**CS336 Lab 6 – Race condition lab**

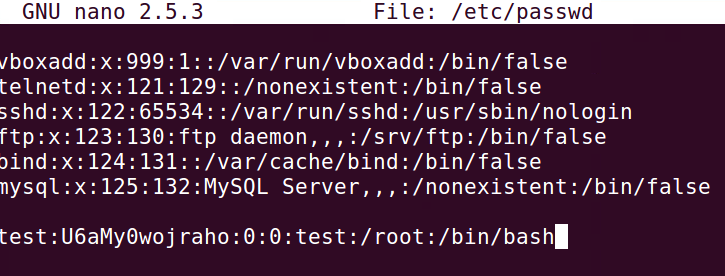
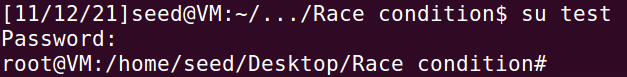
Due: Wednesday, Nov 17, 2:30 pm.

Turn in: Lab report

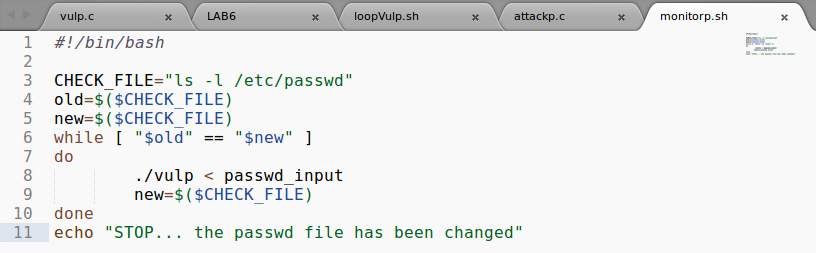
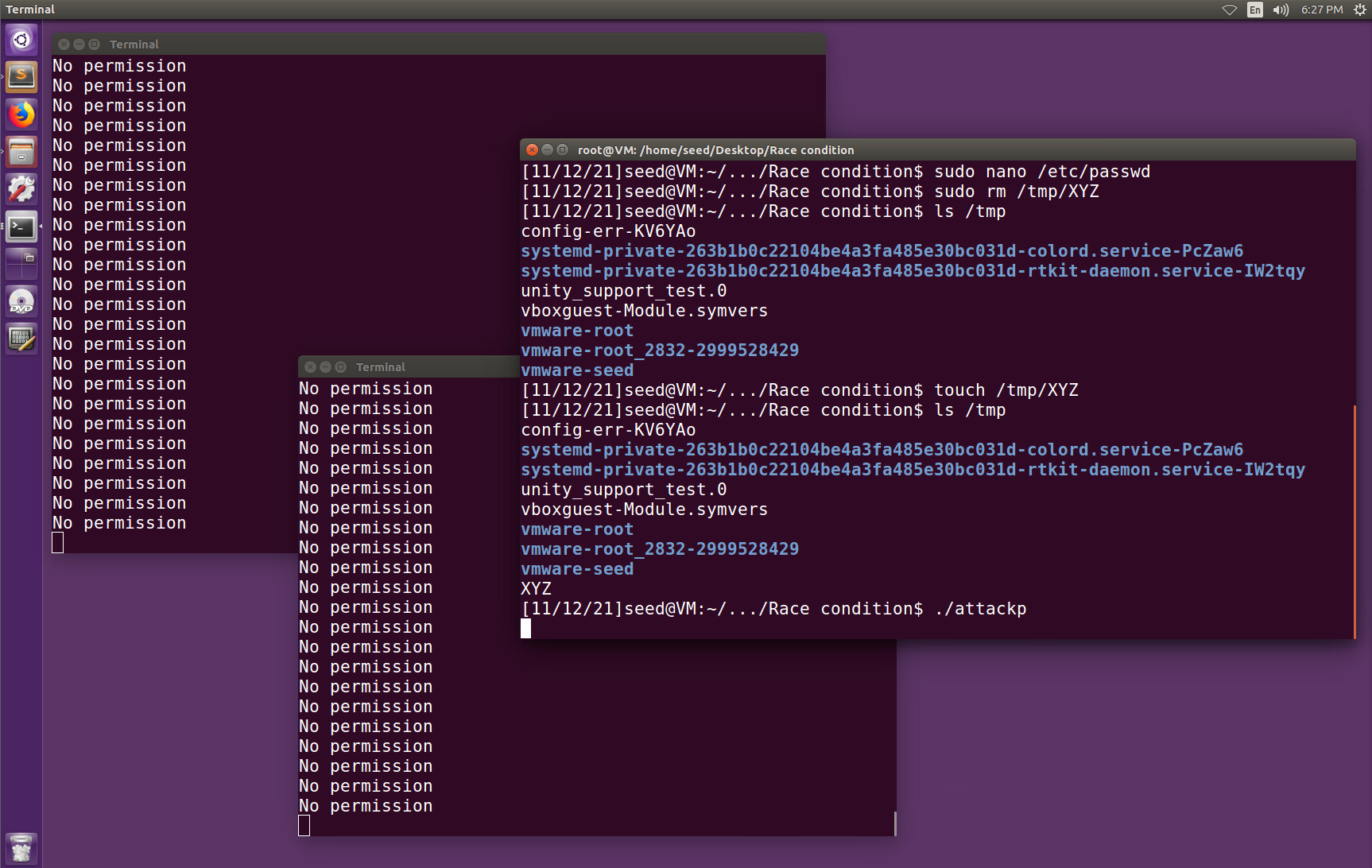
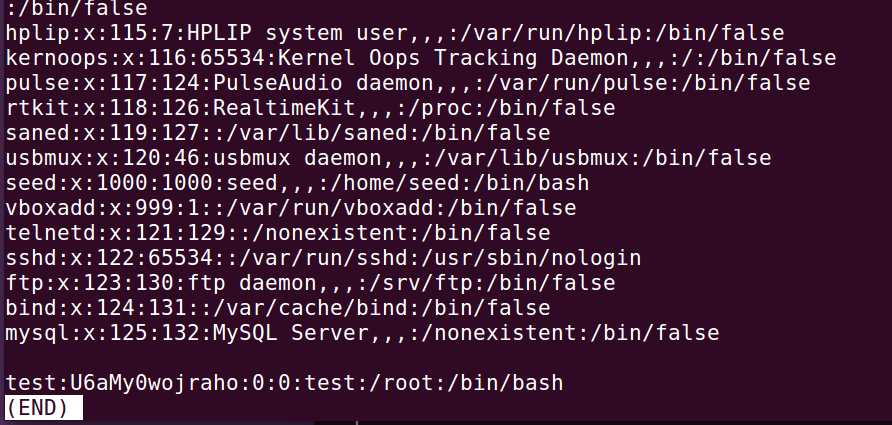
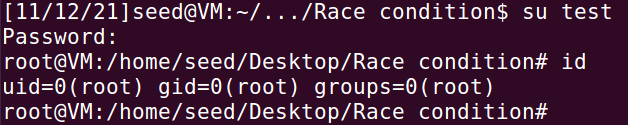
Points: 60 pts

Note: give detailed explanations in addition to the required screenshots.

(15 pts) Task 1: manually change the */etc/passwd* file.

* (5) Take a screenshot of the content of the passwd file after you make the modification.
* 
* (10) Take a screenshot to show that you can login without enter the password, and also show me that you have the root privilege.
* 

(25 pts) Task 2: Launching the race condition attack

* (5) Take a screenshot of your attack program’s code.
* 
* (5) Take a screenshot of the script which checks whether the *passwd* file is modified or not.
* 
* (5) Take a screenshot of all 3 terminals running 3 programs in parallel (one screenshot includes all three terminals)
* 
* (10) Take screenshots to show
* (1) you have launched the race condition attack successfully (2) a new entry has been added to the *passwd* file
* The test login is shown, so the race condition attack was successful, and a new entry was added:
* 
* (3) switch to the test account, and you got the root shell.
* 

(20 pts) Conclusion.

* Answer these questions (5 pts each):
* What is a race condition problem?
  + A race condition happens when multiple processes try to manipulate the same data at the same time. The end result will depend on the order of execution.
* How to exploit it?
  + In our lab, we exploited the race condition problem by running the vulnerable program in a loop, while simultaneously running an attack program that loops. After a few minutes, the timing lines up so that instead of running the code in the vulnerable program, the code in our attack program is run instead. In this case, we used this to gain root access to the virtual machine.
* What are the countermeasures? (Mentioned on the handout)
  + One countermeasure we had to disable was to prevent programs from following symbolic links in globally writable locations, like the tmp folder.
* What have you learned from the lab?
  + This lab again surprised me with how simple things like an infinite loop can be used to exploit an insecure program. I learned what a race condition is, some of the problems it can cause, and the importance of protecting against it.