PSP0201

Week 3 Writeup

Group name: SOLO

ID	NAME	ROLE
1211100574	Ivan Liew Qi Hong	leader

Day 6: [Web Exploitation] Be Careful With What You Wish On Christmas Night

Tools used: Kali Linux, Firefox, OWASP Zap

Solution/Walkthrough:

Question 1

Check OWASP cheat sheet for answer

Question 2

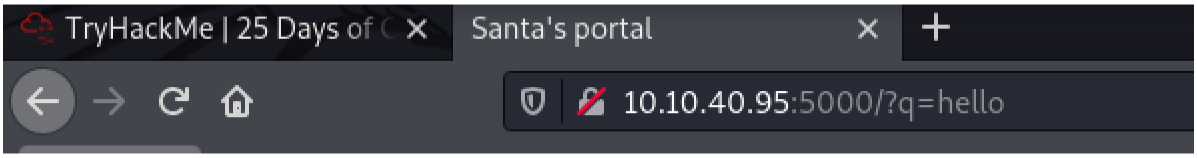
Copy expression from OWASP cheat sheet

Question 3

The answer can be found by reading the text

Question 4

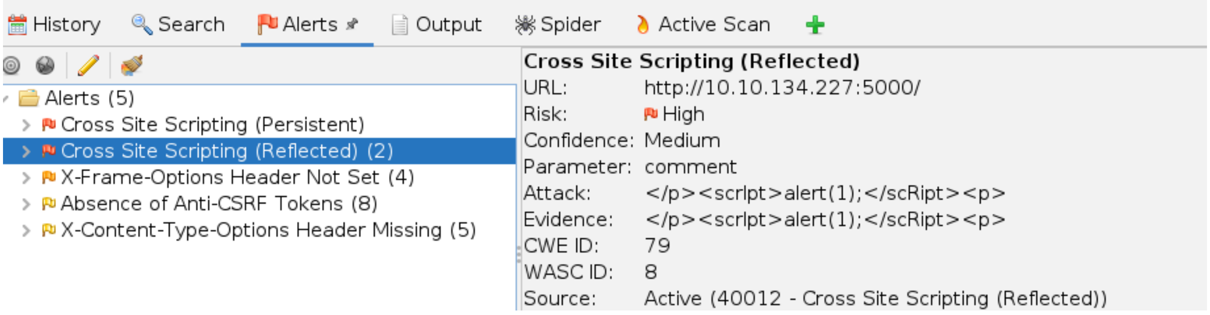
Typing anything in the query will add the query string q which can be abused



10.10.40.95:5000/?q=hello

Question 5

The XSS alerts of hight priority are found in the alerts tab after running the attack which are indicated by the red flag



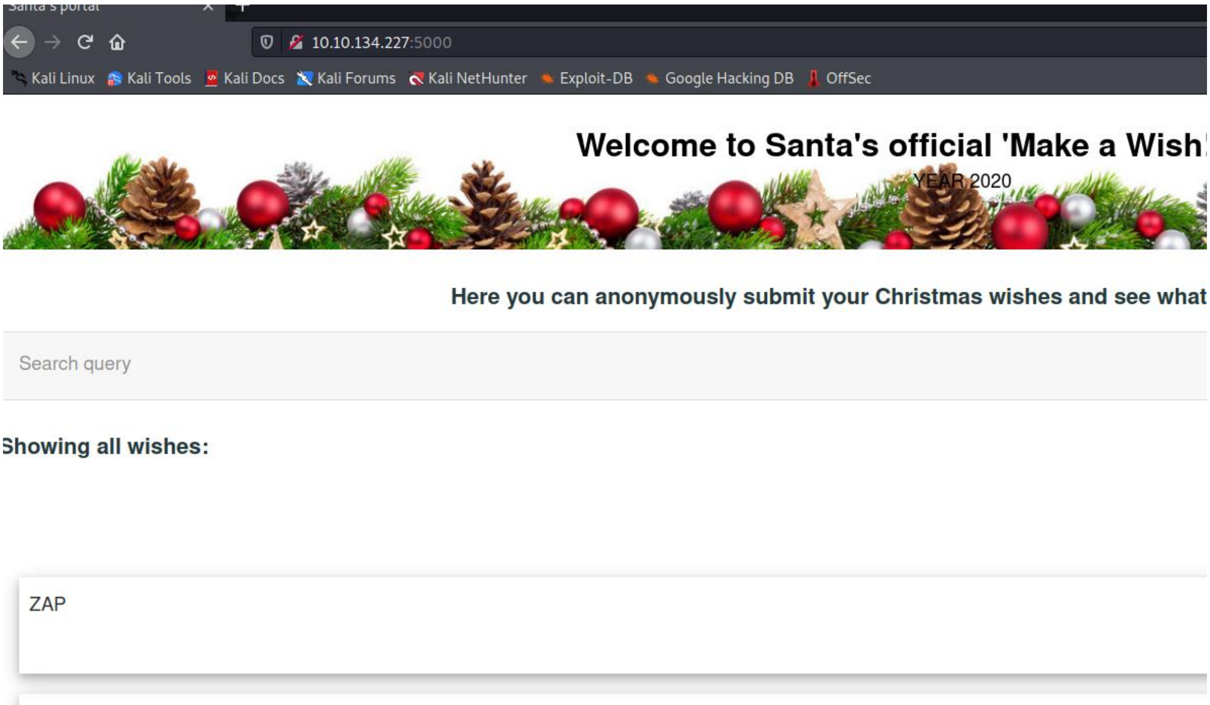
Alerts (5)
- Cross Site Scripting (Persistent)
- Cross Site Scripting (Reflected) (2)
- X-Frame-Options Header Not Set (4)
- Absence of Anti-CSRF Tokens (8)
- X-Content-Type-Options Header Missing (5)

```
Cross Site Scripting (Reflected)
URL:        http://10.10.134.227:5000/
Risk:       High
Confidence: Medium
Parameter:  comment
Attack:     </p><scrIpt>alert(1);</scRipt><p>
Evidence:   </p><scrIpt>alert(1);</scRipt><p>
CWE ID:     79
WASC ID:    8
Source:     Active (40012 - Cross Site Scripting (Reflected))
```

Question 6

Question 7

Closing and reopening the browser, the attack still persists



Santa's portal

10.10.134.227:5000

Welcome to Santa's official 'Make a Wish

YEAR 2020

Here you can anonymously submit your Christmas wishes and see what

Search query

Showing all wishes:

ZAP

Thought Process/Methodology:

I placed the URL of the website into OWASP Zap and proceeded to attack it. It then showed the alerts that it had acquired.

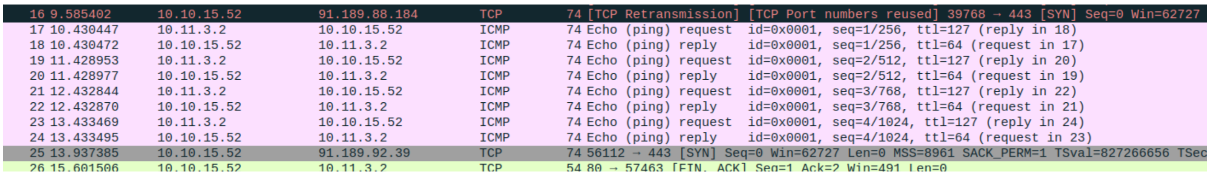
Day 7: [Networking] The Grinch Really Did Steal Christmas

Tools used: Kali Linux, Firefox, Wireshark

Solution/Walkthrough:

Question 1

The Ip address can be found by looking at wireshark after the file has been open



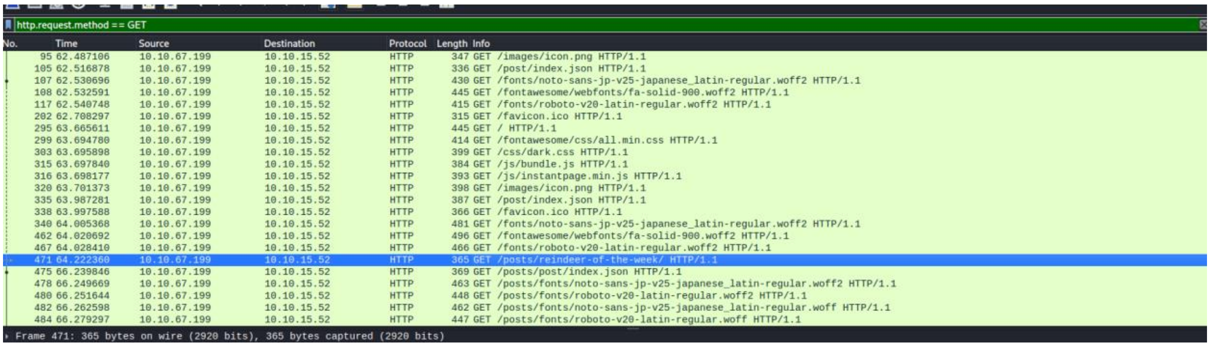
```
16 9.585402   10.10.15.52  91.189.88.184  TCP   74 [TCP Retransmission] [TCP Port numbers reused] 39768 → 443 [SYN] Seq=0 Win=62727
17 10.430447  10.11.3.2    10.10.15.52    ICMP  74 Echo (ping) request  id=0x0001, seq=1/256, ttl=127 (reply in 18)
18 10.430472  10.10.15.52  10.11.3.2      ICMP  74 Echo (ping) reply    id=0x0001, seq=1/256, ttl=64 (request in 17)
19 11.428953  10.11.3.2    10.10.15.52    ICMP  74 Echo (ping) request  id=0x0001, seq=2/512, ttl=127 (reply in 20)
20 11.428977  10.10.15.52  10.11.3.2      ICMP  74 Echo (ping) reply    id=0x0001, seq=2/512, ttl=64 (request in 19)
21 12.432844  10.11.3.2    10.10.15.52    ICMP  74 Echo (ping) request  id=0x0001, seq=3/768, ttl=127 (reply in 22)
22 12.432870  10.10.15.52  10.11.3.2      ICMP  74 Echo (ping) reply    id=0x0001, seq=3/768, ttl=64 (request in 21)
23 13.433469  10.11.3.2    10.10.15.52    ICMP  74 Echo (ping) request  id=0x0001, seq=4/1024, ttl=127 (reply in 24)
24 13.433495  10.10.15.52  10.11.3.2      ICMP  74 Echo (ping) reply    id=0x0001, seq=4/1024, ttl=64 (request in 23)
25 13.937385  10.10.15.52  91.189.92.39   TCP   74 56112 → 443 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 TSval=827266656 TSec
26 15.601506  10.10.15.52  10.11.3.2      TCP   54 80 → 57463 [FIN, ACK] Seq=1 Ack=2 Win=491 Len=0
```

Question 2

The filter is http.request.method == GET

Question 3

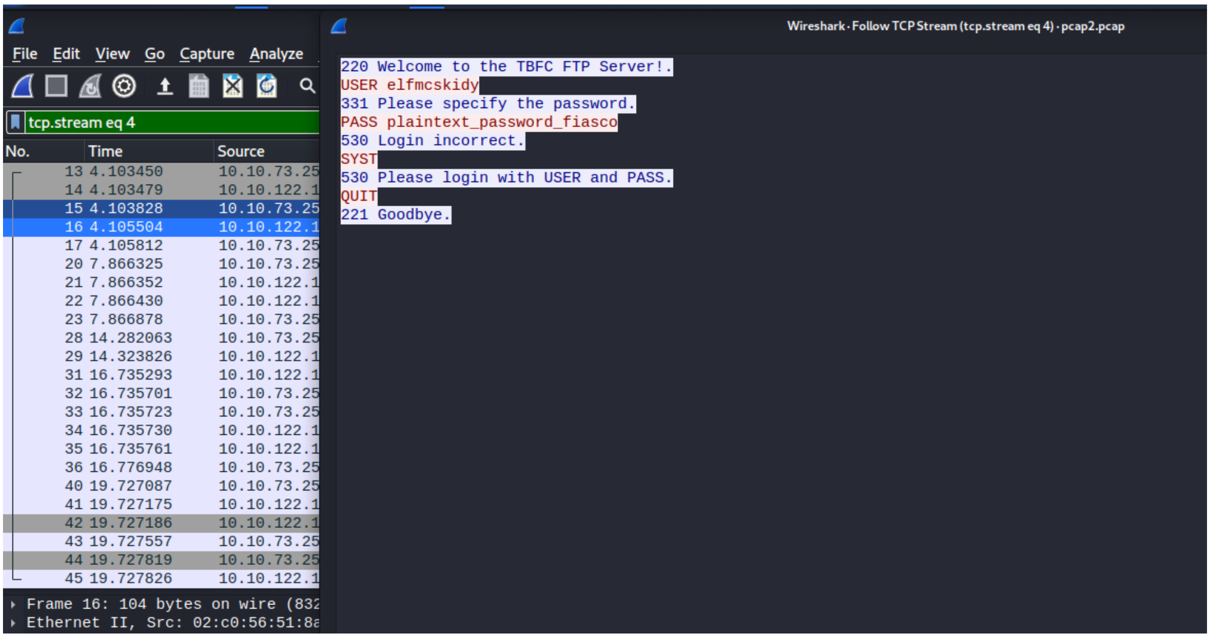
The name of the article can be found after applying the filter



```
http.request.method == GET
No.   Time          Source         Destination    Protocol  Length Info
   95 62.487106     10.10.67.199   10.10.15.52    HTTP      347 GET /images/icon.png HTTP/1.1
  105 62.516878     10.10.67.199   10.10.15.52    HTTP      336 GET /post/index.json HTTP/1.1
  107 62.530696     10.10.67.199   10.10.15.52    HTTP      430 GET /fonts/noto-sans-jp-v25-japanese_latin-regular.woff2 HTTP/1.1
  108 62.532591     10.10.67.199   10.10.15.52    HTTP      445 GET /fontawesome/webfonts/fa-solid-900.woff2 HTTP/1.1
  117 62.540748     10.10.67.199   10.10.15.52    HTTP      415 GET /fonts/roboto-v20-latin-regular.woff2 HTTP/1.1
  202 62.708297     10.10.67.199   10.10.15.52    HTTP      315 GET /favicon.ico HTTP/1.1
  295 63.665611     10.10.67.199   10.10.15.52    HTTP      445 GET / HTTP/1.1
  299 63.694780     10.10.67.199   10.10.15.52    HTTP      414 GET /fontawesome/css/all.min.css HTTP/1.1
  303 63.695898     10.10.67.199   10.10.15.52    HTTP      399 GET /css/dark.css HTTP/1.1
  315 63.697840     10.10.67.199   10.10.15.52    HTTP      384 GET /js/bundle.js HTTP/1.1
  316 63.698177     10.10.67.199   10.10.15.52    HTTP      393 GET /js/instantpage.min.js HTTP/1.1
  320 63.701373     10.10.67.199   10.10.15.52    HTTP      398 GET /images/icon.png HTTP/1.1
  335 63.987281     10.10.67.199   10.10.15.52    HTTP      387 GET /post/index.json HTTP/1.1
  338 63.997588     10.10.67.199   10.10.15.52    HTTP      366 GET /favicon.ico HTTP/1.1
  340 64.005368     10.10.67.199   10.10.15.52    HTTP      481 GET /fonts/noto-sans-jp-v25-japanese_latin-regular.woff2 HTTP/1.1
  462 64.020692     10.10.67.199   10.10.15.52    HTTP      496 GET /fontawesome/webfonts/fa-solid-900.woff2 HTTP/1.1
  467 64.028410     10.10.67.199   10.10.15.52    HTTP      466 GET /fonts/roboto-v20-latin-regular.woff2 HTTP/1.1
  471 64.222360     10.10.67.199   10.10.15.52    HTTP      365 GET /posts/reindeer-of-the-week/ HTTP/1.1
  475 66.239846     10.10.67.199   10.10.15.52    HTTP      369 GET /posts/post/index.json HTTP/1.1
  478 66.249669     10.10.67.199   10.10.15.52    HTTP      463 GET /posts/fonts/noto-sans-jp-v25-japanese_latin-regular.woff2 HTTP/1.1
  480 66.251644     10.10.67.199   10.10.15.52    HTTP      448 GET /posts/fonts/roboto-v20-latin-regular.woff2 HTTP/1.1
  482 66.262598     10.10.67.199   10.10.15.52    HTTP      462 GET /posts/fonts/noto-sans-jp-v25-japanese_latin-regular.woff HTTP/1.1
  484 66.279297     10.10.67.199   10.10.15.52    HTTP      447 GET /posts/fonts/roboto-v20-latin-regular.woff HTTP/1.1
▸ Frame 471: 365 bytes on wire (2920 bits), 365 bytes captured (2920 bits)
```

Question 4

Follow the successful login to find leaked password



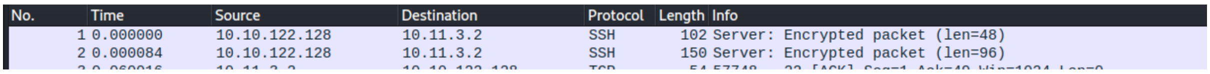
File
Edit
View
Go
Capture
Analyze
Wireshark · Follow TCP Stream (tcp.stream eq 4) · pcap2.pcap

```
tcp.stream eq 4
No. Time       Source
 13 4.103450   10.10.73.25
 14 4.103479   10.10.122.1
 15 4.103828   10.10.73.25
 16 4.105504   10.10.122.1
 17 4.105812   10.10.73.25
 20 7.866325   10.10.73.25
 21 7.866352   10.10.122.1
 22 7.866430   10.10.122.1
 23 7.866878   10.10.73.25
 28 14.282063  10.10.73.25
 29 14.323826  10.10.122.1
 31 16.735293  10.10.122.1
 32 16.735701  10.10.73.25
 33 16.735723  10.10.73.25
 34 16.735730  10.10.122.1
 35 16.735761  10.10.122.1
 36 16.776948  10.10.73.25
 40 19.727087  10.10.73.25
 41 19.727175  10.10.122.1
 42 19.727186  10.10.122.1
 43 19.727557  10.10.73.25
 44 19.727819  10.10.73.25
 45 19.727826  10.10.122.1
▸ Frame 16: 104 bytes on wire (832
▸ Ethernet II, Src: 02:c0:56:51:8a
```

```
220 Welcome to the TBFC FTP Server!.
USER elfmcskidy
331 Please specify the password.
PASS plaintext_password_fiasco
530 Login incorrect.
SYST
530 Please login with USER and PASS.
QUIT
221 Goodbye.
```

Question 5

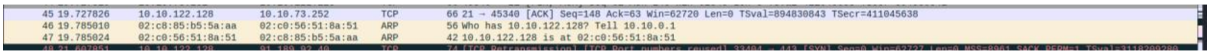
The encrypted protocol can be found at the top



```
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)
```

Question 6

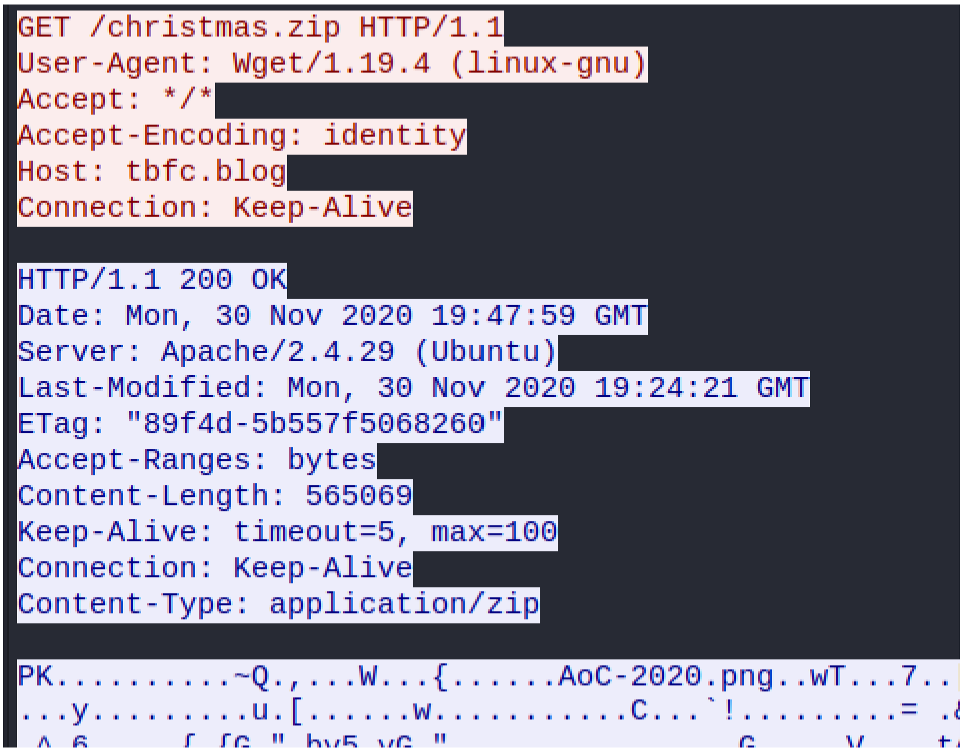
Find the arp communication under protocols



```
45 19.727826   10.10.122.128       10.10.73.252        TCP    66 21 → 45340 [ACK] Seq=148 Ack=63 Win=62720 Len=0 TSval=894830843 TSecr=411045638
46 19.785010   02:c8:85:b5:5a:aa   02:c0:56:51:8a:51   ARP    56 Who has 10.10.122.128? Tell 10.10.0.1
47 19.785024   02:c0:56:51:8a:51   02:c8:85:b5:5a:aa   ARP    42 10.10.122.128 is at 02:c0:56:51:8a:51
48 21.607851   10.10.122.128       91.189.92.40        TCP    74 [TCP Retransmission] [TCP Port numbers reused] 33404 → 443 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 TSval=3118209280
```

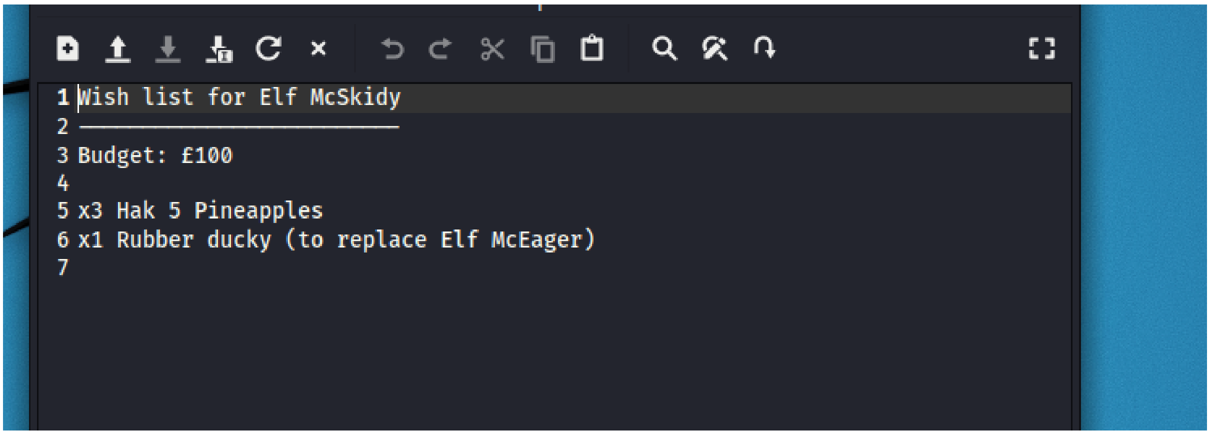
Question 7

Go to http protocol and follow it. It then shows that there is a zip file which can be exported



```
GET /christmas.zip HTTP/1.1
User-Agent: Wget/1.19.4 (linux-gnu)
Accept: */*
Accept-Encoding: identity
Host: tbfc.blog
Connection: Keep-Alive

HTTP/1.1 200 OK
Date: Mon, 30 Nov 2020 19:47:59 GMT
Server: Apache/2.4.29 (Ubuntu)
Last-Modified: Mon, 30 Nov 2020 19:24:21 GMT
ETag: "89f4d-5b557f5068260"
Accept-Ranges: bytes
Content-Length: 565069
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: application/zip

PK..........~Q.,...W...{......AoC-2020.png..wT...7..
...y.........u.[......w...........C...`!.........= ..
```

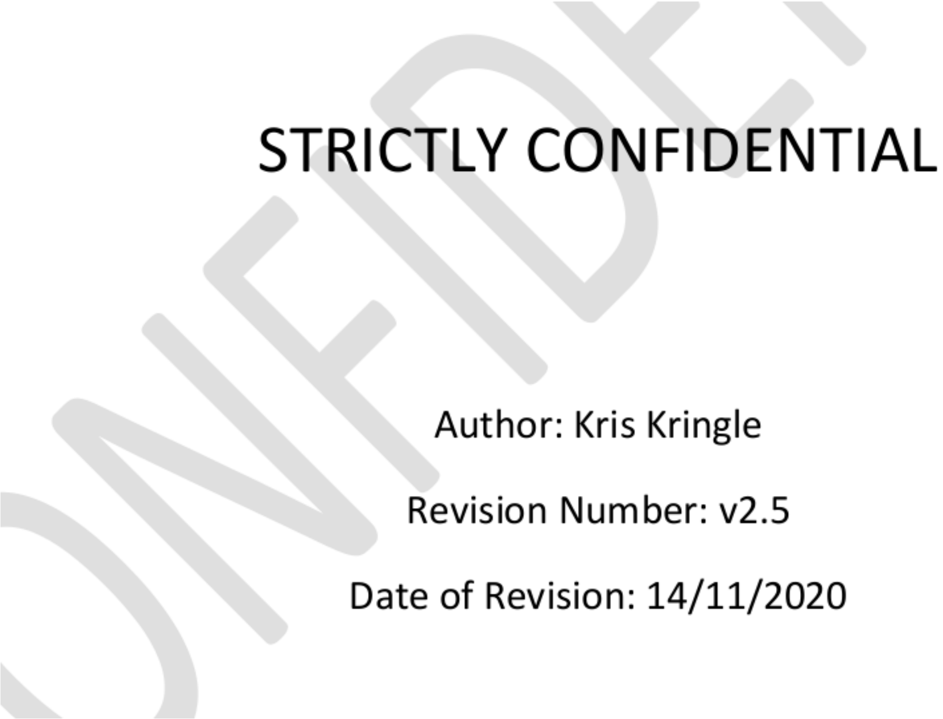
Extract file and open Elf McSkidy's wish list



```
Wish list for Elf McSkidy
-------------------------
Budget: £100

x3 Hak 5 Pineapples
x1 Rubber ducky (to replace Elf McEager)
```

Question 8

Open operation arctic storm pdf and find author



STRICTLY CONFIDENTIAL

Author: Kris Kringle

Revision Number: v2.5

Date of Revision: 14/11/2020

Thought Process/Methodology:

I opened Wireshark and open each of the files. In pcap1.pcap, I filtered to get the name of the article. In pcap2.pcap, I used a tcp port filter to get the password by following one of the ones that have a successful login. I then found the arp protocol which held a conversation. In pcap3.pcap I found a Http with a zip file which I exported and downloaded with the wish list as well as the operation storm pdf.

Day 8: [Networking] What's Under The Christmas Tree

Tools used: Kali Linux, Firefox,

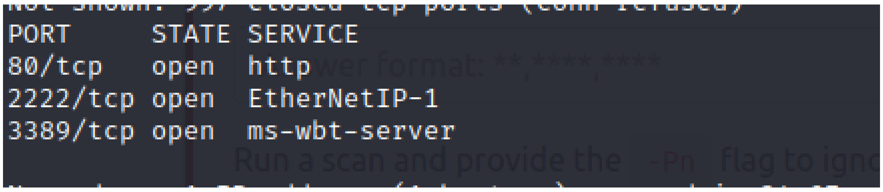
Solution/Walkthrough:

Question 1

Check the internet

Question 2

Put nmap and ip addres in terminal to get ports



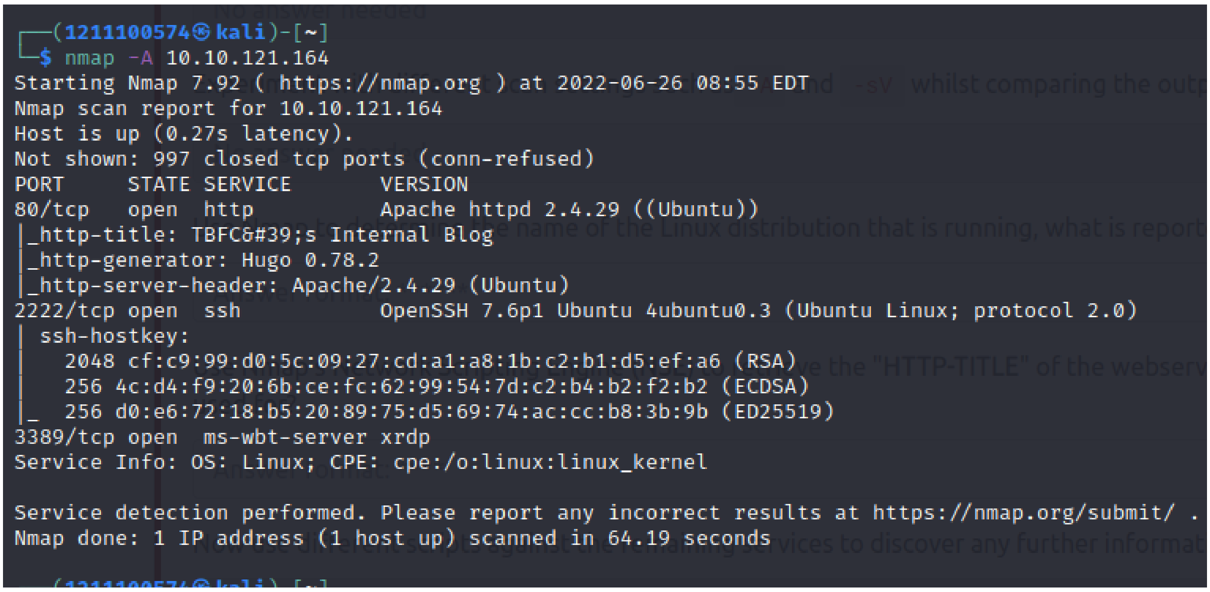
```
Not shown: 997 closed tcp ports (conn-refused)
PORT     STATE SERVICE
80/tcp   open  http
2222/tcp open  EtherNetIP-1
3389/tcp open  ms-wbt-server
```

wer format: **,****,****

Run a scan and provide the -Pn flag to ignc

Question 3

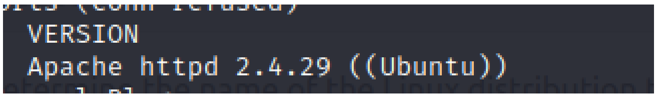
Put nmap -A Ip address to get answer



```
┌──(1211100574㉿kali)-[~]
└─$ nmap -A 10.10.121.164
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-26 08:55 EDT
Nmap scan report for 10.10.121.164
Host is up (0.27s 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: TBFC&#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: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 64.19 seconds
```

Question 4

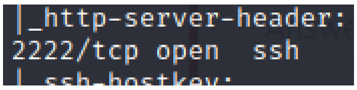
Apache version can also be found in the same command



```
VERSION
Apache httpd 2.4.29 ((Ubuntu))
```

Question 5

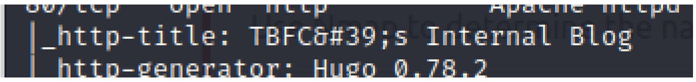
Port 2222 can also be found in the same command



```
|_http-server-header:
2222/tcp open  ssh
```

Question 6

Title is also in the command



```
|_http-title: TBFC&#39;s Internal Blog
```

Thought Process/Methodology:

I entered the nmap command along with the Ip address to get the ports. I then used the nmap command with -A to get information such as distribution, Apache version, what a port is running and the title.

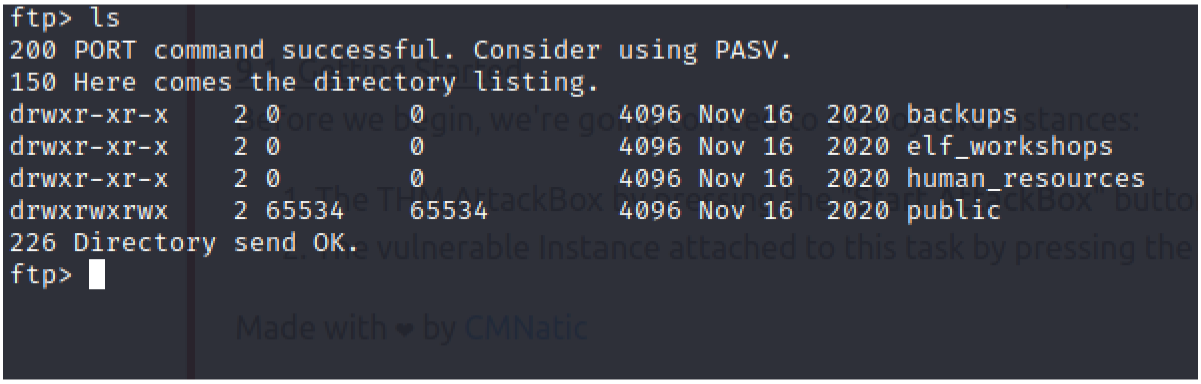
Day 9: [Networking] Anyone Can Be Santa!

Tools used: Kali Linux, Firefox,

Solution/Walkthrough:

Question 1

Using command ls, all available directories are shown



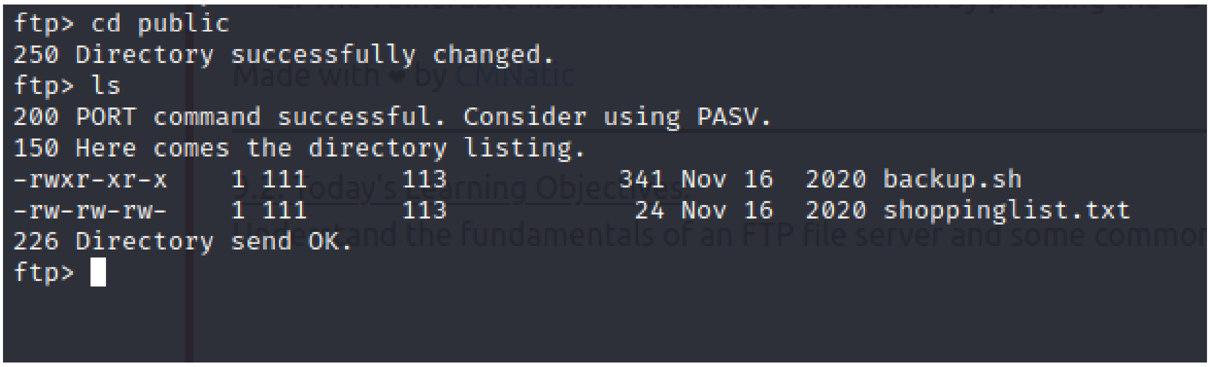
```
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x    2 0        0            4096 Nov 16  2020 backups
drwxr-xr-x    2 0        0            4096 Nov 16  2020 elf_workshops
drwxr-xr-x    2 0        0            4096 Nov 16  2020 human_resources
drwxrwxrwx    2 65534    65534        4096 Nov 16  2020 public
226 Directory send OK.
ftp> 
```

Question 2

Only 1 is shown to has data so it is assumed that the others are currently inaccessible

Question 3

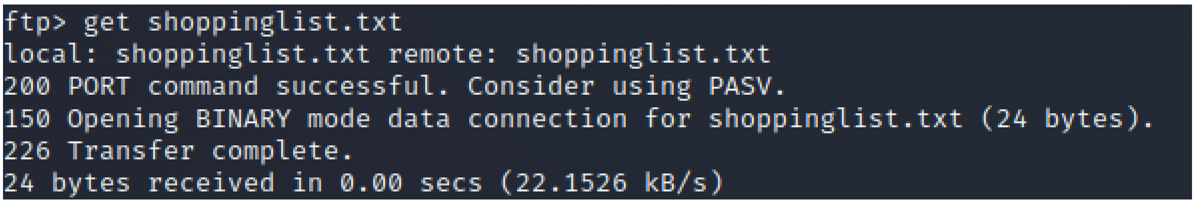
The script can be found after navigating to the public directory and using the ls command again



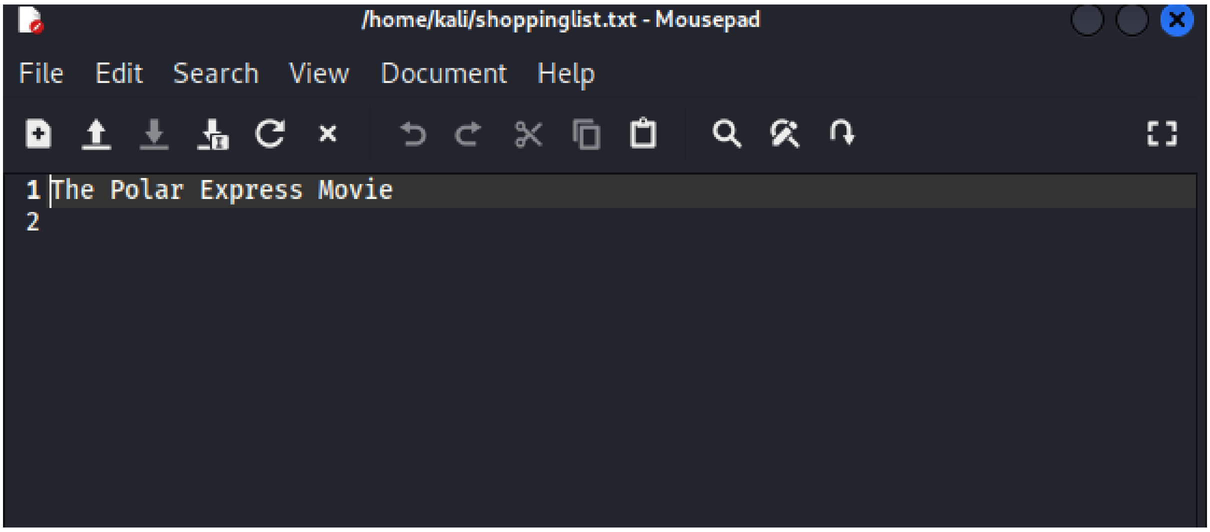
```
ftp> cd public
250 Directory successfully changed.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rwxr-xr-x    1 111      113           341 Nov 16  2020 backup.sh
-rw-rw-rw-    1 111      113            24 Nov 16  2020 shoppinglist.txt
226 Directory send OK.
ftp> 
```

Question 4

Use get to retrieve the text file to find the movie



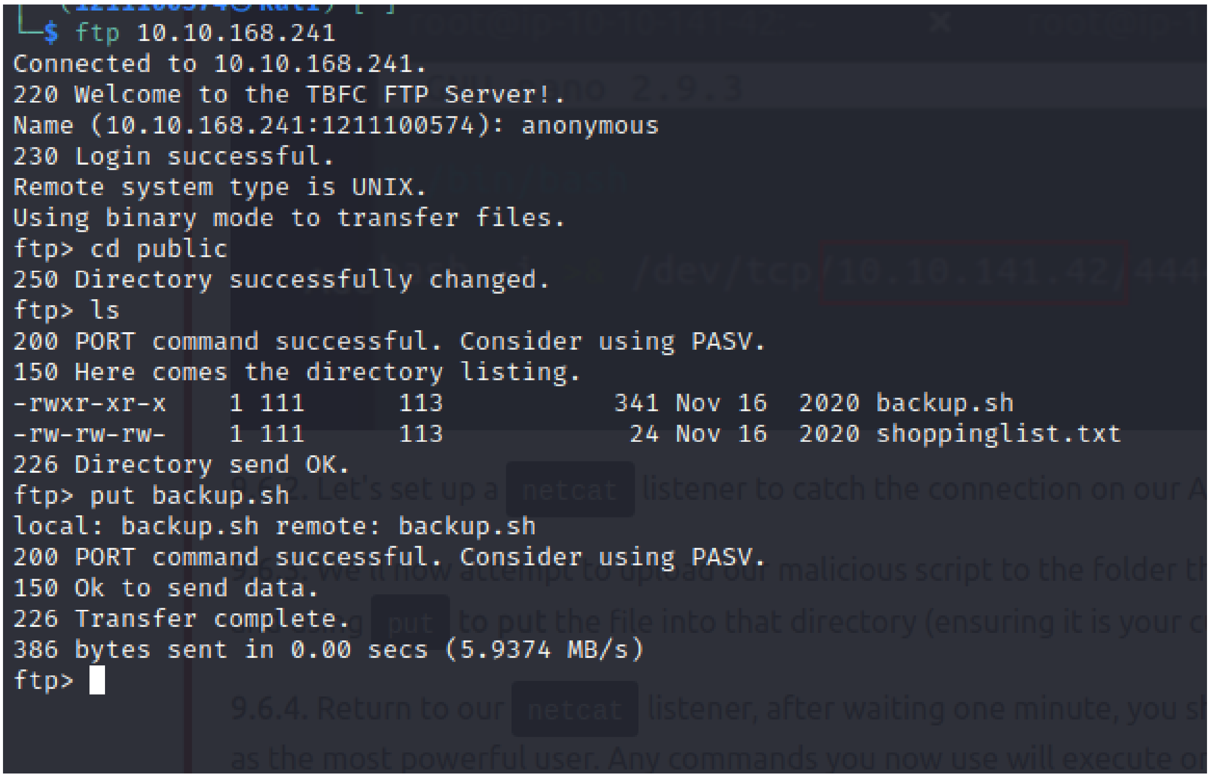
```
ftp> get shoppinglist.txt
local: shoppinglist.txt remote: shoppinglist.txt
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for shoppinglist.txt (24 bytes).
226 Transfer complete.
24 bytes received in 0.00 secs (22.1526 kB/s)
```



```
1 The Polar Express Movie
2
```

Question 5

Open the shell script with a text editor. Return back to ftp and put shell back into public directory.



```
└─$ ftp 10.10.168.241
Connected to 10.10.168.241.
220 Welcome to the TBFC FTP Server!.
Name (10.10.168.241:1211100574): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd public
250 Directory successfully changed.
ftp> ls
200 PORT command successful. Consider using PASV.
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> put backup.sh
local: backup.sh remote: backup.sh
200 PORT command successful. Consider using PASV.
150 Ok to send data.
226 Transfer complete.
386 bytes sent in 0.00 secs (5.9374 MB/s)
ftp> 
```

Get root flag from rootflag.txt

Thought Process/Methodology:

Using FTP commands, I logged in and found files that I have access and no access to. I got the shell script and text file which I then used. I then modified the script with malicious code and put it back. I then netcat the port and got the flag from the text file.

Day 10: [Networking] Don’t Be sELFish!

Tools used: Kali Linux, Firefox,

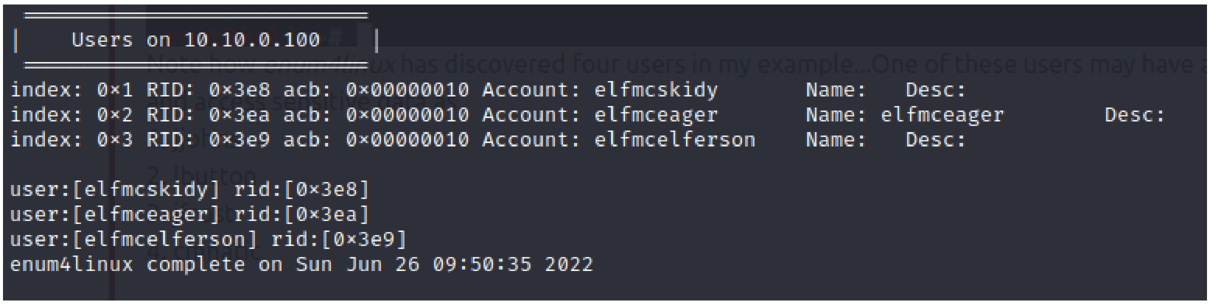
Solution/Walkthrough:

Question 1

Use help options to find correct descriptions

Question 2

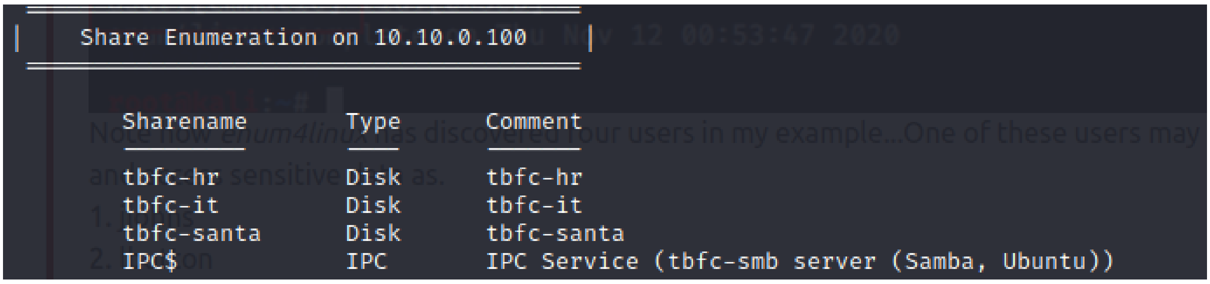
Use -U command with Ip address to find number of users on Samba server



```
 ==========================
|    Users on 10.10.0.100   |
 ==========================
index: 0×1 RID: 0×3e8 acb: 0×00000010 Account: elfmcskidy       Name:   Desc:
index: 0×2 RID: 0×3ea acb: 0×00000010 Account: elfmceager       Name: elfmceager       Desc:
index: 0×3 RID: 0×3e9 acb: 0×00000010 Account: elfmcelferson    Name:   Desc:

user:[elfmcskidy] rid:[0×3e8]
user:[elfmceager] rid:[0×3ea]
user:[elfmcelferson] rid:[0×3e9]
enum4linux complete on Sun Jun 26 09:50:35 2022
```

Question 3

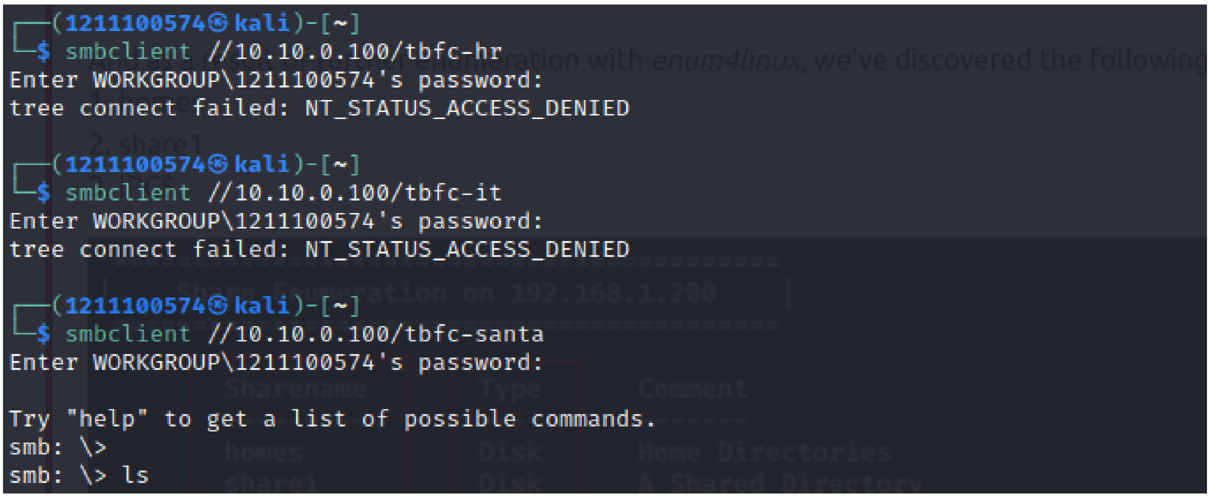
Use -S command to find shares



```
 ====================================
|    Share Enumeration on 10.10.0.100   |
 ====================================

        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))
```

Question 4

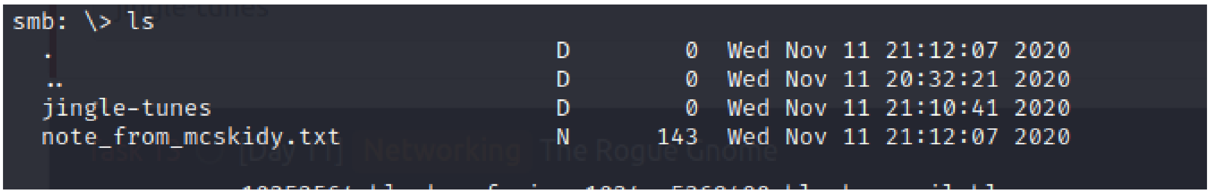
Trying no password on the shares until 1 gave access



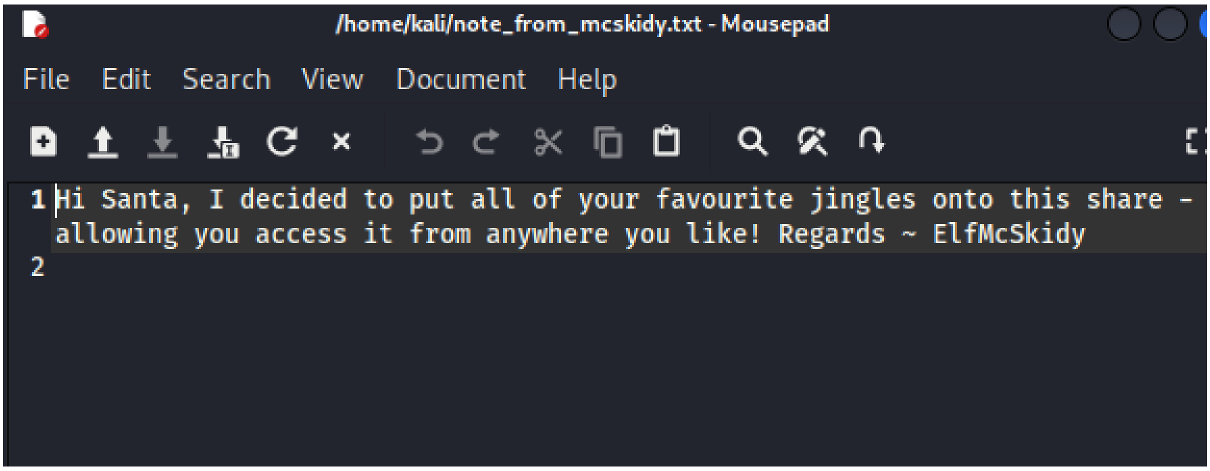
```
┌──(1211100574㉿kali)-[~]
└─$ smbclient //10.10.0.100/tbfc-hr
Enter WORKGROUP\1211100574's password:
tree connect failed: NT_STATUS_ACCESS_DENIED

┌──(1211100574㉿kali)-[~]
└─$ smbclient //10.10.0.100/tbfc-it
Enter WORKGROUP\1211100574's password:
tree connect failed: NT_STATUS_ACCESS_DENIED

┌──(1211100574㉿kali)-[~]
└─$ smbclient //10.10.0.100/tbfc-santa
Enter WORKGROUP\1211100574's password:

Try "help" to get a list of possible commands.
smb: \>
smb: \> ls
```

Question 5

Use ls to find directories and get the text file from mcskidy



```
smb: \> ls
  .                                   D        0  Wed Nov 11 21:12:07 2020
  ..                                  D        0  Wed Nov 11 20:32:21 2020
  jingle-tunes                        D        0  Wed Nov 11 21:10:41 2020
  note_from_mcskidy.txt               N      143  Wed Nov 11 21:12:07 2020
```



```
Hi Santa, I decided to put all of your favourite jingles onto this share - allowing you access it from anywhere you like! Regards ~ ElfMcSkidy
```