Chapter 2
UNIX Utilities for Non-Programmers

Graham Glass and King Ables,
UNIX for Programmers and Users,

Original Notes by Raj Sunderraman
Converted to presentation and updated by
Bing Li
Getting Started

• Obtain an account (snowball.cs.gsu.edu) -OR- install Linux

• Logging in
  • locally: enter username and password
  • remotely: use the ssh/telnet utility to sign on to

• Upon logging on, you will see a prompt (usually $ or %) which is displayed by a special kind of program called a SHELL program.

• To set your password, use passwd
Shells

- Popular shells: (Read-Eval-Print-Loop?)
  - Bourne shell
  - Korn Shell
  - C Shell
  - Bash (Bourne Again SHEll)
  - Zsh

- All have common core functionality, some differences.

- Each shell has its own programming language. (Shell programming)
Running Utilities

• To run a utility, simply type the name of the utility after the prompt.

• Some utilities: ls, cd, man, clear, passwd

• Some utilities are programs in bin folder.

• To logout, enter CTRL-D (or exit)
Special Characters

- List meta characters with `stty -a` command
- `^` means control
- `^C` means hold down control key (CRTL) and press the C key.
- `man stty`
Special Characters

- erase: CTRL-H  erase character before cursor (backspace)
- werase: CTRL-W  erase word before cursor
- kill: CTRL-U  erase entire line

To test these:

- type some input like “one two three four” at the terminal prompt.
- move cursor around
- try the above characters
Special Characters

• intr: CTRL-C    interrupt running program
• susp: CTRL-Z   suspend running program
• stop: CTRL-S/CTRL-Q stop printing to screen
• eof: CTRL-D    give program end of file

To test these:

• Try a command like `find *`
• For eof, try `cat>testfile` then CTRL-D on new line
Example of cat

```bash
bli6@bli6-VirtualBox:~/Documents$ cat > letter
Hi Mom,

Please send me money!

David
bli6@bli6-VirtualBox:~/Documents$ ls -l letter
-rw-rw-r-- 1 bli6 bli6 38 Dec 25 15:07 letter
bli6@bli6-VirtualBox:~/Documents$
```
Some common UNIX utilities

- `pwd`
- `cat, more, head, tail`
- `ls, cd`
- `mv, cp, rm`
- `mkdir, rmdir`
- `file, wc`
- `vi`
- `chmod`
List Files (/s)

- `ls -algFsdR <file-spec>

- options
  - a: hidden files
  - l: long listing
  - g: group
  - F: put character after file name indicating (executable*, link@, directory/, socket=)
  - s: num. disk blocks
  - d: dir details not contents
  - R: recursive listing
Show File Contents(*cat, more*)

- **cat** <file-spec-list>
  - list contents of file(s) on screen without pause

- **more** <file-spec-list>
  - same as cat; pauses after each screen -more-
  - *space bar* takes your to next screen
  - *q* quit
  - *enter* shows next line
  - *h* help key for more commands
Show File Contents (head, tail)

- `head -n <file-spec>`
  - display n lines from front of file

- `tail -n <file-spec>`
  - display n lines from end of file
Working with Directories

- `mkdir <dname>`
  - make a new directory

- `cd <dir>`
  - built-in to shell
  - changes shell to a different directory

- `pwd`
  - print working directory
Rename Files \((mv)\)

- rename files (simply change labels in the file hierarchy)
  - \textit{mv} \(-i\) \texttt{<old> <new>}
    - \texttt{-i} for inquire
  - prompt if \texttt{<new>} already exists
  - \textit{mv} \(-i\) \texttt{<file-spec> <dir>}
  - \textit{mv} \(-i\) \texttt{<dir> <dir>}

Copy Files ($cp$)

- $cp$ -i <old> <new>
  - *i* for inquire
  - prompt if <new> already exists

- $cp$ -i <file-spec> <dir>

- $cp$ -ir <file-spec> <dir>
  - *r* for recursive
Removing Files (rm,rmdir)

- remove (delete) file
  - \textit{rm -fir <file-spec>}
    - \texttt{f}: force; inhibits all prompts/messages
    - \texttt{i}: inquire
    - \texttt{r}: recursive
- Be careful! This is a permanent deletion!
- remove (delete) directory
  - \textit{rmdir <dir>}

File Attributes

- file type: - regular, d dir, b buffered file disk drive, c unbuffered file terminal, l link, p pipe, s socket.

- permissions: Owner/GroupMembers/Public R/W/X 4/2/1

- hard link count
- file owner
- file’s Group
- file size in bytes
- file modification date
- file name
File Type

- `file <fname>`
  - ascertains the type of file (ascii/binary etc.)
  - example:

```
bli6@bli6-VirtualBox:~/Documents$ file test
test: ASCII text
```
File Type

bli6@bli6-VirtualBox:~/Documents$ cd ..
bli6@bli6-VirtualBox:~$ file *
Desktop: directory
Documents: directory
Downloads: directory
examples.desktop: UTF-8 Unicode text
Music: directory
Pictures: directory
Public: directory
Templates: directory
Videos: directory
bli6@bli6-VirtualBox:~$
Change File Permissions

- change file mode. (**chmod**)  u,g,o,a  r,w,x,s
  - user, groups, other, all
  - read, write, executes, (set)
  - plus (+) adds permissions
  - minus (-) subtracts
- Also allows octal numbers
  - 7/7/7
  - 4/2/1
Change File Permissions

• Examples
  
  • chmod -R XXX <file-spec>
  
  • chmod -R g+w <file-spec>
  
  • chmod -R u-rw <file-spec>
  
  • chmod -R u+w,g-r <file-spec>
Change File Permissions

- Notice

  - $7 = 111$ (binary) r “on”, w “on”, x “on”
  - $6 = 110$ (binary) r “on”, w “on”, x “off”
  - $5 = 101$ (binary) r “on”, w “off”, x “on”
Change File Owner

- Change owner of file
  - `chown <new-owner> <file-spec>`
- only super user can do this
The VI Editor

- Visual text editor
- Very common on Unix systems
- Created by Bill Joy in 1976
- Many versions exist, including vim
  - vi improved
The VI Editor

• type `vimtutor` in terminal, learn vim by yourself

• Command mode vs Text Entry Mode
  
  • To enter text entry mode use following command
    
    • i,l,a,A,o,O,R
  
  • To get back to command mode use ESC

• http://www.lagmonster.org/docs/vi.html
The VI Editor

• Cursor Movement
  • up arrow \( k \)
  • down arrow \( j \)
  • left arrow \( h \)
  • right arrow \( l \)
  • beginning of line \( 0 \)
  • first non-whitespace char in line \( \wedge \)
  • end of line \( $ \)
The VI Editor

- Cursor Movement
  - next word \( w \)
  - previous word (back) \( b \)

- Goto line number \( nn \)
  - :nn

- Goto last line
  - :$
The VI Editor

- CTRL-D (down half screen)
- CTRL-U (up half screen)
- CTRL-F (forward one screen)
- CTRL-B (back one screen)
The VI Editor

- Deleting Text:
  - delete character under cursor \( x \)
  - delete word \( dw \)
  - delete line \( dd \) (also \( D \))
  - delete many lines \( \text{:<range>}d \)

- Replacing Text:
  - replace char under cursor \( r \)
  - replace word under cursor \( cw \)
  - replace entire line \( cc \)
  - substitute next 4 characters \( 4s \)
The VI Editor

• Pasting Text:
  • :<range>y                          (yank lines)
  • 5yy                                     (yank 5 lines)
  • p
  • P
• :nnp                                    (after line nn)
The VI Editor

• Searching:
  • search forward /sss/
  • search backward ?sss?
  • next n
  • next in opposite direction N

• Searching/Replacing:
  • :<range>s/sss/ttt/
  • :<range>s/sss/ttt/g
The VI Editor

• Saving/Loading:
  • write to file :w <fname>
  • write again :w
  • write :<range>w <fname>
  • write and quit :wq
  • edit new file :e <fname>
  • read (insert a file) :r
The VI Editor

• Misc
  • re-draw screen                  CTRL-L
  • execute a shell command         :!command

• Quitting:
  • quit                           :q
  • write and quit                 :wq
  • quit without writing           :q!
  • exit and write if changed      :x