

CS-344 - Unix Operating System Fundamentals

Lecture 2

Touring the Features of UNIX/Linux & Editing with the Visual Editor

Based on slides created by Dr. Bangalore for the Spring 2005 offering of the course

Syntax for UNIX Commands

- *command* *[-]option(s)*
[option argument(s)]
[command argument(s)]
- Examples:
 - *ls*
 - *ls -la*
 - *ls -la m**
 - *lpr -Pspr -n 3 proposal.ps*

Sample Commands

- ❑ Identifying current directory
 - `pwd`
- ❑ Listing files
 - `ls`
- ❑ Passing arguments
 - Long listing: `ls -l`
- ❑ Listing processes
 - `ps`
 - `ps -u <userid>`
 - `ps -ef`

9/1/2005

4

Miscellaneous

- ❑ How does the shell distinguish arguments from utilities?
 - `A B C | D E | F > G`
- ❑ Repeat commands (depends on the shell)
 - Repeat previous command - `!!`
 - Repeat a command starting with some string - `!p` (execute last command starting with p)
- ❑ History - using the "*history*" command
- ❑ Shortcuts - using the "*alias*" command
 - Type: `alias h=history` and press ENTER
 - When you type h, the shell will execute the corresponding utility

9/1/2005

5

"man" Command

man provides on-line documentation of the UNIX system itself.

```

mmat.cis.uab.edu - UAB - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
Reformatting page. Please Wait... done
User Commands man(1)
NAME
  man - find and display reference manual page
SYNOPSIS
  man [-.] [-+dfrc] [-M path] [-T manro-package] [-
  +R section] [name ...]
  man [-M path] -k keyword ...
  man [-M path] -f file ...
DESCRIPTION
  The man command displays information from the reference
  manuals. It displays complete manual pages that you select
  by name, or one-line summaries selected either by keyword
  (k), or by the name of an associated file (f). If no
  manual page is located, man prints an error message.
Source Format
--More-- (1)
Connected to host.cis.uab.edu 32x2 - sess28-ebc - /mac-ml3 - none - 88x24
    
```

9/1/2005

6

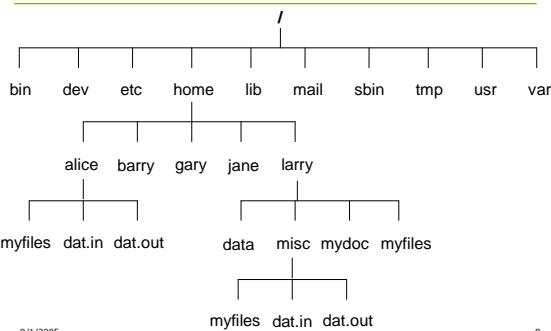
Directory Structure

- ❑ Similar to "Folders" in Windows
- ❑ Natural way to organize files into directories and sub directories
- ❑ "/" (forward slash) is used as a separator between directories and files
- ❑ "/" also denotes the root directory (highest directory in hierarchy)
- ❑ When we login we will be in the HOME directory (e.g., /home/hernadf)
- ❑ "pwd" prints the current directory we are in
- ❑ "cd" is used to change directories

9/1/2005

7

Typical Directory Structure



9/1/2005

8

Working with Directories

- ❑ Change directory
 - `cd <directory-name>`
 - `cd`; `cd mydir`; `cd ./mydir`; `cd ..`; `cd ../misc`; `cd ~/mydir`;
- ❑ Create directory
 - `mkdir <options> <directory-name>`
 - `mkdir mydir`; `mkdir /home/puri/mydir`; `mkdir ../mydir`
- ❑ Remove directory
 - `rmdir <options> <directory-name>`
 - `rmdir mydir`; `rmdir /home/puri/mydir`; `rmdir ../mydir`
- ❑ Details about a command can be obtained using the "man command" – e.g., `man mkdir`

9/1/2005

9

Listing Directories and Files

- ❑ "ls" lists files and directories
- ❑ By default "ls" displays directories and files sorted in alphabetic order column-wise
- ❑ Other options can be used to display details in different formats
 - long listing: `ls -l`
 - long listing in reverse order: `ls -lr`
 - list sorted by time stamp: `ls -t`
 - List all the files including hidden files: `ls -a`
 - see man page for more options

9/1/2005

10

Simple Operations with Files I

- ❑ Creating files - `cat > myfile`
- ❑ Sorting contents of a file - `sort`
- ❑ View Files with "more"
 - search for a string → `/<string>`
 - move forward and backwards → `n` and `b`
- ❑ Other options to view files - `cat`, `head`, `tail`
- ❑ Count lines, words, characters - `wc`

9/1/2005

11

Simple Operations with Files II

- ❑ Copy files - `cp`
- ❑ Rename files - `mv`
- ❑ Remove files - `rm`
- ❑ Locate specific lines in a file - `grep`

9/1/2005

12

Department of Computer and Information Sciences UAB

Redirecting Output II

Utility: ls
Args: -l
Command: ls -l

Utility: ls
Args: -l
File: myfiles
Command: ls -l > myfiles

Utility: ls
Args: -l
Utility: wc
Args: -l
Command: ls -l | wc -l

9/1/2005 16

Department of Computer and Information Sciences UAB

Other Directives

- ❑ Input redirection - "<"
 - e.g., *pr < myfile*
- ❑ Append output to a file - ">>"
 - e.g., *cat newfile >> outfile*
- ❑ Redirect error messages - "2>"
 - e.g., *cat nofile 2> errors*
- ❑ Not display error messages
 - e.g., *cat nofile 2> /dev/null*
- ❑ Execute multiple commands - ";"
 - e.g., *clear; date*

9/1/2005 17

Department of Computer and Information Sciences UAB

Creating Files

- ❑ We have already seen creating files using
 - output redirection with the ">" directive
 - appending a file using the ">>" directive
- ❑ Using cat and ">" directive


```
$ cat > myfile
This is some text being entered into file - myfile
CTRL-D
$ cat myfile
This is some text being entered into file - myfile
$
```
- ❑ By default the keyboard is connected to the input stream of "cat" utility

9/1/2005 18

View Files

- Using "more"
 - `more <filename>`
 - SPACEBAR to move forward
 - move forward and backwards → *n* and *b*
 - search for a string → `/<string>`
- Other options to view files → *cat, head, tail*
- Combining cat (head, tail) and more
 - `cat myfile | more`
 - `tail -200 myfile | more`
- Using editors (vi, emacs, etc.)

9/1/2005

19

"wc" command

- `wc <options> filename`
- Default behavior is to print number of lines, words, and characters followed by the filename
- Count lines → `wc -l <filename>`
- Count words → `wc -w <filename>`
- Count characters → `wc -c <filename>`
- Count lines & words → `wc -lw <filename>`
- Also "wc" can be used with input redirection
 - `cat <filename> | wc -l`
 - `ls | wc`

9/1/2005

20

Copy Files

- Copy files using "cp" command
- Usage: `cp <options> <source> <target>`
- Some common options:
 - Interactive (prompt when overwriting) - *i*
 - Preserve permissions and ownership - *p*
 - Recursive copy - *r*
- Examples:
 - `cp mydoc newdoc`
 - `cp mydoc ../`
 - `cp ../newdoc .`
 - `cp -r mydir /home/puri/old-dir/`

9/1/2005

21

Rename or Move Files

- ❑ Rename/move files using "mv" command
- ❑ Usage: `mv <options> <source> <target>`
- ❑ Some common options:
 - Prompt for confirmation → *i*
 - Move without confirmation (force) → *f*
- ❑ Examples:
 - `mv -i mydoc newdoc`
 - `mv -f mydoc ../`
 - `mv ../newdoc .`
 - `mv mydir /home/puri/old-dir/`

9/1/2005

22

Remove Files

- ❑ Remove files using "rm" command
- ❑ Usage: `rm <options> <filename>`
- ❑ Some common options:
 - Prompt for confirmation → *i*
 - Move without confirmation (force) → *f*
 - Recursive remove → *r*
- ❑ Examples:
 - `rm mydoc`
 - `rm -f mydoc ../newdoc`
 - `rm -i ../newdoc`
 - `rm -r mydir`

9/1/2005

23

Recap ... (1)

- ❑ Simple commands:
 - `pwd` → identify current directory
 - `man` → help on commands
 - `ls` → list files
 - `ps` → list processes
 - `cd` → change directory
 - `cat`, `more`, `head`, `tail` → display files
 - `cp` → copy files
 - `mv` → move (also rename) files
 - `rm` → remove files

9/1/2005

24

Recap ... (2)

- Other commands:
 - history
 - clear
 - date
 - alias
- Input/Output redirection
 - > to file (rewrites the file)
 - < from file
 - >> append
- Execute more than one command
 - ;

9/1/2005

25
