4. Network Protocol Attacks Lab

Introduction

Network protocol attacks focus on exploiting weaknesses in communication protocols rather than individual applications. Many core protocols such as SMB, ARP, and DNS were designed for trusted internal networks and lack built-in security mechanisms. In this lab, the emphasis is on understanding how attackers abuse these protocols through Man-in-the-Middle (MitM) positioning, credential interception, and traffic analysis to gain unauthorized access.

Understanding Network Protocol Attacks

Network protocols define how devices communicate across a network. When these protocols are misconfigured or lack authentication and encryption, attackers can intercept, modify, or replay traffic. Unlike application-level attacks, protocol attacks often provide **network-wide visibility**, enabling credential theft, session hijacking, and lateral movement.

Protocol attacks are particularly dangerous in internal networks, where systems implicitly trust one another.

Man-in-the-Middle (MitM) Attacks

A Man-in-the-Middle attack occurs when an attacker secretly positions themselves between two communicating systems and intercepts or alters the traffic without either party being aware. Once in a MitM position, attackers can observe sensitive data, manipulate requests, or redirect connections.

ARP Spoofing as a MitM Technique

Address Resolution Protocol (ARP) maps IP addresses to MAC addresses. ARP lacks authentication, making it vulnerable to spoofing. By sending forged ARP replies, attackers

can convince both the victim and the gateway that the attacker's machine is the legitimate communication partner.

This allows:

- Traffic interception
- Credential capture
- Protocol manipulation

SMB Relay Attacks

SMB relay attacks exploit Windows authentication mechanisms, particularly NTLM. Instead of cracking captured credentials, the attacker relays NTLM authentication attempts to another machine on the network to gain unauthorized access.

Activity: SMB Relay Attack Using Responder

This attack relays captured NTLM authentication attempts to another host instead of storing them.

1. Identify a Relay Target

Select a second machine on the network with **SMB signing disabled**.
Example relay target:

192.168.0.102

2. Configure Responder for Relay Mode

Edit the configuration file again:

sudo nano /etc/responder/Responder.conf

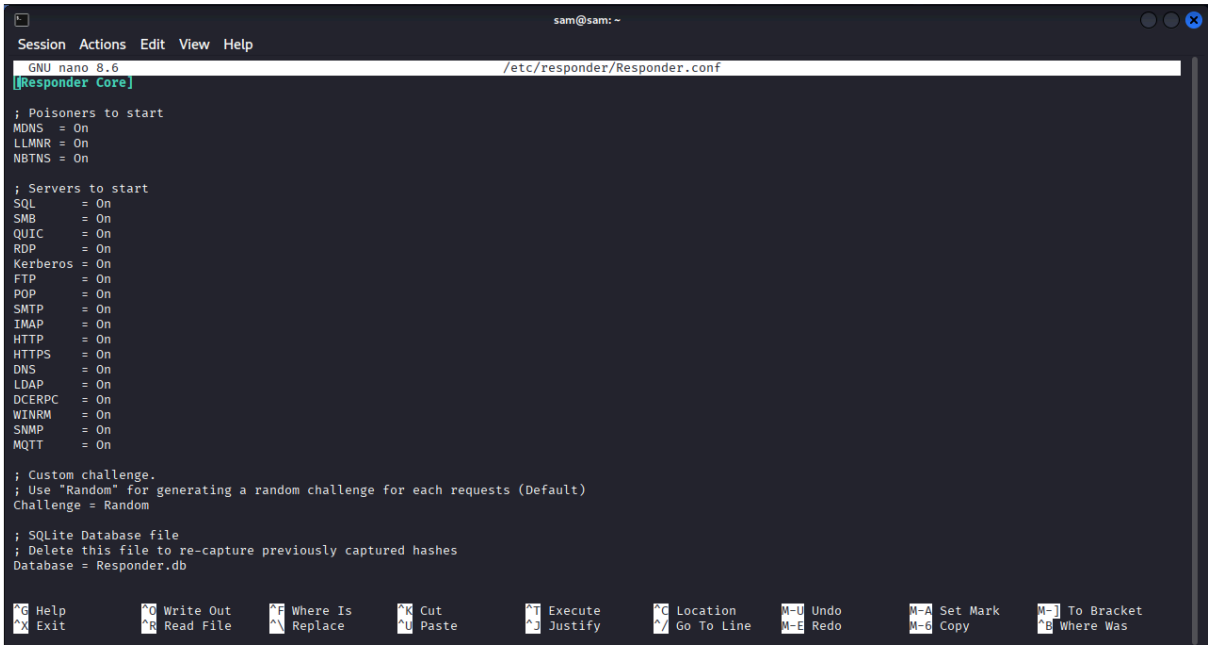

Figure 1: Edit the responder config file

Set the following services to Off:

- SMB
- HTTP

This prevents Responder from capturing hashes locally.

3. Launch Responder in Relay Mode

sudo responder -I eth0 -r 192.168.0.102 -e

- -r specifies the relay target
- -e enables exploitation mode

Figure 2: Responder listening

4. Trigger Authentication

From the original victim machine (192.168.0.102):

\\192.168.0.106\test

5. Observe the Result

Responder intercepts the authentication attempt and relays it to 10.10.10.25.
If successful, the terminal will indicate authenticated access or shell availability.

Why SMB Relay Is Effective

- Does not require password cracking
- Exploits trust relationships between systems
- Works even with strong passwords
- Often succeeds when SMB signing is disabled

SMB Relay Attack Flow

1. Victim attempts to authenticate to a resource.

2. Attacker intercepts the NTLM authentication request.
3. The authentication is relayed to another system.
4. The target system accepts the authentication.
5. Unauthorized access is granted.

Credential Exposure via NTLM

NTLM authentication uses challenge-response mechanisms. While the password is not sent in plaintext, the authentication response (NTLM hash) can still be abused through relay attacks or offline cracking.

Activity: Capturing NTLM Hashes Using Responder

Responder is a powerful network-based attack tool designed to **poison name resolution requests** and coerce systems into authenticating, thereby exposing NTLMv2 hashes.

1. Stop Conflicting Services

Responder must bind to specific network ports. To prevent port conflicts, stop services that may already be using them.

```
sudo systemctl stop smbd nmbd
sudo service apache2 stop
```

2. Configure Responder

The Responder configuration file is located at:

```
/etc/responder/Responder.conf
```

Edit the file to disable unnecessary services, which improves stability and avoids crashes.

```
sudo nano /etc/responder/Responder.conf
```

For basic SMB hash capture:

- Ensure **SMB** and **HTTP** are set to On

- Disable unused services such as **FTP** and **SQL**

3. Run Responder

Launch Responder using your active network interface.

```
sudo responder -I eth0 -A
```

- -I eth0 specifies the network interface
- -A enables aggressive mode

To trigger authentication from a Windows target, execute the following on the victim system:

```
\\10.10.10.5\test
```

4. Capture the NTLM Hash

Responder will intercept the authentication attempt and display the captured NTLMv2 hash in the terminal:

```
SMBv2: [username]::[domain]:[hash_part_1]:[hash_part_2]:[timestamp]
```

This hash can later be cracked or relayed depending on the attack objective.

Security Impact of NTLM Exposure

- Enables lateral movement
- Leads to domain compromise
- Facilitates privilege escalation
- Supports persistent access

DNS Spoofing (Theory)

DNS spoofing involves providing false DNS responses to redirect victims to malicious destinations. When combined with MitM positioning, attackers can manipulate DNS queries to perform phishing, credential harvesting, or malware delivery.

Activity: DNS Spoofing Using Ettercap

This activity demonstrates a **Man-in-the-Middle (MitM)** attack using ARP poisoning combined with DNS spoofing to redirect traffic.

1. Enable IP Forwarding

Enable IP forwarding so Kali can relay traffic between the victim and the gateway.

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

2. Configure DNS Spoofing Rules

Edit the Ettercap DNS configuration file:

```
sudo nano /etc/ettercap/etter.dns
```

Add entries to redirect a target domain to your Kali IP:

```
example.com      A   192.168.0.106
*.example.com    A   192.168.0.106
```

Save and exit.

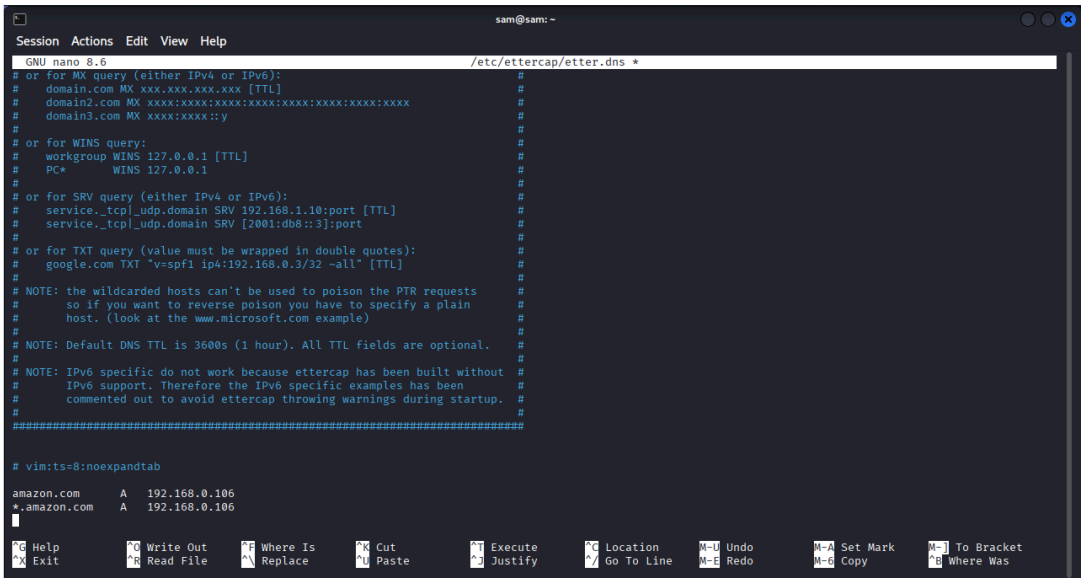

Figure 3: Ettercap config

3. Launch Ettercap in Graphical Mode

sudo ettercap -G

Within the Ettercap GUI:

- Navigate to **Sniff → Unified sniffing…**
- Select your network interface (eth0)
- Go to **Hosts → Scan for hosts**
- View discovered hosts via **Hosts → Hosts list**
- Add:
    - Target IP (10.10.10.20) → **Target 1**
    - Gateway IP (10.10.10.1) → **Target 2**
- Start ARP poisoning via **Mitm → ARP poisoning**
    - Enable *"Sniff remote connections"*
- Activate DNS spoofing:
    - **Plugins → Manage plugins → dns_spoof**

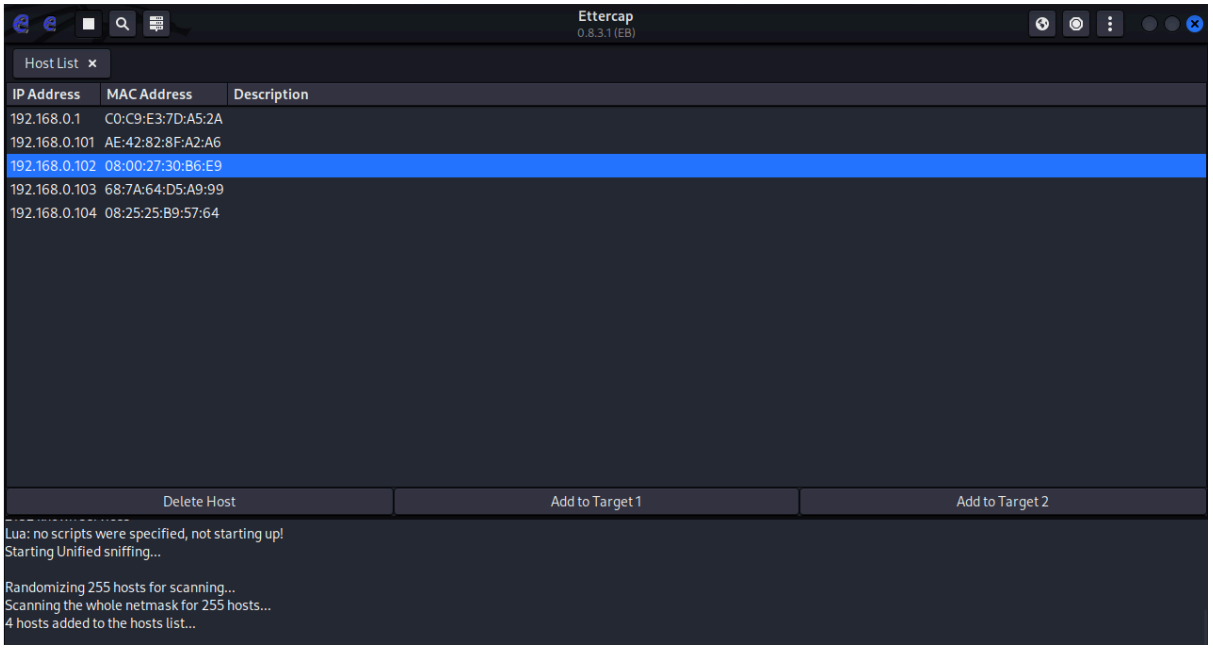

Figure 4: Host list

4. Verify DNS Spoofing

On the target machine, open a browser and navigate to:

example.com

The request should be redirected to your Kali machine, confirming successful DNS spoofing. DNS queries and spoofed responses will appear in the Ettercap console.

Common DNS Spoofing Outcomes

- Traffic redirection
- Credential theft
- Session hijacking
- Malware hosting

Traffic Analysis Using Wireshark

Purpose of Traffic Analysis

Traffic analysis allows attackers and defenders to observe communication patterns, protocol behavior, and security weaknesses. Wireshark captures raw packets and decodes protocol structures, making it an essential tool for understanding attack execution and impact.

Activity: Traffic Analysis Using Wireshark

Wireshark is used to **passively analyze** the network traffic generated during the attacks.

1. Start Wireshark

Launch Wireshark from the applications menu or by running:

wireshark

2. Select Network Interface

Double-click your active interface (eth0) to begin capturing traffic.

3. Apply Display Filters

Use filters to isolate relevant packets:

ARP Poisoning:

arp

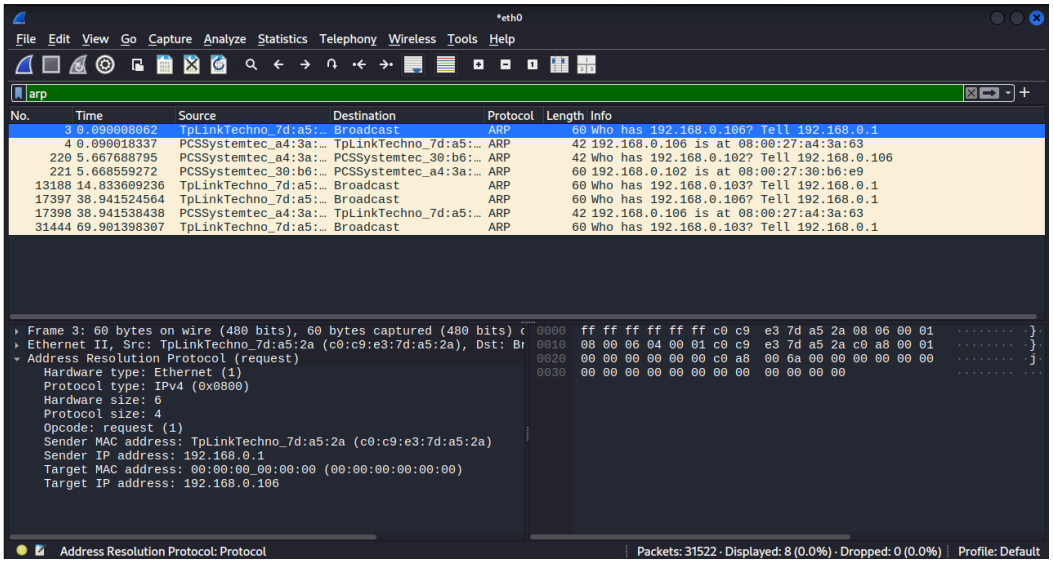

Figure 5: ARP Traffic

SMB Relay Traffic:

smb

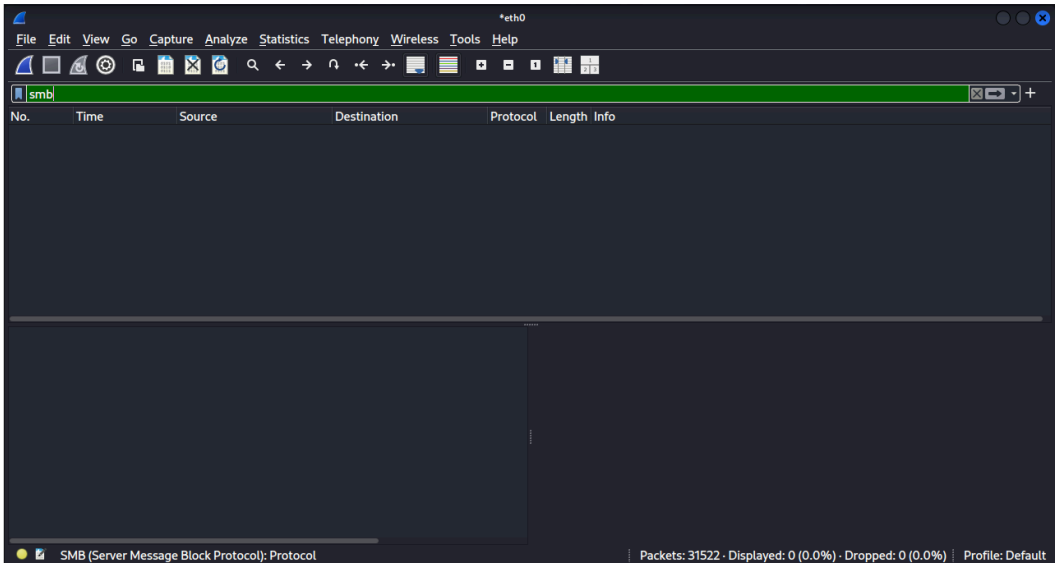

Figure 6: SMB Relay

DNS Spoofing:

dns

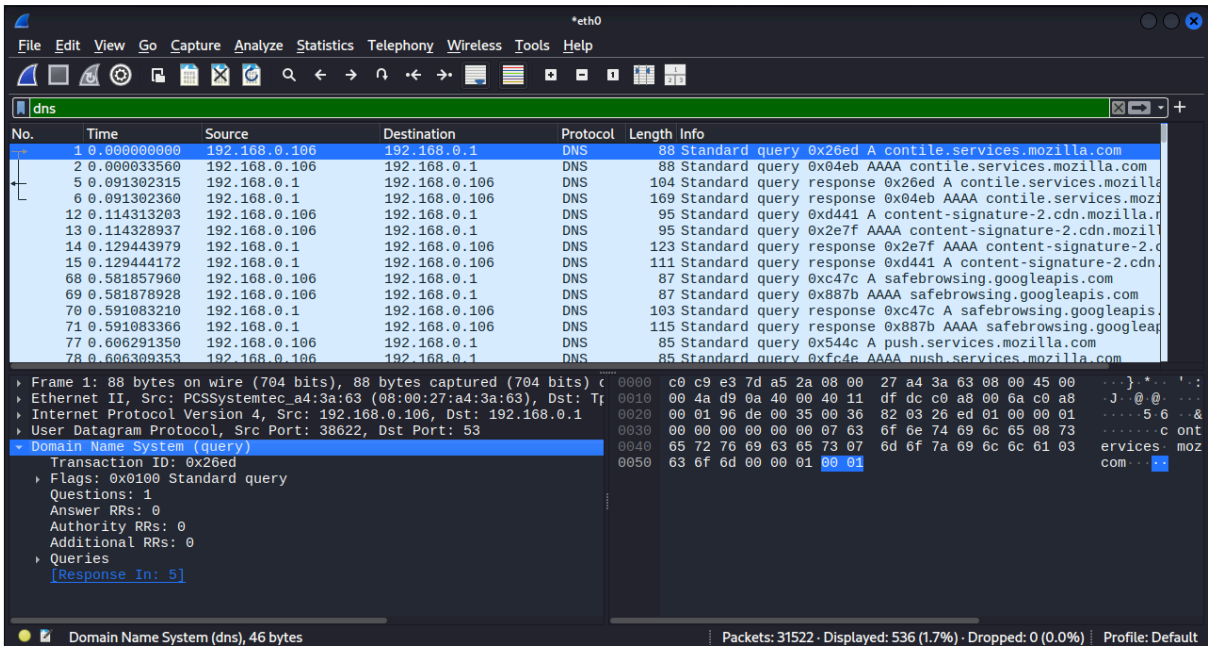

Figure 7: DNS Traffic

You can expand protocol layers such as **NTLMSSP** under SMB to observe authentication negotiations.

Key Protocols Observed

Protocol	Security Risk
ARP	Enables MitM attacks
SMB	Credential relay and theft
DNS	Traffic redirection

Attack Logging and Documentation

Accurate documentation ensures traceability and risk assessment. Logging protocol attacks highlights how simple network weaknesses can lead to credential compromise.

Attack ID	Technique	Target IP	Status	Outcome
015	SMB Relay	192.168.0.102	Success	NTLM Hash

Security Impact

Unsecured network protocols can lead to:

- Credential compromise
- Lateral movement
- Privilege escalation
- Network-wide breaches
- Persistent attacker presence

Network attacks often bypass perimeter defenses and exploit internal trust assumptions.

Conclusion

Network protocol attacks remain a powerful technique for attackers due to the inherent trust and legacy design of many protocols. Understanding these attacks is essential for identifying systemic weaknesses and implementing effective network-level defenses.

Document Information:
Report Generated: December 26, 2025
Author: Soumendu Manna - CyArt VAPT Intern