Enhancing Network Security: Mitigating ARP Spoofing Threats through Advanced Prevention Strategies

Himaneesh Yadala, Vaibhav Anurag, Preetha Sarkar

Department of Computer Science and Engineering
National Institute of Technology Karnataka
Surathkal, Mangalore, India

9739955000, 79799 15189, 6292253051

himaneeshyadala.221cs264@nitk.edu.in, anuragvaibhav.221cs258@nitk.edu.in,
preethasarkar.221cs236@nitk.edu.in

March 18, 2024

1.Abstract

A. Background of the problem statement

What is ARP:
Address Resolution Protocol (ARP) is a crucial component of computer networking, functioning at the Data Link Layer (Layer 2) within the OSI model. Its main role involves associating an IP address with the corresponding Media Access Control (MAC) address within a local network.
Working of ARP:
When a device "A" within a network wishes to communicate with another device "B" in the same network, it requires the MAC address of Device B.
To obtain this MAC address:
Device A first checks its ARP cache, which is essentially a local table storing IP-MAC address mappings, to see if it already possesses the MAC address associated with the destination IP (the IP address of Device B).
If the MAC address is not found in the ARP cache, Device A broadcasts an ARP request packet to the network, seeking the MAC address associated with the IP of Device B. Device B responds to this ARP request by sending an ARP reply packet containing its MAC address back to Device A.
Upon receiving this reply, Device A stores the MAC address of Device B in its ARP cache for future references. ARP caches have a timeout mechanism to ensure that outdated information is periodically refreshed.

B. Challenges and issues of the problem statement

Every device in a network maintains an ARP cache table to store IP-to-MAC address mappings.The ARP cache entries have a timeout after which the source device has to send an ARP request again to resume communication with the destination device. Each time an ARP request is sent the host device blindly cache the ARP replies which makes it prone to an attack known as ARP spoofing attack. During ARP spoofing attacks, the attackers make their MAC address become associated with the IP address of the target device. Consequently, communication occurs between the victim's device and the attackers' system. This enables the attackers to intercept and alter sensitive data.

C. Existing approaches or methods and their issues

Preventing ARP spoofing involves implementing various measures to safeguard against malicious manipulation of the Address Resolution Protocol. Here are some basic approaches:

1) Static ARP Entries: These involve maintaining an ARP table where IP addresses are explicitly mapped to MAC addresses. This method ensures that the ARP cache remains unchanged by unauthorized modifications.

However, managing these ARP entries requires manual configuration, which can become an hassle, especially in extensive networks.

2) ARP Cache Timeout Management; this involves adjusting the duration for which ARP cache entries are valid on a network device. However, shorter timeouts can lead to increased ARP traffic, affecting network performance and longer timeouts, provide a larger window for potential attacks.

3) Network Segmentation; This involves dividing a network into segments to limit the potential impact of security incidents. However, if weak and inadequate access controls can allow unauthorized users to exploit vulnerabilities within segments, compromising the overall security of the network.

D. Your problem statement

Our proposed methodology involves using ICMP probing and Dynamic ARP Inspection to actively monitor ARP packets. DAI uses a DHCP snooping database which contains all the valid MAC addresses for every IP address assigned to each device in the network.The IP addresses mapping is done with the help of a DHCP server present on the network.
Another such method is Dynamic Arp Inspection (DAI). DAI inspects ARP packets on LAN in real time and checks for inconsistent IP-to-MAC mapping. However, it cannot detect the attacks generated outside the protected segment.
A brief description of the process involves periodically sending ICMP echo requests to all hosts on the network, analyzing ICMP response patterns (e.g., response times, packet behavior) to create a baseline of normal behavior for each host, configuring DAI to dynamically update its database according to the ICMP request and response behaviour.
Additional security features will be added to DAI. For example: If anomalies are detected DAI can validate the ARP packet or drop the ARP packet if the IP address in the packet is invalid.
In the data link layer, a cyber-criminal can attack hosts, switches, and routers by "poisoning" their ARP cache tables. Methods that prevent ARP spoofing already exist. Nevertheless, each has its limitations.

E. Objectives of the proposed work

1. Enhanced security: We combine DAI's proactive filtering with ICMP's detection and verification to provide layered protection.

2. Improved network visibility: ICMP tools like ping and traceroute can provide valuable insight into network connectivity and identify potential inconsistencies caused by spoofing attempts.

3. Alerting and Troubleshooting: ICMP error messages generated by DAI enabled switches can quickly alert about potential spoofing and facilitate faster troubleshooting.

2.Introduction

A. Background of the problem statement

Address Resolution Protocol (ARP [1]) is a crucial component of computer networking, functioning at the Data Link Layer (Layer 2) within the OSI model. It's used extensively in computer networks to map Internet Protocol (IP) addresses to it's corresponding Media Access Control (MAC [2]) addresses.
Hosts maintain an ARP cache, sometimes called an ARP table, which is a mapping table between IP addresses and MAC addresses, in order to prevent making the same request in the future. The ARP cache can contain both static and dynamic entries. Dynamic entries are retained for a predetermined amount of time, while static entries are retained until the system is restarted.

When a device "A" within a network wishes to communicate with another device "B" in the same network, it requires the MAC address of Device B. To obtain this MAC address:
Device A first checks its ARP cache to see if it already possesses the MAC address associated with the destination IP (the IP address of Device B).
If Device A's address reference (ARP cache) doesn't list Device B's info (MAC address), Device A sends out a message (ARP request packet) on the local channel (network) asking for whoever has that info (associated with Device B's IP address). Device B, recognizing its own information in the message (ARP request), sends a reply back to Device A providing its MAC address. After receiving this response, Device A stores Device B's contact info in its address book for future calls. These address books have built-in timeout mechanisms to

prompt updates and ensure the information refreshes.

ARP enables communication among devices within the same network. Network devices like routers, switches, and firewalls all rely on ARP to ensure smooth data flow between devices. However, ARP is vulnerable to various attacks, such as ARP spoofing, which can pose a threat to the security and reliability of the network. Therefore, implementing security measures, such as employing network access controls and utilizing tools for detecting ARP cache poisoning, is essential to safeguard against such attacks.

B. Challenges and issues of the problem statement

Every device in a network maintains an ARP cache table to store IP-to-MAC address mappings.The ARP cache entries have a timeout after which the source device has to send an ARP request again to resume communication with the destination device. This prompts the originating host to send a fresh ARP request when attempting to communicate with destination hosts whose MAC addresses are no longer stored. However, with each ARP request, the protocol establishes a new mapping between an IP address and a MAC address, disregarding any previous associations. This vulnerability can be exploited, and is as called ARP Spoofing Attack [3], wherein an attacker sends falsified ARP responses, associating their own MAC address with the IP address of another device on the network.

In a Spoofing attack, the attacker sends a deceptive packet containing an IP to MAC mapping. Within this packet, the IP address is legitimate (belonging to the victim), while the MAC address belongs to the attacker. Consequently, the host caches an incorrect address mapping, leading messages intended for the genuine receiver to be redirected to the attacker instead. This allows the attacker to intercept and manipulate network traffic, allowing access to sensitive information.

ARP spoofing presents a significant security concern, leading to potential attacks like **Man-in-the-Middle (MitM)**. Other network attacks can take advantage of weaknesses in the ARP protocol.
In a Man-in-the-Middle (MitM [4]) attack, the attacker intercepts the communication between two parties, granting them access to either listen in on or alter the transmitted data. Through ARP message spoofing, attackers deceive network devices into directing their traffic through the attacker's system, thus enabling them to intercept, modify, or impersonate as part of the communication between the legitimate parties. These attacks can cause major financial harm and harm an organization's reputation. Therefore, it's crucial to put strong security measures in place to stop them.

C. Existing approaches or methods and their issues

Several methods have been suggested to counter ARP spoofing, including Static ARP entries, Secure ARP, Network Segmentation and ARP Cache Timeout Management.

Static ARP entries [5] are a method of maintaining an ARP table by manually configuring mappings between IP addresses and MAC addresses. Unlike dynamic ARP entries that are automatically populated through ARP requests and responses, static entries remain unchanged until manually updated or removed [6]. This approach provides a level of security by preventing unauthorized modifications to the ARP cache. However, this method lacks scalability and requires considerable time and effort to manage entries. Additionally, it becomes ineffective when new mobile hosts connect to the network.

Secure-ARP [7] as a response to the vulnerabilities exposed by ARP spoofing, it implements a novel approach: it authenticates ARP messages using the Digital Signature Algorithm (DSA) [8]. This safeguard significantly enhances network security.
Only devices with valid signing keys can send authenticated messages, making it much harder for attackers to spoof identities.

Network segmentation [9] refers to the process of splitting a network into smaller, separate sections or sub-networks. This division helps enhance both security and performance by isolating various parts of the network. Through segmentation, organizations can restrict the extent of potential security incidents and regulate the movement of data between segments.
However, this methods limits the flexibility and agility of networks, especially those in dynamic environments where devices frequently keep changing and moving roles. Not only is this technique complicated to implement in large-scale networks, but in some segmentation architectures, the devices or mechanisms responsible for enforcing segmentation policies, such as routers or firewalls, can become single points of failure. A failure or misconfiguration in these devices could potentially compromise the security of the entire network.

Unequal Request-Reply Algorithm: [10] This involves maintaining counters for every port of a switch. One counter increment whenever an ARP request is sent and other increases whenever ARP reply is received. We find the difference between the two to check if the number of replies recieved is balanced with the number of requests sent. If the number of replies is greater than the number of requests sent, then there is ARP attack going on.But it becomes difficult to maintain the counters in large networks.

D. Your problem statement

Our proposed algorithm uses ICMP probing and Dynamic ARP Inspection to actively monitor ARP packets. DAI uses a DHCP snooping database which contains all the valid MAC addresses for every IP address assigned to each device in the network.The IP addresses mapping is done with the help of a DHCP server present on the network. Hence if a device has IP address = X and MAC address = Y then the pair {X,Y} is stored in the database as a valid IP-MAC pair. This is also known as IP-MAC binding.
ICMP probing involves sending ICMP echo request packet(also known as ICMP ping) to a network device.The device sends ICMP echo reply packets in response. ICMP ping command can be implemented in CLI to determine if a device is reachable or not. The Round Trip Time(RTT) of a data packet is one useful information provided by ICMP ping to detect abnormalities in the network.The use of RTT is explained in detail in our propsed algorithm.
The entries in the database help us determine if an ARP packet has been spoofed or not.If a particular IP address contained in the ARP packet is not found in the database(a new IP address) its validity is checked through ICMP probing.

E. Objectives of the proposed work

1. We combine DAI's proactive filtering with ICMP's detection and verification to identify and actively block potential attackers.We verify if the suspect is actually attacking using ICMP and block it from accessing the network using DAI.

2. We use ICMP Ping periodically to determine if a device is reachable and how long it takes for a data packet to reach that device. If the time taken is too long, we check the device's ARP table to make sure it is not being spoofed.

3. We compare every ARP packet using DAI enabled switches to make sure that inconsistencies are dealt with.

3.Detailed Literature Review

3.1: Secure ARP:

This method operates by employing a public key infrastructure (PKI) to authenticate devices within the network. When a device seeks to communicate with another device, it transmits an S-ARP request comprising its MAC and IP addresses, alongside a digital signature generated using its private key. Upon receiving the request, the recipient device utilizes the sender's public key to validate the signature, ensuring the sender's claimed identity.
The granular access control facilitated by S-ARP's digital signatures allows for the selective validation of device identities. This selective validation serves to build trust amongst authorized devices and effectively hinders unauthorized devices, consequently strengthening network security.
As per our understanding, traditional ARP lacks authentication, making it vulnerable to attackers forging messages and impersonating devices. Hence, S-ARP addresses this by adding digital signatures to ARP messages. Devices with valid signing keys can send authenticated messages, verifiable by others. Adding this authentication layer benefits overall network security, making it significantly harder for attackers to exploit ARP vulnerabilities and access sensitive data.

Advantages: S-ARP utilizes digital signatures like DSA to authenticate ARP messages, ensuring only authorized devices can communicate. This significantly reduces the success rate of spoofing attacks compared to traditional, trust-based ARP.
Disadvantages: Compared to standard ARP, S-ARP's reliance on a public key infrastructure (PKI) and a trusted third-party server introduces increased complexity and requires additional infrastructure. This can

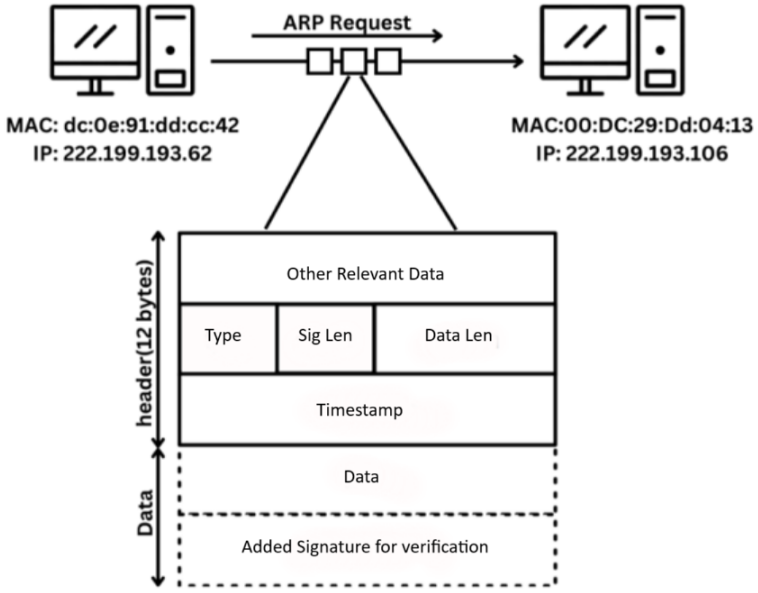

Figure 1: Working of Secure ARP

translate to higher costs and greater difficulty in managing and administering the network.

3.2 Unequal Request Reply Algorithm:

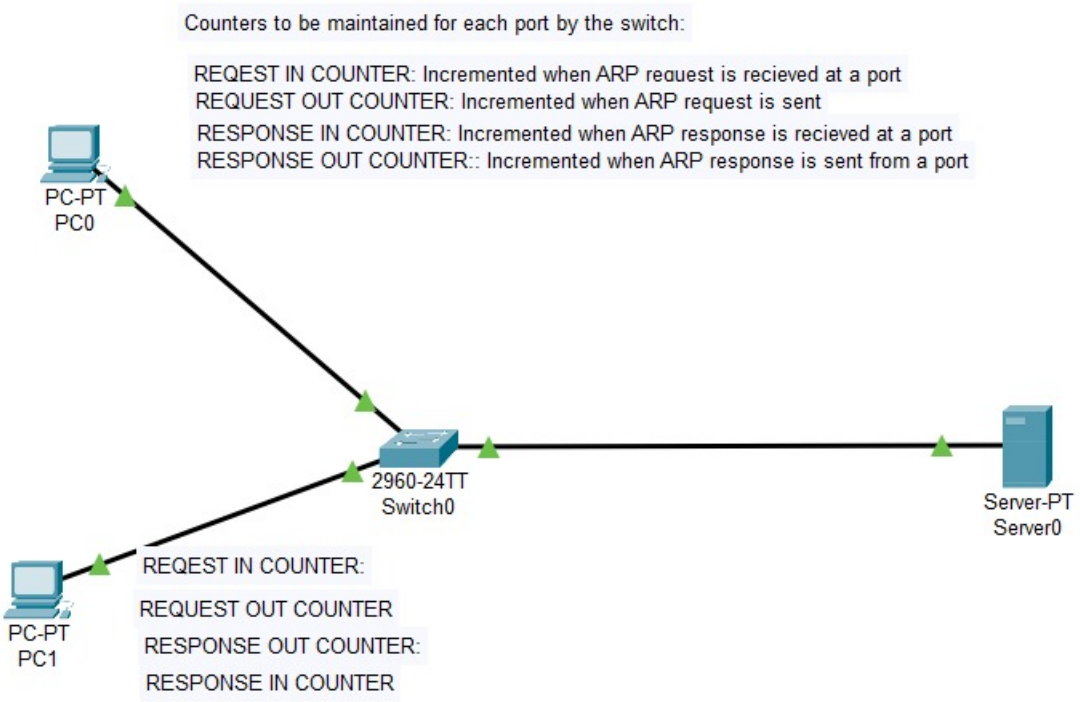

Figure 2: Working of Unequal Request Reply Algorithm

In this algorithm, the switch maintains a counter for ARP packets. The switch is supposed to maintain four counters.
Counter 1: gets appended whenever an ARP request is received at the port.
Counter 2: gets appended whenever ARP request is transmitted out.

Counter 3: gets appended whenever ARP reply is received.
Counter 4: gets appended whenever ARP reply is going out.
For a time interval, we will calculate delta values of the counter 2 and counter 3. This means we are checking the imbalance will be the difference between the deltas of counter 2 and counter 3.
Let the imbalance of the ARP request and replies at a port p be im[p]. Now, if the imbalance is positive, then the number of requests sent out of that particular time interval is greater than the replies received, which is alright. But when the imbalance becomes negative, it means that ARP replies are greater than ARP requests. This directly indicates that an ARP attack is going on.
im[p] = delta(counter2)-delta(counter3)
Since the Switch maintains counters for the ports, we are able to pinpoint the port where the attack is going on.

Advantages: It is easy to implement. It gives the exact port location where ARP spoof is happening.
Disadvantages: Maintaining a counter and analyzing it for every port for every ARP request and response adds significant overhead.
Creates privacy issues because the network usage pattern of a user is being stored.

3.3 Static MAC Entries:

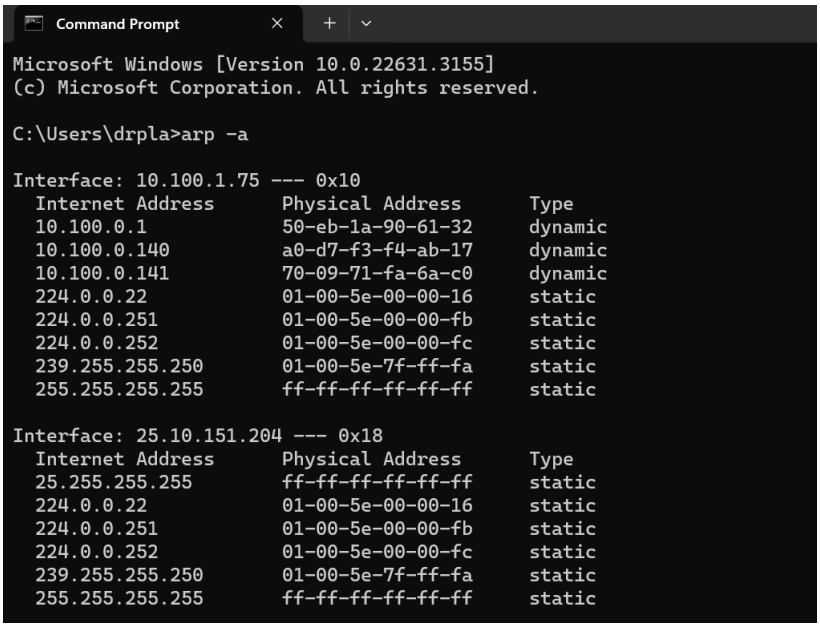

Figure 3: Static MAC Entries on a Network

Instead of relying on dynamic lookups, network administrators can create fixed entries within the ARP cache using a TCP/IP tool. These static ARP caches can be used to facilitate ARP requests for frequently called IP addresses.
The association of these MAC addresses remains unaffected even after dynamic network changes or device movements. This is comparatively an effective solution to preventing ARP spoofing on small networks consisting of a small number of connected devices. This method bypasses the dynamic ARP process where devices broadcast requests and responses to discover each other's MAC addresses.
By our understanding, static MAC entries could have been a potential solution. Their apparent simplicity and ease of implementation, particularly for smaller networks, has a certain appeal. Additionally, the potential for improved performance and predictability in specific scenarios seemed advantageous. However, all devices outside the pre-defined list are vulnerable to dynamic spoofing attacks.
Accessing the ARP entries is easy enough on a network, and it can be done by the "TCP/IP Control Panel" on MAC OS and by the command "arp -a" on the command prompt of a device running on Windows OS.

Advantages: This is a simple solution that skips dynamic ARP requests, which might offer slight speed improvements in specific scenarios. Additionally, manually setting entries is a straightforward process, especially for small networks with few devices.
Disadvantages: Attackers can continuously send spoofed messages, overriding static entries before commu-

nication occurs, rendering them useless. As a network grows or changes, manually managing entries becomes complicated and error-prone, introducing new security risks.

Comparison Between the Existing Methods:

Comparison Table			
Metric	Secure-ARP	Unequal Request Reply Algorithm	Static MAC Entries
Resource Requirements	Requires more resources than static entry approach. Cryptographic keys or certificates can add some memory overhead.	Generates significant overhead to switches. Leads to increased latency.	Minimal. Each static address requires very less memory.
Scalability	The processing overhead can become noticeable in large networks with high traffic volume, potentially impacting overall network performance.	For large networks, maintaining the counters becomes very costly.	Managing large amounts of entries can become is difficult to do manually, even with centralized management tools.
Complexity of implementation	More complex. Requires understanding the specific protocol and configuring authentication settings (keys, certificates, etc.). Usually involves more steps and specialized knowledge.	Simple to implement. Maintaining four counters for each port is easy programmatically.	Relatively simple. Requires manually adding entries for each device, but some switches offer batch configuration or centralized management tools.
Troubleshooting	More complex. Requires understanding of authentication errors, certificate issues, and potential compatibility problems. Troubleshooting logs and debugging tools become crucial.	Difficult. Gives information only about the number of ARP requests and responses.	Relatively straightforward. Identify incorrect entries or device MAC address issues and rectify the mistakes.

Table 1: Comparison Between the Existing Methods

4.Design of our ARP-Spoofing prevention technique

4.1 Terms and definitions:

1. **IP-MAC address mapping/binding**: It is the process of linking a particular IP address with a particular MAC address such that it creates a one-one relationship between the physical address and logical address.

2. **DHCP Snooping Database**: A database which contains records of trusted IP and MAC address pairs.Also known as DHCP snooping binding table.

3. **Invalid ARP packet**: If the destination IP and MAC addresses do not match with the corresponding IP and MAC address in the database the packet is considered invalid. Matching is done based on the IP addresses. These packets are ARP spoofed.

4. **Valid ARP packet**: If the destination IP and MAC addresses match with the corresponding IP and MAC address in the database the packet is considered valid.

5. **Victim Host**: The device on the network who is targeted by the attacker.

6. **Detection Host**: A device on the network whose sole purpose is to continuously monitor the network. It prevents and intercepts any attack attempts on devices in the network.

7. **Attacking host**: The device on the network that is trying to spoof using ARP aiming for attacks like Man in the middle attack.

8. **ICMP ping**: Ping is a tool utilized to ascertain the accessibility of a host on an Internet Protocol (IP) network and to calculate the round-trip time (RTT) for messages transmitted from the originating host to a target computer.

9. **RTT(Round trip time)** - The time taken by a data packet to travel from source computer to destination computer and back to source computer is known as RTT.

10. **DAI enabled switches**: It is a switch in the network which is configured with DAI to prevent an ARP attack. It allows only valid ARP packets through the network and drops an invalid ARP packet.A central switching device called hub is DAI enabled.

4.2 Algorithm Structure:

The provided algorithm can be divided into the six following structures each with its own specific set of functionalities:
1.ARP Packet Catcher Module: It captures all the ARP request and response packets generated within the network. The packets are then transferred to the Invalid Packet Detector Module to analyze the validity of the packets.

2.Invalid Packet Detection Module: This module is used to classify packets as valid or invalid.The result is sent to the response module. Packets with new IP-MAC addresses are forwarded to the ARP Spoof Verification Module.

3.DHCP snooping database: A database created to store all the legitimate IP-MAC bindings, given to the devices in the network by DHCP.

4.ARP Spoof Verification Module: The new ARP packets received from the Invalid Packet Detector Module are analyzed by this module.

5.Response Module: Gives the appropriate response in correspondence with our result from the Invalid Packet detector Module and ARP Spoof verification Module. Packets are either declared valid or invalid.

6.Periodic Network Monitor Module: Periodically Checks if the network is infiltrated or not.

FLOWCHART:

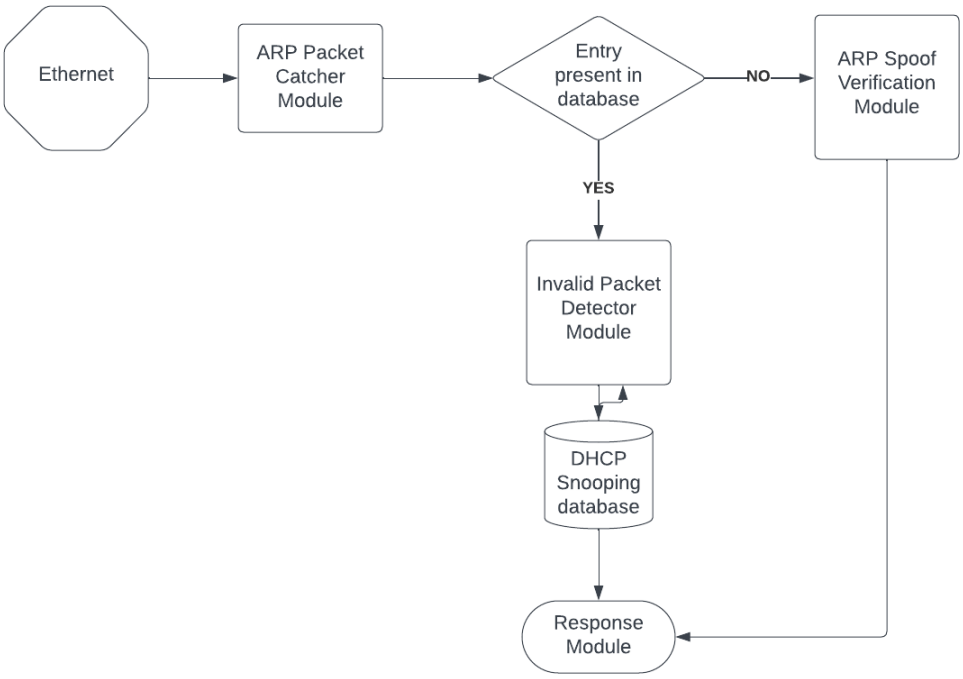

Figure 4: Working of ARP Spoofing Detection Model

4.3 Invalid Packet Detection Module:

The ARP header of an ARP packet contains the following addresses:
1. Source IP address
2. Source MAC address
3. Destination IP address
4. Destination MAC address

In an invalid ARP packet the destination IP address is of victim host while the destination MAC address is of the attacker host. In order to detect this discrepancy the MAC address in the ARP packet is compared with the MAC address stored in our DHCP Database. If addresses don't match it implies the occurrence of an ARP attack.
A hub in the network is configured with DAI to drop the packet if it is spoofed.If the entries in the ARP header match with the database the switch passes the packet to its required destination.
If IP-MAC address map in the ARP header has no entry in the database the ARP spoof verification module is used to validate the packet.

```
Switch#show ip dhcp snooping binding
MacAddress          IpAddress        Lease(sec)  Type           VLAN  Interface
------------------  ---------------  ----------  -------------  ----  ------------------
00:02:16:59:10:57   192.168.10.11    86400       dhcp-snooping  1     FastEthernet0/2
00:90:0C:B0:78:D2   192.168.10.12    86400       dhcp-snooping  1     FastEthernet0/3
00:D0:97:B7:9A:2B   192.168.10.13    86400       dhcp-snooping  1     FastEthernet0/1
Total number of bindings: 3
```

Figure 5: DHCP Snooping Database

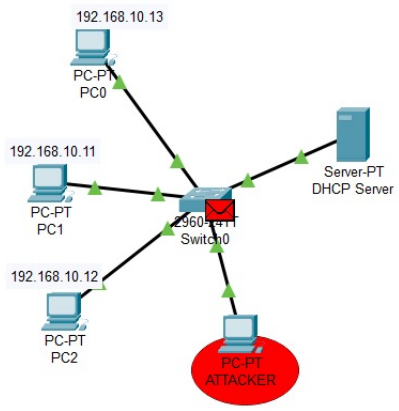

Figure 6:

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Failed	ATTA...	PC1	ICMP		0.000	N	0	(edit)	(delete)

Figure 7: ICMP Ping Packet

4.4 ARP Spoof Verification Module:

We use ICMP probing in order to detect an attack. Since the possibility of an attack is unknown the result of this module can be divided into following three cases:
1. ARP attack with no IP routing enabled
2. ARP attack with IP routing enabled
3. No ARP attack(Valid MAC address)

Case 1:
Detection host creates an ICMP echo request packet using the MAC and IP address provided in the ARP header.This packet is sent to the attacker host. In the attacker's computer the Data Link Layer accepts the packet(since MAC addresses are the same) while it is rejected by the Network Layer(due to different IP addresses). The packet is rejected and no ICMP echo response is received by the detection host. The ARP packet is considered invalid and dropped.

Case 2:
Detection host creates an ICMP echo request packet using the MAC and IP address provided in the ARP header. This packet is sent to the attacker host(because of the MAC address). In the attacker host's computer, it sees that the MAC address of the packet matches its own MAC address but the IP address does not match. Since , IP routing is enabled, it routes the packet to the device with the correct IP address.
Now, when the device receives the Echo request packet, it responds to it. After this, the detector module must sniff for the packet with the same identifier and sequence number as that of the request packet. When found, it compares the MAC address of source in the echo response packet with that in the ARP packet. If they don't match then, we are sure that an attack is taking place. The program then adds the attacker's MAC address to the untrusted database of the switch, blocks the port where the attack was taking place, and notifies the administrator about the attack.

Case 3:
If the host we suspected was legitimate, then the IP-MAC address pair in the sniffed ARP packet must be correct. So, the detection host sends the ICMP request packet to the IP-MAC address in the ARP packet. Then, the suspected host receives the packet and responds with an echo reply packet. If the host was legitimate, the IP-MAC of source in the echo reply packet will be same as that in the IP-MAC of the ARP packet. In this case the program will add this binding as a valid one in the DHCP database.

FLOWCHART:

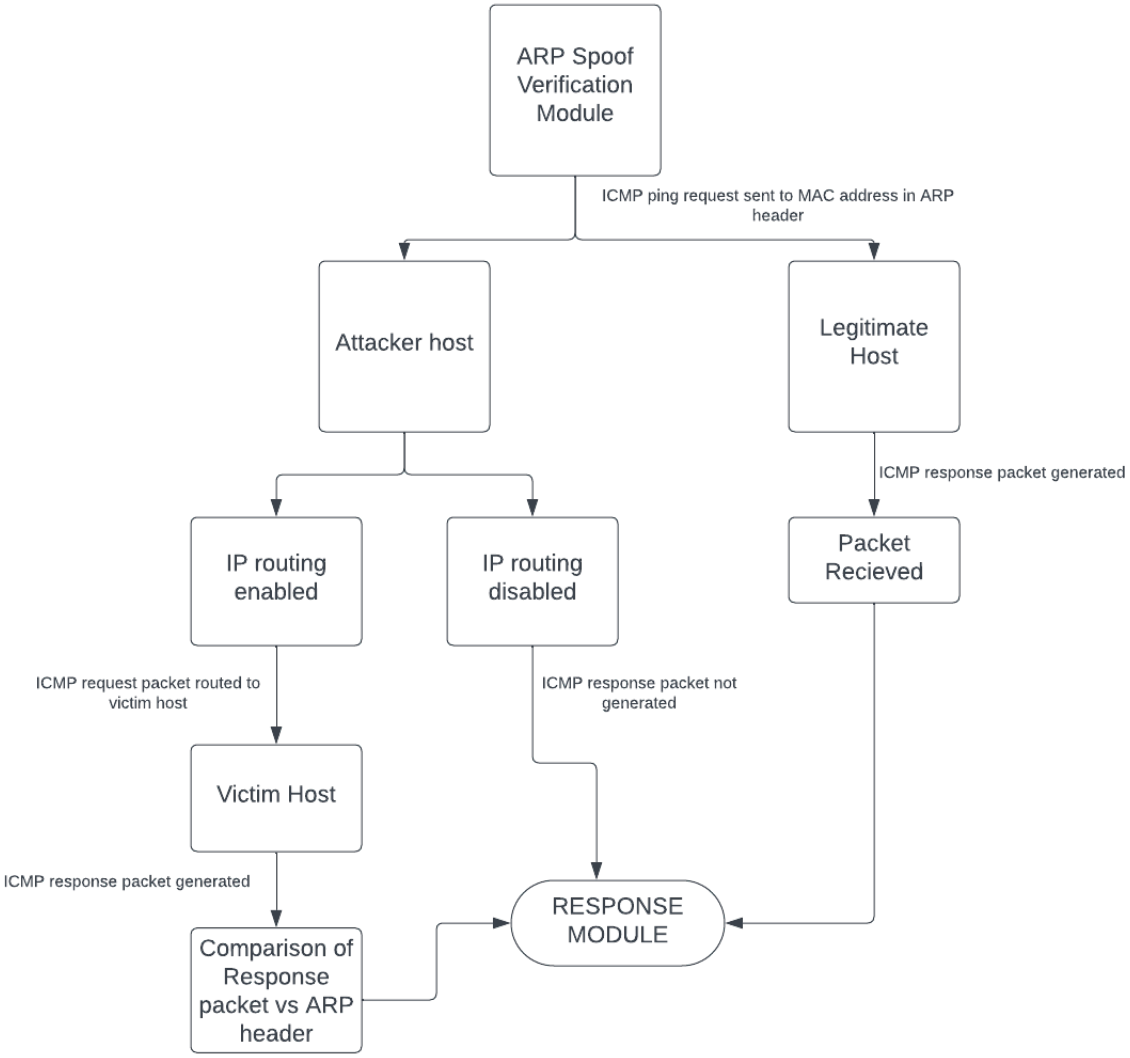

Figure 8: Working of ARP Spoof Verification Module

4.5 Periodic network monitor module:

There might be some advanced attackers that still bypass our security system. For that case, we will monitor the network periodically by sending Echo request packets to every device from the detection host. The detection host will have a database of the devices MAC, IP and average RTT. We will define a threshold limit for every device in the network.
Let the average RTT value for a device be RTT(avg), the received RTT be RTT(curr), and the threshold limit be L.
If RTT(curr) > RTT(avg) + L, we respond by notifying the administrator. Then we compare the ARP cache table of the expectedly attacked device and compare the entries with that of the DHCP cache table. If the entries are the same, we respond that no attack is taking place. If any differences are found, we block the port, add the MAC address of the attacker to the untrusted database of the switch, and block the port where the attacker was connected. Then the program notifies the administrator about the attack.

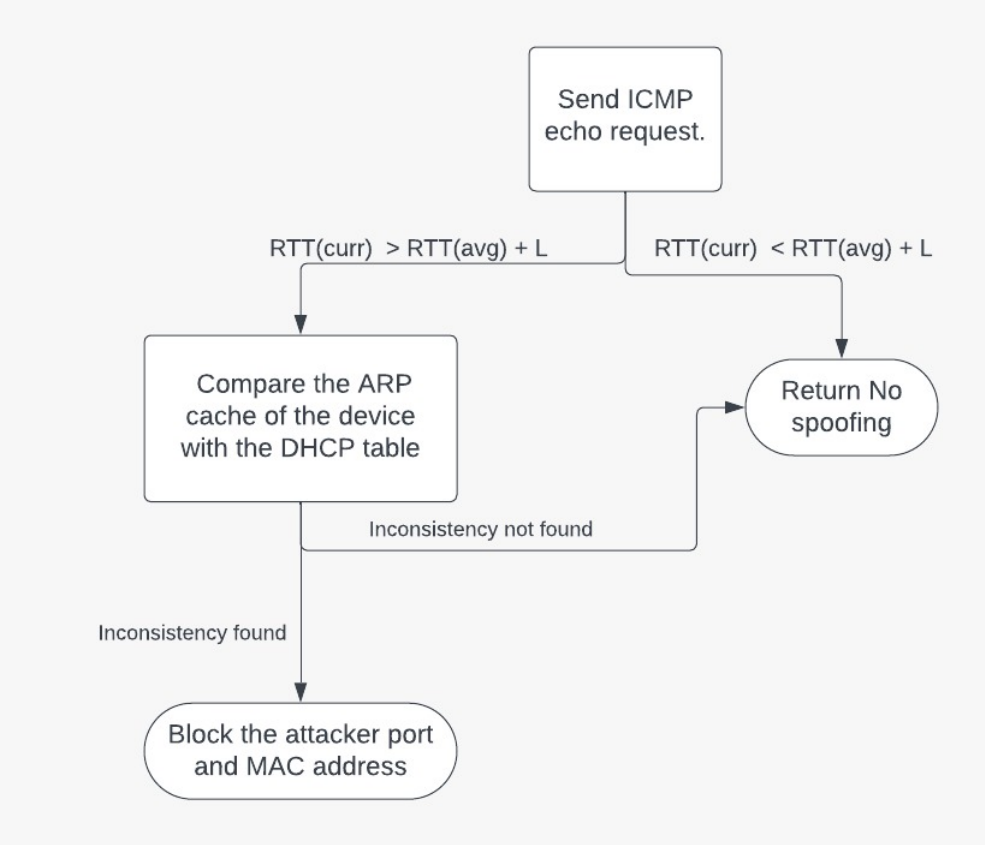

Figure 9: Working of Periodic Network Monitor Module

5. Implementation of our ARP-Spoofing prevention technique

6. Result and Analysis

7. Conclusion

References

[1] Huixing Xi. Research and application of ARP protocol vulnerability attack and defense technology based on trusted network. In *Advances in Materials, Machinery, Electronics (AMME 2017)*, volume 1820 of *American Institute of Physics Conference Series*, page 090019, March 2017.

[2] Wen Zeng, Junsheng Zhang, Ying Li, and Peng Qu. The study on media access control protocol for wireless network in library. *International Journal of Distributed Sensor Networks*, 2015:1–10, 08 2015.

[3] Jaideep singh, Sandeep Dhariwal, and Rajeev Kumar. A detailed survey of arp poisoning detection and mitigation techniques. *International Journal of Control Theory and Applications*, 9, 02 2017.

[4] A. Mali P. Patni K., Iyer R. Sarode. Man-in-the-middle attack in http/2. *International Conference on Intelligent Computing and Control (I2C2)*, 2017.

[5] Khalid M. Amin Ahmed M.Abdel Salam, Wail S. Elkilani. An automated approach for preventing arp spoofing attack using static arp entries. *(IJACSA) International Journal of Advanced Computer Science and Applications*, 2014.

[6] Cisco. Static mac address support on service instances.

[7] Shimpy Goyal Vaishnavi Rohatgi. A detailed survey for detection and mitigation techniques against arp spoofing. *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 2020.

[8] A. Kaur R. Kaur. Digital signature, in: 2012 international conference on computing sciences, pp. 295–301. 2012.

[9] Jussi Toivakka. Network segmentation. *South-Eastern Finland University of Applied Sciences*, December 2018.

[10] Marco Carnut and Joao Gondim. Arp spoofing detection on switched ethernet networks: A feasibility study. 11 2003.

****** END ******