System Programming

Linux command line manual: lab 5

2015 - 2016

Bachelor Electronics/ICT

Course coördinator: *Luc Vandeurzen*
Lab coaches: *Jeroen Van Aken*
Stef Desmet
Tim stas
Luc Vandeurzen

Last update: April 24, 2017

Lab targets: obtain process and thread related info and statistics, including IPC data

Process-related info

Once you implement multi-process, multi-threading programs using IPC, there are a some very powerful commands that help to diagnose and debug your programs. Try out the following commands:

- top : interactive tool to show and refresh a Linux process overview; this command implements a huge amount of options and features (formatting, sorting, filtering, …); the following examples illustrate just a few of them:
  - top : hit 'h' to get the 'top' help;
  - top : hit 'd' to change the refresh interval;
  - top : hit 'V' to get a tree / forest overview of the processes;
  - top <program> : monitor a specific user <program>;
  - top -H : display individual threads i.s.o. a summation per process;
- ps : display information on a snapshot of all or a selection of the active processes; just like 'top', many options to filter processes and format the output are available; just a few examples:
  - ps : without arguments shows only processes associated with the same terminal;
  - ps -ef : show info on all processes on the system;
  - ps -efL : show info about threads;
  - ps -u <user> : filter processes on <user>;
- pgrep : find all processes that match a search pattern; a typical example of 'pgrep' is:
  - pgrep <string> : find all processes that contain <string> in their name (not in the process path!);
- kill : command to send a signal to a process;
  - kill -l : give a list of available signals on the system;
  - kill -9 <pid> : terminate the process with PID <pid>; remark that 9 the signal number of the SIGKILL-signal is;
  - kill -SIGKILL <pid> : same as the previous one;

Note: 'pkill' and 'killall' are variants on 'kill'. Use the man pages to learn more on these two commands.

- pmap <pid> : show the memory map of a process;
- ipcs : show information on IPC resources (shared memory, message queues, semaphores) in use on the system;
  - ipcs -s -i <id> : shows detailed info on the semaphore with ID <id>;
  - ipcs -l : shows system limits of IPR resources;
  - ipcs -m -c : shows info on the creator/owner of message queues;
- The 'stat'-family of performance monitoring tools containing tools like 'vmstat', 'mpstat', 'pidstat' and 'netstat' (it might be needed to install additional packages with 'apt-get install'). We focus here only at 'pidstat'.
  - pidstat : report statistics for Linux processes and threads at kernel level; once more a few examples to illustrate some options:
    - pidstat -p ALL : display all tasks, active or non-active;
    - pidstat -p <pid> : report statistics for the process with ID <pid>; include the '-t' option to get statistics per process thread;
    - pidstat -T CHILD -p <pid> : report statistics for the children of the process with ID <pid>;
    - pidstat -d 5 : report every 5 seconds I/O statistics per task;
- nproc : show the number of processors available
  - int get_nprocs_conf(void) : returns the number of processors configured by the OS;
  - int get_nprocs(void) : returns the number of processors available;
- fuser : show which processes use a given file, socket, fifo, …
  - lsof : list all open files
- Finally, we review the /proc virtual file system (see also Linux CLI lab 1) again: /proc is a virtual file system used by the kernel to send information about system resources (kernel data structures) to processes containing information; a few examples:
  - /proc/<pid>/cmdline : holds the command line (incl. cmd line args) for the process;

- /proc/<pid>/cwd : symbolic link to the current working directory of the process;
- /proc/<pid>/fd : subdirectory containing info on all files the process has open;
- /proc/<pid>/io : file containing I/O statistics for the process;
- /proc/<pid>/status : status info on the process, used for instance by ‘ps’;
- /proc/<pid>/task : contains a subdirectory per process thread with info on the thread;
- /proc/<pid>/limits : shows system limits (file size, stack size, max. processes, max open files, …) for the process;

More info can be found in the man pages: man proc.

Summary: list of commands

- fuser
- ipcs
- kill
- killall
- lsof
- nproc
- pgrep
- pkill
- pidstat
- pmap
- /proc
- ps
- stat-family
- top