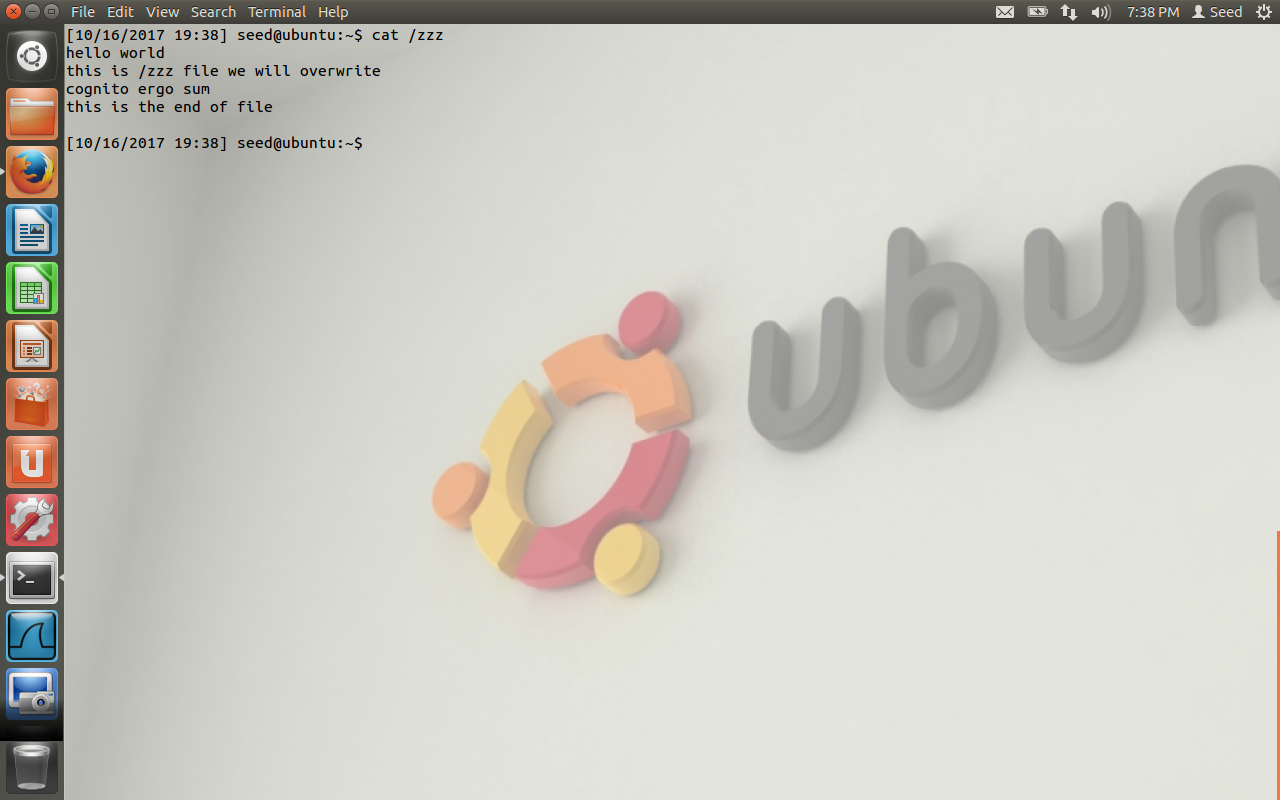
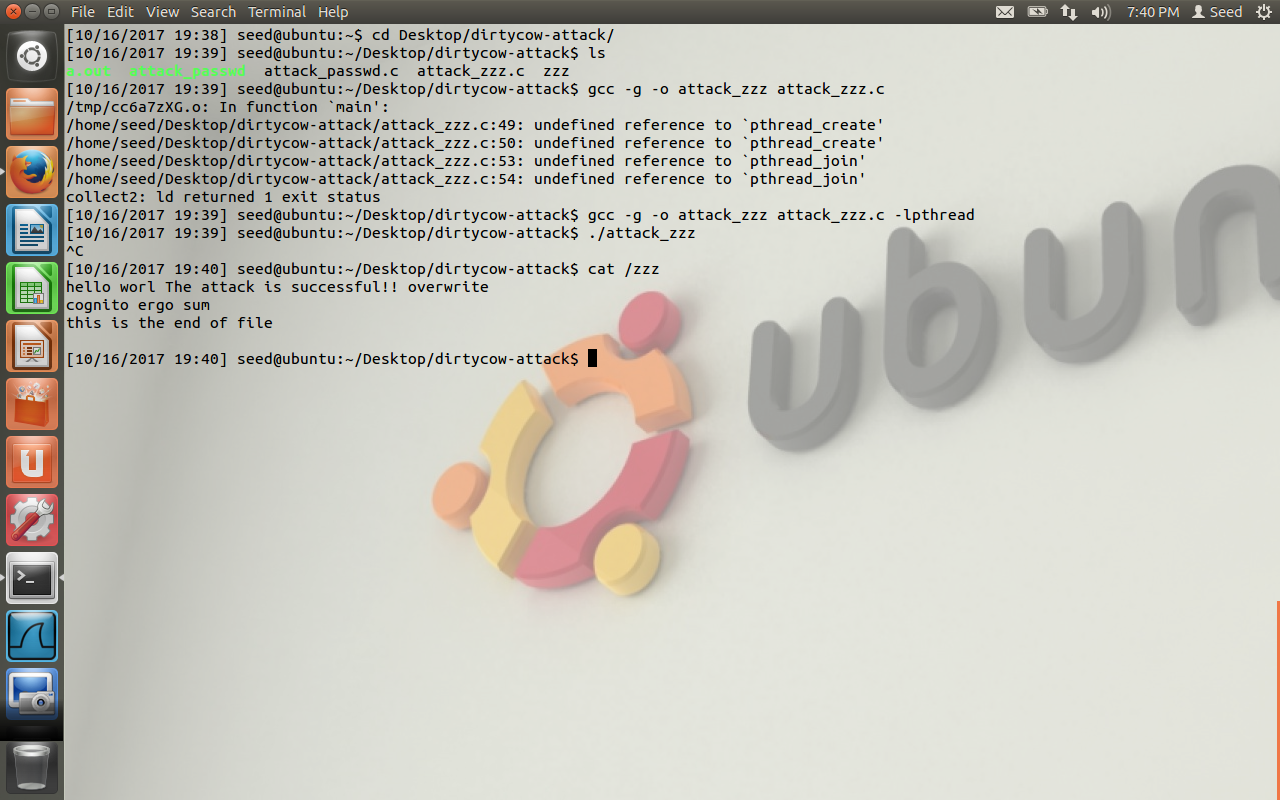
DIRTY COW ATTACK

SUID: 646254141

NAME : JASHWANTH REDDY GANGULA

**Task1:**





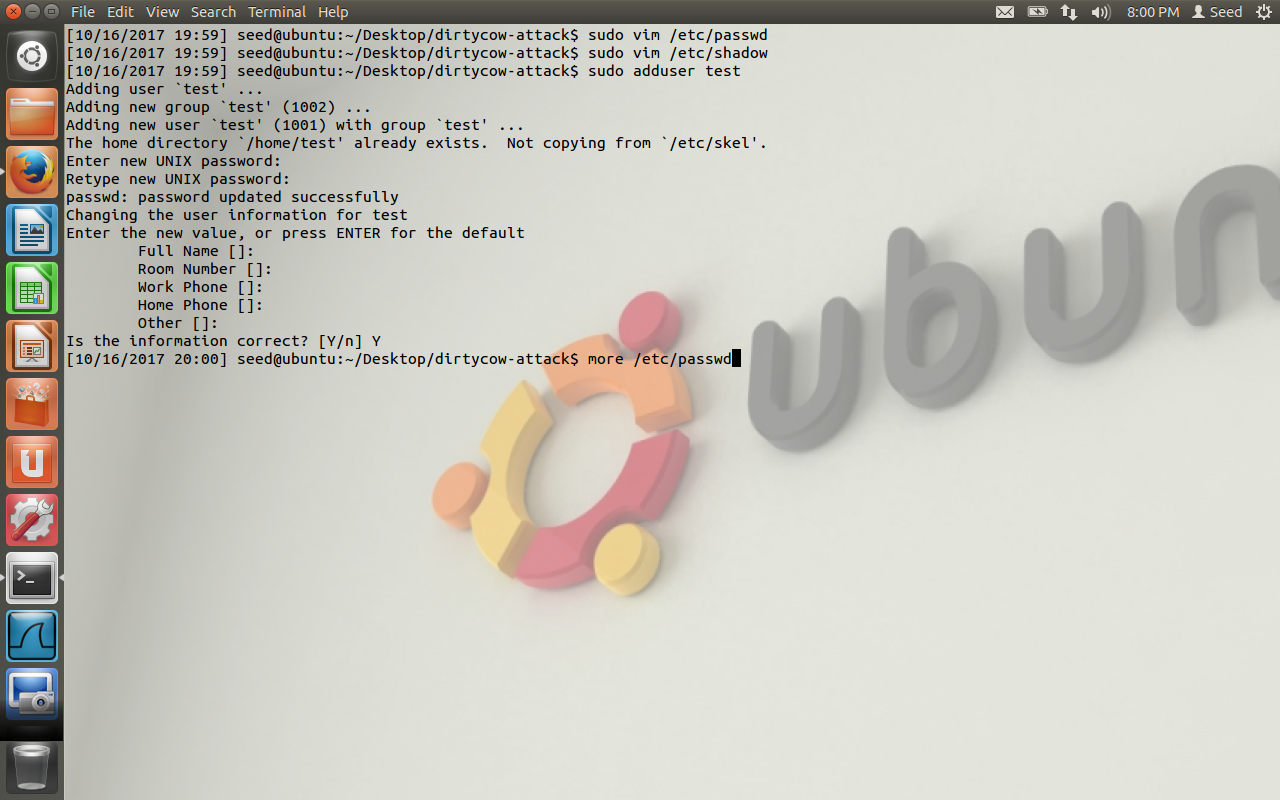
First /zzz file is created in the root directory. In the first screenshot some random text is inserted into the file. Now the attack\_zzz.c is compiled, when the program is run the /zzz is modified as observed in the second screenshot.

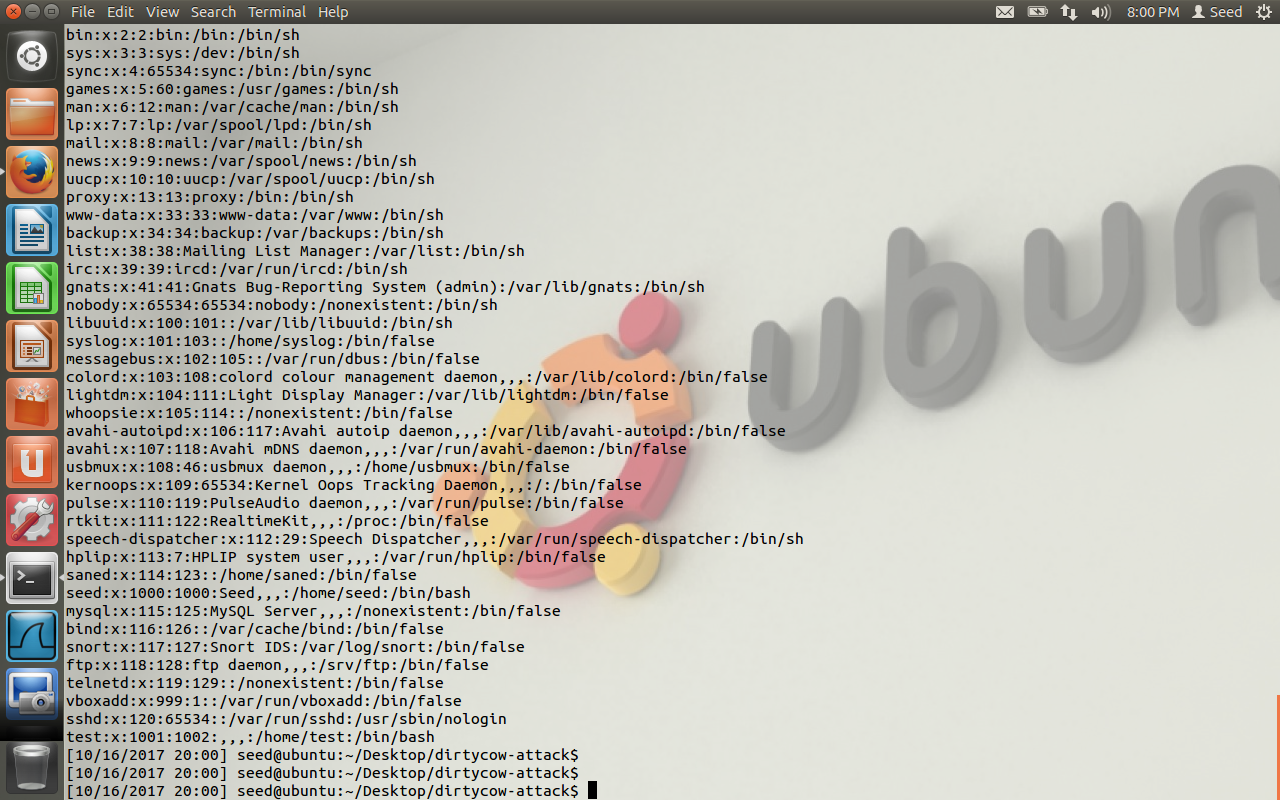
Explanation:

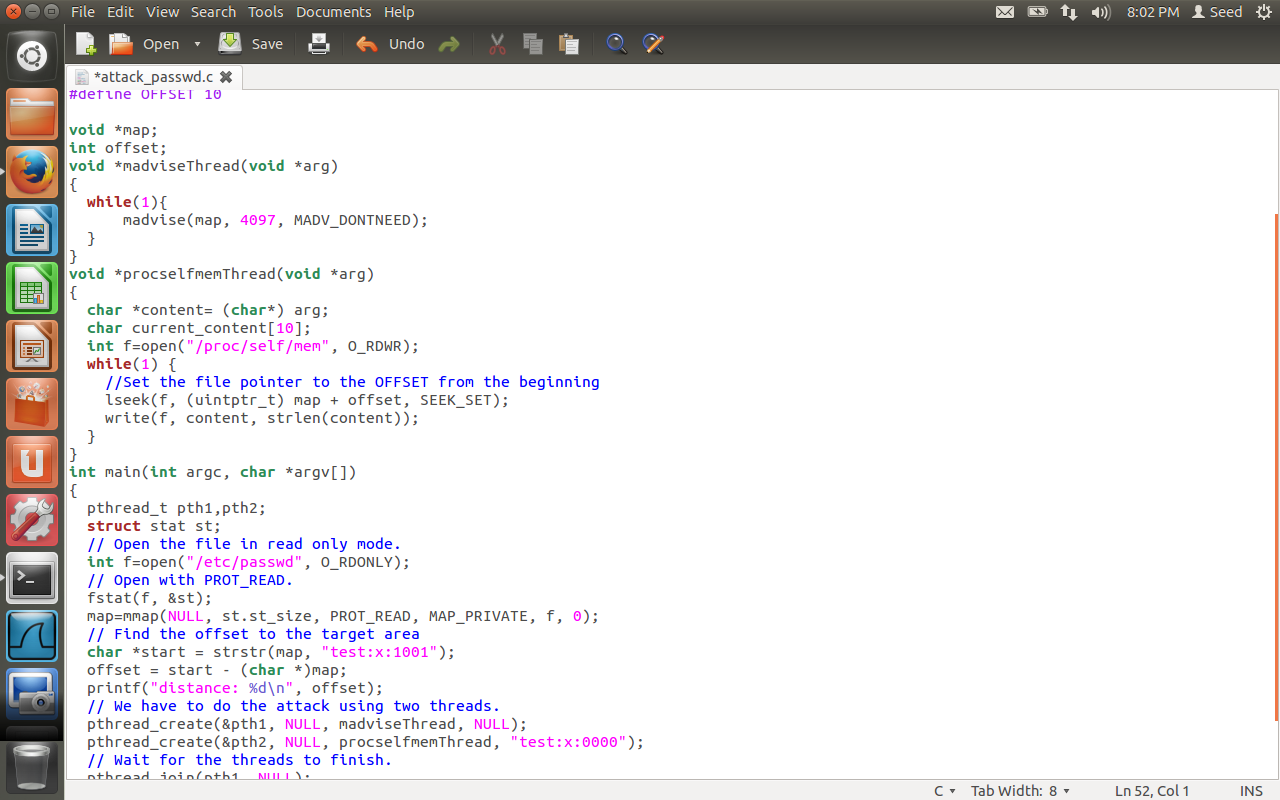
First the file /zzz is opened in read only mode. The mmap is used with the option MAP\_PRIVATE. This creates a private copy on writing mapping. Updates to the mapping are not visible to the other process mapping the same file, and are not carried to the underlying file.

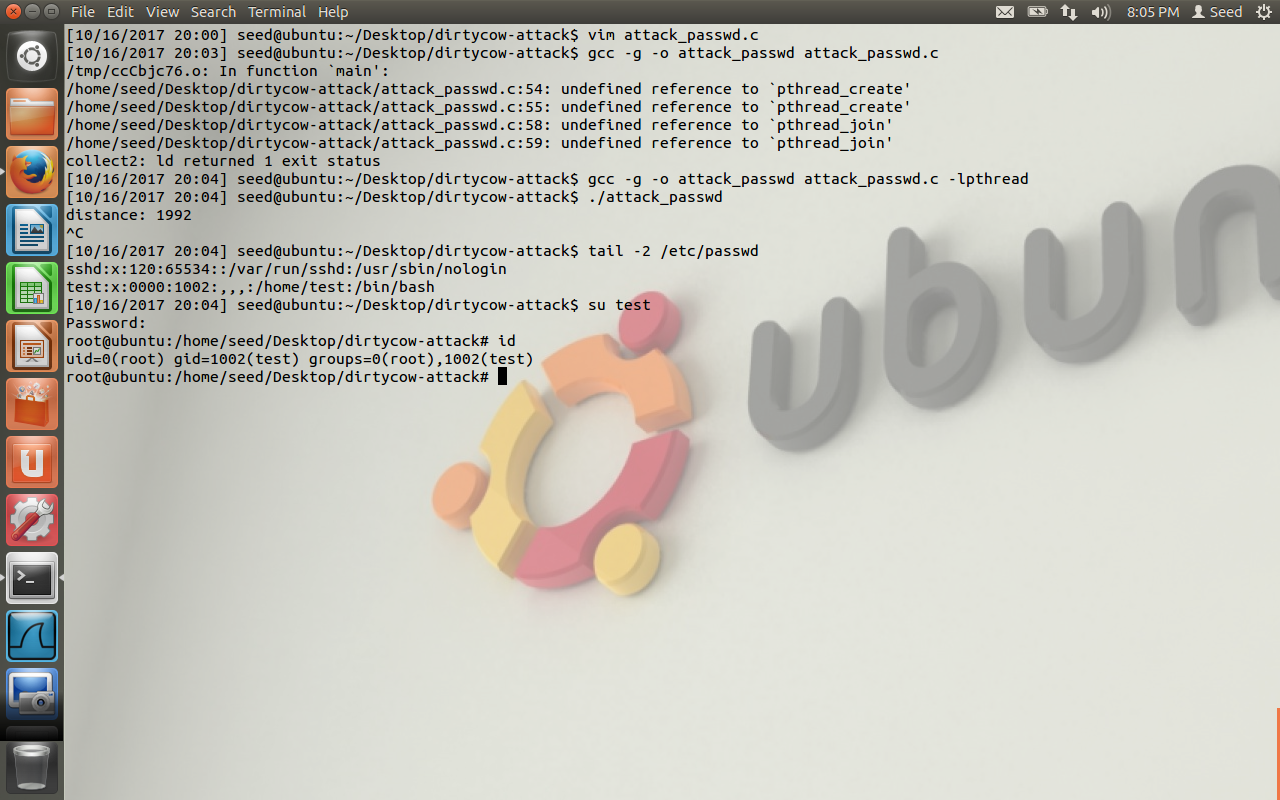
Now we have two threads, thread1 tries to write some content to /zzz file, another thread discards the changes made to the new virtual file. When these two threads are context switching rapidly, the new mapping is discarded, the old file will be written instead of new virtual mapped file. The race condition in this situation helps to attach and modify the contents of /zzz file.

**Task2:**

****

****

****



In the final screenshot, the test:x:000 is added to /etc/passwd file. We are able to login as user ‘test’ and id of test is 0. So we have root privileges. The task2 is an extension of task1, where we search for the string test:x:1001 and we want to replace it with test:x:0000. When we create a user test, the default id created is 1001. We replace it with 0. The attack is successful.