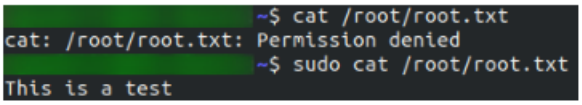
Task2: security Bypass

- CVE-2019-14287 là một lỗ hổng được tìm thấy trong chương trình Unix Sudo bởi một nhà nghiên cứu làm việc cho Apple: Joe Vennix. Thật trùng hợp, anh ấy cũng tìm thấy lỗ hổng mà chúng ta sẽ đề cập trong phần tiếp theo của loạt bài này. Khai thác này kể từ đó đã được sửa, nhưng vẫn có thể xuất hiện trong các phiên bản cũ hơn của Sudo (phiên bản <1.8.28), vì vậy, rất đáng để theo dõi!
- Đối với những người có thể không quen thuộc với nó: sudo là một lệnh trong unix cho phép bạn thực thi các chương trình như những người dùng khác. Điều này thường mặc định cho siêu người dùng (root), nhưng cũng có thể thực thi các chương trình như những người dùng khác bằng cách chỉ định tên người dùng hoặc UID của họ. Ví dụ: sudo thường được sử dụng như vậy: sudo <command>, nhưng bạn có thể chọn thực thi thủ công với tư cách người dùng khác như sau: sudo -u#<id> <command>. Điều này có nghĩa là bạn sẽ giả vờ là một người dùng khác khi thực hiện lệnh đã chọn, lệnh này có thể cấp cho bạn các quyền cao hơn so với những gì bạn có thể có. Như một ví dụ:



- Trong ví dụ này, tài khoản người dùng của tôi không có quyền đọc tệp /root/root.txt, vì vậy tôi đã sử dụng sudo để tạm thời cấp cho mình quyền root để đọc tệp.
- Giống như nhiều lệnh trên hệ thống Unix, sudo có thể được cấu hình bằng cách chỉnh sửa tệp cấu hình trên hệ thống của bạn. Trong trường hợp này, tệp đó có tên /etc/sudoers. Bạn không nên chỉnh sửa trực tiếp tệp này do tầm quan trọng của nó đối với quá trình cài đặt hệ điều hành, tuy nhiên, bạn có thể chỉnh sửa tệp này một cách an toàn bằng lệnh sudo visudo, lệnh này sẽ kiểm tra thời điểm bạn lưu để đảm bảo rằng không có cấu hình sai.
- Lỗ hổng mà chúng tôi quan tâm đối với tác vụ này xảy ra trong một tình huống rất cụ thể. Giả sử bạn có một người dùng mà bạn muốn cấp thêm quyền. Bạn muốn cho phép người dùng này thực thi một chương trình như thể họ là bất kỳ người dùng nào khác, nhưng bạn không muốn cho phép họ thực thi chương trình đó với quyền root. Bạn có thể thêm dòng này vào tệp sudoers:

<user> ALL=(ALL:!root) NOPASSWD: ALL

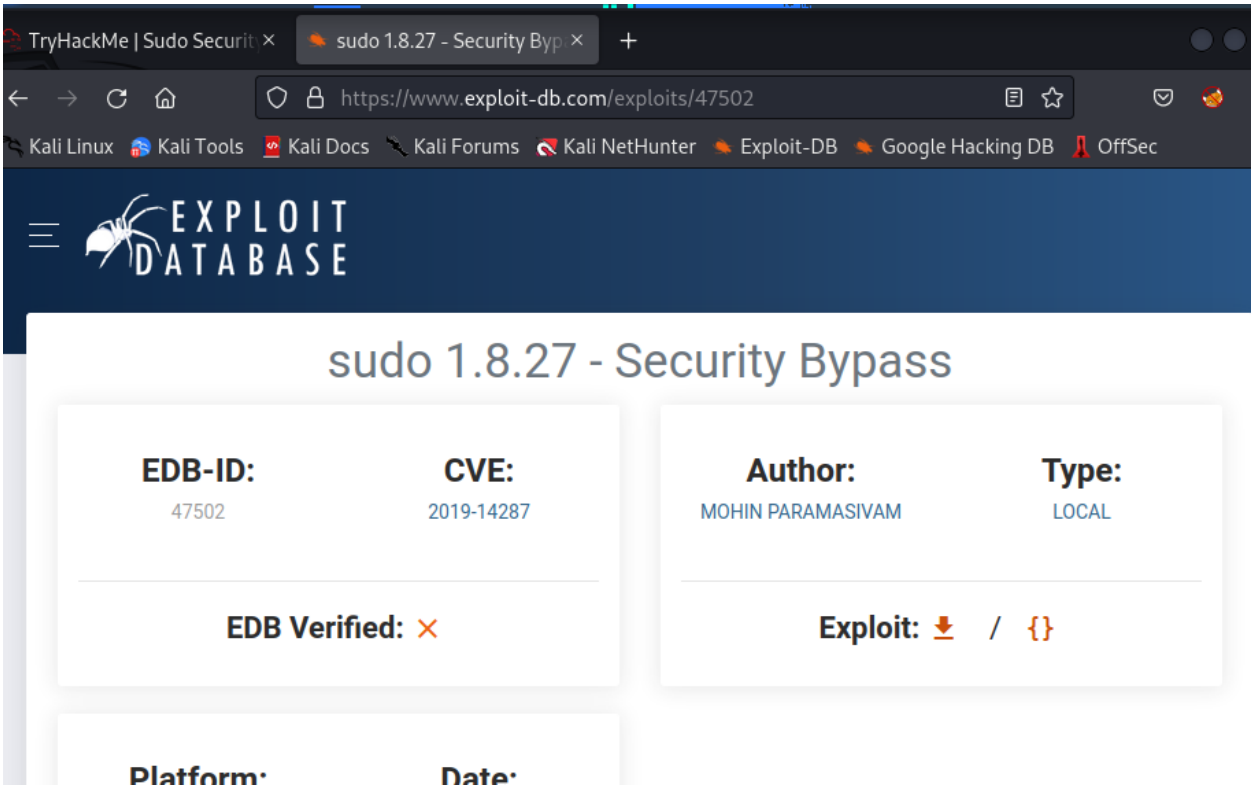
- Điều này sẽ cho phép người dùng của bạn thực thi bất kỳ lệnh nào với tư cách là người dùng khác, nhưng (về mặt lý thuyết) sẽ ngăn họ thực thi lệnh với tư cách là superuser/admin/root. Nói cách khác, bạn có thể giả làm bất kỳ người dùng nào, ngoại trừ quản trị viên.
- Về mặt lý thuyết.
  + Trên thực tế, với các phiên bản Sudo dễ bị tổn thương, bạn có thể vượt qua hạn chế này để thực thi các chương trình với quyền root, điều này rõ ràng là rất tốt cho việc leo thang đặc quyền!

+ Với cấu hình trên, việc sử dụng sudo -u#0 <command> (UID của root luôn bằng 0) sẽ không hoạt động vì chúng tôi không được phép thực thi các lệnh với tư cách là root. Nếu chúng tôi cố gắng thực hiện các lệnh với tư cách là người dùng 0, chúng tôi sẽ gặp lỗi. Nhập CVE-2019-14287.
+ Joe Vennix nhận thấy rằng nếu bạn chỉ định UID là -1 (hoặc giá trị tương đương không dấu: 4294967295), thì Sudo sẽ đọc sai giá trị này thành 0 (tức là gốc). Điều này có nghĩa là bằng cách chỉ định UID là -1 hoặc 4294967295, bạn có thể thực thi lệnh với quyền root, mặc dù bị ngăn chặn rõ ràng. Cần lưu ý rằng điều này sẽ chỉ hoạt động nếu bạn đã được cấp quyền sudo không phải root đối với lệnh, như trong cấu hình ở trên.
+ Trên thực tế, ứng dụng này như sau: sudo -u#-1 <command>

– Bây giờ đến lượt bạn.

+ SSH vào máy mà bạn đã triển khai trước đó, sử dụng cổng 2222.
+ Thông tin đăng nhập là:
+ Tên người dùng: tryhackme
+ Mật khẩu: tryhackme
+ Nếu bạn đang sử dụng Linux, lệnh sẽ giống như sau:
+ ssh -p 2222 tryhackme@MACHINE_IP

What command are you allowed to run with sudo? -> /bin/bash



What is the flag in /root/root.txt?

```
tryhackme@sudo-privesc:/$ sudo -u#-1 /bin/bash
root@sudo-privesc:/# ls
bin  boot  core  dev  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@sudo-privesc:/# cat /root/root.txt
THM{l33t_s3cur1ty_bypass}
root@sudo-privesc:/#
```