PSP0201

Week 2

Writeup

Group Name: WakuWaku

Members

ID	Name	Role
1211103115	Azri Syahmi Bin Azhar	Leader
1211103233	Muhammad Amir Adib Bin Mohd Aminuddin	Member
1211103419	Muhammad Afif Jazimin Bin Idris	Member
1211103284	Miteshwara Rao A/L Subramaniam	Member

Day 1: Web Exploitation – A Christmas Crisis

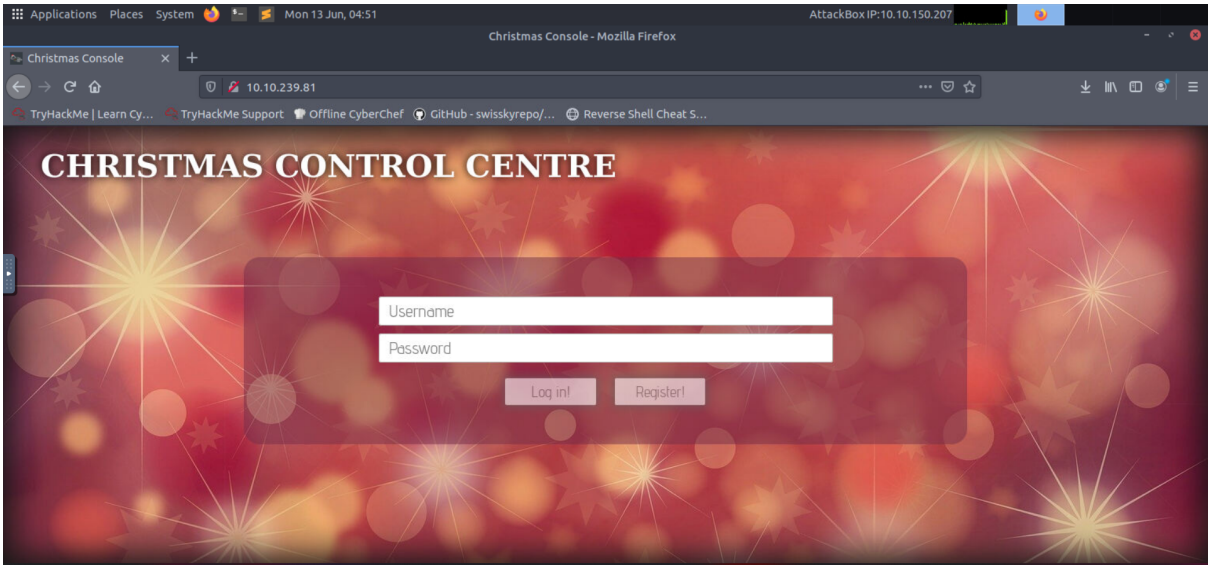
Tools used: Attackbox, Firefox

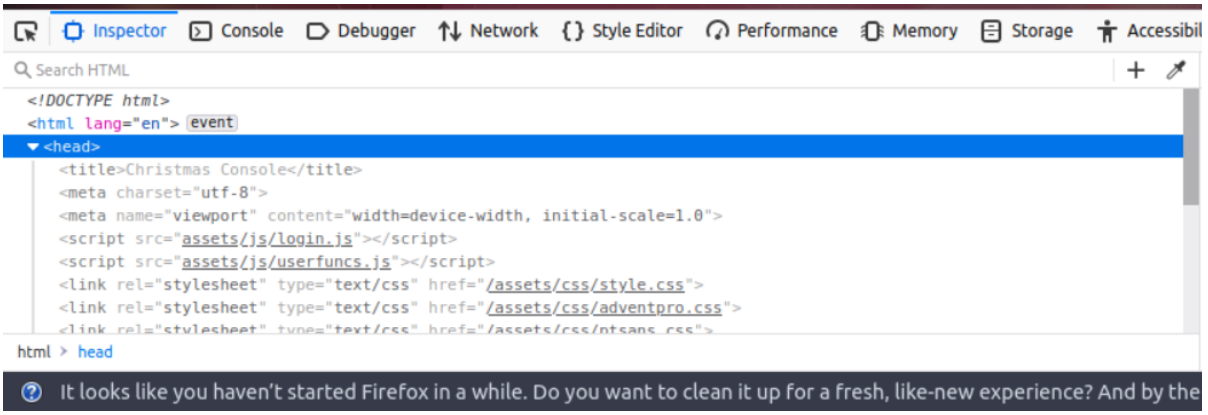
Solution/walkthrough:

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

Answer: Christmas Console

Enter the IP address of the target machine and inspect the website to see the title.

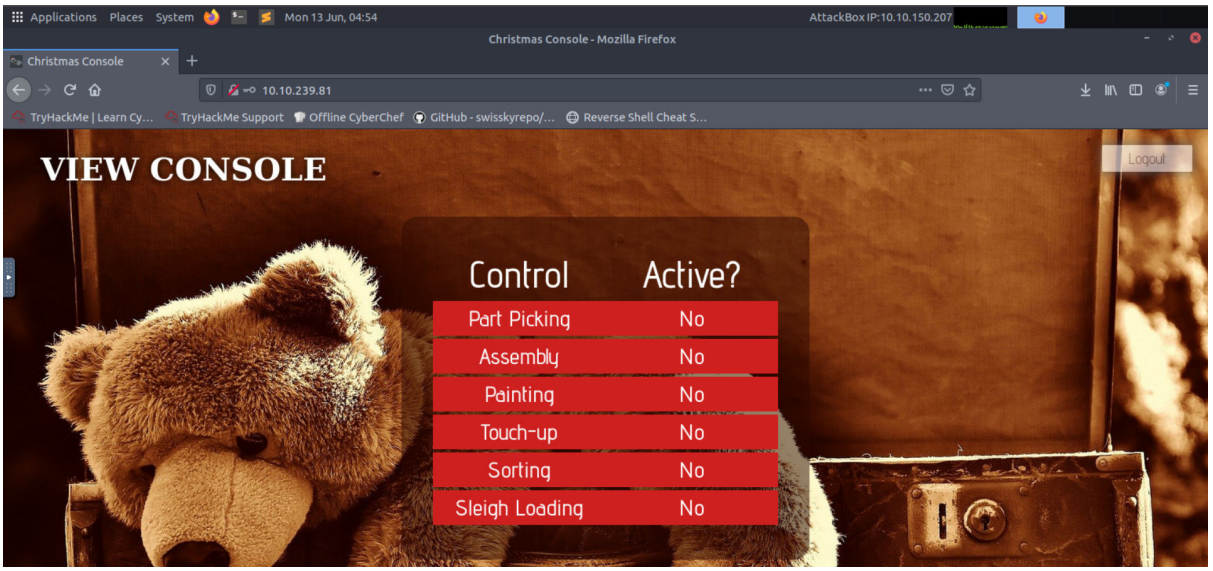




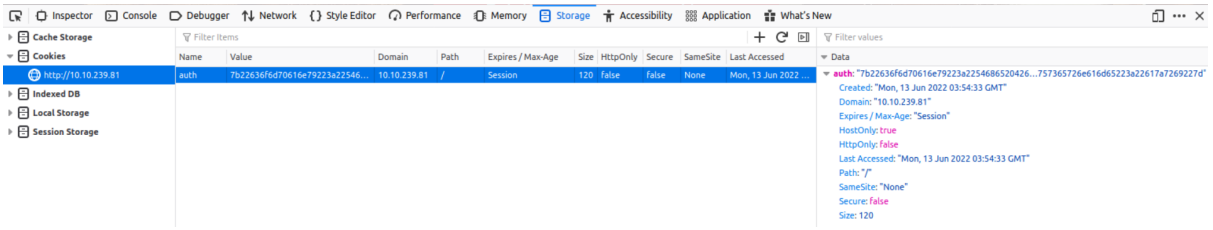
Question 2: What is the name of the cookie used for authentication?

Answer: auth

Create an account and log in to the account. We are taken to the view console where all of Santa's systems are down. Unfortunately, we don't have the authority to turn any of them back on at this time.



Open up the browser developer tools, and navigate to the Storage tab to check on the cookie. The auth cookie is seen.



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

Answer: Hexadecimal

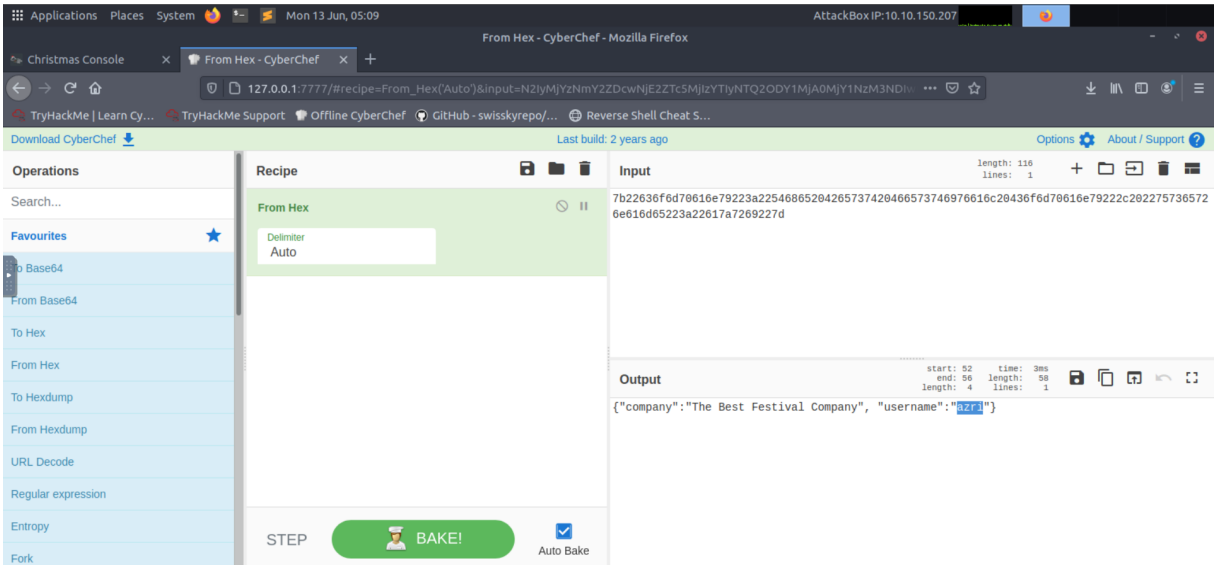
Retrieve the cookie's value. The value consists of a-z and 0-9.

Value
7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e79222c2022757365726e616d65223a2274696d6f746879227d

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

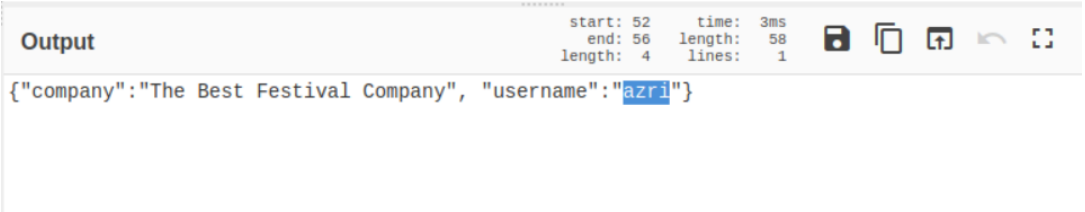
Answer: JSON

Decode the cookie value to a string using Cyberchef. The string consists of curly braces({})



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

Answer: The Best Festival Company



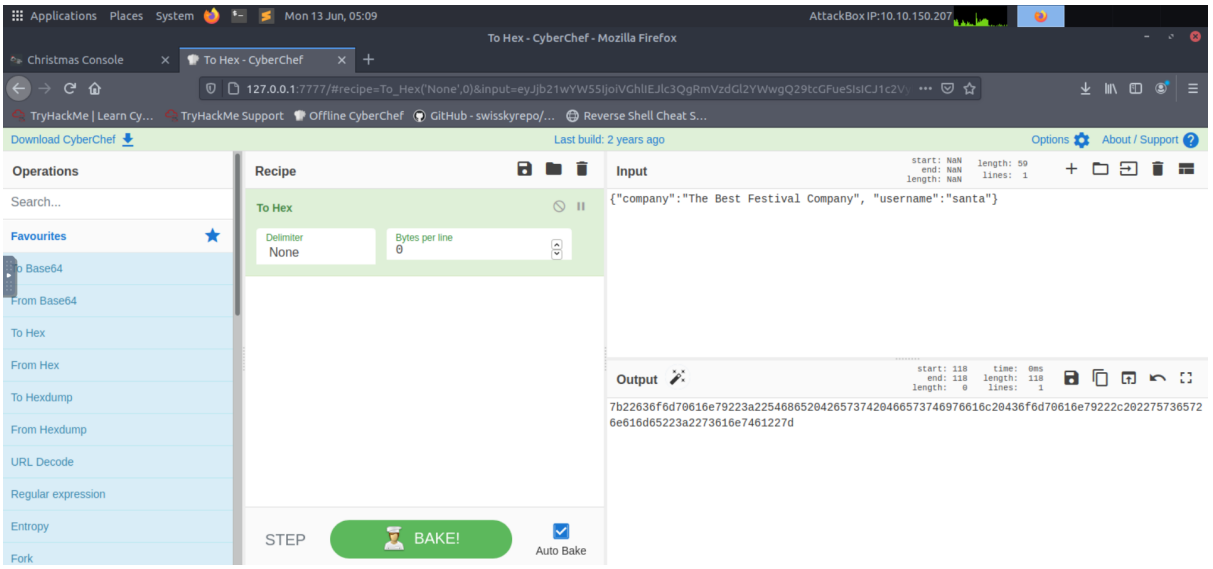
Question 6: What is the other field found in the cookie?

Answer: username

Question 7: What is the value of Santa's cookie?

Answer:
7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e79222c2022757365726e616d65223a2273616e7461227d

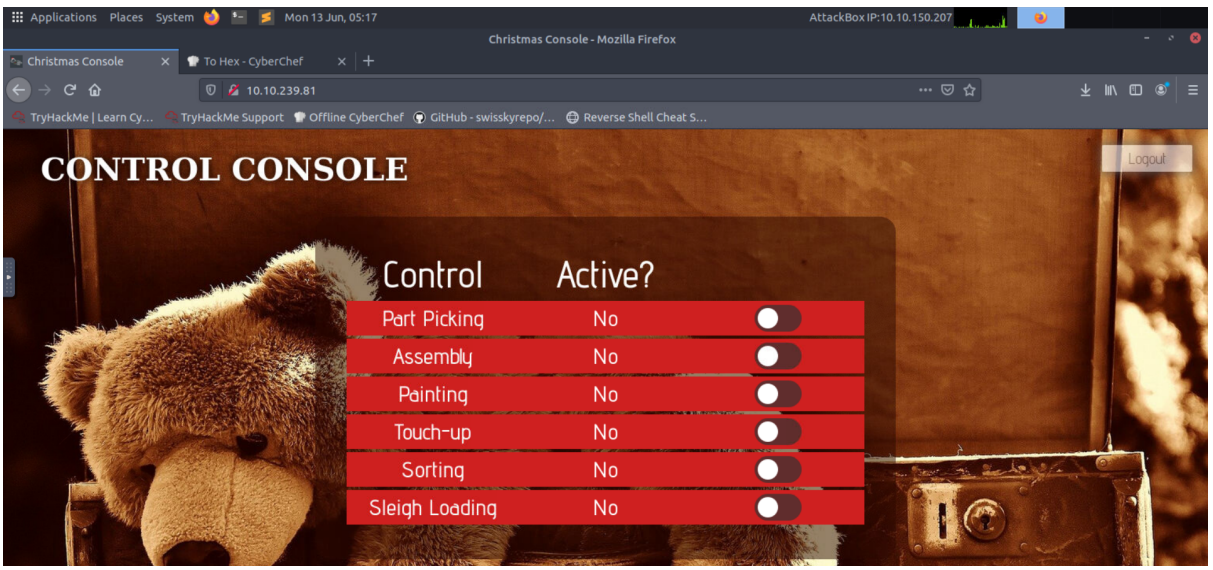
Change the username to ‘santa’, encode back the JSON statement to hex.



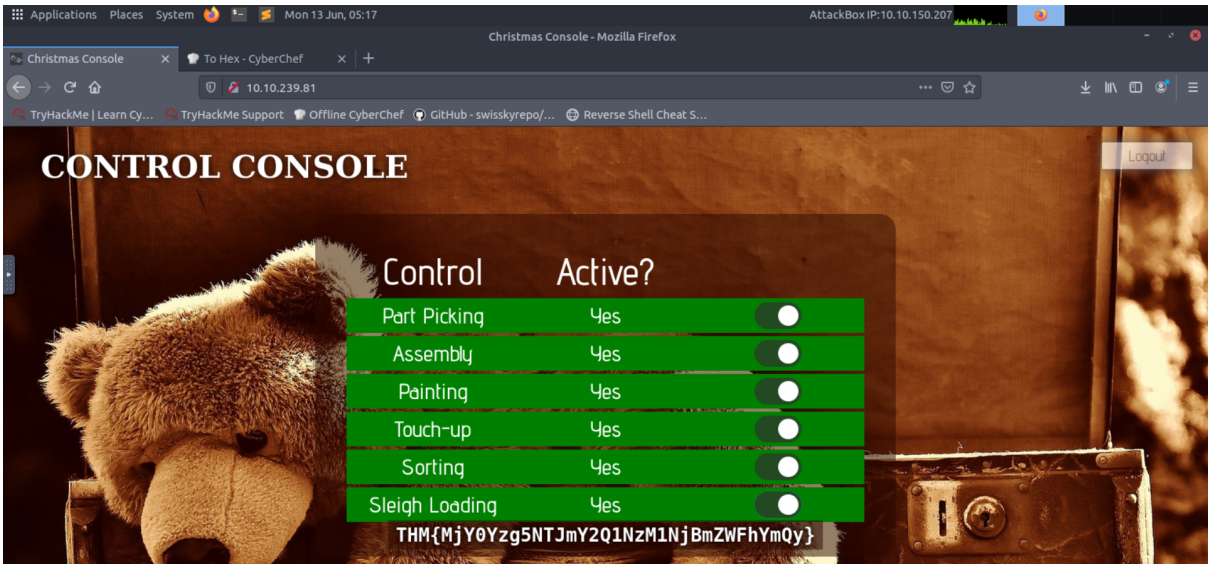
Question 8: What is the flag you're given when the line is fully active?

Answer: THM{MjY0Yzg5NTJmY2Q1NzM1NjBmZWFhYmQy}

Replace the value in the current auth cookie with the hexadecimal value we received and refresh the page. We are now logged in as Santa.



Now having access to the controls, switching on every control shows the flag.



Thought Process/Methodology:

We were directed to a login/registration page after entering the IP address. We then inspect the website to find out the title. We went on to create an account and log in. After logging in, we open the developer tool in our browser and select the Storage tab to view the site cookie. We concluded from the cookie data that it was a hexadecimal value and used Cyberchef to convert it to text. We saw the elements and their values (company: The Best Festival Company, username: azri) The username element was detected in a JSON statement. We changed the administrator account's username to 'santa' and then converted it back to hexadecimal using Cyberchef. We refreshed the page after replacing the cookie value with the converted value. We were now taken to an administrative page (Santa's) where we were able to enable each control, which then displayed the flag.

Day 2: Web Exploitation – The Elf Strikes Back!

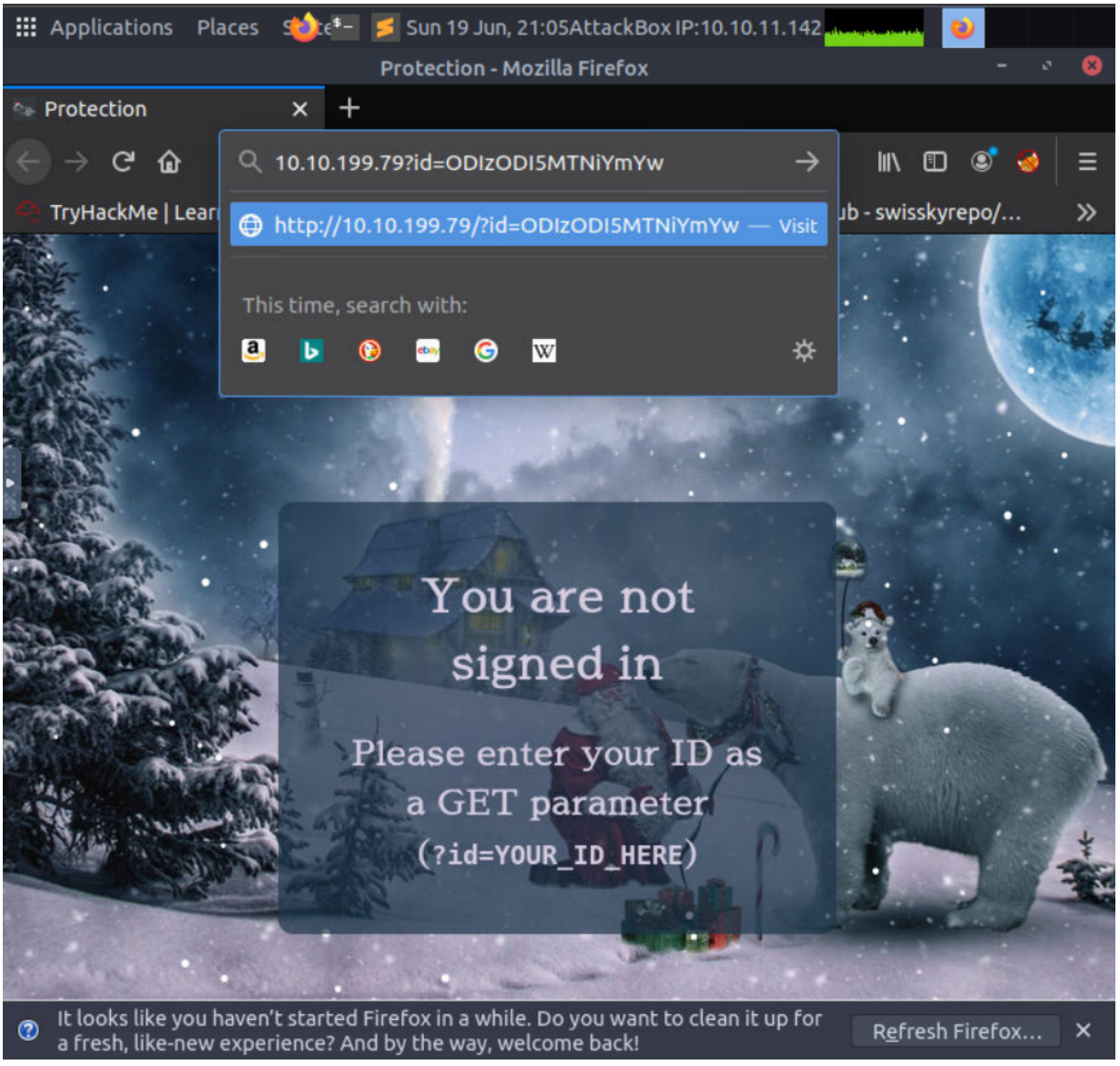
Tools used: Attackbox, Firefox

Solution/walkthrough:

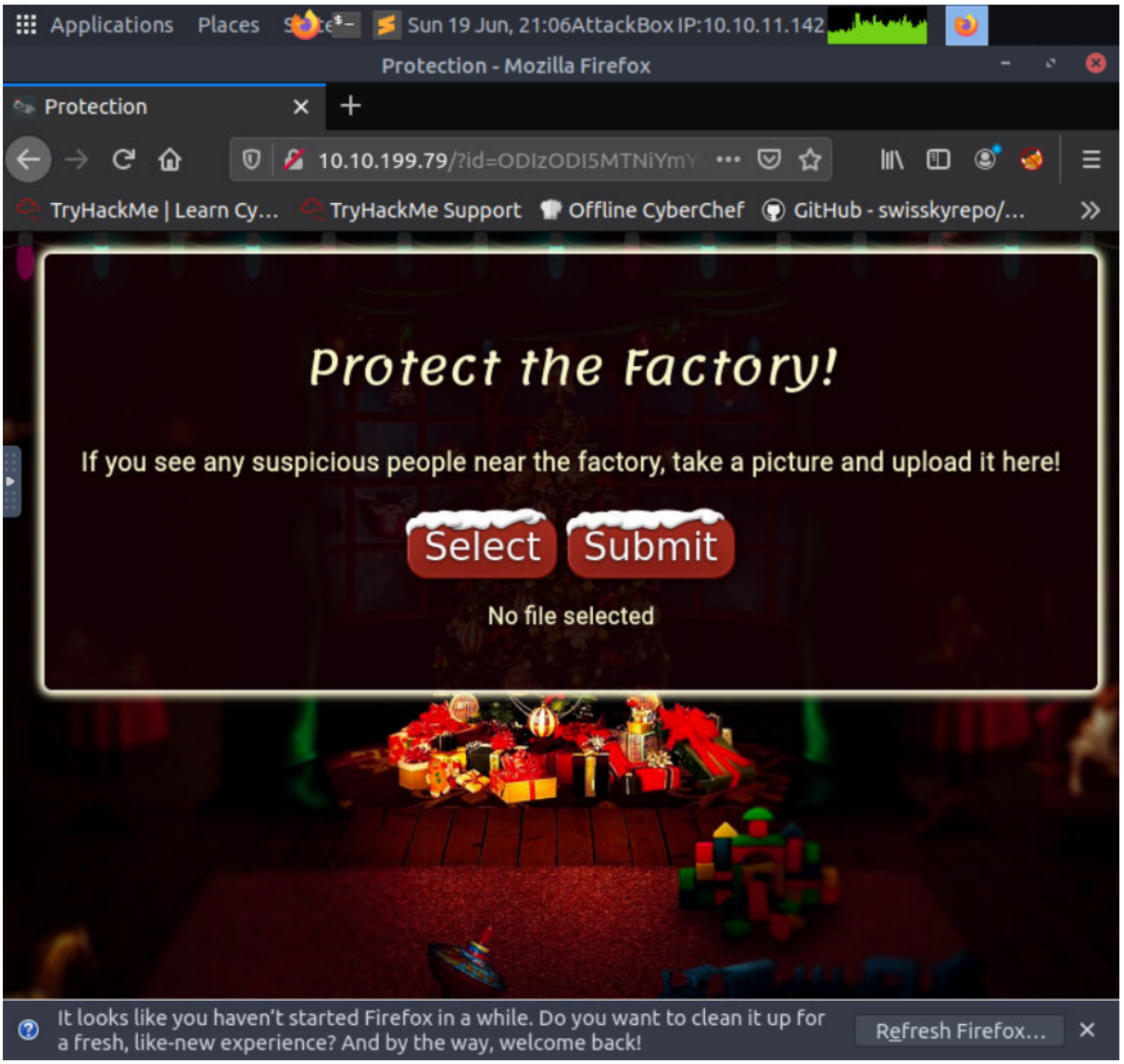
Question 1: What string of text needs adding to the URL to get access to the upload page?

Answer: ?id=ODIzODI5MTNiYmYw

Use a GET parameter at the end of the URL by using 'id' as the parameter name and the given id number (ODIzODI5MTNiYmYw) as the value which makes it as '?id=ODIzODI5MTNiYmYw'.



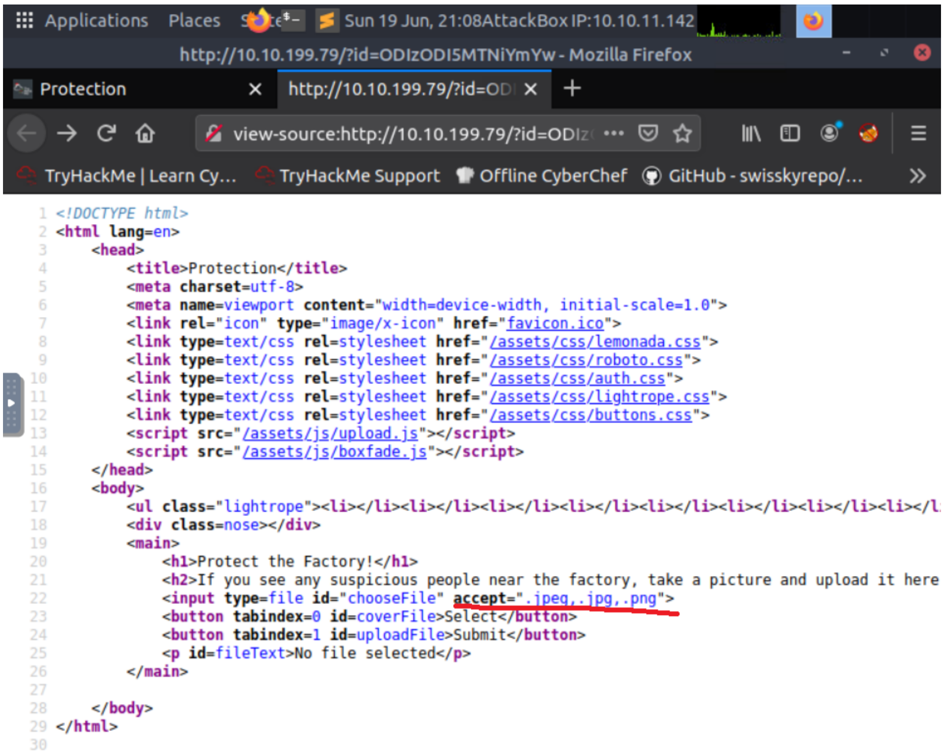
After doing so, it will redirect you to the following page.



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

Answer: Image

Right-click and then click on the view source page. From this, you would be able to see what kind of file format is accepted.

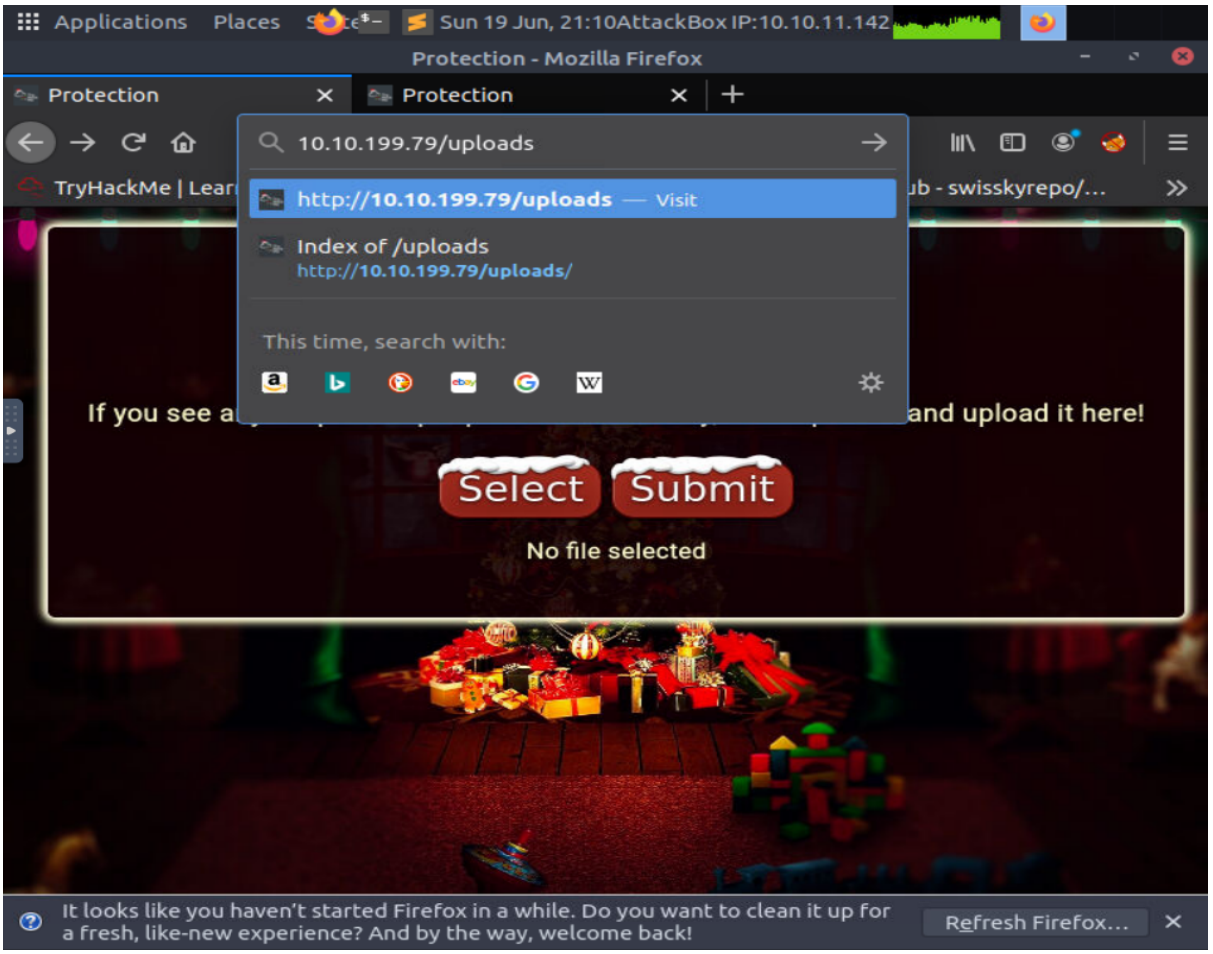



```html
<!DOCTYPE html>
<html lang=en>
    <head>
        <title>Protection</title>
        <meta charset=utf-8>
        <meta name=viewport content="width=device-width, initial-scale=1.0">
        <link rel="icon" type="image/x-icon" href="favicon.ico">
        <link type=text/css rel=stylesheet href="/assets/css/lemonada.css">
        <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></l
        <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
            <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: In which directory are the uploaded files stored?

Answer: /uploads/



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

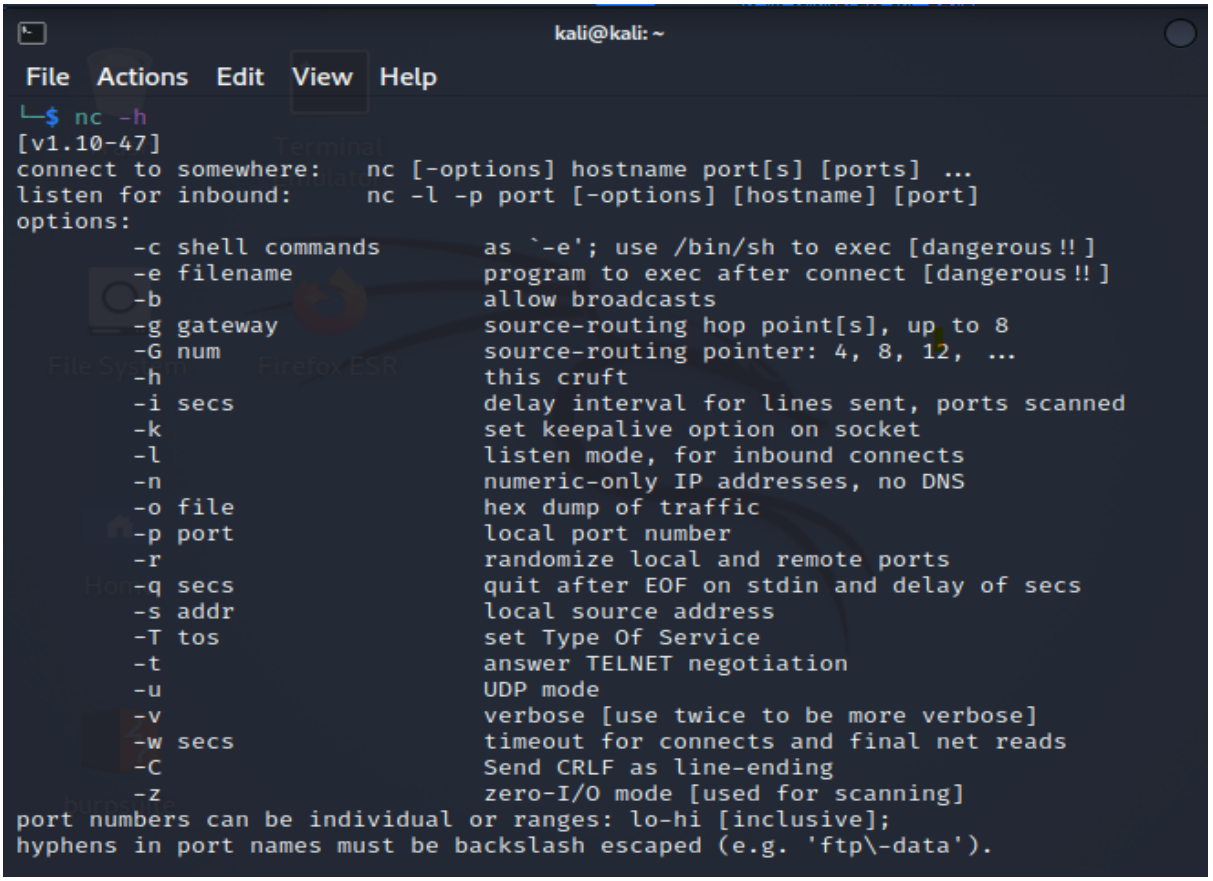
Answer:

l- Used to specify that nc should listen for an incoming connection rather than initiate a connection to a remote host.

v- Have nc give more verbose output.

n- Do not do any DNS or service lookups on any specified addresses, hostnames or ports.

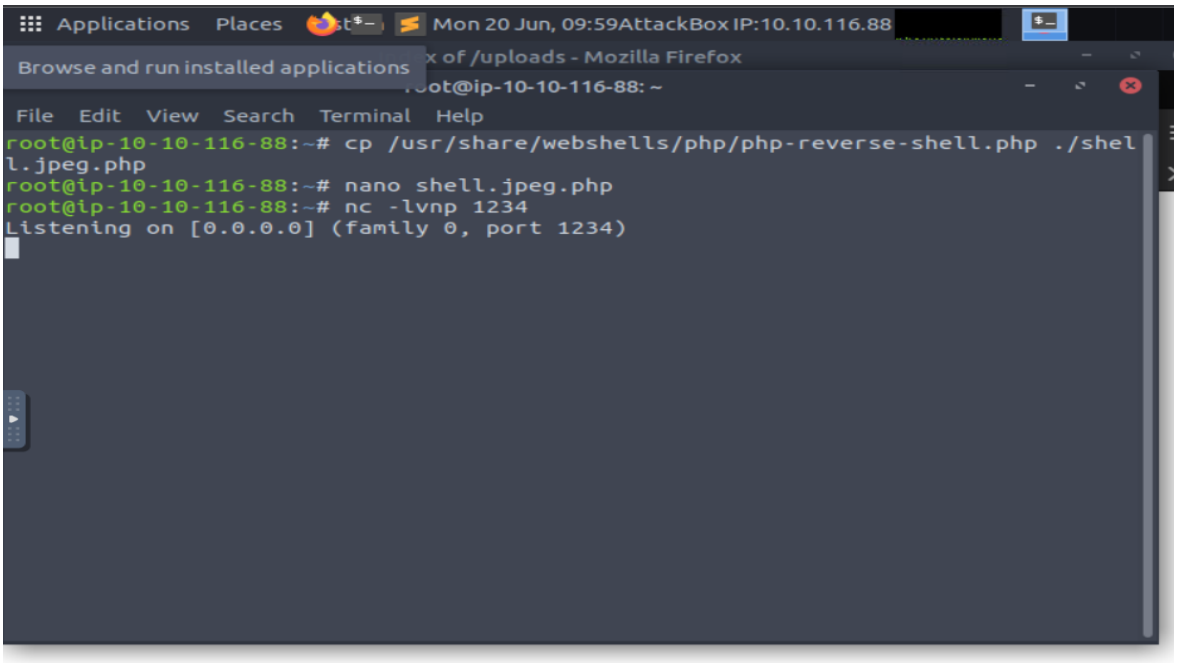
p- Specifies the source port nc should use, subject to privilege restrictions and availability.



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

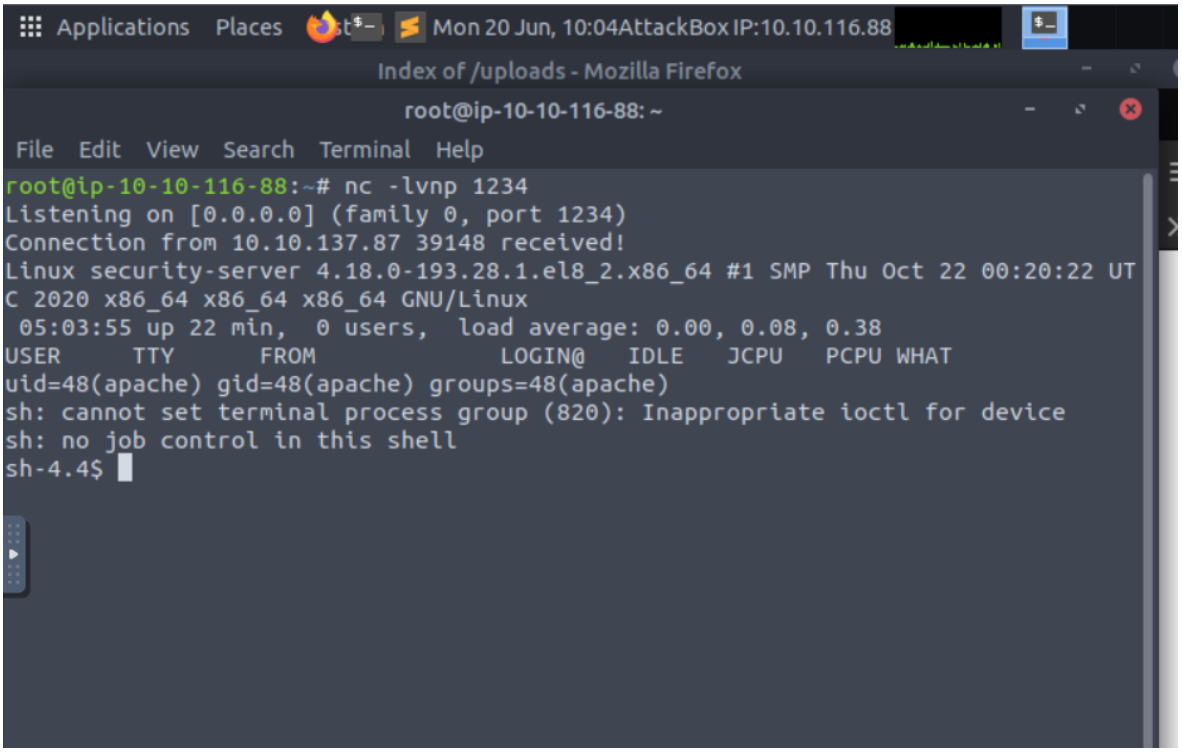
Answer: THM{MGU3Y2UyMGUwNjExYTY4NTAxOWJhMzhh}

Get ready your netcat listener using a common port which in this case I'll be using port 1234, type 'nc -lvnp 1234' in the terminal

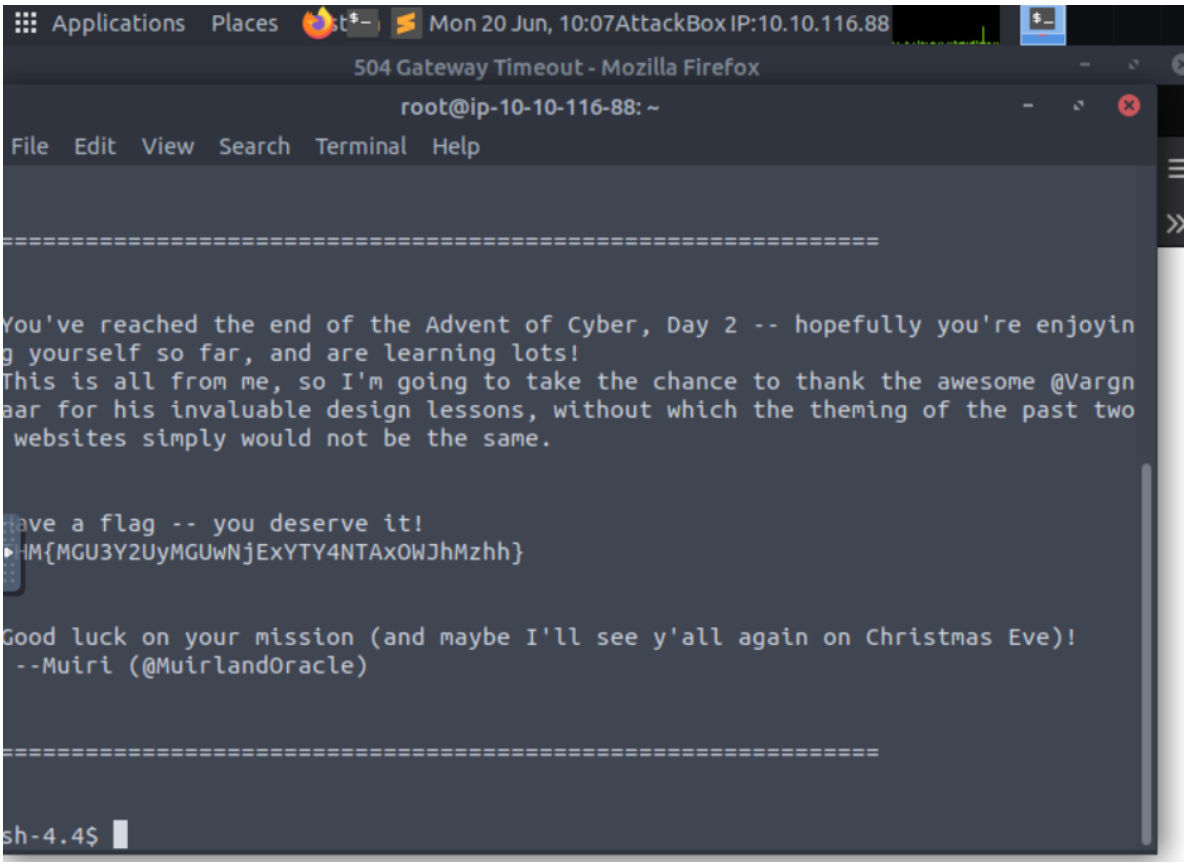


Then click on the 'shell.jpeg.php' in the' /uploads/' page

Then in the terminal, this would appear



Type in the command in the above ‘cat /var/www/flag.txt’ and a flag would appear indicating that it is successful.



Thought Process/Methodology:

First, type in the IP in the URL box in firefox. It will redirect you to a page which shows that you are not signed in. Next, use a GET parameter at the end of the URL by using ‘id’ as the parameter name and the given id number (ODIzODI5MTNiYmYw) as the value which makes it as ‘?id=ODIzODI5MTNiYmYw’. It will redirect you to a page which gives you control to upload files to the website. To make sure what kind of file it accepts, view the page source. Find the php-reverse-shell.php through the given paths then click on it to change the IP and ports to the currently used one. Then, create a shell using the jpeg.php as extensions which makes it as shell.jpeg.php. To do so, simply just type in cp /usr/share/webshells/php/php-reverse-shell.php ./shell.jpeg.php in the terminal Get ready the netcat listener using a common port which in this case I’ll be using port 1234, type ‘nc -lvnp 1234’ in the terminal. Then, submit the shell.jpeg.php file on the website. Next, go to the '/uploads/' page which gives you access to see all the files that have been submitted. From this, we could view our file, and then click it. Go back to the terminal and type in the command ‘cat /var/www/flag.txt’ and a flag would appear indicating that it is successful.

Day 3: Web Exploitation – Christmas Chaos

Tools used: Attackbox, 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 **I**nternet **o**f **T**hings (**IoT**) devices by remotely logging, configuring the device to perform malicious attacks at the control of the attackers; the Mirai botnet infected over 600,000 IoT devices mostly by scanning the internet and using default credentials to gain access.

Question 2: 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):

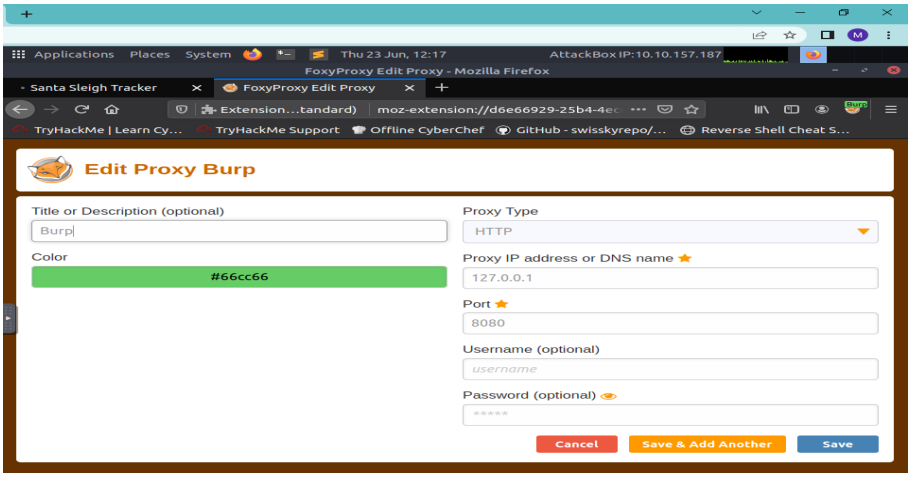
Question 3: Read the report from Hackerone ID:804548 - who was the agent assigned from the Dept of Defense that disclosed the report on Jun 25th?

Answer: ag3nt-j1

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

Answer: 8080

Open safari, direct to FoxyProxy and click options. Then, click on ‘edit’ for burp. It will then show the port number for burp and also the proxy type to be answered for the next question.



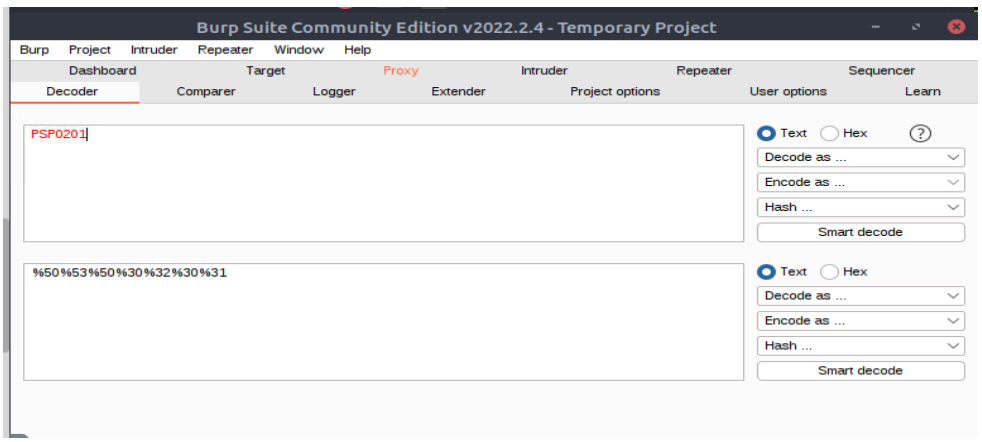
Question 5: Examine the options on FoxyProxy on Burp. What is the proxy type?

Answer: HTTP

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

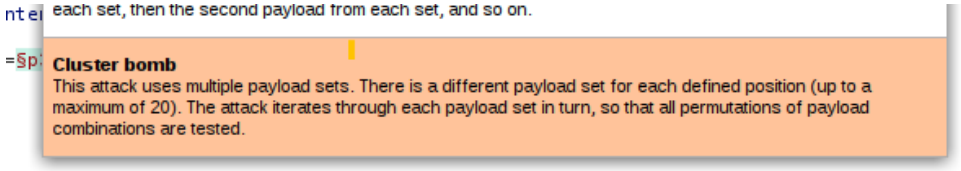
Answer: %50%53%50%30%32%30%31

On BurpSuite, go to the decoder and type “PSP0201” at the space. Then, set it to be encoded as a URL.



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

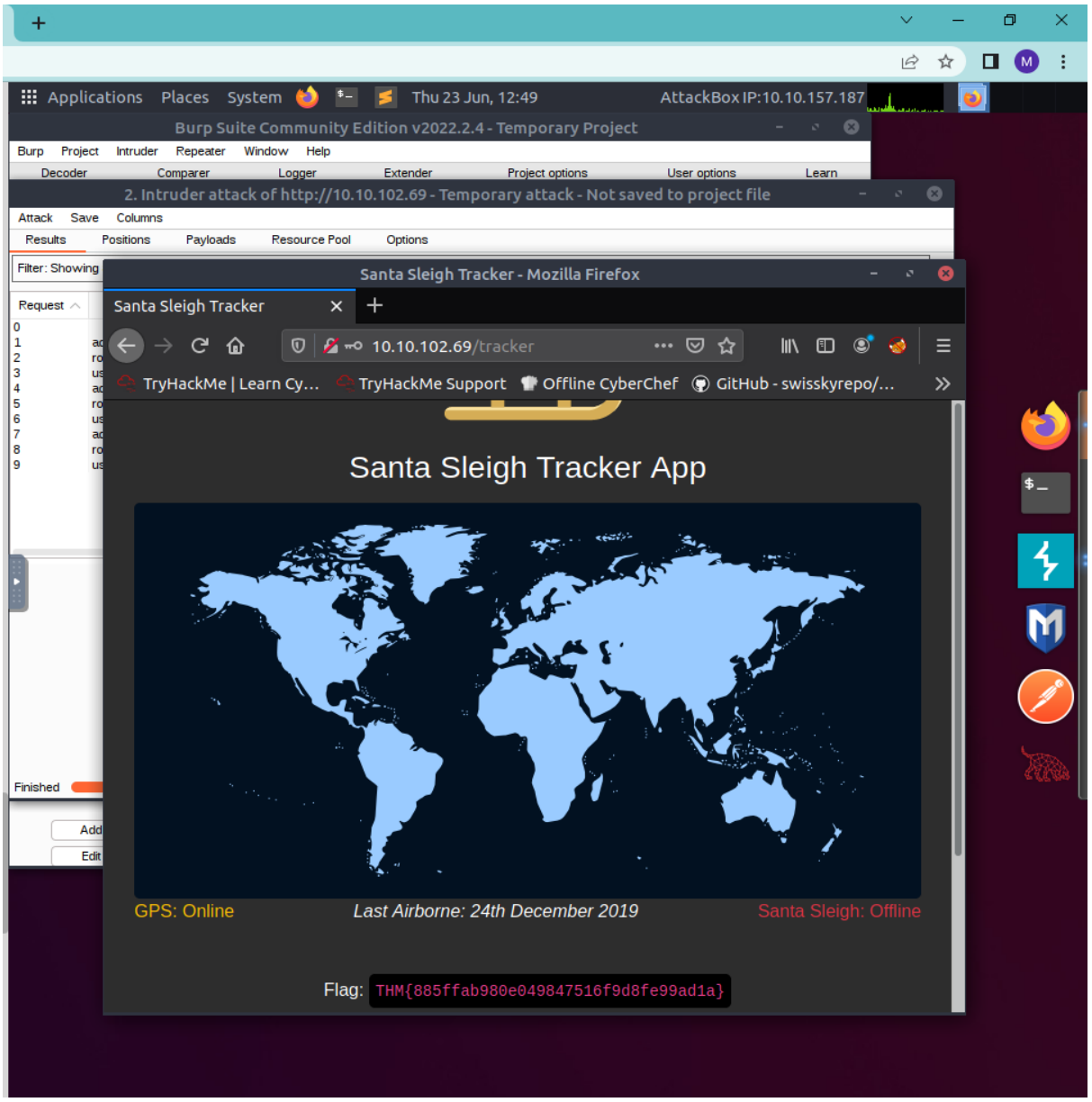
Answer: Cluster bomb



Question 8: What is the flag?

Answer: THM{885ffab980e049847516f9d8fe99ad1a}

Launch the attack on BurpSuite by cluster bomb. On the payload option, insert a sample list of usernames and passwords to be tried. Once we get the correct username and password, turn off burp and login into the website. The flag then will be shown.



Thought Process/Methodology:

We want to get into the website. However, we don't know what is the username and password to log in. Hence, we use BurpSuite to perform dictionary attack. To perform the dictionary attack, we use intruder to loop through and submit a login request using a list of usernames and passwords that are commonly used in hope that the usernames and passwords in the list are correct We also intercept our traffic by proxying through the Burpsuite. Once all the combinations of usernames and passwords are tested, the result will be shown. Typically all incorrect logins will have the same status and length, if a combination is correct it will be different. In the end, we found that the combination that has different status is 'admin' as username and '12345' as password. We have successfully login into the website.

Day 4: Web Exploitation – Santa's watching

Tools used: Attackbox, 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)

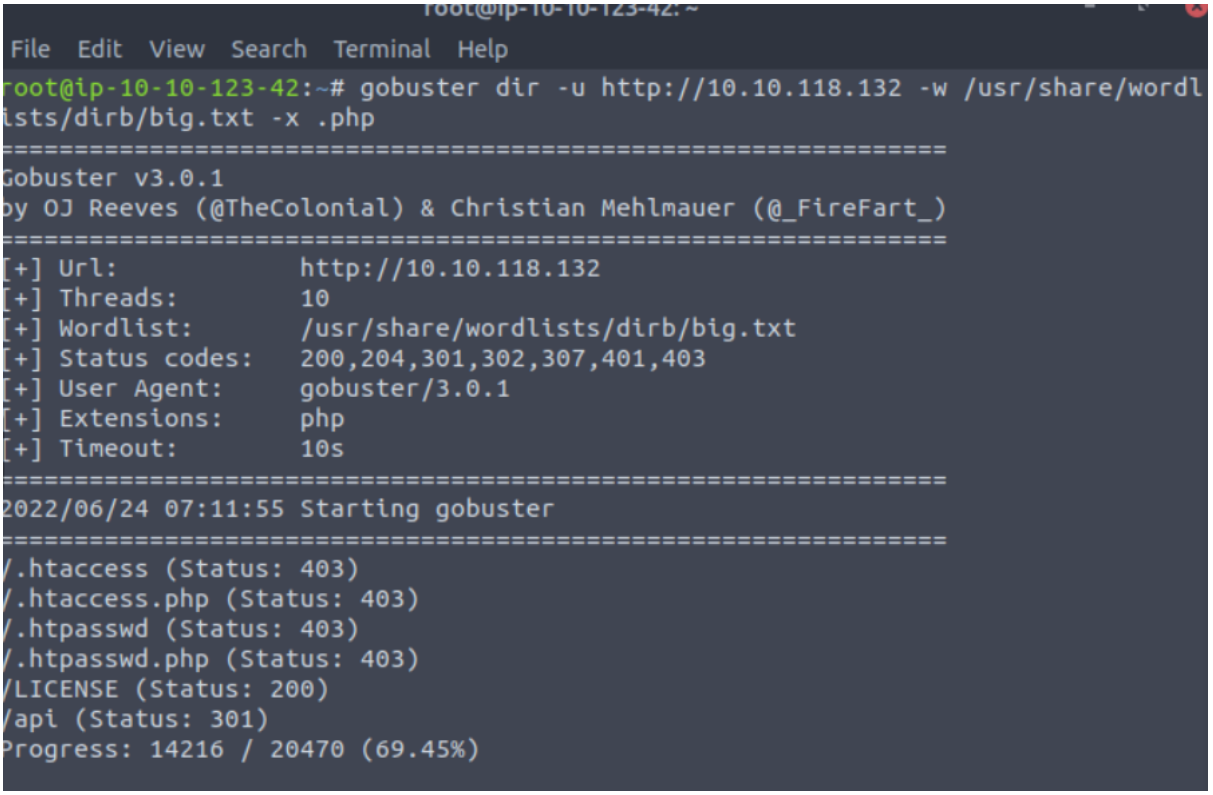
Answer: wfuzz -c -z file,big.txt http://shibes.xyz/api.php?breed=FUZZ

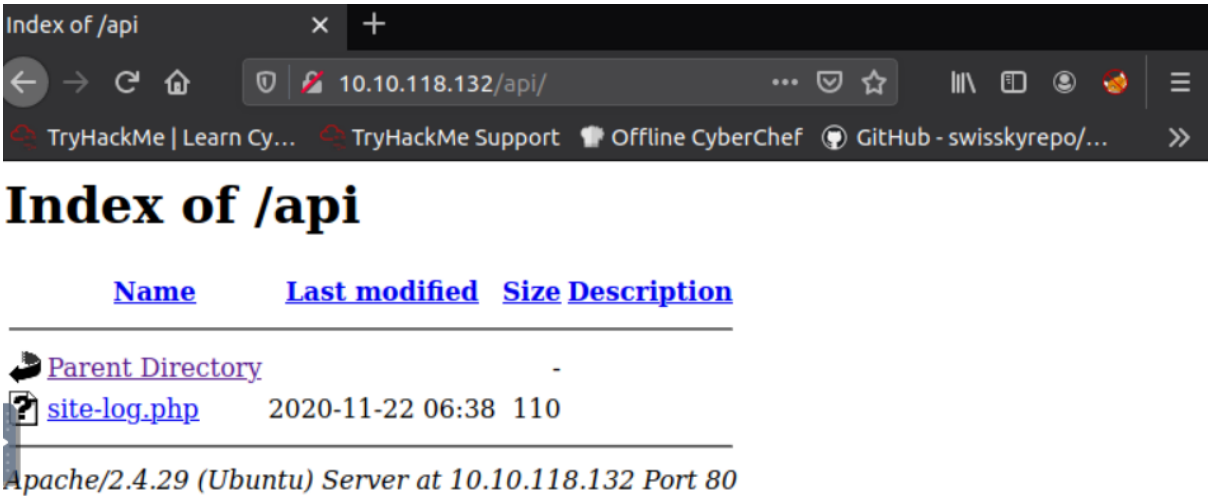
A wfuzz command starts with wfuzz and the -c to show output in colour,-z to specify the file which is big.txt. big.txt is used because that is the file recommended by tryhackme.

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

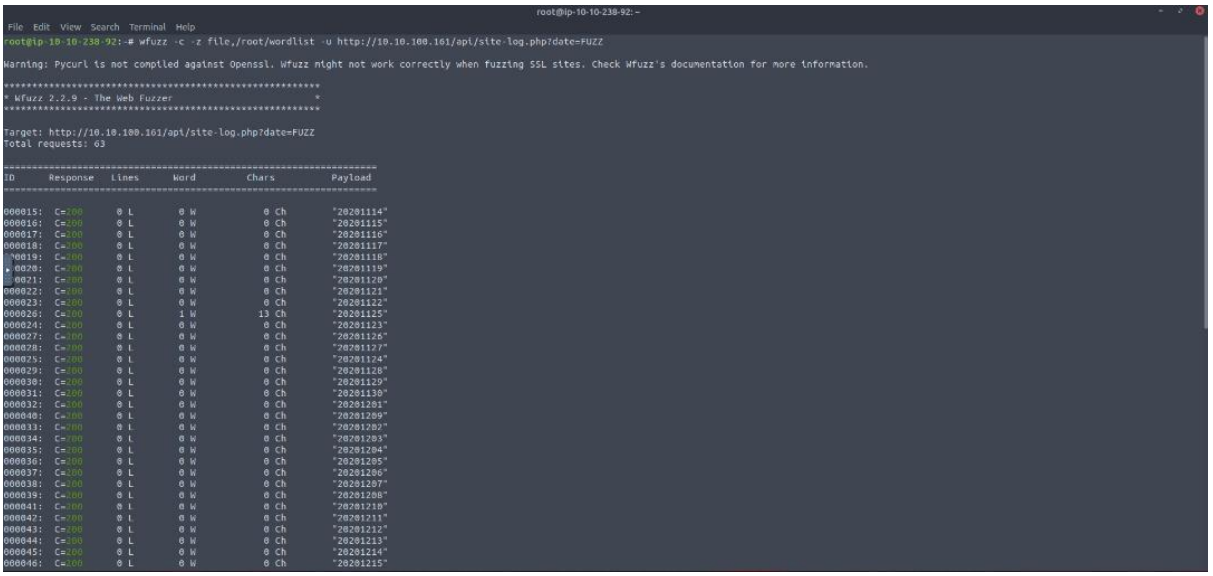
/api command to the ip address shows the file in the API directory

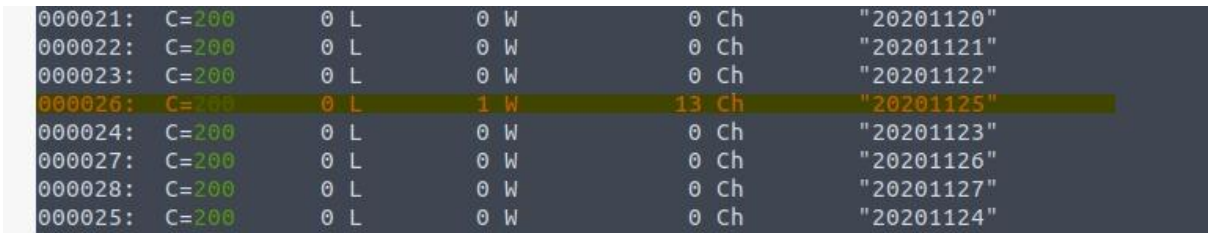


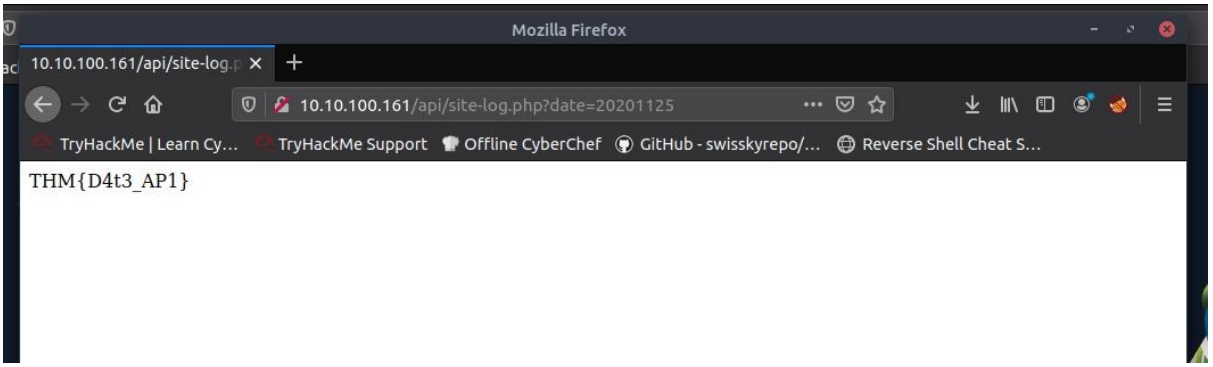


Question 3: Fuzz the date parameter on the file you found in the API directory. What is the flag displayed in the correct post?

Answer: THM{D4T3_AP1}



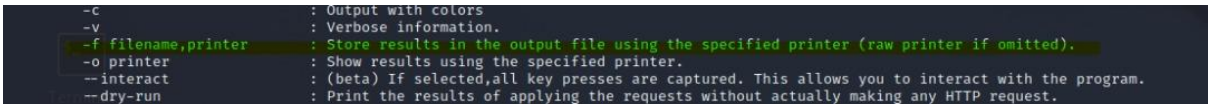




Question 4: Look at wfuzz's help file. What does the -f parameter store results to?

Answer: filename, printer

explanation: based on https://manpages.debian.org/buster/wfuzz/wfuzz.1.en.html -f, stores results in the output file using the specified printer (raw printer if omitted)



Thought Process/Methodology: Firstly we had to figure out a wfuzz command for the website http://shibes.xyz/api.php and were also given the directory. Then we had to use gobuster to find out the directory that had the file we wanted to recover for our login page. Lastly, we fuzz the date parameter on the file we found in the API directory.

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

Tools used: Attackbox, Firefox, BurpSuite

Solution/walkthrough:

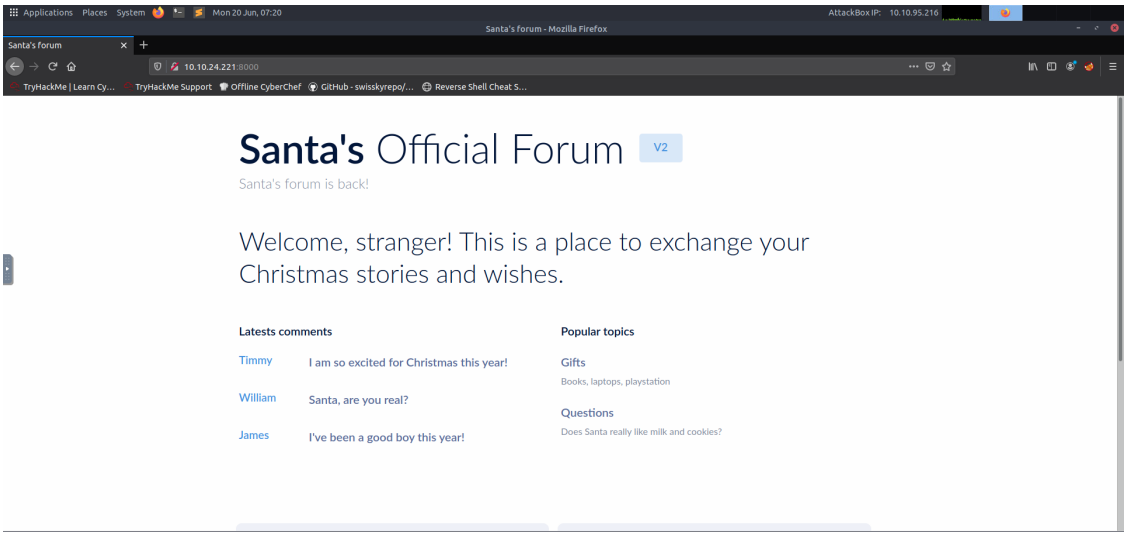
Question 1: What is the default port number for SQL Server running on TCP?

Answer: 1433

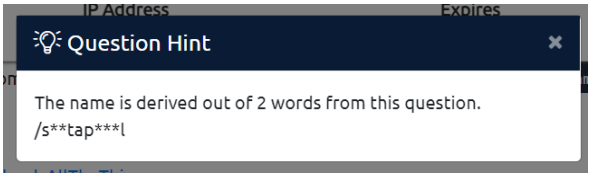
Question 2: Without using directory brute forcing, what's Santa's secret login panel?

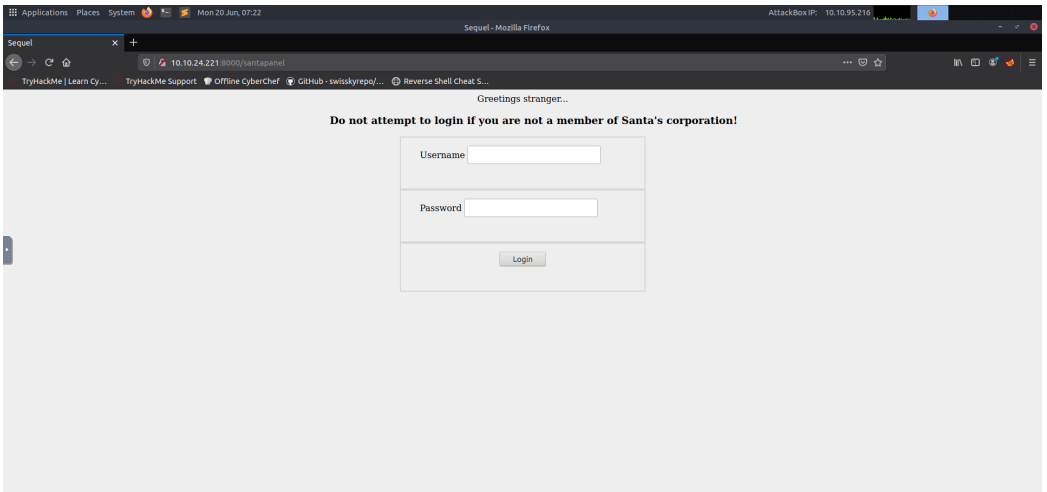
Answer: /santapanel

Visit the Santa's Official Forum. No button that links to Santa's secret login panel.



Guessing the URL using the hint on TryHackMe and successfully getting into Santa's secret login panel





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

Answer: SQLite

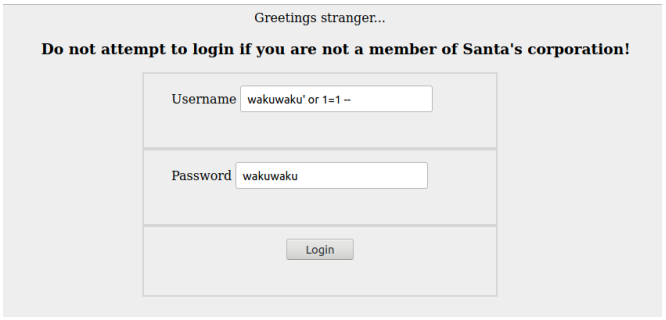
Santa reads some documentation that he wrote when setting up the application, it reads:

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

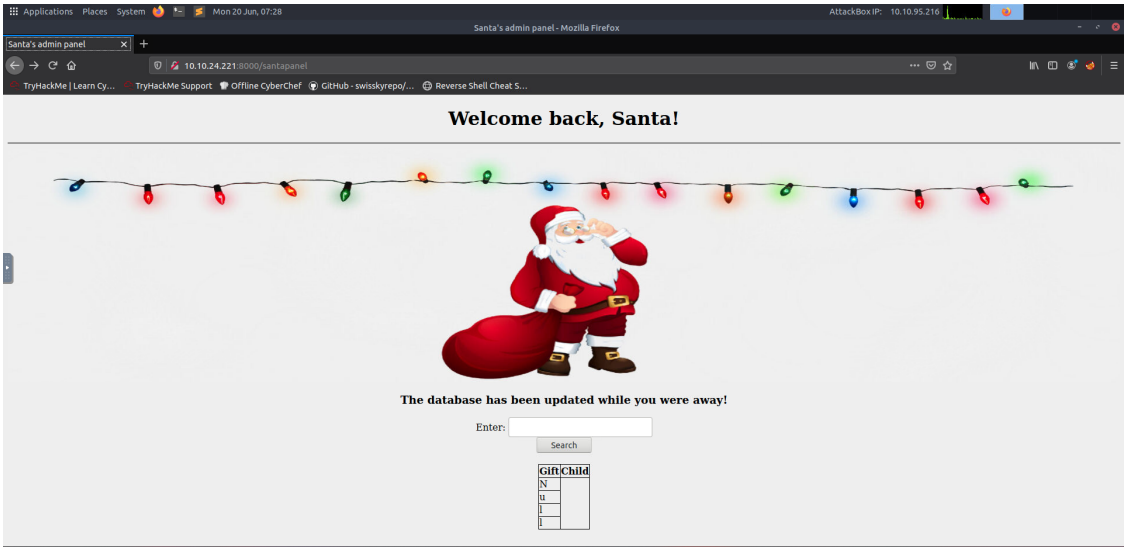
Question 4: How many entries are there in the gift database?

Answer: 22

Bypassing the login page with SQL injection using the payload: `wakuwaku' or 1=1 --`

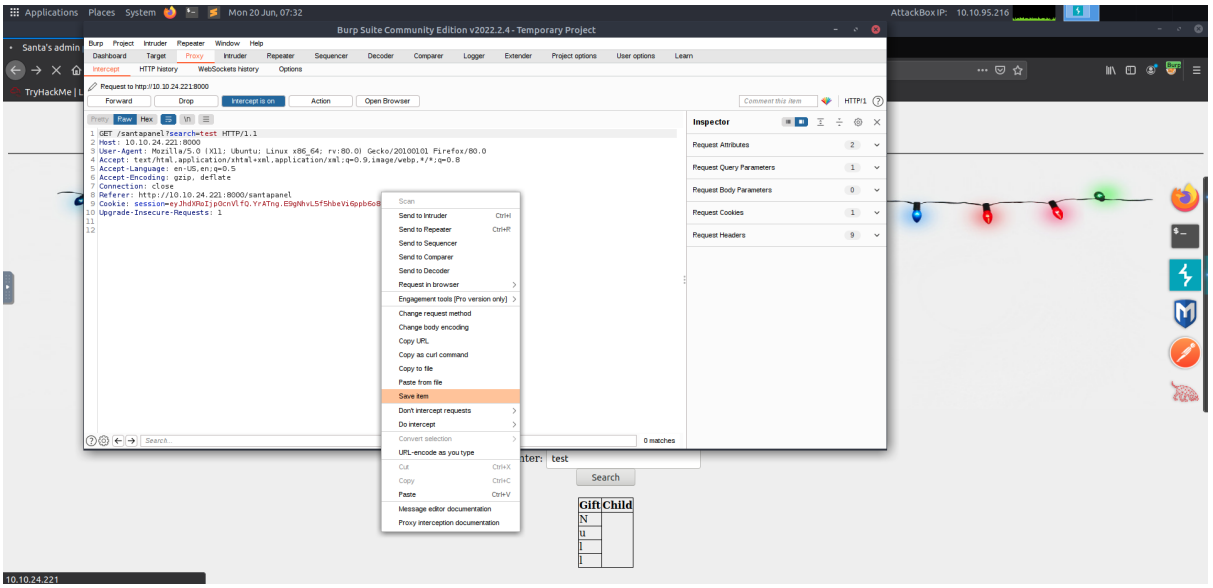


Then, it will redirect us to a new page where we can search the database.



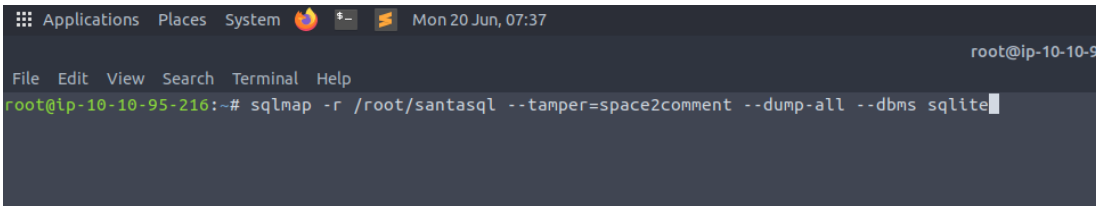
Use SQLMap & BurpSuite to dump all the databases. Enable FoxyProxy in Firefox and make sure intercept is on at the BurpSuite's proxy panel. Do a test request at the webpage.

Save the request at the proxy panel t use it with SQLMap.

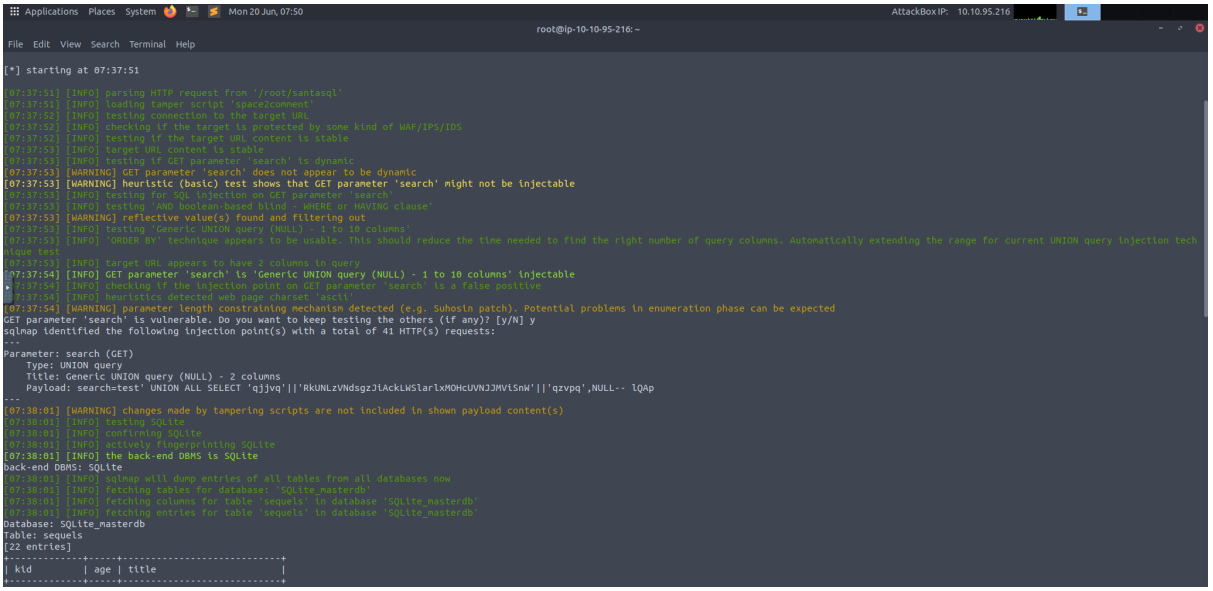


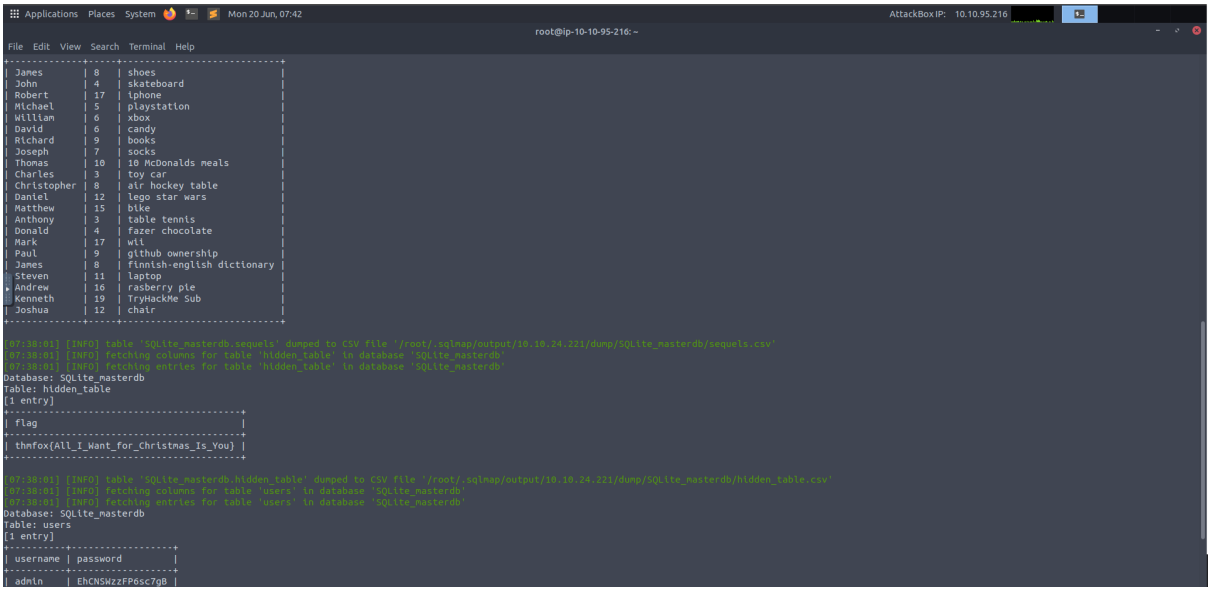
Open terminal to start SQLMap. Use the command given at TryHackMe: `sqlmap -r filename`.

`/root/santasql` is where we saved the request earlier. `--tamper=space2comment` is to bypass the WAF. `--dump-all` is to dump the entire database. `--dbms sqlite` is to tell SQLMap the type of database is SQLite. The following command is what we used:

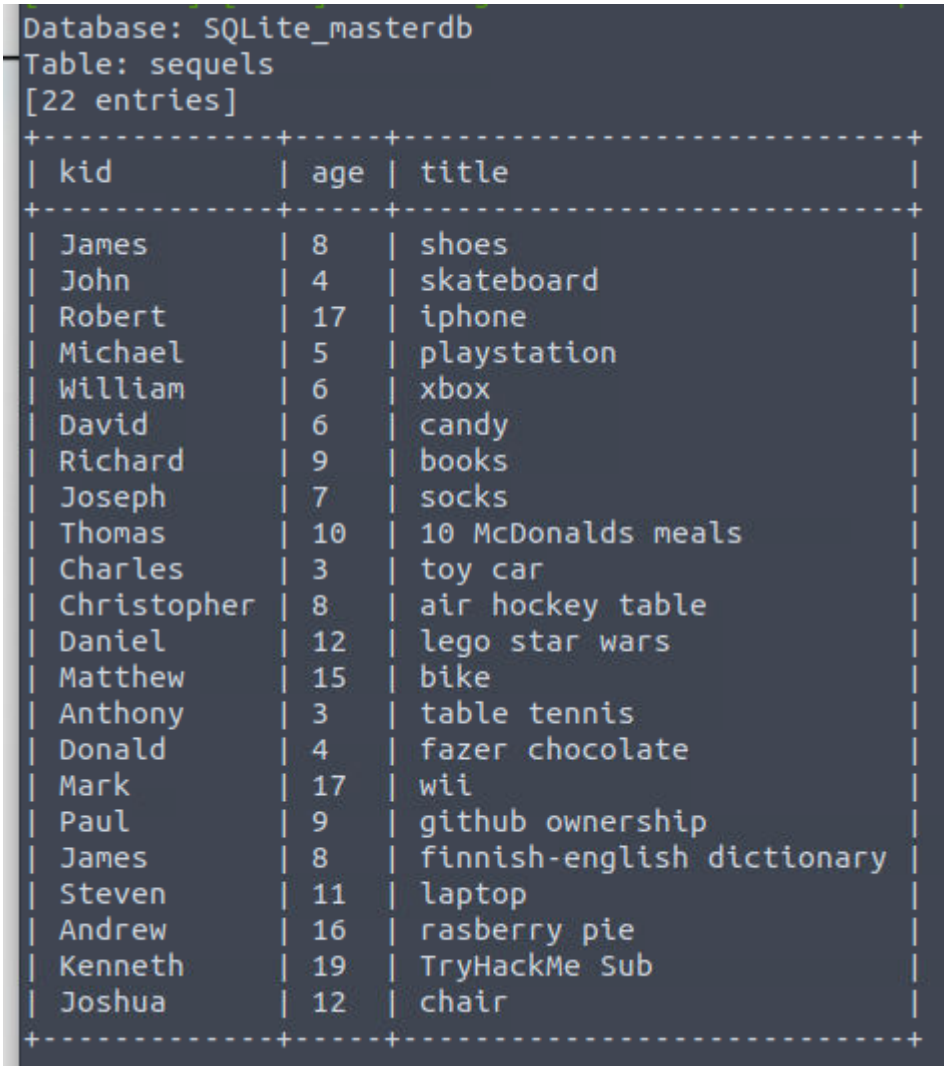


SQLMap then will dump the entire database.





It is shown there are 22 entries. Question 5 and Question 6 will also refer to this database.



Question 5: What is James' age?

Answer: 8

Question 6: What did Paul ask for?

Answer: GitHub ownership

Question 7: What is the flag?

Answer: thmfox{All_I_Want_for_Christmas_Is_You}

```
Database: SQLite_masterdb
Table: hidden_table
[1 entry]
+----------------------------------------+
| flag                                   |
+----------------------------------------+
| thmfox{All_I_Want_for_Christmas_Is_You} |
+----------------------------------------+
```

Question 8: What is admin's password?

Answer: EhCNSWzzFP6sc7gB

```
Database: SQLite_masterdb
Table: users
[1 entry]
+----------+------------------+
| username | password         |
+----------+------------------+
| admin    | EhCNSWzzFP6sc7gB |
+----------+------------------+
```

Thought Process/Methodology:

After we gained access to the target machine, we were shown the Santa's Official Forum page. We then proceeded to visit Santa's secret login page by guessing the URL based on the hint. After that, we bypassed the login page with SQL Injection by inserting a payload `' or 1=1 --`. The `--` in this case has commented out the password checking part. After successfully bypassing it, it redirected us to a new page where we can search for the database. We used SQLMap and BurpSuite to dump the entire database. We did a test request and intercepted it using the FoxyProxy and Burpsuite. The request was saved by right-clicking on it and hitting 'Save Item'. To make it easier to remember, we saved the item with the name santasql in the root folder. We then use SQLMap with this file to output the entire database contents. We use the command `sqlmap -r /root/santasql --tamper=space2comment --dump-all --dbms sqlite`. The terminal then showed an output containing various databases and their contents. The answers for Question 4 - Question 6 were all in the database.