Operating Systems
Lab 3
Linux Kernel Modules

For this lab, you do not need to write any source code. A sequence of three successively more complex kernel modules is provided for you to study. The actual lab activity is just to compile the modules (run make), install each of them, observe their behavior and remove each of them. The more difficult part of the lab is to fully understand the source code. You will need to carefully read the source code and also do some research using the internet (starting with the Linux Kernel Module Programming Guide) and Linux manual pages. The source code is completely provided to you because the commands that are needed are not at all intuitive. However, a great deal can be learned from reading the source code of programs written by others.

Procedures:

Note: You will need to use the root user account on your Linux system to complete the activity.

1. From the study guide, download the tar file, lab3.tar. Extract the file using the command **`tar xvf lab3.tar`**.

2. Look at the Makefile file. See question 2.

3. Type the **`make`** command to compile hello.c, prochello.c and xtime.c to loadable kernel module files. See question 3.

4. First, in another window, tail the system log file (**`tail -f /var/log/messages`**). Now install and remove the module:

   ```
   /sbin/insmod ./hello.ko
   grep hello /proc/modules
   /sbin/rmmod hello
   ```

   Observe that /proc/modules file lists this module as one of the installed modules. See questions 4, 5.

5. Do the same (as above) for prochello.c. In another window tail the system log file (**`tail -f /var/log/messages`**). While the module is installed, cat the referenced file in the /proc file system (/proc/test) several times(**`cat /proc/test`**).

   Notice that this module defines a read function which is executed when the proc file is read. Use the **`ls -ld /proc/test`** command and observe how the output of ls relates to the source code. See questions 6, 7, 8.

6. Do the same (as above) for xtime.c. Wait to remove the module until finished with the lab (after the next step). Verify that the module is installed and works:

```
grep xtime /proc/modules
cat /proc/xtime
```

7. Compile time_test.c (`gcc time_test.c`). Study the source code; run the program; and comment on the observed output. See question 9.

Questions:

1. Based on the Linux Kernel Module Programming Guide (URL on K-State Online), describe what a kernel module is and the advantages of a module compared to functionality compiled into the kernel.

2. What is the directory containing kernel source code used for in the Makefile?

3. What type of files are produced when you type make?

4. Study the source code for hello.c and describe what happens when the module is installed and removed. Look at the man pages for insmod and rmmod and describe what they do.

5. What messages were displayed in /var/log/messages? What is the purpose of this file in a Linux/Unix system?

6. Study the source code for prochello.c and describe what happens when the module is installed, removed and when the proc file is read.

7. For the prochello module, how does the kernel know to run this function on a read request? Contrast what the kernel is doing here to reading a file from the hard drive.

8. For the prochello module, the read function writes a string to the buffer array passed to the function as a pointer, where did this buffer memory come from? Why does the kernel call the read function more than once each time the /proc file is read?

9. What differences exist between the xtime global kernel value and the gettimeofday() system call? Try to explain the difference. You may be able to find some pages on the Internet that will be helpful.