Homework 2 (Due: 12:30pm on 12/4)

Submission Requirement:

- Your homework needs to be typed; handwritten submission will not be accepted.
- If you need to draw diagrams, you can hand-draw them, scan them (e.g. using your smartphone or digital camera), and insert them into your document.
- For on-campus students, you need to submit the hardcopy as well as the softcopy.
- For online students, you only need to submit a softcopy.
- No late homework will be accepted, because we will discuss the answers in class on Dec. 4th. You will receive zero points if you don't submit the homework before I start my lecture on Dec. 4th. The Blackboard's submission system will close on 12:30pm on Dec. 4th, so nobody can submit after that.

Question 1 (10 points): Which access control mechanism, ACL or Capability, is easier to implement privilege enabling/disabling? Why?

Question 2 (65 points): Assume the system has a log file called `/etc/mylog`, which is owned by root and has the 644 permission. That means normal users are not able to write to this file. You tasks is to write a program called "`addlog`", which asks the user to type a message, and then appends this message to `/etc/mylog` (note: it should not overwrite the messages that are already in the log file). Please use two different approaches to implement the program "`addlog`":

- **Set-UID approach** (use `addlog_one` as your file name): please use the Set-UID mechanism to implement this program.
- **Capability approach** (use `addlog_two` as your file name): The Set-UID mechanism is too powerful; in order to allow the program to access the file, the program needs to have the root privilege, causing over-privilege problems. The risk is that once the program is compromised, the attacker can gain the root privilege. In this task, you are not allowed to give the program the root privilege, but still the program should do what is described above. You should make sure that your program is secure, and nobody should be able to gain unauthorized privileges due to your program. You don't need to implement all the necessary access control, but you do need to point out all the risks, and describe (without implementation) how you would like to remedy those risks.

Please implement the above two approaches, and describe your implementation in your homework (screenshots and source code must be included in your submissions). You should also submit your source code to Blackboard (we will compile and run your code if necessary, so if your code cannot be compiled or executed, you don't get any credit).

Question 3 (10 points): After a root process disables its capability, it is compromised by one of the following attacks. Can the attacker use the disabled capability (assuming that the root process will not enable the capability)?

- Race condition attack
- Buffer-overflow attack

Question 4 (15 points): Please describe why the "seed" user in our VM can run any command as the root. You need to provide concrete evidence to support your explanation. You may want to pay attention to the following files:

- `/etc/sudoers`: this is the sudo configuration file.
- `/etc/group`: this file contains the information about groups.