**Assignment 3**

**CIS356** – Spring 2018

**Point Value**: 100 points

**Assignment Due Date**: **In class Thursday, March 1, 2018**

**Submission Instruction**

Please submit your assignment 3 to your instructor in class on the due date and submit your hw3\_lastname\_firstname.pdf both in class and on Schoology. If you want to get extra credits, you MUST demonstrate your bash program and submit it on Schoology.

**Description**

1. Please write down commands to accomplish the following tasks: (**40 points**)
2. Please create a file ***README.txt*** to read/write for the owner and read for everyone else.
3. Please create a file ***Input.txt*** to read/write/execute for the owner and write for everyone else. You MUST use OCT format for file permission.
4. Please create a bash file myscript.sh that count the number of files/directories in directory /home/Ubuntu/. Turn on the file’s setuid bit and setgid bit without changing the current permissions.
5. List the contents of the directory /home/proc/ sorting by modification time and listing the most recently modified file last.
6. By convention, the /tmp directory is available to all users who care to create files there. What prevents one user from reading or deleting another’s temporary files? (**10 points**)
7. Explain the relationship between a file’s UID, a running process’s real UID, and effective UID. Besides file access control, what is the purpose of a process’s effective UID? (**10 points**)
8. Suppose that a user at your site has started a long-running process that is consuming a significant fraction of a machine’s resources. (**40 points**)
9. How would you recognize a process that is hogging resources?
10. Assume that the misbehaving process might be legitimate and doesn’t deserve to die. Show the commands you would use to suspend the process temporarily while you investigate
11. Later, you discover that the process belongs to your boss and must continue running. Show the commands you’d use to resume the task.
12. Alternatively, assume that the process needs to be killed. What signal would you send, and why? What if you needed to guarantee that the process died?

**Extra Credits (1 point)**

Please write a bash script that processes the output of “***ps aux***” to determine the **number of processes** and total **VSZ** and **RSS** of the process running on the system. *You only need to submit your script file on Schoology. You don’t need to print it out and submit it in class.*