PSP0201

Week 3

Writeup

Group Name: 404 Not Found

Members

ID	Name	Role
1211102687	Emily Phang Ru Ying	Leader
1211102975	Loi Xinyi	Member
1211102753	Lim Cai QIng	Member
1211102751	Teo Yu Jie	Member

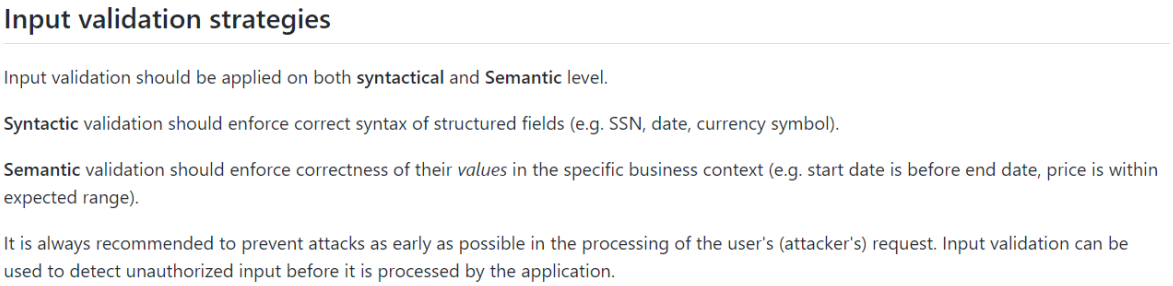
Day 6: Web Exploitation – Be careful with what you wish on a Christmas night

Tools used: Kali Linux, Firefox, OWASP ZAP

Solution/walkthrough:

Question 1

We got the answers by reading the OWASP cheat sheet series.



Input validation strategies

Input validation should be applied on both **syntactical** and **Semantic** level.

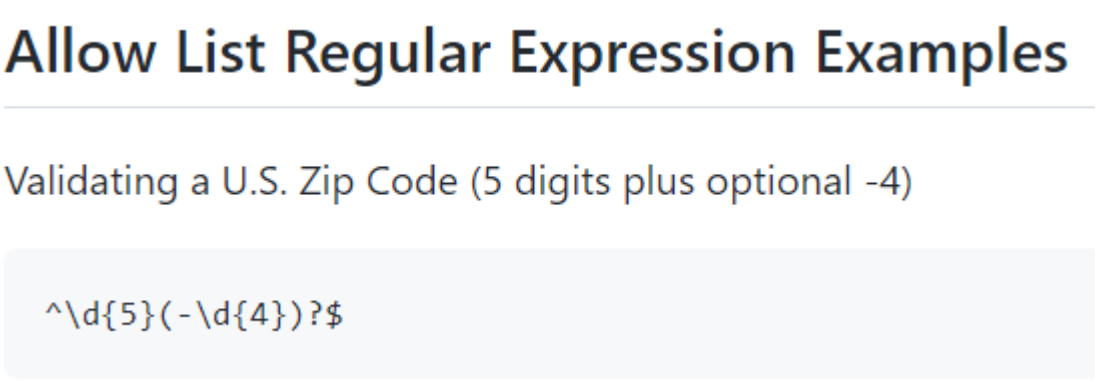
Syntactic validation should enforce correct syntax of structured fields (e.g. SSN, date, currency symbol).

Semantic validation should enforce correctness of their *values* in the specific business context (e.g. start date is before end date, price is within expected range).

It is always recommended to prevent attacks as early as possible in the processing of the user's (attacker's) request. Input validation can be used to detect unauthorized input before it is processed by the application.

Question 2

We got the answers by reading the OWASP cheat sheet series.



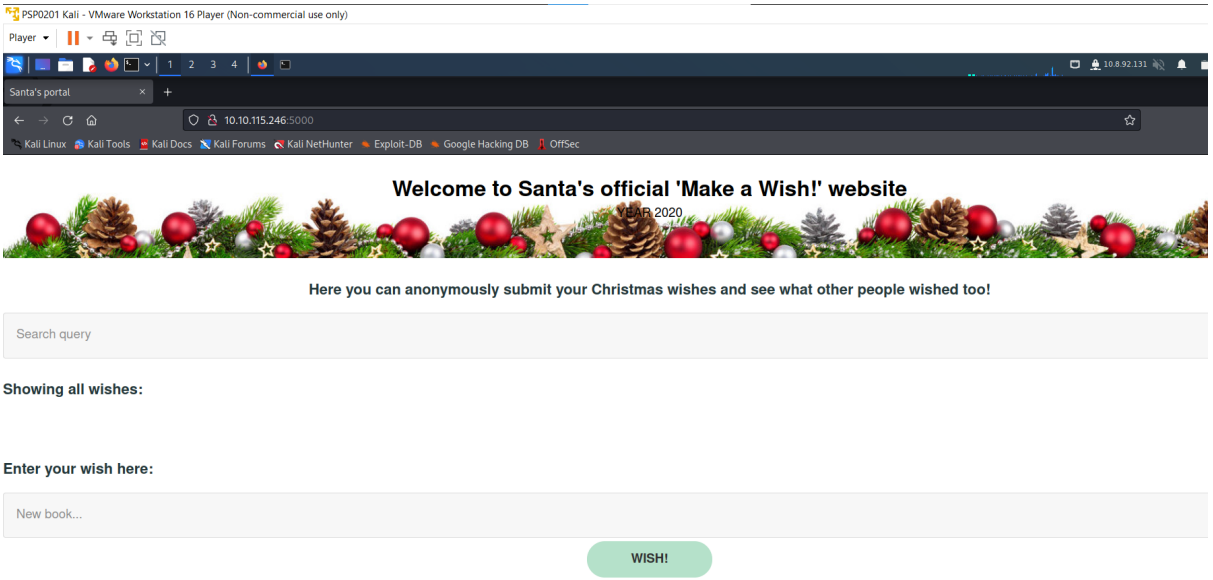
Allow List Regular Expression Examples

Validating a U.S. Zip Code (5 digits plus optional -4)

```
^\d{5}(-\d{4})?$
```

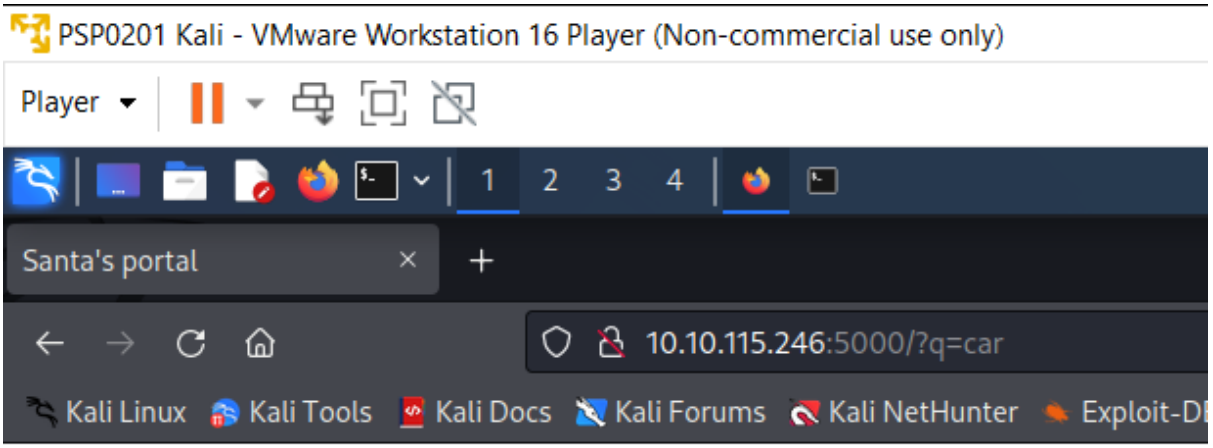
Question 3

We were shown Santa's official 'Make a Wish!' website. A stored XSS was used as a vulnerability type to exploit the application as users are allowed to post their wishes under the post.



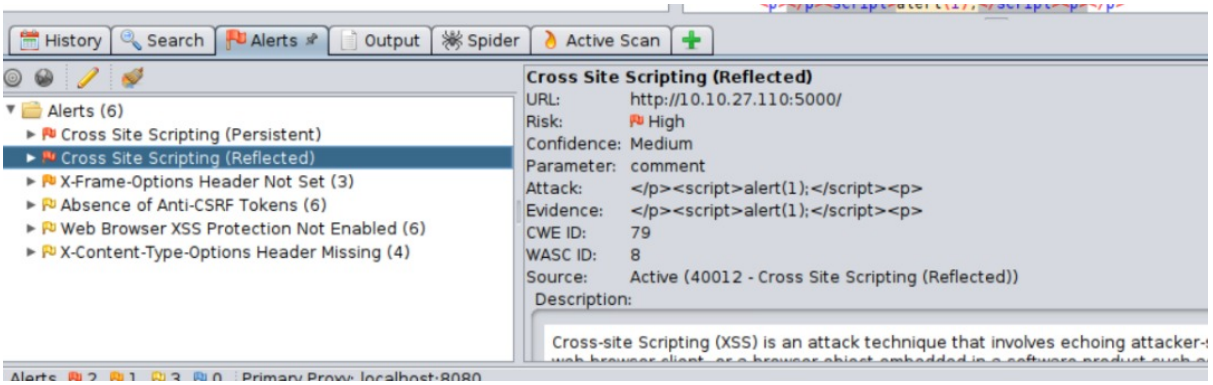
Question 4

After keying in car in the query search bar, q was added as a query string to our browser search bar.



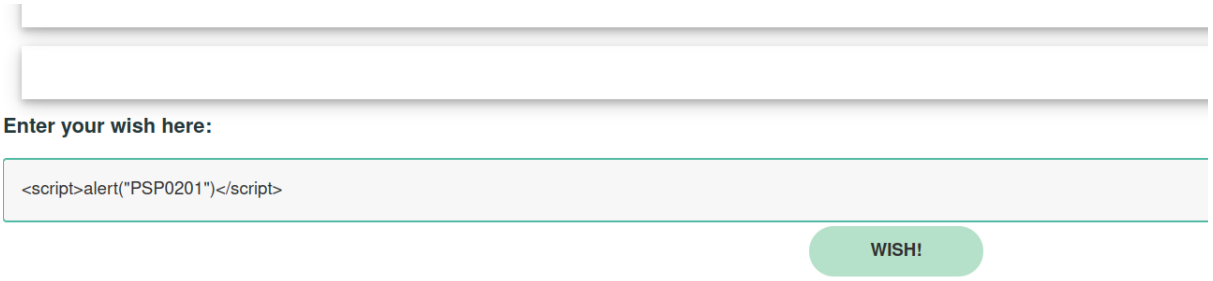
Question 5

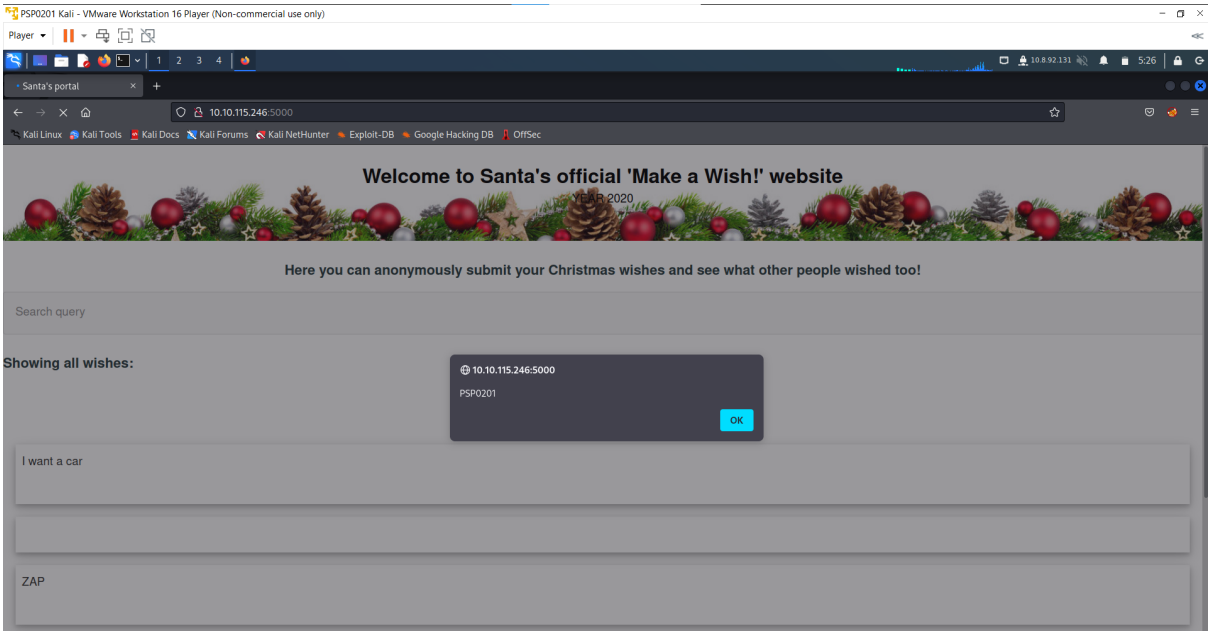
We run a ZAP (zaproxy) automated scan on the target. There are 2 XSS alerts of high priority in the scan as there are only 2 types of XSS shown which are persistent and reflected.



Question 6

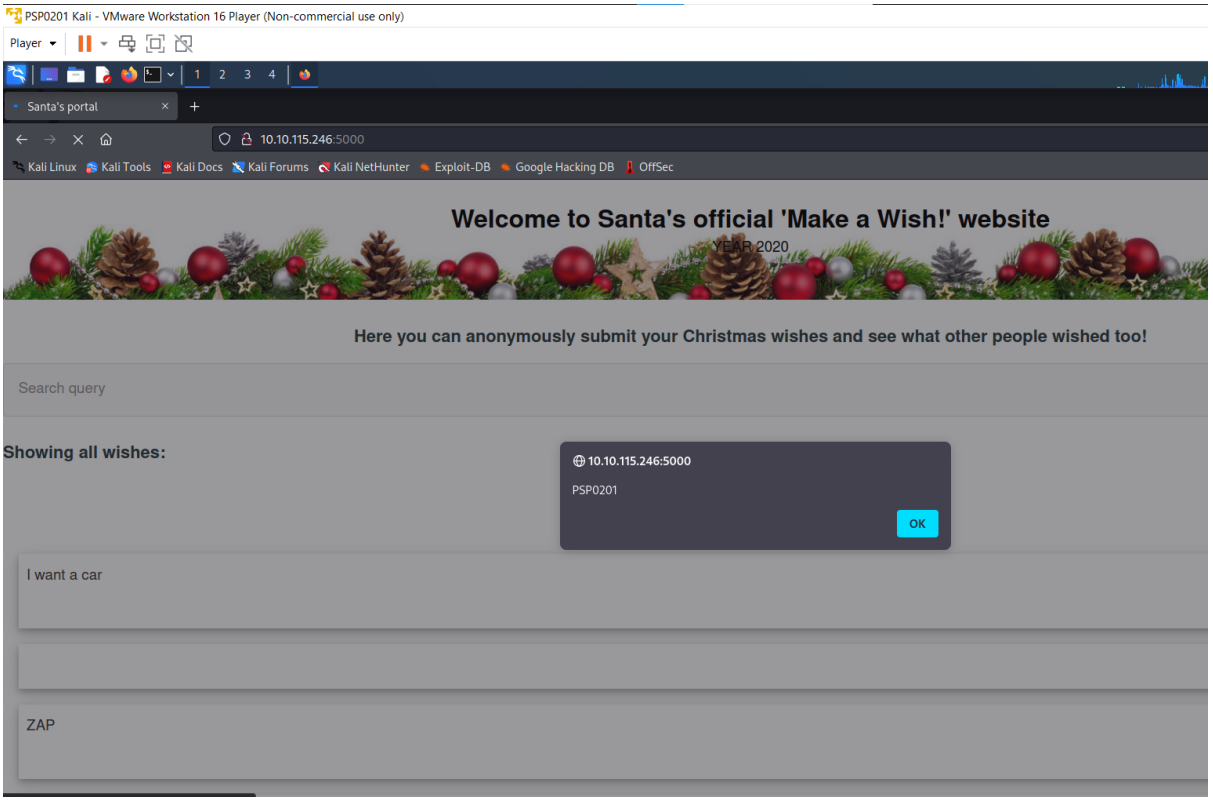
We substitute the 1 in the bracket with"PSP0201" to show an alert saying "PSP0201".





Question 7

After closing the browser and revisiting the site MACHINE-IP:5000, our attack persists.



Thought Process/Methodology:

After reading the OWASP cheat sheet series, we can match the input validation level with the correct description. We were also shown the regular expression used to validate a US Zip code. Then, we paste the machine's IP into the browser's search bar with port 5000. We were shown Santa's official 'Make a Wish!' website. We keyed in “car” in the query search bar and q was added as a query string to our browser search bar. We launched the OWASP ZAP Application and ran a ZAP automated scan on the URL to Santa's official 'Make a Wish!' website. We were shown 2 XSS alerts of high priority in the scan because there are only 2 types of XSS shown which are persistent and reflected. Then we wrote a Javascript code on the wish text box and substituted "PSP0201" into the bracket to show an alert saying "PSP0201". We closed our browser and revisited the site MACHINE-IP:5000 and found out that our XSS attack persists.

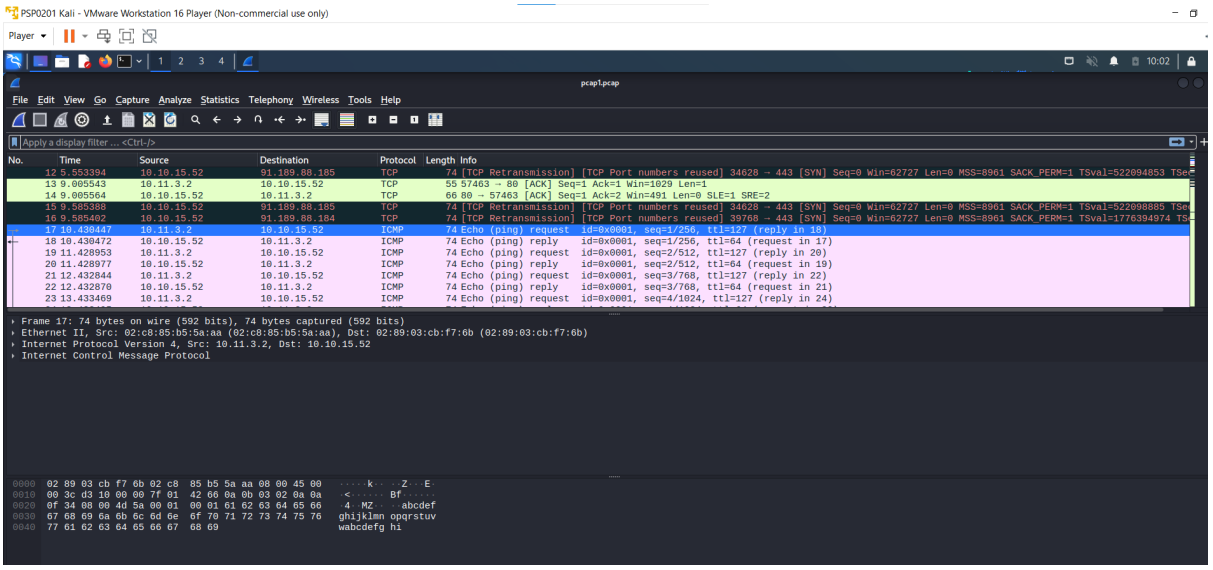
Day 7: Networking – The Grinch Really Did Steal Christmas

Tools used: Kali Linux, Wireshark

Solution/walkthrough:

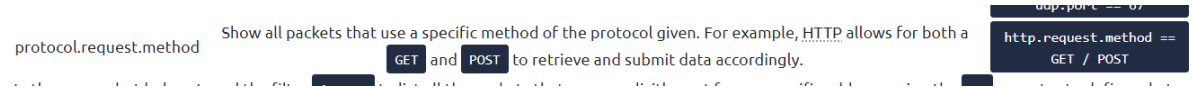
Question 1

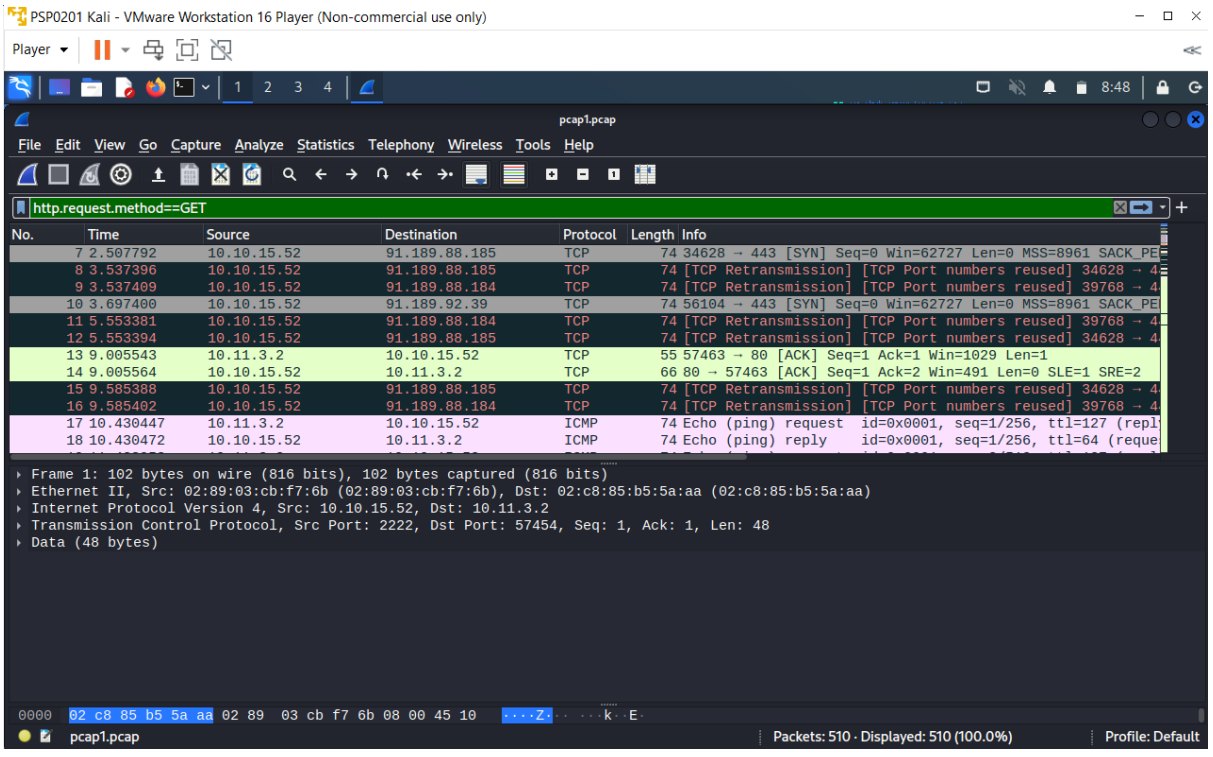
We opened "pcap1.pcap" in Wireshark, and the IP address that initiates an ICMP/ping was shown.



Question 2

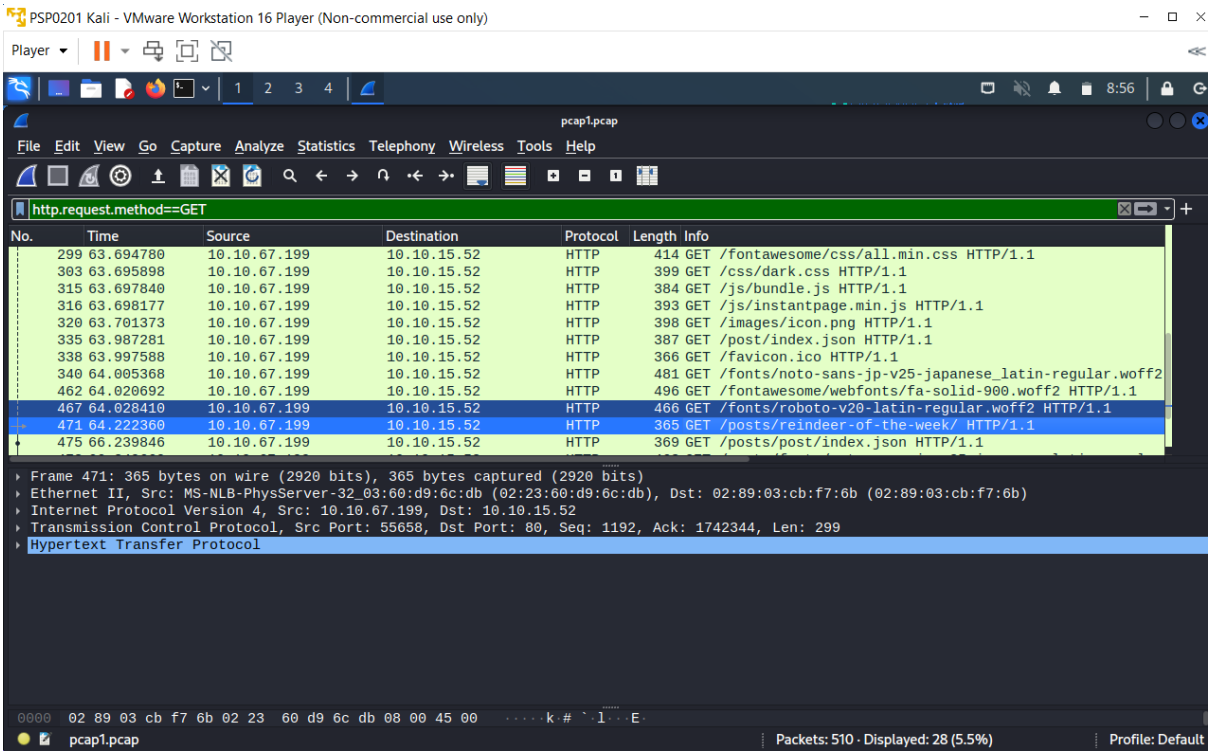
The filter to only see HTTP GET requests were already shown in the description of the task.





Question 3

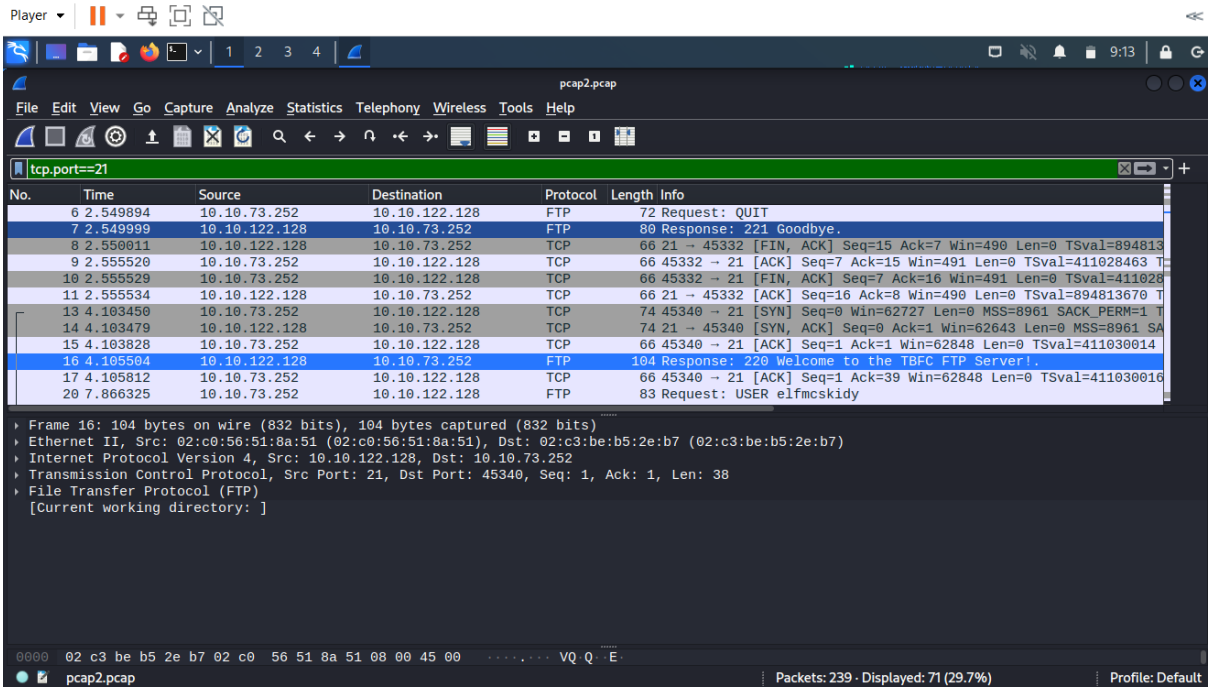
After filtering "pcap1.pcap" in Wireshark, we find /post/ to determine the name of the article that the IP address "10.10.67.199" visited.



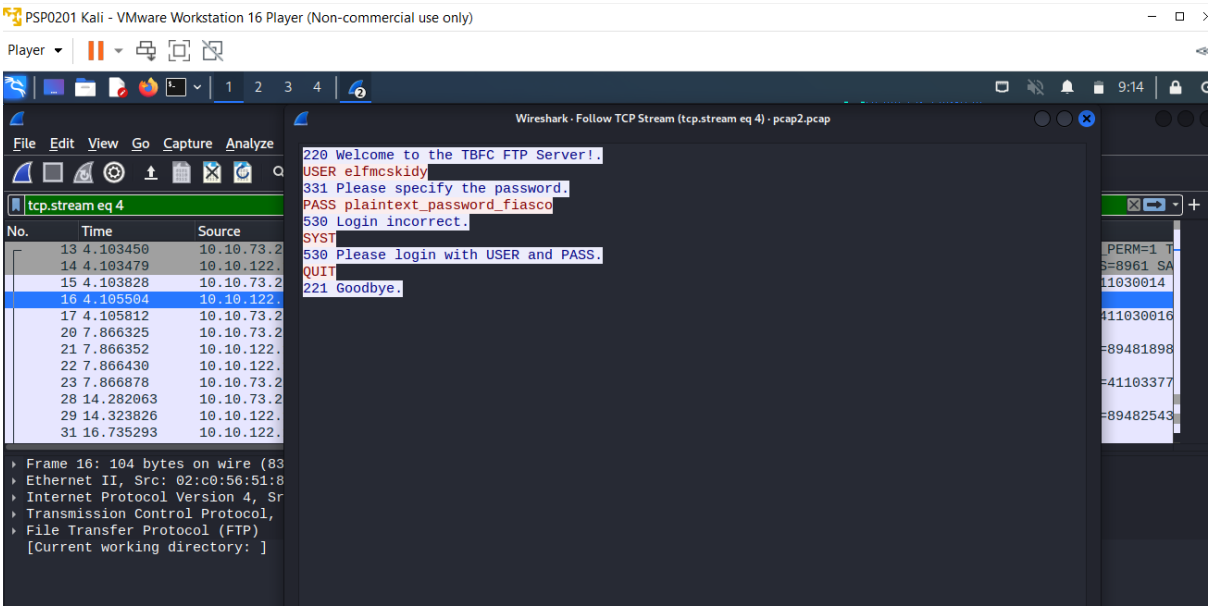
Question 4

We searched the FTP traffic by using the actual port that it was involved. The filter command was shown in the task description.

ip.dst	Show all packets that are destined to the specified IP address	ip.dst == 192.168.1.1
tcp/udp.port	Show all packets that are sent via the protocol and port specified	tcp.port == 22 / udp.port == 67

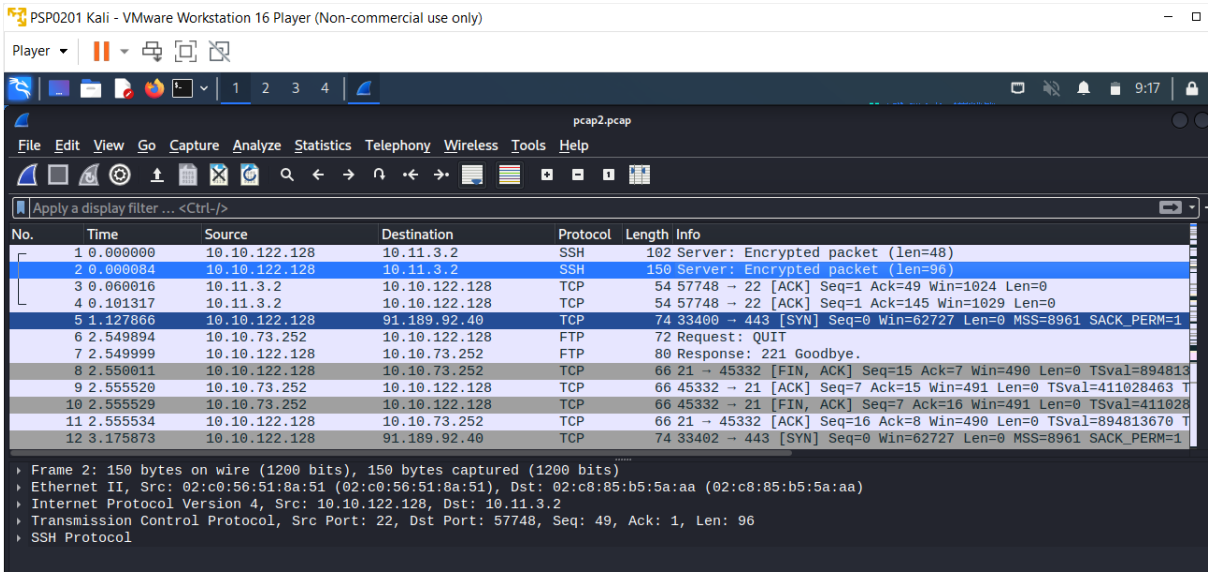


Then we followed the TCP stream of the log that was successfully login and was shown with the leaked user's password.



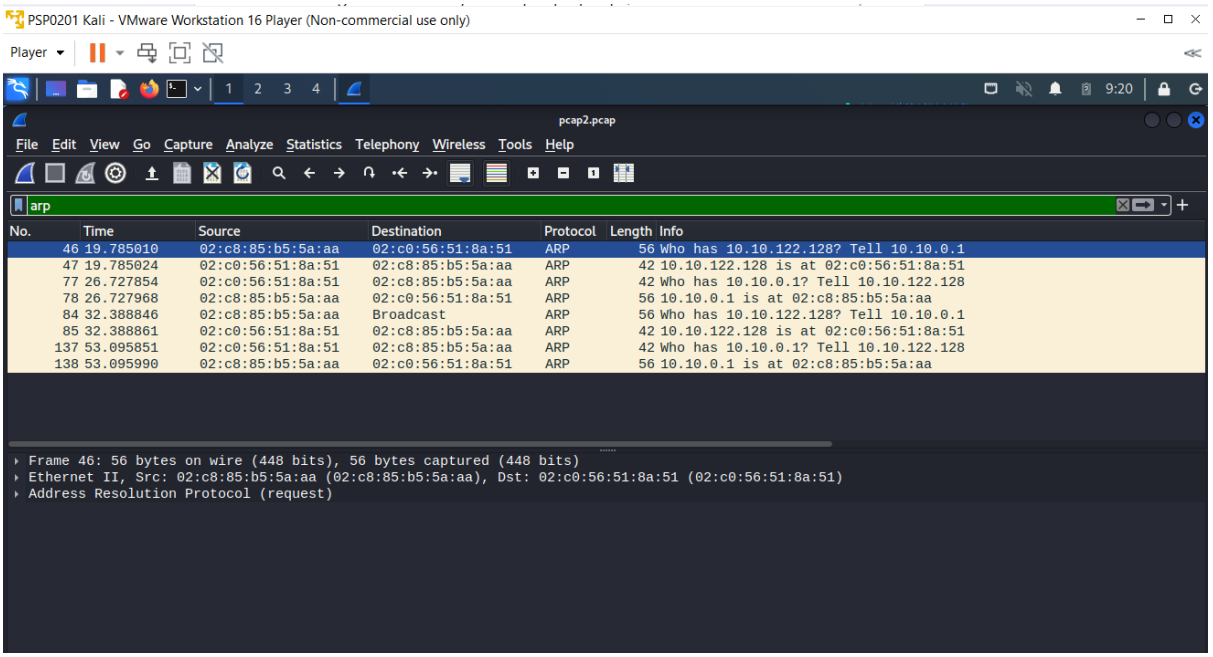
Question 5

The protocol for the encrypted packet is SSH.



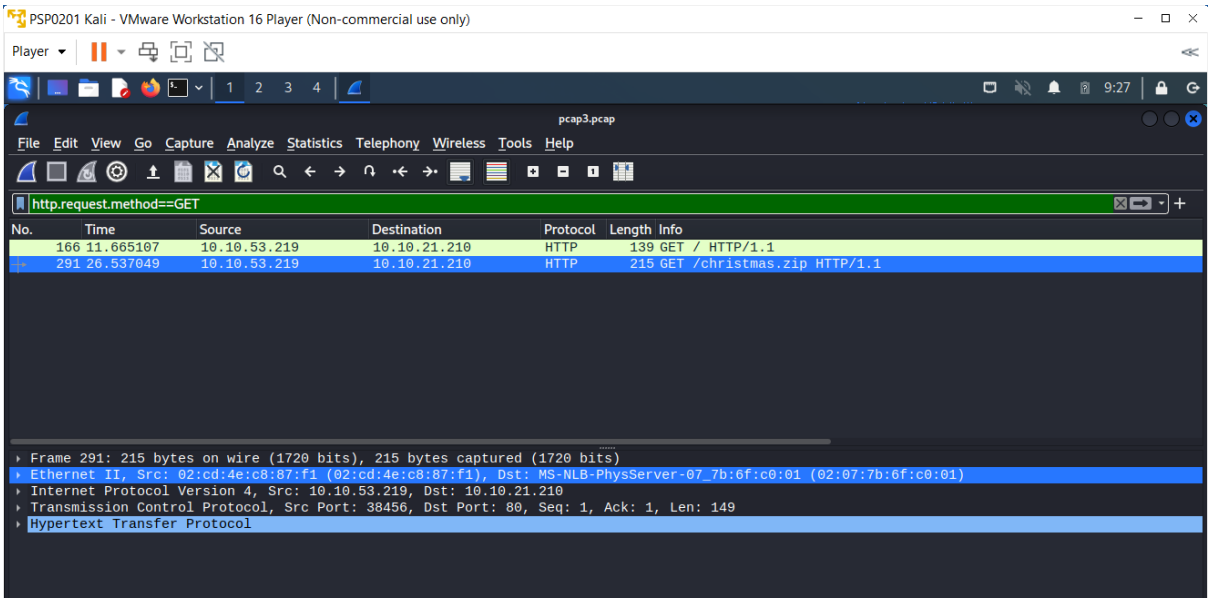
Question 6

After applying the arp filter to pcap2.pcap in Wireshark, it was shown who has 10.10.122.128.

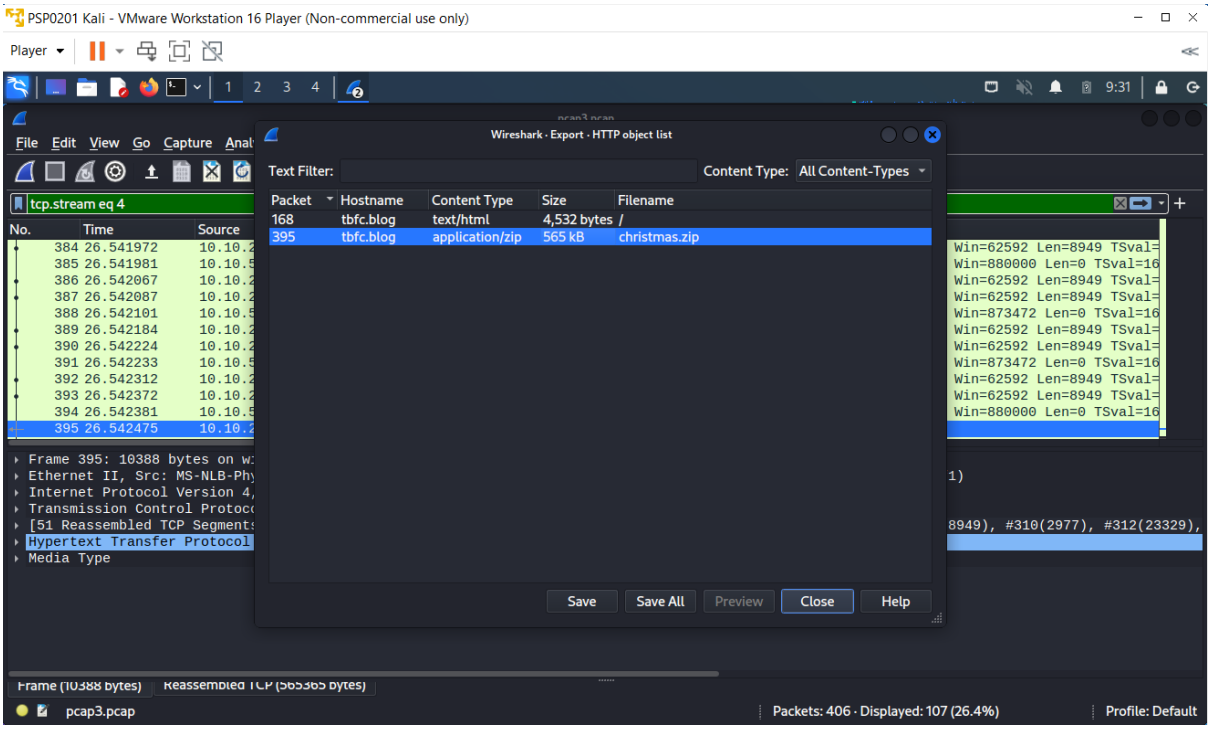


Question 7

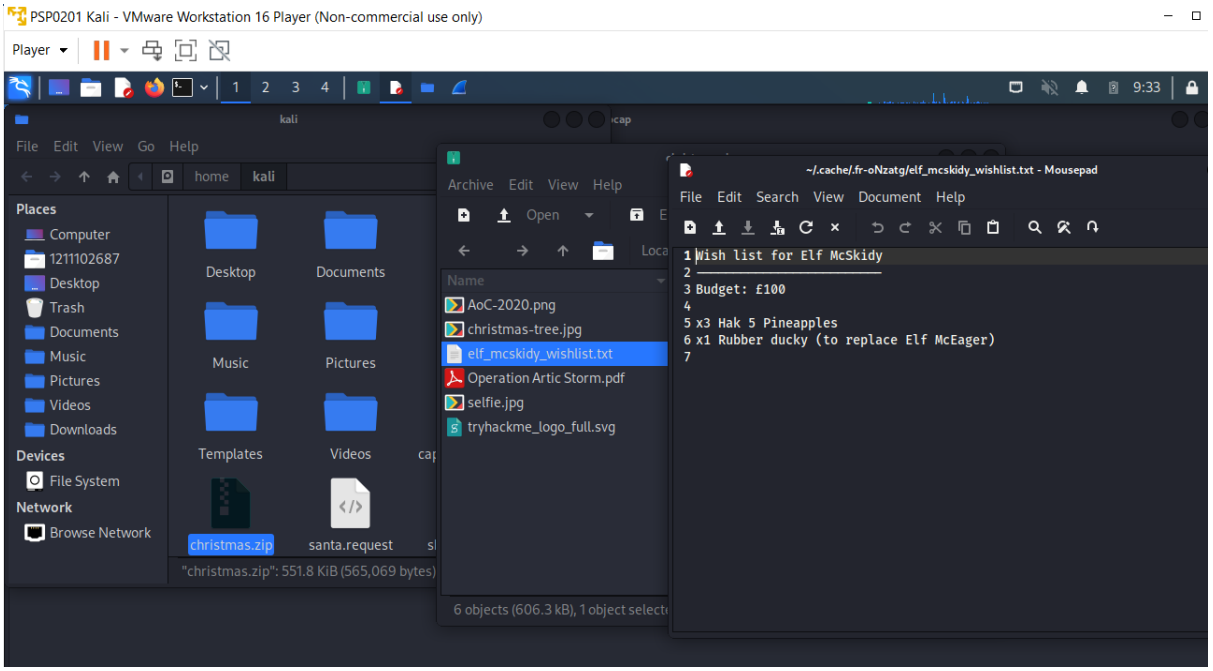
We used HTTP GET request to filter pcap3.pcap and found a Christmas zip file.



We export the christmas zip file.

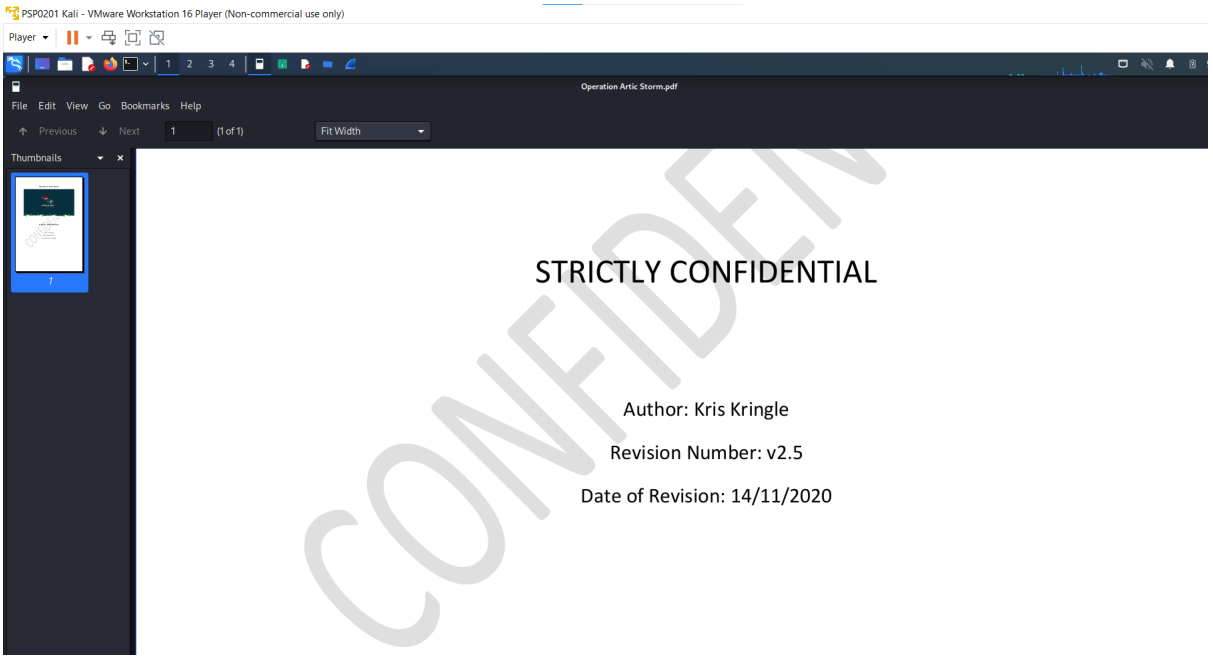


When we opened the christmas zip file, we were shown multiple files. In the elf_mcskidy_wishlist.txt, the item that will be used to replace Elf McEager was shown.



Question 8

After opening Operation Artic Storm.pdf the author for the pdf was shown.



Thought Process/Methodology:

We downloaded the task files from THM. Then, we opened “pcap1.pcap” in Wireshark, and the IP address that initiates an ICMP/ping was shown. We filtered “pcap1.pcap” by using the HTTP GET request and find /post/ to find what article did the IP address “10.10.67.199” visited. We searched the FTP traffic by using the actual port that it was involved. Then we followed the TCP stream that did a successful login and was shown the leaked password. We found out that the protocol for the encrypted packet is SSH. We applied the arp filter to pcap2.pcap in Wireshark and were shown who has 10.10.122.128. We used HTTP GET request to filter pcap3.pcap and found a Christmas zip file. We proceeded to export the file. There were many files in the zip file. The item used to replace Elf McEagerin was shown in the elf_mcskidy_wishlist.txt. The author for Operation Artic Storm was in the Operation Artic Storm pdf too.

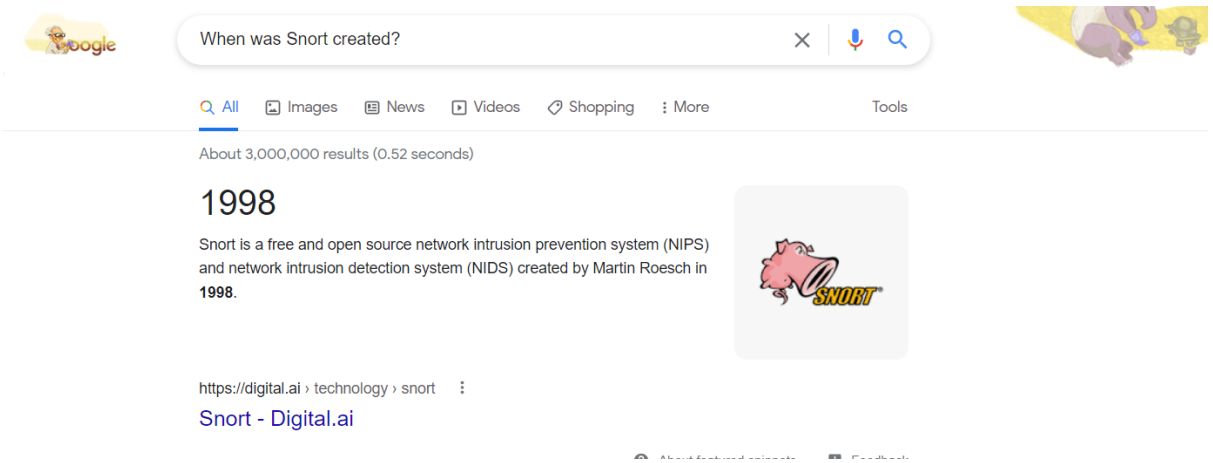
Day 8: Networking – What's Under the Christmas Tree?

Tools used: Kali Linux, Nmap

Solution/walkthrough:

Question 1

Based on Google, Snort was created in 1998.



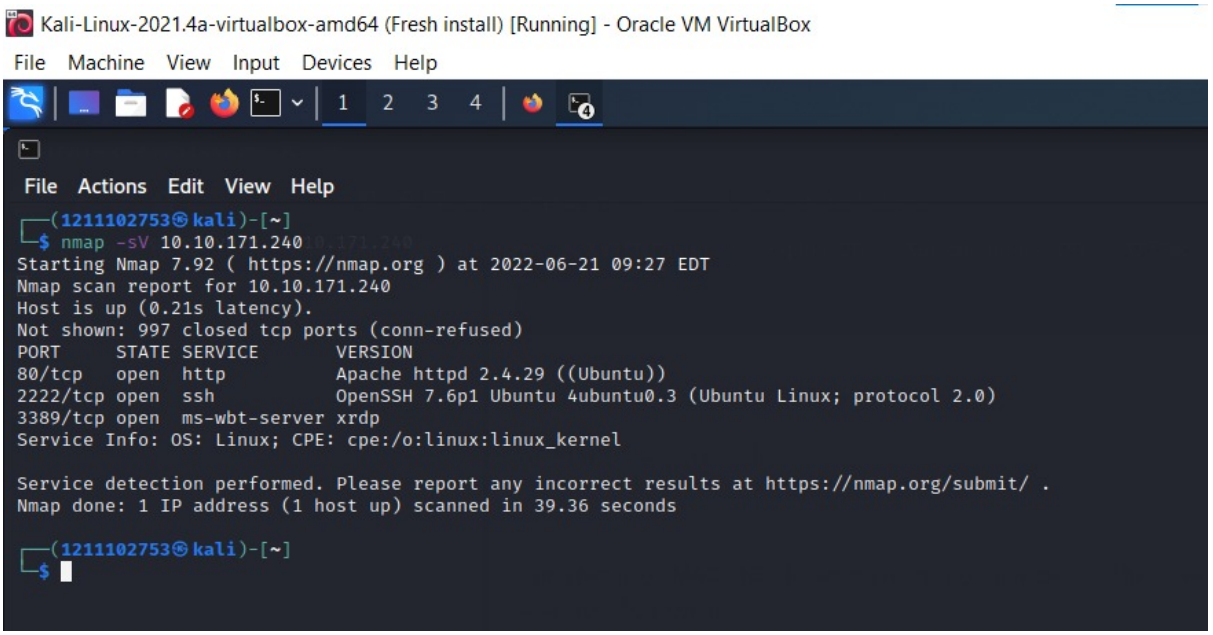
Question 2

We ran nmap on the machine's IP address and were shown the numbers of the 3 ports.

```
┌──(1211102753㉿kali)-[~]
└─$ nmap 10.10.171.240
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-21 09:24 EDT
Nmap scan report for 10.10.171.240
Host is up (0.22s 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 50.82 seconds
```

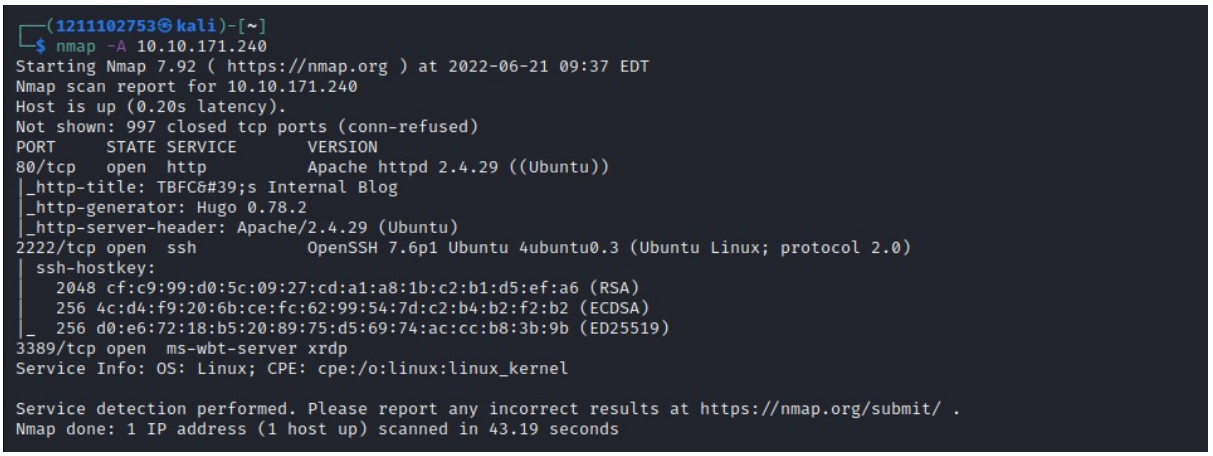
Question 3, 4 and 5

We ran the -sV command on nmap to determine the name of the Linux distribution that is running. We can also see the version of Apache and the service that shows what is running on port 2222 through the same command. (Sv Scan the host using TCP and perform version fingerprinting)



Question 6

We use -A command in Nmap to retrieve the "HTTP-TITLE" of the webserver.



Thought Process/Methodology:

After running nmap on the machine's IP address we were shown the numbers of the 3 ports. Then, we ran the -sV command on nmap. By doing so, we get to determine the name of the Linux distribution that is running. We can also see the version of Apache and the service that shows what is running on port 2222 through the same command. -Sv command is used to scan the host using TCP and perform version fingerprinting. Lastly, we use -A command in Nmap to retrieve the "HTTP-TITLE" of the webserver. -A is used to scan the host to identify services running by matching against Nmap's database with OS detection.

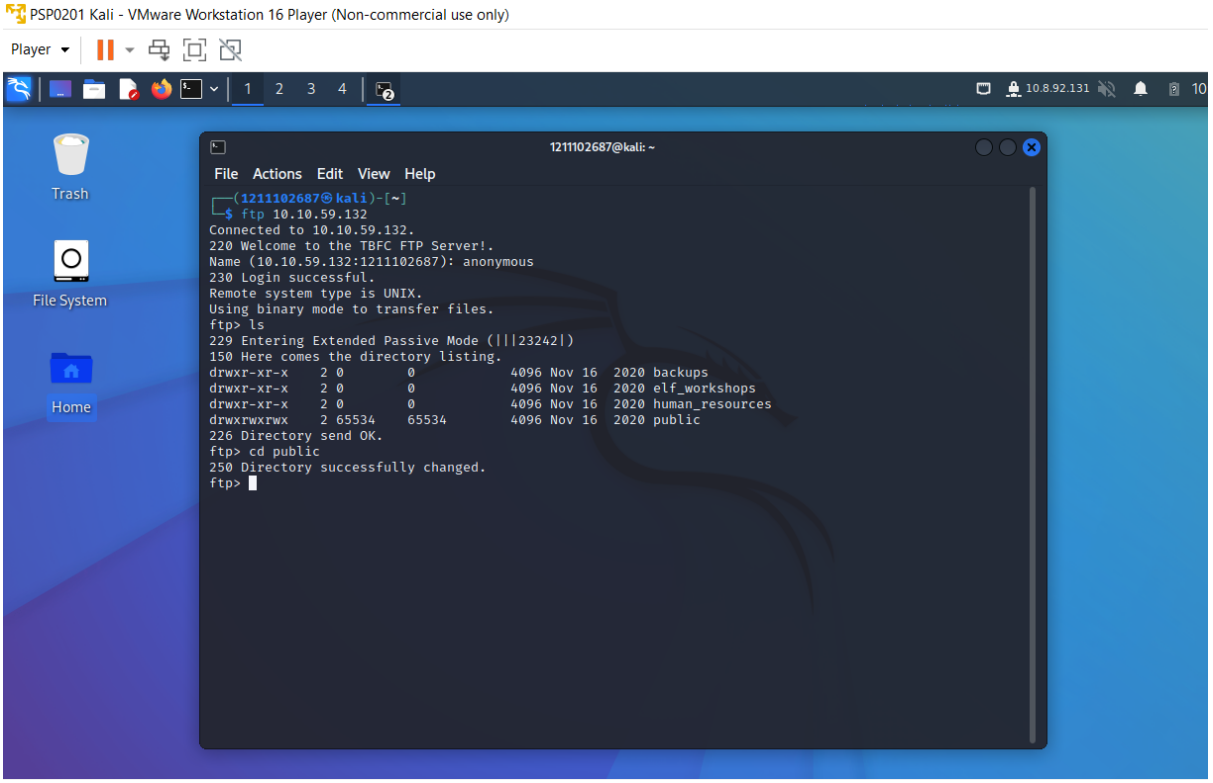
Day 9: Networking – Anyone can be Santa!

Tools used: Kali Linux, nanoshell, netcat listener

Solution/walkthrough:

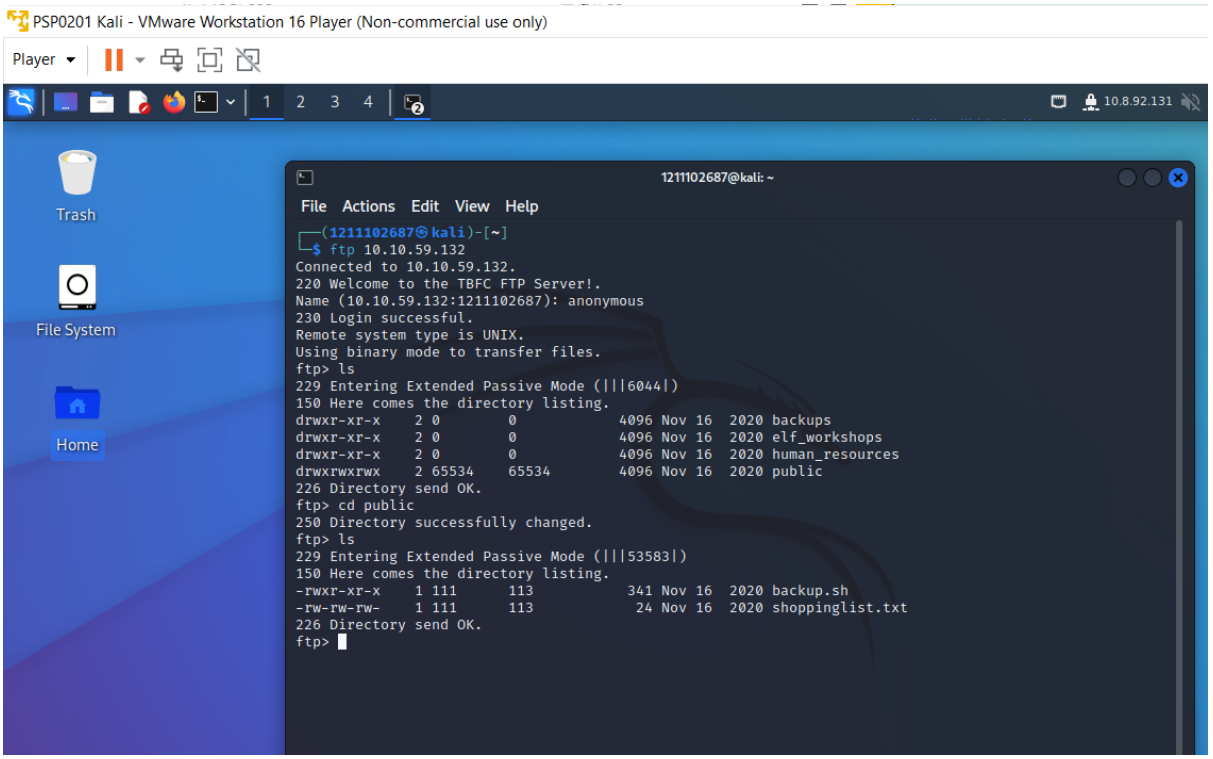
Question 1 and 2

We used ftp to connect to the machine's IP and keyed in anonymous as our name. We looked through the listed files by using the ls command. We found out that only public has data, we tried changing our directory to public and it was successful.



Question 3

We used the ls command again to list the contents in public and found a script and a text.



Question 4

We used the get command to get the files from the server onto our device.

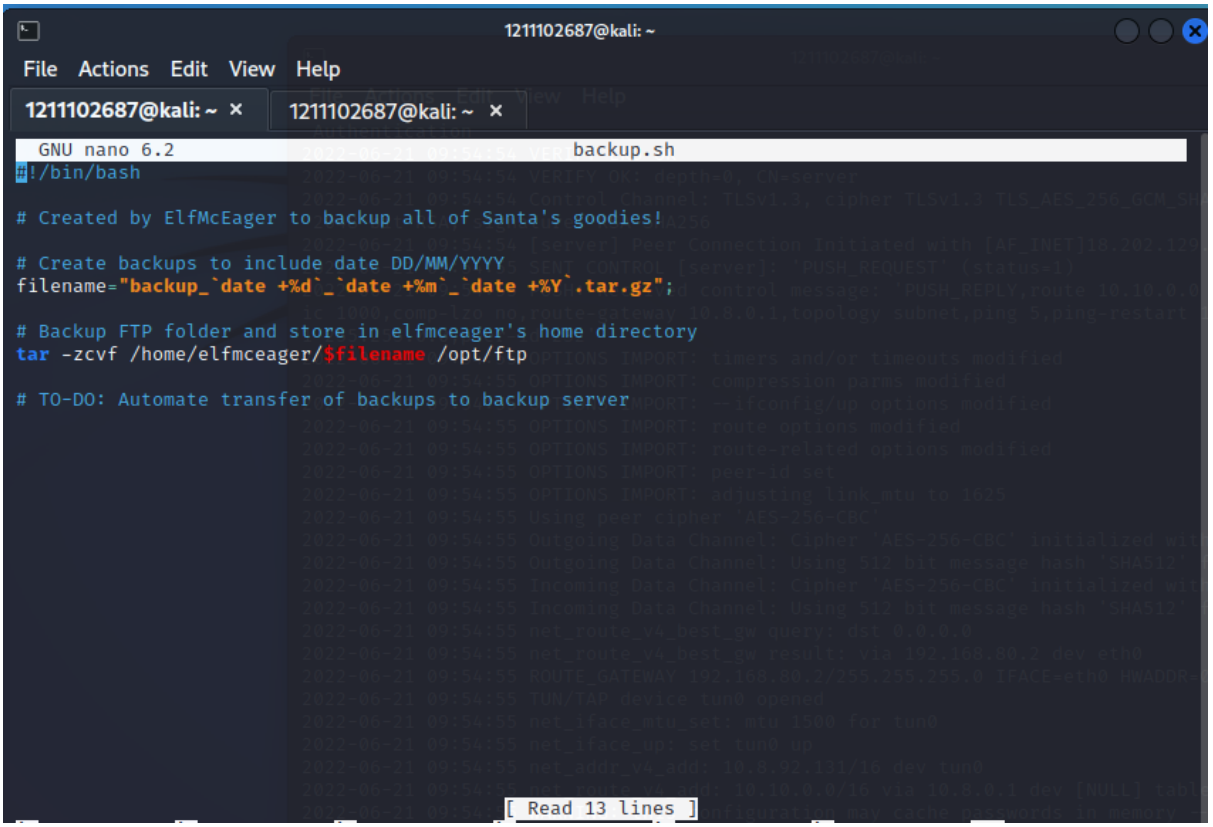
```
229 Entering Extended Passive Mode (|||53583|)
150 Here comes the directory listing.
-rwxr-xr-x    1 111      113           341 Nov 16  2020 backup.sh
-rw-rw-rw-    1 111      113            24 Nov 16  2020 shoppinglist.txt
226 Directory send OK.
ftp> get shoppinglist.txt
local: shoppinglist.txt remote: shoppinglist.txt
229 Entering Extended Passive Mode (|||48217|)
150 Opening BINARY mode data connection for shoppinglist.txt (24 bytes).
100% |*****************************************************|    24       30.55 KiB/s    00:00 ETA
226 Transfer complete.
24 bytes received in 00:00 (0.12 KiB/s)
ftp> get backup.sh
local: backup.sh remote: backup.sh
229 Entering Extended Passive Mode (|||61979|)
150 Opening BINARY mode data connection for backup.sh (341 bytes).
100% |*****************************************************|   341        6.13 MiB/s    00:00 ETA
226 Transfer complete.
341 bytes received in 00:00 (1.65 KiB/s)
ftp>
```

We used the cat command to view the contents of the shopping list text file.

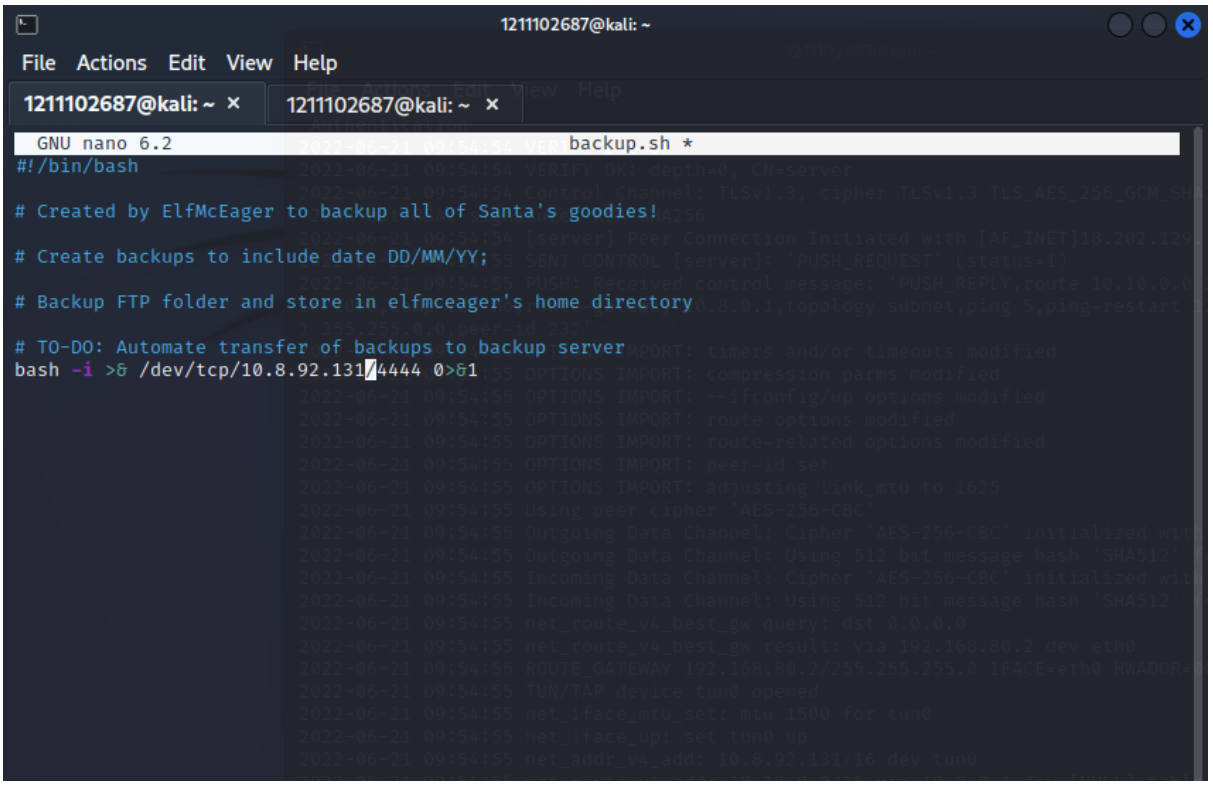
```
┌──(1211102687㉿kali)-[~]
└─$ ls
backup.sh          Desktop    Music      santa.request     Templates
captured.request   Documents  Pictures   shell.jpeg.php    Videos
christmas.zip      Downloads  Public     shoppinglist.txt  wordlist.txt

┌──(1211102687㉿kali)-[~]
└─$ cat shoppinglist.txt
The Polar Express Movie
```

Question 5

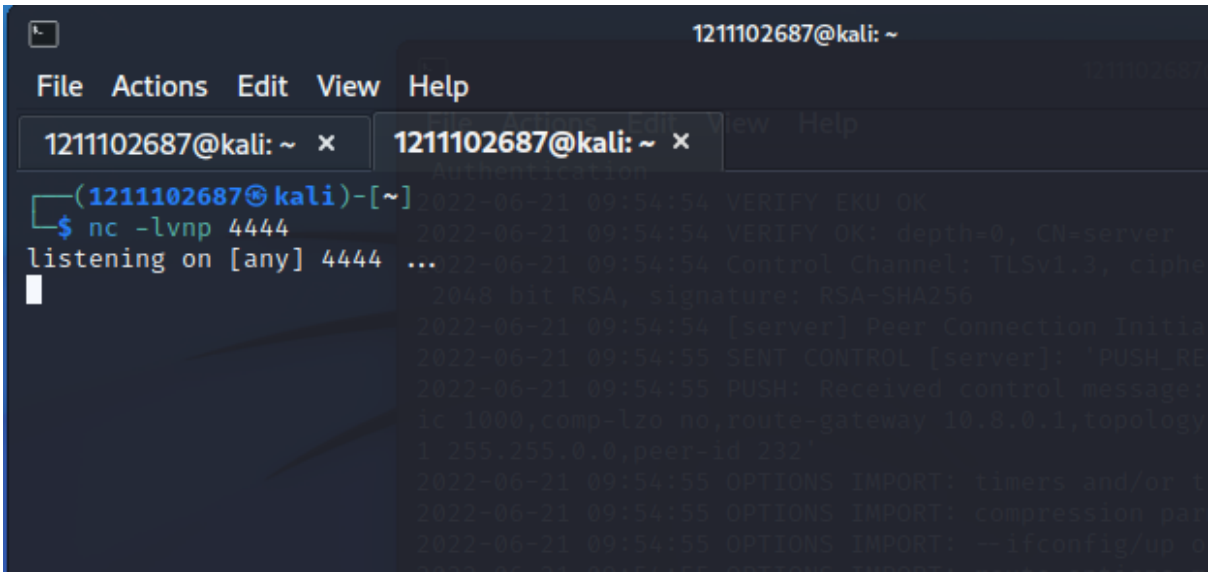
We open the backup.sh on our device using a terminal text editor, nano.



We generate a shell to our AttackBox.



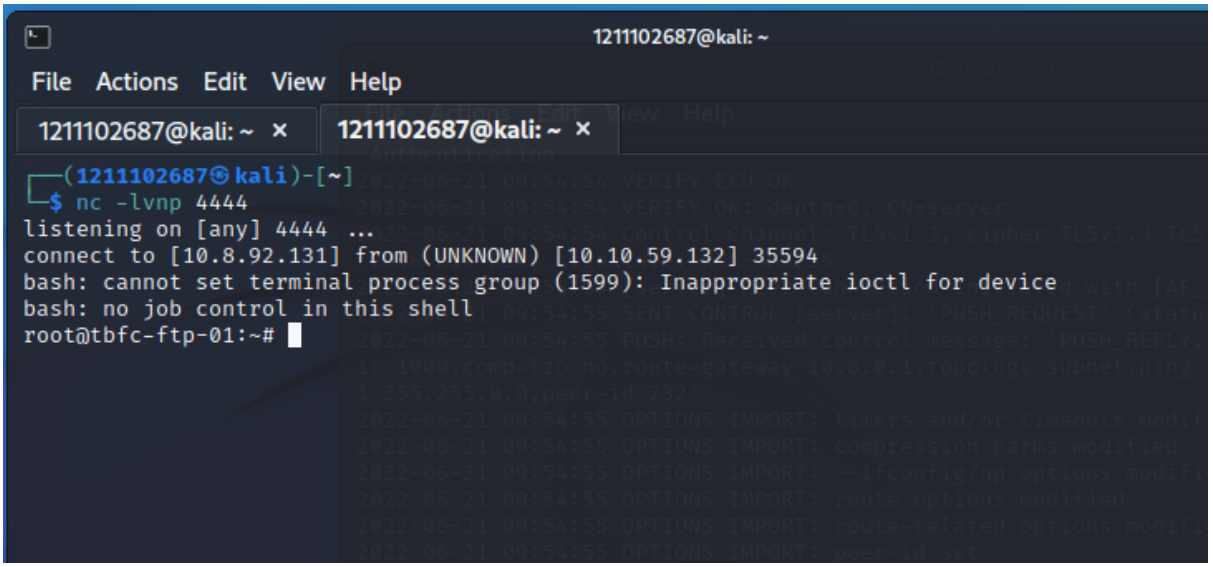
We set up a netcat listener to catch the connection on our AttackBox.



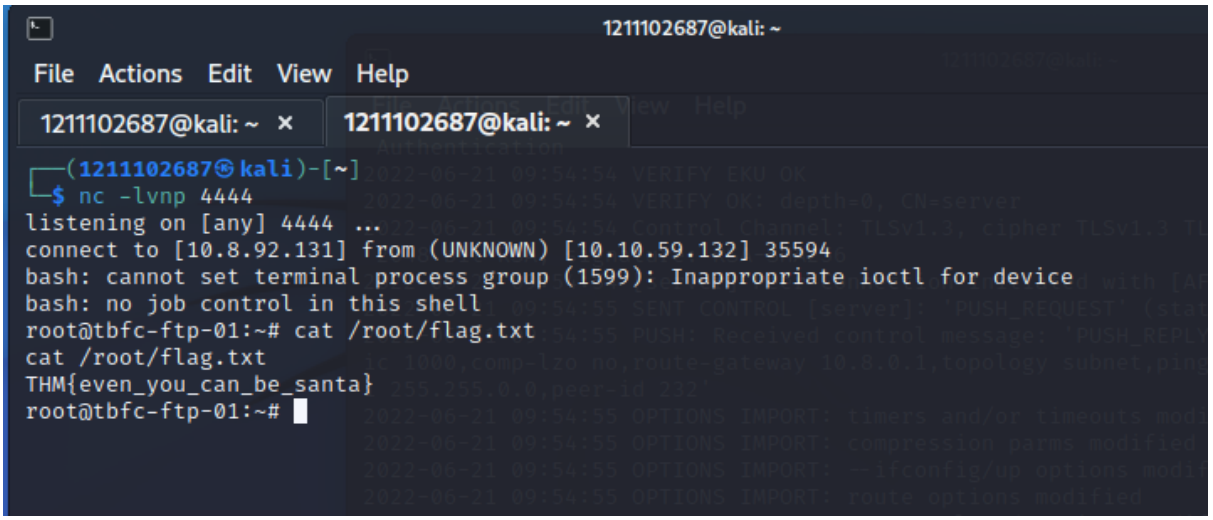
We used PUT to put the file into backup.sh.

```
(1211102687㉿kali)-[~]
$ ftp 10.10.59.132
Connected to 10.10.59.132.
220 Welcome to the TBFC FTP Server!.
Name (10.10.59.132:1211102687): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd public
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||29284|)
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
229 Entering Extended Passive Mode (|||30567|)
150 Ok to send data.
100% |***********************************************************|   276        3.75 MiB/s    00:00 ETA
226 Transfer complete.
276 bytes sent in 00:00 (0.69 KiB/s)
ftp>
```

We returned to our netcat listener and saw an output.



We used the cat command to view the flag text file and the flag was shown.



Thought Process/Methodology:

First, we used ftp to connect to the machine's IP and log in as an anonymous user. We looked through the listed files and found out that only public has data which means that we most likely are only given access to public. We tried changing our directory to public to check and it was successful. We list the contents in public and found a script and text file. We downloaded the files from the server onto our device. We viewed the contents in the shopping list text file. We opened backup.sh on our device using nanoshell and generate a shell to our AttackBox. Then we set up a netcat listener to catch the connection on our AttackBox. We updated the file into backup.sh. After 1 minute, we saw an output when we returned to our netcat listener. We view the flag text file and captured the flag.

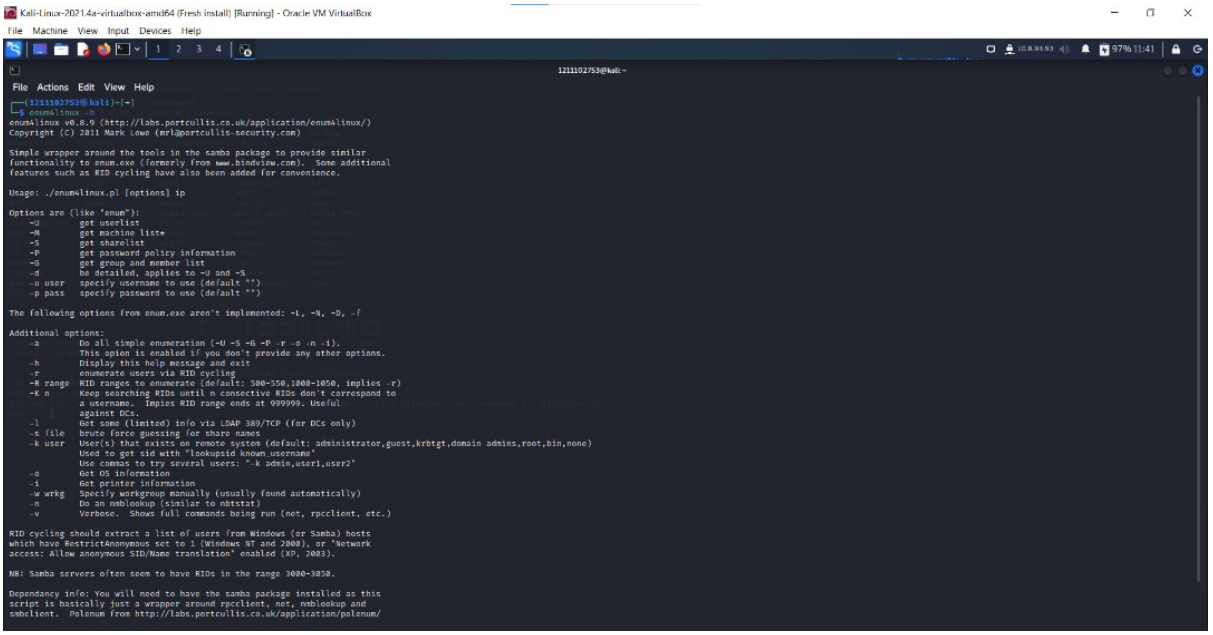
Day 10: Networking – Don't be sElfish!

Tools used: Kali Linux, enum4linux, smbclient

Solution/walkthrough:

Question 1

The answers can be obtained by viewing the help sheet of enum4linux.

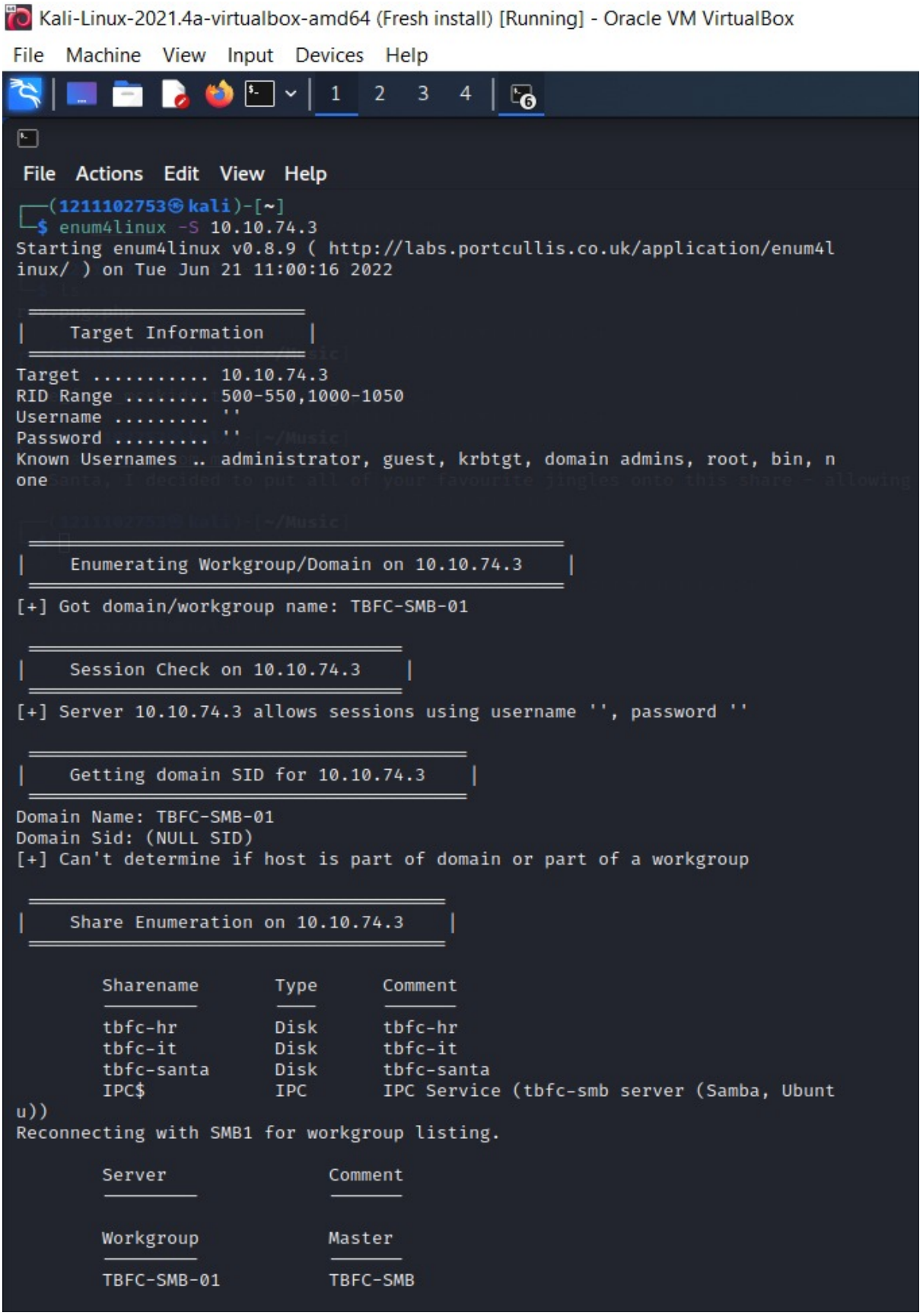


Question 2

We used the command -U to list the possible users on the Samba server.

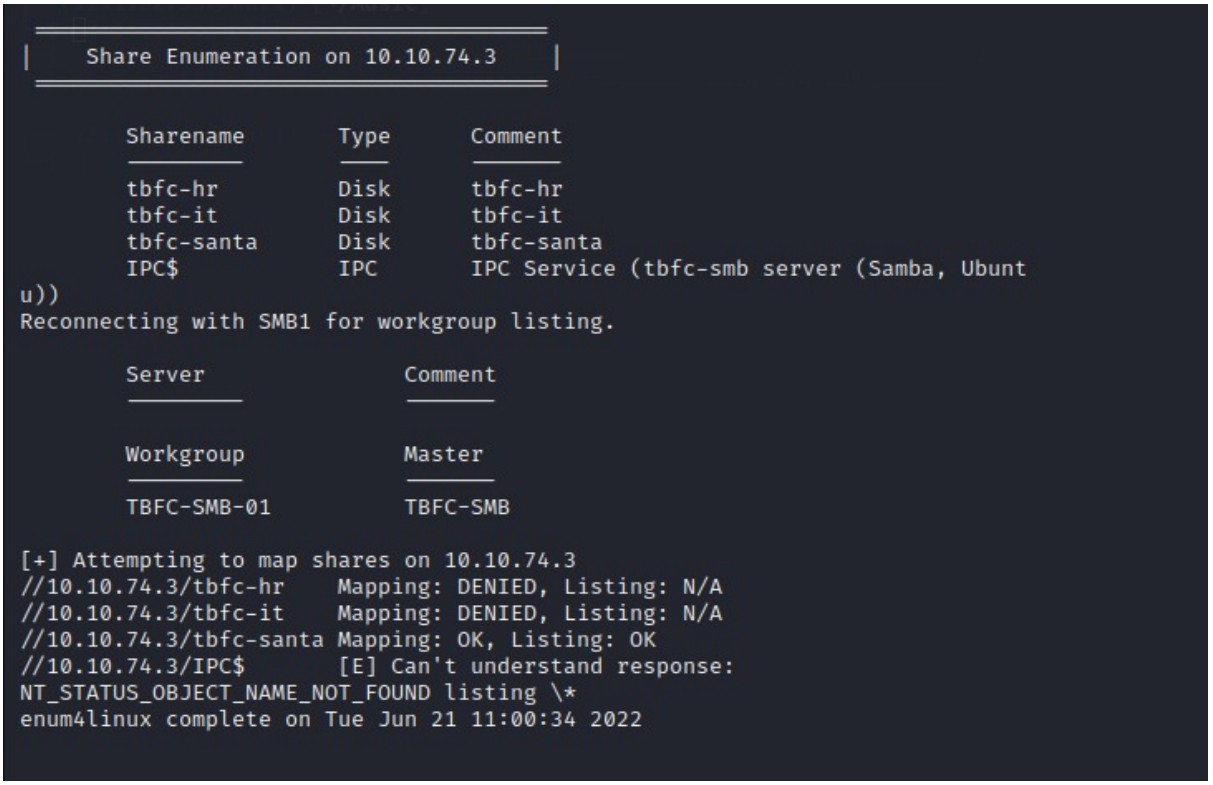
Question 3

We used the command -S to list the possible users on the Samba server.

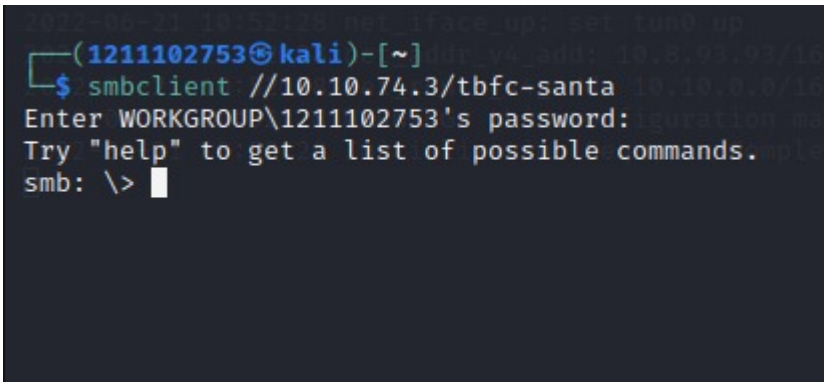


Question 4

In the Samba server share list, it was seen that the mapping and listing for tbfc-santa is the only one that is listed as OK.

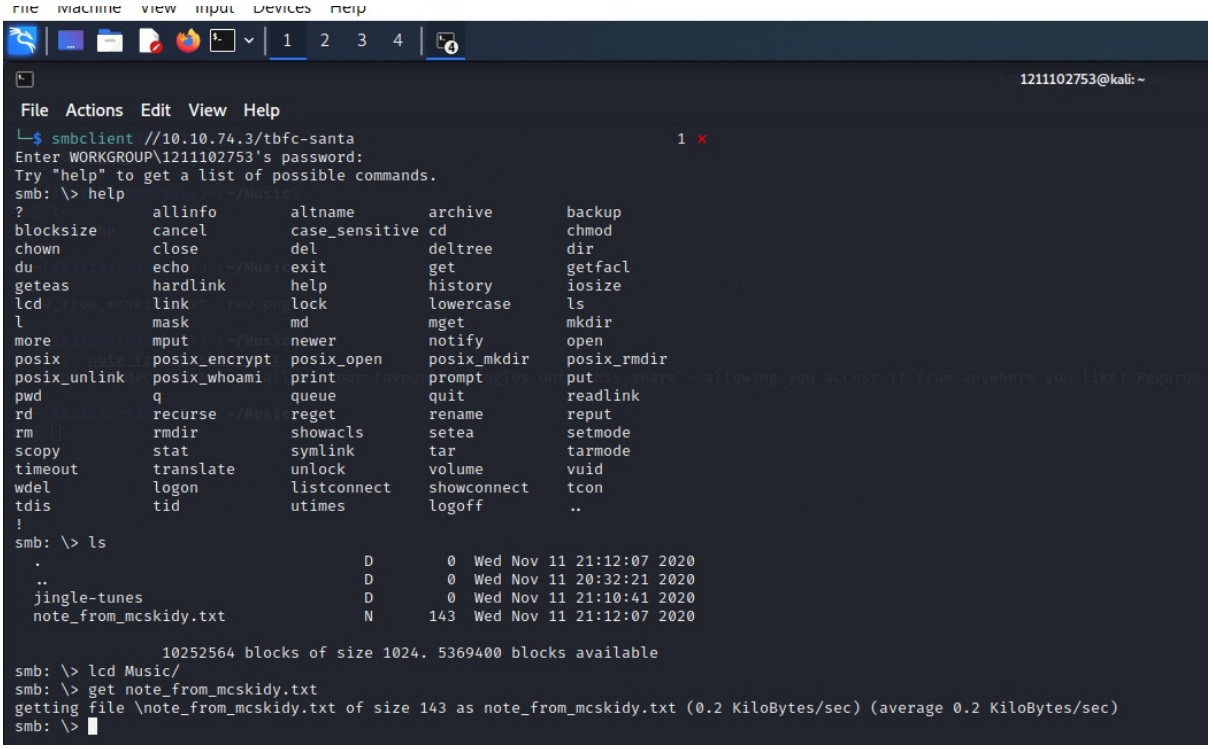


We checked whether tbfc-santa needed a password by using smbclient.



Question 5

We checked the lists available in tbfc-santa. We changed our current directory to our wanted directory(Music) and downloaded the note from mcskidy text file.



We viewed the content in the note from mcskidy text file and found out what directory mcskidy left for santa.

Thought Process/Methodology:

We started off by viewing the help sheet of enum4linux. We used the command -U to list the possible users on the Samba server and used the command -S to list the possible users on the Samba server. The mapping and listing for tbfc-santa is the only one that is listed as OK in the Samba server share list. Therefore, we tried checking tbfc-santa to check if it needed a password by using smbclient. In the end, it turns up that it does not need a password. Then, we checked the lists available in tbfc-santa. We changed our current directory to our wanted directory(Music) and downloaded the note from mcskidy text file. We viewed the content in the note from mcskidy text file and concluded that mcskidy left the directory of jingle-tunes for santa based on the note left in the text file.