Tutorial 1: Introduction to Linux

This tutorial will introduce you to the Computer Science Department's computing environment. The computer labs available for your use are on the main floor of the Math Sciences building. They consist of a combination of machines, some of which run Windows and some of which use the Linux operating system. In this course, you will normally be expected to use the Linux machines. This tutorial includes a gentle but incomplete introduction to Linux. Many textbooks are available which cover this material. Numerous additional tutorials are available on the Internet. You are expected to become familiar with working in Linux without extensive instruction on this topic.

Logging In:

Before you can log in, you must activate your account. Your account is activated by visiting the help desk located inside the computer lab. When you visit the help desk, you will be asked to provide your student ID and agree to the acceptable usage policy. Your password will be provided to you once you agree to the policy, and your account will become active approximately 15 minutes later. If possible, you should try and visit the help desk before you first lab to avoid any lineup and the 15 minute delay before you can log in.

The password that you are provided with initially will only work on the Linux machines. To gain access to the Windows machines, you must change your password. The process of changing your password is described later in this document.

Opening a Terminal:

Much of the work done in this course will involve typing commands instead of clicking on icons. A terminal window provides a place for you to type these commands. To display a terminal window, click on the Applications button in the upper left corner of the screen, select Accessories, and then select Terminal.

You can type commands into the terminal window once it is displayed. Initially, you will probably find this more difficult than clicking on icons. However, using the terminal lets you perform some tasks that are difficult or impossible to complete within the point and click environment.

Changing Your Password:

You are not required to change your password, but may find it beneficial to change your password to something that is easier for you to remember while still being difficult for others to guess. You may also want to change your password so that you can log into the Windows machines. The password that you select is case sensitive, meaning that the letter A and the letter a are treated as different characters. When selecting a new password, you must choose something secure. The new password should be at least 8 characters long, include a mixture of both upper and lower case letters and also include one or more numbers. It should **not** include a word that appears in the dictionary, your name, the name of a family member or pet, or a number associated to you such as your student number, phone number, date of birth or postal code.

The command `passwd` is used to change your password. Type this command into the terminal window and press `enter`. When prompted, enter your old password and then enter the new secure password that you have selected. You will be asked to enter your new password a second time to ensure that you typed it correctly.

Once you change your password, you will be able to use the new password to log into both the Linux and Windows machines. Note that it will take approximately 15 minutes for the new password to propagate throughout the network.

Introduction to Linux:

A myriad of commands are available for use in the terminal window. A very small fraction of them will be highlighted in this document. Numerous introductory Linux tutorials are available on the Internet. (A web browser is available on the lab machines by selecting the Web Browser icon at the top of the screen, or by selecting Applications, Internet, Firefox Web Browser). You can also get information about a specific command using the `man` command described below.

Like Windows, Linux stores information in files. In order to keep things organized, directories are used to hold groups of related files and directories. When you log into the computer and open a terminal window, you will be working in your home directory. This is a directory set aside specifically for your use. Entering the command `pwd` and pressing `enter` will display the path to the directory you are currently working in (in this case, your home directory). For example, my home directory resides at `/home/profs/bdstephe`. This path tells me that my home directory is inside a directory named `profs`, which is inside a directory named `home`.

I can view the list of files in my home directory by using the list command, `ls`. If your home directory is empty then the `ls` command will complete without displaying anything.

On Linux, many commands take optional parameters. These parameters are additional pieces of information that follow a command to control its behavior. For example, providing a `-la` parameter after the `ls` command displays all files, including hidden files, in long output format. Using the `ls -la` command will display at least two lines of output because your home directory includes directories named `.` (the current directory) and `..` (the directory one level above your home directory).

Directories are used to organize information. Without them, you will end up with so many files in your home directory that it will become difficult to keep track of them. You will probably want to create a new directory for each assignment (and possibly each tutorial) in this course. As you take additional computer science courses, you may want to extend this idea so that you have one directory for each course. Then you would create a subdirectory for each assignment and tutorial within the directory for the course.

Use the make directory command, `mkdir`, to create a new directory. The `mkdir` command takes one parameter, which is the name of the directory to create. Note that the parameter does **not** begin with a `-` character. For example, entering the command `mkdir tutorial01` will create a new directory named `tutorial01`. You can confirm that this directory was created by using the `ls` command.

You can change the directory you are working in by using the change directory command, `cd`. Like `mkdir`, `cd` takes the name of the directory that you want to move into as a parameter. For example, `cd tutorial01` will move you into that directory. You can verify that you have moved into the new directory by using the `pwd` command. Using `ls` will reveal that the newly created directory is empty. Using the `cd ..` command will allow you to move up one level and return to your home directory.

The `ls`, `mkdir` and `cd` commands all provide far more functionality than what is described in this document. A detailed description of each command can be found on its manual page. Manual pages are accessed using the `man` command. For example, the command `man ls` will display all of the different parameters that can be passed to the `ls` command. Use `space` or `enter` to display the remainder of the manual page for `ls`. Pressing `q` will exit the manual page without displaying the remaining information. Additional information on using the manual page system is available on the manual page for `man`.

The commands described in the preceding paragraphs provide an initial minimal set of commands to begin organizing the files that you will use in this course. You will also want to learn how to copy files (`cp`), move and rename files (`mv`), and remove files and directories (`rm` and `rmdir`). At a minimum, the other commands that you should explore using the manual pages and Internet include `head`, `tail`, `diff` and `grep`.

Editing Files:
New files can be created using your favorite text editor. Several different editors are available on the system for your use. Of these editors, popular choices include Emacs, Pico and Vi. Tutorials for each of these editors are available on the Internet, and help on at least one of these editors should be available from your TA. Note that your TA will not be an expert in each of these editors as they likely only use one of them with any frequency. You are welcome to select whichever editor you would like. I strongly encourage you to learn to use your editor effectively to help maximize your productivity. At a minimum, you should know how to perform tasks such as search and replace, and jump to a specific line. Enabling color syntax highlighting and fence matching (a feature where the editor highlights the corresponding open parenthesis when a close parenthesis is typed) will also be helpful because these features will help you detect errors more quickly.

Printing:
Printers are available in the computer labs. When your account is created it will include 25 pages of printing. Additional pages of printing can be purchased from the Computer Science Department office on the 6th floor of the ICT building.

Files are printed using the `lpr` command. The name of the file that you want to print is provided as a parameter to lpr. For example, if you want to print the description for assignment one, you could use the command `lpr assignment1.pdf` (assuming that assignment1.pdf resides in the directory that you are currently working in).

Logging Out:

When your work is complete, you can log off of the computer by selecting Log Out from the System menu at the top of the screen. Always ensure that you Log Out before leaving the computer. Do not turn the computer off.

Remote Access:

The computer science department provides several compute servers which can be accessed from around the world. These servers provide functionality similar to terminal running on the Linux machines in the lab. Additional information about these servers is available at:

http://www.cpsc.ucalgary.ca/tech_support/services/remote_access

Accessing the servers remotely requires software that supports the secure shell protocol (SSH). Links to a number of SSH clients are available on the page referred to previously, including puTTY, which is available without cost. The UCIT Windows install of SSH, also available through a link on the web page listed above, is another free alternative that also supports secure file transfer (more on that below!).

To use the UCIT SSH application, click on the "Secure Shell Client" icon (or start the program through the start menu). In the window that pops up, select the "Quick Connect" option. In the "Connect to Remote Host" window that pops up, enter the host name (one of the computer science servers listed in the web site mentioned above) and your computer science user name. Make sure that the port number is 22 and the authentication method is "Password," and then click "Connect". After entering your computer science account password, a secure terminal window, with similar functionality to the terminal windows you have when working on the linux machines in the lab, will be available. To save time, it is also possible to create a profile corresponding to a particular server, so you don't have to enter the server name and your user name every time you connect. After connecting, a window pops up for a few seconds with the option to name and add a profile corresponding to your current connection.

Moving Files to or from Home:

You may elect to do some of your work for this course on your home computer or your laptop. If you choose this option, you may find it necessary to move files between your own machine and the lab computers. Two easy options for moving files include using a USB drive or copying them across the Internet.

USB Drive:

USB Drives used with Windows machines can also be used on Linux. If you plug your USB drive into the USB port on the front of the machine, Linux should mount it automatically. You can access the files on your USB drive by double clicking on the Computer icon on your desktop, and then doubling clicking on the icon for your USB drive.

The files on your USB drive can also be accessed from the terminal. USB drives are mounted in the `/media` directory. If you execute the command `ls /media` you should see a subdirectory that represents your USB drive. My USB drive was made by Kingston so the files on it appear in the directory

`/media/KINGSTON`. You can copy files to and from your USB drive using the same commands that you have used for copying files previously.

Network:

Files can be copied over the Internet using sFTP, a SSH-secured version of the File Transfer Protocol (FTP). Modern versions of Windows include support for accessing files stored on a regular FTP server, but this should in general be avoided, especially for files that may contain sensitive information, as the regular FTP transfers files without any security. The sFTP application does protect the files in transit.

To use the sFTP application, install the UCIT ssh install referred to above. Once that is installed, you may initiate a secure file transfer session by starting the "Secure File Transfer Client." The starting procedure is the same as for the ssh client, and you also will be presented with the option of saving a profile of your current connection. Once successfully logged in, you will see a vertically-split window with a list of your Windows machine's files on the left and the remote server's files on the right. The directories in both parts of the window can be traversed in the usual Windows-based manner. Transferring files from the remote server is done either by right-clicking and selecting "download" or by drag-and-drop from the right side of the sftp window to whatever destination folder (including the desktop) you wish on your Windows machine (located in the left side of the window). Transferring files to the remote server can be done either be selecting the files in the left side of the window, right-clicking, and selecting "upload", or by drag-and-drop.

Getting Additional Help:

Your teaching assistant is available to help you during your scheduled laboratory / tutorial time. "Continuous Tutorial" teaching assistants are also available in the lab at various times through the week. You are free to seek guidance from these teaching assistants any time they are in the lab. However, the teaching assistants will not complete lab exercises or assignments for you.

Questions:

Using online resources, or an introductory UNIX textbook, answer the following questions.

1. What is a path?
2. What is the difference between a relative path and an absolute path?
3. What is a wildcard?
4. Why are wildcards useful?
5. How do you see a list of the commands you have used previously?
6. What is the `find` program used for?
7. How can you use `find` and `grep` together in order to locate a file that resides in your home directory, or one of its subdirectories?