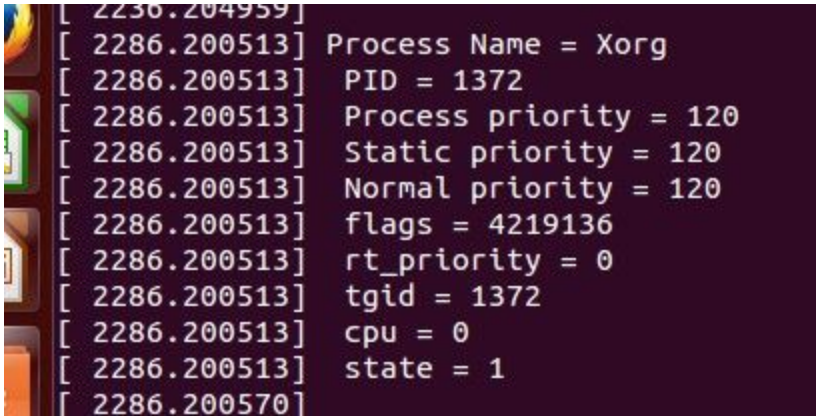
Assignment-1

Saurav Kumar
Ankur Dhaka

1. We downloaded the kernel and extracted the tar.xz file, then we made the folder named sh_task_info, made a file named sh_task_info.c, we wrote the code to print the details of the process corresponding to the process id. We iterate over each process and check whether the process id is equal to the id supplied by the user, then printed every detail related to the process. Then we changed the kernel makefile to include the created system call. Then we changed the file syscalls.h to include the the function prototype of newly created system call, then we included the system call, then we added the system call to the table syscall_32.tbl.
   Compiling:
   1. We used the command “make menuconfig” to create configuration file.
   2. Then we used “make -j 5 KDEB_PKGVERSION=1.arbitrary-name deb-pkg” to compile the kernel.
   3. Then after that for installing the kernel we used dpkg -i linux*.deb
   4. Then after the that we restarted and selected kernel version 3.13 from the grub
   5. Then we made a sample c program and and called syscall() with appropriate system call number and pid for which we want the detail, and then compiled that program using gcc, and then used dmesg command to see the output.
2. The user should give process id as input to the function.

3. Output will be process as follows :



1. Process Name : It give the name of the process.
2. PID : It is the ID of the process.
3. Process priority : It is priority of using the cpu by that process.
4. Static priority : It is the priority given to the process when it is created.
5. Dynamic priority : It is the priority given to the process by the kernel during the life of the process.
6. rt_priority
7. tgid
8. cpu
9. state

4. If the system call returned 0 then the function executed successfully, otherwise it throws an error message when it could not find the process.