Operating Systems (SFWRENG 3SH3), Term 2, Winter 2023

Prof. Neerja Mhaskar

Extra material for Assignment 1

/proc File System

The `/proc` file system is a "pseudo" file system that exists only in kernel memory and is used primarily for querying various kernel and per-process statistics. This exercise involves designing kernel modules that create additional entries in the `/proc` file system involving both kernel statistics and information related to specific processes.

We begin by describing how to create a new entry in the `/proc` file system. The program example named `hello.c` (included with this PDF) creates a `/proc` entry named `/proc/hello`. If a user enters the command

```
cat /proc/hello
```

the `Hello World` message is returned.

In the module entry point `proc_init()`, we create the new `/proc/hello` entry using the `proc_create()` function. This function is passed `proc_ops`, which contains a reference to a struct `file_operations`. This struct initializes the `.owner` and `.read` members. The value of `.read` is the name of the function `proc_read()` that is to be called whenever `/proc/hello` is read.

Examining this `proc_read()` function, we see that the string "`Hello World\n`" is written to the variable buffer where buffer exists in kernel memory. Since `/proc/hello` can be accessed from user space, we must copy the contents of buffer to user space using the kernel function `copy_to_user()`. This function copies the contents of kernel memory buffer to the variable `usr_buf`, which exists in user space.

Each time the `/proc/hello` file is read, the `proc_read()` function is called repeatedly until it returns `0`, so there must be logic to ensure that this function returns `0` once it has collected the data (in this case, the string "`Hello World\n`") that is to go into the corresponding `/proc/hello` file.

Finally, notice that the `/proc/hello` file is removed in the module exit point `proc_exit()` using the function `remove_proc_entry()`.