Linux, X, and Emacs*

August 26, 2008

1 Logging In to a Linux Console

In order to use a Linux workstation, you must have an account on cislinux. If you do not, please get one.

The easiest way to use one of the Linux workstations is to directly log in to one of the workstation consoles. A console not in use should display a login screen, which prompts you first for a username, then a password. Enter your cislinux username and password at the prompt. If things go right, you should be presented with the window manager.

A few things can go wrong when you are logging in. If the screen is blank, type any key and the login screen should appear. If the screen displays a screensaver of some form, it has been "locked" by another student (see notes on locking etiquette, below) and you cannot use it. If the screen display looks like a window manager, then another student is logged into the machine and may still be using it.

2 Connecting to a Linux Machine Remotely

You do not have to be physically seated in front of one of the Linux workstations in order to use it. You can access any of the department's Linux machines remotely via an `ssh` client (such as NiftyTelnet on a Mac, `ttssh` on a PC, or `ssh` on another Linux box or a Mac running OS X).

There are two key advantages of connecting to a Linux machine remotely. First, you can access the Linux machines from any other machine on the Internet – a fact which is important when you don't wish to go to Nichols Hall. Second, via ssh you can still use the machines even when all consoles are actively being used (several people can be logged into the same Linux machine at once). This is important to know when the lab areas are crowded with people near an assignment deadline.

3 Using a Shell

You can accomplish many tasks using the window manager by pointing and clicking, just as on a Mac or in Windows. However, some tasks can only be accomplished (or can be accomplished more conveniently) by typing textual commands in an interactive program called a *shell*. For instance, in a shell, you can easily copy all files whose name match a certain pattern from one directory to another – something that can be more tedious using point-and-click interfaces.

If you are unfamiliar with shells, you should read Chapter 4 (*The Unix Shell*) of Larry Greenfield's *The LINUX User's Guide*. This guide is available on-line at

*This writeup is adapted from course notes of Franklyn Turbak

`http://www.cis.ksu.edu/~ab/Courses/505/fall08/docs/linuxUsersGuide.pdf`

which is linked from the *Resource Links* section of the CIS 505 web page. You should also read Scott Anderson's *Introduction to Unix and the X Window System*, available at :

`http://www.cis.ksu.edu/~ab/Courses/505/fall08/docs/unix-intro.pdf`

and linked from the *Resource Links* section of the CIS 505 web page.

There are three standard ways to create a new shell window:

1. Left click on Applications menu; go to System Tools, and select Terminal. (If the terminal icon already exists on the top of the screen, click the icon).

2. Execute `xterm &` from an existing shell window.

3. Create a shell within Emacs via `M-x shell` (see the section on Emacs, below). It is very convenient to have a shell within Emacs, because then any shell command can be easily executed without leaving Emacs.

   One minor drawback of running a shell under Emacs is that Emacs sometimes interprets or prints character sequences in a different way than a separate shell window would. For instance, an Emacs shell will echo passwords that a normal shell would not. Also, the `ls` command in an Emacs shell may print a lot of annoying formatting characters; these can be removed by first executing `unalias ls` in the Emacs shell.

4 Emacs

Emacs is an extensible, customizable, self-documenting text editor created by Richard Stallman. Many consider it to be one of the greatest programs of all time. It is one of the flagship programs of Stallman's Free Software Foundation and GNU project.

You will be doing all programming this semester – writing, executing, and debugging programs in OCaml – using Emacs. In fact, it is possible to do all your work in the course entirely within Emacs. It is very important to become a proficient Emacs user because this will save you a lot of time during the semester.

A standard way to launch Emacs is this:

- Execute `emacs &` from within a shell. The `&` will create a separate Emacs window. If you are working remotely and do not wish a window to be created, instead execute `emacs -nw` (the `-nw` means "no window").

All Emacs documentation, including a tutorial and reference information, are on-line. If you are unfamiliar with Emacs (or have used it before but are rusty), you should takes the on-line Emacs tutorial. You can do this by typing the `Control` and `h` keys at the same time, followed by the `t` key.[1] This will load a interactive tutorial, whose directions you should follow. When you complete the tutorial, you will know how to do basic editing in Emacs.

[1] In Emacs notation, this keystroke combination is usually written `C-h t` and pronounced "control-h t".

The tutorial teaches you keystroke commands for basic Emacs functionality. If you prefer, most of this functionality can instead be accessed by using a combination of the mouse, menu items, and arrow keys. However, I strongly recommend that you learn the keystroke commands, as they will save you lots of time and make it easier for you to work via ssh.

Another useful introduction to Emacs is Chapter 8 (*Editing Files with Emacs*) of Larry Greenfield's *The LINUX User's Guide*. You will find links to this and several sites containing more detailed Emacs documentation on the CIS 505 web page. You should also read Scott Anderson's *Introduction to the Emacs editor*, available at

`http://www.cis.ksu.edu/~ab/Courses/505/fall08/docs/emacs-intro.pdf`

and linked from the *Resource Links* section of the CIS 505 web page. A particularly useful link is the Emacs reference card you can find at

`http://www.cis.ksu.edu/~ab/Courses/505/fall08/docs/emacs-refcard-letter.pdf`

and linked from the *Resource Links* section of the CIS 505 web page. You may want to print out a copy of this card and carry it with you for handy reference.

It turns out that Emacs even has its own hypertext information system. This system contains detailed documentation on Emacs itself, and is worth exploring to find out more about Emacs. In order to access this information system, type the `Escape` key, followed by the `x` key, followed by the character sequence `info`.[2] This will load up an editor buffer that contains a top-level menu of the system documentation. You can browse this system via mouse clicks, much as you browse web pages in a web browser.

5 Logging Out of a Linux Machine

After you are done using a Linux workstation, you need to logout. You can do this from the window manager.

You know that you have succeeded in logging out when you see the Linux login prompt appear.

If you are logged in remotely, you can log out by executing `logout` in the shell created by `ssh`.

It is important not to accidentally leave yourself logged in to a Linux machine when you are done. If you do so, someone may accidentally or purposely read, modify, or delete your files. Also, you will be tying up an important resource.

[2] This keystroke combintation, pronounced "meta-x info", is usually notated as `M-x info`.