CS 480 M01 Fall 2020 - Homework Assignment #4

Due: 11:59 PM, Monday, Oct 5 2020
Assignment:

Points: 95 points total

I. 15 points : Read all **man pages** listed below (from sections 1 , 5, or 8; try the commands, look into the files, ... while reading) and in a short paragraph (3-5 lines per man page in your words - do not just copy parts of the man page) describe what is each of them for. Also, for the commands choose an extra option (not the help nor version options) and in few more lines describe situations in which you would use the selected option. In case of the commands with long man pages and many options choose at least two extra options and describe situations in which you would use them.

1. at
2. cron
3. crontab (*config file*)
4. date
5. journalctl
6. ps
7. sleep
8. systemctl
9. systemd
10. systemd.service
11. trap
12. uptime
13. w

II. 5 points :
What is the purpose of the shadow password file?

Each exercise below is to be implemented on your cs480 virtual host. Every question asked needs to be answered in your assignment report. For every exercise below describe in sufficient detail the whole process and explain what you did to accomplish the task. Make sure that you list and discuss all important steps made, commands used together with their options, and problems encountered.

III. 9 points
Write a simple shell script or perl script that finds all the hard links of regular files in a subtree under a specified directory (but it will not cross onto another filesystem if that is mounted under the original filesystem file tree).
(*On your cs480 virtual machine create couple of hard links and test your script to make sure that the links are found. Submit a transcript of the session showing that your program works as expected*)

Example of the script's desirable output:

```
itest:~ # /root/4.3.pl
Usage: /root/4.4.pl <DIR_NAME>
exiting ...
itest:~ # /root/4.3.pl /tmp /extra
Usage: /root/4.4.pl <DIR_NAME>
exiting ...
itest:~ # /root/4.3.pl /extra
Hard links under /extra
inode 14:
14 -rw-r--r-- 3 root root 0 Sep 28 21:07 /extra/cs480/test/file2hlagain
14 -rw-r--r-- 3 root root 0 Sep 28 21:07 /extra/cs480/file2hl
14 -rw-r--r-- 3 root root 0 Sep 28 21:07 /extra/file2
inode 655362:
655362 -rwsr-sr-x 2 root root 3 Sep 14 09:17 /extra/cs480/file1
655362 -rwsr-sr-x 2 root root 3 Sep 14 09:17 /extra/cs480/file1hl
itest:~ # /root/4.3.pl /extra/cs480
Hard links under /extra/cs480
inode 14:
14 -rw-r--r-- 3 root root 0 Sep 28 21:07 /extra/cs480/test/file2hlagain
14 -rw-r--r-- 3 root root 0 Sep 28 21:07 /extra/cs480/file2hl
inode 655362:
655362 -rwsr-sr-x 2 root root 3 Sep 14 09:17 /extra/cs480/file1
655362 -rwsr-sr-x 2 root root 3 Sep 14 09:17 /extra/cs480/file1hl
```

Note:
The script does not have to list the directory references to the inodes that are above the specified path (file /extra/file2 is not listed for a search of /extra/cs480/).

IV. 8 points
   a. Create a simple shell script (**state.sh** or **state.csh**) that will append info listed below into **/tmp/state.log** file:
      - current date and time
      - current load and users logged in
      - all processes running on the machine (long listing that, among other info, includes PID and PPID)
   b. Create a cron job that will execute this script every 5 minutes (the state info gets appended into file */tmp/state.log* every five minutes)

V. 8 points
Create a new script **movelog.sh** or **movelog.csh** and another crontab entry so that
   a. The movelog.sh or movelog.csh script will
      - move the /tmp/state.log file into file named statelog.YYYYMMDD where YYYYMMDD is the current date (statelog.20170918, statelog.20170919, ...) and
      - removes old statelog.YYYYMMDD files that are 2 or more weeks old
   b. each statelog.YYYYMMDD file has the state info (date, users, processes) stored every 5 minutes for the whole day (from midnight to the next midnight)

VI. 3 pts: Install GNOME
yast -> Software mngmt -> Filter : Patterns -> and add the GNOME Desktop Environment (X11)

VII. 4 pts: Without using yast (you cannot use yast/yast2) modify your system to have the graphical interface as the default target.

VIII. 13 pts: Write a simple Bash shell script **/root/trap.sh** which will
   1. Write its process ID into the file **`/var/run/trap.pid`** .
   2. Append one line of a text into the file **`/root/trap.log`**.
      The line will include the shell's current name and process ID followed by the current date similar to:
      "./trap.sh [4441] started on Mon Sep 28 10:04:54 MDT 2020"
   3. Trap all these signals: INT, QUIT, HUP, TERM , USR1, USR2 and write a message about it into the log file. Also,
      - Once the TERM signal is received the shell will remove the /var/run/trap.pid and exit.
      - The shell will continue to run when the other signals are received.
   4. Append the messsage **`"./trap.sh [PID] time tick ... DATE"`** into the **`/root/trap.log`** every 10 seconds.
      The DATE and PID are to be the actual current date and process ID of the running process.

Example of the log file content:

```
# cat /root/trap.log
/root/trap.sh [11384] started on Mon Sep 28 10:28:14 MDT 2020
/root/trap.sh [11384] time tick ... Mon Sep 28 10:28:24 MDT 2020
/root/trap.sh [11384] INT received on Mon Sep 28 10:28:29 MDT 2020
/root/trap.sh [11384] time tick ... Mon Sep 28 10:28:29 MDT 2020
/root/trap.sh [11384] time tick ... Mon Sep 28 10:28:39 MDT 2020
/root/trap.sh [11384] USR1 received on Mon Sep 28 10:28:49 MDT 2020
/root/trap.sh [11384] time tick ... Mon Sep 28 10:28:49 MDT 2020
/root/trap.sh [11384] HUP received on Mon Sep 28 10:28:59 MDT 2020
/root/trap.sh [11384] QUIT received on Mon Sep 28 10:28:59 MDT 2020
/root/trap.sh [11384] time tick ... Mon Sep 28 10:28:59 MDT 2020
/root/trap.sh [11384] time tick ... Mon Sep 28 10:29:09 MDT 2020
/root/trap.sh [11384] time tick ... Mon Sep 28 10:29:19 MDT 2020
/root/trap.sh [11384] TERM receved - exiting ... Mon Sep 28 10:29:29 MDT 2020
```

IX. 10 pts: Create and configure new "service" **trap_test.service** that will excute the **trap.sh**.
Create the **/etc/systemd/system/trap_test.service** unit file and use **systemctl** command to add the service among services in the graphical and multi-user targets.
The command **systemctl** will start and stop the "service" (**`systemctl start trap_test.service`** and **`systemctl stop trap_test.service`**).

NOTE: Your solutions to all of the exercises above must be implemented, tested, and execute correctly on your cs480 virtual machine. Among your attachments (as part of the tar.bz2 file) submit also transcript files showing the correct execution of the commands / scripts as you executed them on your cs480 virtual machine. You can use the **script** command to record the session and submit the resulting files (3.4_trascript.txt, 3.7_trascript.txt, ...) with your homework assignment as part of your bz2 file submited into your directory under /home/CS480.

Submitted files, overall quality, length, and feedback:

X. 5 pts : **Submit all relevant files** you modified during the lab (into the /home/CS480/yourcsloginname directory as part of the tar.bz2 file) . Exactly two files need to be submitted: One file being your report in **PDF** is to be submitted on the Blackboard. The other file is the **tar.bz2** file containing everything else and it gets submitted into **/home/CS480** (Pack the files together using **tar** and **bzip2**, and copy them using **scp** from your computer into your cs account)

XI. 10 pts - Submit your report as a **PDF document** through learn.nmsu.edu and all supporting documents, programs, and execution transcripts as part of the tar.bz2 file in your directory under /home/CS480. The report has to contain answers to all questions above, and for the practical problems on your computer describe in detail choices you made, difficulties you encountered, and how did you deal with them.

XII. 5 pts - **Feedback:**In the last paragraph of your report estimate how much time you spent on this assignment. Describe what was hard, easy, interesting, boring or confusing.

Please remember the submission requirements:

Two files need to be submitted for every assignment :

1. The first file is your **typed report** .
   It has to be submitted through **http://learn.nmsu.edu as one file / document** and it has to be **in the PDF format**.
   No other format will be accepted. The name of the file has to contain your name and assignment/lab number separated by "_" (If your name is John Smith then the name of the document submitted for this assignment needs to be **John_Smith_HW04.pdf**).
2. The second file to be submitted into your directory under **/home/CS480** directory has to be a **tar.bz2** file that contains all other files related to the assignment.
   It will contain all relevant files that you modified for the assignment. Everything that was done to finish the lab, modify your system, write, run and verify submitted programs and scripts used needs to be included in the submitted tar.bz2 file. That includes all your system configuration files modified for the lab/assignment as well as programs and scripts written (including complete source code, supporting shell scripts, test data input files, ...). The name of the tar.bz2 file has to contain your CS login name and the lab number separated by "_" (if your cs login name was jsmith then the file's name for this assignment will be **jsmith_Lab04.tar.bz2**). The files submitted as part of the tar.bz2 file have to be exact copies as found on your cs480 lab system.

If you want to correct or add something to already submitted assignment than you need to do so before the deadline and **resubmit the whole report again and also resubmit your new tar.bz2 file** .

Please remember that **late submissions are penalized 20% / day and NO submissions that are late 3 or more days will be accepted**. You can be late at most 2 days to receive any points.