CS344: Operating Systems Lab

Lab # 03 (1 Questions, 100 Points)
Held on 05-Sep-2023
Lab Timings: 09:00 to 12:00 Hours Pages: 2
Submission: 12:00 Hrs, 05-Sep-2023
Instructor Dr. V. Vijaya saradhi
Head TAs Adithya Moorthy & Laxita Agrawal
Department of CSE, IIT Guwahati

a. This assignment is related to chapter 3, `Process Management` in the book `Operating System Principles, Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne`.

b. In order to perform this assignment, understanding of system calls `fork(), umask(), getrlimit(), setsid(), chdir(), openlog(), syslog(), wait(), exit(), kill(), sleep()` are essential.

c. In order to perform this assignment, read the material, chapter 13, "Daemon Processes" of the book titled "Advanced programming in UNIX environment". The material is shared with you through the CSE and M&C branch representatives.

d. Read the manual pages for the above system calls.

e. Read the manual page `man 7 daemon` for a complete understanding of daemon process

f. This programs requires `sudo` privileges. You have to start and stop the daemons in root privilege mode.

g. Must be careful running this program. If implemented incorrectly or run without a proper testing, the program will consume memory and the system stops running at some point.

h. Do not forget to stop the daemon.

Question 1: (100 points)

Daemon Process: Implement the following:

a. (71 marks) Write a `start_daemon` function which performs the following tasks in C language
   i. (5 marks) Create a child process
   ii. (21 marks) Perform the following tasks in order to run the child process as daemon. Each task carry three marks.
      i. Call `umask` function
      ii. Create child using `fork`
      iii. Call `setsid`
      iv. Change working directory to root directory using `chdir`

v. Close all file descriptors.

vi. Initialize the log file using `openlog` and `syslog`

vii. Create daemon only once.

iii. (5 marks) Write a message of successful creation of `daemon` in the `syslog` file.

iv. (30 marks) The daemon should create a child every 20 seconds to compute the function for n whose value is equal to created child's `pid` (stop sequence generation when $n = 1$). Write the sequence generation logic withing the child process itself. Do not call `exec` family of functions immediately after `fork()`.

$$n = \begin{cases} \frac{n}{2} & \texttt{if n is even} \\ \\ 3 \times n + 1 & \texttt{if n is odd} \end{cases} \tag{1}$$

Note Do not forget to shutdown the daemon. A running daemon creates several processes and erroneous implementation may lead the OS to create many children and finally running out of space.

v. (10 marks) Append the sequence generated by the above function into the `syslog` file.

b. (29 marks) Write a function `stop_daemon` function which `kill`s the daemon process by sending `SIGTERM` signal to `pid`.

i. (20 marks) Identifying the `pid` to send `kill` signal

ii. (9 marks) Writing message to `syslog` about stopping of daemon

c. Invoke `start_daemon` in the main function on command line argument "start".

d. Invoke `stop_daemon` in the main function on command line argument "stop".

Example: If the executable name is `run_daemon`, then run this program as follows:

- `sudo ./run_daemon start`
- `sudo ./run_daemon stop`

Important Note. Do not forget to stop the daemon.