

# PSP0201

## Week 3

# Writeup

Group Name: Gold

Members:

ID	Name	Role
1211101707	Nur'aina Binti Ikhwan Moeid	Leader
1211103984	Nur Afreen Junaidah Binti Noorul Mohamed Eliyas	Member
1211101519	Aisyah Binti Ahmad Komarolaili	Member
1211102590	Nur Hanisah Binti Mohd Pauzi	Member

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

Tools used: Kali Linux, Firefox

Solution/walkthrough:

### Question 1:

Examine the OWASP Cheat Sheet. What is the regular expression used to validate a US Zip code?

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

The screenshot shows a GitHub page for the OWASP Cheat Sheet Series. The URL is [github.com/OWASP/CheatSheetSeries/blob/master/cheatsheets/Input\\_Validation\\_Cheat\\_Sheet.md](https://github.com/OWASP/CheatSheetSeries/blob/master/cheatsheets/Input_Validation_Cheat_Sheet.md). The page content includes a section titled "Allow List Regular Expression Examples" which contains the regular expression `^\d{5}(-\d{4})?$/` for validating a U.S. Zip code.

### Question 2:

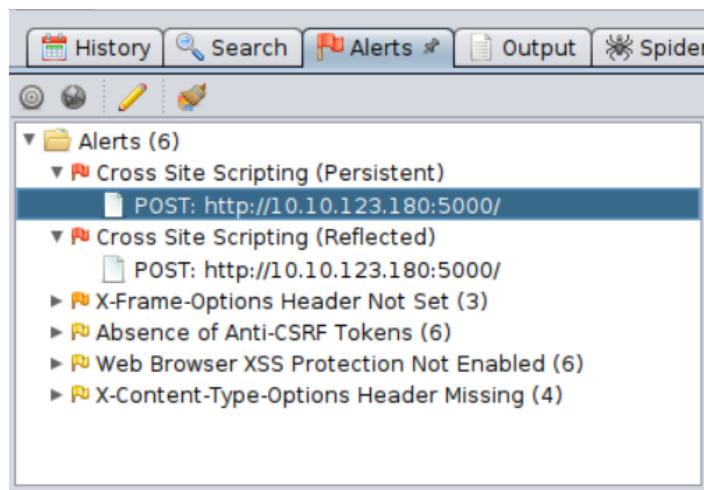
What vulnerability type was used to exploit the application? -Stored

### Question 3:

What query string can be abused to craft a reflected XSS? -q

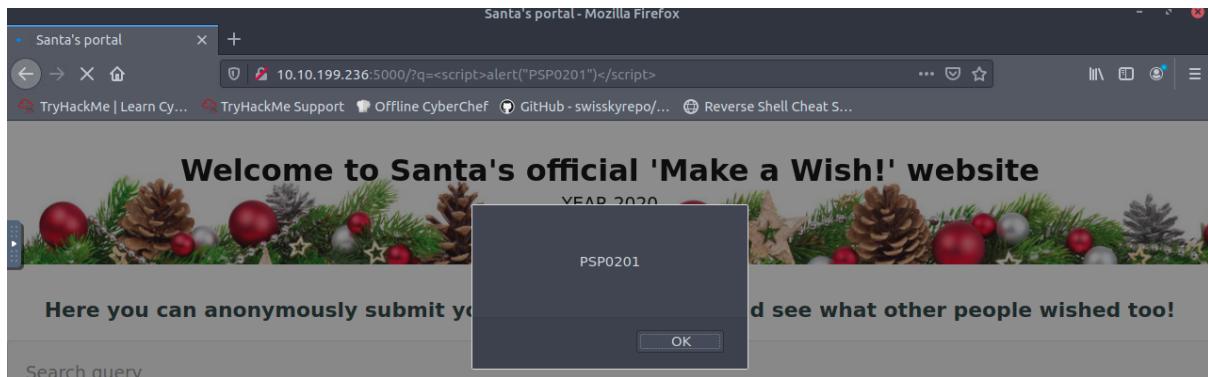
### Question 4:

Run a ZAP (zaproxy) automated scan on the target. How many XSS alerts of high priority are in the scan? -2



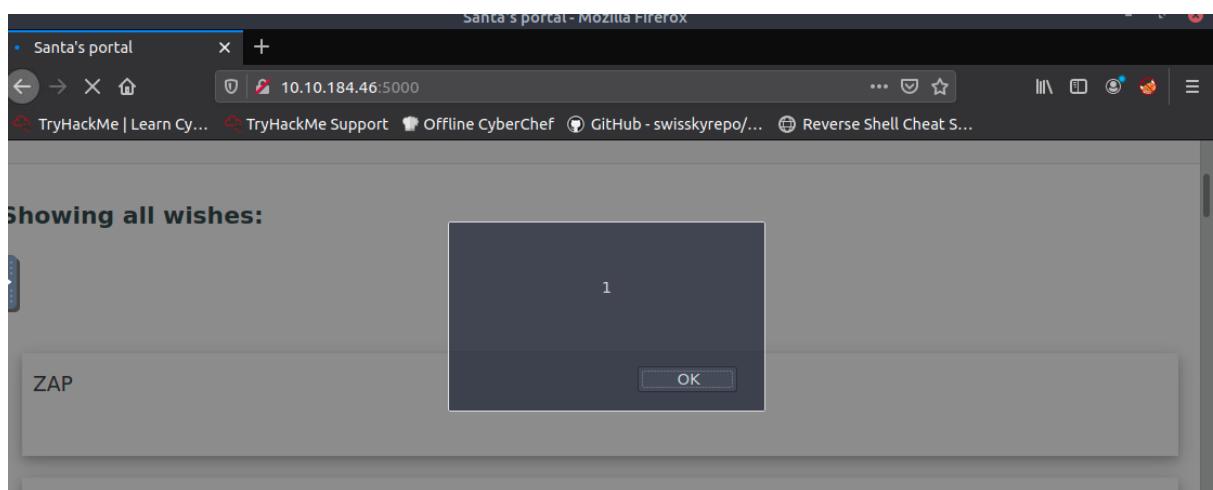
### Question 5:

What Javascript code should you put in the wish text box if you want to show an alert saying "PSP0201"? -<script>alert("PSP0201")</script>



### Question 6:

Close your browser and revisit the site MACHINE-IP:5000 again. Does your XSS attack persist? -yes



### **Thought Process/ Methodology:**

After typing in in IP address given along with ".5000", we will access Santa's official "Make a Wish!" website. When we type out a wish in the box provided, we can see from the URL that the query string abused to craft a reflected XSS is the letter "q". To attack the XSS, we launch the OWASP ZAP Application. On the application, we type in the URL of the site we wish to attack and the machine will do the work for you. From there, we can find details on the website such as the amount of XSS alerts are in the scan, which was 2. When the application is closed, the attack persists.

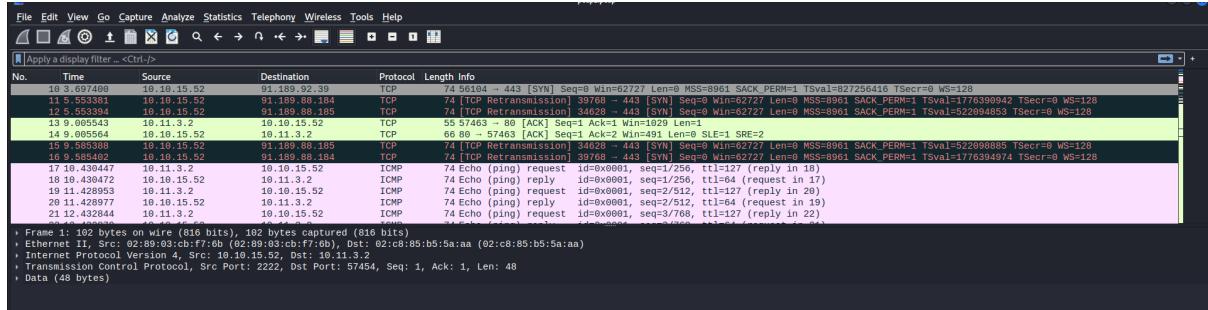
## Day 7: Networking - The Grinch Really Did Steal Christmas

Tools used: Kali Linux, Firefox, Wireshark

### Solution/walkthrough:

#### Question 1:

Open "pcap1.pcap" in Wireshark. What is the IP address that initiates an ICMP/ping? - 10.11.3.2



#### Question 2:

If we only wanted to see HTTP GET requests in our "pcap1.pcap" file, what filter would we use? -

http.request.method == GET

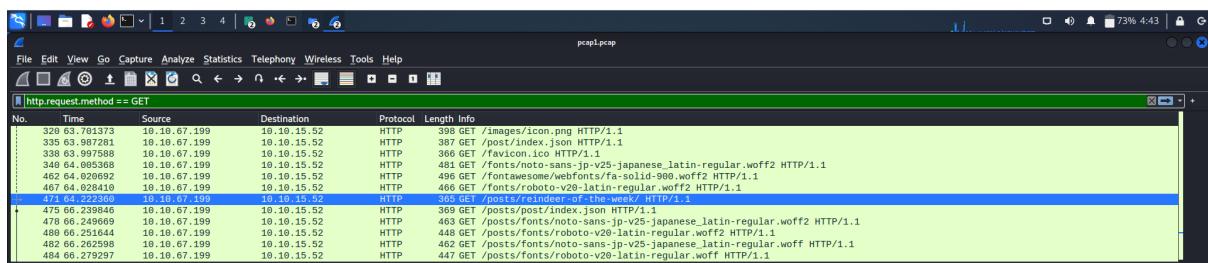
Example

```
ip.src == 192.168.1.1
ip.dst == 192.168.1.1
tcp.port == 22 /
udp.port == 67
http.request.method == GET / POST
```

the == operator to define what

#### Question 3:

Now apply this filter to "pcap1.pcap" in Wireshark, what is the name of the article that the IP address "10.10.67.199" visited? - reindeer-of-the-week



#### Question 4:

Let's begin analysing "pcap2.pcap". Look at the captured FTP traffic; what password was leaked during the login process? - plaintext\_password\_fiasco

No.	Time	Source	Destination	Protocol	Length	Info
94		10.10.73.252	10.10.122.128	FTP	72	Request: QUIT
99		10.10.122.128	10.10.73.252	FTP	80	Response: 221 Goodbye.
94		10.10.122.128	10.10.73.252	FTP	104	Response: 220 Welcome to the TBFC FTP Server!.
25		10.10.73.252	10.10.122.128	FTP	83	Request: USER elfmskidy
80		10.10.122.128	10.10.73.252	FTP	100	Response: 331 Please specify the password.
963		10.10.73.252	10.10.122.128	FTP	98	Request: PASS plaintext_password_fiasco
293		10.10.122.128	10.10.73.252	FTP	88	Response: 530 Login incorrect.
723		10.10.73.252	10.10.122.128	FTP	72	Request: SYST
761		10.10.122.128	10.10.73.252	FTP	104	Response: 538 Please login with USER and PASS.
887		10.10.73.252	10.10.122.128	FTP	72	Request: QUIT
175		10.10.122.128	10.10.73.252	FTP	80	Response: 221 Goodbye.
115		10.10.122.128	10.10.73.252	FTP	104	Response: 220 Welcome to the TBFC FTP Server!

> Frame 6: 72 bytes on wire (576 bits), 72 bytes captured (576 bits)  
> Ethernet II, Src: 02:c3:be:b5:2e:b7 (02:c3:be:b5:2e:b7), Dst: 02:c0:56:51:8a:51 (02:c0:56:51:8a:51)  
> Internet Protocol Version 4, Src: 10.10.73.252, Dst: 10.10.122.128  
> Transmission Control Protocol, Src Port: 45332, Dst Port: 21, Seq: 1, Ack: 1, Len: 6

#### Question 5:

Continuing with our analysis of "pcap2.pcap", what is the name of the protocol that is encrypted? - SSH

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	156	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
4	0.101317	10.11.3.2	10.10.122.128	TCP	54	57748 - 22 [ACK] Seq=1 Ack=145 Win=1024
5	1.127866	10.10.122.128	91.189.92.46	TCP	74	33409 - 443 [SYN] Seq=0 Win=62727 Len=0
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=1
9	2.555520	10.10.73.252	10.10.122.128	TCP	66	45332 - 21 [ACK] Seq=7 Ack=15 Win=491
10	2.555529	10.10.73.252	10.10.122.128	TCP	66	45332 - 21 [FIN, ACK] Seq=7 Ack=16 Win=1
11	2.555534	10.10.122.128	10.10.73.252	TCP	66	21 - 45332 [ACK] Seq=16 Ack=8 Win=490
12	3.175873	10.10.122.128	91.189.92.46	TCP	74	33408 - 443 ISYNL Seq=0 Win=62727 Len=0

> 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: 23, Dst Port: 57249, Seq: 1, Ack: 1, Len: 48

#### Question 6:

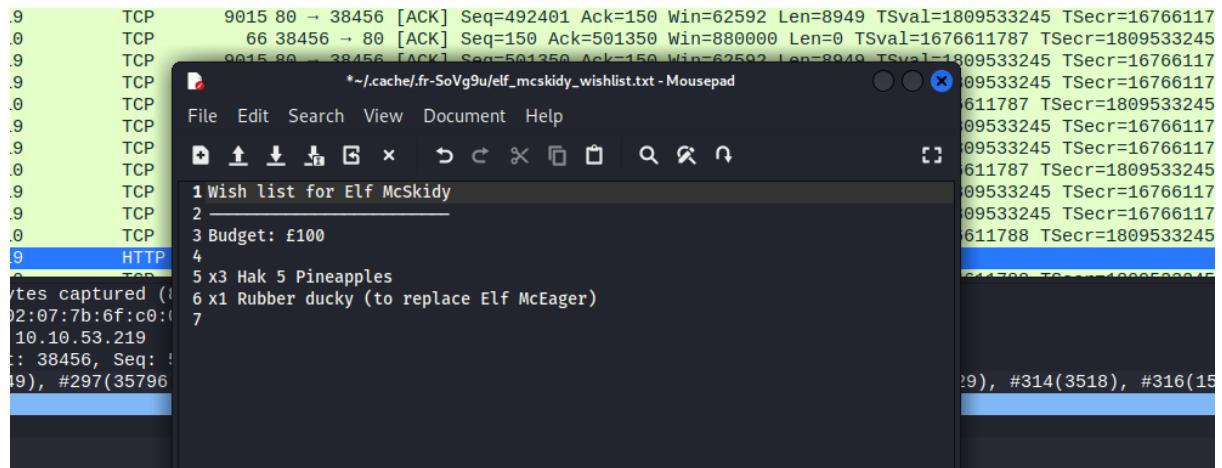
Examine the ARP communications. Who has 10.10.122.128? Tell 10.10.10.1. Answer: 10.10.122.128 is at - 02:c0:56:51:8a:51

No.	Time	Source	Destination	Protocol	Length	Info
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	19.785824	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
77	26.727854	02:c0:56:51:8a:51	02:c8:85:b5:5a:aa	ARP	42	Who has 10.10.0.1? Tell 10.10.122.128
78	26.727968	02:c8:85:b5:5a:aa	02:c0:56:51:8a:51	ARP	56	10.10.0.1 is at 02:c8:85:b5:5a:aa
84	32.388846	02:c8:85:b5:5a:aa	Broadcast	ARP	56	Who has 10.10.122.128? Tell 10.10.0.1
85	32.388861	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
137	53.095851	02:c0:56:51:8a:51	02:c8:85:b5:5a:aa	ARP	42	Who has 10.10.0.1? Tell 10.10.122.128
138	53.095990	02:c8:85:b5:5a:aa	02:c0:56:51:8a:51	ARP	56	10.10.0.1 is at 02:c8:85:b5:5a:aa

> Frame 46: 56 bytes on wire (448 bits), 56 bytes captured (448 bits)  
> Ethernet II, Src: 02:c8:85:b5:5a:aa (02:c8:85:b5:5a:aa), Dst: 02:c0:56:51:8a:51 (02:c0:56:51:8a:51)  
> Address Resolution Protocol (request)

### Question 7:

Analyse "pcap3.pcap" and recover Christmas! What is on Elf McSkidy's wishlist that will be used to replace Elf McEager? - rubber ducky



### Question 8:

Who is the author of Operation Artic Storm? - Kris Kringle

STRICTLY CONFIDENTIAL

Author: Kris Kringle

Revision Number: v2.5

Date of Revision: 14/11/2020

### **Thought Process/ Methodology:**

After having downloaded the file from THM, we can open **pcap1.pcap** file in Wireshark and find the IP address that initiates an ICMP/ping. We use **http.request.method == GET** filter to find the article that the IP address visited. We use the same process for **pcap2.pcap** file to find the password that was leaked. This filter is useful to find specific protocol types that we need to find such as SSH and ARP rather than spending a lot of time going through the list.

## Day 8 : Networking - What's Under The Christmas Tree?

**Tools used:** Kali Linux, Firefox

**Solution/walkthrough:**

Question 1:

When was Snort created? - 1998

# 1998

Snort is a free and open source network intrusion prevention system (NIPS) and network intrusion detection system (NIDS) created by Martin Roesch in **1998**.

Question 2:

Using Nmap on MACHINE\_IP , what are the port numbers of the three services running? -

80,2222,3389

```
Nmap scan report for 10.10.149.237
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))
|_http-title: TBC'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 847.88 seconds
```

Correct Answer

Completed

Completed

Completed

Question 3:

Use Nmap to determine the name of the Linux distribution that is running, what is reported as the most likely distribution to be running? - Ubuntu

```
80/tcp    open  http        Apache httpd 2.4.29 ((Ubuntu))
|_http-title: TBC'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:
```

Completed

Question 4:

What is the version of Apache? - 2.4.29

PORT	STATE	SERVICE	VERSION
80/tcp	open	http	Apache httpd 2.4.29 ((Ubuntu))

Question 5:

What is running on port 2222? - SSH

```
2222/tcp open  ssh
```

Question 6:

Use Nmap's Network Scripting Engine (NSE) to retrieve the "HTTP-TITLE" of the webserver. Based on the value returned, what do we think this website might be used for? - blog

```
|_http-title: TBFC's Internal Blog
```

**Thought Process/ Methodology:**

Using **nmap** on our machine's IP, we can see all the information we need such as ports and the distribution that is running.

### Day 9 : Anyone can be Santa - Prelude:

Tools used : Kali Linux, FireFox, Attackbox

#### **Solution/Walkthrough:**

##### Question 1 :

What are the directories you found on the FTP site? - backups , elf\_workshops, human\_resources, public

```
4096 Nov 16 2020 backups
4096 Nov 16 2020 elf_workshops
4096 Nov 16 2020 human_resources
4096 Nov 16 2020 public
```

##### Question 2 :

Name the directory on the FTP server that has data accessible by the "anonymous" user - public

```
4096 Nov 16 2020 public
```

##### Question 3 :

What script gets executed within this directory - backup.sh

```
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
```

##### Question 4:

What movie did Santa have on his Christmas shopping list? -The Polar Express Movie

```
root@ip-10-10-18-129:~# cat shoppinglist.txt
The Polar Express Movie
```

##### Question 5:

Re-upload this script to contain malicious data (just like we did in section 9.6. Output the contents of /root/flag.txt! -

```
cat /root/flag.txt
THM{even_you_can_be_santa}
```

### **Thought Process/ Methodology:**

We only use the attackbox to start this machine and then click on the terminal then put the 'ftp [IP Addresses given]' in the terminal. After that click enter until it says connected. Put 'anonymous' at the name until it says login successful. You can use the `help` command to list some of the commands you can run whilst connected to the FTP Server. Look at the directories available to use `ls`. We'll navigate to this using `cd` to change our working directory and then `ls` to list the contents. Then use `get` to get the file from the server onto our device. With the file downloaded, open it on our device using a terminal text editor such as nano. Next, replacing the IP\_ADDRESS with your TryHackMe IP, this address is displayed on the navigation bar on the access page. Set up a `netcat` listener to catch the connection on our AttackBox: `nc -lvp 4444`. Now attempt to upload our malicious script to the folder that we have write permissions on the FTP server by returning to our FTP prompt and using `put` to put the file into that directory. Return to our `netcat` listener, after waiting one minute, you should see an output like below and we're done.

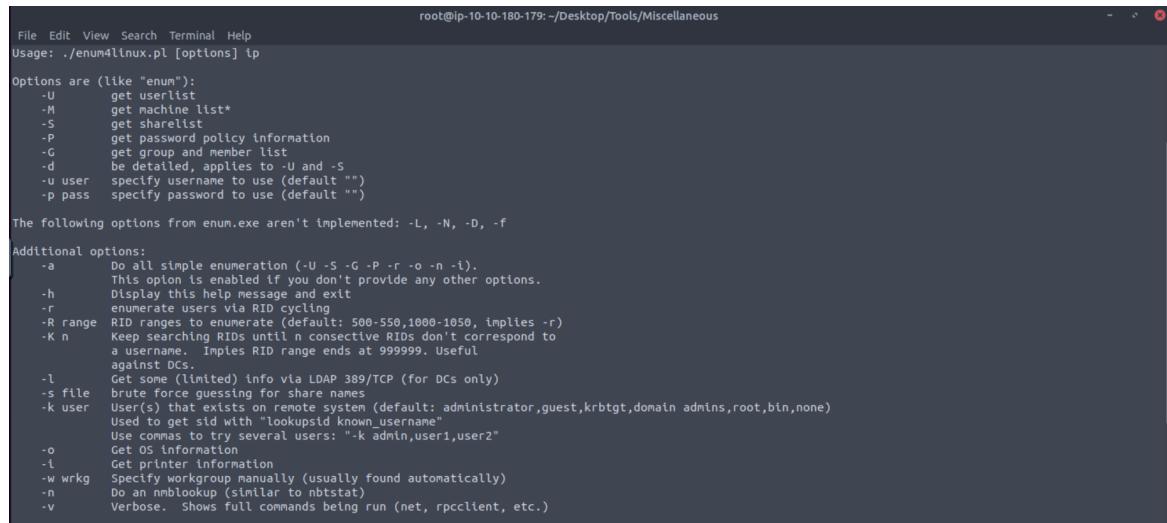
## Day 10 : Networking -Don't be sElfish!

Tools used : THM Attackbox

### Solution/Walkthrough:

#### Question 1

Examine the help options for enum4linux. Match the following flags with the descriptions.



```
root@ip-10-10-180-179:~/Desktop/Tools/Miscellaneous
File Edit View Search Terminal Help
Usage: ./enum4linux.pl [options] [p

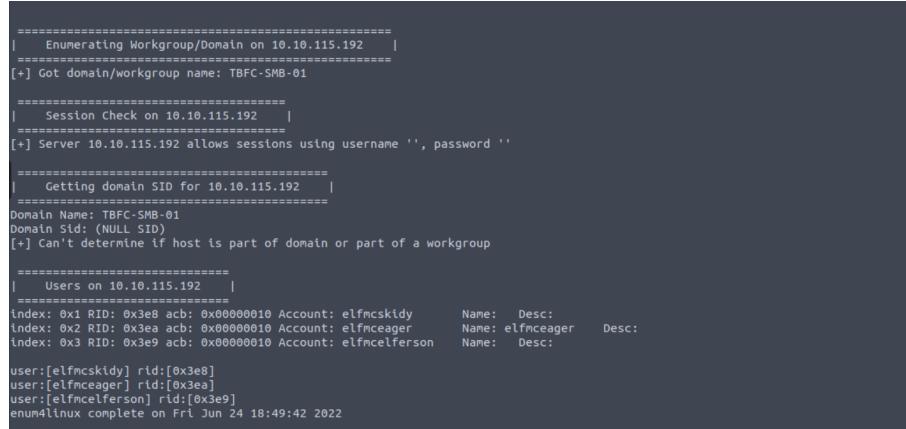
Options are (like "enum"):
-U      get userlist
-M      get machine list*
-S      get sharelist
-P      get password policy information
-G      get group and member list
-D      be detailed, applies to -U and -S
-u user  specify username to use (default "")
-p pass   specify password to use (default "")

The following options from enum.exe aren't implemented: -L, -N, -D, -f

Additional options:
-a      Do all simple enumeration (-U -S -G -P -r -o -n -i).
-h      Display this help message and exit
-r      enumerate users via RID cycling
-R range RID ranges to enumerate (default: 500-550,1000-1050, implies -r)
-K n    Keep searching RIDs until n consecutive RIDs don't correspond to
       a username. Implies RID range ends at 999999. Useful
       against DCs.
-l      Get some (limited) info via LDAP 389/TCP (for DCs only)
-s file brute force guessing for share names
-k user User(s) that exists on remote system (default: administrator,guest,krbtgt,domain admins,root,bin,none)
-o      Get OS information
-i      Get printer information
-w wrkg Specify workgroup manually (usually found automatically)
-n      Do an nblookup (similar to nbtstat)
-v      Verbose. Shows full commands being run (net, rpcclient, etc.)
```

#### Question 2

Using enum4linux, how many users are there on the Samba server? -3



```
=====
|   Enumerating Workgroup/Domain on 10.10.115.192  |
=====
[+] Got domain/workgroup name: TBFC-SMB-01
=====
|   Session Check on 10.10.115.192  |
=====
[+] Server 10.10.115.192 allows sessions using username '', password ''
=====
|   Getting domain SID for 10.10.115.192  |
=====
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 10.10.115.192  |
=====
Index: 0x1 RID: 0x3e8 acb: 0x00000010 Account: elfmcskidy      Name:  Desc:
Index: 0x2 RID: 0x3ea acb: 0x00000010 Account: elfmceager     Name:  Desc:
Index: 0x3 RID: 0x3e9 acb: 0x00000010 Account: elfmcelferson  Name:  Desc:
user:[elfmcskidy] rId:[0x3e8]
user:[elfmceager] rId:[0x3ea]
user:[elfmcelferson] rId:[0x3e9]
enum4linux complete on Fri Jun 24 18:49:42 2022
```

### Question 3

Now how many "shares" are there on the Samba server? -4

```
=====
| Share Enumeration on 10.10.115.192 |
=====
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
```

### Question 4

Use smbclient to try to login to the shares on the Samba server. What share doesn't require a password? -tbfc-santa

```
root@l-p-10-10-180-179:~/Desktop/Tools/Miscellaneous# smbclient //10.10.115.192/tbfc-santa
WARNING: The "syslog" option is deprecated
Enter WORKGROUP\root's password:
Try "help" to get a list of possible commands.
smb: \> █
```

### Question 5

Log in to this share, what directory did ElfMcSkidy leave for Santa? - jingle-tunes

```
█ note_from_mcskidy.txt x
1 Hi Santa, I decided to put all of your favourite jingles onto this
share - allowing you access to it from anywhere you like! Regards -
ElfMcSkidy

smb: \> ls
.
D      0 Thu Nov 12 02:12:07 2020
..
D      0 Thu Nov 12 01:32:21 2020
jingle-tunes
D      0 Thu Nov 12 02:10:41 2020
note_from_mcskidy.txt
N      143 Thu Nov 12 02:12:07 2020

10252564 blocks of size 1024. 5369388 blocks available
smb: \> get note_from_mcskidy.txt
getting file `note_from_mcskidy.txt' of size 143 as `note_from_mcskidy.txt' (34.9 KiloBytes/sec) (average 34.9 KiloBytes/sec)
smb: \> get note_mcskidy.txt
NT_STATUS_OBJECT_NAME_NOT_FOUND opening remote file `note_mcskidy.txt'
smb: \> get note_from_mcskidy.txt
getting file `note_from_mcskidy.txt' of size 143 as `note_from_mcskidy.txt' (69.8 KiloBytes/sec) (average 46.5 KiloBytes/sec)
smb: \> cd jingle-tunes
smb: \> jingle-tunes\> ls
.
D      0 Thu Nov 12 02:10:41 2020
..
D      0 Thu Nov 12 02:12:07 2020

10252564 blocks of size 1024. 5369388 blocks available
```

### **Thought Process/ Methodology:**

We're using the THM AattackBox to solve this task. Using the terminal prompt in the attackbox, we navigate to enum4linux. We run the enum4linux and it list all the possible options for us to use. By referring to all the options listed, we use -U to figure out total of users and -S to list out all the "shares" on Samba server. Then, we use smbclient tool to access the Samba server and its shares by replacing "sharename" in `smbclient //10.10.119.152/**sharename**` with all the possible sharename listed before using try and error method until we found one that doesn't require a password. After logging in the share, we use the ls command to list files and directories in the share.Using the get command to get the directory ElfMcSkidy leave for Santa.