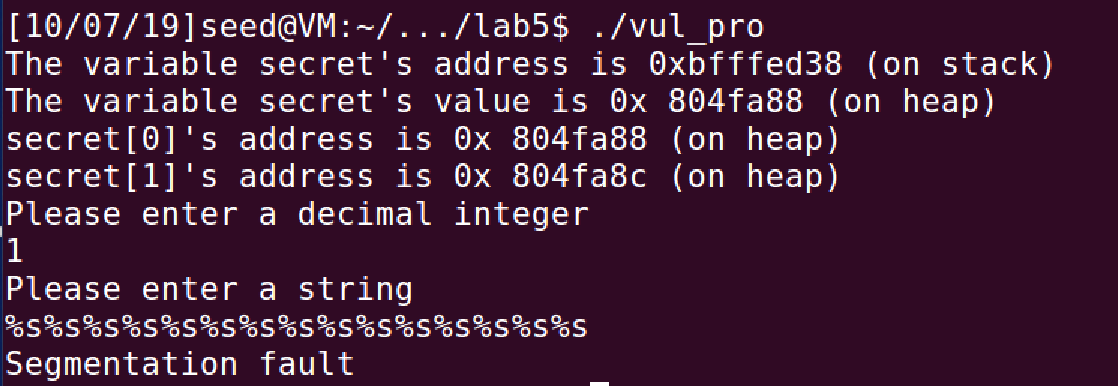
Lab 5

By Marco Lin

Task 1 Exploit the vulnerability

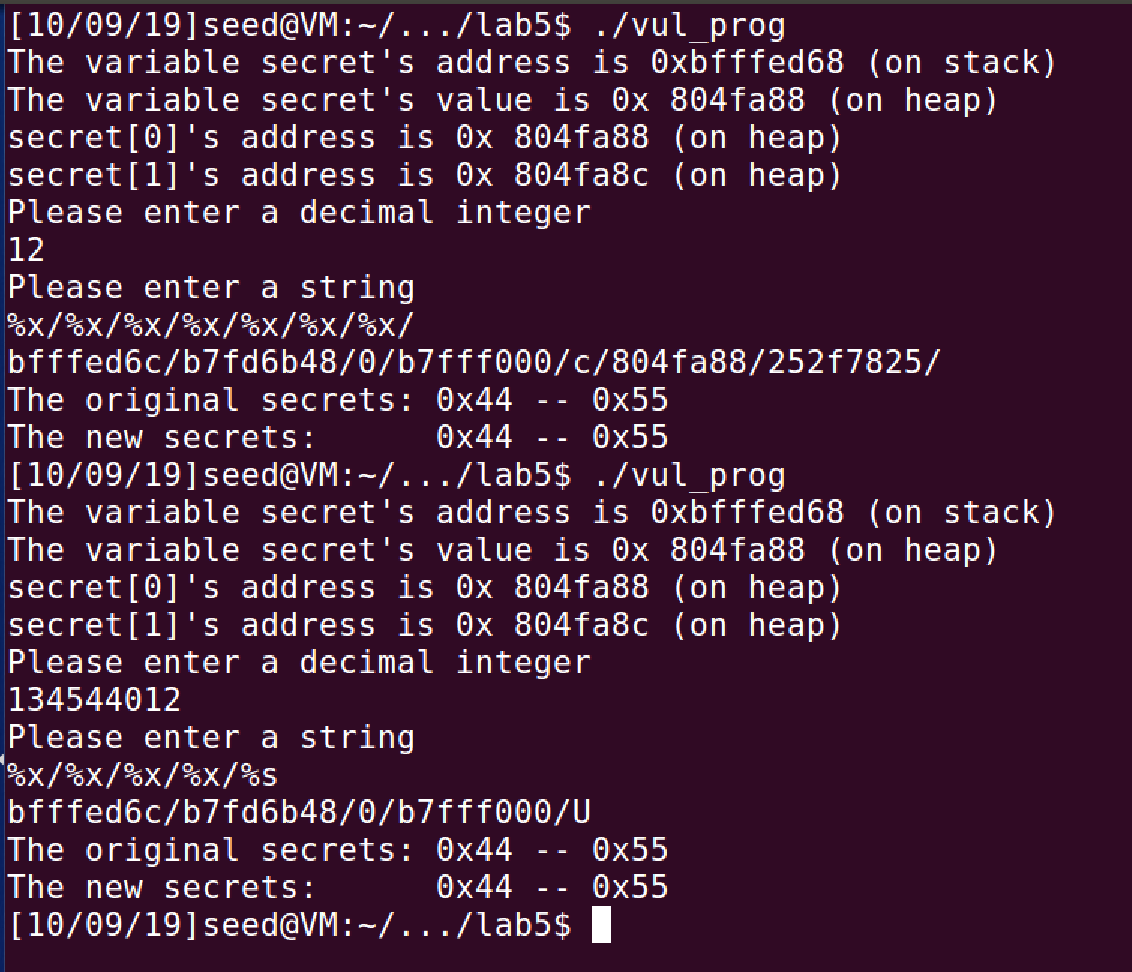
• Crash the program.

In order to crash the program, I inputed many %s.



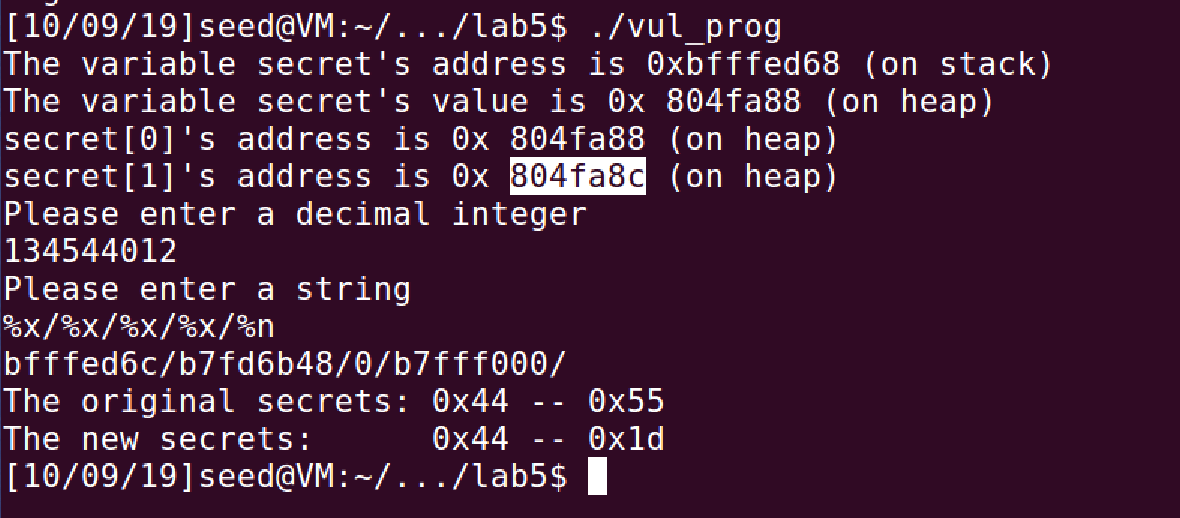
• Print out the secret[1] value.

I entered integer 12 to check that the location of the user\_input was at 4th. Then I transferred the address of secret[1] from hex to Decimal and enter to the input. At the end, I got the secret[1] value. U's ascii value is 55.



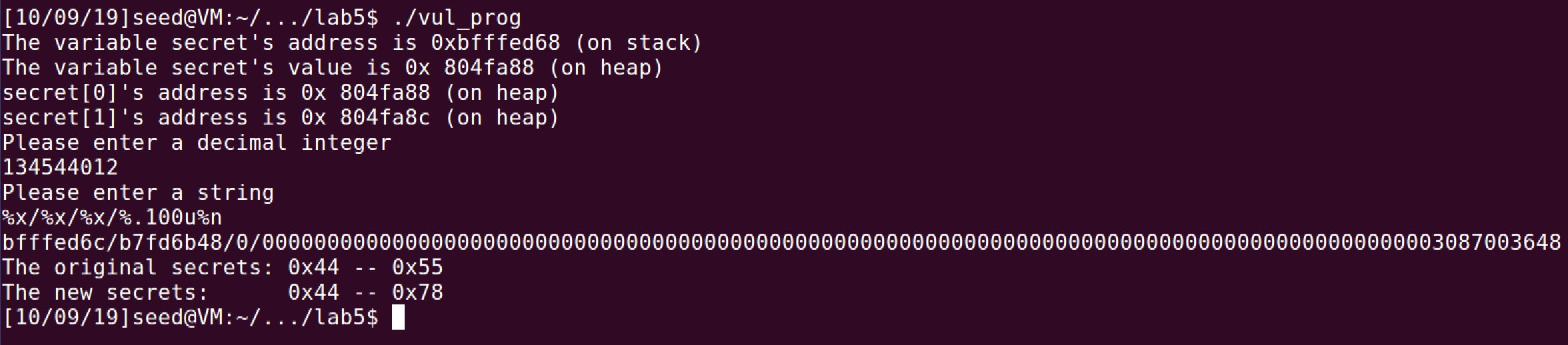
• Modify the secret[1] value.

I changed from %s to %n. It changed the address from 0x55 to 0x1d.



• Modify the secret[1] value to a pre-determined value.

I entered 100u before %n that changed the address of secret from 0x55 to 0x78.



Task 2 Memory randomization

1. When you run the program once again, will you get the same address?

No, this is because the address is changing.

1. I wrote a format string into a file called mystring first. Then using the command vul\_prog < mystring to let vul\_prog program get our address of secret and modify it.

