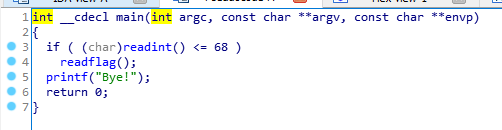
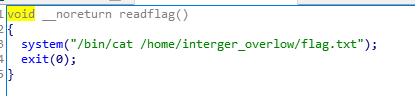
Write-up PISCTF (6/2019)

Team Q3Twolf - Phần pwn

Bài 1: I\_O

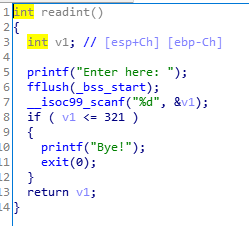
* Phân tích đề bài:





Đọc hàm main, đầu tiên sẽ nhận kết quả từ hàm readint() nếu <= 68 thì sẽ thực thi hàm readflag( ) => có flag. Nếu > 68 thì sẽ in ra chuỗi “Bye!” và kết thúc hàm main.

Phân tích hàm readint( ) để xem hàm này làm gì và trả về gì?



Đầu tiên hàm này khai báo một biến v1 có kiểu dữ liệu int.

Sau đó nhập vào 1 số nguyên và lưu vào v1.

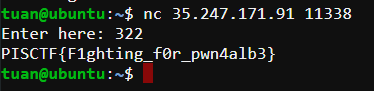
Nếu v1 < = 321 thì in ra chuỗi “Bye!” và thoát chương trình

Ngược lại thì trả về giá trị v1

Mới đọc code thì có thể sẽ thấy ngược đời là ta luôn phải nhập giá trị > 321 để không bị thoát chương trình, nhưng sẽ không thể nào lấy được flag vì 321 không thể nào nhỏ hơn 68.

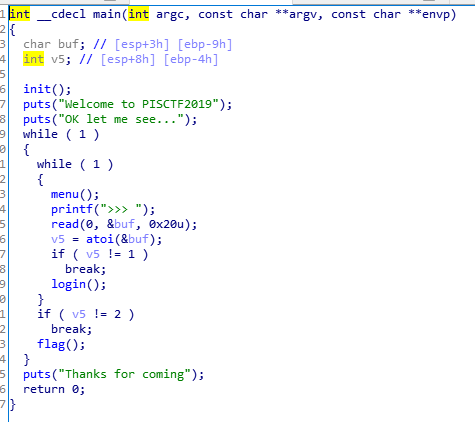
Đọc kĩ lại câu lệnh điều kiện if của hàm main, ta sẽ thấy giá trị v1 được trả lại từ hàm readint() sẽ được ép kiểu qua kiểu char. Ta ôn lại kiến thức, 1 int là 4 bytes, còn 1 char là 1 byte. Tức là chỉ so sánh 1 byte dữ liệu của giá trị chúng ta nhập vào.

Đổi qua hex để dễ thấy. Ta có hex(68) = 0x44, vậy ta sẽ so sánh 1 byte dữ liệu cuối của v1 với 0x44, chỉ cần số nhập vào chuyển qua hexa mà có 1 byte cuối nhỏ hơn 0x44 là được. Ví dụ số 322 có hex(322) = 0x142 , có 1 byte dữ liệu cuối là 0x42 thỏa mãn yêu cầu của chúng ta là nhỏ hơn 0x44 và cũng lớn hơn 321 theo yêu cầu đề bài. Submit kiếm flag nào.

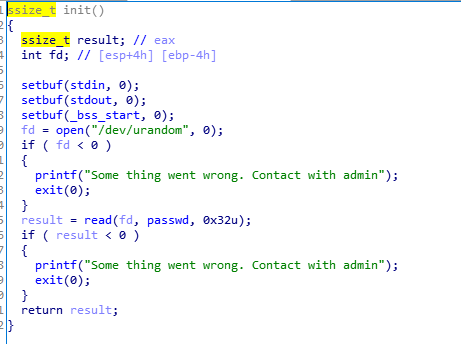


Bài 2: SIMPLE BOF

* Phân tích đề bài:



Đầu tiên hàm main sẽ gọi hàm init():



* Hàm init mở file “dev/urandom” rồi đọc 0x32 bytes dữ liệu vào biến passwd. Nếu xảy ra lỗi thì

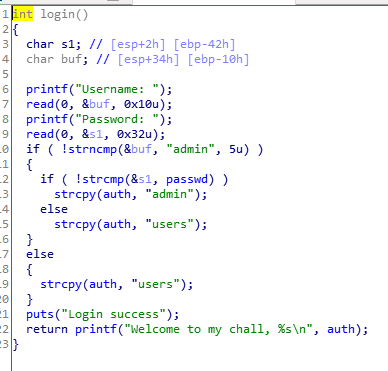
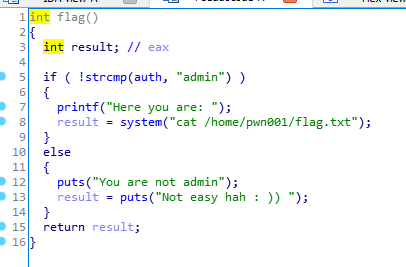
thoát chương trình.

Quay trở lại main, sau hàm init, chúng ta được nhập 0x20 bytes dữ liệu vào biến buf.

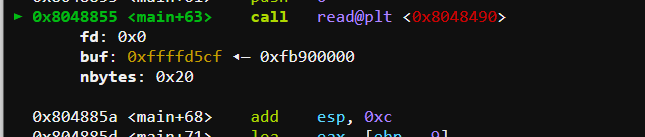
V5 = atoi(buf), tức là cho chúng ta nhập lựa chọn menu

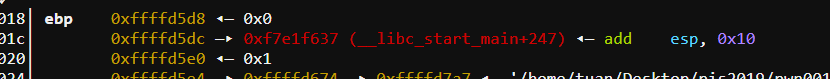
Nếu v5 == 1 thì sẽ vào hàm login(), nếu v5 ==2 thì sẽ vào hàm flag( ) , ngược lại sẽ thoát chương trình.

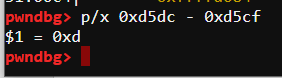
Ta tiếp tục phân tích hàm login() và hàm flag()

* Login():  
  
* Đầu tiên hàm sẽ cho chúng ta nhập username, password. Sau đó nó so sánh chuỗi username vừa nhập với “admin” và nếu username = “admin” thì sẽ so sánh chuỗi password chúng ta nhập với biến passwd, nếu password = passwd thì sẽ copy chuỗi “admin” vào biến auth, còn nếu password != passwd thì sẽ copy chuỗi “users” vào biến auth, tương tự nếu username != “admin” thì cũng sẽ copy chuỗi “users” vào auth. Cuối cùng in ra dòng chữ “wellcome …” và thoát hàm.
* Tiếp tục phân tích hàm flag():  
  
* Đầu tiên hàm sẽ so sánh biến auth() với chuỗi “admin” nếu bằng sẽ lấy được flag. Ngược lại là không được.
* Vậy mục tiêu là phải làm sao cho auth nhận được chuỗi “admin” rồi. Nhưng chúng ta không biết password đúng. Vậy cùng đọc lại đề để tìm những chỗ nghi hoặc có thể dẫn đến 1 số lỗi khác.
* Ta để ý, khi nhập lựa chọn menu, chúng ta chỉ 1 byte để chứa lựa chọn, nhưng đề bài cho tận 0x20 bytes, có thể nó sẽ dẫn tới tràn dữ liệu. mình sẽ debug để kiểm tra.

Mình thấy biến buf được lưu tại vị trí 0xffffd5cf

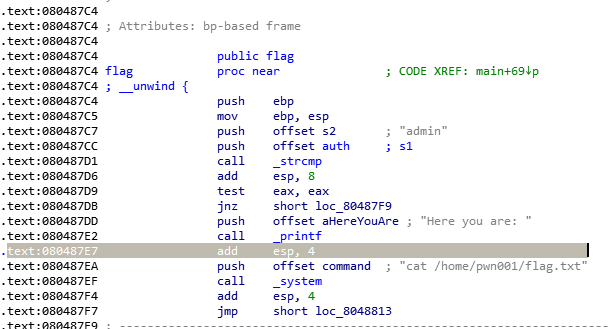


Và địa chỉ return của main lại nằm ở 0xffffd5dc

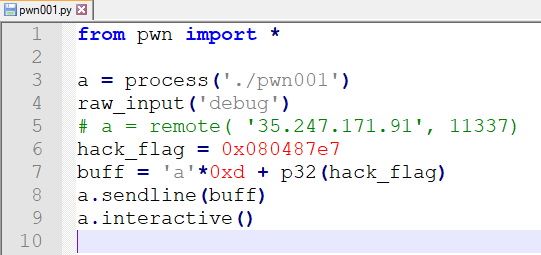
Offset = 0xd 

Vậy tức là nó có thể tràn được địa chỉ return của hàm main. Giờ đã biết sẽ bị tràn, chúng ta sẽ tìm địa chỉ return về sao cho thích hợp. Ban đầu mình có nghĩ sẽ gọi system(/bin/sh) nhưng chợt nhớ ra có hàm flag(), xem thử có thể return đâu đó trong hàm flag() được không.

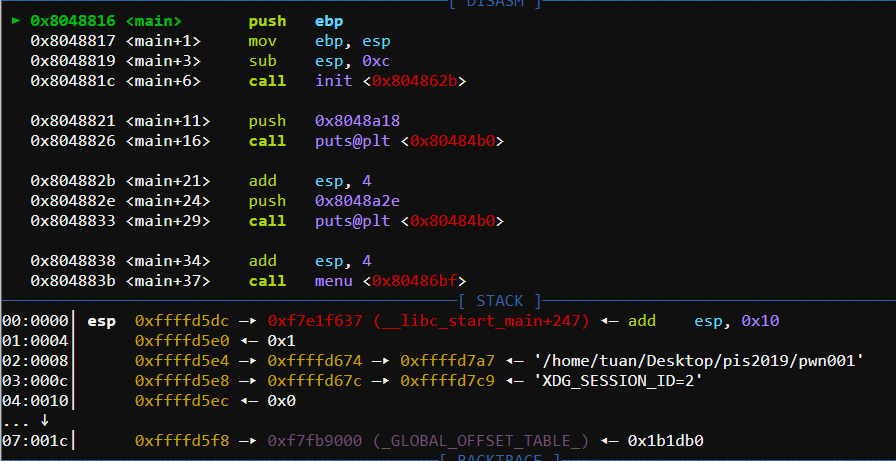
Nếu return về đầu hàm flag thì sẽ bị kiểm tra biến auth, vẫn sẽ fail. Vậy nếu ta bỏ qua kiểm tra và return về địa chỉ thực thi lệnh cat flag thì sao.



Ta thấy ở địa chỉ này, hàm sẽ đưa chuỗi lệnh vào và call system ngay lập tức, không hề bị cản trở bởi bất kì kiểm tra nào khác, vậy ta return về đó thôi.

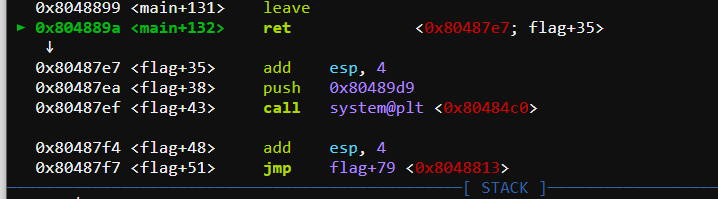


Vì khoảng cách giữa biến buf và return main là 0xd nên mình sẽ truyền cho nó 0xd kí tự, sau đó cộng cho địa chỉ cần return, là sẽ ghi đè lên địa chỉ return cũ thành địa chỉ return ta mong muốn.

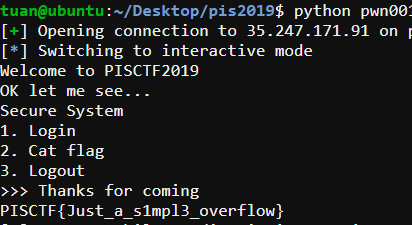
\

Bắt đầu chạy, ta thấy địa chỉ return main chưa thay đổi, vì ta chưa nhập vào.



Sau khi nhập vào ta thấy dữ liệu đã thay đổi. tiếp tục debug để xem có return về địa chỉ ta muốn hay không? Chắc chắn rồi, chạy đi đâu cho thoát  


Và cat flag thôi.



Cuộc thi này mình chỉ làm được 2 bài pwn, khá feed nhưng mình sẽ cố gắng hơn trong tương lại. Chúc các bạn đọc vui vẻ, nếu có bất kì thắc mắc có thể liên hệ mình để hỏi đáp.