

Report

1. The design of the program

1.1

For program1, The structure is that:

Firstly use `fork()` function to fork a child process, split to child process part and parent process part.

In child process part, use the given template to execute the file. And print out message related to child process. In parent process part, after `waitpid()`, status contains all return information of child process. If so, print out "Normal termination".

Using `WIFSIGNALED` to check if child process is terminated by signal. And get the signal value by using `WTERMSIG`. With a signal table found online, complete the corresponding output. And using `WIFSTOPPED` to check if child process is stopped.

1.2

For program2, The structure is that:

Firstly, `program2_init` is invoked by `module_init()`. Then the initialization function call `kthread_create()` and pass `my_fork` as a parameter. After `wake_up_process()`, it goes to `my_fork()`. In `my_fork()`, fork a process to execute the test program, using `kernel_clone()`. Then according to `mywait()` function which calls the `dowait()` function, the parent process will wait until child process terminates.

Also the `kernel_clone()` function will return a pid for child process and we can also know its parent pid. And print them out.

The `do_execve()` function will invoke the test case, which will use `getname_kernel()` to return the path of the test case.

After that within this test program, it will raise signal. The signal could be caught and related message should be printed out in kernel log.

2. To set up the development environment, including how to compile

kernel

- (1) Download and install VMBox and set up the connection within SSH
- (2) Compile the kernel according to tut's ppt
- (3) Find the `kernel_clone()`, `do_execve()`, `getname_kernel()` and `do_wait()` functions and use `EXPORT_SYMBOL()` to export them
- (4) The screenshot of (3)

```

2502     put_pid(pid);
2503     return nr;
2504 }
2505 EXPORT_SYMBOL(kernel_clone);

```

- (5) Compile the kernel 1. make bzImage, 2. make modules 3. Install kernel modules 4. make install
- (6) Reboot

3. Screenshot of your program output

Program1

- (1) In the 'program1' directory, type 'make' command and enter.
- (2) In the 'program1' directory, type './program1 \$TEST_CASE \$ARG1 \$ARG2 ...'

1. Normal

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./normal
I'm the Parent Process, my pid = 4767
I'm the Child Process, my pid = 4768
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the normal program

-----CHILD PROCESS END-----
Parent process receives SIGCHLD signal
Normal termination with EXIT STATUS = 0

```

2. Stop

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./stop
I'm the Parent Process, my pid = 4604
I'm the Child Process, my pid = 4605
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGSTOP program

Parent process receives SIGCHLD signal
child process get SIGSTOP signal

```

3. Abort

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./abort
I'm the Parent Process, my pid = 4794
I'm the Child Process, my pid = 4795
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGABRT program

Parent process receives SIGCHLD signal
child process get SIGABRT signal

```

4. Alarm

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./alarm
I'm the Parent Process, my pid = 4857
I'm the Child Process, my pid = 4858
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGALRM program

Parent process receives SIGCHLD signal
child process get SIGALRM signal

```

5. Bus

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./bus
I'm the Parent Process, my pid = 4908
I'm the Child Process, my pid = 4909
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGBUS program

Parent process receives SIGCHLD signal
child process get SIGBUS signal

```

6. Floating

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./floating
I'm the Parent Process, my pid = 4950
I'm the Child Process, my pid = 4951
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGFPE program

Parent process receives SIGCHLD signal
child process get SIGFPE signal

```

7. Hangup

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./hangup
I'm the Parent Process, my pid = 4990
I'm the Child Process, my pid = 4991
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGHUP program

Parent process receives SIGCHLD signal
child process get SIGHUP signal

```

8. Illegal_instr

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./illegal_instr
I'm the Parent Process, my pid = 5061
I'm the Child Process, my pid = 5062
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGILL program

Parent process receives SIGCHLD signal
child process get SIGILL signal

```

9. Interrupt

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./interrupt
I'm the Parent Process, my pid = 5091
I'm the Child Process, my pid = 5092
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGINT program

Parent process receives SIGCHLD signal
child process get SIGINT signal

```

10. Kill

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./kill
I'm the Parent Process, my pid = 5105
I'm the Child Process, my pid = 5106
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGKILL program

Parent process receives SIGCHLD signal
child process get SIGKILL signal

```

11. Pipe

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./pipe
I'm the Parent Process, my pid = 5182
I'm the Child Process, my pid = 5183
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGPIPE program

Parent process receives SIGCHLD signal
child process get SIGPIPE signal

```

12. Quit

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./quit
I'm the Parent Process, my pid = 5208
I'm the Child Process, my pid = 5209
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGQUIT program

Parent process receives SIGCHLD signal
child process get SIGQUIT signal

```

13. Segment_fault

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./segment_fault
I'm the Parent Process, my pid = 5283
I'm the Child Process, my pid = 5284
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGSEGV program

Parent process receives SIGCHLD signal
child process get SIGSEGV signal

```

14. Terminate


```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./terminate
I'm the Parent Process, my pid = 5311
I'm the Child Process, my pid = 5312
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGTERM program

Parent process receives SIGCHLD signal
child process get SIGTERM signal

```

15. Trap

```

● vagrant@csc3150:~/csc3150/source/program1$ ./program1 ./trap
I'm the Parent Process, my pid = 5349
I'm the Child Process, my pid = 5350
Child process start to execute test program:
-----CHILD PROCESS START-----
This is the SIGTRAP program

Parent process receives SIGCHLD signal
child process get SIGTRAP signal

```

Program2

- (1) Make
- (2) Sudo insmod program2.ko
- (3) Sudo rmmod program2
- (4) Dmesg

1. Term

```

● vagrant@csc3150:~/csc3150/source/program2$ dmesg
[ 5364.787596] [program2] : module_init {JiyuanLong} {120090850}
[ 5364.787598] [program2] : module_init create kthread start
[ 5364.787653] [program2] : module_init kthread start
[ 5364.788466] [program2] : The child process has pid = 6425
[ 5364.788468] [program2] : This is the parent process,pid = 6423
[ 5364.788469] [program2] : child process
[ 5364.788683] [program2] : get SIGTERM signal
[ 5364.788684] [program2] : child process terminated
[ 5364.788684] [program2] : The return signal is 15
[ 5370.414446] [program2] : Module_exit

```

2. Abort

```

[ 5476.906507] [program2] : module_init {JiyuanLong} {120090850}
[ 5476.906509] [program2] : module_init create kthread start
[ 5476.906561] [program2] : module_init kthread start
[ 5476.906976] [program2] : The child process has pid = 6537
[ 5476.906977] [program2] : This is the parent process,pid = 6536
[ 5476.906978] [program2] : child process
[ 5476.986466] [program2] : get SIGABRT signal
[ 5476.986467] [program2] : child process abort
[ 5476.986468] [program2] : The return signal is 6
[ 5482.735026] [program2] : Module_exit
vagrant@csc3150:~/csc3150/source/program2$ dmesg

```

3. Alarm

```

vagrant@csc3150:~/csc3150/source/program2$ sudo dmesg -c
[ 5673.005598] [program2] : module_init {JiyuanLong} {120090850}
[ 5673.005599] [program2] : module_init create kthread start
[ 5673.005662] [program2] : module_init kthread start
[ 5673.005696] [program2] : The child process has pid = 6795
[ 5673.005697] [program2] : This is the parent process,pid = 6794
[ 5673.005699] [program2] : child process
[ 5673.005966] [program2] : get SIGALRM signal
[ 5673.005966] [program2] : child process alarmed]
[ 5673.005967] [program2] : The return signal is 14
[ 5678.093917] [program2] : Module_exit

```

4. Bus

```

vagrant@csc3150:~/csc3150/source/program2$ sudo dmesg
[ 5779.813099] [program2] : module_init {JiyuanLong} {120090850}
[ 5779.813100] [program2] : module_init create kthread start
[ 5779.813155] [program2] : module_init kthread start
[ 5779.813653] [program2] : The child process has pid = 6946
[ 5779.813655] [program2] : This is the parent process,pid = 6944
[ 5779.813656] [program2] : child process
[ 5779.900356] [program2] : get SIGBUS signal
[ 5779.900358] [program2] : child process has bus error
[ 5779.900359] [program2] : The return signal is 7
[ 5785.016340] [program2] : Module_exit

```

5. Float

```

vagrant@csc3150:~/csc3150/source/program2$ sudo dmesg -c
\ [ 5870.827927] [program2] : module_init {JiyuanLong} {120090850}
[ 5870.827929] [program2] : module_init create kthread start
[ 5870.828017] [program2] : module_init kthread start
[ 5870.828576] [program2] : The child process has pid = 7051
[ 5870.828577] [program2] : This is the parent process,pid = 7049
[ 5870.828579] [program2] : child process
[ 5870.909635] [program2] : get SIGFPE signal
[ 5870.909636] [program2] : child process has arithmetic excption
[ 5870.909637] [program2] : The return signal is 8
[ 5877.229045] [program2] : Module_exit

```

6. Hangup

```

vagrant@csc3150:~/csc3150/source/program2$ sudo dmesg -c
[ 6043.177360] [program2] : module_init {JiyuanLong} {120090850}
[ 6043.177361] [program2] : module_init create kthread start
[ 6043.177407] [program2] : module_init kthread start
[ 6043.178183] [program2] : The child process has pid = 7202
[ 6043.178185] [program2] : This is the parent process,pid = 7200
[ 6043.178187] [program2] : child process
[ 6043.178728] [program2] : get SIGUP signal
[ 6043.178729] [program2] : child process hung up
[ 6043.178730] [program2] : The return signal is 1
[ 6049.445768] [program2] : Module_exit

```

7. Illegal

```

vagrant@csc3150:~/csc3150/source/program2$ sudo dmesg -c
[ 6162.190084] [program2] : module_init {JiyuanLong} {120090850}
[ 6162.190086] [program2] : module_init create kthread start
[ 6162.190131] [program2] : module_init kthread start
[ 6162.191035] [program2] : The child process has pid = 7292
[ 6162.191036] [program2] : This is the parent process,pid = 7290
[ 6162.191038] [program2] : child process
[ 6162.271993] [program2] : get SIGILL signal
[ 6162.271996] [program2] : child process has illegal instruction
[ 6162.271996] [program2] : The return signal is 4
[ 6166.242220] [program2] : Module_exit

```

8. Interrupt


```

vagrant@csc3150:~/csc3150/source/program2$ sudo dmesg -c
[ 6260.710962] [program2] : module_init {JiyuanLong} {120090850}
[ 6260.710964] [program2] : module_init create kthread start
[ 6260.711023] [program2] : module_init kthread start
[ 6260.712532] [program2] : The child process has pid = 7387
[ 6260.712533] [program2] : This is the parent process,pid = 7384
[ 6260.712535] [program2] : child process
[ 6260.712852] [program2] : get SIGINT signal
[ 6260.712853] [program2] : child process interrupted
[ 6260.712854] [program2] : The return signal is 2
[ 6264.991999] [program2] : Module_exit

```

9. Kill

```

vagrant@csc3150:~/csc3150/source/program2$ sudo dmesg -c
[ 6325.209548] [program2] : module_init {JiyuanLong} {120090850}
[ 6325.209550] [program2] : module_init create kthread start
[ 6325.209595] [program2] : module_init kthread start
[ 6325.210295] [program2] : The child process has pid = 7477
[ 6325.210297] [program2] : This is the parent process,pid = 7475
[ 6325.210300] [program2] : child process
[ 6325.210660] [program2] : get SIGKILL signal
[ 6325.210661] [program2] : child process killed
[ 6325.210661] [program2] : The return signal is 9
[ 6329.961116] [program2] : Module_exit

```

10. Pipe

```

vagrant@csc3150:~/csc3150/source/program2$ sudo dmesg -c
[ 6395.152883] [program2] : module_init {JiyuanLong} {120090850}
[ 6395.152884] [program2] : module_init create kthread start
[ 6395.152938] [program2] : module_init kthread start
[ 6395.153176] [program2] : The child process has pid = 7604
[ 6395.153177] [program2] : This is the parent process,pid = 7602
[ 6395.153179] [program2] : child process
[ 6395.153409] [program2] : get SIGPIPE signal
[ 6395.153410] [program2] : child process has broken pipe
[ 6395.153410] [program2] : The return signal is 13
[ 6399.225675] [program2] : Module_exit

```

11. Quit

```

vagrant@csc3150:~/csc3150/source/program2$ sudo dmesg -c
[ 6455.954482] [program2] : module_init {JiyuanLong} {120090850}
[ 6455.954484] [program2] : module_init create kthread start
[ 6455.954528] [program2] : module_init kthread start
[ 6455.954559] [program2] : The child process has pid = 7669
[ 6455.954561] [program2] : This is the parent process,pid = 7668
[ 6455.954562] [program2] : child process
[ 6456.038551] [program2] : get SIGQUIT signal
[ 6456.038553] [program2] : child process quits
[ 6456.038554] [program2] : The return signal is 3
[ 6460.754139] [program2] : Module_exit

```

12. Segment fault

```

vagrant@csc3150:~/csc3150/source/program2$ sudo dmesg
[ 6528.105238] [program2] : module_init {JiyuanLong} {120090850}
[ 6528.105240] [program2] : module_init create kthread start
[ 6528.105295] [program2] : module_init kthread start
[ 6528.105692] [program2] : The child process has pid = 7760
[ 6528.105693] [program2] : This is the parent process,pid = 7758
[ 6528.105694] [program2] : child process
[ 6528.184643] [program2] : get SIGSEGV signal
[ 6528.184645] [program2] : child process has segment fault
[ 6528.184645] [program2] : The return signal is 11
[ 6533.041578] [program2] : Module_exit

```

13. Stop

```

vagrant@csc3150:~/csc3150/source/program2$ sudo dmesg -c
[ 6695.046019] [program2] : module_init {JiyuanLong} {120090850}
[ 6695.046022] [program2] : module_init create kthread start
[ 6695.046061] [program2] : module_init kthread start
[ 6695.046621] [program2] : The child process has pid = 8004
[ 6695.046622] [program2] : This is the parent process,pid = 8002
[ 6695.046624] [program2] : child process
[ 6695.046846] [program2] : get SIGSTOP signal
[ 6695.046847] [program2] : child process stopped
[ 6695.046847] [program2] : The return signal is 19
[ 6701.207494] [program2] : Module_exit

```

14. Trap

```

vagrant@csc3150:~/csc3150/source/program2$ sudo dmesg -c
[ 6791.536224] [program2] : module_init {JiyuanLong} {120090850}
[ 6791.536225] [program2] : module_init create kthread start
[ 6791.536275] [program2] : module_init kthread start
[ 6791.537383] [program2] : The child process has pid = 8119
[ 6791.537384] [program2] : This is the parent process,pid = 8117
[ 6791.537385] [program2] : child process
[ 6791.618515] [program2] : get SIGTRAP signal
[ 6791.618518] [program2] : child process trapped
[ 6791.618519] [program2] : The return signal is 5
[ 6796.083673] [program2] : Module_exit

```

15. Normal

```

[ 5278.820616] [program2] : module_init {JiyuanLong} {120090850}
[ 5278.820617] [program2] : module_init create kthread start
[ 5278.820690] [program2] : module_init kthread start
[ 5278.821294] [program2] : The child process has pid = 6284
[ 5278.821296] [program2] : This is the parent process,pid = 6283
[ 5278.821298] [program2] : child process
[ 5278.821302] [program2] : process 'MyThread' launched '/home/vagrant/csc3150/source/program2/test'
[ 5283.822427] [program2] : normal exit
[ 5283.822430] [program2] : The return signal is 0
[ 5289.996427] [program2] : Module_exit

```

4. Things I have learned

I have learned some kernel functions. And how a process was create and how it can be terminated, in other words, the whole life of a process. What's more I have learned to check the kernel functions.

Using clang-format:

```

vagrant@csc3150:~/csc3150/source/program1$ clang-format-8 -i /home/vagrant/csc3150/source/program1/program1.c
vagrant@csc3150:~/csc3150/source/program1$ clang-format-8 -i /home/vagrant/csc3150/source/program1/program1.c

```