

CS6542 - Graduate –Computer Network Defense – Spring 2017

Lab 10-1 page 242
Due 28th February 2016, 6:00 pm.
100 points

Policy on collaboration: All examinations, papers, and other graded work products and assignments are to be completed in conformance with The George Washington University Code of Academic Integrity. Each student is expected to write his or her own HW out independently; you may not copy one another's assignments, even in part. You may not collaborate with others on the test and final.

You are expected to cite all your sources in any written work that is not closed book: papers, books, web sites, discussions with others - faculty, friends, students. For example, if, in a group, one student has a major idea that leads to a solution to a HW problem, all other students in the group should cite this student.

You may not refer to solutions to previous years' problem sets, or ask for help students from previous years. Any violations will be treated as violations of the Code of Academic Integrity.

Please work on each lab and capture screenshot of tasks along with your words and analysis of each slide. PLEASE submit all Labs on Blackboard only. Name your files:

PLEASE submit all Project on Blackboard only

Late submission

Please note that, there is a %10 penalty for late submission until next project due date, and also there is NO grade for project submission after the next project due date.

1. Lab 10-1

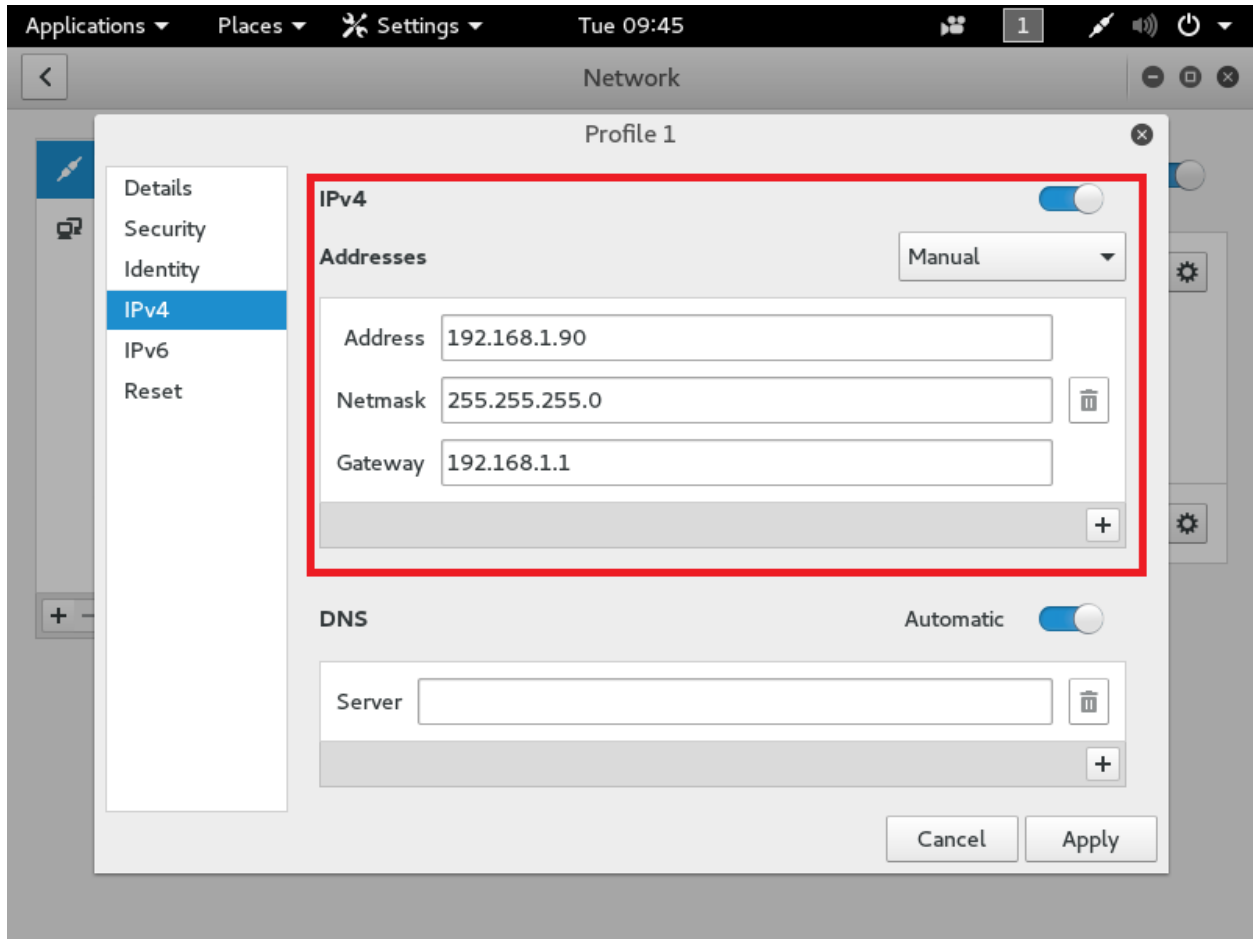
1.1. System Requirements:

Fully-updated 32-bit Kali Linux install

1.1.1. Ensure the virtual machines are configured in this configuration:

Kali Linux Instance:

- 1 Bridged Adapter:
 - IP: 192.168.1.90
 - Netmask: 255.255.255.0
 - Gateway: 192.168.1.1



Network Gateway:

- IP: 192.168.1.1
- Netmask: 255.255.255.0

For more information on the types of VMWare interfaces and the difference between Bridged, NAT, and Host Only mode, see the RedNectar description of each type of interface:

<http://rednectar.net/2011/07/20/vmware-interfaces-tutorial/>

1.1.2. Ensure the common compiler components are installed.

```
root@kali:~# apt-get install build-essential
```

CS6542 - Graduate –Computer Network Defense – Spring 2017

1.2.Tasks:

1. create meet.c file:

Please run the following command to be able to create meet.c file.

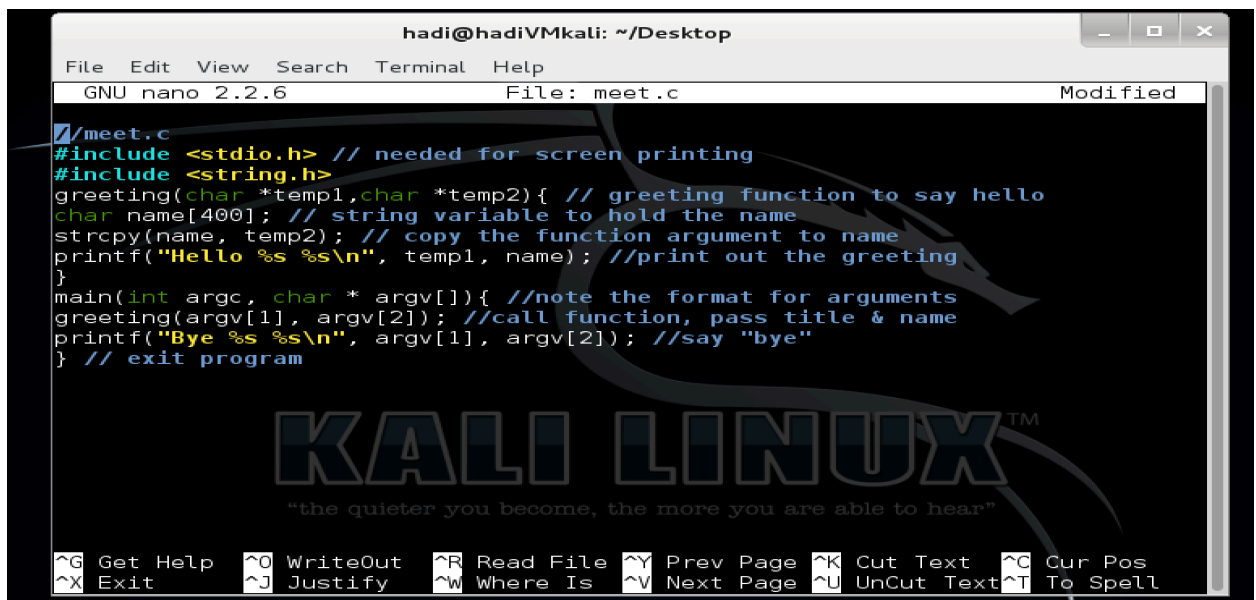
```
root@kali:# vi meet.c
```

Or you can use nano editor by using the following command:

```
hadi@hadiVMkali:~$ cd Desktop/  
hadi@hadiVMkali:~/Desktop$ nano meet.c  
hadi@hadiVMkali:~/Desktop$ ls  
meet.c  
hadi@hadiVMkali:~/Desktop$
```

Then copy and past the following:

```
//meet.c  
#include <stdio.h> // needed for screen printing  
#include <string.h>  
greeting(char *temp1,char *temp2){ // greeting function to say  
hello  
char name[400]; // string variable to hold the name  
strcpy(name, temp2); // copy the function argument to name  
printf("Hello %s %s\n", temp1, name); //print out the greeting  
}  
main(int argc, char * argv[]){ //note the format for arguments  
greeting(argv[1], argv[2]); //call function, pass title & name  
printf("Bye %s %s\n", argv[1], argv[2]); //say "bye"  
} // exit program
```



```
hadi@hadiVMkali: ~/Desktop  
File Edit View Search Terminal Help  
GNU nano 2.2.6 File: meet.c Modified  
//meet.c  
#include <stdio.h> // needed for screen printing  
#include <string.h>  
greeting(char *temp1,char *temp2){ // greeting function to say hello  
char name[400]; // string variable to hold the name  
strcpy(name, temp2); // copy the function argument to name  
printf("Hello %s %s\n", temp1, name); //print out the greeting  
}  
main(int argc, char * argv[]){ //note the format for arguments  
greeting(argv[1], argv[2]); //call function, pass title & name  
printf("Bye %s %s\n", argv[1], argv[2]); //say "bye"  
} // exit program  
KALI LINUX™  
"the quieter you become, the more you are able to hear"  
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos  
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

CS6542 - Graduate –Computer Network Defense – Spring 2017

2. Run 600 A's in meet.c file:

Running 600 "A"'s by making use of the following Perl command and printing them to the standard output to test the command:

```
hadi@hadiVMkali:~/Desktop$ perl -e 'print "A" x 600'
```

[illegible]

3. Compiling the “meet.c” file, type the command

```
root@kali:~/Desktop# gcc -o meet meet.c
```

This command will invoke the GNU C comp

```
hadi@hadiVMkali: ~/Desktop
File Edit View Search Terminal Help
hadi@hadiVMkali:~/Desktop$ gcc -o meet meet.c
hadi@hadiVMkali:~/Desktop$ ls
meet  meet.c
hadi@hadiVMkali:~/Desktop$
```

4. Testing its execution by generating through a Perl command 10 “A”s:

```
hadi@hadiVMkali:~/Desktop$ ./meet Mr `perl -e 'print "A" x 10`
```

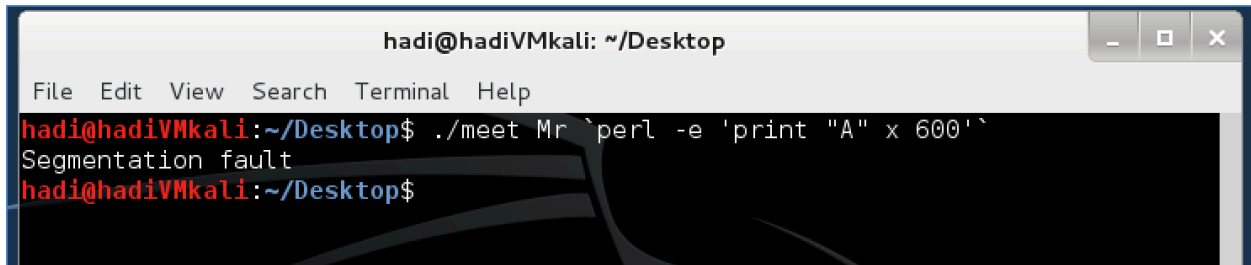


The screenshot shows a terminal window titled "hadi@hadiVMkali: ~/Desktop". The menu bar includes "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal content shows the user running the command `./meet Mr `perl -e 'print "A" x 10``, which results in the output "Hello Mr AAAAAAAAAA" followed by "Bye Mr AAAAAAAAAA". The prompt `hadi@hadiVMkali:~/Desktop$` is visible at the bottom.

CS6542 - Graduate –Computer Network Defense – Spring 2017

5. Overflow meet.c with 600 A's and get "segmentation fault" comment on your terminal.

```
root@kali:~/Desktop# ./meet Mr `perl -e 'print "A" x 600`
```

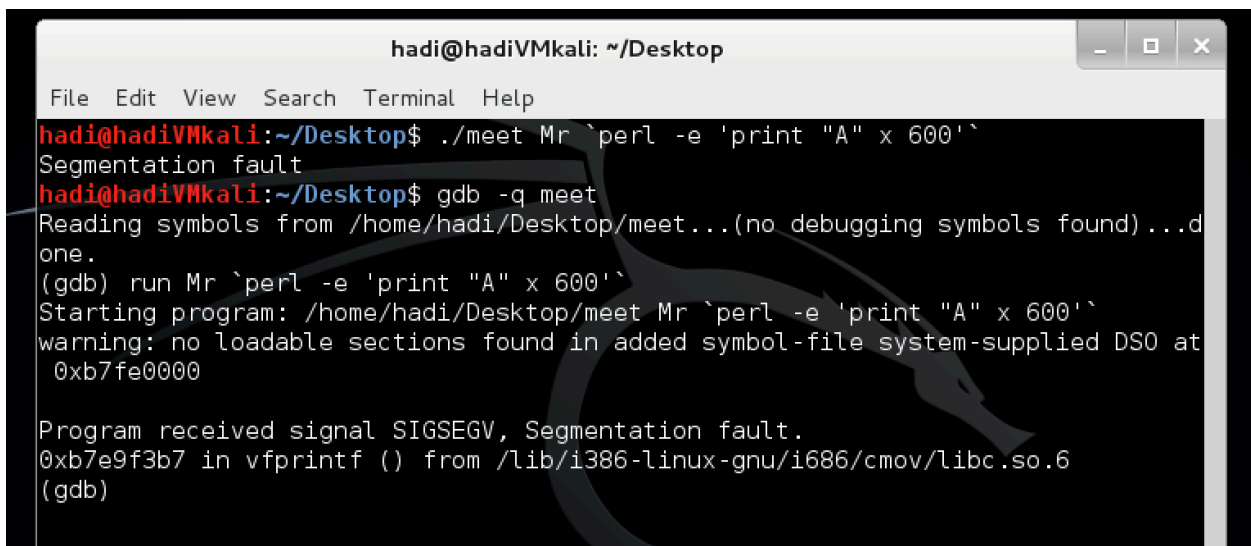
A screenshot of a terminal window titled 'hadi@hadiVMkali: ~/Desktop'. The terminal shows the command './meet Mr `perl -e 'print "A" x 600`' being executed, which results in a 'Segmentation fault' message. The prompt then returns to 'hadi@hadiVMkali:~/Desktop\$'.

6. Verification by gdb compiler

```
root@kali:~/Desktop# gdb -q meet
```

Then your kali is waiting you enter the following command to defub

```
(gdb) run Mr `perl -e 'print "A" x 600`
```

A screenshot of a terminal window titled 'hadi@hadiVMkali: ~/Desktop'. The terminal shows the command './meet Mr `perl -e 'print "A" x 600`' being executed, which results in a 'Segmentation fault'. Then, the command 'gdb -q meet' is entered, and the terminal shows the gdb prompt '(gdb)'. The command 'run Mr `perl -e 'print "A" x 600`' is entered, and the terminal shows the message 'Starting program: /home/hadi/Desktop/meet Mr `perl -e 'print "A" x 600`''. The terminal then shows the message 'Program received signal SIGSEGV, Segmentation fault. 0xb7e9f3b7 in vfprintf () from /lib/i386-linux-gnu/i686/cmov/libc.so.6'. The prompt then returns to '(gdb)'.

Please take the screen shot of the result:

NOTE: Your values will be different—it is the concept we are trying to get across here, not the memory values.

7. Please screenshot of your current value of your "eip" for this process.