



Lab5: Network Penetration Testing Methodology-Internal (2/3)

INFO40587: ETHICAL HACKING

Kevin Harianto | 991602128 | June 14, 2024

Contents

Contents

Contents	1
Executive Summary	4
Exercise 8: Create a Python Script to Grab the Banner of the ssh Service	5
Exercise 8, Step 3: Open a terminal window on the Parrot machine and enter the following code	5
Exercise 9: Use Metasploit to Detect Version of HTTP	7
Exercise 9, Step 5: Enter ls -lx	7
Exercise 9, Step 11: An example of an excerpt of the code is shown in the screenshot	8
Exercise 9, Step 12: The key to this routine is in the defined class:	9
Exercise 10: Enumerating SMB	10
Exercise 10, Step 5: In the terminal window, type nmap -sC 192.168.0.7 and press Enter	11
Exercise 10, Step 6: The output of the command in step 5 reveals more details than that of the command in step 4. The scan may take approximately 5 to 10 minutes to complete.	
.....	12
Exercise 11: Pentesting Misconfigured RPC Service and NFS Shares	13
Exercise 11, 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 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 172.19.19.51 (RPC Server Ubuntu) as our target.....	14
Exercise 11, Step 5: Nmap takes around 30 seconds to complete the scan. On completing the scan, you will observe that the services rpc, ftp, nfs and mountd are running on the victim machine. From the scan, it is observed that an NFS File system is mounted on the remote machine. In this lab, we shall focus on the RPC, NFS and mountd services	
.....	15
Exercise 11, Step 7: We observe that nfs and mountd services are active on the remote machine	15
Exercise 11, Step 8: Now, we shall issue the showmount command to discover NFS shares listed in /etc/exports file of the remote machine. Type showmount -e 172.19.19.51 and press Enter. This will display all the NFS shares on the remote machine as shown in the screenshot below	16
Exercise 11, Step 9: As we saw in the previous task, the /home file system was shared on the remote machine. We will be mounting this file system on the Parrot machine to the mnt directory. To mount, type sudo mount -t nfs 172.19.19.51:/home /mnt -o noblock	

and press Enter. Type toor and press Enter when prompted.	17
Exercise 11, Step 11: Type ls and press Enter to view the files and directories contained in the /home folder i.e., /mnt.....	18
Exercise 11, Step 13: On entering the command in the previous task, the cat command displays the file contents in the secret.txt file successfully, meaning we have successfully mounted the remote file system and accessed the contents in it.	19
Exercise 11, Step 15: Now, we shall see if we are able to tamper/delete the files in the remote file system. Type rm administrator/Documents/secret.txt and press Enter. Type y and press Enter to confirm the deletion. To confirm that the file has been successfully deleted, type cat administrator/Documents/secret.txt and press Enter. The terminal displays an error stating no such file or directory has been found. This proves that we have unrestricted access to the file system.....	20
Exercise 12: Enumerating Logged on Users Using Finger Protocol.....	21
Exercise 12, Step 6: You will observe that the port 79 is open in the Nmap result, meaning finger service is running on the target machine.	22
Exercise 12, Step 8: Finger client returns the logged in user information such as the login name, name of the user and login time as shown in the screenshot below.....	23
Exercise 12, Step 9: Since we found the username, we shall use this to extract additional information such as the name of the user, home directory, login name, and shell. Type finger Admin@192.168.0.50 and press Enter	24
Exercise 12, Step 11: Type Admin and press Enter. This displays the enumerated user information as shown in the screenshot below	25
Exercise 12, Step 12: To safeguard your machine from returning the logged in user information, it is recommended to disable finger service on the machine by editing the finger text file located in the /etc/xinetd.d	25
Exercise 13: Performing Man-in-the-Middle Attack using Cain & Abel.....	27
Exercise 13, Step 15: Select the added IP address in the Configuration/Routed packets, and click Start/Stop APR (third icon from left) icon. Cain begins ARP poisoning in between these machines.....	28
Exercise 13, Step 20: Now launch a command prompt in the machine, type ftp 172.19.19.9 (IP address of FTP Server machine) and press Enter. When prompted for the Username, type "Martin" and press Enter. When prompted for the password, type "mystery" and press Enter.....	29
Exercise 13, Step 21: You will observe that Cain & Abel captured some packets which can be observed under the Packets field.	30
Exercise 13, Step 22: Click the Passwords tab in the Cain & Abel GUI. Select FTP from the left pane under the Passwords section. You will observe the credentials being captured by Cain & Abel as shown in the screenshot	30
Exercise 14: Auditing a Machine for Weak Passwords Using LophCrack.....	31
Exercise 14, Step 13: In the Job Scheduling window, select Run this job immediately and	

click Next.....	32
Exercise 14, Step 15: A caution box appears regarding changed LC7Agent on the remote machine as shown in the screenshot. Click Yes.	32
Exercise 14, Step 16: LophCrack will begin to decode the hashes. You can see the Progress bar in the lower right-hand corner of the window. Once done with the password auditing, it displays the weak passwords set for the respective user accounts present in Advertisement Dept machine as shown in the screenshot.....	33
Exercise 14, Step 20: To open the saved result, navigate to Desktop and double-click the Credentials.lcs file to view result.....	34
Exercise 15: Automating Penetration Testing Tasks Using Bash Scripting.....	35
Exercise 15, Step 18: Now, minimize the text editor window and maximize the command line terminal. Nmap has performed live host identification on the given IP Address range. Once the live hosts are identified, the script is written in such a way, that a new nmap scan is initiated to find the machines (among the identified live hosts) that have the FTP port open. The live machines with the FTP port open are displayed as shown in the screenshot.....	36
Exercise 14, Step 23: Minimize the text editor window and maximize the command line terminal. On issuing the IP Address, Hydra begins to a perform Dictionary attack on the machine and starts displaying the user credentials as shown in the screenshot.....	37
Exercise 14, Step 30: On issuing the user credentials, you will be logged in to the FTP Server, as shown in the screenshot	38

Executive Summary

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

The objective of this lab is the continuation of internal penetration testing, where tools such as Python, Metasploit, nmap, Finger as well as GUI tools such as Cain and Abel and LophCrack will be leveraged.

Approach:

Starting off with Python in order to grab the banner of the ssh Service in relation for executing reconnaissance on Exercise 8, we have basically broken down a Python script and how it could be used to gain insight on a socket/connection. This means that we basically leverage python from the terminal which allowed us to learn about the importance of Python and how it could be leveraged to gain insight on a remote system or connection. This leads to my observation of how sockets can be used to gain insight on Services with Banner grab.

2nd Approach:

I have also leveraged Metasploit to fingerprint the version of HTTP and determine if there are any possible vulnerabilities. I also was able to observe the configuration files for later modification.

3rd Approach:

This approach I have learned from the lab involves leveraging nmap to gain insight on the SMB. This allowed me to learn about how nmap -sC can gain insight on the IP targeted such as the security mode.

4th Approach:

This approach goes more in depth with nmap from exercise 11, however it showcases how nmap is able to go through the subnet of the IP target for any live machines to target, as well as to look into any misconfigured services, specifically RPC. This allowed me to observe and understand how nmap is not just able to be used for fingerprinting in relation to search up vulnerabilities but is able to actively find the vulnerable systems and highlight them themselves.

5th Approach:

This approach involves the Finger protocol on the terminal, and it has allowed me to observe how the command is able to return logged in user information.

6th Approach:

From exercise 13, I have approached penetration testing using Cain and Able where I learned how to input a scan in between IP addresses to intercept passwords.

7th Approach:

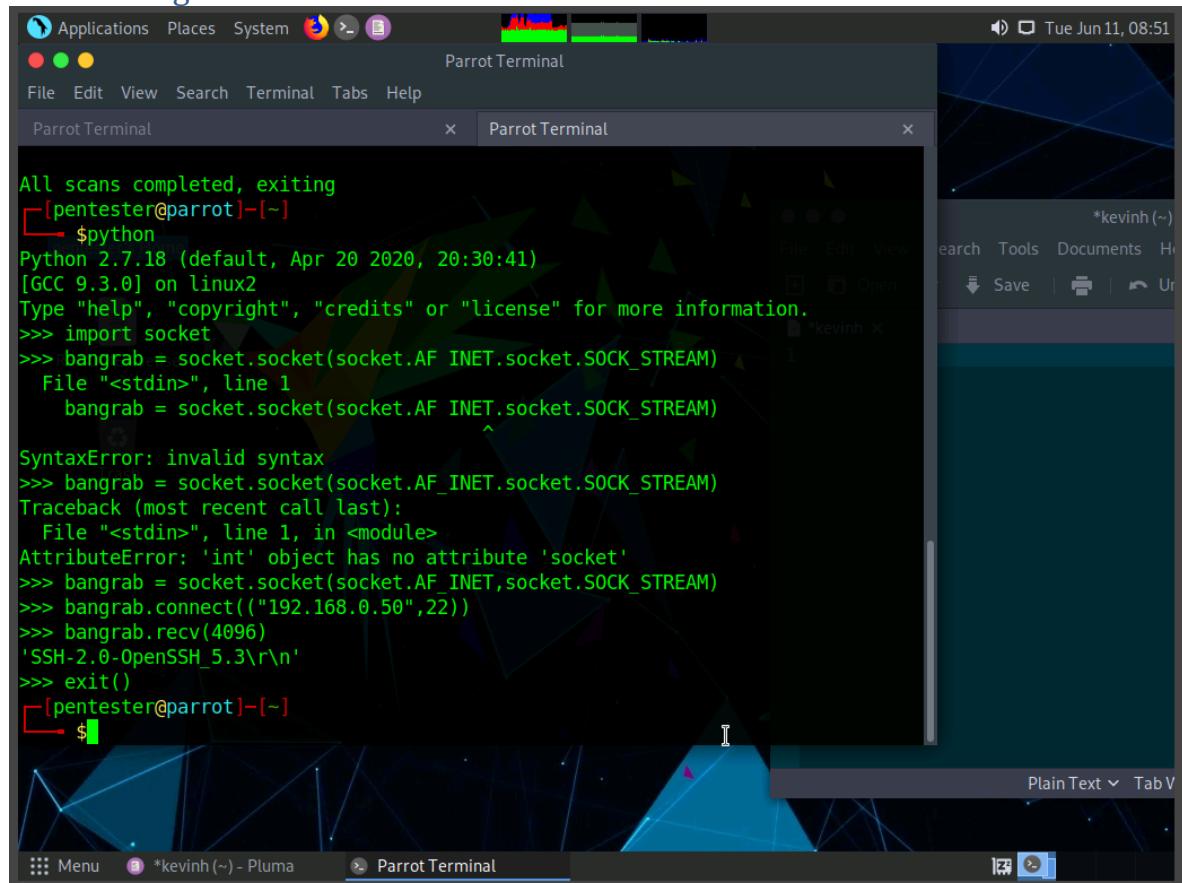
From exercise 14, I have approached the penetration testing methodology using a Brute Force method using LophCrack where I learned about how dictionary attacks can be simplified in a GUI manner to make password cracking easier.

8th Approach: I have approached penetration testing using Bash Scripts instead of Python where I learned several commands as well as insight on some of the Bash Script functionalities for executing penetration testing.

Exercise 8: Create a Python Script to Grab the Banner of the ssh Service

8.1 OUTPUT SCREENSHOTS

Exercise 8, Step 3: Open a terminal window on the Parrot machine and enter the following code



The screenshot shows a terminal window titled "Parrot Terminal" running on a Parrot OS desktop environment. The terminal displays the following Python session:

```
All scans completed, exiting
[pentester@parrot] ~
$ python
Python 2.7.18 (default, Apr 20 2020, 20:30:41)
[GCC 9.3.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import socket
>>> bangrab = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
  File "<stdin>", line 1
    bangrab = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
               ^
SyntaxError: invalid syntax
>>> bangrab = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'int' object has no attribute 'socket'
>>> bangrab = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
>>> bangrab.connect(("192.168.0.50",22))
>>> bangrab.recv(4096)
'SSH-2.0-OpenSSH_5.3\r\n'
>>> exit()
[pentester@parrot] ~
$
```

8.2 Questions

**Network Penetration Testing
Methodology-Internal**

X Save &
Exit

Instructions Resources ?

main()

Usage of the Script

\#python <script_name.py>

```
#!/usr/bin/python
import socket
import sys
import os
#grab the banner
def grab_banner(ip_address,port):
    try:
        s=socket.socket()
        s.connect((ip_address,port))
        p=ip_address + ":" + banner
        except:
            return
    def checkVuln(banner):
        if len(banner) > 2:
            filename = "vuln.txt"
            for line in filename.readlines():
                line = line.strip("\n")
                if banner in line:
                    print "Is vulnerable" + banner
                    else:
                        print "Is not vulnerable"
    def main():
        portlist = [21,22,25,80,110]
        for i in range(255):
            for port in portlist:
                ip_address = "192.168.0." + str(i) # change
                grab_banner(ip_address, port)
                _name_... == __main__
    main()
    Usage Of The Script
    # python <script_name.py>
```

✓ 7. Please see the appendix for select coding examples.

Question 6.8.1

Create a Python script to grab the banner of the secure shell (SSH) service. Flag submission is not required for this task; enter "No flag" as the answer.

No flag

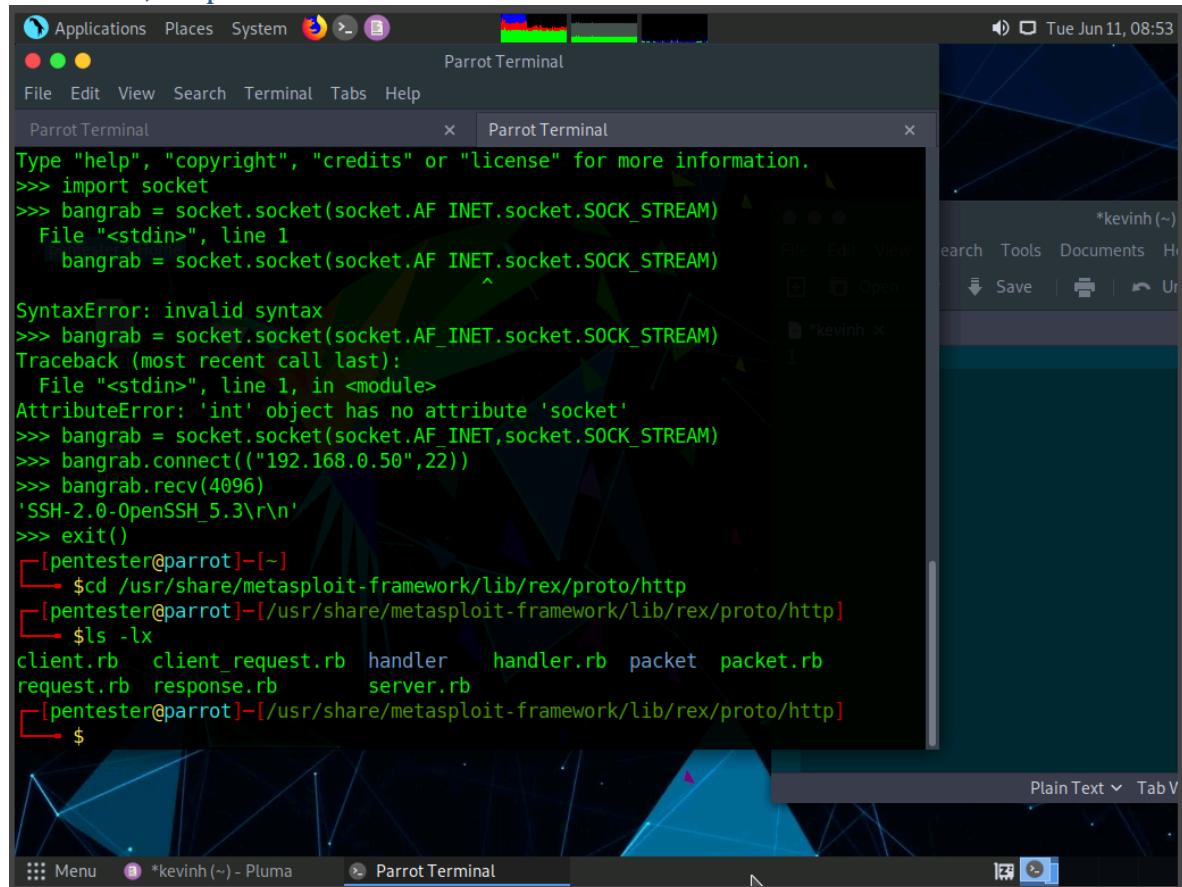
Score

✓ Correct

Exercise 9: Use Metasploit to Detect Version of HTTP

9.1 OUTPUT SCREENSHOTS

Exercise 9, Step 5: Enter ls -lx

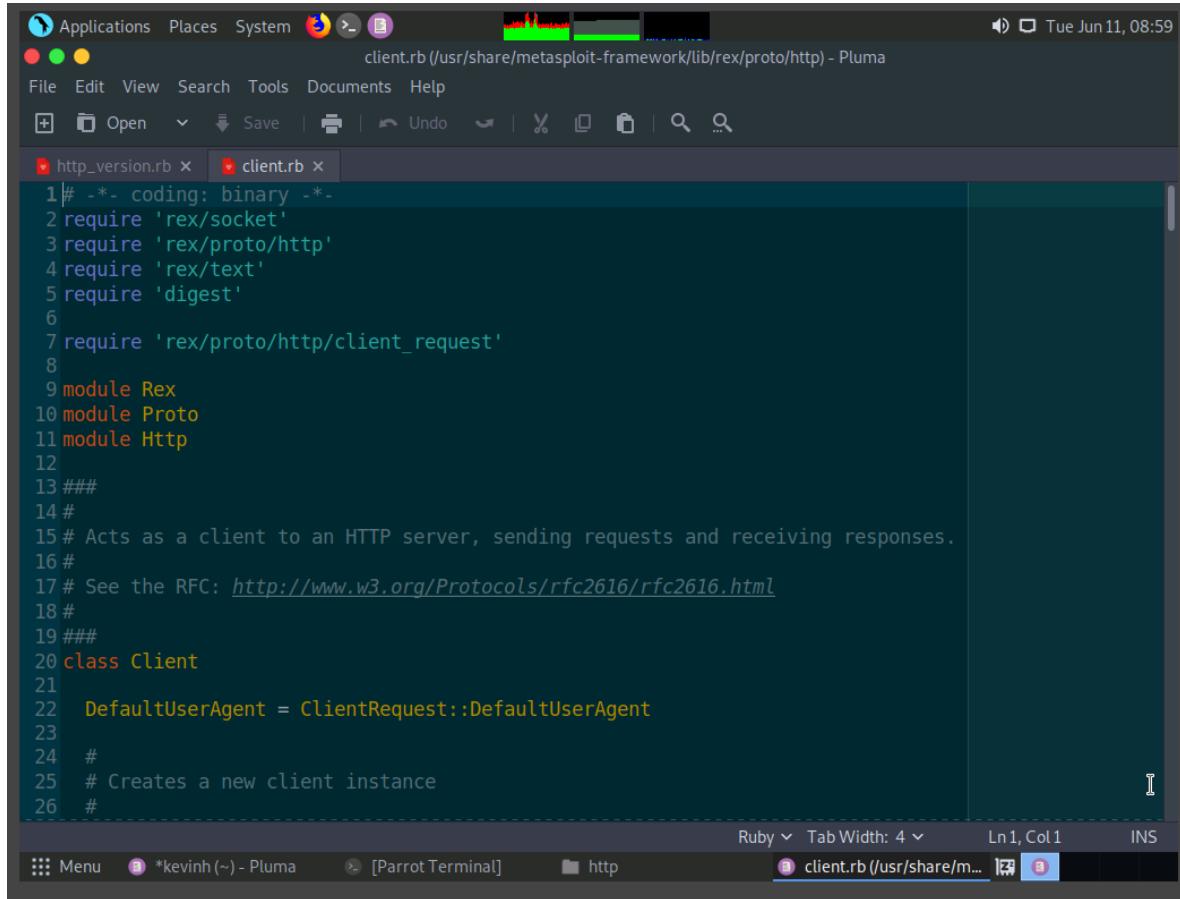


The screenshot shows a Parrot OS desktop environment. In the foreground, a terminal window titled "Parrot Terminal" is open. The terminal displays the following session:

```
Type "help", "copyright", "credits" or "license" for more information.
>>> import socket
>>> bangrab = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
  File "<stdin>", line 1
    bangrab = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
               ^
SyntaxError: invalid syntax
>>> bangrab = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'int' object has no attribute 'socket'
>>> bangrab = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
>>> bangrab.connect(("192.168.0.50",22))
>>> bangrab.recv(4096)
'SSH-2.0-OpenSSH_5.3\r\n'
>>> exit()
[pentester@parrot]~]
└─$ cd /usr/share/metasploit-framework/lib/rex/proto/http
[pentester@parrot]~/metasploit-framework/lib/rex/proto/http]
└─$ ls -lx
client.rb  client_request.rb  handler  handler.rb  packet  packet.rb
request.rb  response.rb      server.rb
[pentester@parrot]~/metasploit-framework/lib/rex/proto/http]
└─$
```

The terminal window is part of the "Pluma" application, as indicated by the window title and taskbar icon.

Exercise 9, Step 11: An example of an excerpt of the code is shown in the screenshot



The screenshot shows a Linux desktop environment with a terminal window titled "client.rb (/usr/share/metasploit-framework/lib/rex/proto/http) - Pluma". The terminal window contains the following Ruby code:

```
1#-*- coding: binary -*-
2require 'rex/socket'
3require 'rex/proto/http'
4require 'rex/text'
5require 'digest'
6
7require 'rex/proto/http/client_request'
8
9module Rex
10module Proto
11module Http
12
13###
14#
15# Acts as a client to an HTTP server, sending requests and receiving responses.
16#
17# See the RFC: http://www.w3.org/Protocols/rfc2616/rfc2616.html
18#
19###
20class Client
21
22  DefaultUserAgent = ClientRequest::DefaultUserAgent
23
24  #
25  # Creates a new client instance
26  #
```

The terminal window also shows the following status bar information: Ruby ▾ Tab Width: 4 ▾ Ln1, Col1 INS. The bottom of the window shows tabs for "Menu", "*kevinh (~) - Pluma", "[Parrot Terminal]", and "http".

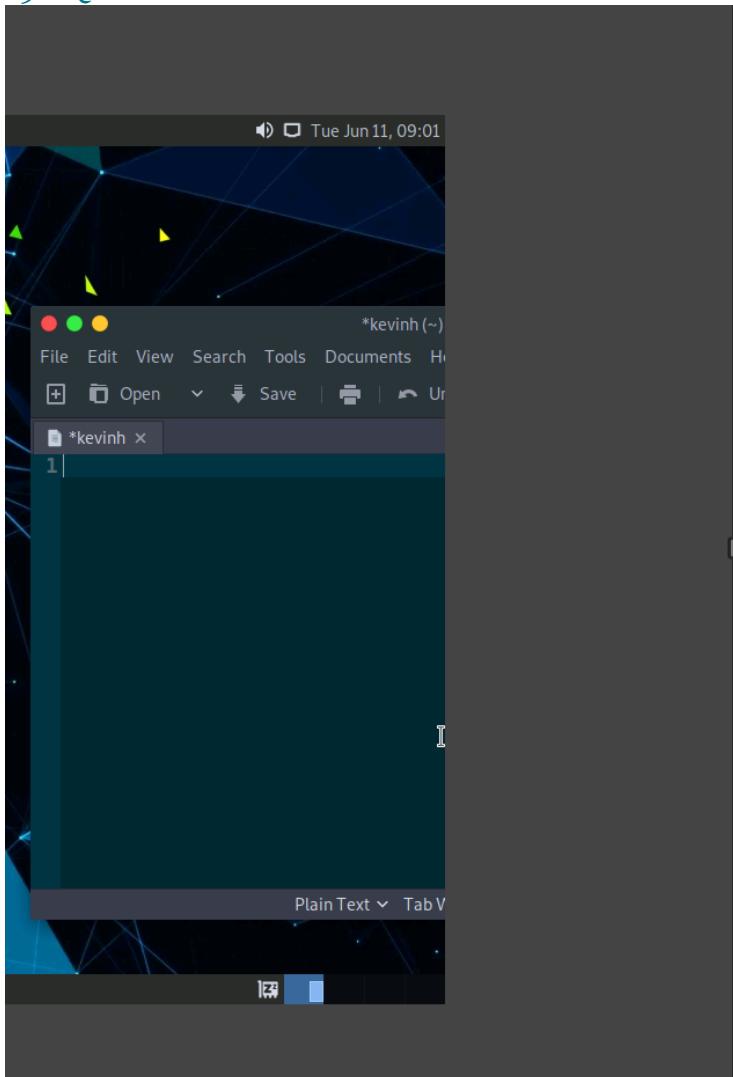
Exercise 9, Step 12: The key to this routine is in the defined class:

The screenshot shows a Linux desktop interface with a terminal window and a code editor window. The terminal window is titled 'Parrot Terminal' and contains a tooltip pointing to line 46 of the code in the editor. The code editor window is titled 'client.rb [Read-Only] (/usr/share/metasploit-framework/lib/rex/proto/http) - Pluma' and displays the following Ruby code:

```
24  #
25  # Creates a new client instance
26  #
27  def initialize(host, port = 80, context = {}, ssl = nil, ssl_version = nil, proxies = nil,
28    username = '', password = '')
29    self.hostname = host
30    self.port     = port.to_i
31    self.context  = context
32    self.ssl      = ssl
33    self.ssl_version = ssl_version
34    self.proxies = proxies
35    self.username = username
36    self.password = password
37  # Take ClientRequest's defaults, but override with our own
38  self.config = Http::ClientRequest::DefaultConfig.merge({
39    'read_max_data' => (1024*1024*1),
40    'vhost'           => self.hostname,
41  })
42
43  # XXX: This info should all be controlled by ClientRequest
44  self.config_types = {
45    'uri_encode_mode' => ['hex-normal', 'hex-all',
46      'normal', 'u-random', 'u-all'],
47    'uri_encode_count' => 'integer'.

```

9.2 QUESTIONS



The terminal window shows a single character '1' entered at the prompt. The window title is '*kevinh (~)'.

Network Penetration testing
Methodology-Internal

Save & Exit

Instructions Resources

- o self.context = context
- o self.ssl = ssl
- o self.ssl_version = ssl_version
- o self.proxies = proxies
- o self.username = username
- o self.password = password

13. Once you have reviewed the file, close all open windows. As the class shows, you have covered most requirements when acting as a client for a web server.

14. This is the process you should follow when you are working as a practitioner and professional security and penetration tester. Always investigate the code that is being used BEFORE you ever deploy it on a site.

15. This concludes the lab exercise.

Question 6.9.1

Use the Metasploit tool to detect the HTTP version. Flag submission is not required for this task; enter "No flag" as the answer.

No flag

Score

✓ Correct

← Previous Next →

Exercise 10: Enumerating SMB

10.1 OUTPUT SCREENSHOTS

Exercise 10, Step 5: In the terminal window, type nmap -sC 192.168.0.7 and press Enter

The screenshot shows a terminal window titled "Parrot Terminal" running on a Parrot OS desktop environment. The terminal displays the results of an nmap scan against the IP address 192.168.0.7. The output includes:

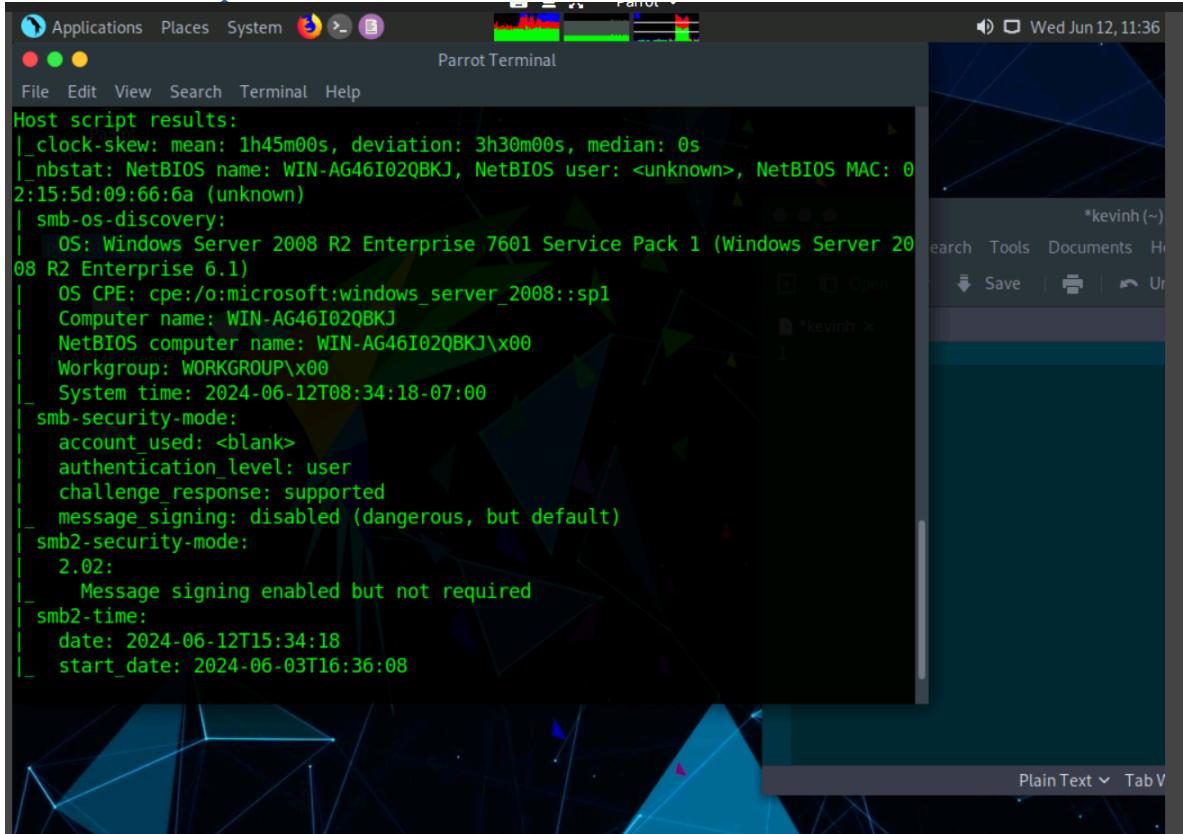
```
File Edit View Terminal Help
3389/tcp open ms-wbt-server
49152/tcp open unknown
49153/tcp open unknown
49154/tcp open unknown
49155/tcp open unknown
49156/tcp open unknown
49159/tcp open unknown
MAC Address: 02:15:5D:09:66:6A (Unknown)

Host script results:
| smb-os-discovery:
|_ OS: Windows Server 2008 R2 Enterprise 7601 Service Pack 1 (Windows Server 2008 R2 Enterprise 6.1)
|_ OS CPE: cpe:/o:microsoft:windows_server_2008::sp1
| Computer name: WIN-AG46I02QBKJ
| NetBIOS computer name: WIN-AG46I02QBKJ\x00
| Workgroup: WORKGROUP\x00
|_ System time: 2024-06-12T08:33:56-07:00

Nmap done: 1 IP address (1 host up) scanned in 6.24 seconds
[pentester@parrot] ~
$ nmap -sC 192.168.0.7
Starting Nmap 7.80 ( https://nmap.org ) at 2024-06-12 11:34 EDT
```

The terminal window has a dark blue background with a geometric pattern. The status bar at the bottom right shows "Plain Text" and "Tab V". The desktop environment includes a dock with icons for Applications, Places, System, and a terminal icon.

Exercise 10, Step 6: The output of the command in step 5 reveals more details than that of the command in step 4. The scan may take approximately 5 to 10 minutes to complete.



The screenshot shows a terminal window titled "Parrot Terminal" running on Parrot OS. The window displays the output of a host script, specifically focusing on SMB security mode. The output includes details such as the operating system (Windows Server 2008 R2 Enterprise), CPE information, computer name, NetBIOS computer name, workgroup, system time, and SMB security modes (smb-security-mode and smb2-security-mode). The terminal interface has a dark theme with green text, and the background features a blue and purple geometric pattern. The top right corner of the screen shows the date and time as "Wed Jun 12, 11:36".

```
Host script results:
|_clock-skew: mean: 1h45m00s, deviation: 3h30m00s, median: 0s
|_nbstat: NetBIOS name: WIN-AG46I02QBKJ, NetBIOS user: <unknown>, NetBIOS MAC: 0
|2:15:5d:09:66:6a (unknown)
| smb-os-discovery:
|   OS: Windows Server 2008 R2 Enterprise 7601 Service Pack 1 (Windows Server 2008 R2 Enterprise 6.1)
|   OS CPE: cpe:/o:microsoft:windows_server_2008::sp1
|   Computer name: WIN-AG46I02QBKJ
|   NetBIOS computer name: WIN-AG46I02QBKJ\x00
|   Workgroup: WORKGROUP\x00
|   System time: 2024-06-12T08:34:18-07:00
| smb-security-mode:
|   account_used: <blank>
|   authentication_level: user
|   challenge_response: supported
|   message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|   2.02:
|       Message signing enabled but not required
| smb2-time:
|   date: 2024-06-12T15:34:18
|   start_date: 2024-06-03T16:36:08
```

10.2 QUESTIONS

The screenshot shows a terminal window on the left and a penetration testing exercise interface on the right.

Terminal Window (Left):

- Time: Wed Jun 12, 11:36
- User: *kevinh (~)
- Commands: search, Tools, Documents, Help
- File Operations: Save, Print, Undo
- Text Area: Plain Text, Tab View

Exercise Interface (Right):

Network Penetration Testing Methodology-Internal

Save & Exit

Instructions Resources ? ⚙️

Perform SMB enumeration on the target IP address, 192.168.0.7, using the Nmap tool. Enter the http-title of the web page hosting the IP address 192.168.0.7.

IIS7

Score

Correct

Question 6.2.4

Perform SMB enumeration on the target IP address, 192.168.0.7, using the Nmap tool. Are the login attempts successful on the target machine (Yes/No)?

No

Score

Correct

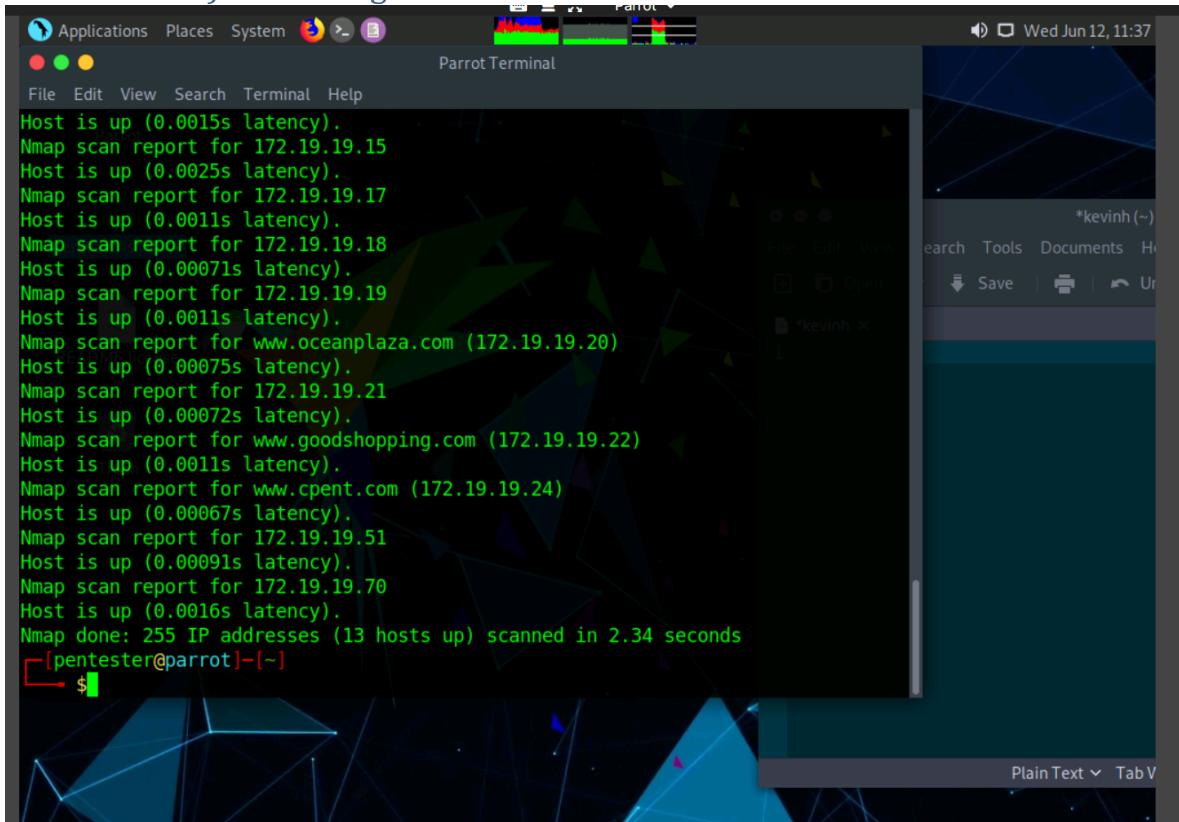
← Previous Next →

1 Hr 5 Min Remaining

Exercise 11: Pentesting Misconfigured RPC Service and NFS Shares

11.1 OUTPUT SCREENSHOTS

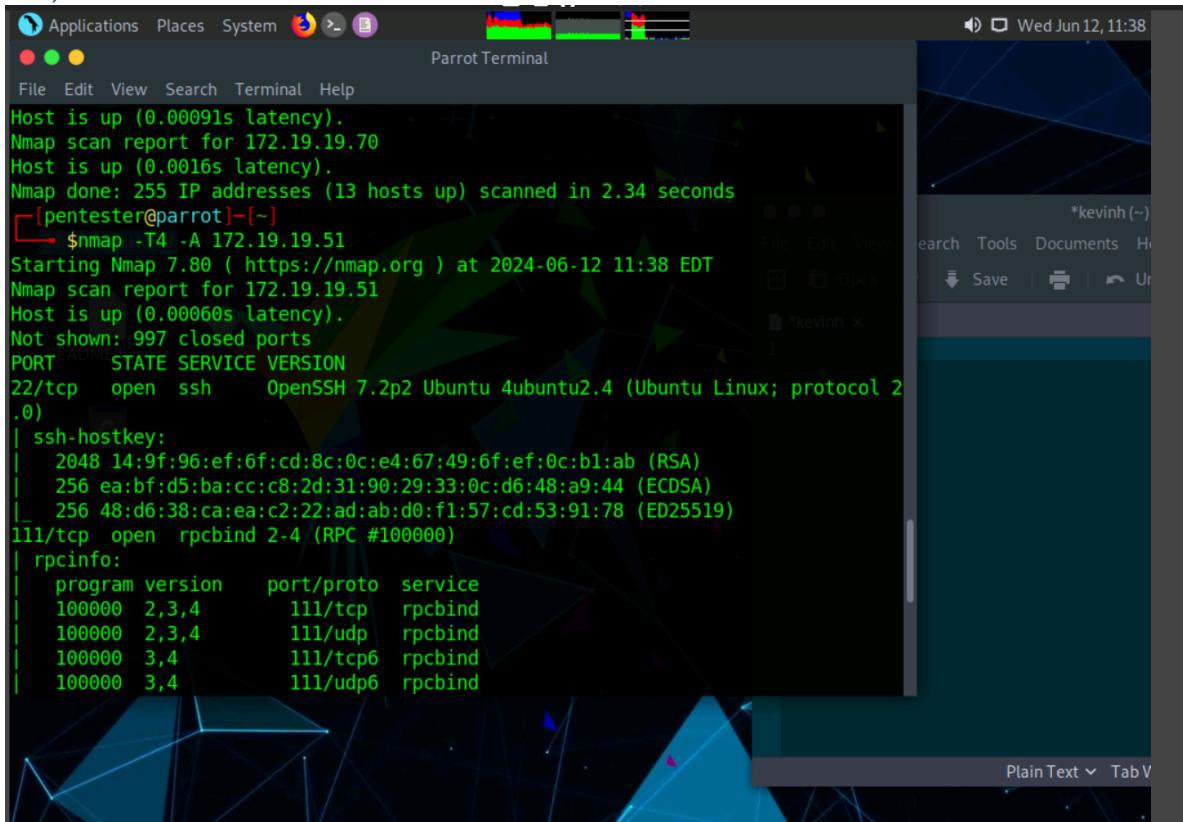
Exercise 11, Step 3: In this lab, we will be scanning a subnet for live machines. Select one machine and pentest the machine to gain access to it. For doing a quick scan, we will do a ping sweep using Nmap. In this lab, we are choosing an internal network for pentesting. Launch a command line terminal, 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 172.19.19.51 (RPC Server Ubuntu) as our target.



The screenshot shows a Parrot OS desktop environment. In the center is a terminal window titled "Parrot Terminal". The terminal displays the output of an Nmap ping sweep command. The output shows 13 hosts up from a total of 255 scanned IP addresses. The hosts listed are 172.19.19.15, 172.19.19.17, 172.19.19.18, 172.19.19.19, 172.19.19.20, 172.19.19.21, 172.19.19.22, 172.19.19.24, 172.19.19.51, 172.19.19.70, and 172.19.19.70 again. The command used was "nmap -sP 172.19.19.1-255". The terminal window has a dark background with a green and blue geometric pattern. The desktop environment includes a menu bar with "Applications", "Places", "System", and a system tray icon for "Parrot". A file manager window is visible in the background, showing a file named "kevinh".

```
Host is up (0.0015s latency).
Nmap scan report for 172.19.19.15
Host is up (0.0025s latency).
Nmap scan report for 172.19.19.17
Host is up (0.0011s latency).
Nmap scan report for 172.19.19.18
Host is up (0.00071s latency).
Nmap scan report for 172.19.19.19
Host is up (0.0011s latency).
Nmap scan report for www.oceanplaza.com (172.19.19.20)
Host is up (0.00075s latency).
Nmap scan report for 172.19.19.21
Host is up (0.00072s latency).
Nmap scan report for www.goodshopping.com (172.19.19.22)
Host is up (0.0011s latency).
Nmap scan report for www.cpent.com (172.19.19.24)
Host is up (0.00067s latency).
Nmap scan report for 172.19.19.51
Host is up (0.00091s latency).
Nmap scan report for 172.19.19.70
Host is up (0.0016s latency).
Nmap done: 255 IP addresses (13 hosts up) scanned in 2.34 seconds
[pentester@parrot]~$
```

Exercise 11, Step 5: Nmap takes around 30 seconds to complete the scan. On completing the scan, you will observe that the services rpc, ftp, nfs and mountd are running on the victim machine. From the scan, it is observed that an NFS File system is mounted on the remote machine. In this lab, we shall focus on the RPC, NFS and mountd services



The screenshot shows a Parrot OS desktop environment. A terminal window titled "Parrot Terminal" is open, displaying the output of an Nmap scan. The terminal window has a dark background with green text. The desktop interface includes a top bar with icons for Applications, Places, System, and a volume slider. To the right of the terminal, there is a file manager window showing a single file named "kevinh". The bottom right corner of the screen shows the text "Plain Text Tab V".

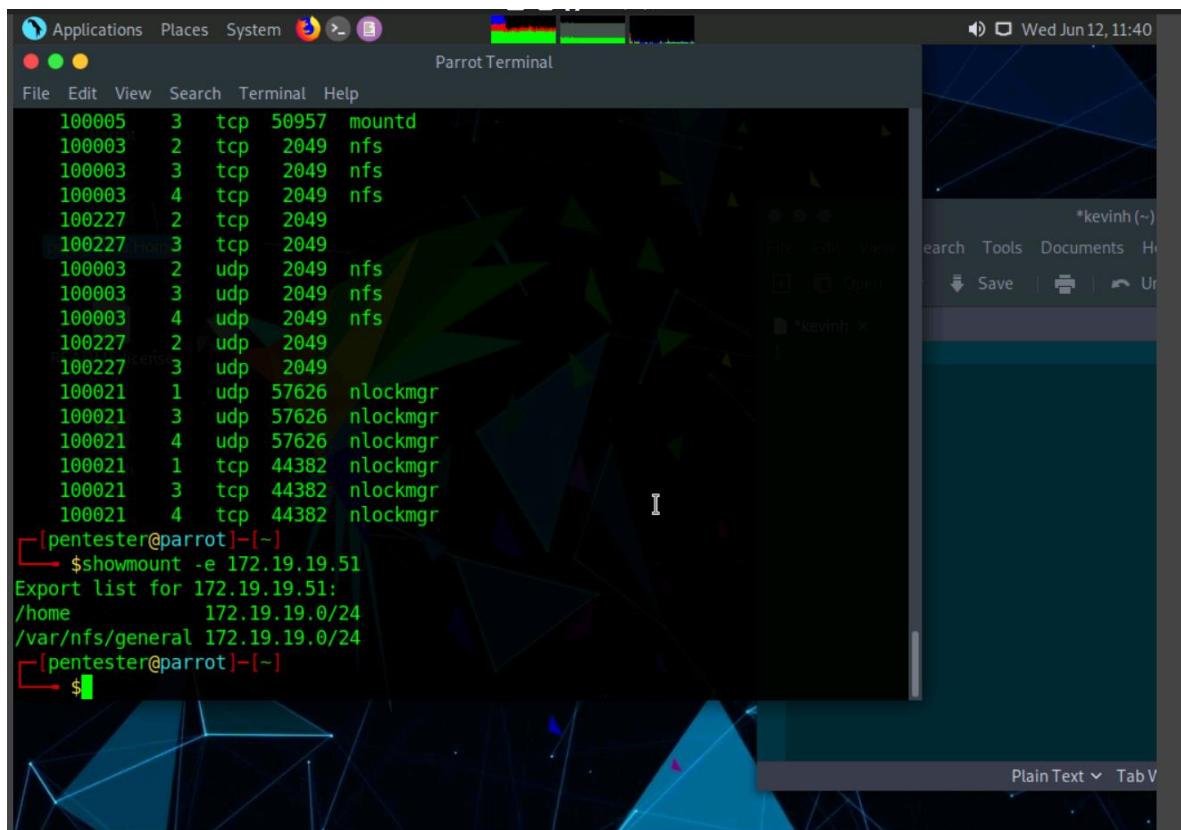
```
Host is up (0.00091s latency).
Nmap scan report for 172.19.19.70
Host is up (0.0016s latency).
Nmap done: 255 IP addresses (13 hosts up) scanned in 2.34 seconds
[pentester@parrot]~[~]
└─ $nmap -T4 -A 172.19.19.51
Starting Nmap 7.80 ( https://nmap.org ) at 2024-06-12 11:38 EDT
Nmap scan report for 172.19.19.51
Host is up (0.00060s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.4 (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)
111/tcp   open  rpcbind 2-4 (RPC #100000)
| rpcinfo:
|   program version  port/proto  service
|   100000  2,3,4      111/tcp    rpcbind
|   100000  2,3,4      111/udp   rpcbind
|   100000  3,4       111/tcp6   rpcbind
|   100000  3,4       111/udp6   rpcbind
```

Exercise 11, Step 7: We observe that nfs and mountd services are active on the remote machine

The screenshot shows a Parrot OS desktop environment. In the foreground, a terminal window titled "Parrot Terminal" is open, displaying the output of the "netstat -an" command. The output lists numerous network connections, primarily on port 2049, which is associated with the "nfs" service. Other connections listed include mountd and nlockmgr services. The terminal window has a red bracket highlighting the command prompt and the beginning of the output. The background features a dark, abstract geometric wallpaper.

```
File Edit View Search Terminal Help
100005 1 udp 43733 mountd
100005 1 tcp 58247 mountd
100005 2 udp 60464 mountd
100005 2 tcp 54613 mountd
100005 3 udp 49862 mountd
100005 3 tcp 50957 mountd
100003 2 tcp 2049 nfs
100003 3 tcp 2049 nfs
100003 4 tcp 2049 nfs
100227 2 tcp 2049
100227 3 tcp 2049
100003 2 udp 2049 nfs
100003 3 udp 2049 nfs
100003 4 udp 2049 nfs
100227 2 udp 2049
100227 3 udp 2049
100021 1 udp 57626 nlockmgr
100021 3 udp 57626 nlockmgr
100021 4 udp 57626 nlockmgr
100021 1 tcp 44382 nlockmgr
100021 3 tcp 44382 nlockmgr
100021 4 tcp 44382 nlockmgr
[pentester@parrot] ~]$
```

Exercise 11, Step 8: Now, we shall issue the showmount command to discover NFS shares listed in /etc/exports file of the remote machine. Type showmount -e 172.19.19.51 and press Enter. This will display all the NFS shares on the remote machine as shown in the screenshot below

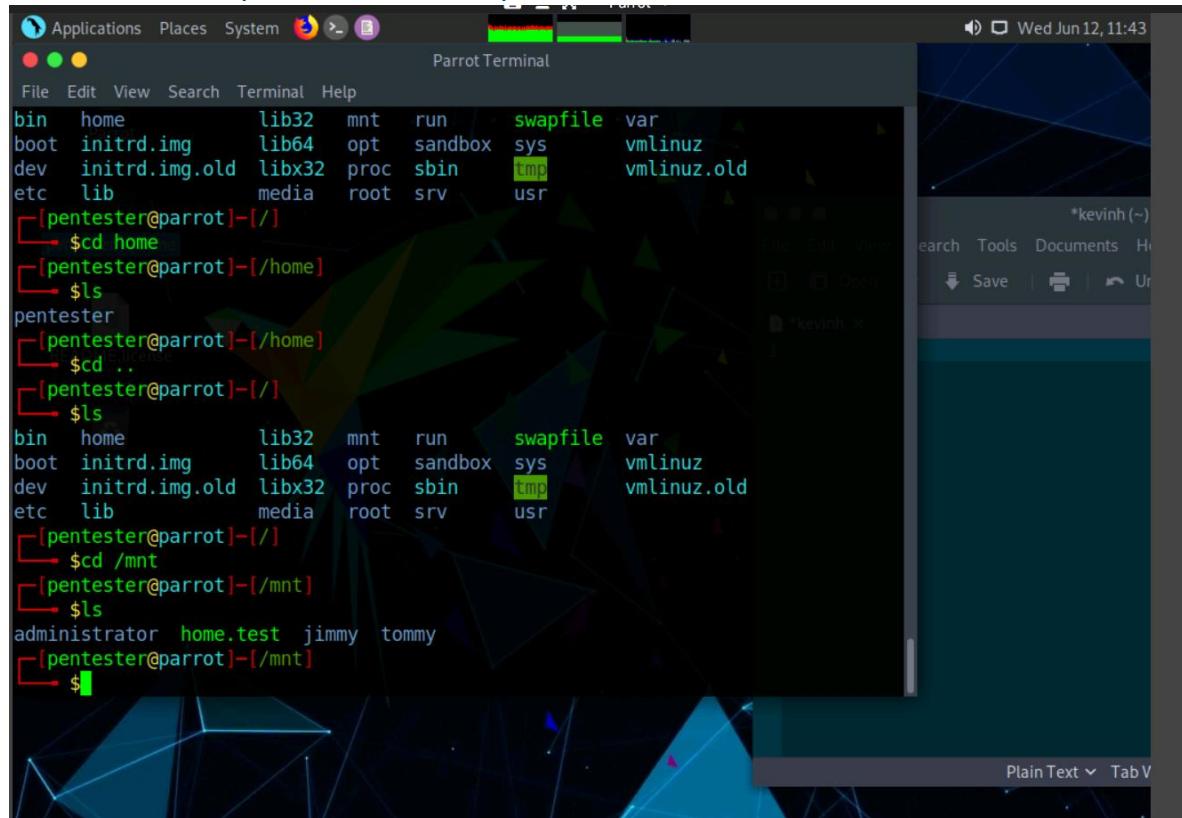


The screenshot shows a terminal window titled "Parrot Terminal" running on a Parrot OS desktop environment. The terminal displays several lines of network traffic or log entries, followed by the command \$showmount -e 172.19.19.51, which lists exports for the IP address 172.19.19.51. The output shows two exports: /home (IP 172.19.19.0/24) and /var/nfs/general (IP 172.19.19.0/24). The terminal prompt [pentester@parrot]~\$ is visible at the bottom.

```
100005 3 tcp 50957 mountd
100003 2 tcp 2049 nfs
100003 3 tcp 2049 nfs
100003 4 tcp 2049 nfs
100227 2 tcp 2049
100227 3 tcp 2049
100003 2 udp 2049 nfs
100003 3 udp 2049 nfs
100003 4 udp 2049 nfs
100227 2 udp 2049
100227 3 udp 2049
100021 1 udp 57626 nlockmgr
100021 3 udp 57626 nlockmgr
100021 4 udp 57626 nlockmgr
100021 1 tcp 44382 nlockmgr
100021 3 tcp 44382 nlockmgr
100021 4 tcp 44382 nlockmgr
[pentester@parrot]~$
$ showmount -e 172.19.19.51
Export list for 172.19.19.51:
/home          172.19.19.0/24
/var/nfs/general 172.19.19.0/24
[pentester@parrot]~$
$
```

Exercise 11, Step 9: As we saw in the previous task, the /home file system was shared on the remote machine. We will be mounting this file system on the Parrot machine to the mnt directory. To mount, type sudo mount -t nfs 172.19.19.51:/home /mnt -o nolock and press Enter. Type toor and press Enter when prompted.

Exercise 11, Step 11: Type ls and press Enter to view the files and directories contained in the /home folder i.e., /mnt.



The screenshot shows a terminal window titled "Parrot Terminal" running on a Parrot OS desktop environment. The terminal displays a command-line session where the user navigates through the file system:

```
[pentester@parrot]~$ cd home
[pentester@parrot]~/home$ ls
pentester
[pentester@parrot]~/home$ cd ..
[pentester@parrot]~/~$ ls
bin  home  lib32  mnt  run  swapfile  var
boot initrd.img  lib64  opt  sandbox  sys  vmlinuz
dev  initrd.img.old  libx32  proc  sbin  tmp  vmlinuz.old
etc  lib  media  root  srv  usr

[pentester@parrot]~$ cd /mnt
[pentester@parrot]~/mnt$ ls
administrator  home.test  jimmy  tommy

[pentester@parrot]~/mnt$
```

The terminal window has a dark theme with green text for output and red text for errors. The desktop background features a blue and green abstract geometric pattern. A file manager window titled "kevinh (~)" is visible in the background, showing a list of files in the current directory.

Exercise 11, Step 13: On entering the command in the previous task, the cat command displays the file contents in the secret.txt file successfully, meaning we have successfully mounted the remote file system and accessed the contents in it.

The screenshot shows a Parrot Terminal window with the following session log:

```
[pentester@parrot]~$ cd home
[pentester@parrot]~/home$ ls
pentester's Home
[pentester@parrot]~/home$ cd ..
[pentester@parrot]~$ ls
bin  home      lib32   mnt   run    swapfile  var
boot initrd.img  lib64   opt   sandbox  sys    vmlinuz
dev  initrd.img.old libx32  proc  sbin    tmp    vmlinuz.old
etc  lib       media   root  srv    usr
[pentester@parrot]~$ cd /mnt
[pentester@parrot]~/mnt$ ls
administrator  home.test  jimmy  tommy
[pentester@parrot]~/mnt$ cat administrator/Documents/secret.txt
my account number: 1234567890
[pentester@parrot]~/mnt$
```

The terminal window is titled "Parrot Terminal" and is running on a Parrot OS desktop environment. A file browser window is visible in the background, showing a file named "secret.txt". The terminal output shows the user navigating to the mounted directory at /mnt and reading the contents of the secret.txt file, which contains the text "my account number: 1234567890".

Exercise 11, Step 15: Now, we shall see if we are able to tamper/delete the files in the remote file system. Type rm administrator/Documents/secret.txt and press Enter. Type y and press Enter to confirm the deletion. To confirm that the file has been successfully deleted, type cat administrator/Documents/secret.txt and press Enter. The terminal displays an error stating no such file or directory has been found. This proves that we have unrestricted access to the file system

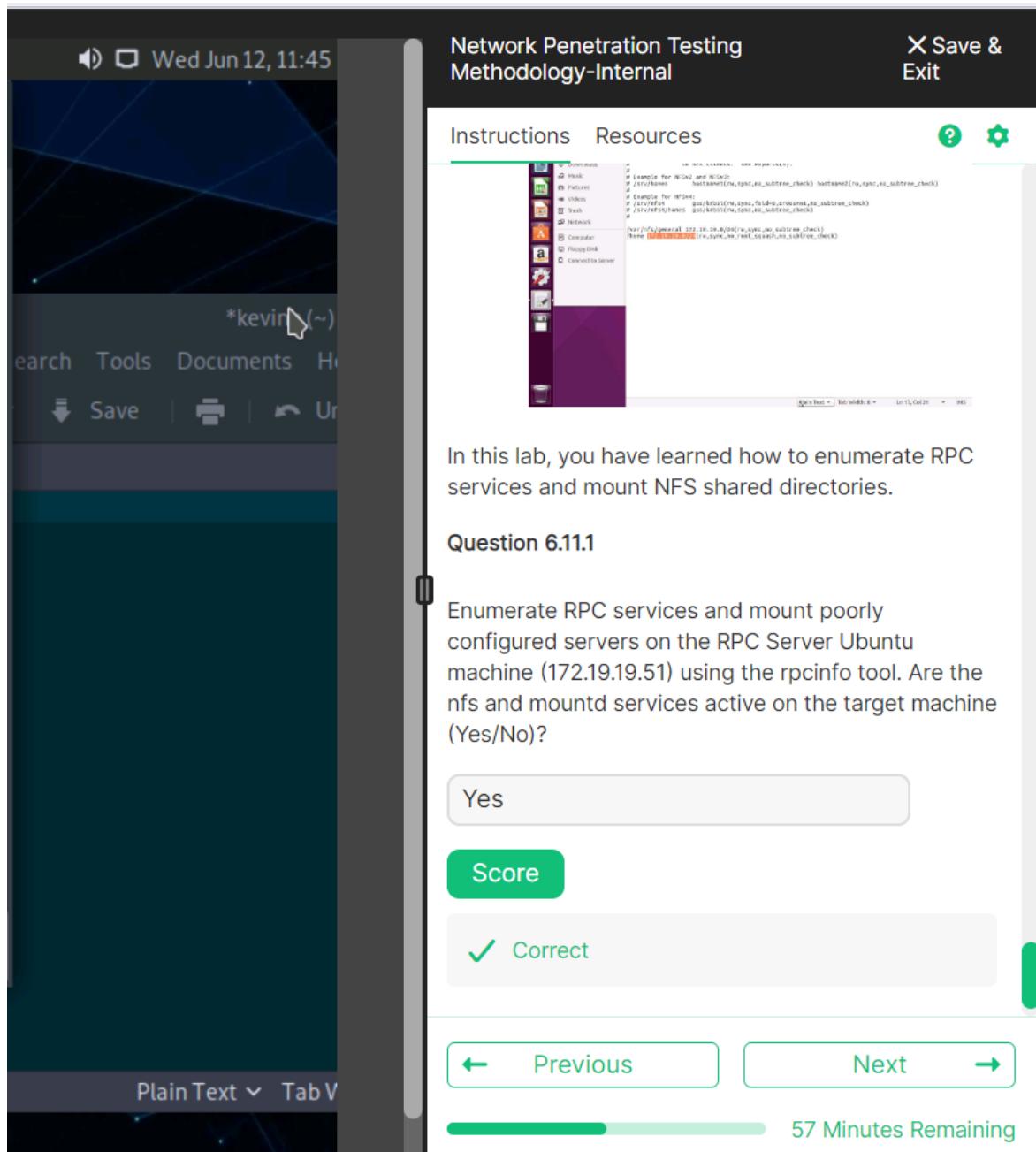
The screenshot shows a Parrot OS desktop environment. On the left, a terminal window titled "Parrot Terminal" is open, displaying a session where the user gains root privileges and deletes a file. The terminal history is as follows:

```
[pentester@parrot]~$ ls
pentester
[pentester@parrot]~$ cd ..
[pentester@parrot]~/[]
[pentester@parrot]~/[]
$ ls
bin  home      lib32  mnt  run  swapfile  var
boot initrd.img  lib64  opt  sandbox  sys  vmlinuz
dev  initrd.img.old  libx32 proc  sbin    tmp  vmlinuz.old
etc  lib        media   root  srv     usr

[pentester@parrot]~/[]
$ cd /mnt
[pentester@parrot]~/mnt[]
$ ls
administrator  home.test  jimmy  tommy
[pentester@parrot]~/mnt[]
$ cat administrator/Documents/secret.txt
my account number: 1234567890
[pentester@parrot]~/mnt[]
$ rm administrator/Documents/secret.txt
rm: remove write-protected regular file 'administrator/Documents/secret.txt'?
[pentester@parrot]~/mnt[]
$
```

On the right, a file manager window titled "kevinh (-)" is visible, showing a list of files in a folder.

11.2 QUESTIONS



Exercise 12: Enumerating Logged on Users Using Finger Protocol

12.1 OUTPUT SCREENSHOTS

Exercise 12, Step 6: You will observe that the port 79 is open in the Nmap result, meaning finger service is running on the target machine.

The screenshot shows a Kali Linux desktop environment. In the foreground, a terminal window titled "Parrot Terminal" is open, displaying a session where the user has gained root privileges (pentester@parrot). The user navigates to the mounted directory at /mnt, finds a file named "secret.txt", reads its contents (containing the administrator's account number: 1234567890), and then deletes it. The user then runs an Nmap scan on the host 192.168.0.50, specifically targeting port 79. The output shows that port 79 is open and fingerprinted as "finger".

```
[pentester@parrot]~$ cd /mnt
[pentester@parrot]~/mnt$ ls
administrator home.test jimmy tommy
[pentester@parrot]~/mnt$ cat administrator/Documents/secret.txt
my account number: 1234567890
[pentester@parrot]~/mnt$ rm administrator/Documents/secret.txt
rm: remove write-protected regular file 'administrator/Documents/secret.txt'?
[pentester@parrot]~/mnt$ nmap -p 79 192.168.0.50
Starting Nmap 7.80 ( https://nmap.org ) at 2024-06-13 15:14 EDT
Nmap scan report for 192.168.0.50
Host is up (0.0017s latency).

PORT      STATE SERVICE
79/tcp    open  finger

Nmap done: 1 IP address (1 host up) scanned in 0.16 seconds
[pentester@parrot]~/mnt$
```

Exercise 12, Step 8: Finger client returns the logged in user information such as the login name, name of the user and login time as shown in the screenshot below

The screenshot shows a Parrot OS desktop environment. In the center is a terminal window titled "Parrot Terminal" with the command-line interface visible. To the right of the terminal is a file viewer window titled "Pluma" showing a file named "secret.txt". The terminal output includes commands like "cat", "rm", and "nmap", and a finger command that lists user information.

```
administrator home.test jimmy tommy
[pentester@parrot]~[~/mnt]
└── $ cat administrator/Documents/secret.txt
my account number: 1234567890
[pentester@parrot]~[~/mnt]
└── $ rm administrator/Documents/secret.txt
rm: remove write-protected regular file 'administrator/Documents/secret.txt'?
[pentester@parrot]~[~/mnt]
└── $ nmap -p 79 192.168.0.50
Starting Nmap 7.80 ( https://nmap.org ) at 2024-06-13 15:14 EDT
Nmap scan report for 192.168.0.50
Host is up (0.0017s latency).

PORT      STATE SERVICE
79/tcp    open  finger

Nmap done: 1 IP address (1 host up) scanned in 0.16 seconds
[pentester@parrot]~[~/mnt]
└── $finger @192.168.0.50
Login      Name      Tty      Idle  Login Time   Office     Office Phone
Admin      Admin      *:0      Jun  3 07:45
Admin      Admin      pts/0      3 Jun  3 07:45 (:0)
[pentester@parrot]~[~/mnt]
└── $
```

The file viewer window shows the contents of "secret.txt" which contains the text "my account number: 1234567890".

Exercise 12, Step 9: Since we found the username, we shall use this to extract additional information such as the name of the user, home directory, login name, and shell. Type finger Admin@192.168.0.50 and press Enter

The screenshot shows a Parrot OS desktop environment. In the foreground, a terminal window titled "Parrot Terminal" is open, displaying the output of a Nmap scan. The output shows a host at 192.168.0.50 is up and listening on port 79/tcp. A finger command is run, showing that the user "Admin" is logged in via TTY *:0, last seen on June 3 at 07:45. The finger command also provides the user's name (Admin), shell (/bin/bash), and login history, indicating they were last on since June 3 at 07:45 and are currently idle.

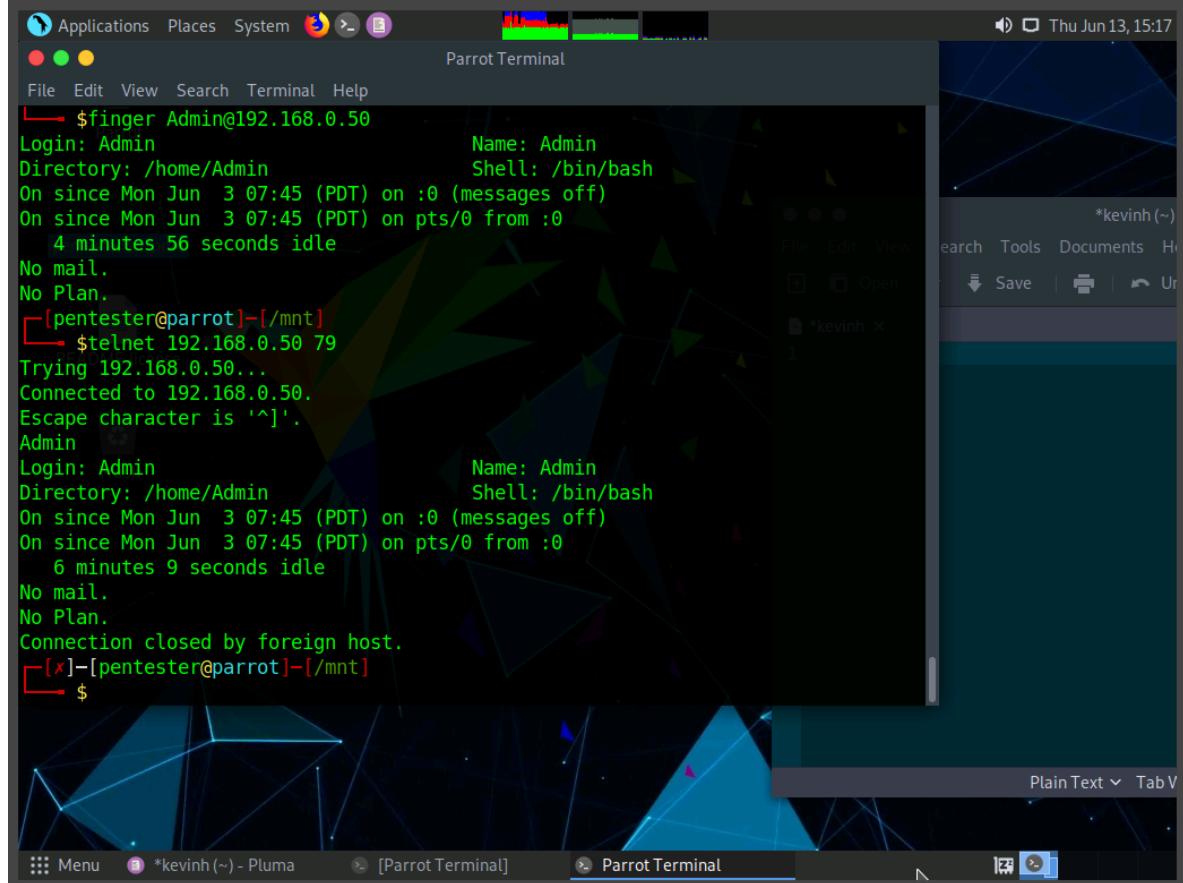
```
Starting Nmap 7.80 ( https://nmap.org ) at 2024-06-13 15:14 EDT
Nmap scan report for 192.168.0.50
Host is up (0.0017s latency).

PORT      STATE SERVICE
79/tcp    open  finger

Nmap done: 1 IP address (1 host up) scanned in 0.16 seconds
[pentester@parrot]:~$finger @192.168.0.50
Login: Admin          Name: Admin
Directory: /home/Admin   Shell: /bin/bash
On since Mon Jun  3 07:45 (PDT) on :0 (messages off)
On since Mon Jun  3 07:45 (PDT) on pts/0 from :
        4 minutes 56 seconds idle
No mail.
No Plan.
[pentester@parrot]:~$
```

In the background, a Pluma file editor window titled "*kevinh (~)" is visible. The status bar of the terminal window indicates "Click to start dragging '*kevinh (~) - Pluma'".

Exercise 12, Step 11: Type Admin and press Enter. This displays the enumerated user information as shown in the screenshot below



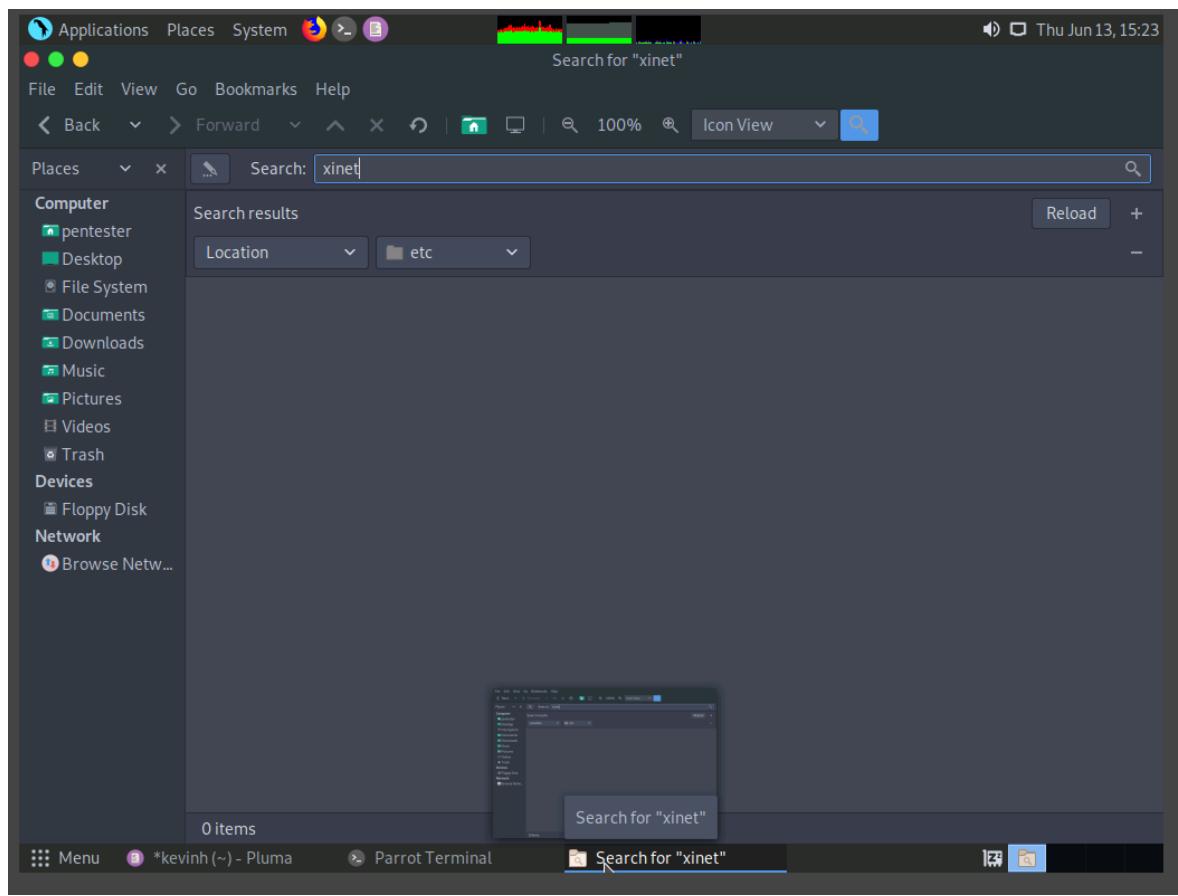
The screenshot shows a Parrot OS desktop environment. In the center is a terminal window titled "Parrot Terminal" with the command \$finger Admin@192.168.0.50. The output of the command is displayed in green text:

```
$finger Admin@192.168.0.50
Login: Admin
Directory: /home/Admin
Name: Admin
Shell: /bin/bash
On since Mon Jun 3 07:45 (PDT) on :0 (messages off)
On since Mon Jun 3 07:45 (PDT) on pts/0 from :
    4 minutes 56 seconds idle
No mail.
No Plan.
[pentester@parrot]~[~]
$telnet 192.168.0.50 79
Trying 192.168.0.50...
Connected to 192.168.0.50.
Escape character is '^'.
Admin
Login: Admin
Directory: /home/Admin
Name: Admin
Shell: /bin/bash
On since Mon Jun 3 07:45 (PDT) on :0 (messages off)
On since Mon Jun 3 07:45 (PDT) on pts/0 from :
    6 minutes 9 seconds idle
No mail.
No Plan.
Connection closed by foreign host.
[x]-[pentester@parrot]~[~]
$
```

The desktop background is a dark blue geometric pattern. A file manager window titled "Pluma" is visible in the background, showing a file named "kevinh (~)". The bottom of the screen shows the desktop menu and taskbar.

Exercise 12, Step 12: To safeguard your machine from returning the logged in user information, it is recommended to disable finger service on the machine by editing the finger text file located in the /etc/xinetd.d.

NOTE: Due to the fact that the remote host machine is not able to be logged in remotely by the current machine. The text file is unable to be accessed remotely as per the Lab.



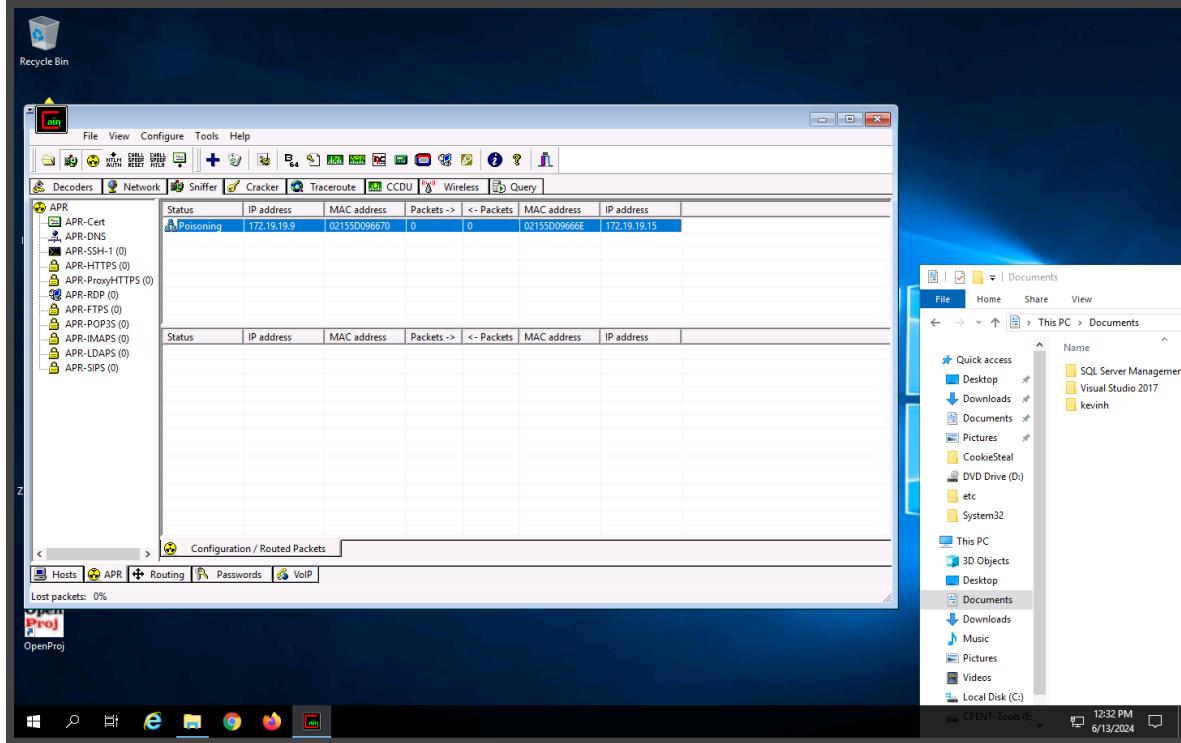
12.2 QUESTIONS

The screenshot shows a dual-pane interface for network penetration testing. On the left is a terminal window titled '*kevin (~)' showing a blank command-line interface. On the right is an exercise interface with the title 'Network Penetration Testing Methodology-Internal'. The interface includes tabs for 'Instructions' (selected) and 'Resources'. The 'Instructions' tab contains text: 'vulnerability and you are not required to log in to the machine to view the above-mentioned file.' Below this is a note: 'The finger text file is located in /etc/xinetd.d.'. A preview window shows a terminal session with the output of the 'finger' command. Below the preview is a summary: 'In this lab, you have learned how to enumerate user information using finger client.' A question section asks: 'Enumerate logged-on users on the target Red Hat Enterprise machine (192.168.0.50) using the finger client. Enter the name of the logged-on user on the target machine.' A text input field contains 'Admin', which is highlighted as a correct answer ('Correct'). A progress bar at the bottom indicates '43 Minutes Remaining'.

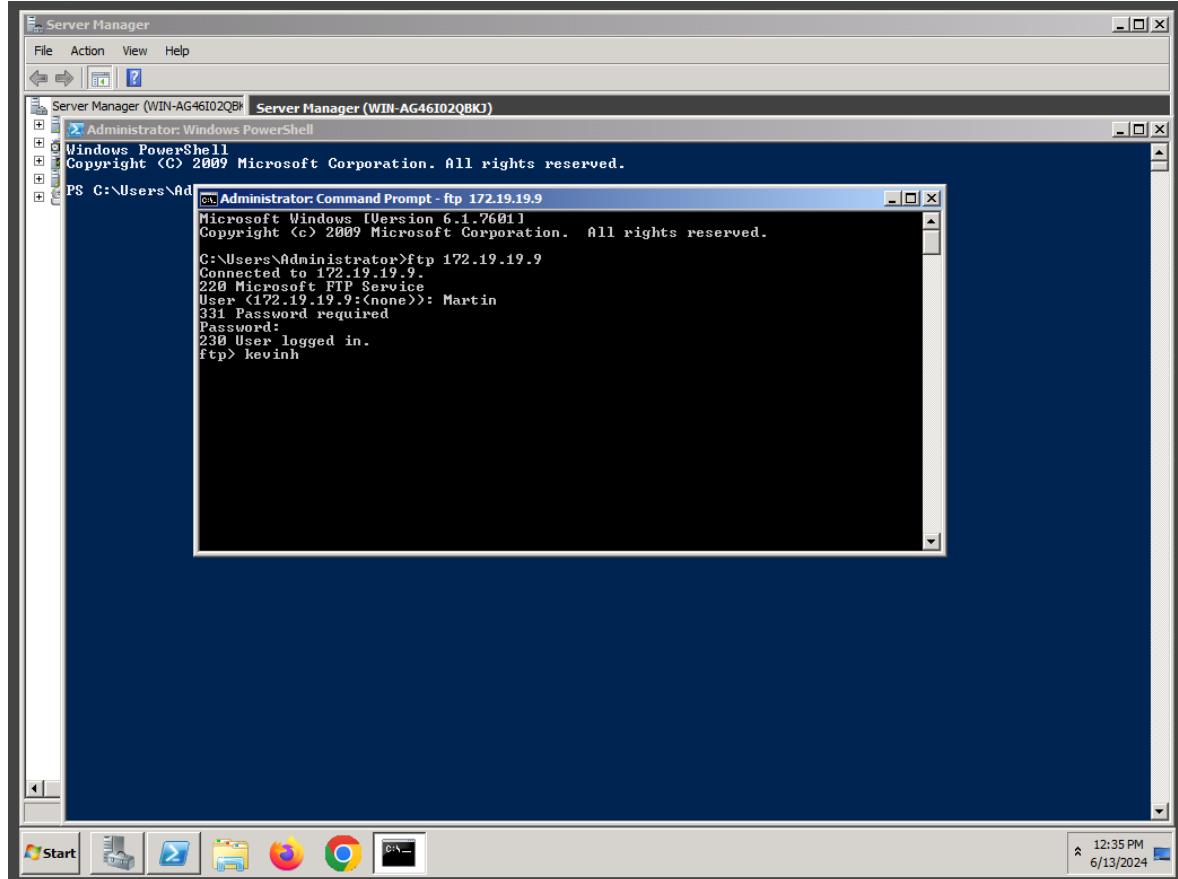
Exercise 13: Performing Man-in-the-Middle Attack using Cain & Abel

13.1 OUTPUT SCREENSHOTS

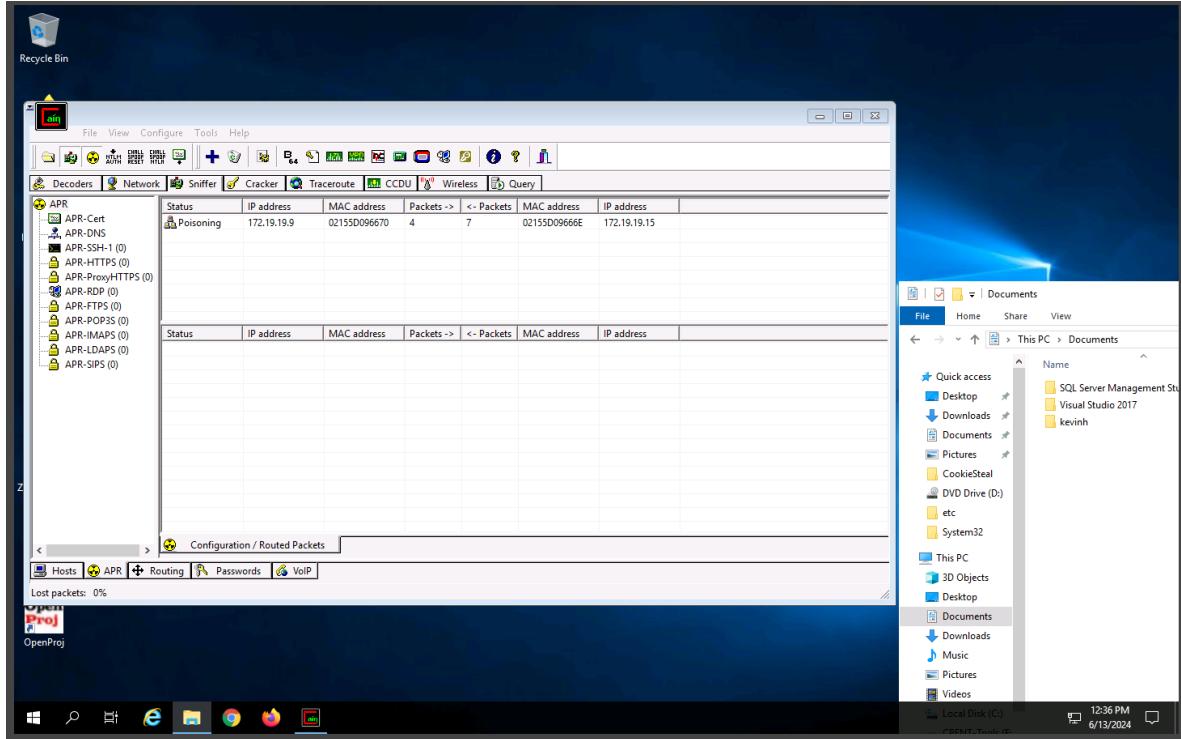
Exercise 13, Step 15: Select the added IP address in the Configuration/Routed packets, and click Start/Stop APR (third icon from left) icon. Cain begins ARP poisoning in between these machines.



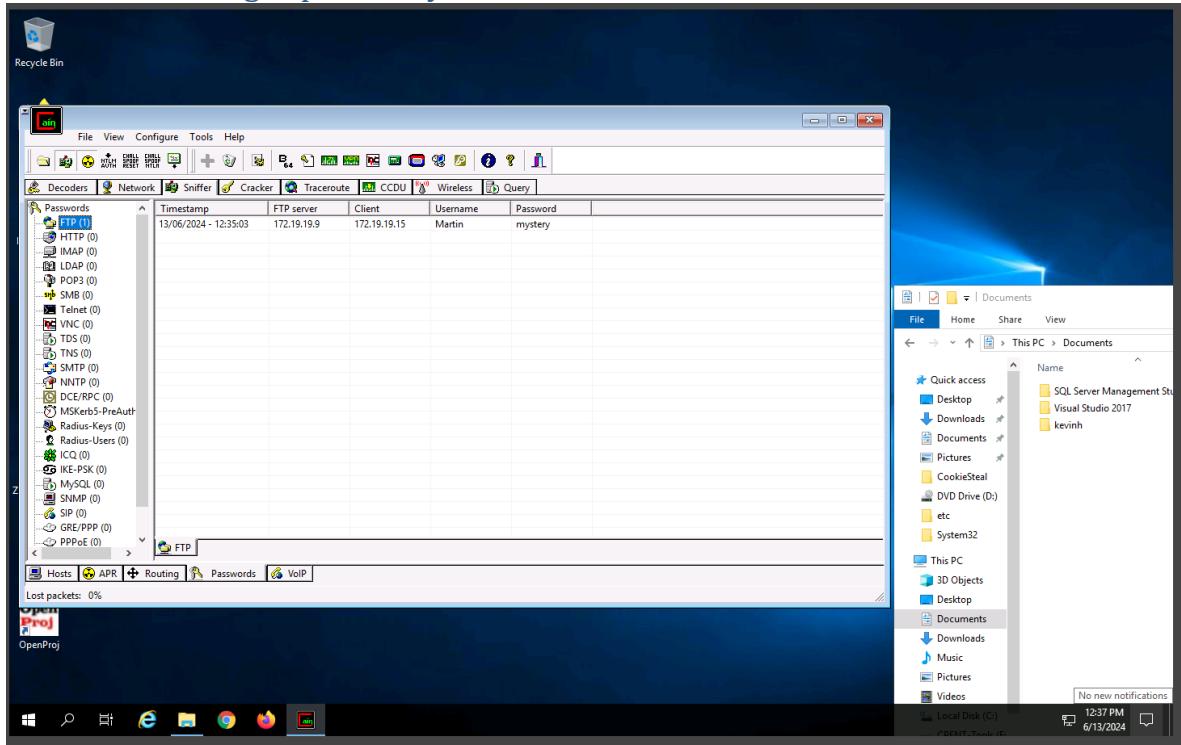
Exercise 13, Step 20: Now launch a command prompt in the machine, type ftp 172.19.19.9 (IP address of FTP Server machine) and press Enter. When prompted for the Username, type "Martin" and press Enter. When prompted for the password, type "mystery" and press Enter



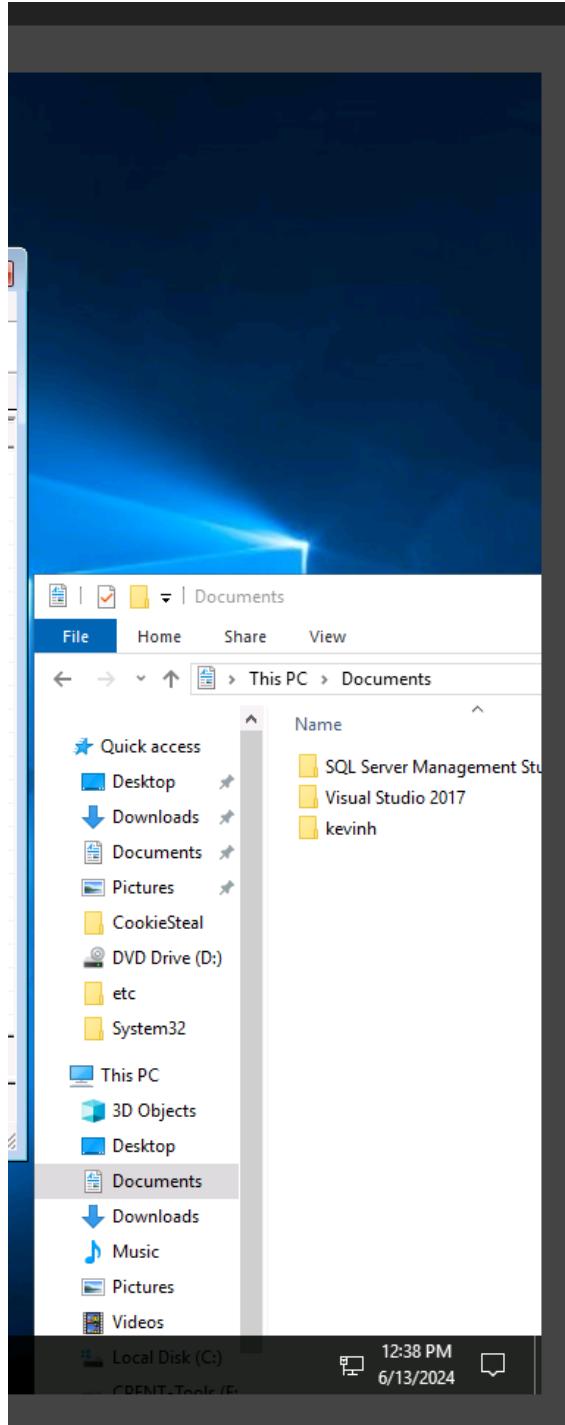
Exercise 13, Step 21: You will observe that Cain & Abel captured some packets which can be observed under the Packets field.



Exercise 13, Step 22: Click the Passwords tab in the Cain & Abel GUI. Select FTP from the left pane under the Passwords section. You will observe the credentials being captured by Cain & Abel as shown in the screenshot



13.2 QUESTIONS

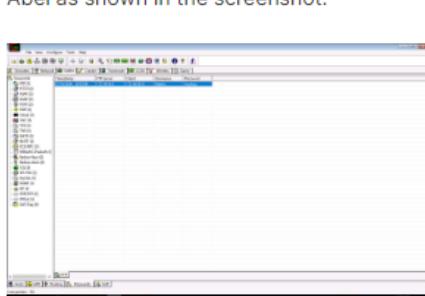


Network Penetration Testing
Methodology-Internal

Save &
Exit

Instructions Resources

22. Click the **Passwords** tab in the Cain & Abel GUI. Select **FTP** from the left pane under the **Passwords** section. You will observe the credentials being captured by Cain & Abel as shown in the screenshot.



23. This way, you have successfully captured user credentials traversing in clear-text. In this lab, you have learned how to capture user credentials in a switch based network.

Question 6.13.1

Run Cain and Abel on the Windows Server 2019 machine and perform an MITM attack to sniff traffic between the AdvertisementDept and Web Server machines. Which tab under Cain lists all the discoverable machines in the network?

Sniffer

Score

Correct

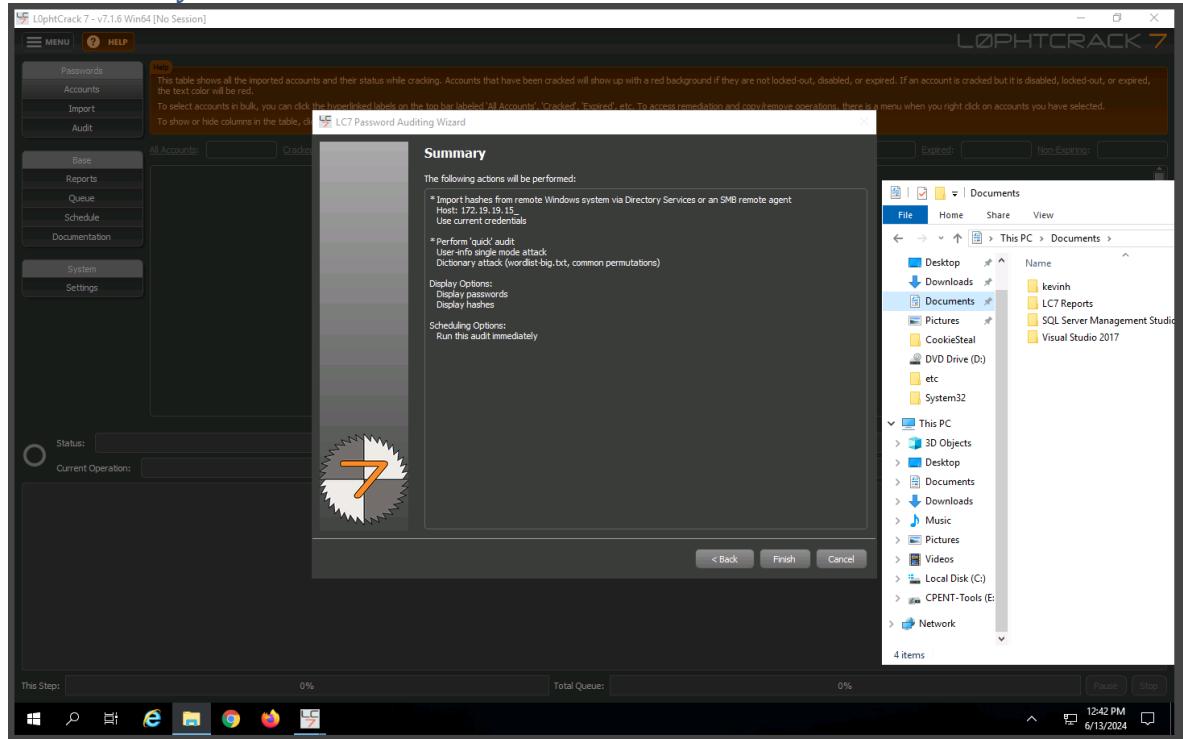
Previous Next

29 Minutes Remaining

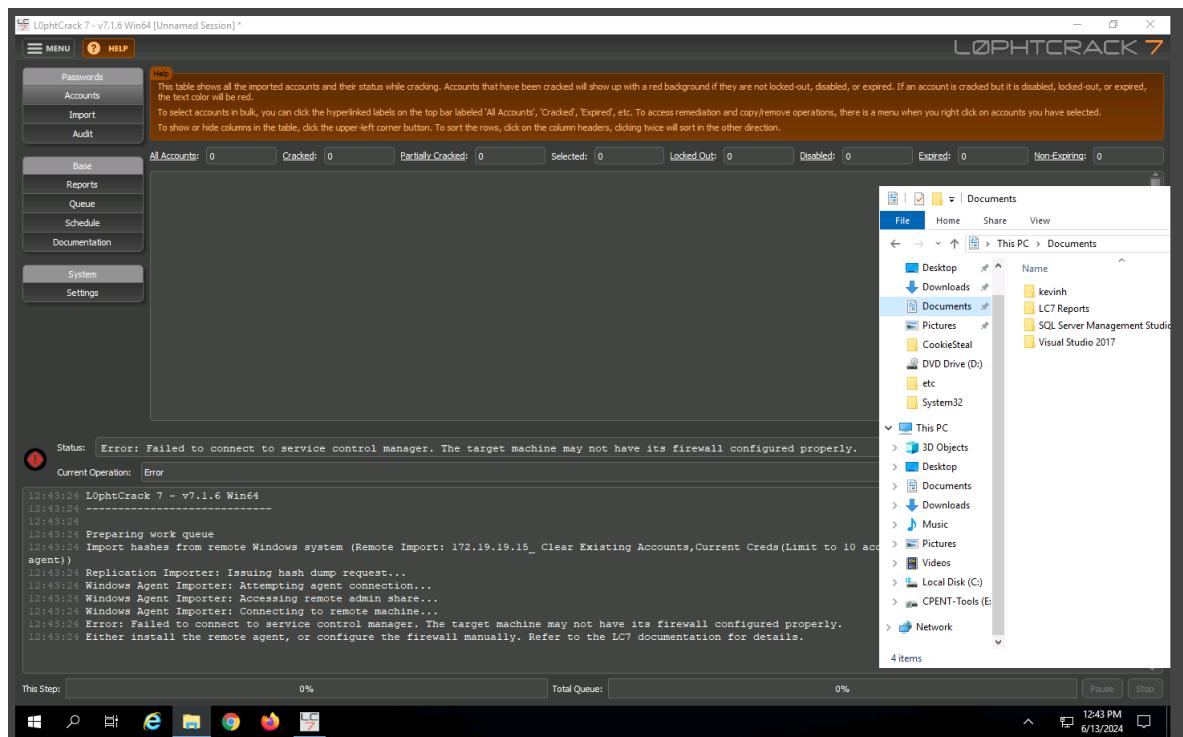
Exercise 14: Auditing a Machine for Weak Passwords Using LophCrack

14.1 OUTPUT SCREENSHOTS

Exercise 14, Step 13: In the Job Scheduling window, select Run this job immediately and click Next.

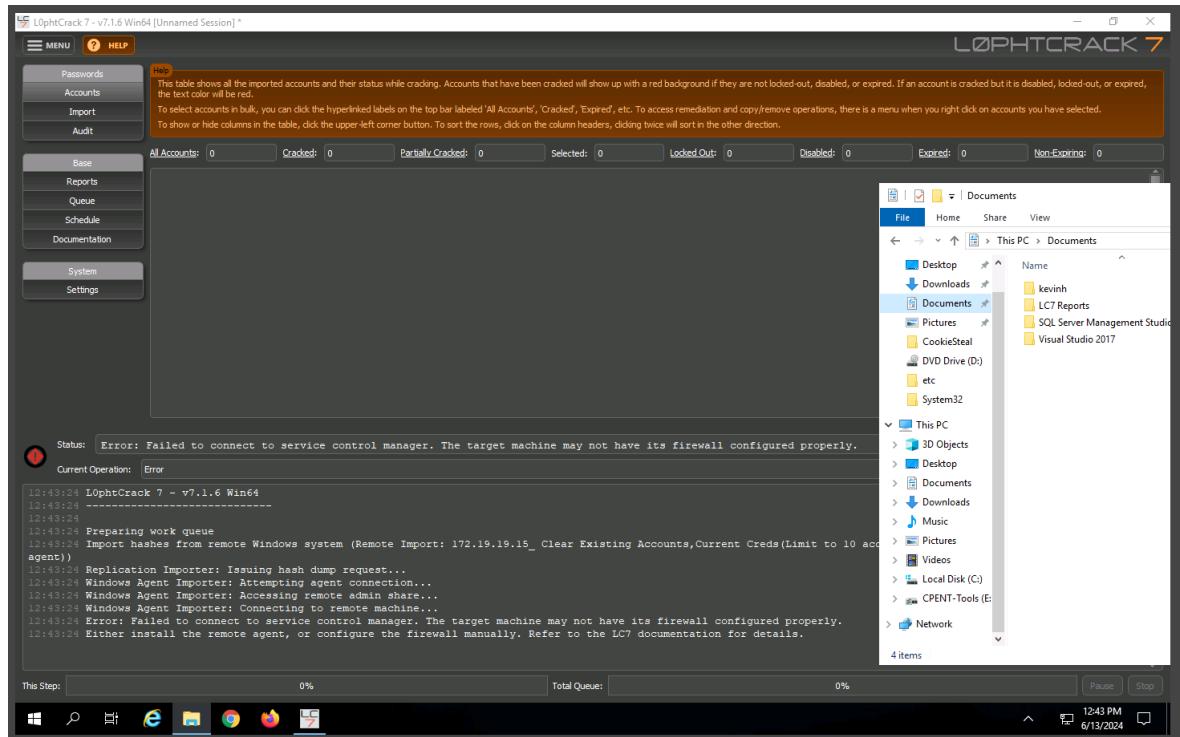


Exercise 14, Step 15: A caution box appears regarding changed LC7Agent on the remote machine as shown in the screenshot. Click Yes.

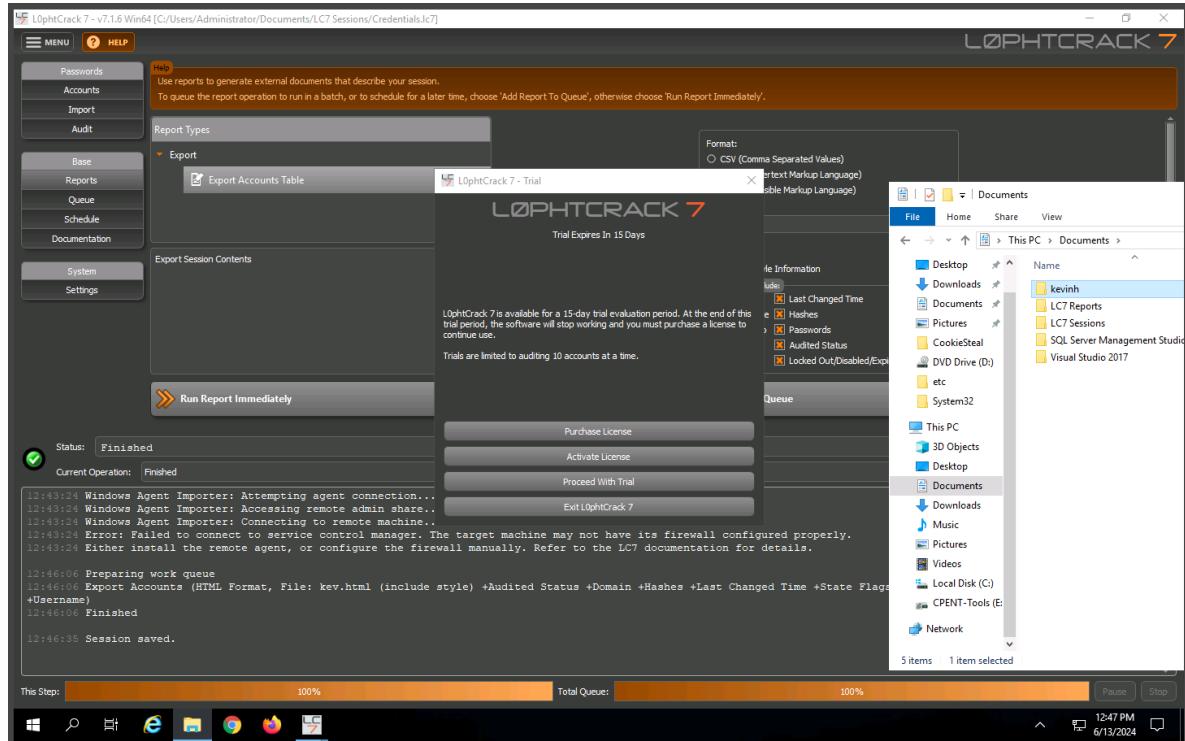


Exercise 14, Step 16: L0phtCrack will begin to decode the hashes. You can see the Progress bar in the lower right-hand corner of the window. Once done with the password auditing, it displays the weak passwords set for the respective user accounts present in Advertisement Dept machine as shown in the screenshot

NOTE: The tool is currently experiencing an error due to the establishment of a firewall incorporated between labs. Still able to run the tool, just not obtain the desired output of data.



Exercise 14, Step 20: To open the saved result, navigate to Desktop and double-click the Credentials.lcs file to view result.



14.2 QUESTIONS

Network Penetration Testing
Methodology-Internal

Save & Exit

Instructions Resources

22. Now you can see the saved result in the L0phtCrack window.

23. Close all the open windows.

In this lab you have learnt how to extract the **Administrators** password using L0phtCrack.

Question 6.14.1

Install the L0phtCrack tool available at E:\CPENT Module 06 Network Penetration Testing Methodology-Internal\L0phtCrack on the Windows Server 2019 machine. Perform the password auditing of the Advertising Department machine, which is at 172.19.19.15. Enter the password associated with user steve on this machine.

italy

Score

Correct

Previous Next

20 Minutes Remaining

Exercise 15: Automating Penetration Testing Tasks Using Bash Scripting

15.1 OUTPUT SCREENSHOTS

Exercise 15, Step 18: Now, minimize the text editor window and maximize the command line terminal. Nmap has performed live host identification on the given IP Address range. Once the live hosts are identified, the script is written in such a way, that a new nmap scan is initiated to find the machines (among the identified live hosts) that have the FTP port open. The live machines with the FTP port open are displayed as shown in the screenshot

The screenshot shows a Parrot OS desktop environment. In the foreground, a terminal window titled "Parrot Terminal" is maximized, displaying the output of an nmap scan. The output shows:

```
Applications Places System Parrot Terminal
File Edit View Search Terminal Help
PORT      STATE SERVICE
21/tcp    open  ftp
Nmap scan report for www.cpent.com (172.19.19.24)
Host is up (0.00046s latency).
PORT      STATE SERVICE
21/tcp closed ftp
Nmap done: 11 IP addresses (11 hosts up) scanned in 0.14 seconds
Nmap has performed a scan to identify the hosts which have FTP port open on them
.
172.19.19.7
172.19.19.9
172.19.19.10
172.19.19.17
172.19.19.19
172.19.19.22

Enter the IP address of the machine on which you want to perform FTP dictionary attack.
The script will perform dictionary attack on the selected host using hydra.
```

In the background, a file manager window titled "Pluma" is visible, showing a list of files. The desktop interface includes icons for Applications, Places, System, and a terminal icon. The taskbar at the bottom shows three terminal windows: "Parrot Terminal", "*kevinh (~) - Pluma", and "[pentest.sh (~) - Pluma]".

Exercise 14, Step 23: Minimize the text editor window and maximize the command line terminal. On issuing the IP Address, Hydra begins to perform Dictionary attack on the machine and starts displaying the user credentials as shown in the screenshot.

The screenshot shows a Linux desktop environment with a terminal window titled "Parrot Terminal". The terminal displays the output of the Hydra dictionary attack script. The text in the terminal is as follows:

```
Nmap has performed a scan to identify the hosts which have FTP port open on them
172.19.19.7
172.19.19.9
172.19.19.10
172.19.19.17
172.19.19.19
172.19.19.22

Enter the IP address of the machine on which you want to perform FTP dictionary attack.
The script will perform dictionary attack on the selected host using hydra.
172.19.19.9
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-06-13 15:55:
54
[DATA] max 16 tasks per 1 server, overall 16 tasks, 12669 login tries (l:41/p:309), ~792 tries per task
[DATA] attacking ftp://172.19.19.9:21/
```

The terminal window is the active application in the taskbar, which also lists other open windows like "Pluma" and "pentest.sh". The desktop environment includes a top bar with system icons and a date/time indicator.

Exercise 14, Step 30: On issuing the user credentials, you will be logged in to the FTP Server, as shown in the screenshot

The screenshot shows a Parrot OS desktop environment. In the foreground, a terminal window titled "Parrot Terminal" is open, displaying the following text:

```
Enter the IP address of the machine on which you want to perform FTP dictionary attack.  
The script will perform dictionary attack on the selected host using hydra.  
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).  
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-06-13 15:52:04  
Error: no target server given, nor -M option used  
  
Now that you have obtained the credentials, let us login to the FTP Server.  
Enter the IP address of the FTP server:  
172.19.19.9  
Connected to 172.19.19.9.  
220 Microsoft FTP Service  
Name (172.19.19.9:pentester): jason  
331 Password required  
Password:  
230 User logged in.  
Remote system type is Windows_NT.  
ftp> [REDACTED]
```

In the background, a file manager window titled "Pluma" is visible, showing a folder structure under the path "/pentest.sh". The status bar of the file manager indicates the current user is "kevinh (~)".

15.2 QUESTIONS

NOTE: due to the lab locking up and was unable to re-specify the FTP target as well as to interact with the lab, I was unable to find the answer for the final question. I was still able to learn how to use FTP still however.

Network Penetration Testing
Methodology-Internal

X Save & Exit

Instructions Resources ?

In this lab, you have successfully performed subnet scan, found machines having FTP ports open, performed dictionary attack to attain credentials, and successfully logged in to the server using the obtained credentials.

Question 6.15.1

"Using bash scripting, perform live host and FTP port identification on the target IP address range 172.19.19.7. Enter the number of hosts that have the FTP port open.
Note: The Bash script file (pentest.sh) is available in the Home Folder directory of the Parrot machine."

Score

Correct

Question 6.15.2

Using bash scripting, perform a dictionary attack on the target IP address, 172.19.19.17, to obtain FTP credentials. Enter the password associated with user ruby on the target machine.

Score

You have reached the max attempts for this question

[← Previous](#) [Next →](#)

21 Minutes Remaining

