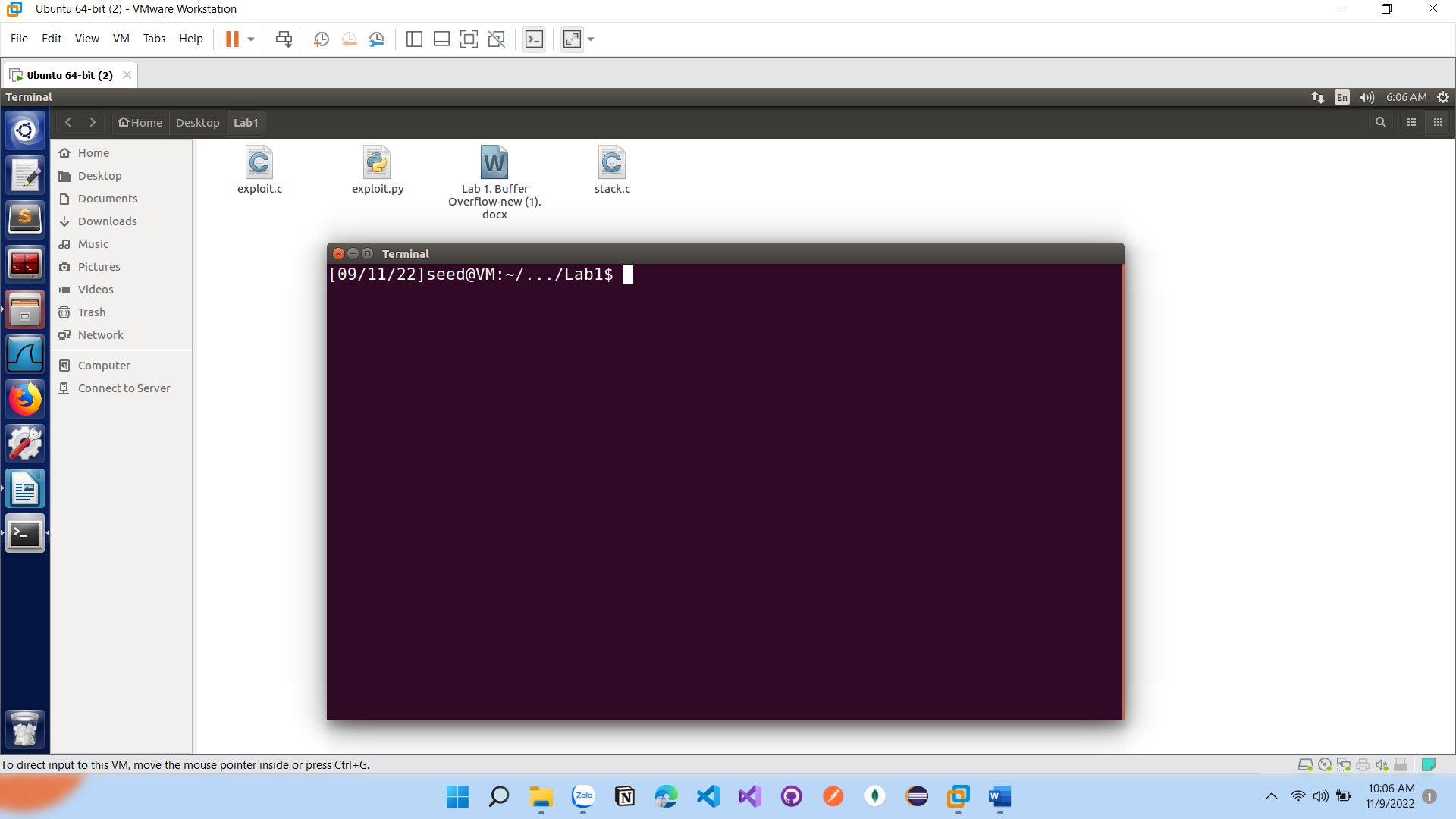
Tên: Lê Hải Đăng

Mã số sinh viên: 20110243

**BÁO CÁO THỰC HÀNH LAB1: BUFFER OVERFLOW**

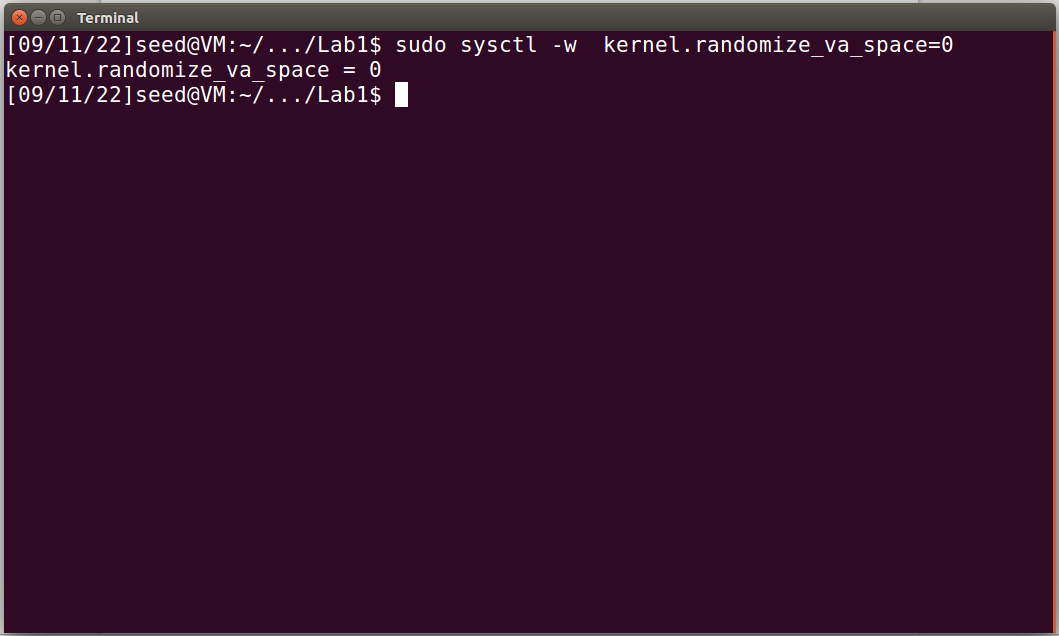
**Step 0: Preparation**

Cài đặt ubuntu 16.4 và chuẩn bị sẵn các file source code



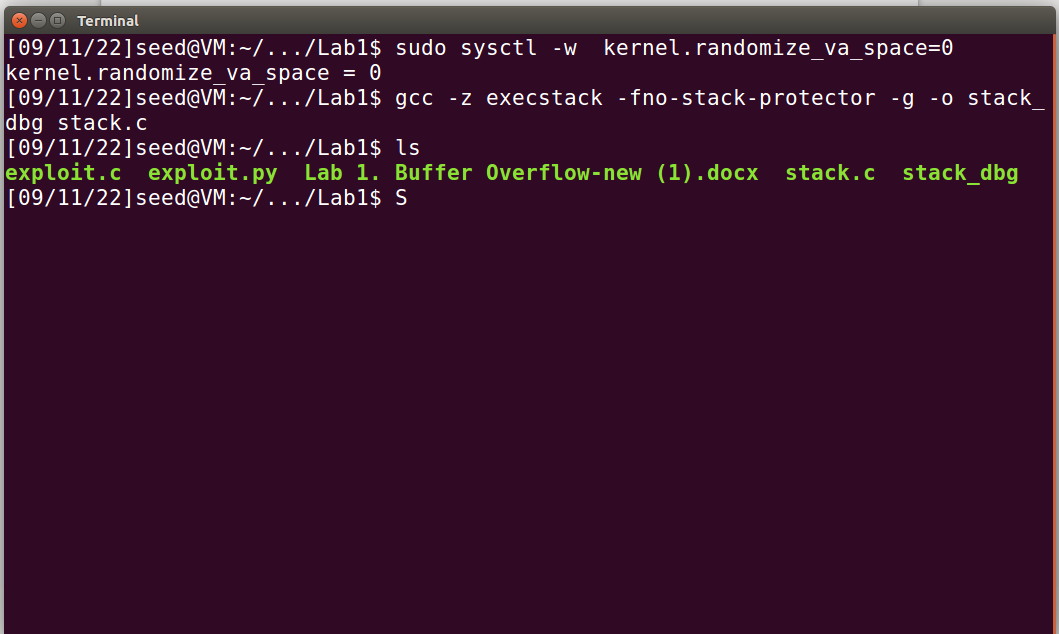
**Step 1. Disable address randomization**

Tắt chế độ random địa chỉ để có thể xác định được địa chỉ mong muốn

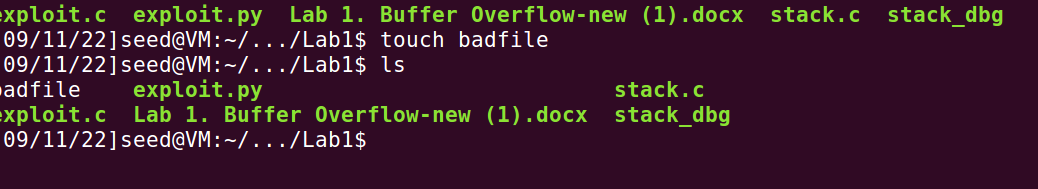


**Step 2. Finding the address of the inject code**

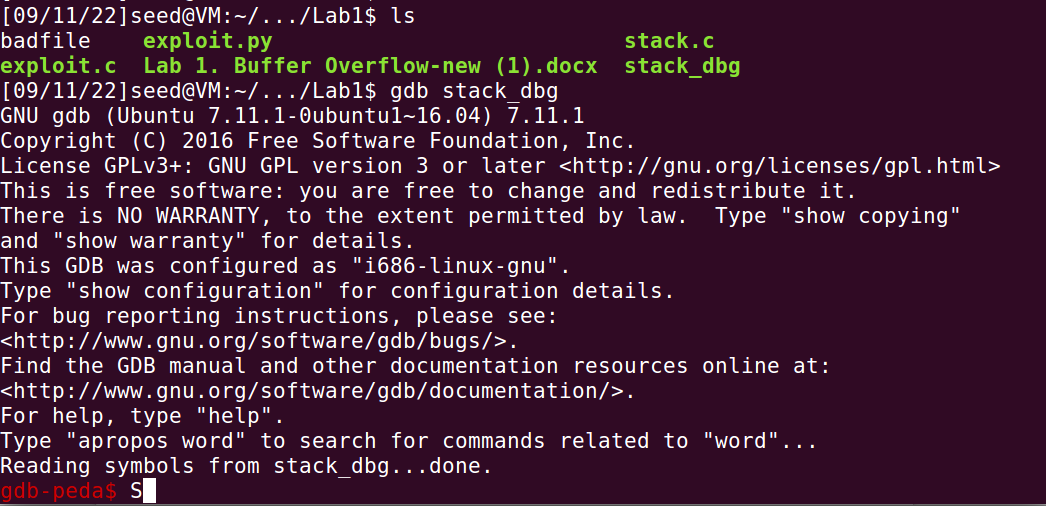
Đầu tiên, tắt chế độ bảo vệ của hệ điều hành ubuntu và compile file stack.c



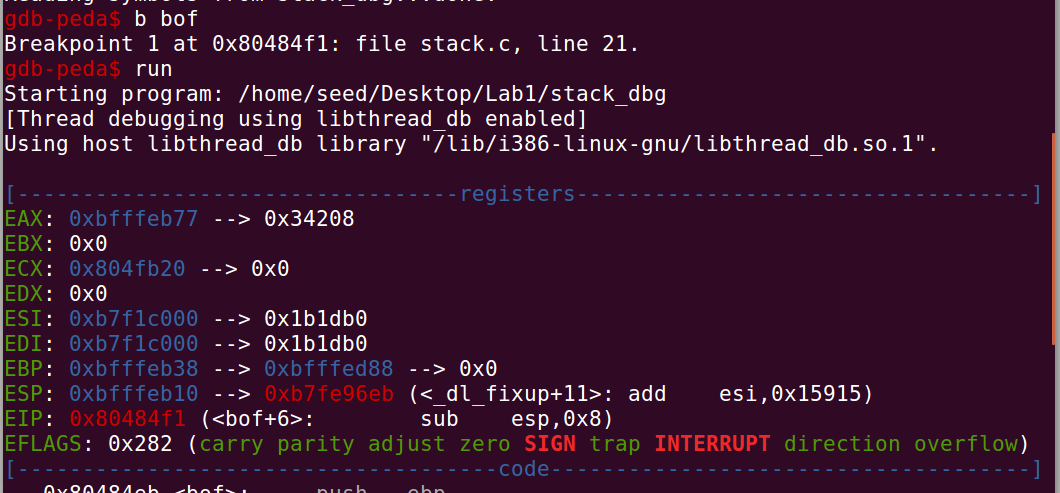
Tiếp theo, tạo 1 file tên là “badfile” trong đường dẫn hiện hành



Chạy file stack\_dbg ở chế độ debug để dịch ngược mã nguồn và xem code assembly của file này.



Đặt 1 break point ở hàm bof, sau đó dùng lệnh run để tiến hành debug

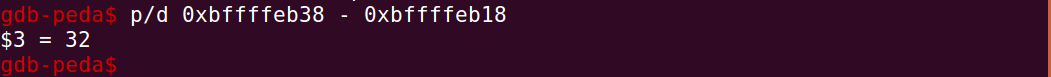




Tìm địa chỉ của con trỏ previous point frame (ebp) và buffer



Ta thấy khoảng cách giữa ebp và buffer là 0xbfffecc8 – 0xbfffeca8 = c8 – a8 = 20 ( HEX ) = 32 (DEC), để kiểm tra lại ta dùng lệnh:



Như vậy ta có thể tính ra địa chỉ của return address

= ebp + 32 + 4

= edp + 36

**Step 3. Edit exploit.c**

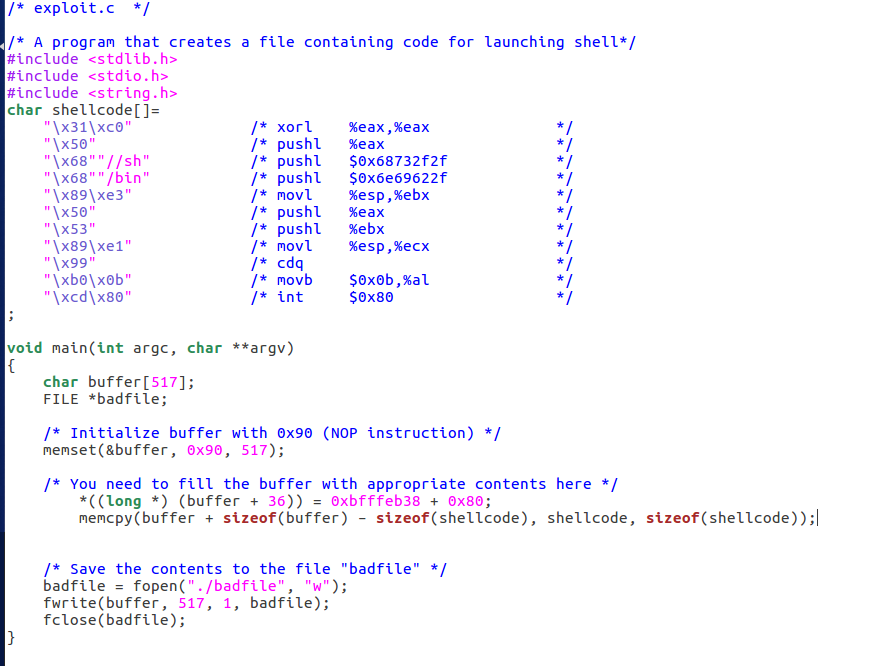
Ta có công thức tính địa chỉ trả về mã độc là : ebp + [randomNum], trong đó kết quả phải không chứa số 0 ( vì câu lệnh copy gặp \0 sẽ lập tức kết thúc).

Ta có địa chỉ ebp = “0xbfffeb38”

Lấy ebp + 0xb80 = bfffebb8 ( Không có số 0 )



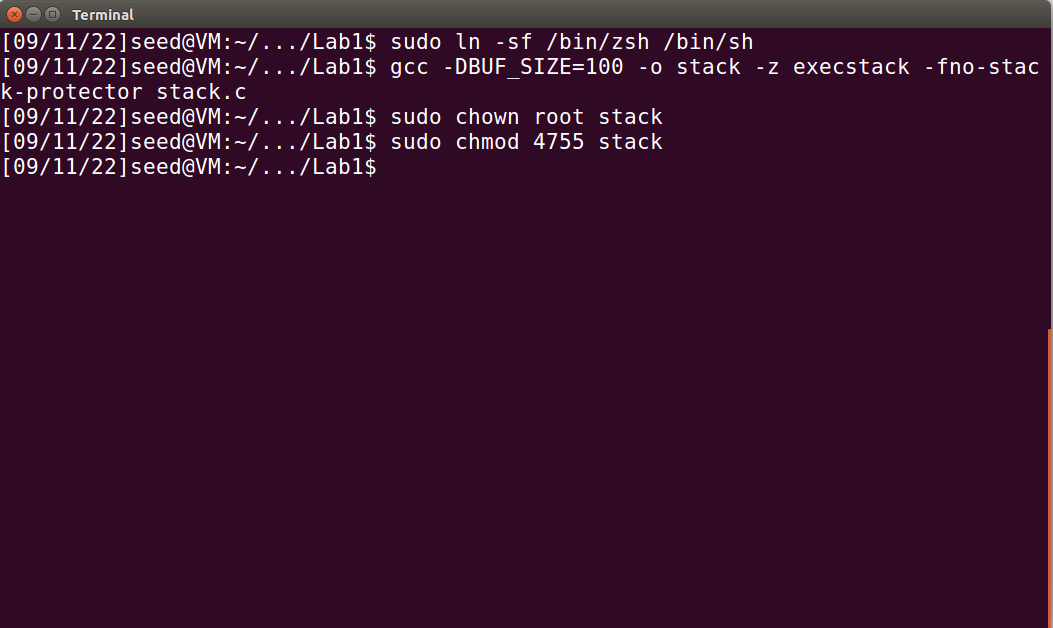
Tiếp theo, thêm code và thay đổi địa chỉ của ebp trong file exploit.c



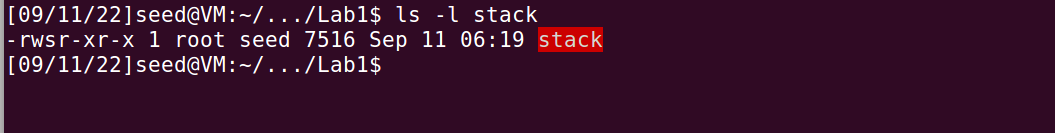
**Step 4. Execute**

Cấp đặt quyền root cho chương trình “stack” bằng cách tắt trình bảo vệ và bật bit đặc quyền để ứng dụng có thể chạy ở quyền root

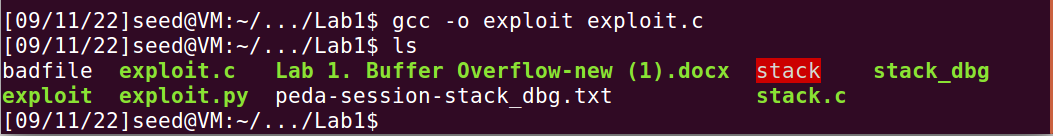
Đồng thời thay đổi giá chị của biến DBUF\_SIZE trong file stack.c thành 100



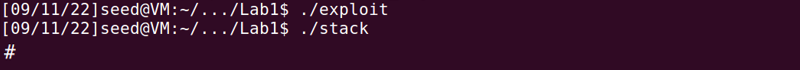
Tới bước này, có thể thấy chương trình đã có đặc quyền root



Tiếp theo, compile file exploit thành file thực thi



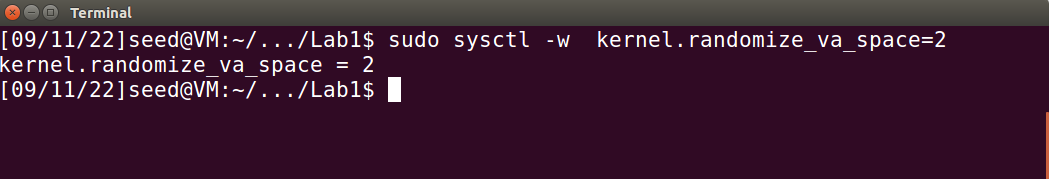
Tiến hành chạy chương trình cùng mới đoạn mã độc



Ta thấy kết quả là chúng ta đã truy cập được quyền admin mà không cần mật khẩu.

**Step 5. Defeating Address Randomization (ASLR)**

Để bảo vệ lại các chương trình, ta bật lại chế độ random địa chỉ



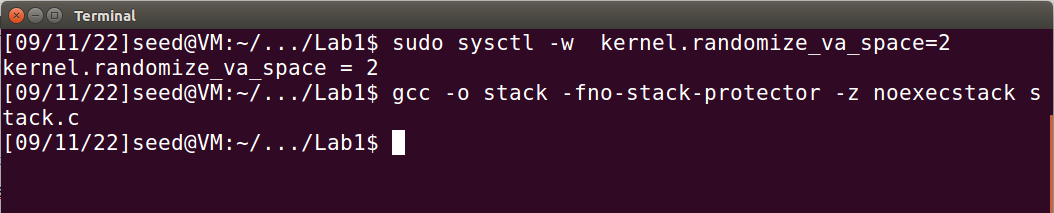
**Step 6. Turn on the StackGuard Protection**

Khi compile trường trình, ta không nên tắt chế độ bảo vệ thông qua

-fno-stack-protector

**Step 7. Turn on the Non-executable Stack Protection**

Compile lại chương trình stack, xử dụng noexecstack

****