Linux Notes

Linux is a free operating system that runs on PCs and that uses the X Window system as its basic graphical interface. One of the main differences between Linux and Windows/MAC-OS is that in Linux, events are application driven; that is, the user selects an application, and then chooses a file that the application should use or modify. Windows/MAC-OS, on the other hand, is more file driven: the user clicks on a file, and then the operating system determines which application to start. As you have no doubt noticed, this often does not work quite right. Many computer scientists prefer Linux because the user is in more direct control as to which application to run. Furthermore, the Linux user has easier access to many system commands. However, Linux is more difficult for casual users for precisely these same reasons – the user must understand the underlying system. This is becoming less of an issue because of continual improvements in Linux distributions that include nice window managers which look and work similar to Windows.

- Window Managers

  When you login, you'll notice that a default window manager (a Graphical User Interface) starts up. These are built on top of X, for which we should all be thankful. There are many flavors of window manager, including: Gnome (gnome.org), KDE (kde.org), LXDE (lxde.org), and more. Each of these set up the screen with varying complexity and user control. The important thing to note about any of these window managers is that you can set up any one you wish, and customize the colors and the look of the windows to your liking. This is something for you to experiment with.

- Linux Distributions

  There are free (and some not free) distributions of Linux, such as Debian (debian.org), Fedora (fedoraproject.org), Red Hat (redhat.com), SUSE (suse.com), Ubuntu (ubuntu.com), and many, many more. Currently in csLab, we are using Ubuntu 13.xx. This distribution is free, easily installed, and comes with a wealth of software, or "packages." To facilitate software installation, Ubuntu comes with a "package manager." The user can easily choose packages to update or install, and Ubuntu will automatically do so. In addition, all of the dependencies are automatically found and any additional required software will be installed. Ubuntu uses Gnome as the default windowing system, but this can be changed to KDE as well.

- Basic Linux Commands

  If you are using one of the more powerful window managers, then you can get away with not typing many Linux commands. However, you might get annoyed when you realize you could do something with one typed command compared to clicking on several folders/icons. Here is a quick tutorial on the very basics to get you started.

  The first thing to realize is that the file system is stored as a tree (upside down). Folders are called **directories** and are denoted with a slash and/or color. The topmost directory is simply `/`. From there, other directories form the branches of the tree. One of these directories is `/student` which contains all of your "home" directories; i.e., the folders that hold all of your files on the server. When you login, you are in your home directory: `/mgousie`.

The following are some basic commands to navigate the file system and to work with files; words in *italics* should be replaced with actual file/directory names. Most commands have options that are added starting with a – (see `ls` below).

Command	Description
Files/Directories	
ls	list contents of current directory (colors denote files/directories)
ls -F	list contents, but put * next to executables and / next to directories
ls -l	list contents with size and other information of each file
pwd	returns current path (i.e., which directory you are in)
mkdir *myDir*	create a new directory called *myDir*
cd *myDir*	move down to the *myDir* directory
cd ..	move up to previous directory
cd */usr/local*	move directly to the directory */usr/local*
	i.e., you can jump to any directory you wish
cat *fun*	show the contents of file *fun*
more *fun*	show the contents of file *fun* one page at a time
less *fun*	show the contents of file *fun*, allowing for scrolling in both directions
cp *old new*	copy file *old* to file *new*
mv *old new*	rename (move) file *old* to file *new*
	Note that you can copy or move across directories
rm *afile*	delete (remove) file *afile*
rmdir *adir*	delete (remove) directory *adir*
Editors	
OpenOffice	includes a typical word processor; start this from menu
soffice	start Open Office from command line; can then choose application
kwrite (or similar) *blip*	start a plain text editor with file *blip*
gedit *blah*	start the gedit editor with the file *blah*; a good choice for beginners
emacs *bloop*	start the emacs editor; much more functionality than you'll ever need
vi *blech*	start the vi editor; extremely cryptic
Compilers	
g++ *program.cpp*	invoke the gnu C++ compiler and create the executable *a.out*
g++ *prog.cpp* -o *prog*	invoke the C++ compiler and create the executable *prog*
./a.out or *./prog*	execute the file *a.out* or *prog*
gcc *program.c*	same as above, but invokes the C compiler
python	start interactive Python session
python *prog.py*	interpret and run *prog.py*
gprolog	start interactive PROLOG session
gplc *prog.pl*	invoke PROLOG compiler
xspim	invokes the SPIM IDE (Integrated Development Environment)
Miscellaneous	
man *command*	displays the on-line manual page for *command* (including all options)
spell *flim*	spell check the file *flim*; returns a list of misspelled words
wc *flam*	check the size (in lines, words, and characters) of *flam*
date	returns the current date and time
cal	shows calendar for the current month
cal *year*	shows calendar for the year *year*
grep *word afile*	look for *word* in the file *afile*
sort *zoi*	sort the lines in file *zoi*
sort *zoi* > *zim*	**redirect** the output of the sort into file *zim* (redirect input with <)
grep *word afile* | sort	**pipe** the output of grep to the sort function
top	shows all currently running processes and usage statistics
history	shows past commands; type "! *n*" to repeat command number *n*
up arrow	repeats last command
lpr *afile*	print the file *afile*