Lab One

Title: Add a System to a Linux Kernel

Learning Objectives: Understand the build process of a modern kernel, walkthrough the procedure of adding a system call and the Software Engineering discipline required in Kernel coding

Preparation:

Work with a recent version of Ubuntu —- 20.04 or 20.10. Use that in the VM that you get from CSC or the one that you use at home. My suggestion would be to use Virtualbox so that kernel corruptions can be contained in case you are using your own machine. Type uname -r and you should see that you are working on a version of 5.4.0.xx-generic kernel. The following steps are for a 64 bit version.

Step 1: Download a recent version of the kernel — which will be 5.8.1 something.

wget -P ~/ https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.8.1.tar.xz

Step 2: Get all packages to compile kernels

apt-get install gcc &&
apt-get install libncurses5-dev &&
apt-get install bison &&
apt-get install flex &&
apt-get install libssl-dev &&
apt-get install libelf-dev &&
apt-get install dwarves &&
apt-get update &&

Step 3: Extract the package

tar -xvf ~/Downloads/linux-5.8.1.tar.xz -C ~/

The Syscall Part

Step 1: Create a directory mySyscall

Step 2: Create a file mySyscall/mySyscall.c with some variant of this code

```
#include <linux/kernel.h>

asmlinkage long sys_hello(void)
{
    //printk prints to the kernel's log file.
    printk("Hello world This is abide.YYYY\n");
    return 0;
}
```

Step 3: Create the mySyscall/Makefile with the following line

obj-y := mySyscall.o

Step 4: Alter the top level Makefile by appending mySyscall to the core-y line (Understand the Makefile)

core-y += kernel/ certs/ mm/ fs/ ipc/ security/ crypto/ block/ mySyscall/

Step 5: Add the prototype in include/Linux/syscalls.h (Understand asmlinkage)

asmlinkage long sys_mySyscall()

Step 6: Add the syscall to syscall table

arch/x86/entry/syscalls/syscall_64.tbl

548 64 mySyscall sys_mySyscall

Compile

Step 1: make menuconfig — change name of kernel to abide.YYYY where abide is your name and YYYY are the last 4 digits of your entry number

Either do

make menuconfig

or

make defconfig

Use LOCALVERSION=

Step 2: make

Step 3: sudo make module_install

Step 4: sudo make install

Step 5: sudo update-grub

Reboot

Test

Step 1: uname -r should print 5.8.1

Step 2: Create a file mySyscallTest.c with the following code

```
#include <stdio.h>
#include <linux/kernel.h>
#include <sys/syscall.h>
#include <unistd.h>

int main()
{
    long int mySyscallTest = syscall(548);
    printf("System call sys_mySyscall returned %ld\n", mySyscallTest);
    return 0;
}
```

Step 3: Compile : gcc …

Step 4: ./a.out

Step 5: See the effect in dmesg using sudo dmesg

Check the Last Line

Summit a patch file (create the patch file) containing your changes in Moodle along with the dmesg output.

You will have to reboot your machine and choose the new kernel and demonstrate it works during the demo to get ANY marks in the assignment.