OSYS 1000 ASSIGNMENT 9

**UNIX**

**Due: Wednesday, Apr. 17, 2019 (in class)**

# Instructions

1. This assignment is in the form of a checklist that will be applied **to items you have completed** on the CentOS Virtual Machine and in its Terminal interface.
2. This assignment mostly involves having an executable shell script running on a schedule in your CentOS terminal.
3. The items in the list should have been covered in the following class videos:
   1. UNIX Basic Shell Scripting - Part 1
   2. UNIX Basic Shell Scripting - Part 2
   3. UNIX Basic Shell Scripting - Part 3
   4. UNIX Basic Shell Scripting - Part 4
   5. UNIX Scheduling with Cron
   6. UNIX Advanced Shell Scripting - Functions & Includes
   7. UNIX Advanced Shell Scripting – Sed & Awk
4. You may also want to check out the additional resources listed in the recent Weekly Agendas for more help if necessary.
5. On the day that the assignment is due you will demonstrate the completion of the tasks to the instructor in class and get the checklist completed/marked.
6. That’s it.

**Student Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Student ID: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

|  |  |  |
| --- | --- | --- |
| Item # | Item Description | Student has completed (Y/N)? |
| - | Log into CentOS. |  |
| - | Make sure you are in your home folder. |  |
| - | CREATE THE FOLLOWING SCRIPT FILE IN THE SUBFOLDER THAT IS ALREADY IN YOUR PATH. |  |
| - | Create a new ***executable script file*** called **assign9\_mike.sh**. Have a comment at the top of the script file marking yourself as the author. |  |
| - | Make sure that your script outputs to the screen when run directly. We will only redirect output to a log file when setting up the cron schedule. | filepath=”home/osys1000/logFile.txt”i |
| 1. | Separate the output of each task in the script with an output line of characters like “++++++++++++++++++++++++++++++++”. Store the output characters in a variable to make it easier to reprint. | sep = “++++” |
| 2. | Output a line stating that the script is starting and indicating the time the script started. | date  echo date >> $filepath |
| 3. | The script will output the current user running the script, but if the user is “root” it will replace that with “classified”. | whoami | sed s/root/classified/  echo ‘whoami | sed s/root/classified/’ >> $filepath |
| 4. | The script will output the contents of the /etc/fstab file without the line on the swap partition (i.e. only excluding that line). | cat /etc/fstab | sed /swap/d  echo ‘cat /etc/fstab | sed /swap/d’ >> $filepath |
| 5. | The script will output a full directory listing of the /var/log directory showing only the file name and THEN the file size. | ls -la /var/log | awk ‘{print $9 “ “ $5}’  echo ‘ls -la /var/log | awk ‘{print $9 “ “ $5}’’ >> $filepath |
| 6. | Set up a cron task to run your script automatically via your own user’s crontab file. **Make sure to write/appending to a log file when running the script in cron so we can see the results. (HINT: We will also be running the script interactively as “root” when we test it as well).**  (3 pts). | Sudo vi /etc/crontab |