Derrick Sung

**Buffer Overflow**

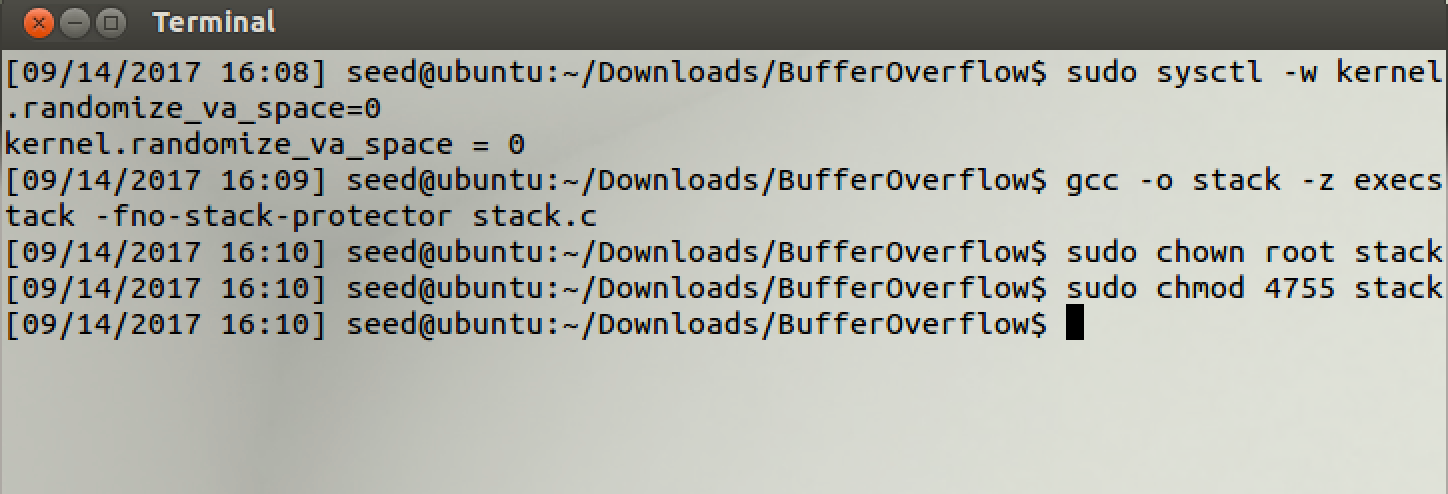
**Code:**

* **exploit.c (The two lines of code needed)**\*((long\*)(buffer + (0xbffff108 – 0xbffff0e8) + 4)) = (int)buffer; //set value for the return address

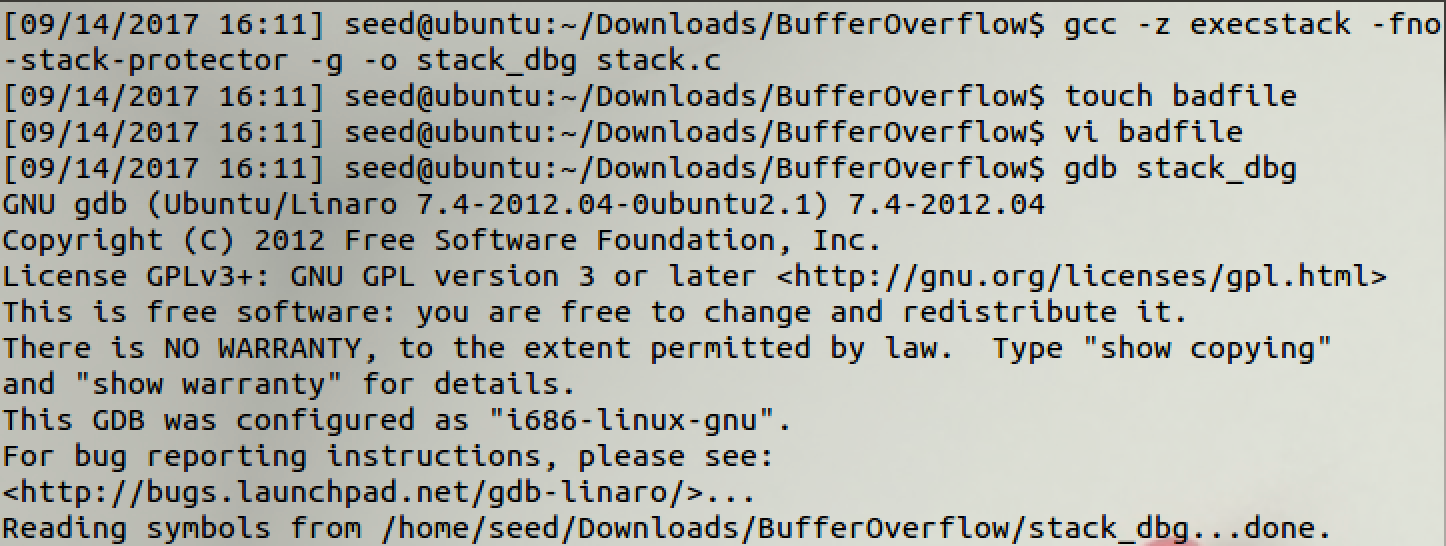
memcpy(buffer + sizeof(buffer) – sizeof(shellcode), shellcode, sizeof(shellcode)); //place shellcode towards the end of the bugger

**Result:**

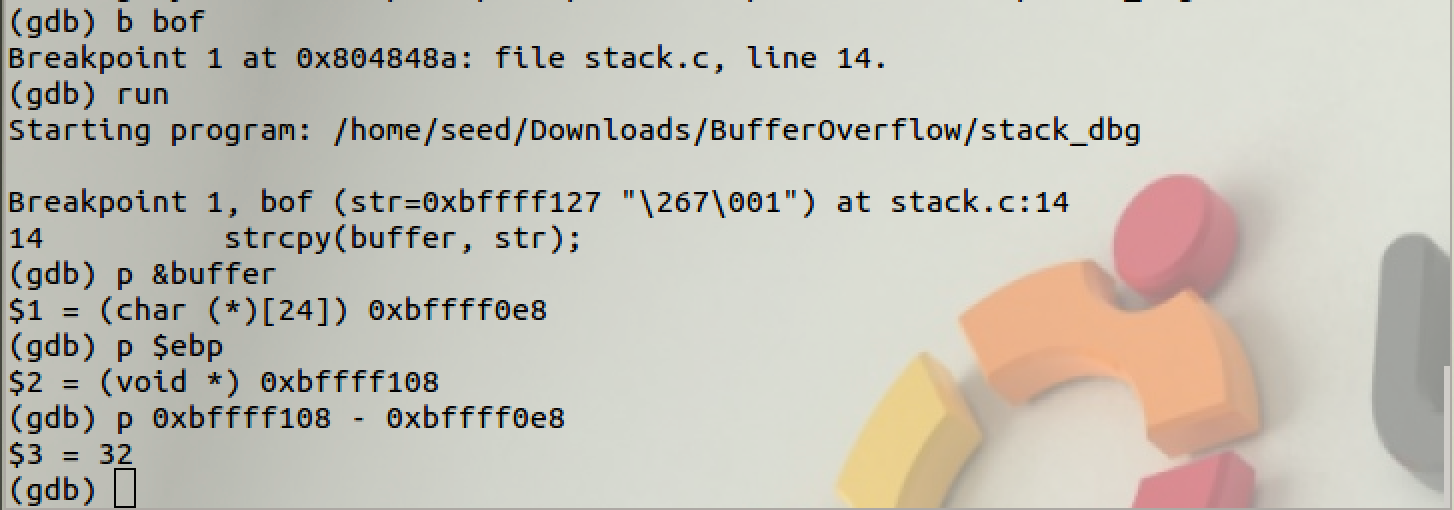
1. **Disable the randomizer, compile stack.c, and change the owner.**



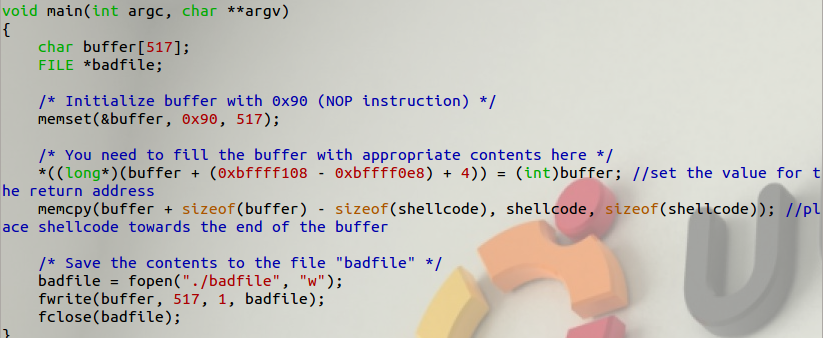
1. **Once owner has been changed, run debug mode.**



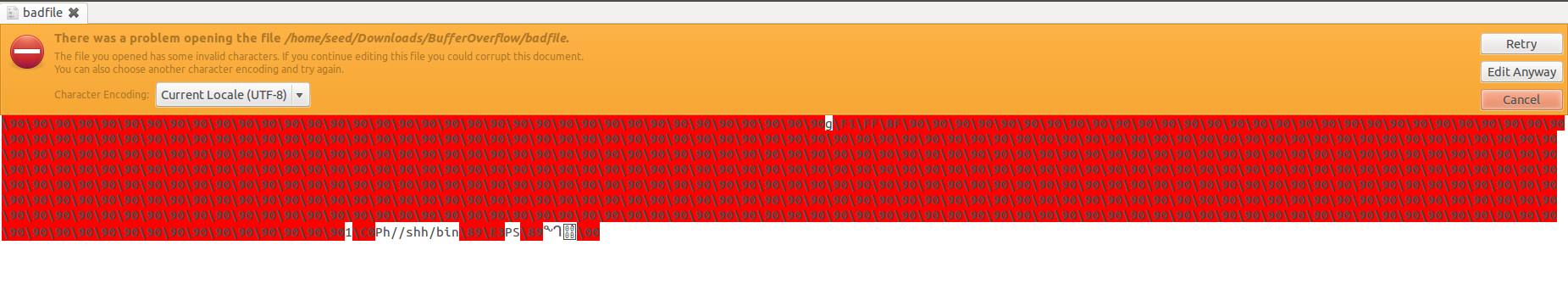
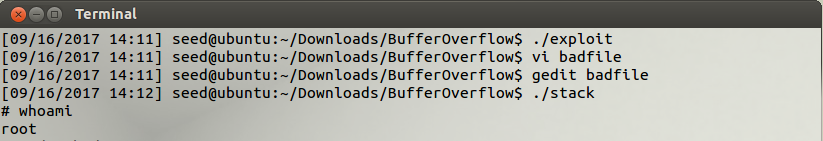
1. **Break at line 14. Print the address of buffer and ebp, then subtract the values from &buffer from $ebp. (i.e $ebp - &buffer)**



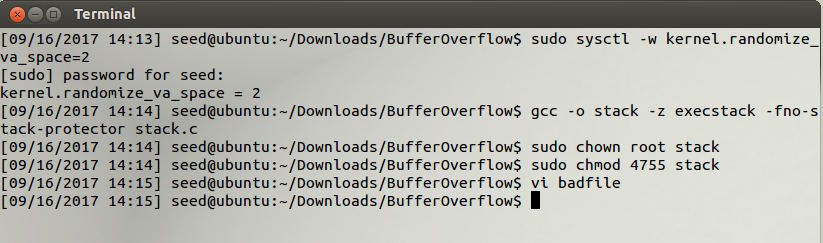
1. **Add finishing code to exploit.c (We take the value from the subtraction we did above and added in 4 as in 4bytes from the previous frame ptr on stack. This results in the return address.)**



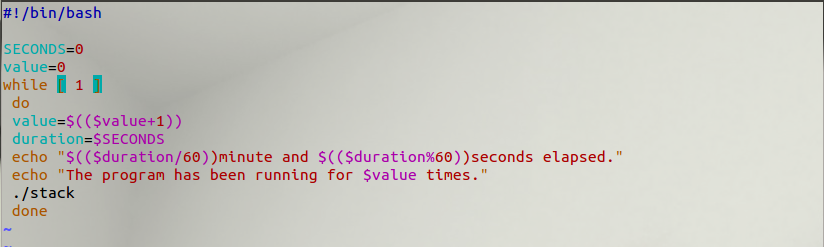
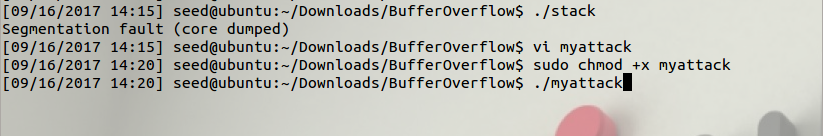
1. **Run exploit.c and stack.c, you will result with badfile**



1. **Re-enable the randomizer again, compile stack.c and change the owner again.**



1. **Run stack.c and result in the segmentation fault. Create myattack for the loop and run myattack.**



**Result: 15minutes and 16seconds when we hit root.**

