**Report**

120090581 丁奕杰

First, the basic idea of the first program is that the program runs the process in user mode. The main process executes the test program and waits for its return signal. Upon receiving a termination signal, the parent prints out information about the signal. Based on the above information, the program will use the fork() function to fork the child process at the beginning. In this way, the child process is distinguished from the parent process and the corresponding signal information is printed.

Then, the second program will create a kernel thread. In the thread, the program will fork a child process and make it to execute another program. The parent process will wait for the child’s terminated signal and print out related information.

Listed instructions are used to compile kernel:

◦ cp KERNEL\_FILE.tar.xz /home/seed/work

◦ cd /home/seed/work

◦ $sudo tar xvf KERNEL\_FILE.tar.xz

◦ $sudo su

◦ $cd /home/seed/work /KERNEL\_FIL

◦ $make mrproper

◦ $make clean

◦ $make menuconfi

◦ $make bzImage -j$(nproc)

◦ $make modules -j$(nproc)

◦ $make modules\_install

◦ $make instal

The version of kernel is 5.10.5