



# Vidyavardhini's College of Engineering & Technology

## Department of Artificial Intelligence and Data Science

### EXPERIMENT 09

**Aim:** Detect ARP spoofing using nmap and/or open source tool ARPWATCH and wireshark **Objectives:**

- To understand ARP spoofing.
- To understand ARPWATCH and use it to detect ARP spoofing.

#### Theory:

##### 1. Nmap (Network Mapper):

While Nmap isn't specifically designed for ARP spoofing detection, it can be used indirectly. Nmap can perform a quick network scan to identify active devices and their MAC addresses. You can then compare this information with the ARP table on your machine (using `arp -a` on Linux/macOS) to identify any discrepancies. For example, if Nmap identifies a device with a specific IP address but the ARP table shows a different MAC address associated with that IP, it might indicate ARP spoofing. However, this method can be unreliable as legitimate network configurations can also cause MAC address changes.

##### 2. Arpwatch:

Arpwatch is a dedicated tool for monitoring ARP activity on your network. It keeps track of learned MAC addresses for IPs and monitors for any changes. Here's how it helps detect ARP spoofing:

**Database:** Arpwatch maintains a database of learned IP/MAC mappings.

**Monitoring:** It continuously monitors ARP packets on the network.

**Alerting:** If Arpwatch detects an unsolicited ARP reply (attacker trying to modify the ARP table) or a change in the MAC address associated with a known IP, it raises an alert in the system logs.

##### 3. Wireshark:

Wireshark is a powerful network packet analyzer. While not solely for ARP spoofing detection, it can be used for in-depth analysis of network traffic. Here's how it helps:

**Packet Capture:** Wireshark can capture live network traffic.

**Filtering:** You can filter captured packets to focus specifically on ARP traffic.

**Analysis:** By examining ARP packets, you can identify inconsistencies. For instance, if you see multiple ARP replies for the same IP address with different MAC addresses, it might indicate ARP spoofing.

#### Implementation :

##### Using nmap

`nmap` is primarily a network scanning tool, but it can be used to detect ARP spoofing by checking for duplicate IP addresses or multiple MAC addresses associated with a single IP.



# Vidyavardhini's College of Engineering & Technology

## Department of Artificial Intelligence and Data Science

### 1. Scan the Network:

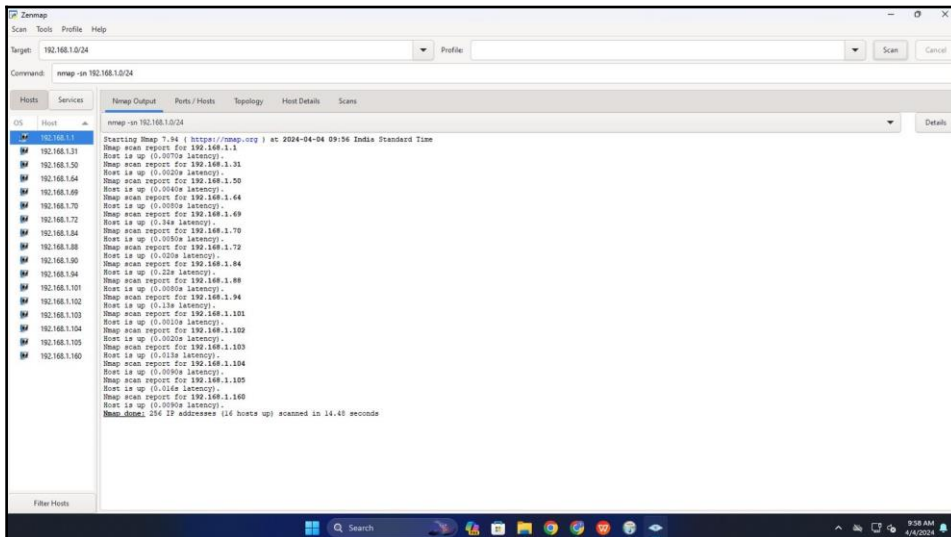
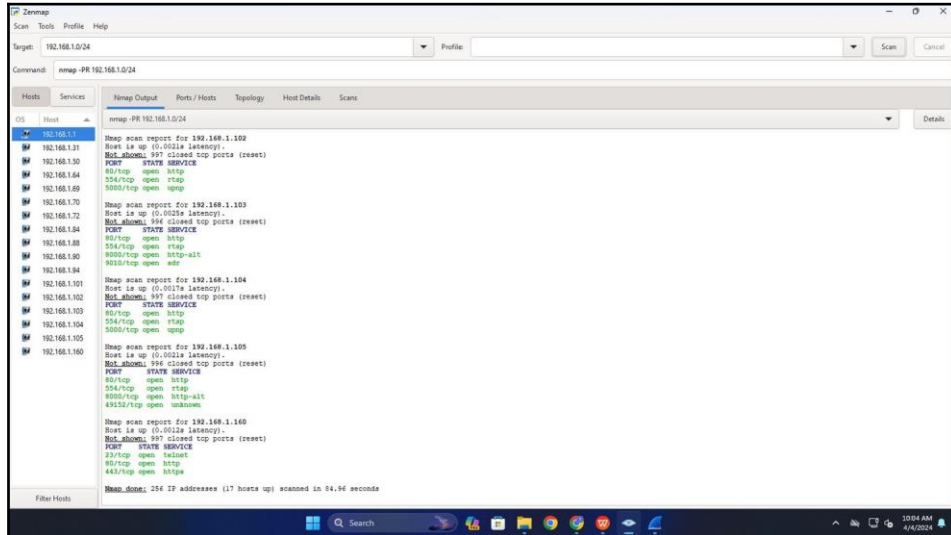
Use `nmap` to scan your network and identify all live hosts.

```
nmap -sn 192.168.1.0/24
```

### 2. Check for Duplicate IPs:

Look for multiple MAC addresses associated with a single IP, which could indicate an ARP spoofing attack.

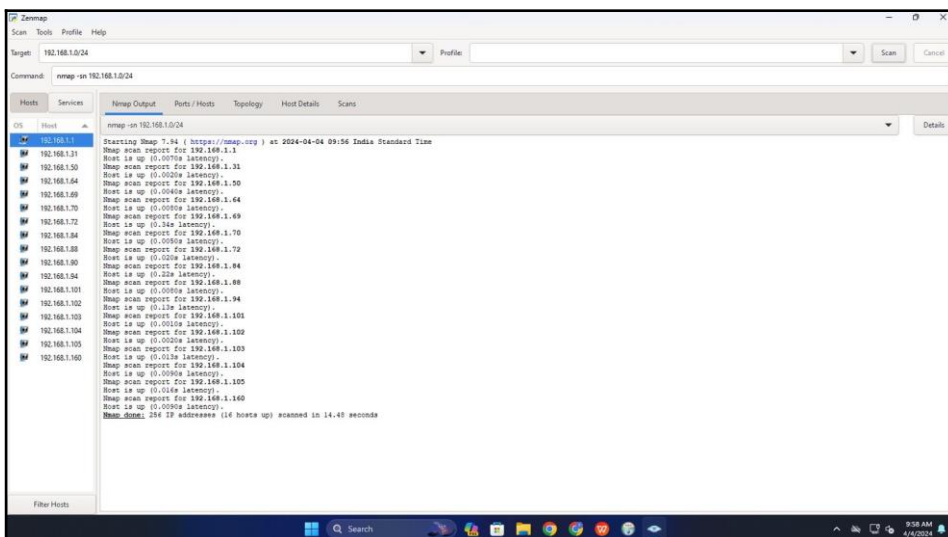
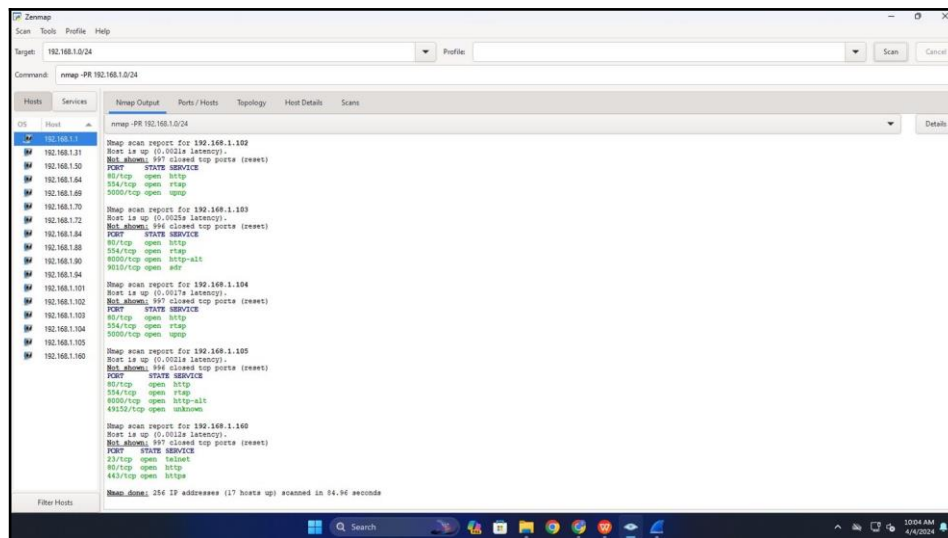
**nmap -PR 192.168.1.0/24 Output:**





# Vidyavardhini's College of Engineering & Technology

## Department of Artificial Intelligence and Data Science



### • Using Wireshark

`Wireshark` is a powerful network protocol analyzer that can be used to capture and analyze network traffic, making it suitable for detecting ARP spoofing attacks.

#### 1. Capture Traffic:

Start capturing traffic on the network interface where you suspect ARP spoofing is happening. In Wireshark, select the appropriate network interface and start the capture.

#### 2. Filter ARP Packets:

Use Wireshark's display filters to only show ARP packets.

arp

#### 3. Analyze ARP Requests and Replies:

Look for inconsistencies in ARP requests and replies. In a typical ARP spoofing attack, you might see:

- ARP replies for ARP requests that were never sent.



- Pay close attention to the source and destination MAC addresses, as well as the IP addresses involved in ARP requests and replies.

```
> Frame 184736: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on Interface \Device\NPF_{C3B8F3A6-C52F-4D52-B079-C255ID6E0593}, Ld 0
+ Ethernet II, Src: HewlettP_09:Bc:c2 [3c:52:b2:09:b0:c2], Dst: Broadcast ff:ff:ff:ff:ff:ff
+ Destination: Broadcast ff:ff:ff:ff:ff:ff
Address: Broadcast ff:ff:ff:ff:ff:ff
.... :.. .... = LG bit: Locally administered address (this is NOT the factory default)
.... +1 ..... = DG bit: Group address (multicast/broadcast)
Source: HewlettP_09:Bc:c2 [3c:52:b2:09:b0:c2]
Address: HewlettP_09:Bc:c2 [3c:52:b2:09:b0:c2]
.... .0 ..... = LG bit: Globally unique address (factory default)
.... ..0 ..... = IG bit: Individual address (unicast)
Type: ARP (0x0806)
Payload: ooooooooooooooooooooooooonooooooooooooooooo
+ Address Resolution Protocol Request
Hardware type: Ethernet (1)
Protocol type: IPv4 (0x0800)
Hardware size: 6
Protocol size: 4
Opcode request (1)
Sender MAC address: HewlettP_09:Bc:c2 [3c:52:b2:09:b0:c2]
Sender IP address: 192.168.12.241
Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)
Target IP address: 192.168.12.13
> [Duplicate IP address detected for 192.168.12.241 (3c:52:b2:09:b0:c2) - also in use by 48-bebdb:9f:ef:bc (frame 57193)]
```

```
0000  ff ff ff ff ff ff 3c 52 b2 09 bc c2 00 00 01   .....R .....
0010  00 00 00 00 00 00 01 3e 52 b2 09 bc c8 ad 0c f1   .....R .....
0020  00 00 00 00 00 00 c8 ad 0c 52 00 00 00 00 00 00   .....
0030  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   .....
```

[illegible]

In conclusion, ARP spoofing poses a significant security threat by allowing attackers to intercept network traffic. Utilizing tools like Nmap, ARPWATCH, and Wireshark can aid in detecting such attacks. Nmap provides a quick network scan to identify discrepancies in IP-MAC mappings, ARPWATCH continuously monitors ARP activity for any anomalies, and Wireshark allows for in-depth analysis of ARP packets for inconsistencies. By



# **Vidyavardhini's College of Engineering & Technology**

## **Department of Artificial Intelligence and Data Science**

combining these tools and techniques, network administrators can enhance their ability to identify and mitigate ARP spoofing attacks, bolstering overall network security.