Incident Response Report: ARP Poisoning Attack

Mohammed Firdaws Alnuur

Future Interns

Nairobi, Kenya

Feb 2025

Declaration and approval

I declare that this report is my original work and has not been previously submitted for approval. To the best of my knowledge, all sources have been cited appropriately

Student Name: Mohammed Firdaws Alnuur

Email: firdawsalnuur4@gmail.com

Abstract

This report examines an Address Resolution Protocol (ARP) poisoning attack conducted as part of a simulated cybersecurity incident response exercise. The objective was to analyze the attack methodology, assess its impact, and develop mitigation strategies. The attack exploited ARP vulnerabilities using Ettercap to manipulate ARP tables, allowing unauthorized interception of network traffic and credential theft. The report details the attack chain, indicators of compromise, and countermeasures, including network monitoring, ARP cache management, and security protocol enforcement. By implementing best practices such as Dynamic ARP Inspection and encrypted communication, organizations can enhance network security and prevent similar attacks.

Chapter 1:   Introduction

1.1   Executive Summary

This report analyzes an Address Resolution Protocol (ARP) poisoning attack conducted as part of a simulated cybersecurity incident response exercise. The attack exploited the lack of authentication in ARP, allowing an attacker to intercept network traffic and capture sensitive data. The report provides a detailed overview of the attack, its impact, and the mitigation steps taken to prevent recurrence.

1.2 Incident Overview

Incident Type:

ARP Poisoning Attack

Initial Access Vector:

Exploitation of ARP protocol vulnerabilities using Ettercap to manipulate ARP tables and intercept traffic.

1.3   Objectives

1   Simulate an ARP poisoning attack
2   Capture network traffic and sensitive credentials
3   Implement mitigation strategies

1.4 Attack Chain Analysis

1. **Reconnaissance:**
The attack occurred due to the vulnerability of the Address Resolution Protocol (ARP), which lacks authentication. This allowed an attacker to:

- Use Ettercap to scan the network and identify hosts.
- Perform ARP poisoning, tricking devices into associating the attacker's MAC address with a legitimate IP address (., 192.168.100.1).
- Capture sensitive data, such as login credentials, using Wireshark.

2. **Initial Exploitation:**

- Performed ARP poisoning by sending spoofed ARP replies, associating the attacker's MAC address with a legitimate IP address (192.168.100.1).

3. **Persistence:**
- Continued to manipulate ARP tables to sustain network interception.

4. **Data Exfiltration:**
- Captured network traffic using Wireshark.
- Extracted login credentials from intercepted packets.

5. **Analysis and Response:**

- Detected anomalies in ARP tables.

- Identified changes in MAC addresses.

1.5   Indicators of Compromise (IOCs)

Type	Details
Malicious MAC Address	08-00-27-38-c8-03
Targeted IP Address	192.168.100.1
Attack Tool Used	Ettercap
Captured Data	Login credentials via Wireshark

Chapter 2. Impact Analysis

2.1: Affected Systems:

- Devices connected to the local network were vulnerable to traffic interception.

Potential Damage:

- Unauthorized access to sensitive credentials.
- Possible session hijacking.
- Man-in-the-Middle (MitM) attack execution.

2.2: Mitigation Steps Taken

1. **Detection:**
   Identify the malicious MAC address:

   - Use arp -a (Windows) or arp -n (Linux) to check the ARP table for anomalies.
   - Compare against known device MAC addresses.

1. **Containment and Eradication:**

   - Cleared ARP cache:
       - Windows: netsh interface ip delete arpcache
       - Linux/macOS: sudo ip -s -s neigh flush all
   - Restarted affected devices to refresh ARP tables.

2. **Recovery:**

   - Implemented static ARP entries for critical network components.
   - Deployed ARP monitoring tools (Arpwatch, XArp) to detect future spoofing attempts.

2.3: Prevention Recommendations

1. **Network Security Enhancements:**

   - Implement Dynamic ARP Inspection (DAI) on managed switches.

- Enable port security to restrict allowed MAC addresses.
- Use encrypted communication protocols (HTTPS, SSH, VPN) to mitigate data interception risks.
2. **Monitoring and Response:**

- Deploy intrusion detection systems (IDS) to monitor ARP spoofing activities.
- Conduct periodic network traffic analysis with Wireshark or similar tools.

2.4: Recommendations to Prevent Future Attacks

- Implement Dynamic ARP Inspection (DAI) on managed switches to verify ARP packets.
- Use static ARP entries for essential systems to prevent ARP spoofing.
- Enable port security on network switches to limit allowed MAC addresses.
- Implement network segmentation to isolate critical systems.
- Use encrypted protocols (HTTPS, SSH, VPN) to protect data from interception.
- Monitor network traffic regularly using tools like Wireshark or IDS/IPS solutions.

Conclusion

The ARP poisoning attack successfully demonstrated how adversaries can exploit unsecured network protocols to intercept sensitive data. By implementing network security measures such as Dynamic ARP Inspection, static ARP entries, and encrypted communication, organizations can significantly reduce the risk of ARP-based attacks. Regular monitoring and awareness training are essential in mitigating future threats.
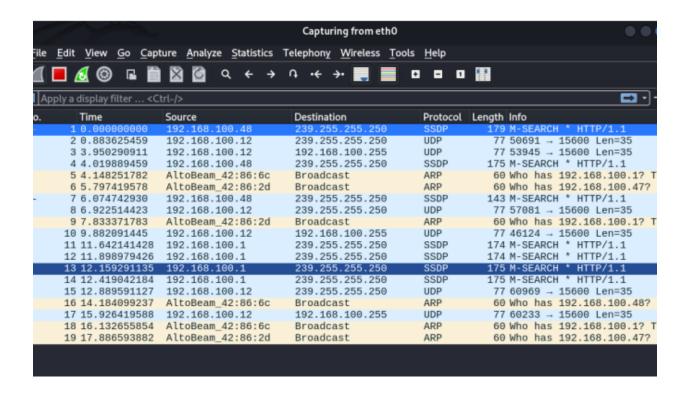
Chapter 3 : Appendices

```
Wireless LAN adapter WiFi:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::fd9f:72e7:a3ed:43dd%12
   IPv4 Address. . . . . . . . . . . : 192.168.100.8
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.100.1
```

```
Ethernet adapter Bluetooth Network Connection:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Ethernet adapter vEthernet (WSL (Hyper-V firewall)):

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::869:bb10:be3b:5594%51
   IPv4 Address. . . . . . . . . . . : 172.22.32.1
   Subnet Mask . . . . . . . . . . . : 255.255.240.0
   Default Gateway . . . . . . . . . :

C:\Users\firda>
```

```
th0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
     inet 192.168.100.40  netmask 255.255.255.0  broadcast 192.168.100.255
     inet6 fe80::a00:27ff:fe38:c803  prefixlen 64  scopeid 0x20<link>
     ether 08:00:27:38:c8:03  txqueuelen 1000  (Ethernet)
     RX packets 102  bytes 9174 (8.9 KiB)
     RX errors 0  dropped 0  overruns 0  frame 0
     TX packets 41  bytes 5292 (5.1 KiB)
     TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
Interface: 192.168.100.8 --- 0xc
  Internet Address      Physical Address      Type
  192.168.100.1         c8-84-cf-e9-7f-52     dynamic
  192.168.100.3         62-14-b8-56-e1-99     dynamic
  192.168.100.12        e8-aa-cb-24-11-7e     dynamic
  192.168.100.40        08-00-27-38-c8-03     dynamic
  192.168.100.255       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
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

Ensuring that Wireshark is running in the background

Opening Ettercap to simulate the attack then I went ahead an continued Scanning for hosts i got.Which are a total of 8 hosts.

Setting 192.168.100.1 (c8-84-cf-e9-7f-52) as Target 1

```
Starting Unified sniffing...

Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
3 hosts added to the hosts list...
Host 192.168.100.1 added to TARGET1
```

Starting ARP poisoning

The mac address is now changed from c8-84-cf-e9-7f-52 to 08-00-27-38-c8-03

```
C:\Users\firda>arp -a

Interface: 192.168.199.1 --- 0xa
  Internet Address       Physical Address      Type
  192.168.199.255        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
  239.255.255.250        01-00-5e-7f-ff-fa     static

Interface: 192.168.100.8 --- 0xc
  Internet Address       Physical Address      Type
  192.168.100.1          08-00-27-38-c8-03     dynamic
  192.168.100.3          62-14-b8-56-e1-99     dynamic
  192.168.100.12         e8-aa-cb-24-11-7e     dynamic
  192.168.100.40         08-00-27-38-c8-03     dynamic
  192.168.100.255        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
  239.255.255.250        01-00-5e-7f-ff-fa     static
  255.255.255.255        ff-ff-ff-ff-ff-ff     static
```

Capturing login details

```
C:\Users\firda>arp -a

Interface: 192.168.199.1 --- 0xa
  Internet Address      Physical Address      Type
  192.168.199.255       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
  239.255.255.250       01-00-5e-7f-ff-fa     static

Interface: 192.168.100.8 --- 0xc
  Internet Address      Physical Address      Type
  192.168.100.1         c8-84-cf-e9-7f-52     dynamic
  192.168.100.3         62-14-b8-56-e1-99     dynamic
  192.168.100.12        e8-aa-cb-24-11-7e     dynamic
  192.168.100.40        08-00-27-38-c8-03     dynamic
  192.168.100.255       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
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

Login credentials extracted

Ettercap analysis

Wireshark capture