The Description of Project

Name: Xi Jingwei Student ID:517030910116 Email:jingweixi@sjtu.edu.cn

1 Problem1

1.1 Files

1.1.1 ptree.c

Function static int ptree(struct prinfo *buf, int *nr)

Call function dfs and use read_lock and read_unlock to protect the data.Then copy data from task[] to buf.

Function void dfs(struct task_struct start, int deep)

Search all processes and store information in dfs order.

1.1.2 Makefile

Use Makefile to compile the ptree.c file.

1.2 Process

a. I searched a lot of information of task_struct and list function in Linux kernel on the internet. Then I know how to use the function list_entry and list_for_each.

b. When I coded the dfs function, I met many problems. I don't know how to depth-first-search a tree without knowing all the children of parent. Then I search on Google and I find node -> children will point to the head of children list. And I use list_for_each function to depth-first-search the process tree.

c. At problem1, I don't know the difference between struct task_struct and list_head. And I didn't know how to transfer them. With the help of friends, I learn to use list_entry((&node , struct task_struct, sibling) to transfer them.

2 Problem2

2.1 Files

2.1.1 ptree.c

The program will call ptree function in kernel and print the entire process tree (in DFS order) using tabs to indent children with respect to their parents.

2.1.2 Android.mk

The make file for project.

2.2 Process

At first, I used syscall 391 in the program, but I could not make it work in kernel and I can not find the problem. With the help of TAs, I changed syscall from 391 to 356 and then it worked.

2.3 Result

```
The number of task is 59!
Print start
swapper, 0, 0, 0, 1, 0, 0
  init, 1, 1, 0, 45, 2, 0
    ueventd, 45, 1, 1, 0, 61, 0
    logd, 61, 1, 1, 0, 62, 1036
    vold, 62, 1, 1, 0, 68, 0
    healthd, 68, 1, 1, 0, 69, 0
    lmkd, 69, 1, 1, 0, 70, 0
    servicemanager, 70, 1, 1, 0, 71, 1000
    surfaceflinger, 71, 1, 1, 0, 73, 1000
    qemud, 73, 1, 1, 0, 76, 0
    sh, 76, 1, 1, 0, 77, 2000
    adbd, 77, 1, 1, 197, 78, 0
      sh, 197, 1, 77, 370, 1, 0
        ptreeARM, 370, 0, 197, 0, 1, 0
    netd, 78, 1, 1, 371, 79, 0
      sh, 371, 0, 78, 0, 1, 0
    debuggerd, 79, 1, 1, 0, 80, 0
    rild, 80, 1, 1, 0, 81, 1001
    drmserver, 81, 1, 1, 0, 82, 1019
    mediaserver, 82, 1, 1, 0, 83, 1013
    installd, 83, 1, 1, 0, 84, 0
    keystore, 84, 1, 1, 0, 85, 1017
    main, 85, 1, 1, 235, 86, 0
      system_server, 235, 1, 85, 0, 1, 1000
    gatekeeperd, 86, 1, 1, 0, 89, 1000
    perfprofd, 89, 1, 1, 0, 90, 0
    fingerprintd, 90, 1, 1, 0, 122, 1000
    bootanimation, 122, 1, 1, 0, 1, 1003
  kthreadd, 2, 1, 0, 3, 0, 0
    ksoftirqd/0, 3, 1, 2, 0, 4, 0
    kworker/0:0, 4, 1, 2, 0, 5, 0
    kworker/u:0, 5, 1, 2, 0, 6, 0
```

```
    khelper, 6, 1, 2, 0, 7, 0
    sync_supers, 7, 1, 2, 0, 8, 0
    bdi-default, 8, 1, 2, 0, 9, 0
    kblockd, 9, 1, 2, 0, 10, 0
    rpciod, 10, 1, 2, 0, 11, 0
    kworker/0:1, 11, 1, 2, 0, 12, 0
    kswapd0, 12, 1, 2, 0, 13, 0
    fsnotify_mark, 13, 1, 2, 0, 14, 0
    crypto, 14, 1, 2, 0, 25, 0
    kworker/u:1, 25, 1, 2, 0, 30, 0
    mtdblock0, 30, 1, 2, 0, 35, 0
    mtdblock1, 35, 1, 2, 0, 40, 0
    mtdblock2, 40, 1, 2, 0, 41, 0
    binder, 41, 1, 2, 0, 42, 0
    deferwq, 42, 1, 2, 0, 43, 0
    kworker/u:2, 43, 1, 2, 0, 44, 0
    mmcqd/0, 44, 1, 2, 0, 47, 0
    jbd2/mtdblock0-, 47, 1, 2, 0, 48, 0
    ext4-dio-unwrit, 48, 1, 2, 0, 51, 0
    flush-31:1, 51, 1, 2, 0, 53, 0
    jbd2/mtdblock1-, 53, 1, 2, 0, 54, 0
    ext4-dio-unwrit, 54, 1, 2, 0, 57, 0
    flush-31:2, 57, 1, 2, 0, 59, 0
    jbd2/mtdblock2-, 59, 1, 2, 0, 60, 0
    ext4-dio-unwrit, 60, 1, 2, 0, 63, 0
    kworker/0:2, 63, 1, 2, 0, 94, 0
    kauditd, 94, 1, 2, 0, 0, 0
Print end
```

3 Problem3

3.1 Files

3.1.1 process.c

The program enerate a new process and output “StudentIDParent”with PID, then generates its children process output “StudentIDChild”with PID.

And in child process it will execute ptree.

3.1.2 Android.mk

The make file for project.

3.2 Process

I searched on the internet for how to get the process id, then I get know that getpid() function can get the id of process. And after searching the relationship of parent process and child process, I know the pid = fork() in parent process will return the child process id.

3.3 Result

4 Problem3

4.1 Files

4.1.1 process.c

The program enerate a new process and output "StudentIDParent"with PID, then generates its children process output "StudentIDChild"with PID.

And in child process it will execute ptree.

4.1.2 Android.mk

The make file for project.

4.2 Process

I searched on the internet for how to get the process id, then I get know that getpid() function can get the id of process. And after searching the relationship of parent process and child process, I know the pid = fork() in parent process will return the child process id.

4.3 Result

```
517030910116 Parent pid = 701
517030910116 Child pid = 702
The number of task is 59!
Print start
swapper, 0, 0, 0, 1, 0, 0
  init, 1, 1, 0, 45, 2, 0
    ueventd, 45, 1, 1, 0, 61, 0
    logd, 61, 1, 1, 0, 62, 1036
    vold, 62, 1, 1, 0, 68, 0
    healthd, 68, 1, 1, 0, 69, 0
    lmkd, 69, 1, 1, 0, 70, 0
    servicemanager, 70, 1, 1, 0, 71, 1000
    surfaceflinger, 71, 1, 1, 0, 73, 1000
    qemud, 73, 1, 1, 0, 76, 0
    sh, 76, 1, 1, 0, 77, 2000
    adbd, 77, 1, 1, 197, 78, 0
```

```
    sh, 197, 1, 77, 370, 1, 0
      ptreeARM, 370, 0, 197, 0, 1, 0
  netd, 78, 1, 1, 371, 79, 0
    sh, 371, 0, 78, 0, 1, 0
  debuggerd, 79, 1, 1, 0, 80, 0
  rild, 80, 1, 1, 0, 81, 1001
  drmserver, 81, 1, 1, 0, 82, 1019
  mediaserver, 82, 1, 1, 0, 83, 1013
  installd, 83, 1, 1, 0, 84, 0
  keystore, 84, 1, 1, 0, 85, 1017
  main, 85, 1, 1, 235, 86, 0
    system_server, 235, 1, 85, 0, 1, 1000
  gatekeeperd, 86, 1, 1, 0, 89, 1000
  perfprofd, 89, 1, 1, 0, 90, 0
  fingerprintd, 90, 1, 1, 0, 122, 1000
  bootanimation, 122, 1, 1, 0, 1, 1003
kthreadd, 2, 1, 0, 3, 0, 0
  ksoftirqd/0, 3, 1, 2, 0, 4, 0
  kworker/0:0, 4, 1, 2, 0, 5, 0
  kworker/u:0, 5, 1, 2, 0, 6, 0
  khelper, 6, 1, 2, 0, 7, 0
  sync_supers, 7, 1, 2, 0, 8, 0
  bdi-default, 8, 1, 2, 0, 9, 0
  kblockd, 9, 1, 2, 0, 10, 0
  rpciod, 10, 1, 2, 0, 11, 0
  kworker/0:1, 11, 1, 2, 0, 12, 0
  kswapd0, 12, 1, 2, 0, 13, 0
  fsnotify_mark, 13, 1, 2, 0, 14, 0
  crypto, 14, 1, 2, 0, 25, 0
  kworker/u:1, 25, 1, 2, 0, 30, 0
  mtdblock0, 30, 1, 2, 0, 35, 0
  mtdblock1, 35, 1, 2, 0, 40, 0
  mtdblock2, 40, 1, 2, 0, 41, 0
  binder, 41, 1, 2, 0, 42, 0
  deferwq, 42, 1, 2, 0, 43, 0
  kworker/u:2, 43, 1, 2, 0, 44, 0
  mmcqd/0, 44, 1, 2, 0, 47, 0
  jbd2/mtdblock0-, 47, 1, 2, 0, 48, 0
  ext4-dio-unwrit, 48, 1, 2, 0, 51, 0
  flush-31:1, 51, 1, 2, 0, 53, 0
  jbd2/mtdblock1-, 53, 1, 2, 0, 54, 0
  ext4-dio-unwrit, 54, 1, 2, 0, 57, 0
  flush-31:2, 57, 1, 2, 0, 59, 0
  jbd2/mtdblock2-, 59, 1, 2, 0, 60, 0
  ext4-dio-unwrit, 60, 1, 2, 0, 63, 0
  kworker/0:2, 63, 1, 2, 0, 94, 0
```

```
    kauditd, 94, 1, 2, 0, 0, 0
Print end
```

5 Problem4

5.1 Files

5.1.1 server.c

Function int main()

The server will listen to the port. If there is a client want to connect, it will connect to the client and create a new thread to call serve function.

Function void *serve(void *clientfd)

It will judge whether server could serve client or not. If the count of client that server served concurrently is less than two, the server can receive the client message and change it. Then it will send the message changed to the client.

5.1.2 client.c

The program client.c will send message to server and receive the message changed by server.

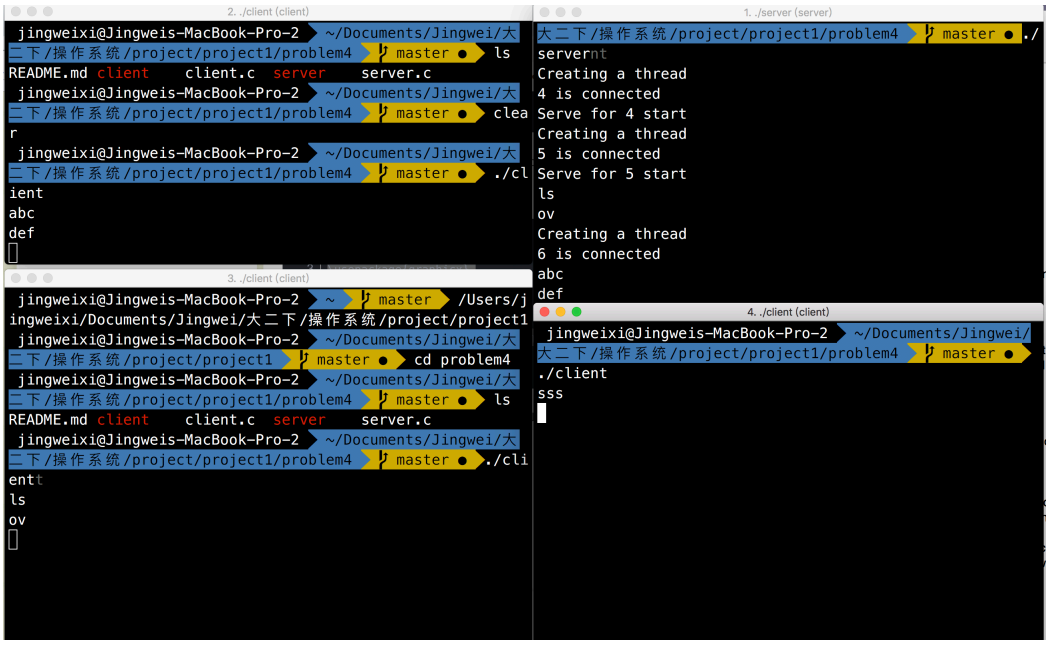
5.2 Process

1. I met a lot of problems in server problem. At first, I don't know the how to create thread and how to deal with critical section problem. I find answers on the internet and use pthread and mutex.

2. I use two mutex. One mutex is for variable count which is the number of client which are served by server. The other mutex is to make new coming clients to wait if there are two clients served concurrently.

5.3 Result

When client1 and client2 are served by server, client3 need to wait.



When client1 quit and end the service, the client3 will be served by server.