Stanley Onyedika Epuna

Personal Report:
Capture the Flag 4
STANLEY EPUNA: 45731365
23rd June 2020

Table of Contents

Stanley Onyedika Epuna
23/06/2020

DISCLAIMER

Disclaimer This report will highlight the attempts and successes of penetration testing on a CTF system provided to us for educational purposes only. Testing permissions were given by the owner of the system, allowing us to test our web penetration skills. The CTF was run ethically, and only techniques and tools learned in classes and practicals were used.

ABSTRACT

This report outlines the process and execution of implemented penetration testing techniques. Capture the flag events are a form of hacking competition where individuals/teams face off against other teams or a server to capture the most flags among all the participants. CTF's are used in universities to help students practice their system penetration testing skills in a controlled and safe environment where it is ethically acceptable to 'hack' a structure or system. The following is an evaluation and reflection of my final CTF experience, of what went well and what could have been done better; and the overall benefit of using CTF events to further our education on Offensive Security.

Introduction

Capture the Flag (CTF):

These are events used for teaching purposes where students are allowed to practice with their skills on different challenges with the degree of difficulties. When a flag is solved, the flag captured is submitted to the CTF Moby server, resulting in individual points. This task is to capture six (6) flags. You can only participate in this CTF task if you are running a Linux system or macOS, and if you are connected to the Macquarie VPN or inside the campus.

This CTF comprises of System, Web, and Network penetration testing, where the aim is to attack and detect vulnerabilities.

Findings

Testing stage:

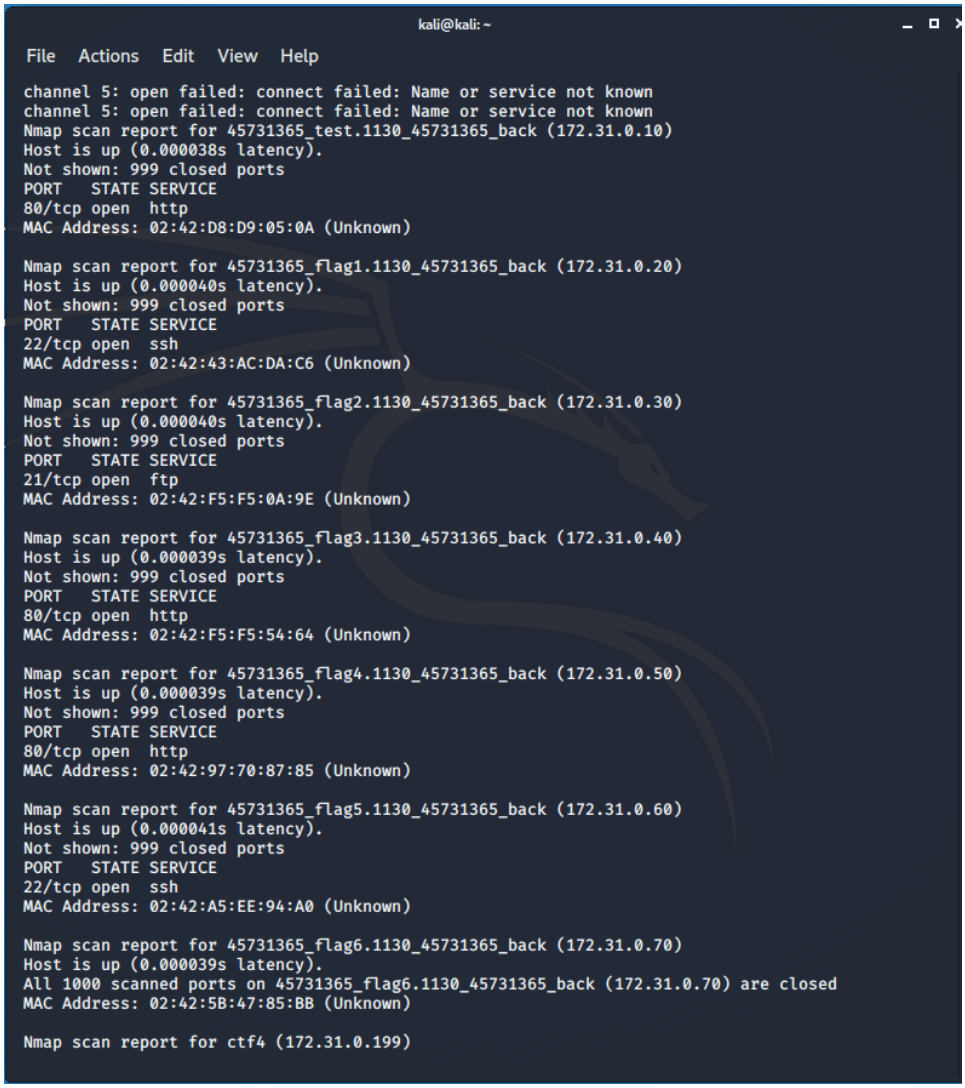
The day before the CTF, I was given the IP address for my CTF4 challenges (10.46.225.199) and also was instructed to use a secure shell(ssh) to access the flags. The command used was ssh -D 8080 45731365@10.46.225.199. This CTF was conducted using Kali Linux and with the following tools, Burp suite, nmap, and gobuster.

CONNECTION NEEDED

1. Macquarie VPN or access to Macquarie University
2. The website where the flags would be submitted: **https://moby.science.mq.edu.au**
3. Kali Linux or macOS
3. The server for the flags: https://172.31.0/24
4. SSH Login information (Ip address, username, password)
5. Burp Suite.
6. Foxy Proxy configuration (SOCKSV5 and HTTP)

The first step taken was to use nmap to scan for clues relating to the flag. The IP address (172.31.0.0), with a subnet mask of /24 was given, so I needed to look at the open ports between 172.31.0.0-172.31.0.254.

nmap 172.31.0.0/24



FLAG 1: Master of Domain

I reached on the web for this hint “Master of Domain”, I found out this has to do with taking control of the domain. Also, I checked the nmap scan and I figured out it was running an ssh service, meaning I could access it through ssh. Domain means a group of devices on a network that can be accessed or administered using a set of rules. To solve this flag, I had to use ssh to the flag1 ip address, then type the command “**domainname**”.

ssh 172.31.0.20
password: lumpysugar82

```
45731365@ctf4:~$ ssh 172.31.0.20
The authenticity of host '172.31.0.20 (172.31.0.20)' can't be established.
ECDSA key fingerprint is SHA256:544g2ImzcfsfYSSU03H/1LclkxzLASKtfmSg2v405pY.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '172.31.0.20' (ECDSA) to the list of known hosts.
45731365@172.31.0.20's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-106-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

45731365@flag1:~$ domainname
ad6c8d53960accf168a8dcf7710ef945
45731365@flag1:~$
```

The flag is: **ad6c8d53960accf168a8dcf7710ef945**

REFLECTION:

I took some time trying to figure this flag out, tried many suitable commands for the domain but no success, but later figured it out using the internet and nmap. The purpose of this flag is to give a piece of knowledge on how to access a domain using ssh and take the privilege of it.

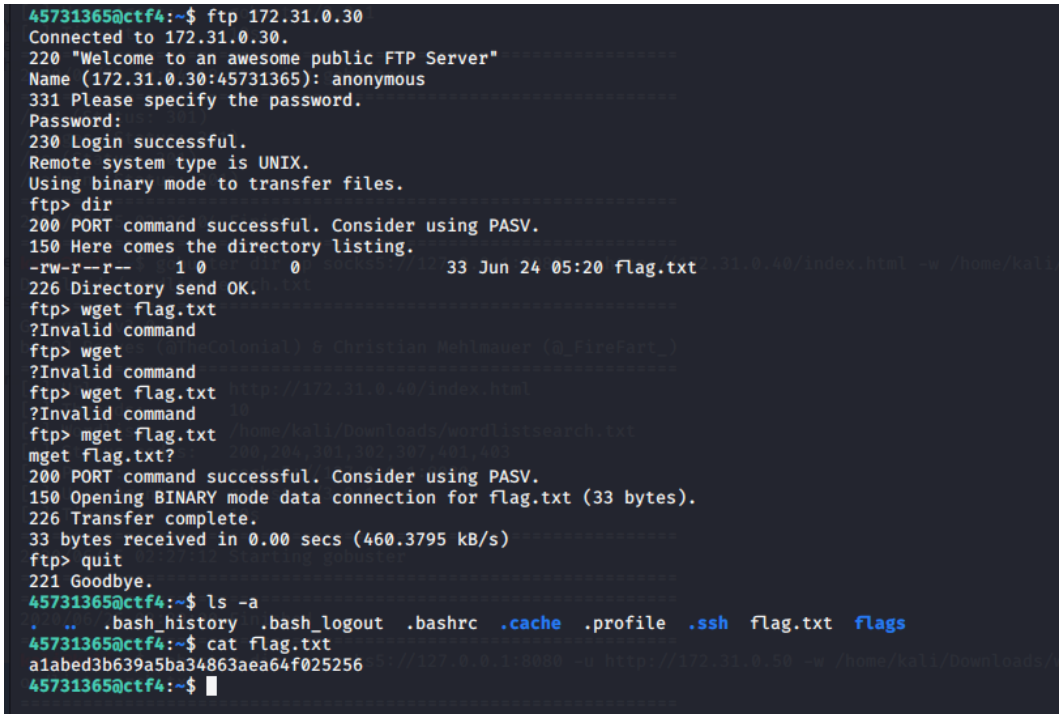
FLAG 2: Filed Away

Undoubtedly this flag is related to a file transfer. Did some research, found out the only way files could be transferred is through FTP or SFTP. Consulted my nmap scan, I got to know that it was running on an FTP server. When I tried logging in, I was asked with a username and password, I attempted my ssh username and password on it and I failed. With the help of Google, I found I could do an anonymous login. After it was accessed, I opened the directory to see if I could get any flag. The flag was right there in the directory. To be able to get and transfer the flag to the main host server I had to input **mget** command.

mget: it is used to retrieve content from web servers. It supports downloading via, HTTP, HTTPS, FTP.

ftp 172.31.0.30
username: anonymous
password was left blank.

dir
get flag.txt (this gets the file and ask where to save it, I clicked enter and it was saved in the Server)



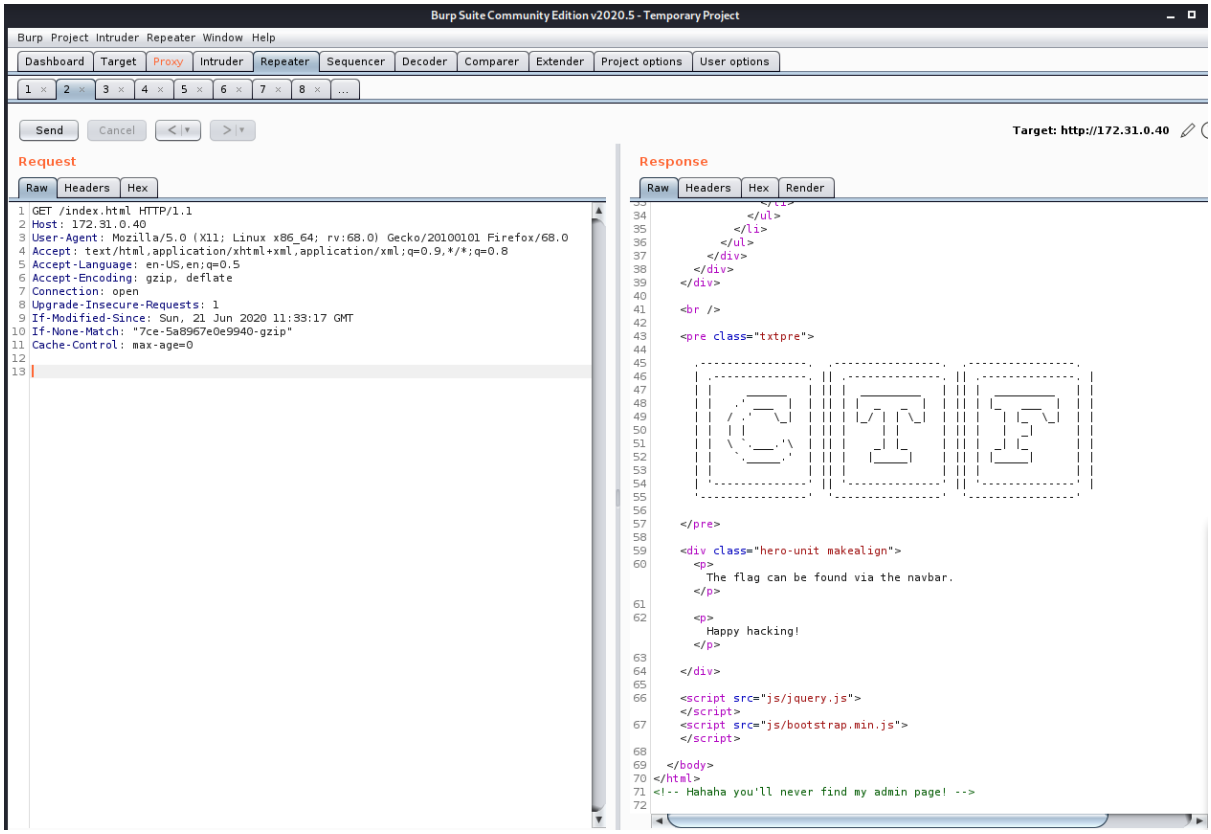
The flag is: **a1abed3b639a5ba34863aea64f025256**

Reflection:

This flag shows the capabilities of using file transfer protocol (FTP). A computer that has an FTP address is dedicated to receiving an FTP connection. This computer is referred to as an FTP server. This flag shows how to get information from an FTP server and transfer it to the main server so it could be easily accessed.

FLAG 3: Administrative deficit

This task is web-based according to the hint in the filename flag3.txt. With the name “administrative deficit” it suggests an issue with the database server. Firstly, we need to get the address of the task from nmap (172.31.0.40) and put it in a web browser. After the website was opened, I accessed the page source and I found something interesting on the HTML body which says **<!-- Hahaha you'll never find my admin page! -->**. The hint was to access the admin page. I tried to alter the web address to http://172.31.0.40/index.html/?page=admin but I was not successful.

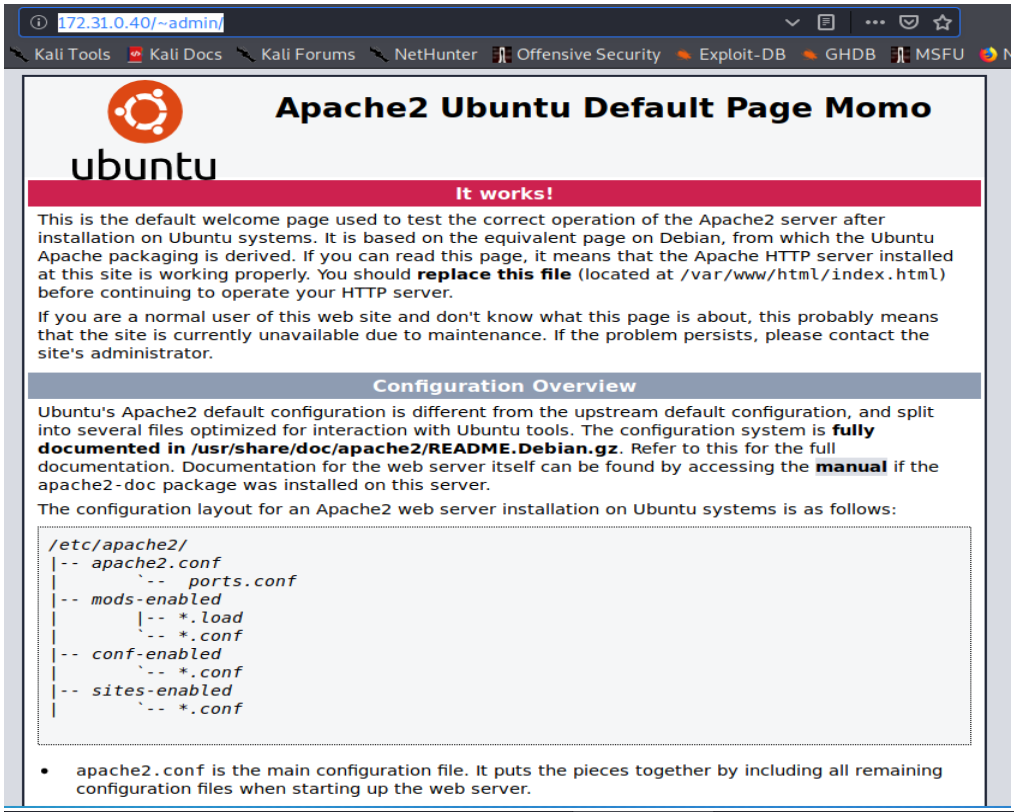


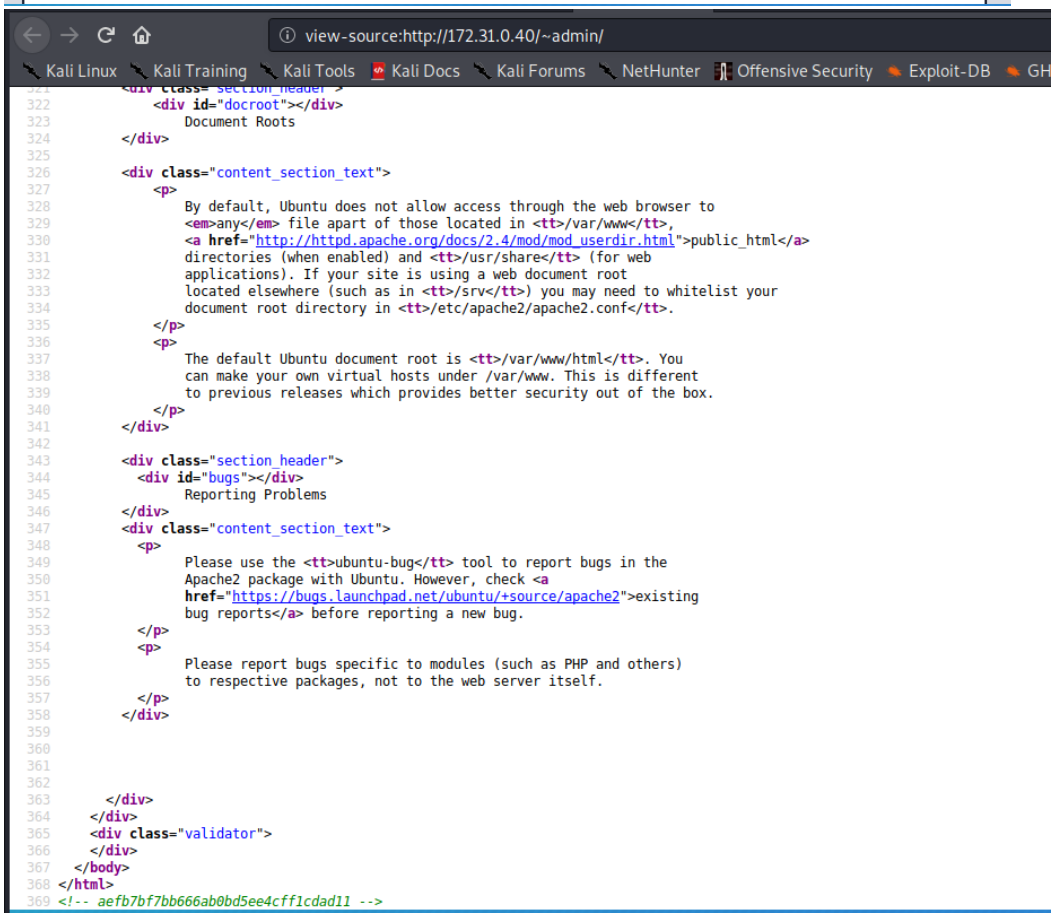
Finally thought of using gobuster with socksv5 proxy to solve the task, which gave the admin details

gobuster dir -p socks5://127.0.0.1:8080 -u http://172.31.0.40 -w /home/kali/Downloads/wordlistsearch.txt

```
kali@kali:~$ gobuster dir -p socks5://127.0.0.1:8080 -u http://172.31.0.40 -w /home/kali/Downloads/w
ordlistsearch.txt
===============================================================
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
===============================================================
[+] Url:            http://172.31.0.40
[+] Threads:        10
[+] Wordlist:       /home/kali/Downloads/wordlistsearch.txt
[+] Status codes:   200,204,301,302,307,401,403
[+] Proxy:          socks5://127.0.0.1:8080
[+] User Agent:     gobuster/3.0.1
[+] Timeout:        10s
===============================================================
2020/06/25 02:24:18 Starting gobuster
===============================================================
/css (Status: 301)
/images (Status: 301)
/js (Status: 301)
/~admin (Status: 301)
===============================================================
2020/06/25 02:26:04 Finished
===============================================================
```

I proceed to the admin page: http://172.31.0.40/~admin/ which is the Apache2 Ubuntu default page. Then I went to check the page source for the flag. The flag was located at the bottom of the page.





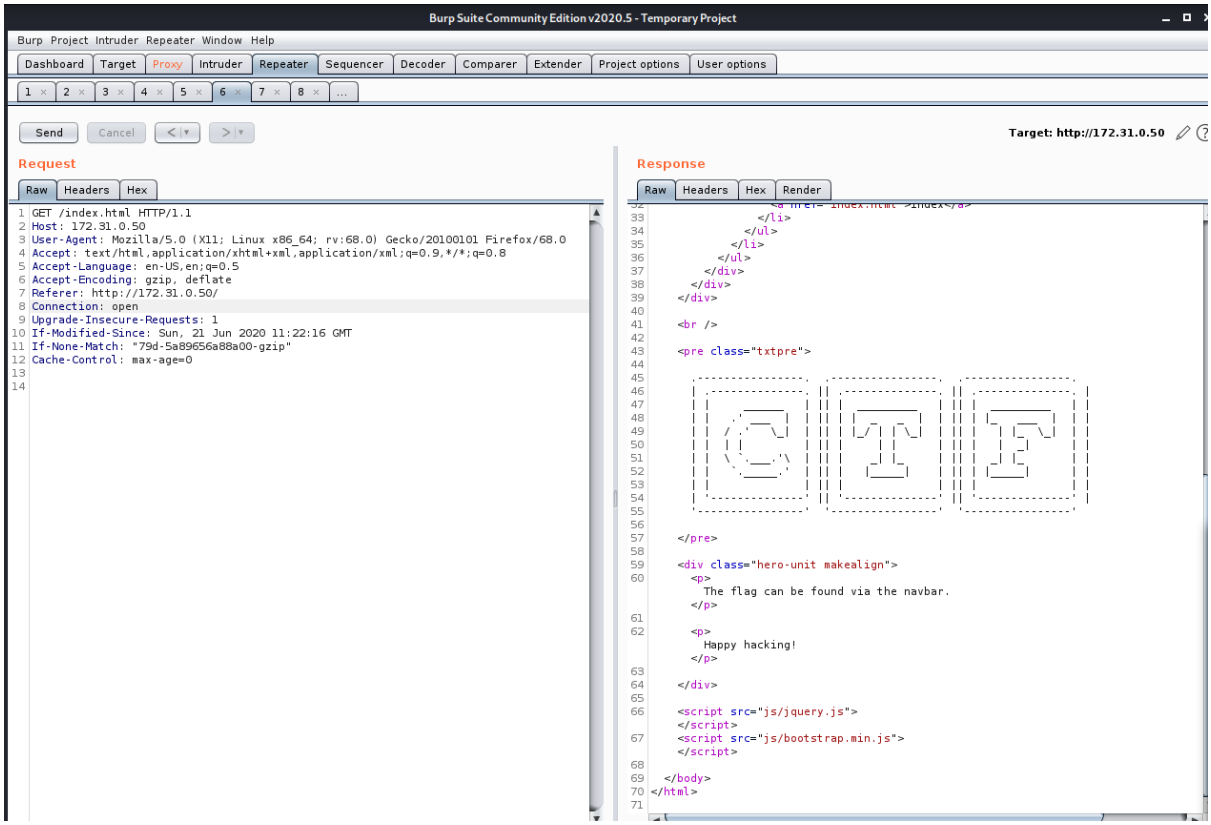
The flag is **aefb7bf7bb666ab0ba5ee4cff1cdad11**

Reflection:

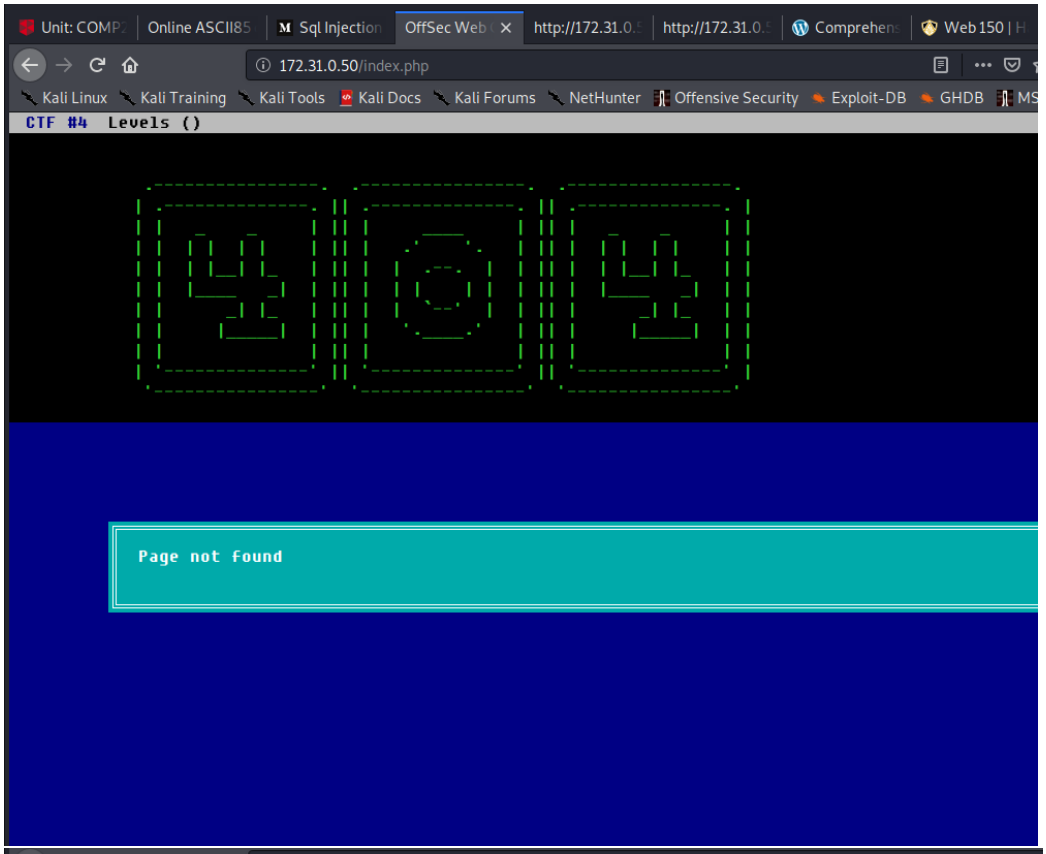
I did not realize on time that this task had to do with gobuster. I had to do some google search on how to link the socks proxy on gobuster to get the admin page. Also, Thanks to Mr. Damian who restarted the server for students to solve the remaining flags.

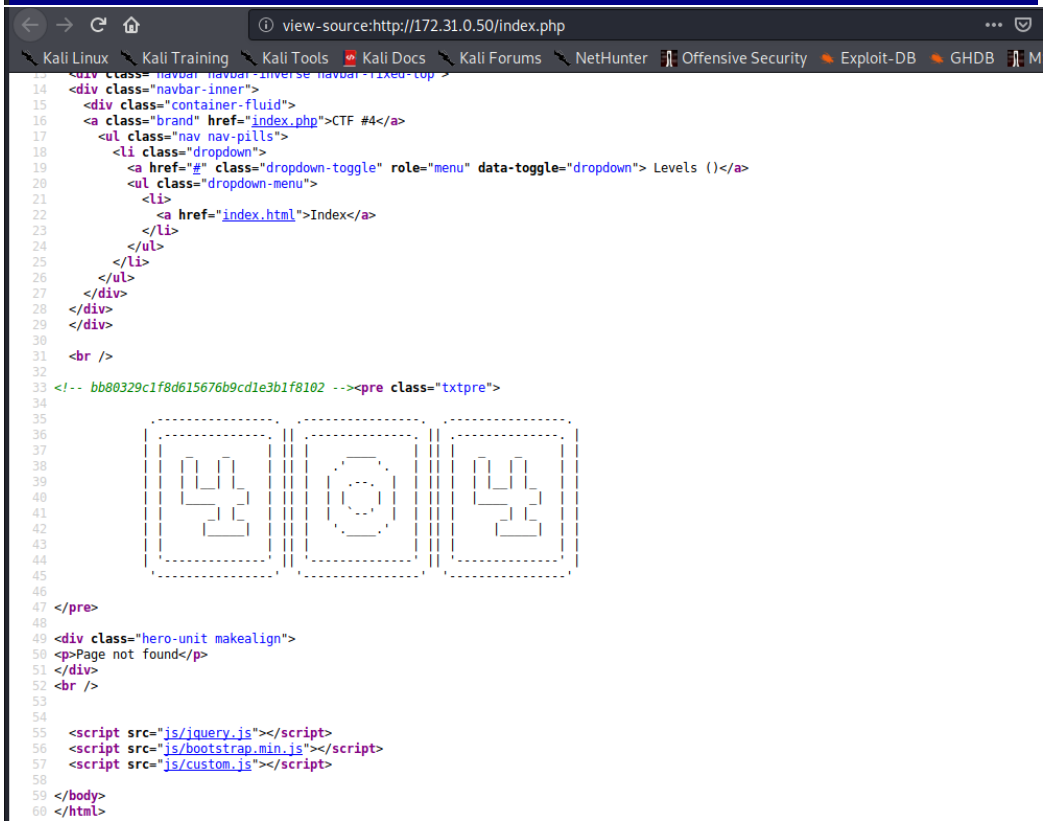
FLAG 4: Hidden in plain sight

Doing the cat command + flag4.txt shows that this is a web-based task. I got to check nmap and copied the address of flag4 “172.31.0.50” and inputted it into a Firefox browser. With the name “hidden in plain sight”, this flag has to do with accessing the page source and viewing the flag. There was no clue on the page source. I opened burp suite to intercept the packet by changing the connection from closed to open and sending it to the repeater and also changing the values of the connection from closed to open. When I clicked the send button, I was hoping I would get a response that includes the flag but was not successful.



I eventually solved the flag after so many trials, what I did was to change the address from **172.31.0.50/index.html** to **172.31.0.50/index.php** then check the page source for the flag.





The flag is: **bb80329c1f8d615676b9cd1e3b1f8102**

Reflection:

I remembered in the CTF2 I did, there was a flag which had to do with **Indices**, where you change the IP address from **10.46.225.212/index.html** to **10.46.225.212/index.php** and the

flag would be revealed. It took me so long to think about this method but later figured it out.

FLAG 5: Secret Identity

For flag 5, it has something to do with being anonymous, with the hint “secret identity”. I went over to the scanned result of nmap to check which service and port number flag5 is running on. This flag5 has a TCP port and also a ssh service running. With all these hints stated, it is clear that this flag could be accessed through ssh with this address **172.31.0.60.**

```
45731365@ctf4:~/flags$ ssh 172.31.0.60
45731365@172.31.0.60's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-106-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Fri Jun 26 23:01:00 2020 from 172.31.0.199
```

I used commands like ps: -aux, printenv, lslogins -u to get some information relating to the flag but was not successful in it.

```
45731365@flag5:~$ printenv
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33;01:cd=40;33;01:or=40;31;0
1:mi=00:su=37;41:sg=30;43:ca=30;41:tw=30;42:ow=34;42:st=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc
=01;31:*.arj=01;31:*.taz=01;31:*.lha=01;31:*.lz4=01;31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.txz=01
;31:*.tzo=01;31:*.t7z=01;31:*.zip=01;31:*.z=01;31:*.Z=01;31:*.dz=01;31:*.gz=01;31:*.lrz=01;31:*.lz=0
1;31:*.lzo=01;31:*.xz=01;31:*.zst=01;31:*.tzst=01;31:*.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.tbz2=01;31
:*.tz=01;31:*.deb=01;31:*.rpm=01;31:*.jar=01;31:*.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;31:*.al
z=01;31:*.ace=01;31:*.zoo=01;31:*.cpio=01;31:*.7z=01;31:*.rz=01;31:*.cab=01;31:*.wim=01;31:*.swm=01;
31:*.dwm=01;31:*.esd=01;31:*.jpg=01;35:*.jpeg=01;35:*.mjpg=01;35:*.mjpeg=01;35:*.gif=01;35:*.bmp=01;
35:*.pbm=01;35:*.pgm=01;35:*.ppm=01;35:*.tga=01;35:*.xbm=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;35:
*.png=01;35:*.svg=01;35:*.svgz=01;35:*.mng=01;35:*.pcx=01;35:*.mov=01;35:*.mpg=01;35:*.mpeg=01;35:*.
m2v=01;35:*.mkv=01;35:*.webm=01;35:*.ogm=01;35:*.mp4=01;35:*.m4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt
=01;35:*.nuv=01;35:*.wmv=01;35:*.asf=01;35:*.rm=01;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;35:*.fli=01;
35:*.flv=01;35:*.gl=01;35:*.dl=01;35:*.xcf=01;35:*.xwd=01;35:*.yuv=01;35:*.cgm=01;35:*.emf=01;35:*.o
gv=01;35:*.ogx=01;35:*.aac=00;36:*.au=00;36:*.flac=00;36:*.m4a=00;36:*.mid=00;36:*.midi=00;36:*.mka=
00;36:*.mp3=00;36:*.mpc=00;36:*.ogg=00;36:*.ra=00;36:*.wav=00;36:*.oga=00;36:*.opus=00;36:*.spx=00;3
6:*.xspf=00;36:
SSH_CONNECTION=172.31.0.199 46280 172.31.0.60 22
LESSCLOSE=/usr/bin/lesspipe %s %s
LANG=C.UTF-8
USER=45731365
PWD=/home/45731365
HOME=/home/45731365
SSH_CLIENT=172.31.0.199 46280 22
SSH_TTY=/dev/pts/0
MAIL=/var/mail/45731365
TERM=xterm-256color
SHELL=/bin/bash
SHLVL=1
LOGNAME=45731365
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games
LESSOPEN=| /usr/bin/lesspipe %s
_=/usr/bin/printenv
45731365@flag5:~$
45731365@flag5:~$ ps -aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0  55468 20588 ?        Ss   Jun24   0:26 /usr/bin/python /usr/bin/supervisor
root         8  0.0  0.0  72300  6440 ?        S    Jun24   0:00 /usr/sbin/sshd -D
root       572  0.0  0.0 105688  7176 ?        Ss   02:49   0:00 sshd: 45731365 [priv]
45731365   587  0.0  0.0 107984  5524 ?        R    02:49   0:00 sshd: 45731365@pts/0
45731365   588  0.0  0.0  20256  3832 pts/0    Ss   02:49   0:00 -bash
45731365   604  0.0  0.0  38448  3400 pts/0    R+   02:58   0:00 ps -aux
45731365@flag5:~$ ^C
45731365@flag5:~$ lslogins -u
 UID USER      PROC PWD-LOCK PWD-DENY LAST-LOGIN GECOS
   0 root         3                              root
1000 45731365    3                     02:49
1001 alice       0                              Alice
```

Reflection:

Quality time was spent on this task because the hints were so clear. I am sure that this flag is located in this address: **172.31.0.60** but could not find a way to access it.

FLAG 6: What's my config?

Firstly, this is a network CTF task. The flag hint indicates that it is a configuration file in the server. I was checking the directories in the server if I could see a config file but was not successful in it. I tried a reverse lookup host to get valid information related to the flag, it did not plan out well. I got a "connection refused" indicator, this suggests that this flag has nothing to do with the Domain Name System (DNS) lookup.

```
45731365@ctf4:~$ dig -x 172.31.0.70

; <<>> DiG 9.11.3-1ubuntu1.12-Ubuntu <<>> -x 172.31.0.70
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8924
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;70.0.31.172.in-addr.arpa.      IN      PTR

;; ANSWER SECTION:
70.0.31.172.in-addr.arpa. 600   IN      PTR     45731365_flag6.1130_45731365_back.

;; Query time: 0 msec
;; SERVER: 127.0.0.11#53(127.0.0.11)
;; WHEN: Sat Jun 27 03:54:37 UTC 2020
;; MSG SIZE  rcvd: 113

45731365@ctf4:~$ host -t axfr 45731365_flag6.1130_45731365_back 172.31.0.70
Trying "45731365_flag6.1130_45731365_back"
;; Connection to 172.31.0.70#53(172.31.0.70) for 45731365_flag6.1130_45731365_back failed: connectio
n refused.
45731365@ctf4:~$
```

Reflection:

In as much I didn't get this flag, I am still sceptical of where this flag resides in. All scanned ports on this address: **172.31.0.70** were closed and also there wasn't any service running. With the flag hint, this flag could be located in a configuration directory.

CONCLUSION AND REFLECTION

Due to the growing activeness of cyber theft around the globe, cybersecurity industries are creating events such as CTF and Hack The Box exercises for students in the information security field to practice their skills and also to learn how to act when an attack is being made.

Out of all the flags I attempted, I found flag 5 and flag 6 are quite challenging and difficult to understand. I lacked more depth in the network pen testing due to me paying much interest in the system and web penetration test. My performance can be improved in the future if I continue to try some hack the box challenges to improve my understanding and learning.

References

Nmap Cheet Sheet. (2019, August 28). Retrieved from HACKER TARGET: https://hackertarget.com/nmap-cheatsheet-a-quick-reference-guide/

Reeves, O. (2020). *Gobuster Package Description.* Retrieved from Kali Tools: https://tools.kali.org/web-applications/gobuster

SSH Command. (2019, December 17). Retrieved from Linuxize: https://linuxize.com/post/ssh-command-in-linux/

Exploiting HTTP request smuggling vulnerabilities. (n.d.). Retrieved May 15, 2020, from https://portswigger.net/web-security/request-smuggling/exploiting

HTTP response header injection. (n.d.). Retrieved May 11, 2020, from https://portswigger.net/kb/issues/00200200_http-response-header-injection