庄蘅萌
120090648
HW1_report

Program Design

Program1:

1. First, use a fork() function to fork the process and record the return value into pid_t pid.
2. Use pid to check the completion of function and to identify the parent progress and child process.
3. In child process, execute test program with execve().
4. In parent process, wait and record the child process' termination status. With the help of WIFEXITED(),WIFEXITED(),WIFSIGNALED() to output signal.

Program2:

1. Export the needed fuction in linux file. And extern in my file.
2. Recompile the modified kernel.
3. In the init of module, call my_fork().
4. In my fork, use kernel_clone() to create and execute the child process my_exec in the argument .stack of the kernel_clone_args.
5. For the child process, call the do_execve to execve the target file in user mode.
6. For parent process, wait for the child process to finish with the func my_wait(), which call the do_wait() inside.
7. Exit the module.
8. Make the ko file, insert module, remove module, see messge in kernel.

Environment setting and kernel compilation

Environment:

1. Download and install VMbox and vagrant.
2. Install the vscode remote connect extension.
3. Set the config in vscode with vagrant ssh.
4. Vagrant up/halt to use the vagrant.
5. Download source code on internet.
6. Get into root account.
7. Unzip the souce file in target path.
8. Install needed dependency and tools.
9. Clean previous setting.
10. Set the configuration from boot.
11. Build kernel Image and modules
12. Install kernel modules
13. Install kernel

14. reboot

Recompile:

1. Set the configuration.
2. Build kernel Image and modules
3. Install kernel modules
4. Install kernel
5. Reboot

Sample output

Program1:

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./abort
Process start to fork
I'm the Parent Process, my pid = 3099
I'm the Child Process, my pid = 3100
Child process start to execute test program:
------------CHILD PROCESS START------------
This is the SIGABRT program

Parent process receives SIGCHLD signal
Child process terminated with status 6
```

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./alarm
Process start to fork
I'm the Parent Process, my pid = 3201
I'm the Child Process, my pid = 3202
Child process start to execute test program:
------------CHILD PROCESS START------------
This is the SIGALRM program

Parent process receives SIGCHLD signal
Child process terminated with status 14
```

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./bus
Process start to fork
I'm the Parent Process, my pid = 3246
I'm the Child Process, my pid = 3247
Child process start to execute test program:
------------CHILD PROCESS START------------
This is the SIGBUS program

Parent process receives SIGCHLD signal
Child process terminated with status 7
vagrant@csc3150:~/csc3150/ASS1$
```

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./floating
Process start to fork
I'm the Parent Process, my pid = 3312
I'm the Child Process, my pid = 3313
Child process start to execute test program:
------------CHILD PROCESS START------------
This is the SIGFPE program

Parent process receives SIGCHLD signal
Child process terminated with status 8
```

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./hangup
Process start to fork
I'm the Parent Process, my pid = 3398
I'm the Child Process, my pid = 3399
Child process start to execute test program:
------------CHILD PROCESS START------------
This is the SIGHUP program

Parent process receives SIGCHLD signal
Child process terminated with status 1
```

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./illegal_instr
Process start to fork
I'm the Parent Process, my pid = 3439
I'm the Child Process, my pid = 3440
Child process start to execute test program:
------------CHILD PROCESS START------------
This is the SIGILL program

Parent process receives SIGCHLD signal
Child process terminated with status 4
```

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./interrupt
Process start to fork
I'm the Parent Process, my pid = 3503
I'm the Child Process, my pid = 3504
Child process start to execute test program:
------------CHILD PROCESS START------------
This is the SIGINT program

Parent process receives SIGCHLD signal
Child process terminated with status 2
```

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./kill
Process start to fork
I'm the Parent Process, my pid = 3566
I'm the Child Process, my pid = 3567
Child process start to execute test program:
------------CHILD PROCESS START------------
This is the SIGKILL program

Parent process receives SIGCHLD signal
Child process terminated with status 9
```

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./normal
Process start to fork
I'm the Parent Process, my pid = 3613
I'm the Child Process, my pid = 3614
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
```

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./pipe
Process start to fork
I'm the Parent Process, my pid = 3688
I'm the Child Process, my pid = 3689
Child process start to execute test program:
------------CHILD PROCESS START------------
This is the SIGPIPE program

Parent process receives SIGCHLD signal
Child process terminated with status 13
```

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./segment_fault
Process start to fork
I'm the Parent Process, my pid = 3729
I'm the Child Process, my pid = 3730
Child process start to execute test program:
------------CHILD PROCESS START------------
This is the SIGSEGV program

Parent process receives SIGCHLD signal
Child process terminated with status 11
```

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./stop
Process start to fork
I'm the Parent Process, my pid = 3780
I'm the Child Process, my pid = 3781
Child process start to execute test program:
------------CHILD PROCESS START------------
This is the SIGSTOP program

Parent process receives SIGCHLD signal
Child process stopped
```

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./terminate
Process start to fork
I'm the Parent Process, my pid = 3827
I'm the Child Process, my pid = 3828
Child process start to execute test program:
------------CHILD PROCESS START------------
This is the SIGTERM program

Parent process receives SIGCHLD signal
Child process terminated with status 15
```

```
vagrant@csc3150:~/csc3150/ASS1$ ./program1 ./trap
Process start to fork
I'm the Parent Process, my pid = 3845
I'm the Child Process, my pid = 3846
Child process start to execute test program:
------------CHILD PROCESS START------------
This is the SIGTRAP program

Parent process receives SIGCHLD signal
Child process terminated with status 5
```

Program2:

```
root@csc3150:/home/vagrant/csc3150# insmod program2.ko
root@csc3150:/home/vagrant/csc3150# rmmod program2.ko
root@csc3150:/home/vagrant/csc3150# dmesg
[ 4576.269374] [program2] : Module_init {Zhuanghengmeng} {120090648}
[ 4576.269376] [program2] : module_init create kthread start
[ 4576.269400] [program2] : module_init kthread start
[ 4576.269460] [program2] : This is the parent process, pid = 10550
[ 4576.269460] [program2] : This is the child process the pid is 10551
[ 4576.269462] [Do_Fork] : The return signal is -10
[ 4580.711994] [program2] : Module_exit
root@csc3150:/home/vagrant/csc3150#
```

What I learn

- How to fork, execute, wait and show child process status in user mode.
- The signal converting between kernel mode and user mode
- How to clone, wait, execute in kernel mode.
- Don't be angry when bug exist, there will be more.