-defined networking.

REFERENCES

[1] Alharbi, T., Aljuhani, A., & Alharbi, S. (2021). Scalable and secure SDN based ethernet architecture by suppress-ing broadcast traffic. Egyptian Informatics Journal, 22(4), 453-462.

[2] Al Sukkar, G., Saifan, R., Khwaldeh, S., Maqableh, M., & Jafar, I. (2016). Address Resolution Protocol (ARP): Spoofing Attack and Proposal for a Defense Mechanism. Com-munications and Network, 8(3), 118-130.

[3] Arkko, J., Kempf, J., Zill, B., & Nikander, P. (2005). Secure Neighbor Discovery (SEND). RFC 3971.

[4] Bruschi, D., Ornaghi, A., & Rosti, E. (2003). S-ARP: an address resolution protocol that is secure. Proceedings of the 19th Annual Computer Security Applications Conference.

[5] Gouda, M.G. & Huang, C.T. (2003). A Secure Address Resolution Protocol. Computer Networks, 41(1), 57-71.

[6] Lootah, W., Enck, W., & McDaniel, P. (2007). TARP: A Ticket-Based Address Resolu-tion Protocol. Computer Networks.

[7] Mvah, F., Tchendji, V.K., Djamegni, C.T., Anwar, A.H., Tosh, D.K., &Kamhoua, C. (2024). Countering ARP Spoofing Attacks in Software-Defined Networks Using a Game-Theoretic Approach. Computers & Security, 139, 103696.

[8] Narten, T., Nordmark, E., Simpson, W., & Soliman, H. (2007). Neighbor Discovery for IP version 6 (IPv6). RFC 4861.

[9] Plummer, D.C. (1982). A Protocol for Resolving Ethernet Addresses to a Network Pro-tocol Address to Transmit on Ethernet Interfaces. RFC 826.

[10] Prabadevi, B. & Jeyanthi, N. (2017). An Analysis of Proposed Solutions to ARP Poi-soning Attacks and Their Impacts on Medical Computing. International Journal of Medical Engineering and Informatics.

[11] Salim, H., Li, Z., Tu, H., & Guo, Z. (2012). Prevention of ARP Spoofing via Gratuitous Decision Packet. 2012 11th International Symposium on Distributed Computing and Appli-cations to Business, Engineering & Science.

[12] Shah, Z. & Cosgrove, S. (2019). A survey on how to address the ARP cache poisoning problem in Software-Defined Networking. IEEE Access / Electronics, 8(10), 1095.

[13] Song, M.S., Lee, J.D., Jeong, Y.S., Jeong, H.Y., & Park, J.H. (2014). A New Detection Scheme for ARP Spoof-ing Attacks Based on Routing Trace for Ubiquitous Environments. The Scientific World Journal.

[14] Tripathi, N. &Mehtre, B. M. (2013). Secondary Cache Approach based on ICMP for Detection and Prevention of ARP Poisoning. Proceedings of the IEEE International Confer-ence on Computational Intelligence and Computing Re-search.

[15] Tripathi, N. &Mehtre, B. M. (2014). A Survey of the Various Techniques for Mitigat-ing ARP Poisoning Attacks. International Workshop on Control, Instrumentation, Commu-nication and Computational Technologies.