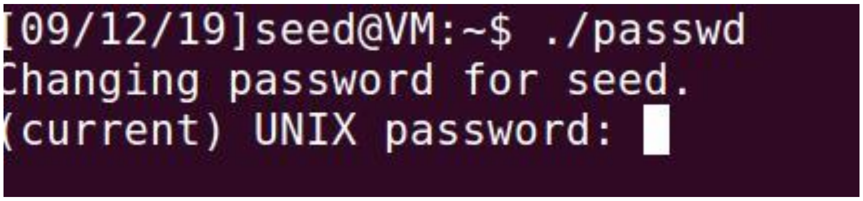
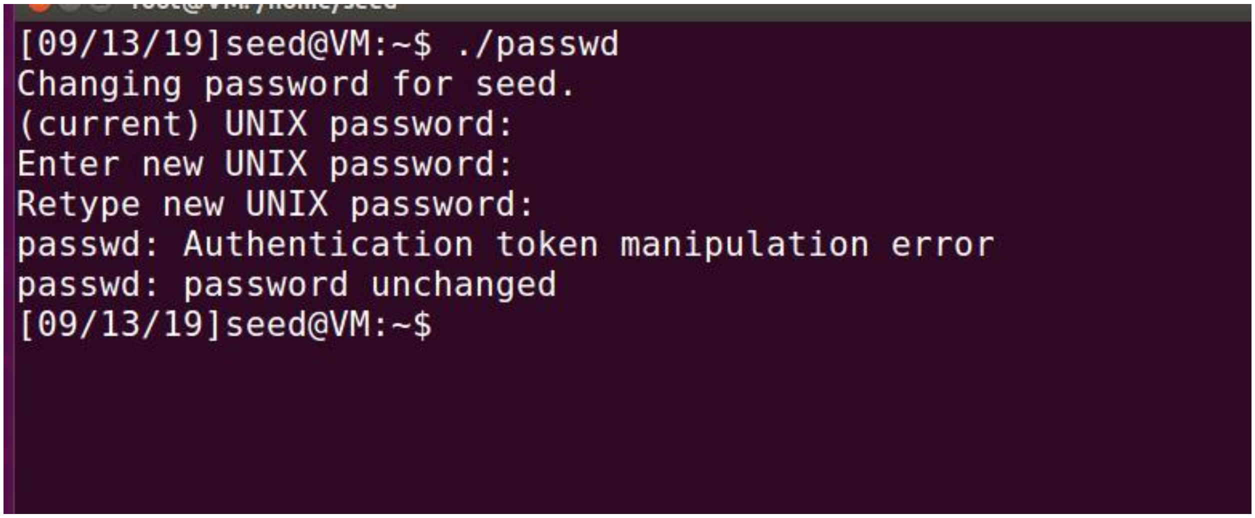
Lab1 - Set-UID Program Vulnerability Lab

Name: Ravi Teja Thota

1. As passwd program is only accessible to root. The passwd program’s privileges are changed to setuid by root so that the User will be having root access for temporary purpose.

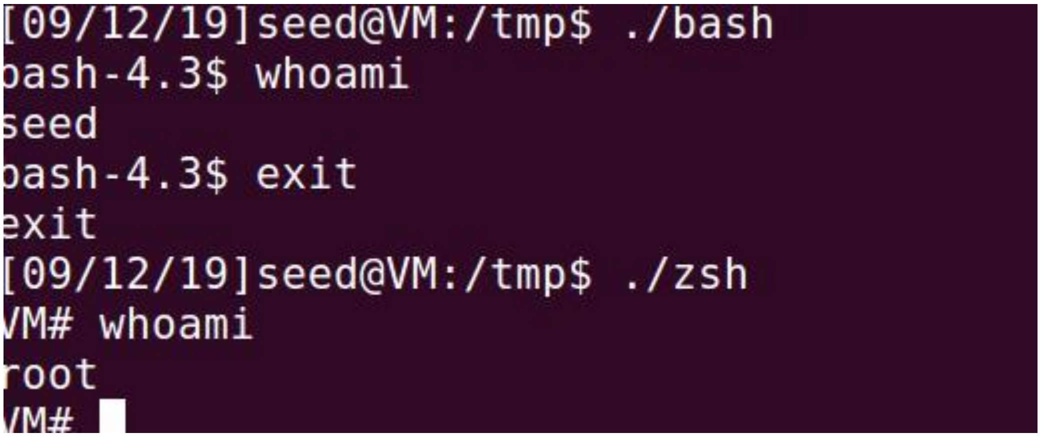


```
[09/12/19]seed@VM:~$ ./passwd
Changing password for seed.
(current) UNIX password: 
```



```
[09/13/19]seed@VM:~$ ./passwd
Changing password for seed.
(current) UNIX password:
Enter new UNIX password:
Retype new UNIX password:
passwd: Authentication token manipulation error
passwd: password unchanged
[09/13/19]seed@VM:~$
```

2. I copied Bash and ZSH programs from root to seed and gave them both setuid privileges logging into to root account. The zsh program gave me root access but the bash program didn't allow me root access due to its self-defense mechanism.



```
[09/12/19]seed@VM:/tmp$ ./bash
bash-4.3$ whoami
seed
bash-4.3$ exit
exit
[09/12/19]seed@VM:/tmp$ ./zsh
VM# whoami
root
VM# 
```

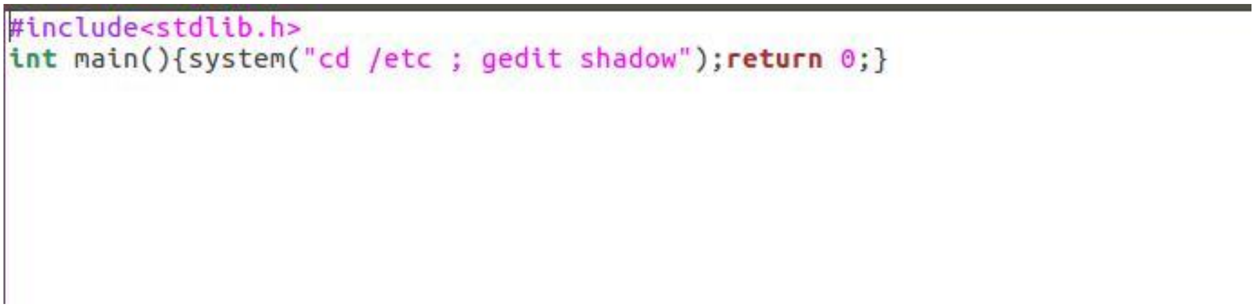
3.

I copied the following program and gave it Setuid privileges to list all the programs to in present directory.

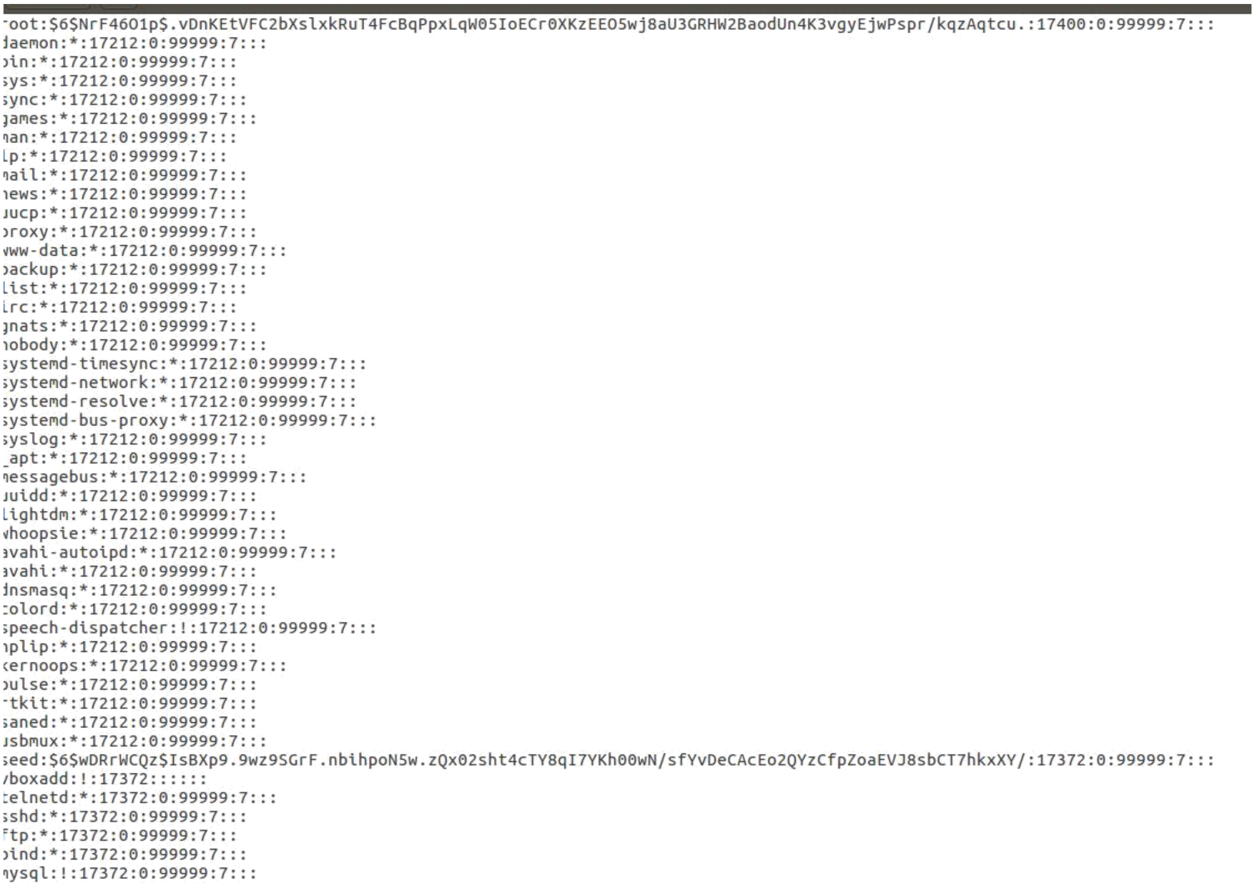
```
int main()

{
system("ls");
return 0;
}
```

As the task suggested I made changes to the program so that i can view the data present in etc/shadow. I compiled the program giving it setuid privileges and run the program. The following screenshots are the outputs.



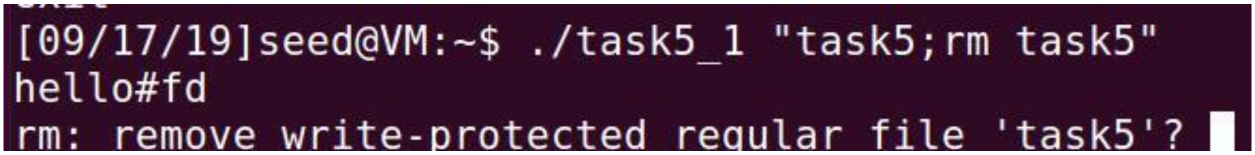
```
#include<stdlib.h>
int main(){system("cd /etc ; gedit shadow");return 0;}
```



```
oot:$6$NrF46O1p$.vDnKEtVFC2bXslxkRuT4FcBqPpxLqW05IoECr0XKzEEO5wj8aU3GRHW2BaodUn4K3vgyEjwPspr/kqzAqtcu.:17400:0:99999:7:::
aemon:*:17212:0:99999:7:::
in:*:17212:0:99999:7:::
ys:*:17212:0:99999:7:::
ync:*:17212:0:99999:7:::
ames:*:17212:0:99999:7:::
an:*:17212:0:99999:7:::
p:*:17212:0:99999:7:::
ail:*:17212:0:99999:7:::
ews:*:17212:0:99999:7:::
ucp:*:17212:0:99999:7:::
roxy:*:17212:0:99999:7:::
ww-data:*:17212:0:99999:7:::
ackup:*:17212:0:99999:7:::
ist:*:17212:0:99999:7:::
rc:*:17212:0:99999:7:::
nats:*:17212:0:99999:7:::
obody:*:17212:0:99999:7:::
ystemd-timesync:*:17212:0:99999:7:::
ystemd-network:*:17212:0:99999:7:::
ystemd-resolve:*:17212:0:99999:7:::
ystemd-bus-proxy:*:17212:0:99999:7:::
yslog:*:17212:0:99999:7:::
apt:*:17212:0:99999:7:::
essagebus:*:17212:0:99999:7:::
uidd:*:17212:0:99999:7:::
ightdm:*:17212:0:99999:7:::
hoopsie:*:17212:0:99999:7:::
vahi-autoipd:*:17212:0:99999:7:::
vahi:*:17212:0:99999:7:::
nsmasq:*:17212:0:99999:7:::
olord:*:17212:0:99999:7:::
peech-dispatcher:!:17212:0:99999:7:::
plip:*:17212:0:99999:7:::
ernoops:*:17212:0:99999:7:::
ulse:*:17212:0:99999:7:::
tkit:*:17212:0:99999:7:::
aned:*:17212:0:99999:7:::
sbmux:*:17212:0:99999:7:::
eed:$6$wDRrWCQz$IsBXp9.9wz9SGrF.nbihpoN5w.zQx02sht4cTY8qI7YKh00wN/sfYvDeCAcEo2QYzCfpZoaEVJ8sbCT7hkxXY/:17372:0:99999:7:::
boxadd:!:17372::::::
elnetd:*:17372:0:99999:7:::
shd:*:17372:0:99999:7:::
tp:*:17372:0:99999:7:::
ind:*:17372:0:99999:7:::
ysql:!:17372:0:99999:7:::
```

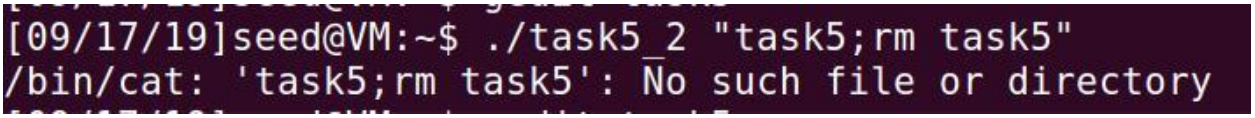
5. **Observation of the task** The System () allows us to execute multiple commands in a single attempt as it can sort out multiple commands, whereas execve() doesn't allow us to execute multiple commands as it doesn't have ability to execute multiple commands in a single attempt.

The below screen shot is when the system executes the program in system() command



```
[09/17/19]seed@VM:~$ ./task5_1 "task5;rm task5"
hello#fd
rm: remove write-protected regular file 'task5'?
```

The below screenshot is when the system execute the program in execve() command.



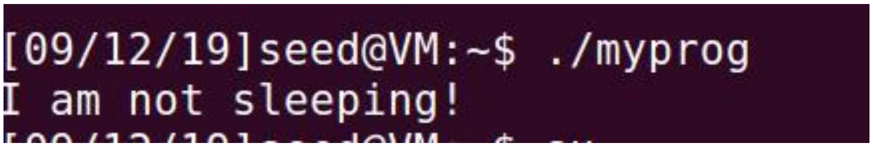
```
[09/17/19]seed@VM:~$ ./task5_2 "task5;rm task5"
/bin/cat: 'task5;rm task5': No such file or directory
```

6.

The given program in the lab has been executed and compiled.

```
export LD_PRELOAD=./libmylib.so.1.0.1
```

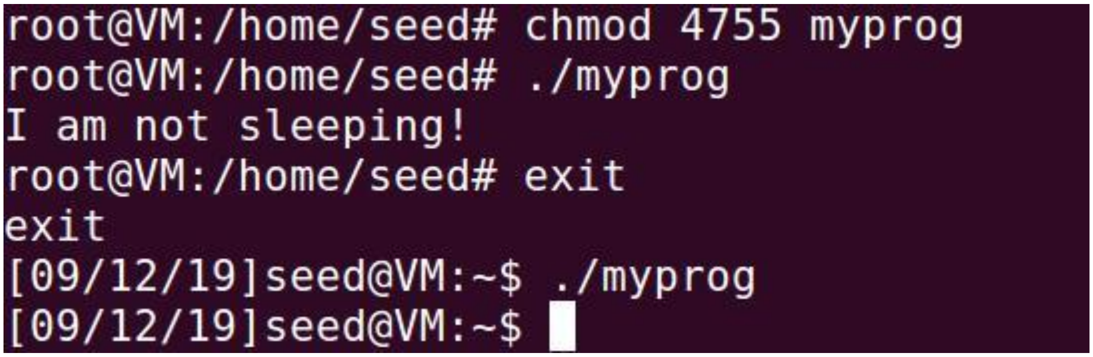
following Command is executed to preload the library. Then compiled the program and gave it setuid privileges and came back to seed and executed the executable.



```
[09/12/19]seed@VM:~$ ./myprog
I am not sleeping!
[09/12/19]seed@VM:~$ su
```

Now again I went to root and preloaded the library and executed the executable and output is shown in below screenshot. Now I exit root and execute the same executable is the seed account without preload the library and i observed that as there is no preloaded library we have no output as shown in below picture.

The runtime linker will ignore the LD_PRELOAD environment variable when the seed
Account uses the Set-UID program, myprog, because the effective user is seed. The library interposition will not work unless the library is actually preloaded each time, the
Elevated privilege belongs to the root which also does not have the libmylib preloaded



```
root@VM:/home/seed# chmod 4755 myprog
root@VM:/home/seed# ./myprog
I am not sleeping!
root@VM:/home/seed# exit
exit
[09/12/19]seed@VM:~$ ./myprog
[09/12/19]seed@VM:~$ 
```