CISP 360 Course Tools:
Linux, Vi, G++, Pine and WS-FTP

General Information

Programs written for CISP 360 are to be created in the Linux environment. This document describes logging into the Linux server, Linux commands commonly helpful to CISP 360 students and the use of various tools.

All work will be done through Linux's command line (non-GUI) interface. The mouse will therefore be mostly useless in this environment. One notable exception is that it can be used to select, copy and paste text. Please also note that Linux, and its tools, are case sensitive.

Logging into the Linux Server

Logging into the server requires a program which uses the telnet protocol. A telnet icon can be found on the desktop of Business Division lab computers. Double-click this icon and log in as described under *Log Into the Linux Server*.

Logging Into the Server From Home

If you would like to log into the server from home, you will need to log into your Internet Service Provider (ISP) and use a telnet program. A program called Telnet comes with the Windows operating system (any version) and can be used to log into the server. Students have had problems using this program, however, and it is therefore not recommended.

One freeware telnet program for Windows machines is Putty. It can be downloaded from www.tucows.com. To configure Putty after downloading the **putty.exe** file from Tucows:

1. Create a Windows shortcut for the **putty.exe** file (right click the executable file and click **Create Shortcut**).
2. Open the shortcut's **Properties** window and click the **Shortcut** tab.
3. Add the server's domain name (cis.scc.losrios.edu) or IP address (165.196.201.5) in the **Target:** box <u>after</u> the text already in the box. Click **OK**.

To run Putty, double-click the shortcut instead of the executable file.

Log Into the Linux Server

1. Start the telnet program. If you are logging in from home, and are not already logged into your ISP, you will be asked to log in.
2. Type your Linux ID at the **login** prompt (all lower case with no spaces) and press **Enter**.
3. Type the default password provided by your instructor at the **Password** prompt (all lower case with no spaces) and press **Enter**.
4. If you type either your ID or password incorrectly you will be asked to try again. After three attempts you will need to reconnect to the server.
5. When logged into the server, your current directory will be your home directory.

If you are logging in for the first time, give yourself a new password. Please see the **passwd** command in the *Linux Commands* section of this document.

Linux Commands

Linux commands are similar to, if not exactly the same as, Unix commands. Some commands are also similar to those used in DOS. The commands given below are often helpful to CISP 360 students. They are grouped together solely for organizational purposes.

Options have the form **-x** and must be separated from commands by a space. If you would like to use more than one option at the same time, list them after the **-** sign. For example, to list all of the files in the current directory, including hidden files, in long format, type **ls –la**. Hidden files are files that have names which start with a period.

	Command	Description	Examples/Notes
Miscellaneous	passwd	Change your password. You will be asked to enter the current password and your new password twice. Choose a password of six or more letters and numbers. Remember: Linux is case sensitive.	
	exit	Log out of Linux	
	clear	Clear the screen	
	lpr0	Print a file to the <u>server's</u> printer (0 is zero)	`lpr0 work.cpp`
	man <u>or</u> info	Obtain information (help) about a specific Linux command. Press the **spacebar** to display the next screen or **q** to quit.	• `man rm` • `info ls`
Directory Related	pwd	Display the full path name of the working (current) directory.	
	ls	List the files and subdirectories in the current directory.	• `ls` (List the directory's contents.) • `ls -l` (List the contents in long format.) • `ls –a` (List hidden files.)
	mkdir	Create a subdirectory.	`mkdir dirname` (Create the subdirectory dirname.)
	cd	Make another directory your current directory (i.e. change directory).	• `cd dirname` (Make the dirname subdirectory the current directory.) • `cd ..` (Make the parent directory the current directory)
	rmdir	Remove (delete) an empty subdirectory.	`rmdir dirname` (Delete the dirname subdirectory.)
File Related	cat	List the contents of a file to the screen.	`cat file1`
	more	Display the contents of a file a screen at a time. Press the **spacebar** to display the next screen or **q** to quit.	`more file1`
	cp	Copy a file	• `cp file1 file2` (Copies file1 to file2 creating file2 or overlaying it if it already exists.) • `cp file1 dirname` (Copies file1 into dirname subdirectory overlaying file1in dirname if it already exists.)
	mv	Move or rename a file or directory.	• `mv file1 file2` (If file2 does not exist, file1will be renamed file2. If file2 exists, file2 will be overwritten.) • `mv file1 dirname` (Moves file1 into the dirname subdirectory.)
	rm	Remove (delete) a file	`rm file1`

Vi (Visual Interface)

The vi text editor is a standard Unix editor. Vim, (Vi IMproved), a vi clone is used on the server. Most of Vim's commands are the same as vi's. The editor will be referred to as "vi" throughout this document.

<u>Start Vi</u>

Start the editor by typing **vi** and a file name, for example **vi myprog.cpp**, and press **Enter**. Be sure to include the **.cpp** extension as the editor will not assume you are creating a C++ source code file. If the file does not exist, the editor will assume you are creating a new file with the name provided. If the file does exist, the editor will open the file and display its contents.

If you type **vi** and press **Enter** without typing a file name, Vim's main screen will appear. Type **:q** and then **Enter** to return to the Linux prompt so you can restart the editor.

<u>Modes</u>

When working in vi you will either be in command or insert mode. Command mode is the mode in which the commands below can be used. It is the default mode.

Insert mode is the mode in which you can actually type new text. Enter insert mode by using a command listed under *Insert Commands* (the command most commonly used is **i**). Press **Esc** to return to command mode.

<u>Vi Commands</u>

Commands are grouped together solely for organizational purposes. Please note: A word is a sequence of characters until punctuation or a space occurs. A typed line ends when **Enter** is typed. As a result, a line which wraps is still considered by vi to be a single line.

	Command	Description	Command	Description
Insert Commands	i	Insert text before cursor.	o	Insert (open) blank line below current line.
	A	Append text to end of current line.	O	Insert (open) blank line above current line.
Save/Quit Commands	:wq	Write (save) file and quit vi.	:w	Write file without leaving vi. Use `:w filename` to save file with a new name.
	:q!	Discard changes and quit vi.		
Movement Commands	w	Move ahead one word.	H	Move to top of window (Home).
	b	Move back one word.	M	Move to middle of window.
			L	Move to last line in window.
	0	Move to start of current line.	$	Move to end of current line.
	^F	Scroll forwards (down) one window.	^B	Scroll backwards (up) one window.
	:**n** *or* **n**G	Go to line **n** in file. Use **1** for **n** to go to the first line in the file.	G *or* :$	Go to the end of file.
Replace/ Change Commands	r	Change (replace) the character under cursor.	cw	Change word.
			ncw	Change **n** words.
	R	Replace multiple characters starting at character under cursor.	cc	Change line.
Copy Commands	yw	Copy word.	yy	Copy line.
	nyw	Copy **n** words.	**n**yy	Copy **n** lines.
Delete Commands	x	Delete character under cursor.	**n**dw	Delete **n** words.
	nx	Delete **n** characters.	dd	Delete current line.
	dw	Delete word.	**n**dd	Delete **n** lines.

	Command	Description	Command	Description
Put (Paste) Commands	p	Put deleted/copied text after or below cursor.	P	Put deleted/copied text before or above cursor.
Search	/**x**	Search forward for text **x**.	/ *or* n	Repeat previous search forward.
Search and Replace	:s/**x**/**y**/g**n**	Replaces every occurrence of **x** with **y** for the next **n** lines (including the current line).	:%s/**x**/**y**/	Replaces first occurrence of **x** on a line with **y** for every line in the file.
	:%s/**x**/**y**/g	Replaces every occurrence of **x** in the file with **y**.		
Misc Commands.	:set nu	Display line numbers.	^g	Displays current line number.
	u	Undo last change (can be repeated to undo history).	J	Join two lines.
	:l	Displays current line, including non-printing chars (l is lowercase L).		

Swap Files

While working in vi, your unsaved changes are written to a swap (temporary) file. If vi is not exited cleanly (e.g. the power goes off), the swap file will remain in your directory. The next time you open your file to work on it, vi will see the swap file in your directory and display a screen stating there may be changes to recover.

Whether there are actually any changes to recover is your decision. If you do want to recover unsaved changes, type **:recover** while your file is open in vi and then save your file with the changes incorporated.

After you have recovered your changes, or decided there are none to recover, delete the swap file from your directory or else every time you open the file you will again be told there may be changes to recover. A swap file's name is based upon the name of the file that was being modified. For example, if the file was named **myprog.cpp**, the swap file would be named **.myprog.cpp.swp**. As swap files are hidden files, use the **-a** option with **ls** to see them. The **rm** command can be used to delete them (e.g. **rm .myprog.cpp.swp**).

Additional Vi and Vim Resources

Vi Lover's Home Page: www.thomer.com/thomer/vi/vi.html   Vim Home Page: www.vim.org
Learning the vi Editor, Linda Lamb and Arnold Robbins, O'Reilly & Associates, Sixth Edition, 1998.

G++ Compiler

The C++ compiler for CISP 360 is g++. An overview of using g++ and a section on running executable files is provided in this section.

Compile a Source Code File

To compile a program, call g++ using the following general form. The compiler options are described in the table.

g++   option(s)   source code file name

Option	Description	Example
-Wall	Tells the compiler to display all warning messages. Error messages will always be displayed.	`g++ -Wall demo.cpp`
-o	Tells the compiler to give the executable file a name other than the default name of **a.out**. This option is always followed by the name to be given to the executable file.	`g++ -o demo demo.cpp` (Executable file will be given the name **demo**).

More than one option can be used at a time. Both the **-Wall** and **-o** options should always be used to ensure all warnings are displayed and to give the executable file a meaningful name. For example, to compile **myprog.cpp** and give the

executable file the name **myprog** (it is common to give an executable the same name as the source code file), type the following at the Linux prompt:

> **g++ -Wall -o myprog myprog.cpp**.

The name of the source code file and the options used can appear in any order. It is important, however, that the **-o** option and executable file name not be separated. These examples are equivalent to the one above:

> **g++ -o myprog myprog.cpp -Wall** and **g++ myprog.cpp -o myprog -Wall**

Please note:
- The compiler will assume the source code file is in the current directory.
- Only source code files with **.cpp** extensions will be used by g++.
- The .exe extension for executable file names is not used in Unix/Linux environments.
- Accidentally using the source code file name as the executable file name in the call to g++ (e.g. g++ -o myprog.cpp myprog -Wall) has been known to destroy the contents of the source code file.

Running a Program

Linux does not, by default, search the current directory for an executable file when its name is typed at the prompt to run it. Type **./** in front of an executable name to tell Linux the file is in the current directory. For example, to run a program named **myprog** type **./myprog** at the prompt and press **Enter**.

Pine (Program for Internet News & Email)

Pine is a menu-driven email program. A tutorial is available at www.washington.edu/pine/tutorial.4.

To start Pine, type **pine** at the Linux prompt and press **Enter**. Pine's commands are given at the bottom of the screen. If a command appears with a caret (^), the **Ctrl** key should be held down when the letter is pressed. For example, the command to send a message is **^X** (Ctrl-X). Either uppercase or lowercase letters can be used for commands.

To create an email message, select **Compose Message** from Pine's main menu. A screen will appear similar to other email programs (e.g. To line, Subject line, etc.). To attach a file(s) to your message:

1. Use the down arrow to move the cursor to the **Attchmnt** line.
2. Use **To Files** (**^T**) to display the files and folders in your home directory.
3. If the file to attach is in a subdirectory, use the arrows to highlight the name of the subdirectory in which the file is located and press **Enter**.
4. Use the arrows to highlight the name of the file to attach and press **Enter** to select/attach the file.
5. Repeat steps 2 through 4 for each additional file you want to attach.

Messages are stored in folders. The default folder for messages sent to you is the **INBOX** folder. The **sent-mail** folder is the default folder where copies of messages you sent to others are saved. If a message is sent to an invalid address, you may not receive any notice that the message was not delivered.

A deleted message can be "undeleted" unless you answer yes when Pine asks you if you want to expunge messages in a specified folder.

Transferring Files

Transferring a file to or from the Linux server requires using an ftp program. One such ftp program is WS_FTP LE. Configuring and using WS_FTP LE is described below.

<u>Configuration and Login</u>

1. Start WS_FTP LE. A **Session Properties** window will appear.
2. If using your own computer:
    a) Type the server's domain name, **cis.scc.losrios.edu**, in the **Host_Name/Address** box.
    b) Make sure **Unix (standard)** is selected in the **Host_Type** box.
3. Type your Linux ID in the **User ID** box, your password in the **Password** box and click **OK**.

The **Sessions Properties** window will close, and the main WS_FTP LE window will display, as folders, the local drives on the left and the contents of your home directory on the right.

<u>Transferring a File from the Server to a Diskette in the A: Drive</u>

1. Make sure the directory you want to copy from is the current directory in the server window.
2. Select the file(s) to copy. Click each file once while holding down the **Ctrl** key.
3. Under **Local System**, double click **[-a-]** to select the A: drive.
4. Select the **ASCII** radio button.
5. Click the ← button to transfer files to your diskette. The files will appear on the left when transferred.
6. Click the **Exit** button to log out of Linux and close WS FTP.

Please note that transferred file(s) are copied, not moved. If a file(s) already exists at the location to which it is being copied, is will be overwritten.