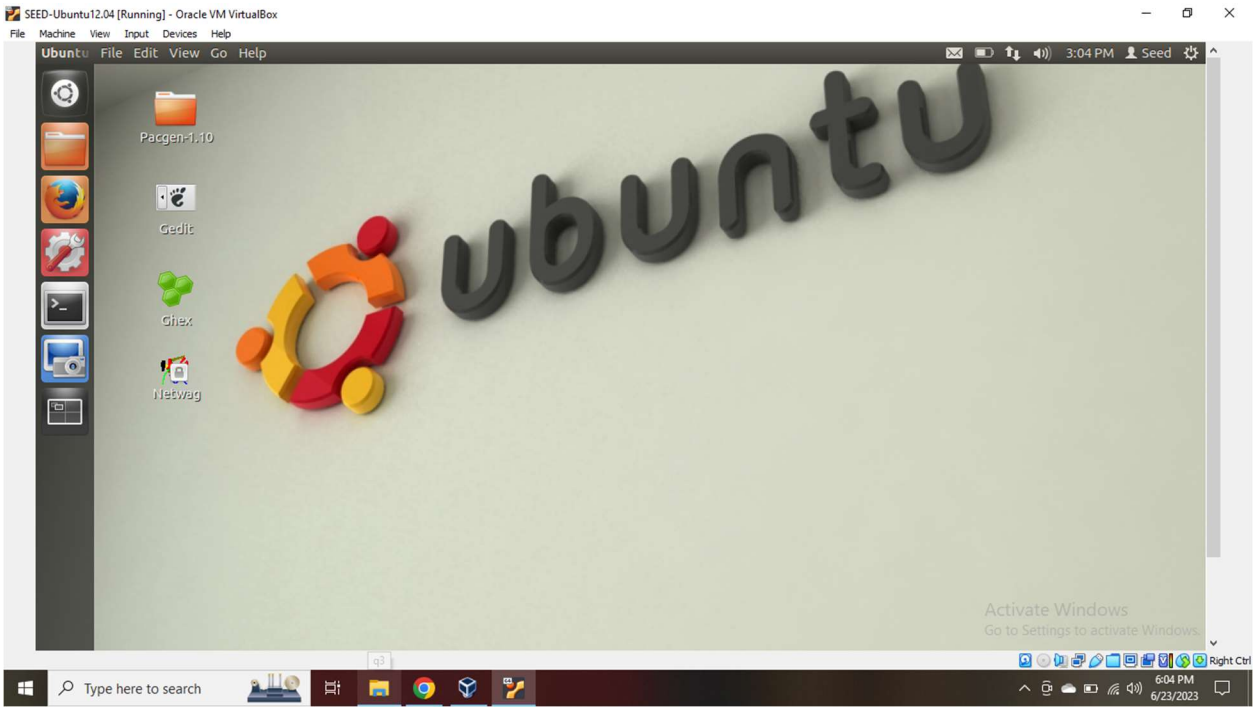
Information and Networking Security

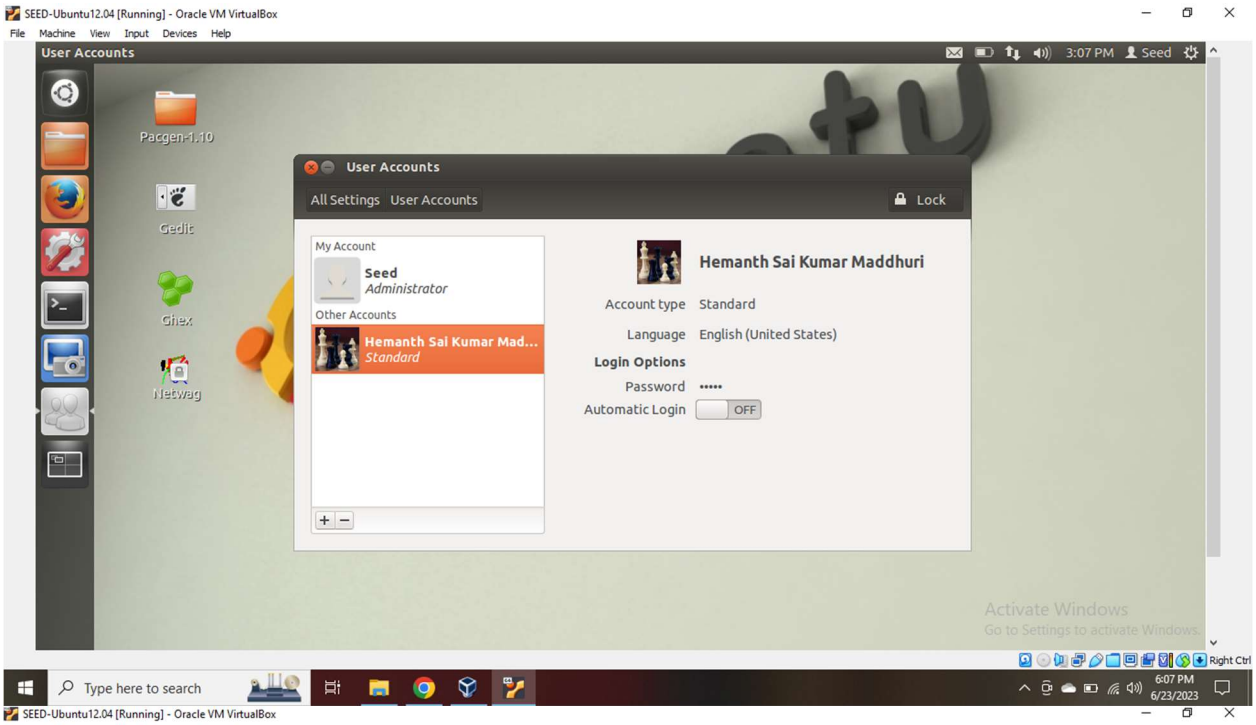
Quiz - 3

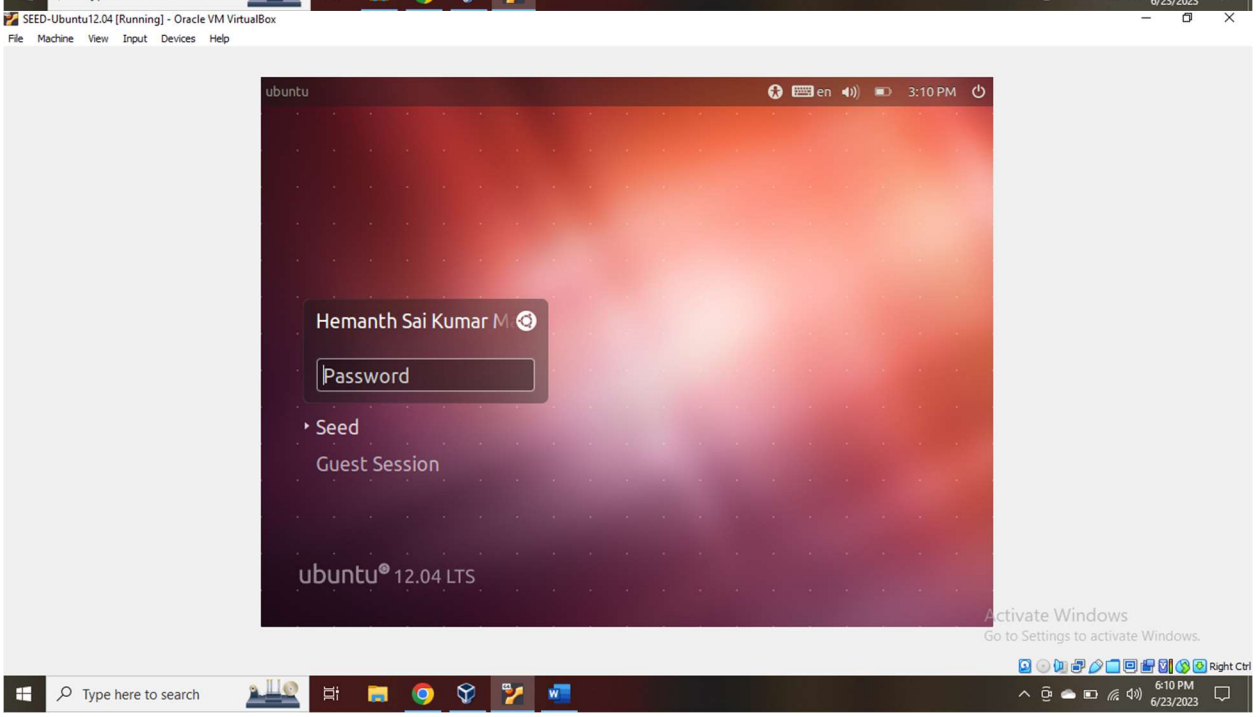
Name: Hemanth Sai Kumar Maddhuri **ID:** 999902480

As instructed in the manual we need to perform Dirty Cow Attack on Seed Labs Ubuntu 12.04. So, I have installed Ubuntu 12.04 on VirtualBox VMware from the following link https://seed.nyc3.cdn.digitaloceanspaces.com/SEEDUbuntu12.04.zip.

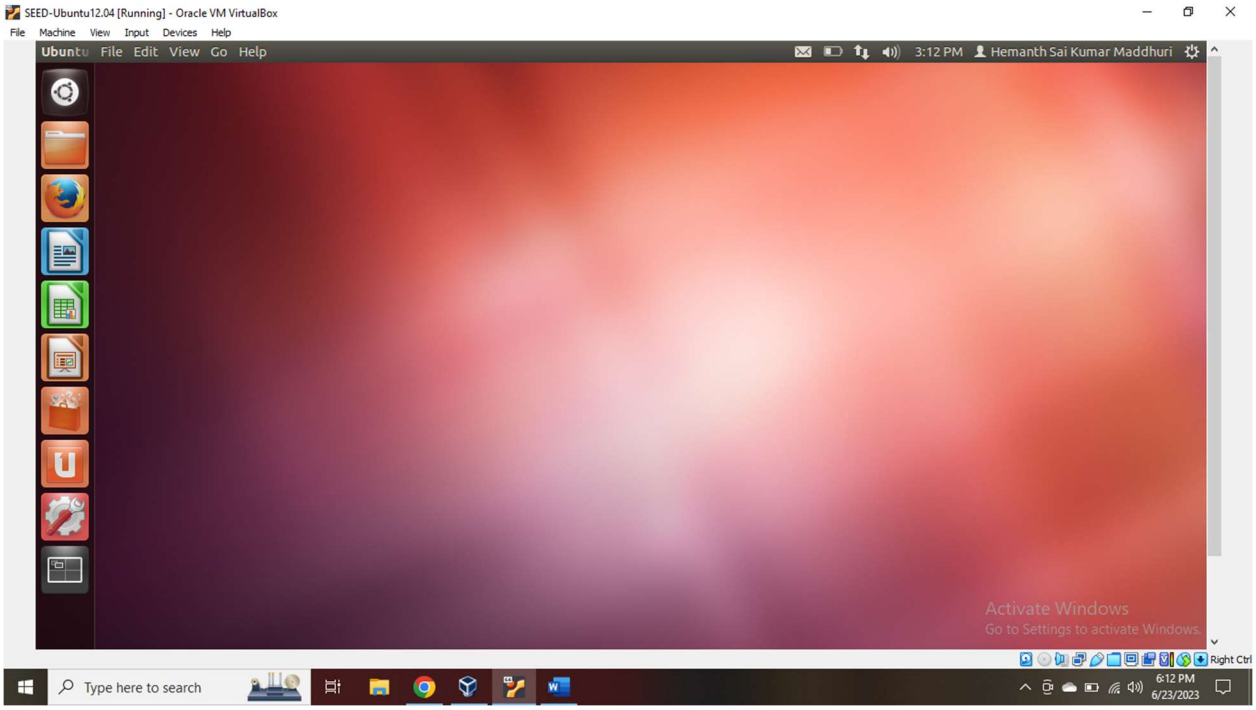


Then I set my username which has first+lastname.

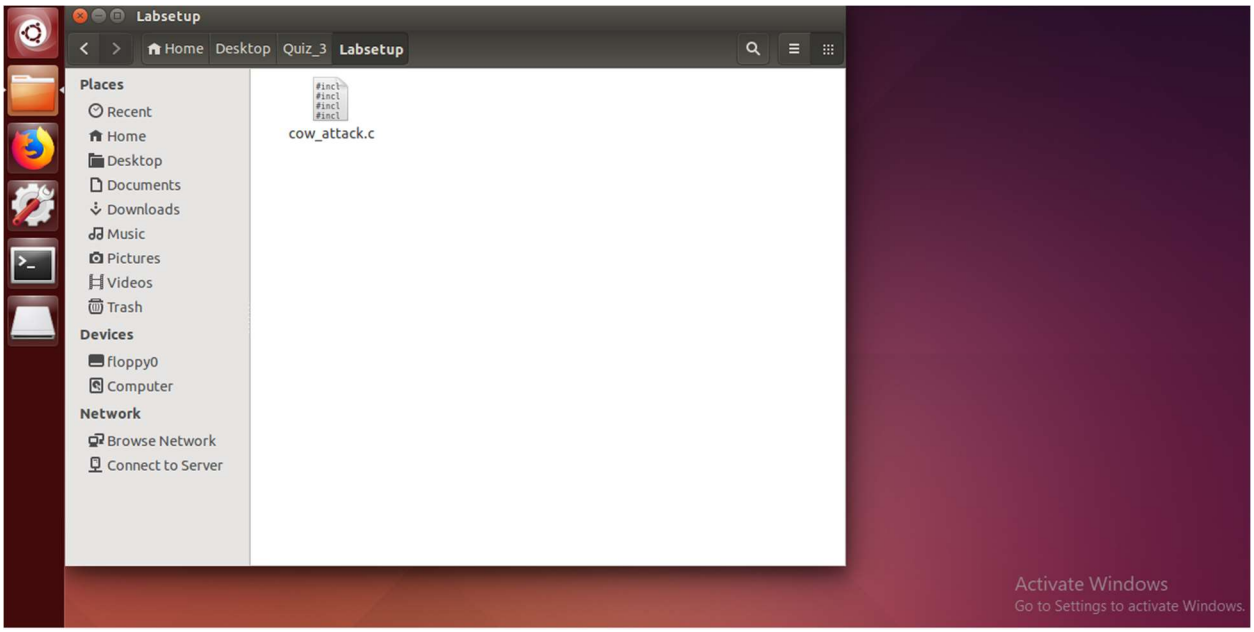




Which eventually loads like this after signing up,

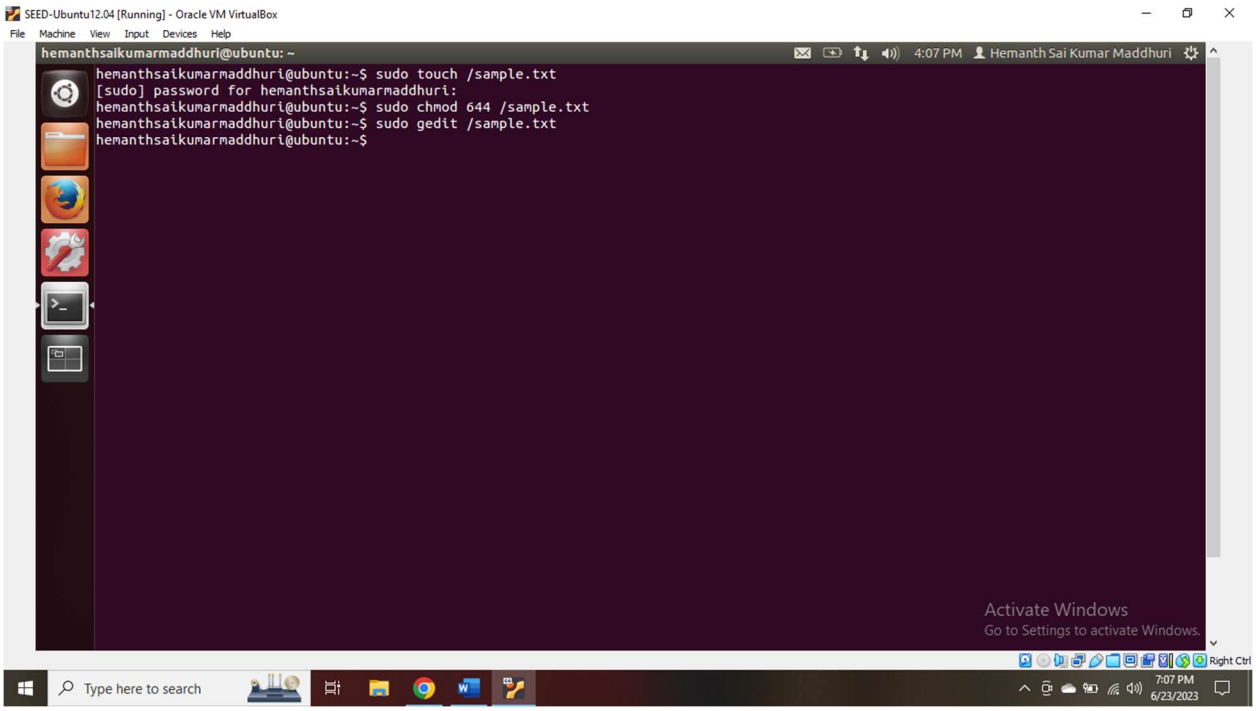


Then I downloaded the Labsetup file which contains DirtyCowAttack.c file from the link https://seedsecuritylabs.org/Labs_20.04/Files/Dirty_COW/Labsetup.zip.

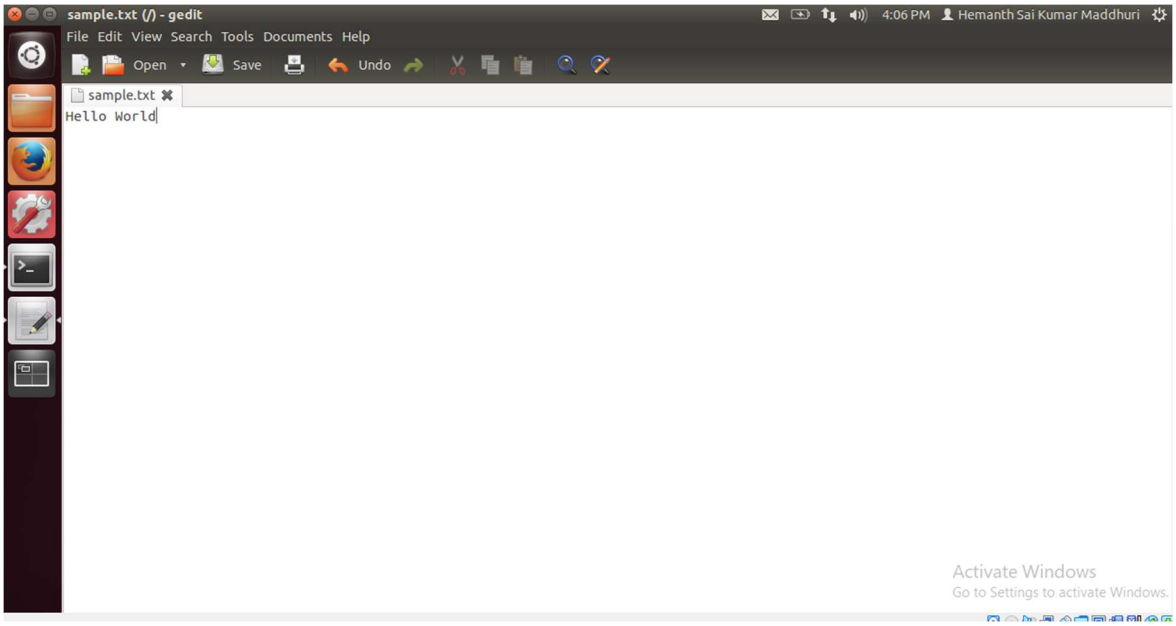


2 Task 1: Modify a Dummy Read-Only File:

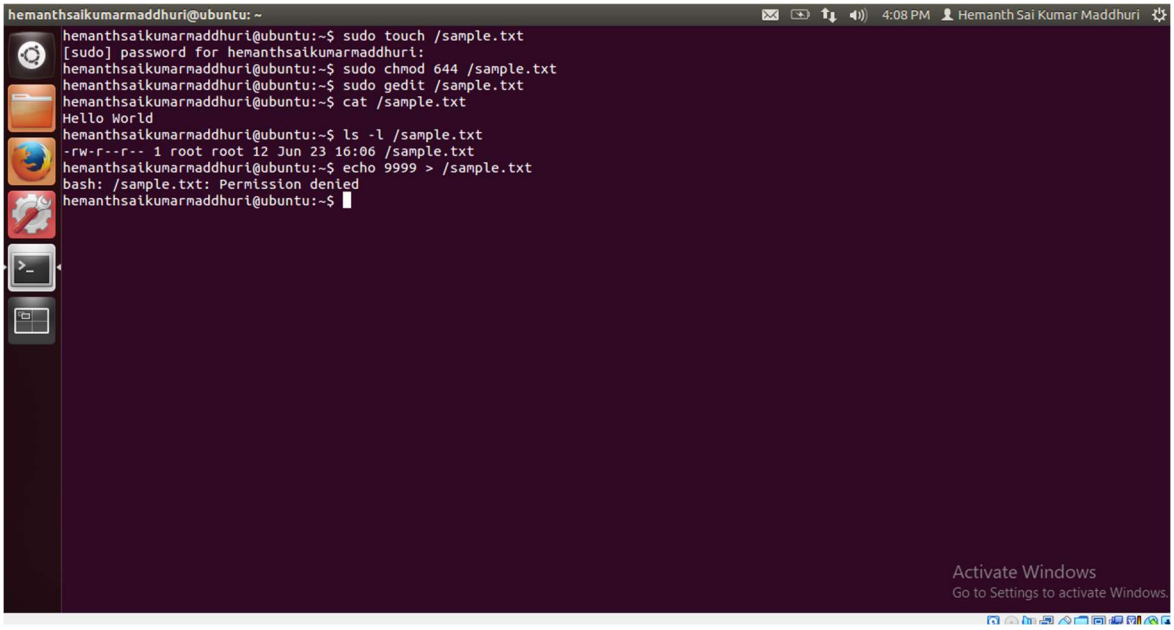
Creating a Dummy file named **“sample.txt”** in root directory. As per the given instructions the permission of the file is modified and is opened to write something.



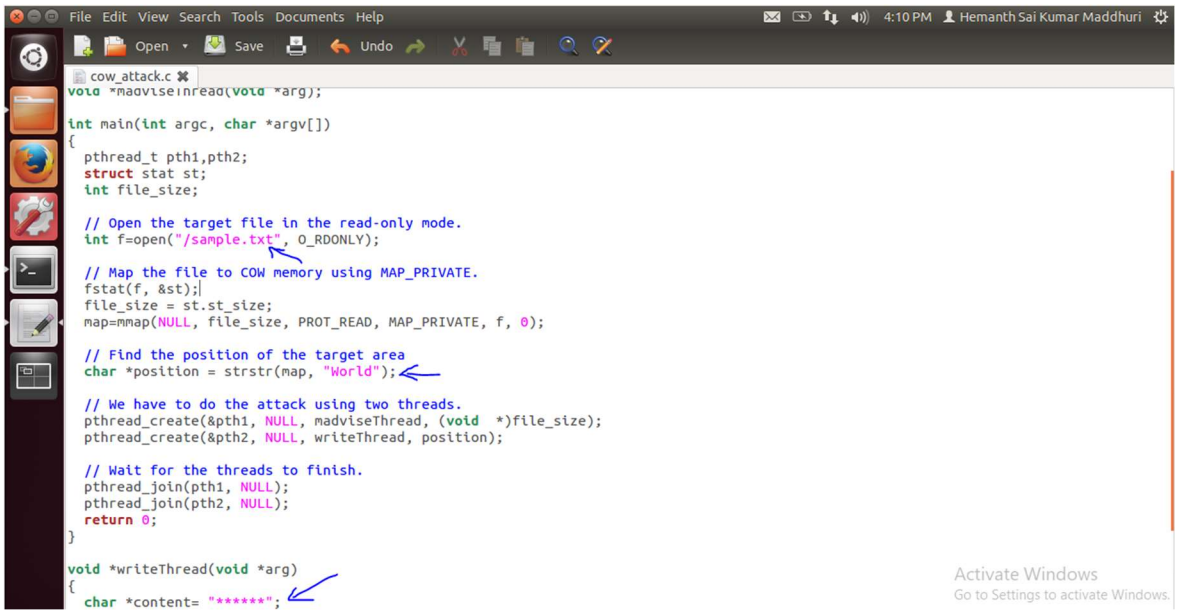
The dummy file **“sample.txt”** is opened using **“gedit”** and **“Hello World”** input is written to it.



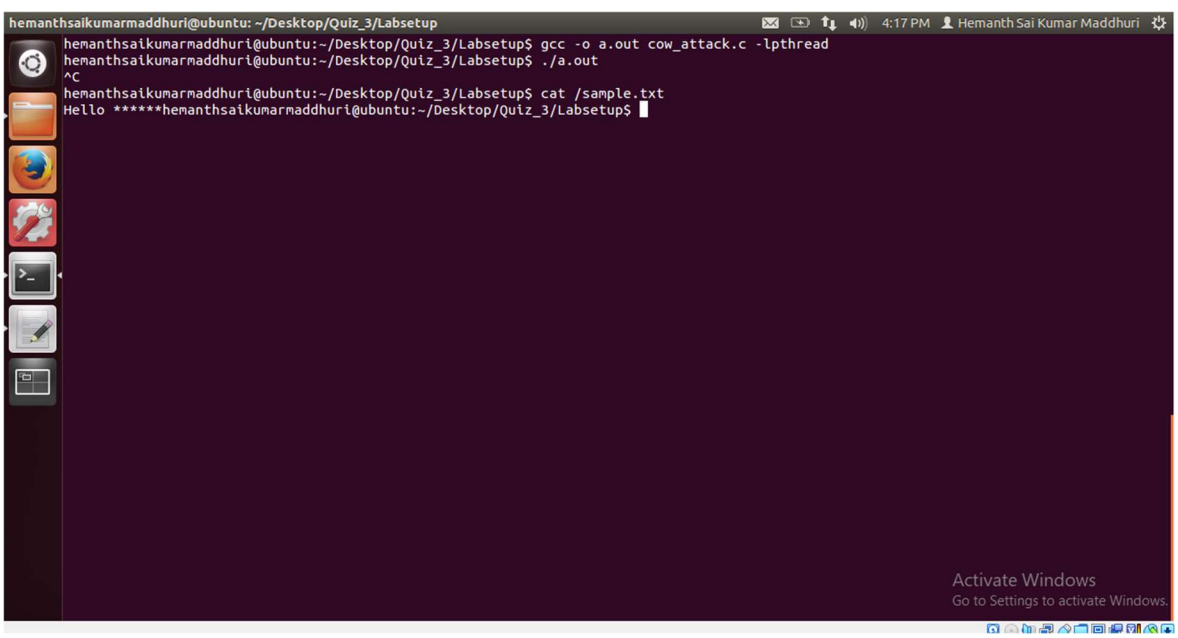
Later we list the dummy file using command **“ls -l /sample.txt”** and try to feed some input using command **“echo”** but it fails as we only have read permission.



Then we edit the main thread in attack file to open **"/sample.txt"** file and the string function as **"World"** as we are trying to replace the word **"World"** with **"*******"**.

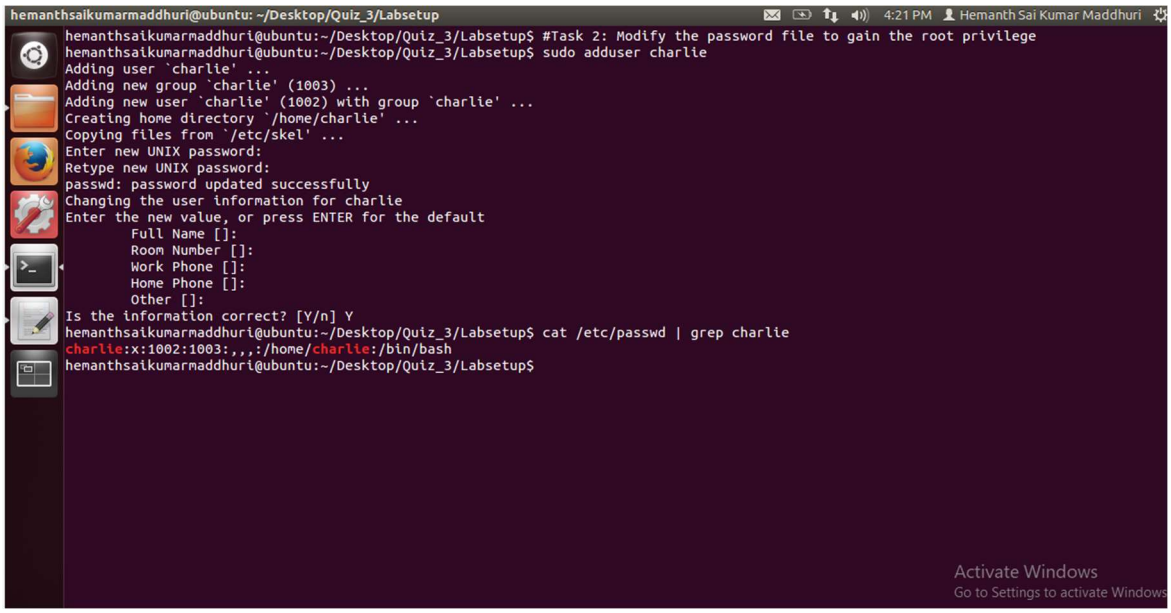


Then we compile and execute the attack file using command **"gcc -o a.out cow_attack.c -lpthread"**. Here **-lpthread"** is the option used to manage the threads to run in loop. After compilation of the attack file, we run **"a.out"** using command **"./a.out"**. As we can see that **"/sample.txt"** has been modified to **"Hello ******"** after the attack.

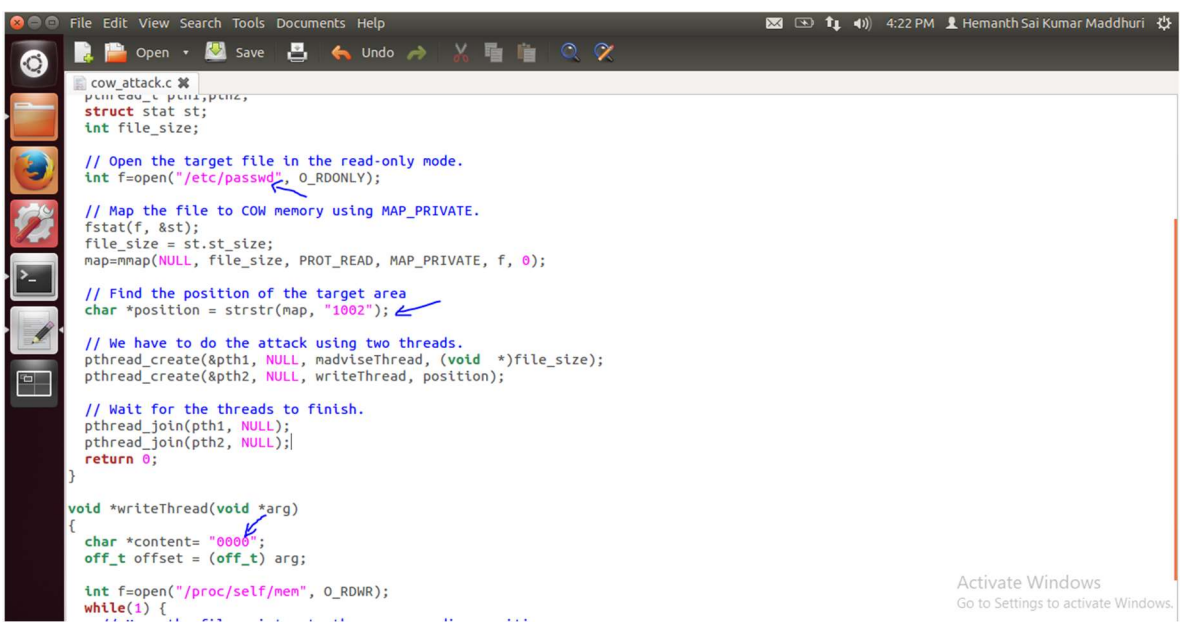


3 Task 2: Modify the Password File to Gain the Root Privilege:

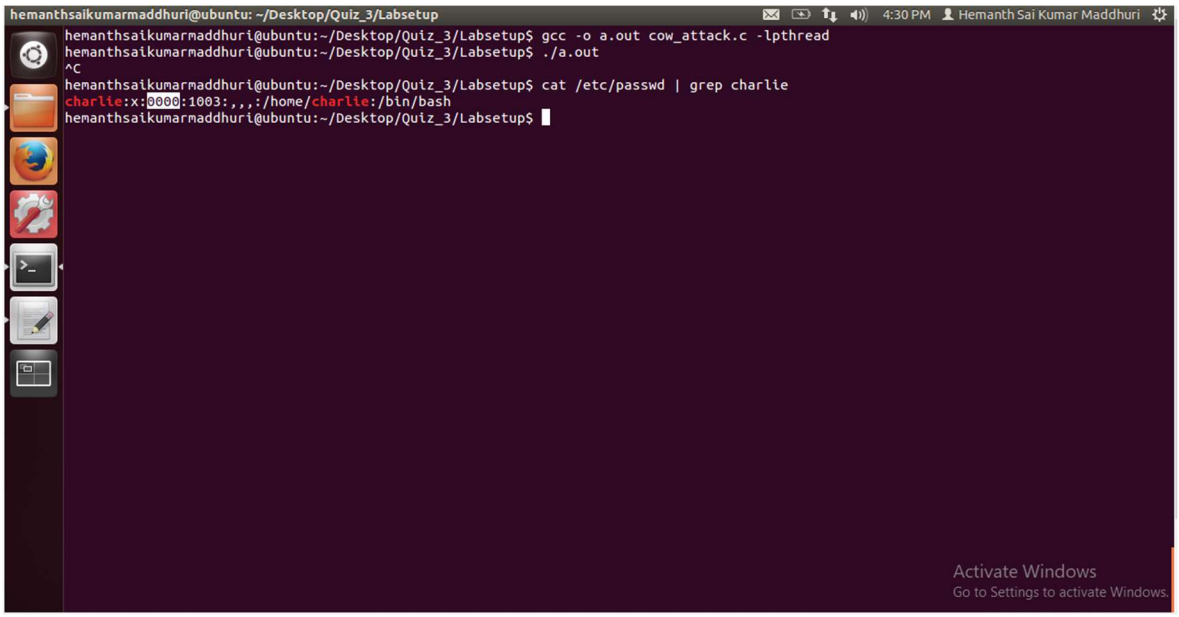
As instructed, we added a new user named **"Charlie"**. Then we check the details of the added user and find that his **UID is 1002**.



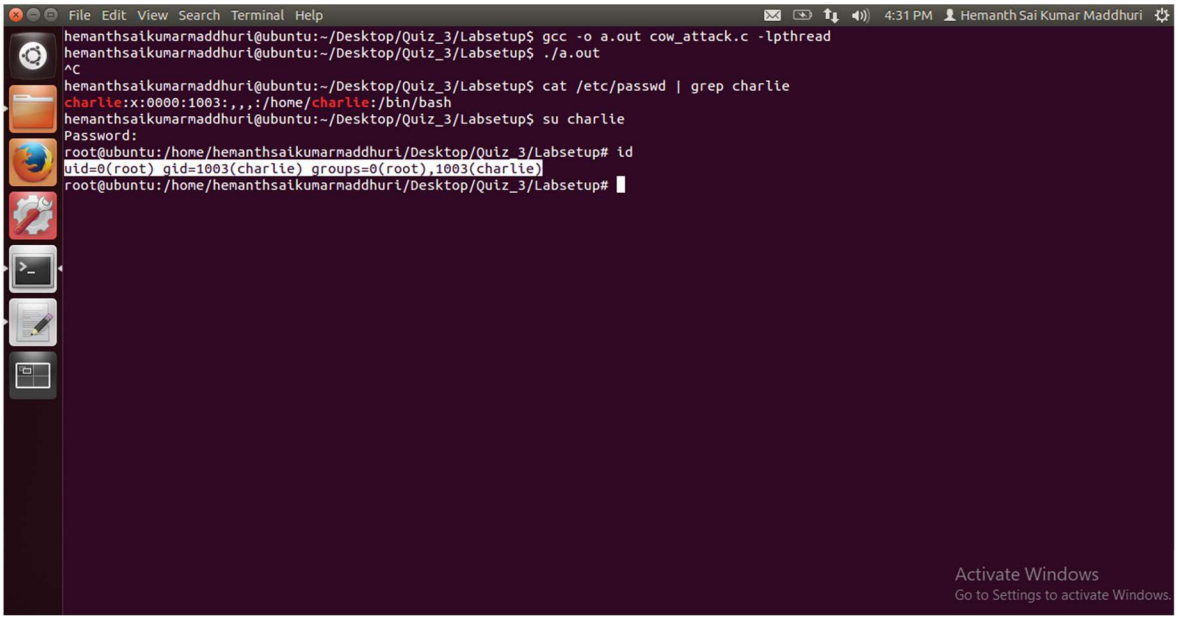
As we want to attack the file **"/etc/passwd"**, we change the attack file main thread as shown below. Then update the search string or target area as **1002 (UID)** and we update the variable content as **"0000"** in the write thread function.



Then we compile the file **"cow_attack.c"** using command **"gcc -o a.out cow_attack.c -lpthread"** as we did before on dummy file. In the below screenshot we can clearly see that password of the user charlie is set to **"0000"**.



We then verify if the user charlie is modified to root user or not, using command **"su charlie"** as instructed in the lab manual. As we can see clearly after running the command, we login as root user and we can see symbol **"#"** in the command line. The attack is successful on the file **"/etc/passwd"** which is verified by command **"id"** in root terminal.



Summary:

From this lab I have understood that Dirty Cow attack is also like Race Condition Attack as we did in last lab because in the Race Condition lab, we always use to programs run simultaneously and we are doing the same here in Dirty Cow too. I have observed that the command given for launching the attack in the lab manual is **"gcc cow_attack.c -lpthread"** which did not work in my case, so I had to change the command to **"gcc -o a.out cow_attack.c -lpthread"** to proceed further. When compared to other attacks this attack was bit easier and I observed that attackers can gain the root privilege easily using this attack. But this lab is tested on Ubuntu 12.04 version whereas I want to try running the similar attack on latest version Ubuntu 20.04 to see whether it is easier to attack in latest version too, I guess not as Ubuntu 20.04 has built in protection schemes.

Dirty Cow Attack tested on Ubuntu 20.04 (out of personal interest) but it was not successful as expected.