Name : Himal Basnet (1001659428), Sailesh Thapa (1001659442)

Professor Jia Rao

CSE 3320-002-Operating System

Project 1 - Process and Thread

February 20, 2020

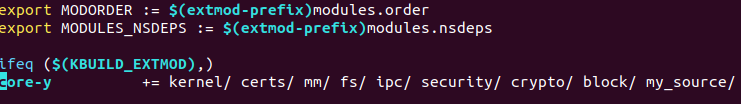
The purpose of this project was to learn to manage processes and threads in a Linux environment.

In this project, we got familiar with linux kernel and learned to build a linux kernel. Following are the steps in building linux kernel and implementing system call:

1. A stable version of linux kernel source package was downloaded from kernel.org
2. The new kernel was configured, compiled, and installed on our machine.
3. Three system call functions were defined in sys\_helloworld.c, sys\_print\_self.c, sys\_print\_other.c respectively under my\_source directory.
4. In the same directory, Makefile was created with following content

**obj-y := sys\_helloworld.o sys\_print\_self.o sys\_print\_other.o**

1. My\_source directory was added in kernel’s Makefile so that compiler knows the path for the new system call.



1. New system call were added in system call table which is in syscall\_64.tbl

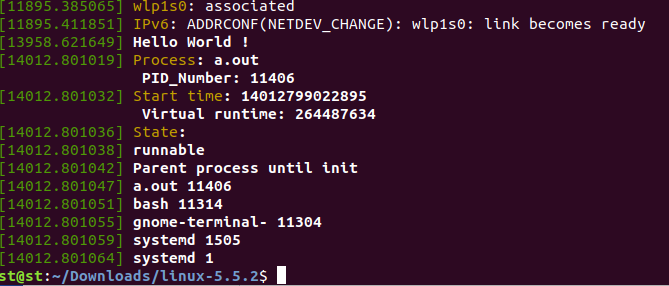


1. New system call were declared in header file syscalls.h

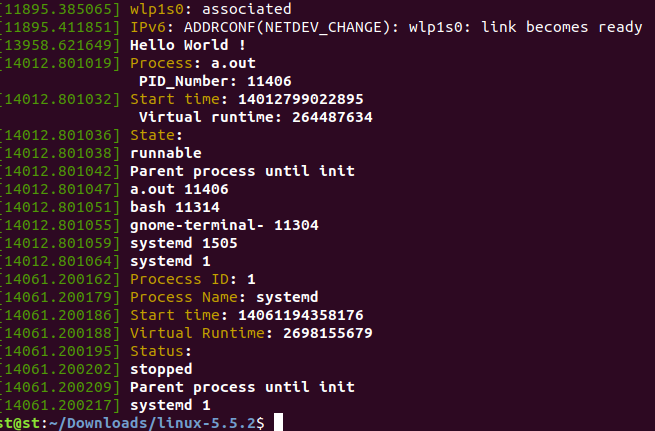


1. The kernel was recompiled and rebooted to fully integrate the system call.
2. To test the system call, test programs were written. The names of the file were test\_helloworld.c, test\_print\_self.c, test\_print\_other.c
3. These programs were compiled and executed to test system calls.
4. The output of the program was seen as expected.

The figure below shows our output.



The output for the first and second system call.



The output for all system call together.

Some of the problems we faced and how we solved it are as follows:

1. For the first time we downloaded Ubuntu in our system we didn’t allocate enough memory so, we had to download it again and configure it to 50GB.
2. The first compilation of our Linux kernel took about 8 hours. The problem was we just used 1 CPU and for the next time we used 3 CPU which decreased the compilation time to 1-2 hrs.
3. Some of the latest kernel versions didn’t work but 5.0.0 and 5.5.2 did work in our system.
4. We were not familiar with VIM for the first time but after this project we are good at it.
5. For the first time we were scared about the project hearing about all 15 millions lines of kernel code but while doing this project we got familiar with some of the files which were really helpful to learn some part of the kernel and how to invoke kernel mode from user mode using system call.