College of Science and Technology

Royal University of Bhutan

LESSON – 14

MANAGING PROCESSES

LEARNING OUTCOMES

- Identify parent process and child process
- Set priority of processes by using nice and renice
- Manage processes using top utility

PROCESS MANAGEMENT

- Introduction to process management:
  - Shell Jobs -> refers to as interactive processes. Shell jobs are commands started from the command line.
  - Daemons are processes that provide services. They are running with root privileges. Eg. httpd, cron, sshd

- Foreground vs Background Jobs
  - Any executed command is started as a foreground job.
  - Some processes do not require user interaction (Background)

PROCESS MANAGEMENT

- Managing Shell Jobs:
  - The **jobs** is a command to see all current jobs.

- Job Management overview:
  - & (use at the end of a command line)
  - Ctrl+Z
  - Ctrl+Z+t
  - Ctrl+C
  - bg
  - fg
  - Jobs   (-l, -r, -s)

PROCESS MANAGEMENT

- Process States in Linux (**top** command):

A process in Linux can go through different states after it’s created and before it’s terminated.

These states are:

- Running
- Sleeping
  - Interruptible sleep
  - Uninterruptible sleep
- Stopped
- Zombie
- A process in running state means that it is running or it’s ready to run.
- The process is in a sleeping state when it is waiting for a resource to be available.
- A process in Interruptible sleep will wakeup to handle signals, whereas a process in Uninterruptible sleep will not.
- A process enters a stopped state when it receives a stop signal.
- Zombie state is when a process is dead but the entry for the process is still present in the table.

ACTIVITY I

- Managing Jobs:

1. Open a root shell and type the following commands:

```
sleep 3600 &
dd if=/dev/zero of=/dev/null &
sleep 7200
```

1. Now you have two background and one foreground process. You have to wait for 2 hours to get the shell back. Press **Ctrl+Z** to stop it.
2. Type jobs and note down what do you see?
3. Type bg 3 to continue running job 3 in the background and type jobs command. What is your observation?
4. Type fg 1 to move job 1 to the foreground and then press Ctrl+C. Type jobs and note down your observation.
5. Use the same approach to cancel jobs 2 and 3 also.
6. Open a second terminal, type dd if=/dev/zero of=/dev/null &
7. Type exit to close the second terminal.
8. From other terminal, start top. You will see that the dd job is still running. From top, use k to kill the dd job.

PROCESS MANAGEMENT

- Managing Parent Child Relations:
  - When a process is started from a shell, it becomes a child process of that shell.
  - All processes started from a shell are terminated when that shell is stopped.

PROCESS MANAGEMENT

- Using Common Command-Line Tools for Process Management:
  - ps -> list running processes
  - ps au -> displays information about all processes running on the system
  - ps –ef -> will show all information about all processes

Example: ps au | grep nano

PROCESS MANAGEMENT

- Sending Signals to Process with:
  - kill
  - killall
  - Pkill
- To see man 7 signal
- 3 signals works for all processes
  - SIGTERM (15)
  - SIGKILL (9)
  - SIGHUP (1)
- Use kill –l to show a list of available signals that can be used with kill

PROCESS MANAGEMENT

- Process Management:
  - Adjusting Process Priority
  - killall
  - Pkill
- To see man 7 signal
- 3 signals works for all processes
  - SIGTERM (15)
  - SIGKILL (9)
  - SIGHUP (1)
- Use kill –l to show a list of available signals that can be used with kill

PROCESS MANAGEMENT

- Adjusting Process Priority:
  - **nice**
    Example: nice –n 5 dd if=/dev/zero of=/dev/null &
  - **renice**
    Example: renice –n 10 –p 1234 (assume PID for about process is 1234)

-20	20
Most priority process	Least priority process

** nice value only controls CPU time assigned to process and not utilisation of memory and I/O devices.

PROCESS MANAGEMENT

- Using top to Manage Process:
  - **top**
  - From top, type k to prompt for the PID to kill the process.

- To renice a running process from top
  - Type r, Enter PID for renice and after entering PID, you will be prompted for the nice value you want to use.

ACTIVITY II

- Setting process priority and killing it:

1. Launch the command **dd if=/dev/zero of=/dev/null** three times as a background job.
2. Increase the priority of these commands using nice value -5. Change the priority of the same process again, but use this time the value -15. Observe the difference.
3. Kill all the dd processes you just started.

SUMMARY

- In this lesson, you have learnt that:

- The managing processes
- The process can set priority using nice and renice commands
- The processes can be killed by sending signal and also from top command