Linux Kernel Module for Listing Tasks

Team 35

Lu Yu 011972844

Gaochao Wang 013769431

Course: CMPE 180C Operating System Fall 2018

2018/11/15

**Table of Content**

[**1. Project Abstract**](#_gy57xh4jlujj) **2**

[**2. Executive Summary**](#_bbtp36xo3r0a) **2**

[**3. Introduction**](#_4bvjw2i6p2pv) **3**

[**4. Background and Objectives**](#_uoqk6zskj6kg) **3**

[4.1 Linux Kernel](#_6ho3zjjcucgo) 3

[4.2 Loadable Kernel Modules](#_h63uwe9mzdd6) 4

[4.3 Depth-First-Search](#_fct298ycestg) 4

[4.4 Objectives](#_gal1o6yz2cf0) 5

[**5. Approach**](#_i6o7fq48ny3h) **5**

[5.1 Environment Configuration](#_622ivjlx9j1n) 5

[5.2 Compiling Kernel Module](#_48hyvtkpydek) 6

[5.3 Loading and Removing Kernel Modules](#_814ogpga26y4) 7

[5.4 Understanding Kernel Data Structure](#_ndgjz6zcm0xk) 7

[5.5 Linear Order Listing Processes Implementation](#_4mz0wdalgpsw) 9

[5.6 Depth-First-Search Order Listing Processes Implementation](#_g3tbwa13m85i) 9

[**6. Result**](#_tidr9imvlp7q) **10**

[6.1 Linear Listing Processes](#_cfxqce6ya6a7) 10

[6.2 Depth-First-Search Order Listing Processes](#_dk5qsfhfycr) 15

[**7. Reference**](#_pph6r1n6nq8e) **25**

[**8. Appendix**](#_20nrc6oe8jq1) **26**

[8.1 Makefile](#_jg7uavagfd3h) 26

[8.2 Code](#_x2jzgefu93n6) 26

# 

# 1. Project Abstract

The goal of our project is to design and implement Linux kernel modules that can list out all the running tasks in a Linux system. Currently, there are several ways to view the current processes using Linux command line, ps and pstree. ps command shows PID – process ID, TTY – terminal type that the user is logged into, TIME – amount of CPU in minutes and seconds that the process has been running and CMD – name of the command that launched the process. pstree command shows the processes in a tree hierarchy. There are many options for the two command to show more details. In our project, we developed two kernel modules that lists out all process in Linux system in two different orders. One is linear order by process ID, the other is depth-first-search order by process relation tree.

**Weekly Schedule**

Week 1: review the programming project in the textbook Chapter 2 to learn how to create/load/remove a kernel module in Linux system

Week 2: develop/test the 1st kernel module for listing the running processes linearly. Compare the result with Linux command(ps -el) output.

Week 3: develop/test the 2nd kernel module, listing the running processes in the order of depth first search approach. Compare the result with Linux command(ps -eLf and pstree) output.

# 2. Executive Summary

The objective of this project is to develop two different Linux kernel modules using different search algorithms to list all the running processes in the operating system. Therefore, the entire project is mainly focused on the design and implementation of two loadable Linux kernel modules. This leads us to learn about the basics knowledge of development and implementation of Linux kernel modules in Linux virtual machine, which includes developing Linux kernel modules using C programming language, compiling the code with the terminal application, printing output to kernel log buffer, the data structure and basic commands of kernel. Finally, two Linux kernel modules, one using linear search algorithm and the other using depth-first search algorithm, are developed to list all the running processes in the operating system. And as a result, all the running processes are successfully displayed by both Linux kernel module. The performance difference of the two Linux kernel modules shows that how different kernel modules can affect the kernel behavior. Through the entire process, the proposed objectives have been achieved.

# 3. Introduction

The benefits of having a kernel module or just a function that lists the running processes in the operating system can be traced to everyday use of computers and different operating systems. Task Manager, and Activity Monitor are two examples of very key components in Windows and MacOS that have utilized this key functionality. Knowing what processes are currently running in your operating system provides great useful information in many different situations. We have knowledge of the Linux process Process Control Block with the task\_struct data structure. For example, determining process states (which are actually currently running in the operating system) is important when figuring out what process is currently taking up memory. Also, we can figure out which processes have what type of priority, who they are created by, what time they were created, etc. All this information can be important for process management.

This report shows how to develop and implement a Linux kernel module (KML) and discusses the performance of two search algorithm, linear search and depth-first-search, in listing all the running processes in the operating system.

# 4. Background and Objectives

## 4.1 Linux Kernel

The Linux kernel is an open-source monolithic Unix-like computer operating system kernel. The Linux family of operating systems is based on this kernel and deployed on both traditional computer systems and various embedded devices. It is the largest and most complex kernel over other types of kernels and is developed in C programming language. Although Linux kernel has some design flaws, developers are able to avoid these flaws by developing kernel modules that can add functionalities to the basic kernel as Linux kernel is an open-source operating system kernel.

Developing kernel modules is directly interacting with the kernel. It is important to be aware of that these are kernel code that invoke kernel functions and can crash the system if any error exists in the code.

## 4.2 Loadable Kernel Modules

Loadable kernel modules are object files that contain code to extend the kernel of an operating system. Since it is very difficult to fully predict what features and what functionalities an operating system will need when it is initially designed, the arrival of loadable kernel modules is extremely effective and advantageous in the sense that these modules can be loaded and removed from a running kernel, and it is not necessary to either recompile or reboot the kernel for new functionalities/features. In addition, LKM’s save main/physical memory by loading the kernel modules at run time. Typical examples of loadable kernel modules include device drivers to support new hardware or file systems, adding system calls, or in our case, listing all the running processes in an operating system. Also, when a loadable kernel module is no longer being used, and is no longer required, it can be removed to free memory and other resources.

## 4.3 Depth-First-Search

Depth-first search (DFS) is an traversing approach on a tree or graph data structures. The approach starts at the root node (any node in the case of a graph) and explores as far down as possible along each branch before backtracking to next branch.

In our project, a recursive method is applied to implement backtracking and DFS. Moreover, the backtracking means when you are moving forward and there are no more nodes along the current path, you move backwards on the same path to find nodes to traverse. All the nodes will be visited on the current path until all the unvisited nodes have been traversed. It starts with traversing nodes from root until no more children before moving onto the next sibling. It explores a path as far as possible, then backtracks. The most memory it can take up is the longest possible path.

## 4.4 Objectives

Objectives include learning how to create Linux kernel modules, understanding kernel linked-list data structures, understanding macro operations on lists, and how to iterate over tasks linearly and depth-first search process tree.

Deliverables include a kernel module that iterates through all processes and displays the process name, state, and process id of each task; another kernel module that does the same thing, but in a DFS order.

# 5. Approach

## 5.1 Environment Configuration

For this project, a virtual machine running Linux 3.1 kernel image with Debian (v 8.2) distribution is used.

If using other Linux kernel, the following two commands can be executed in super user mode to make sure essential components required for compiling the kernel module program have been installed.

sudo apt-get install build -essential

sudo apt- get install linux-headers-$(uname -r)

Otherwise, just follow the configuring procedure at <http://os-book.com/> to set up a Linux virtual machine that works well for compiling and running the kernel modules

5.2 Understanding Kernel Module Functions

Kernel modules must have at least two functions: an initialization function called init\_module() which is called when the module is insmoded into the kernel, and an cleanup function called cleanup\_module() which is called just before it is rmmoded. Starting with kernel 2.3.13, name of the start and end functions of a module can be customised, and this approach is preferred. The module entry point function must return an integer value, with 0 representing success and any other value representing failure. The module exit point function returns void. Neither the module entry point nor the module exit point is passed any parameters.

Using macro module\_init (function) and module\_exit (function) to registered the customized functions for the modul start and end functions. printk() is the kernel equivalent of printf(), and it writes output to kernel log buffer. printk() also allows us to specify a priority flag whose values are given in the <linux/printk.h> include file. In our case, the priority is KERN INFO, which is an informational message. “dmesg” command can be used to read kernel log buffer.

MODULE LICENSE(), MODULE DESCRIPTION(), and MODULE AUTHOR() are macro to write the software license, description and author. It is standard practice in developing kernel modules.

All these macro are defined in /linux/module.h.

## 5.2 Compiling Kernel Module

To compile the c programming code, we use Makefile to simply the operation. Please refer to the Makefile in the appendix to see the details. The "all" and "clean" targets were added to compile and delete generated files.

To compile the module, just enter the command line:

make

/lib/modules/$(shell uname -r)/build is the path to the kernel source, where uname-r is to print system kernel release information.

M=$PWD is to informs kbuild that an external module is being built. The value given to "M" is the absolute path of the directory where the external module (kbuild file) is located.( "Kbuild" is the build system used by the Linux kernel. Modules must use kbuild to stay compatible with changes in the build infrastructure.)

obj-m := <module\_name>.o is the parameter used by the kbuild system to build <module\_name>.o from <module\_name>.c, and after linking, will result in the kernel module <module\_name>.ko along with several other files.

To delete all the module related, just enter the command line:

make clean

## 5.3 Loading and Removing Kernel Modules

There are several commands we use to execute our project after code compilation.

**lsmod** lists all current kernel modules running, so you can check to see if your module is running.

**dmesg** command outputs the kernel log buffer to the terminal; use dmesg to check that your module is working correctly.

**insmod** is a command to insert a module into the kernel.

**rmmod** is a command to remove a module from the kernel.

## 5.4 Understanding Kernel Data Structure

In this project, some kernel data structure and C macros were used to list processes. We will illustrate them below.

1. struct list\_head, defined in linux/types.h, which is included in list.h and "module.h".

struct list\_head{  
 struct list\_head \*next;  
 struct list\_head \*prev;  
 }

The struct list\_head is used to put into another data structure to chain them together to form a linked list.

1. struct task\_struct, defined in linux/sched.h.

It is used to describe a specific process containing all the information about it. Among all the fields in the structure, there were 6 fields that are related to our project.

struct task\_struct{  
 …

char comm[TASK\_COMM\_LEN];

pid\_t pid;  
 volatile long state;  
 struct list\_head children;  
 struct list\_head sibling;

struct [list\_head](https://elixir.bootlin.com/linux/v4.14/ident/list_head) [tasks](https://elixir.bootlin.com/linux/v4.14/ident/tasks);

…

}

To understand these fields, let us take a look at the process family tree first. All processes are descendants of the init process (PID 1). The kernel starts init in the last step of the boot process. The init process reads the system initscripts and executes more programs, eventually completing the boot process.

Every process on the system has exactly one parent. Every process has zero or more children. Processes that are all direct children of the same parent are called siblings.

All process forms a circular linked list in the order of ID by the field struct list\_head tasks ; all the siblings of a parent process forms a circular linked list by the field struct list\_head sibling; and the field struct list\_head children is pointing to one of child processes.

The field comm is the command to start the process, which can also be the name of process; pid is the process ID; state is the process state.

1. for\_each\_process() is a macro defined in linux/sched.h, which iterates over the entire task/process circular linked list:

#define for\_each\_process(p) \

for (p = &init\_task ; (p = next\_task(p)) != &init\_task ; )

This macro is composed by several macros as listed below:

* 1. next\_task() is a macro defined in linux/sched.h, which returns the next task in the circular linked list:

#define next\_task(p) \

list\_entry((p)->tasks.next, struct task\_struct, tasks)

* 1. list\_entry() is a macro defined in linux/list.h, which finds the pointer of the struct type that contains the ptr as member:

/\*

\* list\_entry - get the struct for this entry

\* @ptr: the &struct list\_head pointer.

\* @type: the type of the struct this is embedded in.

\* @member: the name of the list\_struct within the struct.

\*/

#define list\_entry(ptr, type, member) \

container\_of(ptr, type, member)

* 1. container\_of() is the macro that implements the function of list\_entry, defined in linux/kernel.h:

#define container\_of(ptr, type, member) ({ \

const typeof( ((type \*)0)->member ) \*\_\_mptr = (ptr); \

(type \*)( (char \*)\_\_mptr - offsetof(type,member) );})

1. list\_for\_each() is a macro, defined in linux/list.h, which is used to traverse a circular linked list starting from head->next:

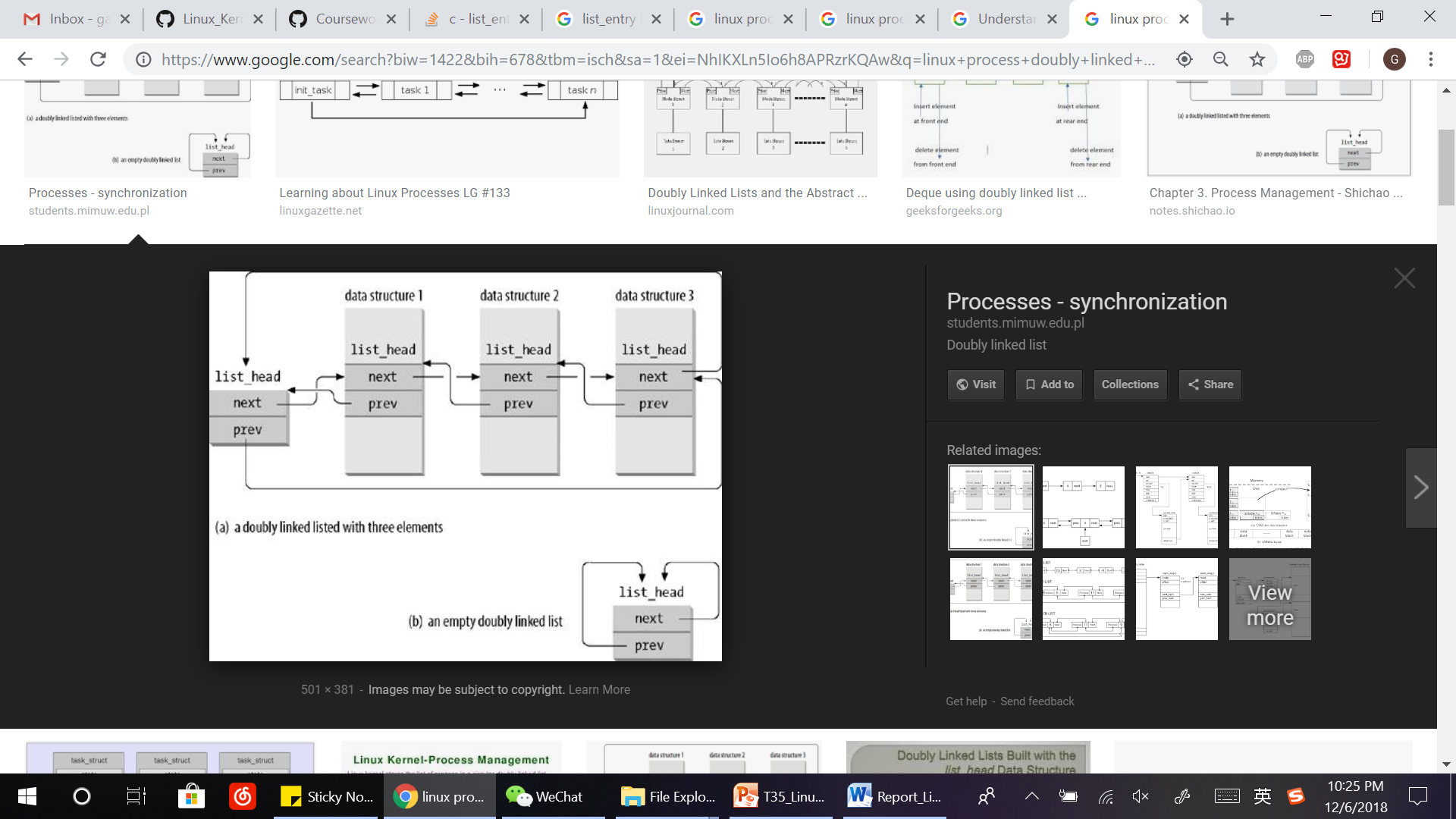
#define list\_for\_each(pos, head) \

for (pos = (head)->next; pos != (head); pos = pos->next)

## 5.5 Linear Order Listing Processes Implementation

The first module is to outputs process name, state, and process id of each task in the order of id. From the discussion in 5.4 Understanding Kernel Data Structure, we learnt that macro for\_each\_process(task) can be used to traverse all the process by the ID order. In each step of the iteration, we can print out the process id, name and state from the struct task\_struct fields. Please refer to the code section to understand this approach.

printk is used to print the information to kernel log buffer. dmesg command can be used to read kernel log buffer.



## 5.6 Depth-First-Search Order Listing Processes Implementation

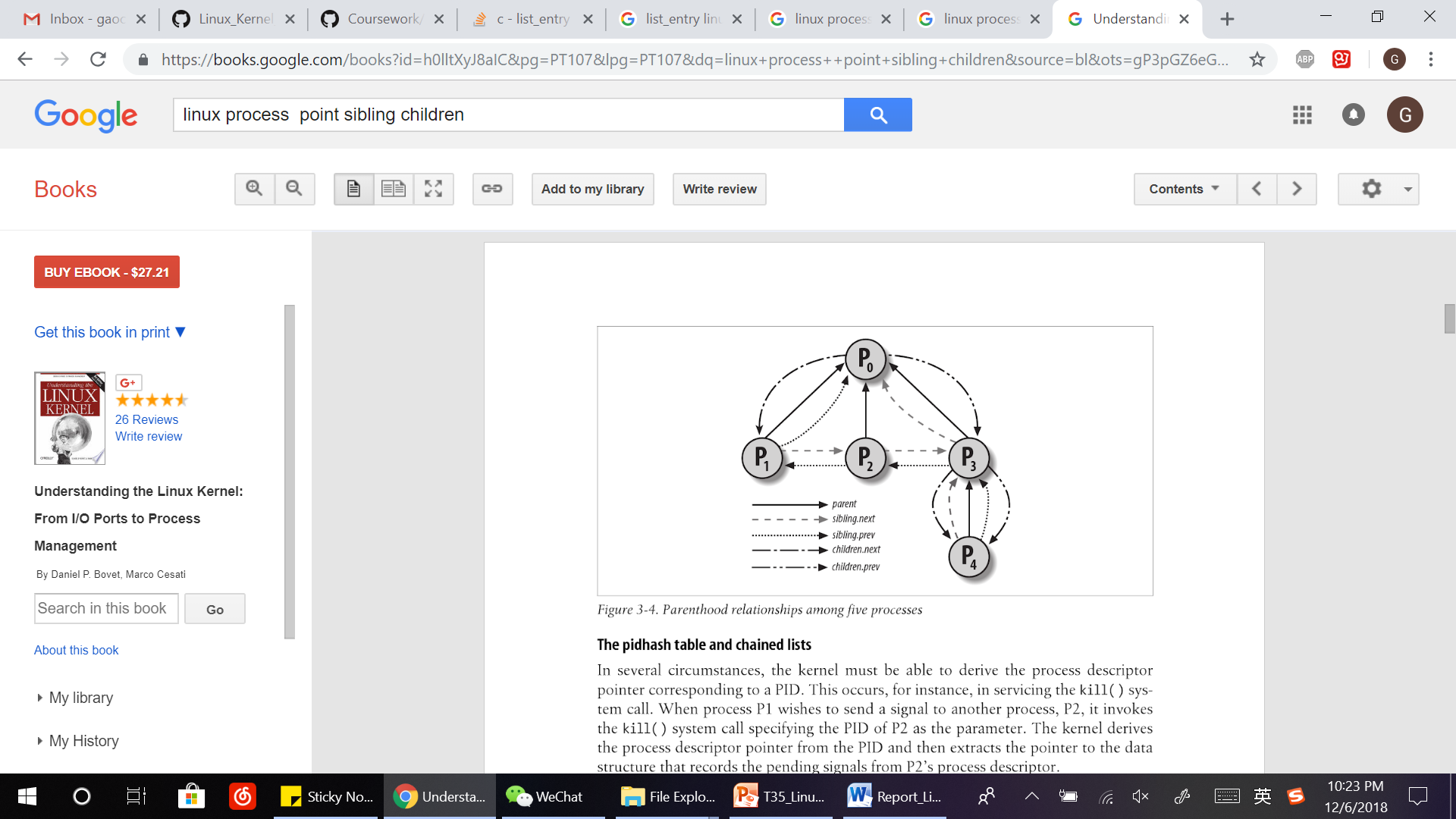
The second module is to outputs process name, state, and process id of each task in the order of DFS. From the discussion in 5.4 Understanding Kernel Data Structure, we learnt that the struct task\_struct has the tree information in the fields as children and sibling.

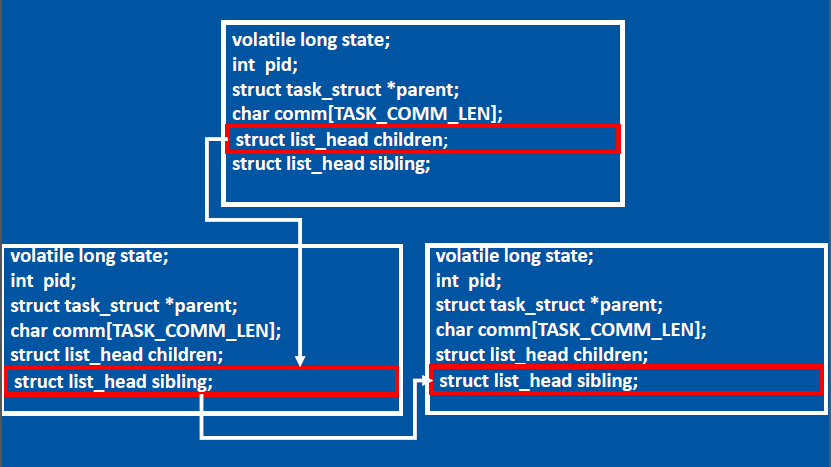
we start from root process, the root->children.next is the list\_head sibling of the child process. we can get 2 information from it.

* + - 1. The sibling.next will go to the next child of root, and so list\_for\_each will iterate all the children of root.
      2. list\_entry(children.next, struct task struct, silbing) will get the process whose sibling as the children.next, which is the child process of root.

If we process b first, we will go along the child path, and then go back to a., which is the DFS path. Finish the traversal in this manner, and output process information in each step. Then, we will meet the goal of output the process in the DFS order. Please refer to the code section to understand this approach.

As the same, printk and dmesg were used to write/read the processor information.





# 6. Result

## 6.1 Linear Listing Processes

After compiling the code and insearting the module, we can check and see the listing\_linear module is running by command lsmod.

oscreader@OSC:~/OS\_team35$ lsmod  
Module Size Used by  
listing\_linear 12436 0

Then remove the module, and use dmesg to check kernel buffer log.

[ 659.430764] listing\_linear\_module loaded  
[ 659.430769] P-ID Name State  
[ 659.430771] 1 systemd 1  
[ 659.430774] 2 kthreadd 1  
[ 659.430776] 3 ksoftirqd/0 1  
[ 659.430779] 4 kworker/0:0 1  
[ 659.430780] 5 kworker/0:0H 1  
[ 659.430783] 6 kworker/u2:0 1  
[ 659.430785] 7 rcu\_sched 1  
[ 659.430787] 8 rcu\_bh 1  
[ 659.430789] 9 migration/0 1  
[ 659.430792] 10 watchdog/0 1  
[ 659.430793] 11 khelper 1  
[ 659.430796] 12 kdevtmpfs 1  
[ 659.430798] 13 netns 1  
[ 659.430801] 14 khungtaskd 1  
[ 659.430802] 15 writeback 1  
[ 659.430805] 16 ksmd 1  
[ 659.430807] 17 khugepaged 1  
[ 659.430810] 18 crypto 1  
[ 659.430811] 19 kintegrityd 1  
[ 659.430814] 20 bioset 1  
[ 659.430816] 21 kblockd 1  
[ 659.430819] 23 kswapd0 1  
[ 659.430821] 24 fsnotify\_mark 1  
[ 659.430824] 30 kthrotld 1  
[ 659.430826] 31 ipv6\_addrconf 1  
[ 659.430829] 32 deferwq 1  
[ 659.430830] 66 khubd 1  
[ 659.430834] 67 ata\_sff 1  
[ 659.430835] 69 scsi\_eh\_0 1  
[ 659.430838] 70 scsi\_tmf\_0 1  
[ 659.430840] 71 scsi\_eh\_1 1  
[ 659.430843] 72 kworker/u2:2 1  
[ 659.430845] 73 scsi\_tmf\_1 1  
[ 659.430848] 74 scsi\_eh\_2 1  
[ 659.430850] 75 scsi\_tmf\_2 1  
[ 659.430853] 81 kworker/0:1H 1  
[ 659.430855] 105 jbd2/sda1-8 1  
[ 659.430858] 106 ext4-rsv-conver 1  
[ 659.430860] 137 kauditd 1  
[ 659.430864] 144 kworker/0:3 1  
[ 659.430866] 151 systemd-udevd 1  
[ 659.430869] 152 systemd-journal 1  
[ 659.430871] 207 kpsmoused 1  
[ 659.430874] 356 rpcbind 1  
[ 659.430876] 369 rpc.statd 1  
[ 659.430879] 377 rpciod 1  
[ 659.430881] 379 nfsiod 1  
[ 659.430884] 389 rpc.idmapd 1  
[ 659.430886] 404 accounts-daemon 1  
[ 659.430889] 405 cron 1  
[ 659.430891] 406 ModemManager 1  
[ 659.430896] 408 NetworkManager 1  
[ 659.430898] 410 sshd 1  
[ 659.430901] 411 atd 1  
[ 659.430903] 413 systemd-logind 1  
[ 659.430906] 419 avahi-daemon 1  
[ 659.430908] 420 dbus-daemon 1  
[ 659.430911] 435 avahi-daemon 1  
[ 659.430913] 459 rsyslogd 1  
[ 659.430916] 463 acpid 1  
[ 659.430918] 466 cupsd 1  
[ 659.430921] 467 cups-browsed 1  
[ 659.430923] 478 minissdpd 1  
[ 659.430926] 480 cfg80211 1  
[ 659.430928] 503 polkitd 1  
[ 659.430931] 505 agetty 1  
[ 659.430933] 536 gdm3 1  
[ 659.430936] 570 Xorg 1  
[ 659.430938] 758 exim4 1  
[ 659.430941] 819 systemd 1  
[ 659.430943] 820 (sd-pam) 1  
[ 659.430946] 850 upowerd 1  
[ 659.430948] 860 colord 1  
[ 659.430951] 867 pulseaudio 1  
[ 659.430953] 868 rtkit-daemon 1  
[ 659.430956] 897 wpa\_supplicant 1  
[ 659.430959] 900 packagekitd 1  
[ 659.430962] 914 gdm-session-wor 1  
[ 659.430964] 917 systemd 1  
[ 659.430966] 918 (sd-pam) 1  
[ 659.430969] 922 gnome-keyring-d 1  
[ 659.430972] 925 x-session-manag 1  
[ 659.430974] 962 ssh-agent 1  
[ 659.430977] 965 dbus-launch 1  
[ 659.430980] 966 dbus-daemon 1  
[ 659.430983] 969 at-spi-bus-laun 1  
[ 659.430985] 973 dbus-daemon 1  
[ 659.430989] 976 at-spi2-registr 1  
[ 659.430991] 987 gnome-settings- 1  
[ 659.430994] 1002 pulseaudio 1  
[ 659.430997] 1009 gvfsd 1  
[ 659.431001] 1010 start-pulseaudi 1  
[ 659.431003] 1011 xprop 1  
[ 659.431007] 1016 gvfsd-fuse 1  
[ 659.431009] 1024 gvfs-udisks2-vo 1  
[ 659.431012] 1026 udisksd 1  
[ 659.431014] 1035 gvfs-goa-volume 1  
[ 659.431018] 1038 goa-daemon 1  
[ 659.431020] 1045 mission-control 1  
[ 659.431023] 1047 gvfs-gphoto2-vo 1  
[ 659.431025] 1054 gvfs-mtp-volume 1  
[ 659.431028] 1058 gvfs-afc-volume 1  
[ 659.431030] 1062 gnome-shell 0  
[ 659.431033] 1067 gsd-printer 1  
[ 659.431035] 1080 gnome-shell-cal 1  
[ 659.431039] 1084 evolution-sourc 1  
[ 659.431040] 1097 evolution-alarm 1  
[ 659.431044] 1098 nm-applet 1  
[ 659.431046] 1100 tracker-miner-u 1  
[ 659.431049] 1102 nautilus 1  
[ 659.431051] 1104 tracker-extract 1  
[ 659.431055] 1107 tracker-store 1  
[ 659.431057] 1108 zeitgeist-datah 1  
[ 659.431102] 1109 tracker-miner-f 1  
[ 659.431105] 1111 tracker-miner-a 1  
[ 659.431108] 1112 applet.py 1  
[ 659.431110] 1126 zeitgeist-daemo 1  
[ 659.431113] 1139 zeitgeist-fts 1  
[ 659.431115] 1147 cat 1  
[ 659.431119] 1153 evolution-calen 1  
[ 659.431121] 1158 gconfd-2 1  
[ 659.431125] 1177 gvfsd-trash 1  
[ 659.431127] 1189 gvfsd-burn 1  
[ 659.431130] 1211 gvfsd-metadata 1  
[ 659.431132] 1257 dconf-service 1  
[ 659.431135] 1280 gnome-terminal- 1  
[ 659.431138] 1283 gnome-pty-helpe 1  
[ 659.431141] 1284 bash 1  
[ 659.431143] 1309 dhclient 1  
[ 659.431147] 1737 gedit 1  
[ 659.431149] 1975 kworker/0:1 1  
  
[ 666.295859] listing\_linear\_module un-loaded

In the above log, we can see the when the module is loaded and removed, and also the process information in the order of ID. We can check our result by comparing it with the bash command ps -el  
  
oscreader@OSC:~/OS\_team35$ ps -el  
F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY TIME CMD  
4 S 0 1 0 0 80 0 - 10081 ep\_pol ? 00:00:01 systemd  
1 S 0 2 0 0 80 0 - 0 kthrea ? 00:00:00 kthreadd  
1 S 0 3 2 0 80 0 - 0 smpboo ? 00:00:00 ksoftirqd/0  
1 S 0 5 2 0 60 -20 - 0 worker ? 00:00:00 kworker/0:0H  
1 S 0 6 2 0 80 0 - 0 worker ? 00:00:00 kworker/u2:0  
1 S 0 7 2 0 80 0 - 0 rcu\_gp ? 00:00:00 rcu\_sched  
1 S 0 8 2 0 80 0 - 0 rcu\_gp ? 00:00:00 rcu\_bh  
1 S 0 9 2 0 -40 - - 0 smpboo ? 00:00:00 migration/0  
5 S 0 10 2 0 -40 - - 0 smpboo ? 00:00:00 watchdog/0  
1 S 0 11 2 0 60 -20 - 0 rescue ? 00:00:00 khelper  
5 S 0 12 2 0 80 0 - 0 devtmp ? 00:00:00 kdevtmpfs  
1 S 0 13 2 0 60 -20 - 0 rescue ? 00:00:00 netns  
1 S 0 14 2 0 80 0 - 0 watchd ? 00:00:00 khungtaskd  
1 S 0 15 2 0 60 -20 - 0 rescue ? 00:00:00 writeback  
1 S 0 16 2 0 85 5 - 0 ksm\_sc ? 00:00:00 ksmd  
1 S 0 17 2 0 99 19 - 0 khugep ? 00:00:00 khugepaged  
1 S 0 18 2 0 60 -20 - 0 rescue ? 00:00:00 crypto  
1 S 0 19 2 0 60 -20 - 0 rescue ? 00:00:00 kintegrityd  
1 S 0 20 2 0 60 -20 - 0 rescue ? 00:00:00 bioset  
1 S 0 21 2 0 60 -20 - 0 rescue ? 00:00:00 kblockd  
1 S 0 23 2 0 80 0 - 0 kswapd ? 00:00:00 kswapd0  
1 S 0 24 2 0 80 0 - 0 fsnoti ? 00:00:00 fsnotify\_mark  
1 S 0 30 2 0 60 -20 - 0 rescue ? 00:00:00 kthrotld  
1 S 0 31 2 0 60 -20 - 0 rescue ? 00:00:00 ipv6\_addrconf  
1 S 0 32 2 0 60 -20 - 0 rescue ? 00:00:00 deferwq  
1 S 0 66 2 0 80 0 - 0 hub\_th ? 00:00:00 khubd  
1 S 0 67 2 0 60 -20 - 0 rescue ? 00:00:00 ata\_sff  
1 S 0 69 2 0 80 0 - 0 scsi\_e ? 00:00:00 scsi\_eh\_0  
1 S 0 70 2 0 60 -20 - 0 rescue ? 00:00:00 scsi\_tmf\_0  
1 S 0 71 2 0 80 0 - 0 scsi\_e ? 00:00:00 scsi\_eh\_1  
1 S 0 72 2 0 80 0 - 0 worker ? 00:00:00 kworker/u2:2  
1 S 0 73 2 0 60 -20 - 0 rescue ? 00:00:00 scsi\_tmf\_1  
1 S 0 74 2 0 80 0 - 0 scsi\_e ? 00:00:00 scsi\_eh\_2  
1 S 0 75 2 0 60 -20 - 0 rescue ? 00:00:00 scsi\_tmf\_2  
1 S 0 81 2 0 60 -20 - 0 worker ? 00:00:00 kworker/0:1H  
1 S 0 105 2 0 80 0 - 0 kjourn ? 00:00:00 jbd2/sda1-8  
1 S 0 106 2 0 60 -20 - 0 rescue ? 00:00:00 ext4-rsv-conve  
1 S 0 137 2 0 80 0 - 0 kaudit ? 00:00:00 kauditd  
1 S 0 144 2 0 80 0 - 0 worker ? 00:00:00 kworker/0:3  
4 S 0 151 1 0 80 0 - 3196 ep\_pol ? 00:00:00 systemd-udevd  
4 S 0 152 1 0 80 0 - 2098 ep\_pol ? 00:00:00 systemd-journa  
1 S 0 207 2 0 60 -20 - 0 rescue ? 00:00:00 kpsmoused  
5 S 0 356 1 0 80 0 - 1091 poll\_s ? 00:00:00 rpcbind  
5 S 107 369 1 0 80 0 - 1140 poll\_s ? 00:00:00 rpc.statd  
1 S 0 377 2 0 60 -20 - 0 rescue ? 00:00:00 rpciod  
1 S 0 379 2 0 60 -20 - 0 rescue ? 00:00:00 nfsiod  
1 S 0 389 1 0 80 0 - 727 ep\_pol ? 00:00:00 rpc.idmapd  
4 S 0 404 1 0 80 0 - 9118 poll\_s ? 00:00:00 accounts-daemo  
4 S 0 405 1 0 80 0 - 1248 hrtime ? 00:00:00 cron  
4 S 0 406 1 0 80 0 - 10005 poll\_s ? 00:00:00 ModemManager  
4 S 0 408 1 0 80 0 - 13882 poll\_s ? 00:00:00 NetworkManager  
4 S 0 410 1 0 80 0 - 2020 poll\_s ? 00:00:00 sshd  
4 S 1 411 1 0 80 0 - 659 hrtime ? 00:00:00 atd  
4 S 0 413 1 0 80 0 - 936 ep\_pol ? 00:00:00 systemd-logind  
4 S 105 419 1 0 80 0 - 887 poll\_s ? 00:00:00 avahi-daemon  
4 S 104 420 1 0 80 0 - 1449 ep\_pol ? 00:00:01 dbus-daemon  
1 S 105 435 419 0 80 0 - 887 unix\_s ? 00:00:00 avahi-daemon  
4 S 0 459 1 0 80 0 - 7771 poll\_s ? 00:00:00 rsyslogd  
0 S 0 463 1 0 80 0 - 546 poll\_s ? 00:00:00 acpid  
4 S 0 466 1 0 80 0 - 2972 ep\_pol ? 00:00:00 cupsd  
4 S 0 467 1 0 80 0 - 2501 poll\_s ? 00:00:00 cups-browsed  
5 S 0 478 1 0 80 0 - 541 poll\_s ? 00:00:00 minissdpd  
1 S 0 480 2 0 60 -20 - 0 rescue ? 00:00:00 cfg80211  
4 S 0 503 1 0 80 0 - 9077 poll\_s ? 00:00:00 polkitd  
4 S 0 505 1 0 80 0 - 1041 n\_tty\_ tty1 00:00:00 agetty  
4 S 0 536 1 0 80 0 - 11682 poll\_s ? 00:00:00 gdm3  
4 S 0 570 536 4 80 0 - 28881 poll\_s tty7 00:00:29 Xorg  
5 S 106 758 1 0 80 0 - 2476 poll\_s ? 00:00:00 exim4  
4 S 118 819 1 0 80 0 - 1206 ep\_pol ? 00:00:00 systemd  
5 S 118 820 819 0 80 0 - 1594 sigtim ? 00:00:00 (sd-pam)  
4 S 0 850 1 0 80 0 - 10430 poll\_s ? 00:00:00 upowerd  
4 S 108 860 1 0 80 0 - 10495 poll\_s ? 00:00:00 colord  
1 S 118 867 1 0 80 0 - 25071 poll\_s ? 00:00:00 pulseaudio  
4 S 114 868 1 0 81 1 - 5345 poll\_s ? 00:00:00 rtkit-daemon  
4 S 0 897 1 0 80 0 - 1909 poll\_s ? 00:00:00 wpa\_supplicant  
4 S 0 900 1 0 80 0 - 12821 poll\_s ? 00:00:00 packagekitd  
4 S 0 914 536 0 80 0 - 7403 poll\_s ? 00:00:00 gdm-session-wo  
4 S 1000 917 1 0 80 0 - 1206 ep\_pol ? 00:00:00 systemd  
5 S 1000 918 917 0 80 0 - 1594 sigtim ? 00:00:00 (sd-pam)  
1 S 1000 922 1 0 80 0 - 11664 poll\_s ? 00:00:00 gnome-keyring-  
4 S 1000 925 914 0 80 0 - 17253 poll\_s ? 00:00:00 x-session-mana  
1 S 1000 962 925 0 80 0 - 1103 poll\_s ? 00:00:00 ssh-agent  
1 S 1000 965 1 0 80 0 - 991 poll\_s ? 00:00:00 dbus-launch  
1 S 1000 966 1 0 80 0 - 1422 ep\_pol ? 00:00:00 dbus-daemon  
0 S 1000 969 1 0 80 0 - 10813 poll\_s ? 00:00:00 at-spi-bus-lau  
0 S 1000 973 969 0 80 0 - 1305 ep\_pol ? 00:00:00 dbus-daemon  
0 S 1000 976 1 0 80 0 - 4558 poll\_s ? 00:00:00 at-spi2-regist  
0 S 1000 987 925 0 80 0 - 62250 poll\_s ? 00:00:00 gnome-settings  
1 S 1000 1002 1 0 80 0 - 25087 poll\_s ? 00:00:00 pulseaudio  
0 S 1000 1009 1 0 80 0 - 7845 poll\_s ? 00:00:00 gvfsd  
1 S 1000 1010 1 0 80 0 - 565 wait ? 00:00:00 start-pulseaud  
0 S 1000 1011 1010 0 80 0 - 1573 poll\_s ? 00:00:00 xprop  
0 S 1000 1016 1 0 80 0 - 11408 futex\_ ? 00:00:00 gvfsd-fuse  
0 S 1000 1024 1 0 80 0 - 10197 poll\_s ? 00:00:00 gvfs-udisks2-v  
4 S 0 1026 1 0 80 0 - 13091 poll\_s ? 00:00:00 udisksd  
0 S 1000 1035 1 0 80 0 - 7614 poll\_s ? 00:00:00 gvfs-goa-volum  
0 S 1000 1038 1 0 80 0 - 35415 poll\_s ? 00:00:00 goa-daemon  
0 S 1000 1045 1 0 80 0 - 10641 poll\_s ? 00:00:00 mission-contro  
0 S 1000 1047 1 0 80 0 - 7759 poll\_s ? 00:00:00 gvfs-gphoto2-v  
0 S 1000 1054 1 0 80 0 - 7544 poll\_s ? 00:00:00 gvfs-mtp-volum  
0 S 1000 1058 1 0 80 0 - 10490 poll\_s ? 00:00:00 gvfs-afc-volum  
0 S 1000 1062 925 27 80 0 - 107409 poll\_s ? 00:03:02 gnome-shell  
0 S 1000 1067 1 0 80 0 - 14114 poll\_s ? 00:00:00 gsd-printer  
0 S 1000 1080 1 0 80 0 - 22980 poll\_s ? 00:00:00 gnome-shell-ca  
0 S 1000 1084 1 0 80 0 - 26243 poll\_s ? 00:00:00 evolution-sour  
0 S 1000 1097 925 0 80 0 - 41174 poll\_s ? 00:00:00 evolution-alar  
0 S 1000 1098 925 0 80 0 - 18213 poll\_s ? 00:00:00 nm-applet  
0 S 1000 1100 925 0 99 - - 10834 poll\_s ? 00:00:00 tracker-miner-  
0 S 1000 1102 925 0 80 0 - 30376 poll\_s ? 00:00:04 nautilus  
0 S 1000 1104 925 0 99 19 - 41069 poll\_s ? 00:00:00 tracker-extrac  
0 S 1000 1107 1 0 80 0 - 19731 poll\_s ? 00:00:00 tracker-store  
0 S 1000 1108 925 0 80 0 - 16390 poll\_s ? 00:00:03 zeitgeist-data  
0 S 1000 1109 925 0 99 19 - 24661 poll\_s ? 00:00:00 tracker-miner-  
0 S 1000 1111 925 0 99 - - 10900 poll\_s ? 00:00:00 tracker-miner-  
0 S 1000 1112 925 0 80 0 - 9647 poll\_s ? 00:00:00 applet.py  
0 S 1000 1126 1 0 80 0 - 11730 poll\_s ? 00:00:00 zeitgeist-daem  
0 S 1000 1139 1 0 80 0 - 17530 poll\_s ? 00:00:00 zeitgeist-fts  
0 S 1000 1147 1139 0 80 0 - 973 unix\_s ? 00:00:00 cat  
0 S 1000 1153 1 0 80 0 - 44184 poll\_s ? 00:00:00 evolution-cale  
0 S 1000 1158 1 0 80 0 - 2313 poll\_s ? 00:00:00 gconfd-2  
0 S 1000 1177 1 0 80 0 - 16136 poll\_s ? 00:00:00 gvfsd-trash  
0 S 1000 1189 1 0 80 0 - 9895 poll\_s ? 00:00:00 gvfsd-burn  
0 S 1000 1211 1 0 80 0 - 5252 poll\_s ? 00:00:00 gvfsd-metadata  
0 S 1000 1257 1 0 80 0 - 6278 poll\_s ? 00:00:00 dconf-service  
0 S 1000 1280 1 1 80 0 - 16517 poll\_s ? 00:00:07 gnome-terminal  
0 S 1000 1283 1280 0 80 0 - 601 unix\_s ? 00:00:00 gnome-pty-help  
0 S 1000 1284 1280 0 80 0 - 1658 wait pts/0 00:00:00 bash  
4 S 0 1309 408 0 80 0 - 2308 poll\_s ? 00:00:00 dhclient  
0 S 1000 1737 1 1 80 0 - 26354 poll\_s ? 00:00:08 gedit  
1 S 0 1975 2 0 80 0 - 0 worker ? 00:00:00 kworker/0:1

## 6.2 Depth-First-Search Order Listing Processes

Similarly we can get the result for listing DFS module, as shown below.  
  
oscreader@OSC:~/OS\_team35$ lsmod  
Module Size Used by  
listing\_dfs 12453 0   
  
  
[ 1669.634103] listing\_DFS\_module loaded  
[ 1669.634107] P-ID Name State  
[ 1669.634110] 0 swapper/0 0  
[ 1669.634113] 1 systemd 1  
[ 1669.634114] 151 systemd-udevd 1  
[ 1669.634117] 4073 systemd-udevd 0  
[ 1669.634118] 152 systemd-journal 1  
[ 1669.634121] 356 rpcbind 1  
[ 1669.634123] 369 rpc.statd 1  
[ 1669.634126] 389 rpc.idmapd 1  
[ 1669.634127] 404 accounts-daemon 1  
[ 1669.634130] 405 cron 1  
[ 1669.634131] 406 ModemManager 1  
[ 1669.634134] 408 NetworkManager 1  
[ 1669.634136] 1309 dhclient 1  
[ 1669.634139] 410 sshd 1  
[ 1669.634140] 411 atd 1  
[ 1669.634143] 413 systemd-logind 1  
[ 1669.634145] 419 avahi-daemon 1  
[ 1669.634147] 435 avahi-daemon 1  
[ 1669.634149] 420 dbus-daemon 1  
[ 1669.634152] 459 rsyslogd 1  
[ 1669.634154] 463 acpid 1  
[ 1669.634156] 466 cupsd 1  
[ 1669.634158] 467 cups-browsed 1  
[ 1669.634161] 478 minissdpd 1  
[ 1669.634162] 503 polkitd 1  
[ 1669.634165] 505 agetty 1  
[ 1669.634167] 536 gdm3 1  
[ 1669.634169] 570 Xorg 1  
[ 1669.634171] 914 gdm-session-wor 1  
[ 1669.634174] 925 x-session-manag 1  
[ 1669.634176] 962 ssh-agent 1  
[ 1669.634178] 987 gnome-settings- 1  
[ 1669.634180] 1062 gnome-shell 0  
[ 1669.634182] 1097 evolution-alarm 1  
[ 1669.634184] 1098 nm-applet 1  
[ 1669.634187] 1100 tracker-miner-u 1  
[ 1669.634189] 1102 nautilus 1  
[ 1669.634191] 1104 tracker-extract 1  
[ 1669.634193] 1108 zeitgeist-datah 1  
[ 1669.634196] 1109 tracker-miner-f 1  
[ 1669.634198] 1111 tracker-miner-a 1  
[ 1669.634200] 1112 applet.py 1  
[ 1669.634202] 758 exim4 1  
[ 1669.634205] 819 systemd 1  
[ 1669.634206] 820 (sd-pam) 1  
[ 1669.634209] 850 upowerd 1  
[ 1669.634211] 860 colord 1  
[ 1669.634213] 868 rtkit-daemon 1  
[ 1669.634215] 867 pulseaudio 1  
[ 1669.634218] 897 wpa\_supplicant 1  
[ 1669.634220] 900 packagekitd 1  
[ 1669.634222] 917 systemd 1  
[ 1669.634224] 918 (sd-pam) 1  
[ 1669.634226] 922 gnome-keyring-d 1  
[ 1669.634228] 966 dbus-daemon 1  
[ 1669.634231] 965 dbus-launch 1  
[ 1669.634233] 969 at-spi-bus-laun 1  
[ 1669.634235] 973 dbus-daemon 1  
[ 1669.634237] 976 at-spi2-registr 1  
[ 1669.634240] 1002 pulseaudio 1  
[ 1669.634241] 1010 start-pulseaudi 1  
[ 1669.634244] 1011 xprop 1  
[ 1669.634246] 1009 gvfsd 1  
[ 1669.634249] 1016 gvfsd-fuse 1  
[ 1669.634250] 1026 udisksd 1  
[ 1669.634253] 1024 gvfs-udisks2-vo 1  
[ 1669.634255] 1038 goa-daemon 1  
[ 1669.634258] 1035 gvfs-goa-volume 1  
[ 1669.634259] 1047 gvfs-gphoto2-vo 1  
[ 1669.634262] 1054 gvfs-mtp-volume 1  
[ 1669.634263] 1045 mission-control 1  
[ 1669.634266] 1058 gvfs-afc-volume 1  
[ 1669.634268] 1067 gsd-printer 1  
[ 1669.634271] 1084 evolution-sourc 1  
[ 1669.634272] 1080 gnome-shell-cal 1  
[ 1669.634275] 1107 tracker-store 1  
[ 1669.634277] 1126 zeitgeist-daemo 1  
[ 1669.634280] 1139 zeitgeist-fts 1  
[ 1669.634281] 1147 cat 1  
[ 1669.634284] 1158 gconfd-2 1  
[ 1669.634286] 1177 gvfsd-trash 1  
[ 1669.634288] 1189 gvfsd-burn 1  
[ 1669.634290] 1153 evolution-calen 1  
[ 1669.634293] 1211 gvfsd-metadata 1  
[ 1669.634294] 1257 dconf-service 1  
[ 1669.634298] 1280 gnome-terminal- 1  
[ 1669.634299] 1283 gnome-pty-helpe 1  
[ 1669.634302] 1284 bash 1  
[ 1669.634304] 4071 sudo 1  
[ 1669.634306] 4072 insmod 0  
[ 1669.634308] 3170 gedit 1  
[ 1669.634311] 2 kthreadd 1  
[ 1669.634312] 3 ksoftirqd/0 1  
[ 1669.634380] 5 kworker/0:0H 1  
[ 1669.634382] 6 kworker/u2:0 1  
[ 1669.634385] 7 rcu\_sched 1  
[ 1669.634386] 8 rcu\_bh 1  
[ 1669.634389] 9 migration/0 1  
[ 1669.634390] 10 watchdog/0 1  
[ 1669.634393] 11 khelper 1  
[ 1669.634395] 12 kdevtmpfs 1  
[ 1669.634397] 13 netns 1  
[ 1669.634399] 14 khungtaskd 1  
[ 1669.634401] 15 writeback 1  
[ 1669.634403] 16 ksmd 1  
[ 1669.634406] 17 khugepaged 1  
[ 1669.634407] 18 crypto 1  
[ 1669.634410] 19 kintegrityd 1  
[ 1669.634411] 20 bioset 1  
[ 1669.634414] 21 kblockd 1  
[ 1669.634415] 23 kswapd0 1  
[ 1669.634418] 24 fsnotify\_mark 1  
[ 1669.634420] 30 kthrotld 1  
[ 1669.634422] 31 ipv6\_addrconf 1  
[ 1669.634424] 32 deferwq 1  
[ 1669.634427] 66 khubd 1  
[ 1669.634428] 67 ata\_sff 1  
[ 1669.634431] 69 scsi\_eh\_0 1  
[ 1669.634433] 70 scsi\_tmf\_0 1  
[ 1669.634435] 71 scsi\_eh\_1 1  
[ 1669.634437] 72 kworker/u2:2 1  
[ 1669.634439] 73 scsi\_tmf\_1 1  
[ 1669.634441] 74 scsi\_eh\_2 1  
[ 1669.634444] 75 scsi\_tmf\_2 1  
[ 1669.634445] 81 kworker/0:1H 1  
[ 1669.634448] 105 jbd2/sda1-8 1  
[ 1669.634449] 106 ext4-rsv-conver 1  
[ 1669.634452] 137 kauditd 1  
[ 1669.634454] 144 kworker/0:3 1  
[ 1669.634456] 207 kpsmoused 1  
[ 1669.634458] 377 rpciod 1  
[ 1669.634461] 379 nfsiod 1  
[ 1669.634462] 480 cfg80211 1  
[ 1669.634465] 3158 kworker/0:0 1  
[ 1669.634467] 4066 kworker/0:1 1  
[ 1719.222799] listing\_DFS\_module un-loaded  
  
 Similarly, in the above log we can see the when the module is loaded and removed, and also the process information in the order of DFS. We can check our result by comparing it with the bash command pstree and ps -eLf.

oscreader@OSC:~/OS\_team35$ pstree  
systemd─┬─ModemManager─┬─{gdbus}  
 │ └─{gmain}  
 ├─NetworkManager─┬─dhclient  
 │ ├─{NetworkManager}  
 │ ├─{gdbus}  
 │ └─{gmain}  
 ├─accounts-daemon─┬─{gdbus}  
 │ └─{gmain}  
 ├─acpid  
 ├─agetty  
 ├─at-spi-bus-laun─┬─dbus-daemon  
 │ ├─{dconf worker}  
 │ ├─{gdbus}  
 │ └─{gmain}  
 ├─at-spi2-registr───{gdbus}  
 ├─atd  
 ├─avahi-daemon───avahi-daemon  
 ├─colord─┬─{gdbus}  
 │ └─{gmain}  
 ├─cron  
 ├─cups-browsed  
 ├─cupsd  
 ├─2\*[dbus-daemon]  
 ├─dbus-launch  
 ├─dconf-service─┬─{gdbus}  
 │ └─{gmain}  
 ├─evolution-calen─┬─{dconf worker}  
 │ ├─{evolution-calen}  
 │ ├─{gdbus}  
 │ ├─{gmain}  
 │ └─{pool}  
 ├─evolution-sourc─┬─{gdbus}  
 │ └─{gmain}  
 ├─exim4  
 ├─gconfd-2  
 ├─gdm3─┬─Xorg  
 │ ├─gdm-session-wor─┬─x-session-manag─┬─applet.py───{gmain}  
 │ │ │ ├─evolution-alarm─┬─{cal-client-db+  
 │ │ │ │ ├─{dconf worker}  
 │ │ │ │ ├─{evolution-ala+  
 │ │ │ │ └─{gdbus}  
 │ │ │ ├─gnome-settings-─┬─{dconf worker}  
 │ │ │ │ ├─{gdbus}  
 │ │ │ │ ├─{gmain}  
 │ │ │ │ ├─{pool}  
 │ │ │ │ └─{threaded-ml}  
 │ │ │ ├─gnome-shell─┬─{JS GC Helper}  
 │ │ │ │ ├─{JS Sour~ Thread}  
 │ │ │ │ ├─{dconf worker}  
 │ │ │ │ ├─{gdbus}  
 │ │ │ │ ├─{gmain}  
 │ │ │ │ ├─{pool}  
 │ │ │ │ └─{threaded-ml}  
 │ │ │ ├─nautilus─┬─{dconf worker}  
 │ │ │ │ ├─{gdbus}  
 │ │ │ │ └─{gmain}  
 │ │ │ ├─nm-applet─┬─{dconf worker}  
 │ │ │ │ └─{gdbus}  
 │ │ │ ├─ssh-agent  
 │ │ │ ├─tracker-extract─┬─{dconf worker}  
 │ │ │ │ ├─{gdbus}  
 │ │ │ │ ├─{gmain}  
 │ │ │ │ └─10\*[{pool}]  
 │ │ │ ├─tracker-miner-a─┬─{gdbus}  
 │ │ │ │ └─{gmain}  
 │ │ │ ├─tracker-miner-f─┬─{dconf worker}  
 │ │ │ │ ├─{gdbus}  
 │ │ │ │ └─{gmain}  
 │ │ │ ├─tracker-miner-u─┬─{gdbus}  
 │ │ │ │ └─{gmain}  
 │ │ │ ├─zeitgeist-datah─┬─{gdbus}  
 │ │ │ │ ├─{gmain}  
 │ │ │ │ └─{pool}  
 │ │ │ ├─{dconf worker}  
 │ │ │ ├─{gdbus}  
 │ │ │ └─{gmain}  
 │ │ ├─{gdbus}  
 │ │ └─{gmain}  
 │ ├─{gdbus}  
 │ ├─{gdm SIGUSR1 cat}  
 │ └─{gmain}  
 ├─gedit─┬─{dconf worker}  
 │ ├─{gdbus}  
 │ └─{gmain}  
 ├─gnome-keyring-d─┬─{dconf worker}  
 │ ├─{gdbus}  
 │ ├─{gmain}  
 │ └─{timer}  
 ├─gnome-shell-cal─┬─{cal-client-dbus}  
 │ ├─{dconf worker}  
 │ ├─{gdbus}  
 │ └─{gnome-shell-cal}  
 ├─gnome-terminal-─┬─bash───pstree  
 │ ├─gnome-pty-helpe  
 │ ├─{dconf worker}  
 │ ├─{gdbus}  
 │ └─{gmain}  
 ├─goa-daemon─┬─{gdbus}  
 │ ├─{gmain}  
 │ └─{pool}  
 ├─gsd-printer───{gdbus}  
 ├─gvfs-afc-volume─┬─{gdbus}  
 │ └─{gvfs-afc-volume}  
 ├─gvfs-goa-volume───{gdbus}  
 ├─gvfs-gphoto2-vo───{gdbus}  
 ├─gvfs-mtp-volume───{gdbus}  
 ├─gvfs-udisks2-vo─┬─{gdbus}  
 │ └─{gmain}  
 ├─gvfsd───{gdbus}  
 ├─gvfsd-burn─┬─{gdbus}  
 │ └─{pool}  
 ├─gvfsd-fuse─┬─{gdbus}  
 │ ├─{gvfs-fuse-sub}  
 │ └─2\*[{gvfsd-fuse}]  
 ├─gvfsd-metadata───{gdbus}  
 ├─gvfsd-trash─┬─{gdbus}  
 │ ├─{gmain}  
 │ └─3\*[{pool}]  
 ├─minissdpd  
 ├─mission-control─┬─{dconf worker}  
 │ └─{gdbus}  
 ├─packagekitd─┬─{gdbus}  
 │ └─{gmain}  
 ├─polkitd─┬─{gdbus}  
 │ └─{gmain}  
 ├─2\*[pulseaudio─┬─{alsa-sink-Intel}]  
 │ └─{alsa-source-Int}]  
 ├─rpc.idmapd  
 ├─rpc.statd  
 ├─rpcbind  
 ├─rsyslogd─┬─{in:imklog}  
 │ ├─{in:imuxsock}  
 │ └─{rs:main Q:Reg}  
 ├─rtkit-daemon───2\*[{rtkit-daemon}]  
 ├─sshd  
 ├─start-pulseaudi───xprop  
 ├─2\*[systemd───(sd-pam)]  
 ├─systemd-journal  
 ├─systemd-logind  
 ├─systemd-udevd  
 ├─tracker-store─┬─{dconf worker}  
 │ ├─{gdbus}  
 │ ├─{gmain}  
 │ └─4\*[{pool}]  
 ├─udisksd─┬─{cleanup}  
 │ ├─{gdbus}  
 │ ├─{gmain}  
 │ └─{probing-thread}  
 ├─upowerd─┬─{gdbus}  
 │ └─{gmain}  
 ├─wpa\_supplicant  
 ├─zeitgeist-daemo───{gdbus}  
 └─zeitgeist-fts─┬─cat  
 ├─{gdbus}  
 └─{gmain}  
  
  
  
oscreader@OSC:~/OS\_team35$ ps -eLf  
UID PID PPID LWP C NLWP STIME TTY TIME CMD  
root 1 0 1 0 1 21:46 ? 00:00:01 /sbin/init  
root 2 0 2 0 1 21:46 ? 00:00:00 [kthreadd]  
root 3 2 3 0 1 21:46 ? 00:00:00 [ksoftirqd/0]  
root 5 2 5 0 1 21:46 ? 00:00:00 [kworker/0:0H]  
root 6 2 6 0 1 21:46 ? 00:00:00 [kworker/u2:0]  
root 7 2 7 0 1 21:46 ? 00:00:00 [rcu\_sched]  
root 8 2 8 0 1 21:46 ? 00:00:00 [rcu\_bh]  
root 9 2 9 0 1 21:46 ? 00:00:00 [migration/0]  
root 10 2 10 0 1 21:46 ? 00:00:00 [watchdog/0]  
root 11 2 11 0 1 21:46 ? 00:00:00 [khelper]  
root 12 2 12 0 1 21:46 ? 00:00:00 [kdevtmpfs]  
root 13 2 13 0 1 21:46 ? 00:00:00 [netns]  
root 14 2 14 0 1 21:46 ? 00:00:00 [khungtaskd]  
root 15 2 15 0 1 21:46 ? 00:00:00 [writeback]  
root 16 2 16 0 1 21:46 ? 00:00:00 [ksmd]  
root 17 2 17 0 1 21:46 ? 00:00:00 [khugepaged]  
root 18 2 18 0 1 21:46 ? 00:00:00 [crypto]  
root 19 2 19 0 1 21:46 ? 00:00:00 [kintegrityd]  
root 20 2 20 0 1 21:46 ? 00:00:00 [bioset]  
root 21 2 21 0 1 21:46 ? 00:00:00 [kblockd]  
root 23 2 23 0 1 21:46 ? 00:00:00 [kswapd0]  
root 24 2 24 0 1 21:46 ? 00:00:00 [fsnotify\_mark]  
root 30 2 30 0 1 21:46 ? 00:00:00 [kthrotld]  
root 31 2 31 0 1 21:46 ? 00:00:00 [ipv6\_addrconf]  
root 32 2 32 0 1 21:46 ? 00:00:00 [deferwq]  
root 66 2 66 0 1 21:46 ? 00:00:00 [khubd]  
root 67 2 67 0 1 21:46 ? 00:00:00 [ata\_sff]  
root 69 2 69 0 1 21:46 ? 00:00:00 [scsi\_eh\_0]  
root 70 2 70 0 1 21:46 ? 00:00:00 [scsi\_tmf\_0]  
root 71 2 71 0 1 21:46 ? 00:00:00 [scsi\_eh\_1]  
root 72 2 72 0 1 21:46 ? 00:00:00 [kworker/u2:2]  
root 73 2 73 0 1 21:46 ? 00:00:00 [scsi\_tmf\_1]  
root 74 2 74 0 1 21:46 ? 00:00:00 [scsi\_eh\_2]  
root 75 2 75 0 1 21:46 ? 00:00:00 [scsi\_tmf\_2]  
root 81 2 81 0 1 21:46 ? 00:00:00 [kworker/0:1H]  
root 105 2 105 0 1 21:46 ? 00:00:00 [jbd2/sda1-8]  
root 106 2 106 0 1 21:46 ? 00:00:00 [ext4-rsv-conver]  
root 137 2 137 0 1 21:46 ? 00:00:00 [kauditd]  
root 144 2 144 0 1 21:46 ? 00:00:00 [kworker/0:3]  
root 151 1 151 0 1 21:46 ? 00:00:00 /lib/systemd/systemd-udevd  
root 152 1 152 0 1 21:46 ? 00:00:01 /lib/systemd/systemd-journa  
root 207 2 207 0 1 21:46 ? 00:00:00 [kpsmoused]  
root 356 1 356 0 1 21:46 ? 00:00:00 /sbin/rpcbind -w  
statd 369 1 369 0 1 21:46 ? 00:00:00 /sbin/rpc.statd  
root 377 2 377 0 1 21:46 ? 00:00:00 [rpciod]  
root 379 2 379 0 1 21:46 ? 00:00:00 [nfsiod]  
root 389 1 389 0 1 21:46 ? 00:00:00 /usr/sbin/rpc.idmapd  
root 404 1 404 0 3 21:46 ? 00:00:00 /usr/lib/accountsservice/ac  
root 404 1 434 0 3 21:46 ? 00:00:00 /usr/lib/accountsservice/ac  
root 404 1 454 0 3 21:46 ? 00:00:00 /usr/lib/accountsservice/ac  
root 405 1 405 0 1 21:46 ? 00:00:00 /usr/sbin/cron -f  
root 406 1 406 0 3 21:46 ? 00:00:00 /usr/sbin/ModemManager  
root 406 1 436 0 3 21:46 ? 00:00:00 /usr/sbin/ModemManager  
root 406 1 455 0 3 21:46 ? 00:00:00 /usr/sbin/ModemManager  
root 408 1 408 0 4 21:46 ? 00:00:00 /usr/sbin/NetworkManager --  
root 408 1 456 0 4 21:46 ? 00:00:00 /usr/sbin/NetworkManager --  
root 408 1 484 0 4 21:46 ? 00:00:00 /usr/sbin/NetworkManager --  
root 408 1 496 0 4 21:46 ? 00:00:00 /usr/sbin/NetworkManager --  
root 410 1 410 0 1 21:46 ? 00:00:00 /usr/sbin/sshd -D  
daemon 411 1 411 0 1 21:46 ? 00:00:00 /usr/sbin/atd -f  
root 413 1 413 0 1 21:46 ? 00:00:00 /lib/systemd/systemd-logind  
avahi 419 1 419 0 1 21:46 ? 00:00:00 avahi-daemon: running [OSC.  
message+ 420 1 420 0 1 21:46 ? 00:00:01 /usr/bin/dbus-daemon --syst  
avahi 435 419 435 0 1 21:46 ? 00:00:00 avahi-daemon: chroot helper  
root 459 1 459 0 4 21:46 ? 00:00:00 /usr/sbin/rsyslogd -n  
root 459 1 490 0 4 21:46 ? 00:00:00 /usr/sbin/rsyslogd -n  
root 459 1 491 0 4 21:46 ? 00:00:00 /usr/sbin/rsyslogd -n  
root 459 1 492 0 4 21:46 ? 00:00:00 /usr/sbin/rsyslogd -n  
root 463 1 463 0 1 21:46 ? 00:00:00 /usr/sbin/acpid  
root 466 1 466 0 1 21:46 ? 00:00:00 /usr/sbin/cupsd -f  
root 467 1 467 0 1 21:46 ? 00:00:00 /usr/sbin/cups-browsed  
root 478 1 478 0 1 21:46 ? 00:00:00 /usr/sbin/minissdpd -i 0.0.  
root 480 2 480 0 1 21:46 ? 00:00:00 [cfg80211]  
root 503 1 503 0 3 21:46 ? 00:00:00 /usr/lib/policykit-1/polkit  
root 503 1 515 0 3 21:46 ? 00:00:00 /usr/lib/policykit-1/polkit  
root 503 1 518 0 3 21:46 ? 00:00:00 /usr/lib/policykit-1/polkit  
root 505 1 505 0 1 21:46 tty1 00:00:00 /sbin/agetty --noclear tty1  
root 536 1 536 0 4 21:46 ? 00:00:00 /usr/sbin/gdm3  
root 536 1 553 0 4 21:46 ? 00:00:00 /usr/sbin/gdm3  
root 536 1 556 0 4 21:46 ? 00:00:00 /usr/sbin/gdm3  
root 536 1 567 0 4 21:46 ? 00:00:00 /usr/sbin/gdm3  
root 570 536 570 3 1 21:46 tty7 00:00:59 /usr/bin/Xorg :0 -novtswitc  
Debian-+ 758 1 758 0 1 21:46 ? 00:00:00 /usr/sbin/exim4 -bd -q30m  
Debian-+ 819 1 819 0 1 21:46 ? 00:00:00 /lib/systemd/systemd --user  
Debian-+ 820 819 820 0 1 21:46 ? 00:00:00 (sd-pam)   
root 850 1 850 0 3 21:46 ? 00:00:00 /usr/lib/upower/upowerd  
root 850 1 851 0 3 21:46 ? 00:00:00 /usr/lib/upower/upowerd  
root 850 1 852 0 3 21:46 ? 00:00:00 /usr/lib/upower/upowerd  
colord 860 1 860 0 3 21:46 ? 00:00:00 /usr/lib/colord/colord  
colord 860 1 863 0 3 21:46 ? 00:00:00 /usr/lib/colord/colord  
colord 860 1 864 0 3 21:46 ? 00:00:00 /usr/lib/colord/colord  
Debian-+ 867 1 867 0 3 21:46 ? 00:00:00 /usr/bin/pulseaudio --start  
Debian-+ 867 1 872 0 3 21:46 ? 00:00:00 /usr/bin/pulseaudio --start  
Debian-+ 867 1 873 0 3 21:46 ? 00:00:00 /usr/bin/pulseaudio --start  
rtkit 868 1 868 0 3 21:46 ? 00:00:00 /usr/lib/rtkit/rtkit-daemon  
rtkit 868 1 869 0 3 21:46 ? 00:00:00 /usr/lib/rtkit/rtkit-daemon  
rtkit 868 1 870 0 3 21:46 ? 00:00:00 /usr/lib/rtkit/rtkit-daemon  
root 897 1 897 0 1 21:46 ? 00:00:00 /sbin/wpa\_supplicant -u -s   
root 900 1 900 0 3 21:46 ? 00:00:00 /usr/lib/packagekit/package  
root 900 1 901 0 3 21:46 ? 00:00:00 /usr/lib/packagekit/package  
root 900 1 903 0 3 21:46 ? 00:00:00 /usr/lib/packagekit/package  
root 914 536 914 0 3 21:47 ? 00:00:00 gdm-session-worker [pam/gdm  
root 914 536 915 0 3 21:47 ? 00:00:00 gdm-session-worker [pam/gdm  
root 914 536 916 0 3 21:47 ? 00:00:00 gdm-session-worker [pam/gdm  
oscread+ 917 1 917 0 1 21:47 ? 00:00:00 /lib/systemd/systemd --user  
oscread+ 918 917 918 0 1 21:47 ? 00:00:00 (sd-pam)   
oscread+ 922 1 922 0 5 21:47 ? 00:00:00 /usr/bin/gnome-keyring-daem  
oscread+ 922 1 923 0 5 21:47 ? 00:00:00 /usr/bin/gnome-keyring-daem  
oscread+ 922 1 991 0 5 21:47 ? 00:00:00 /usr/bin/gnome-keyring-daem  
oscread+ 922 1 992 0 5 21:47 ? 00:00:00 /usr/bin/gnome-keyring-daem  
oscread+ 922 1 995 0 5 21:47 ? 00:00:00 /usr/bin/gnome-keyring-daem  
oscread+ 925 914 925 0 4 21:47 ? 00:00:00 x-session-manager  
oscread+ 925 914 977 0 4 21:47 ? 00:00:00 x-session-manager  
oscread+ 925 914 978 0 4 21:47 ? 00:00:00 x-session-manager  
oscread+ 925 914 981 0 4 21:47 ? 00:00:00 x-session-manager  
oscread+ 962 925 962 0 1 21:47 ? 00:00:00 /usr/bin/ssh-agent /usr/bin  
oscread+ 965 1 965 0 1 21:47 ? 00:00:00 /usr/bin/dbus-launch --exit  
oscread+ 966 1 966 0 1 21:47 ? 00:00:00 /usr/bin/dbus-daemon --fork  
oscread+ 969 1 969 0 4 21:47 ? 00:00:00 /usr/lib/at-spi2-core/at-sp  
oscread+ 969 1 970 0 4 21:47 ? 00:00:00 /usr/lib/at-spi2-core/at-sp  
oscread+ 969 1 972 0 4 21:47 ? 00:00:00 /usr/lib/at-spi2-core/at-sp  
oscread+ 969 1 974 0 4 21:47 ? 00:00:00 /usr/lib/at-spi2-core/at-sp  
oscread+ 973 969 973 0 1 21:47 ? 00:00:00 /usr/bin/dbus-daemon --conf  
oscread+ 976 1 976 0 2 21:47 ? 00:00:00 /usr/lib/at-spi2-core/at-sp  
oscread+ 976 1 980 0 2 21:47 ? 00:00:00 /usr/lib/at-spi2-core/at-sp  
oscread+ 987 925 987 0 6 21:47 ? 00:00:00 /usr/lib/gnome-settings-dae  
oscread+ 987 925 997 0 6 21:47 ? 00:00:00 /usr/lib/gnome-settings-dae  
oscread+ 987 925 998 0 6 21:47 ? 00:00:00 /usr/lib/gnome-settings-dae  
oscread+ 987 925 1000 0 6 21:47 ? 00:00:00 /usr/lib/gnome-settings-dae  
oscread+ 987 925 1014 0 6 21:47 ? 00:00:00 /usr/lib/gnome-settings-dae  
oscread+ 987 925 1729 0 6 21:50 ? 00:00:00 /usr/lib/gnome-settings-dae  
oscread+ 1002 1 1002 0 3 21:47 ? 00:00:00 /usr/bin/pulseaudio --start  
oscread+ 1002 1 1003 0 3 21:47 ? 00:00:00 /usr/bin/pulseaudio --start  
oscread+ 1002 1 1004 0 3 21:47 ? 00:00:00 /usr/bin/pulseaudio --start  
oscread+ 1009 1 1009 0 2 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd  
oscread+ 1009 1 1012 0 2 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd  
oscread+ 1010 1 1010 0 1 21:47 ? 00:00:00 /bin/sh /usr/bin/start-puls  
oscread+ 1011 1010 1011 0 1 21:47 ? 00:00:00 /usr/bin/xprop -root -spy  
oscread+ 1016 1 1016 0 5 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-fuse /r  
oscread+ 1016 1 1019 0 5 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-fuse /r  
oscread+ 1016 1 1020 0 5 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-fuse /r  
oscread+ 1016 1 1021 0 5 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-fuse /r  
oscread+ 1016 1 1022 0 5 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-fuse /r  
oscread+ 1024 1 1024 0 3 21:47 ? 00:00:00 /usr/lib/gvfs/gvfs-udisks2-  
oscread+ 1024 1 1025 0 3 21:47 ? 00:00:00 /usr/lib/gvfs/gvfs-udisks2-  
oscread+ 1024 1 1032 0 3 21:47 ? 00:00:00 /usr/lib/gvfs/gvfs-udisks2-  
root 1026 1 1026 0 5 21:47 ? 00:00:00 /usr/lib/udisks2/udisksd --  
root 1026 1 1027 0 5 21:47 ? 00:00:00 /usr/lib/udisks2/udisksd --  
root 1026 1 1029 0 5 21:47 ? 00:00:00 /usr/lib/udisks2/udisksd --  
root 1026 1 1030 0 5 21:47 ? 00:00:00 /usr/lib/udisks2/udisksd --  
root 1026 1 1031 0 5 21:47 ? 00:00:00 /usr/lib/udisks2/udisksd --  
oscread+ 1035 1 1035 0 2 21:47 ? 00:00:00 /usr/lib/gvfs/gvfs-goa-volu  
oscread+ 1035 1 1036 0 2 21:47 ? 00:00:00 /usr/lib/gvfs/gvfs-goa-volu  
oscread+ 1038 1 1038 0 4 21:47 ? 00:00:00 /usr/lib/gnome-online-accou  
oscread+ 1038 1 1039 0 4 21:47 ? 00:00:00 /usr/lib/gnome-online-accou  
oscread+ 1038 1 1041 0 4 21:47 ? 00:00:00 /usr/lib/gnome-online-accou  
oscread+ 1038 1 1044 0 4 21:47 ? 00:00:00 /usr/lib/gnome-online-accou  
oscread+ 1045 1 1045 0 3 21:47 ? 00:00:00 /usr/lib/telepathy/mission-  
oscread+ 1045 1 1051 0 3 21:47 ? 00:00:00 /usr/lib/telepathy/mission-  
oscread+ 1045 1 1052 0 3 21:47 ? 00:00:00 /usr/lib/telepathy/mission-  
oscread+ 1047 1 1047 0 2 21:47 ? 00:00:00 /usr/lib/gvfs/gvfs-gphoto2-  
oscread+ 1047 1 1050 0 2 21:47 ? 00:00:00 /usr/lib/gvfs/gvfs-gphoto2-  
oscread+ 1054 1 1054 0 2 21:47 ? 00:00:00 /usr/lib/gvfs/gvfs-mtp-volu  
oscread+ 1054 1 1056 0 2 21:47 ? 00:00:00 /usr/lib/gvfs/gvfs-mtp-volu  
oscread+ 1058 1 1058 0 3 21:47 ? 00:00:00 /usr/lib/gvfs/gvfs-afc-volu  
oscread+ 1058 1 1059 0 3 21:47 ? 00:00:00 /usr/lib/gvfs/gvfs-afc-volu  
oscread+ 1058 1 1061 0 3 21:47 ? 00:00:00 /usr/lib/gvfs/gvfs-afc-volu  
oscread+ 1062 925 1062 18 7 21:47 ? 00:05:57 /usr/bin/gnome-shell  
oscread+ 1062 925 1065 0 7 21:47 ? 00:00:00 /usr/bin/gnome-shell  
oscread+ 1062 925 1071 0 7 21:47 ? 00:00:00 /usr/bin/gnome-shell  
oscread+ 1062 925 1073 0 7 21:47 ? 00:00:00 /usr/bin/gnome-shell  
oscread+ 1062 925 1074 0 7 21:47 ? 00:00:00 /usr/bin/gnome-shell  
oscread+ 1062 925 1075 0 7 21:47 ? 00:00:00 /usr/bin/gnome-shell  
oscread+ 1062 925 1076 0 7 21:47 ? 00:00:00 /usr/bin/gnome-shell  
oscread+ 1067 1 1067 0 2 21:47 ? 00:00:00 /usr/lib/gnome-settings-dae  
oscread+ 1067 1 1068 0 2 21:47 ? 00:00:00 /usr/lib/gnome-settings-dae  
oscread+ 1080 1 1080 0 5 21:47 ? 00:00:00 /usr/lib/gnome-shell/gnome-  
oscread+ 1080 1 1082 0 5 21:47 ? 00:00:00 /usr/lib/gnome-shell/gnome-  
oscread+ 1080 1 1089 0 5 21:47 ? 00:00:00 /usr/lib/gnome-shell/gnome-  
oscread+ 1080 1 1090 0 5 21:47 ? 00:00:00 /usr/lib/gnome-shell/gnome-  
oscread+ 1080 1 1151 0 5 21:47 ? 00:00:00 /usr/lib/gnome-shell/gnome-  
oscread+ 1084 1 1084 0 3 21:47 ? 00:00:00 /usr/lib/evolution/evolutio  
oscread+ 1084 1 1085 0 3 21:47 ? 00:00:00 /usr/lib/evolution/evolutio  
oscread+ 1084 1 1086 0 3 21:47 ? 00:00:00 /usr/lib/evolution/evolutio  
oscread+ 1097 925 1097 0 5 21:47 ? 00:00:00 /usr/lib/evolution/3.12/evo  
oscread+ 1097 925 1149 0 5 21:47 ? 00:00:00 /usr/lib/evolution/3.12/evo  
oscread+ 1097 925 1161 0 5 21:47 ? 00:00:00 /usr/lib/evolution/3.12/evo  
oscread+ 1097 925 1162 0 5 21:47 ? 00:00:00 /usr/lib/evolution/3.12/evo  
oscread+ 1097 925 1165 0 5 21:47 ? 00:00:00 /usr/lib/evolution/3.12/evo  
oscread+ 1098 925 1098 0 3 21:47 ? 00:00:00 nm-applet  
oscread+ 1098 925 1128 0 3 21:47 ? 00:00:00 nm-applet  
oscread+ 1098 925 1154 0 3 21:47 ? 00:00:00 nm-applet  
oscread+ 1100 925 1100 0 3 21:47 ? 00:00:00 /usr/lib/tracker/tracker-mi  
oscread+ 1100 925 1222 0 3 21:47 ? 00:00:00 /usr/lib/tracker/tracker-mi  
oscread+ 1100 925 1229 0 3 21:47 ? 00:00:00 /usr/lib/tracker/tracker-mi  
oscread+ 1102 925 1102 0 5 21:47 ? 00:00:07 nautilus -n  
oscread+ 1102 925 1131 0 5 21:47 ? 00:00:00 nautilus -n  
oscread+ 1102 925 1133 0 5 21:47 ? 00:00:00 nautilus -n  
oscread+ 1102 925 1156 0 5 21:47 ? 00:00:00 nautilus -n  
oscread+ 1102 925 4114 0 5 22:18 ? 00:00:00 nautilus -n  
oscread+ 1104 925 1104 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1104 925 1114 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1104 925 1116 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1104 925 1117 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1104 925 1166 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1104 925 1167 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1104 925 1168 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1104 925 1169 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1104 925 1170 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1104 925 1171 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1104 925 1172 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1104 925 1173 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1104 925 1174 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1104 925 1175 0 14 21:47 ? 00:00:00 /usr/lib/tracker/tracker-ex  
oscread+ 1107 1 1107 0 8 21:47 ? 00:00:00 /usr/lib/tracker/tracker-st  
oscread+ 1107 1 1110 0 8 21:47 ? 00:00:00 /usr/lib/tracker/tracker-st  
oscread+ 1107 1 1113 0 8 21:47 ? 00:00:00 /usr/lib/tracker/tracker-st  
oscread+ 1107 1 1115 0 8 21:47 ? 00:00:00 /usr/lib/tracker/tracker-st  
oscread+ 1107 1 1121 0 8 21:47 ? 00:00:00 /usr/lib/tracker/tracker-st  
oscread+ 1107 1 1122 0 8 21:47 ? 00:00:00 /usr/lib/tracker/tracker-st  
oscread+ 1107 1 1123 0 8 21:47 ? 00:00:00 /usr/lib/tracker/tracker-st  
oscread+ 1107 1 1124 0 8 21:47 ? 00:00:00 /usr/lib/tracker/tracker-st  
oscread+ 1108 925 1108 0 4 21:47 ? 00:00:08 zeitgeist-datahub  
oscread+ 1108 925 1119 0 4 21:47 ? 00:00:00 zeitgeist-datahub  
oscread+ 1108 925 1141 0 4 21:47 ? 00:00:00 zeitgeist-datahub  
oscread+ 1108 925 1143 0 4 21:47 ? 00:00:00 zeitgeist-datahub  
oscread+ 1109 925 1109 0 4 21:47 ? 00:00:00 /usr/lib/tracker/tracker-mi  
oscread+ 1109 925 1120 0 4 21:47 ? 00:00:00 /usr/lib/tracker/tracker-mi  
oscread+ 1109 925 1127 0 4 21:47 ? 00:00:00 /usr/lib/tracker/tracker-mi  
oscread+ 1109 925 1129 0 4 21:47 ? 00:00:00 /usr/lib/tracker/tracker-mi  
oscread+ 1111 925 1111 0 3 21:47 ? 00:00:00 /usr/lib/tracker/tracker-mi  
oscread+ 1111 925 1225 0 3 21:47 ? 00:00:00 /usr/lib/tracker/tracker-mi  
oscread+ 1111 925 1231 0 3 21:47 ? 00:00:00 /usr/lib/tracker/tracker-mi  
oscread+ 1112 925 1112 0 2 21:47 ? 00:00:00 /usr/bin/python /usr/share/  
oscread+ 1112 925 1160 0 2 21:47 ? 00:00:00 /usr/bin/python /usr/share/  
oscread+ 1126 1 1126 0 2 21:47 ? 00:00:00 /usr/bin/zeitgeist-daemon  
oscread+ 1126 1 1130 0 2 21:47 ? 00:00:00 /usr/bin/zeitgeist-daemon  
oscread+ 1139 1 1139 0 3 21:47 ? 00:00:00 /usr/lib/i386-linux-gnu/zei  
oscread+ 1139 1 1144 0 3 21:47 ? 00:00:00 /usr/lib/i386-linux-gnu/zei  
oscread+ 1139 1 1159 0 3 21:47 ? 00:00:00 /usr/lib/i386-linux-gnu/zei  
oscread+ 1147 1139 1147 0 1 21:47 ? 00:00:00 /bin/cat  
oscread+ 1153 1 1153 0 6 21:47 ? 00:00:00 /usr/lib/evolution/evolutio  
oscread+ 1153 1 1200 0 6 21:47 ? 00:00:00 /usr/lib/evolution/evolutio  
oscread+ 1153 1 1204 0 6 21:47 ? 00:00:00 /usr/lib/evolution/evolutio  
oscread+ 1153 1 1205 0 6 21:47 ? 00:00:00 /usr/lib/evolution/evolutio  
oscread+ 1153 1 1206 0 6 21:47 ? 00:00:00 /usr/lib/evolution/evolutio  
oscread+ 1153 1 1213 0 6 21:47 ? 00:00:00 /usr/lib/evolution/evolutio  
oscread+ 1158 1 1158 0 1 21:47 ? 00:00:00 /usr/lib/i386-linux-gnu/gco  
oscread+ 1177 1 1177 0 6 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-trash -  
oscread+ 1177 1 1178 0 6 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-trash -  
oscread+ 1177 1 1179 0 6 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-trash -  
oscread+ 1177 1 1181 0 6 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-trash -  
oscread+ 1177 1 1193 0 6 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-trash -  
oscread+ 1177 1 1194 0 6 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-trash -  
oscread+ 1189 1 1189 0 3 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-burn --  
oscread+ 1189 1 1190 0 3 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-burn --  
oscread+ 1189 1 1192 0 3 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-burn --  
oscread+ 1211 1 1211 0 2 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-metadat  
oscread+ 1211 1 1214 0 2 21:47 ? 00:00:00 /usr/lib/gvfs/gvfsd-metadat  
oscread+ 1257 1 1257 0 3 21:48 ? 00:00:00 /usr/lib/dconf/dconf-servic  
oscread+ 1257 1 1258 0 3 21:48 ? 00:00:00 /usr/lib/dconf/dconf-servic  
oscread+ 1257 1 1259 0 3 21:48 ? 00:00:00 /usr/lib/dconf/dconf-servic  
oscread+ 1280 1 1280 0 4 21:48 ? 00:00:10 /usr/lib/gnome-terminal/gno  
oscread+ 1280 1 1281 0 4 21:48 ? 00:00:00 /usr/lib/gnome-terminal/gno  
oscread+ 1280 1 1282 0 4 21:48 ? 00:00:00 /usr/lib/gnome-terminal/gno  
oscread+ 1280 1 1285 0 4 21:48 ? 00:00:00 /usr/lib/gnome-terminal/gno  
oscread+ 1283 1280 1283 0 1 21:48 ? 00:00:00 gnome-pty-helper  
oscread+ 1284 1280 1284 0 1 21:48 pts/0 00:00:00 bash  
root 1309 408 1309 0 1 21:49 ? 00:00:00 /sbin/dhclient -d -q -sf /u  
oscread+ 3170 1 3170 3 5 22:11 ? 00:00:14 /usr/bin/gedit --gapplicati  
oscread+ 3170 1 3171 0 5 22:11 ? 00:00:00 /usr/bin/gedit --gapplicati  
oscread+ 3170 1 3172 0 5 22:11 ? 00:00:00 /usr/bin/gedit --gapplicati  
oscread+ 3170 1 3173 0 5 22:11 ? 00:00:00 /usr/bin/gedit --gapplicati  
oscread+ 3170 1 4119 0 5 22:18 ? 00:00:00 /usr/bin/gedit --gapplicati  
root 4066 2 4066 0 1 22:13 ? 00:00:00 [kworker/0:1]  
root 4116 2 4116 0 1 22:18 ? 00:00:00 [kworker/0:0]  
oscread+ 4120 1284 4120 0 1 22:18 pts/0 00:00:00 ps -eLf

# 7. Reference

Operating System Concepts Book by Abraham Silberschatz

<https://www.kernel.org/doc/Documentation/kbuild/modules.txt>

<https://www.tldp.org/LDP/lkmpg/2.6/lkmpg.pdf>

<https://linuxgazette.net/133/saha.html>

<https://stackoverflow.com/questions/5550404/list-entry-in-linux>

<https://linux.die.net/>

# 8. Appendix

## 8.1 Makefile

obj-m += listing\_linear.o #filename for listing linear code

obj-m += listing\_dfs.o #filename for listing dfs code

all:

make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules

clean:

make -C /lib/modules/$(shell uname -r)/build M=$(PWD) clean

## 8.2 Code

1. First module for listing process linearly by process id:

#include <linux/module.h>

#include <linux/sched.h>

int listing\_linear\_init(void)

{

printk(KERN\_INFO "listing\_linear\_module launched\n");

struct task\_struct \*process;

printk(KERN\_INFO "P-ID Name State");

for\_each\_process(process) {

printk(KERN\_INFO "%-6d%-17s%-1ld", process->pid, process->comm, process->state);

}

return 0;

}

void listing\_linear\_exit(void)

{

printk(KERN\_INFO "listing\_linear\_module un-loaded\n");

}

module\_init(listing\_linear\_init);

module\_exit(listing\_linear\_exit);

MODULE\_LICENSE("GPL");

MODULE\_DESCRIPTION("Listing Linux Kernel Module Linearly by PId");

MODULE\_AUTHOR("Lu/Gaochao");

1. Second module for listing process in the order of DFS:

#include <linux/module.h>

#include <linux/sched.h>

void helper(struct task\_struct \*process){

struct task\_struct \*next\_process;

struct list\_head \*list;

printk(KERN\_INFO "%-6d%-17s%-1ld", process->pid, process->comm, process->state);

list\_for\_each(list, &process->children) {

next\_process = list\_entry(list, struct task\_struct, sibling);

helper(next\_process);

}

}

int listing\_dfs\_init(void)

{

printk(KERN\_INFO "listing\_DFS\_module loaded\n");

printk(KERN\_INFO "P-ID Name State");

helper(&init\_task);

return 0;

}

void listing\_dfs\_exit(void)

{

printk(KERN\_INFO "listing\_DFS\_module un-loaded\n");

}

module\_init(listing\_dfs\_init);

module\_exit(listing\_dfs\_exit);

MODULE\_LICENSE("GPL");

MODULE\_DESCRIPTION("Listing Linux Kernel Module DFS");

MODULE\_AUTHOR("Gaochao/Lu");