CMSI 387-01

OPERATING SYSTEMS

Spring 2014

Assignment 0220

This assignment "breaks through" the command line for an initial taste of operating system programming.

Outcomes

This assignment will affect your proficiency measures for outcomes *2d* and *4a–4f*.

Not for Submission

- If you have it, read Chapters 1 and 2 in SGG.
- You will likely want to do this assignment in a virtual machine such as VirtualBox or VMWare Player. For uniformity, use the latest Ubuntu Linux distribution (*http://www.ubuntu.com*).

For Submission

System Call Practice

Pick at least three (3) Linux system calls *except for the ones demonstrated in the* `syscall-samples` *and* `kernelwrite` *bazaar code* and write "wrapper" programs for each of them, for a total of three (3) new commands written by you.

- Each program must invoke the system call *using* `syscall` *directly* (i.e., no C convenience functions). If the system call result requires custom data structures use definitions from kernel-level header files and not the standard C library.
- Each program must produce some demonstrable, comprehensible result (e.g., output, changes to the file system, etc.).
- This one time, yes, to expose the low-level underpinnings of what you are doing, you may forego including some header files and use operating system constants and literals directly.
- You may use additional system calls to help do your work, but each program should center around one particular system call's functionality.
- You may resort to "normal" C (data structures, libraries, etc.) to process the input or output of your system call (e.g., parsing command-line parameters, producing formatted output, etc.).
- Implement whatever options or switches you like, to make your programs more flexible or powerful. These are your commands now; make them whatever you'd like them to be.

Commit your work (source code only—no build products please) under *homework/my-own-commands*.