Sanskriti Singh
2001CS60

CS392 Assignment-1

Q1

First copy the ‘passwd’ to another directory (say */tmp/*)

```
(base) sanskriti@sans-ubuntu:~$ which passwd
/usr/bin/passwd
(base) sanskriti@sans-ubuntu:~$ ls -al /usr/bin/passwd
-rwsr-xr-x 1 root root 59976 Nov 24 17:35 /usr/bin/passwd
(base) sanskriti@sans-ubuntu:~$ cp /usr/bin/passwd /tmp/
(base) sanskriti@sans-ubuntu:~$ ls -al /tmp/passwd
-rwxr-xr-x 1 sanskriti sanskriti 59976 Jan 24 19:33 /tmp/passwd
(base) sanskriti@sans-ubuntu:~$
```

Then execute the ‘passwd’ file

```
(base) sanskriti@sans-ubuntu:~$ cd /tmp/
(base) sanskriti@sans-ubuntu:/tmp$ ./passwd
Changing password for sanskriti.
Current password:
New password:
Retype new password:
passwd: Authentication token manipulation error
passwd: password unchanged
(base) sanskriti@sans-ubuntu:/tmp$
```

Password is unchanged

Q1 Doing the same with ‘chsh’ file

```
(base) sanskriti@sans-ubuntu:/tmp$ which chsh
/usr/bin/chsh
(base) sanskriti@sans-ubuntu:/tmp$ ls -al /usr/bin/chsh
-rwsr-xr-x 1 root root 44808 Nov 24 17:35 /usr/bin/chsh
(base) sanskriti@sans-ubuntu:/tmp$ cp /usr/bin/chsh /tmp/
(base) sanskriti@sans-ubuntu:/tmp$ ls -al /tmp/chsh
-rwxr-xr-x 1 sanskriti sanskriti 44808 Jan 24 19:50 /tmp/chsh
(base) sanskriti@sans-ubuntu:/tmp$ ./chsh
Password:
Changing the login shell for sanskriti
Enter the new value, or press ENTER for the default
        Login Shell [/bin/bash]:
Cannot change ID to root.
(base) sanskriti@sans-ubuntu:/tmp$
```

Cannot change ID to root

If we had the root privilege, we would be able to make changes to the password or even to ID. Since the output as shown in these images give error messages, it’s clear that there are no root privileges.

Q2-a

Copying and running ‘bin/zsh’ as normal user

```
(base) sanskriti@sans-ubuntu:~$ cd /tmp/
(base) sanskriti@sans-ubuntu:/tmp$ sudo su
root@sans-ubuntu:/tmp# cp /usr/bin/zsh /tmp/
root@sans-ubuntu:/tmp# chmod u+s zsh
root@sans-ubuntu:/tmp# ls -al zsh
-rwsr-xr-x 1 root root 1013328 Jan 24 21:31 zsh
root@sans-ubuntu:/tmp# exit
exit
(base) sanskriti@sans-ubuntu:/tmp$ ./zsh
sans-ubuntu# id
uid=1000(sanskriti) gid=1000(sanskriti) euid=0(root) groups=1000(sanskriti),4(ad
m),24(cdrom),27(sudo),30(dip),46(plugdev),120(lpadmin),132(lxd),133(sambashare)
sans-ubuntu#
```

Now, normal user gets root privilege

Q2-b

Copying and running ‘bin/bash’ as normal user

```
(base) sanskriti@sans-ubuntu:~$ cd /tmp/
(base) sanskriti@sans-ubuntu:/tmp$ sudo su
[sudo] password for sanskriti:
root@sans-ubuntu:/tmp# cp /bin/bash /tmp/
root@sans-ubuntu:/tmp# chmod u+s bash
root@sans-ubuntu:/tmp# exit
exit
(base) sanskriti@sans-ubuntu:/tmp$ ls -al bash
-rwsr-xr-x 1 root root 1396520 Jan 24 21:37 bash
(base) sanskriti@sans-ubuntu:/tmp$ ./bash
bash-5.1$ id
uid=1000(sanskriti) gid=1000(sanskriti) groups=1000(sanskriti),4(adm),24(cdrom),
27(sudo),30(dip),46(plugdev),120(lpadmin),132(lxd),133(sambashare)
bash-5.1$
```

Here, normal user does not get root privilege

Q3

Using system()

```
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$ ls -a /bin/sh
lrwxrwxrwx 1 root root 4 Mar 23  2022 /bin/sh -> dash
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$ sudo -u
[sudo] password for sanskriti:
root@sans-ubuntu:/home/sanskriti/CS392_SSD/Assign1# gcc -o Assign1a Assign1a.c
root@sans-ubuntu:/home/sanskriti/CS392_SSD/Assign1# chmod u+s Assign1a
root@sans-ubuntu:/home/sanskriti/CS392_SSD/Assign1# exit
exit
```

```
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$ ls -al file Assign1a
-rwsr-xr-x 1 root      root      16184 Jan 24 22:04 Assign1a
-rw-rw-r-- 1 sanskriti sanskriti    12 Jan 24 22:06 file
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$ ./Assign1a "file;mv file file_
new"
hello worls
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$ ls file*
file_new
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$
```

‘Assign1a’ is not safe as Anil now can read, write or move files which should be only be allowed to root users

Q3

Using execve()

```
(base) sanskriti@sans-ubuntu:~$ cd 'CS392_SSD/Assign1/'
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$ sudo su
[sudo] password for sanskriti:
root@sans-ubuntu:/home/sanskriti/CS392_SSD/Assign1# gcc -o Assign1b Assign1b.c
Assign1b.c: In function ‘main’:
Assign1b.c:17:5: warning: implicit declaration of function ‘execve’ [-Wimplicit-
function-declaration]
   17 |     execve(v[0], v, 0);
      |     ^~~~~~
root@sans-ubuntu:/home/sanskriti/CS392_SSD/Assign1# chmod u+s Assign1b
root@sans-ubuntu:/home/sanskriti/CS392_SSD/Assign1# exit
exit
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$ ./Assign1b "file;mv file file_
new_new
/bin/cat: 'file;mv file file_new_new': No such file or directory
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$ ls file*
file_new
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$
```

The attack by normal user (here, Anil) is ineffective

Q3

Explanation:

The reason why the before attack is effective in case of system() because system() call ‘/bin/sh’, which links ‘dash’.

```
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$ ls -al /bin/sh
lrwxrwxrwx 1 root root 4 Mar 23  2022 /bin/sh -> dash
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$ sudo su
[sudo] password for sanskriti:
```

After running cat file with root privilege, it runs “mv file file_new”.

But in the case of execve(), it will regard “file;mv file file_new_new” as a folder name, so system will prompt there have no the file.

```
exit
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$ ./Assign1b "file;mv file file_
new_new"
/bin/cat: 'file;mv file file_new_new': No such file or directory
(base) sanskriti@sans-ubuntu:~/CS392_SSD/Assign1$ ls file*
file_new
```