**Assignment 5**

1.How would you check

**1.a Memory used by a process(RAM)**

$ free –m

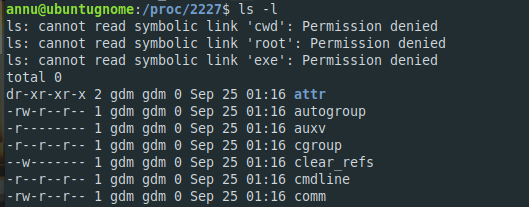
-m : display the memory in mb

**$** /proc/meminfo

$ top

**1. b Total number of open files by a process**

**$ cd /proc/[PID]/fd/**



**1.c Running duration of a process**

**i. Find the process id**

$ ps aux | grep -i <process name>

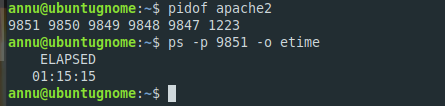
$ pidof <process name>

ii. how long process was running

$ ps –p <pid> -o etime

$ ps

TIME header in ps show the total execution time



**2. What is File Descriptor**

**->** When you open a file, the OS creates an entry to represent the file and store the information about the opened file . So if there are 100 files opened in your then there will be 100 entries in OS. These entries are represented by integers like( ..100.101.102….). This entry number is called file descriptor. So, it is just a number which represents an opened file in OS. The entries are made in Process Table.

**3. How to kill a process**

a. forcefully

$ kill -9 <pid>

b. gracefully

$ kill -15 <pid>

**4. What are the signals?**

Signals are interrupts sent to a process and based on the signals it responds accordingly. The events can vary from user requests to illegal memory access errors. Some signals, such as the interrupt signal, indicate that a user has asked the program to do something that is not in the usual flow of control.

Command to display list of signals :-

$ kill –l

|  |  |  |
| --- | --- | --- |
| **Signal Name** | **Signal Number** | **Description** |
| SIGHUP | 1 | Hang up detected on controlling terminal or death of controlling process |
| SIGINT | 2 | Issued if the user sends an interrupt signal (Ctrl + C) |
| SIGQUIT | 3 | Issued if the user sends a quit signal (Ctrl + D) |
| SIGFPE | 8 | Issued if an illegal mathematical operation is attempted |
| SIGKILL | 9 | If a process gets this signal it must quit immediately and will not perform any clean-up operations |
| SIGALRM | 14 | Alarm clock signal (used for timers) |
| SIGTERM | 15 | Software termination signal (sent by kill by default) |

**5. What is parent process id?**

Parent Process:- The process which create other processes (child process) is called a praendt process. In addition to process id , each process is assigned a parent process ID (PPID) that tells which process started it. The PPID is the PID of process’s parent.

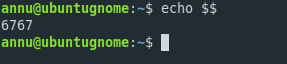
Command to get the PPID.

ps –f <PID>

**6. Print PID of current shell.**

**-> echo $$**

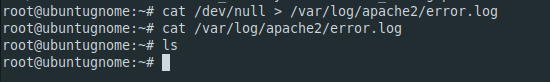
**->ps -f**

****

**7. How to clear a log file of running process?**

It will clear the whole content and remove all lines inside the file

cat /dev/null > /var/log/apache2/access\_log

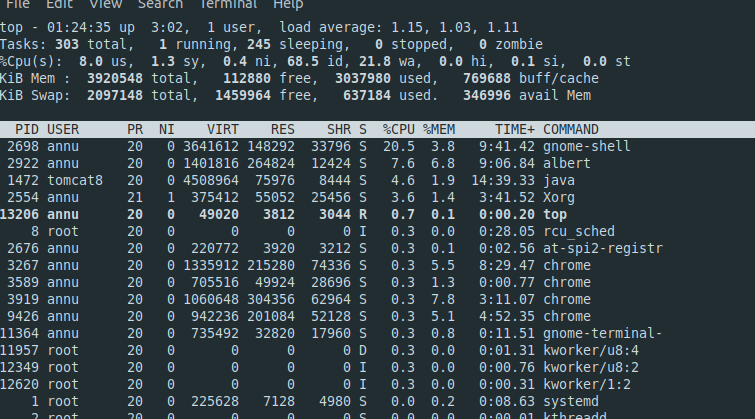


**8. What will happen if you delete a log file of running process?**

When you delete a log file a running process continue in writing to the logfile at the same offset as before the logfile moved. This results in a new logfile being created which is just as big as the old one, but it now contains thousands (or millions) of null characters.

**9. How do you check all the running process in the system?**

**$** top



S header having status R tells the running process

**10. How do you check those processes that are waiting for the resources?**

ps aux

STAT header with status S tells the process waiting for event,

D flag tell the process waiting for I/O.

**11. What init process is responsible for?**

Whenever a system boots, firstly the init process is started which is actually responsible for running other start scripts which mainly involves initialization of hardware, bringing up the network, starting the GUI. The init finds the default run level of the system so that it could run the start scripts corresponding to the default run level.

**12. What are Running, Waiting, Stopped and zombie processes?**

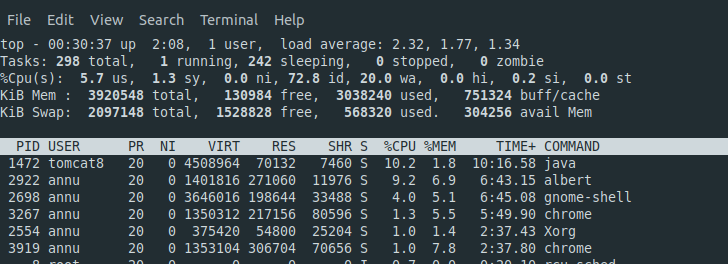
**Running:** process which is executing or ready to run is called running process

**Waiting:** process which is waiting for an event to occur or for system resources.

It is of 2 types: - Interruptible & uninterruptible waiting process

**Stopped:** process which receive a suspend signal (SIGSTOP, SIGTSTP) and do not perform any processing. It can only be resumed if SIGCONT signal is sent

**Zombie:** A process which is killed but still shows its entry in the process status or the process table is called a zombie process; they are dead and are not used.



**13. How do you elevate the priority of a process?**

renice: to change the priority of a running process

**Syntax**: renice [ renice value] [process id]



**14. What are stdin,stdout and stderr and how do we used them?**

**a. stdin(Standard Input)**

Generally standard input comes from input device like keyboard , but Linux

also allows you to take standard input from file.

**Eg : cat < myscript**

This would tell cat to take input from file myscript instead of from the keyboard.

**b. stdout**

Standard output , as created at process creating time, goes to the console, terminal. So console or terminal is the device for accepting the output. But we can change this and redirect the output to a file

**Eg: ls –al > myscript**

**c.stderr**

Every command sends its output to one of the 2 places :

* **valid output**
* **an error message**

It does the same with the errors as it does with the standard output

Eg : find / -name “.htaccess”