

Ve 280

Programming and Elementary Data Structures

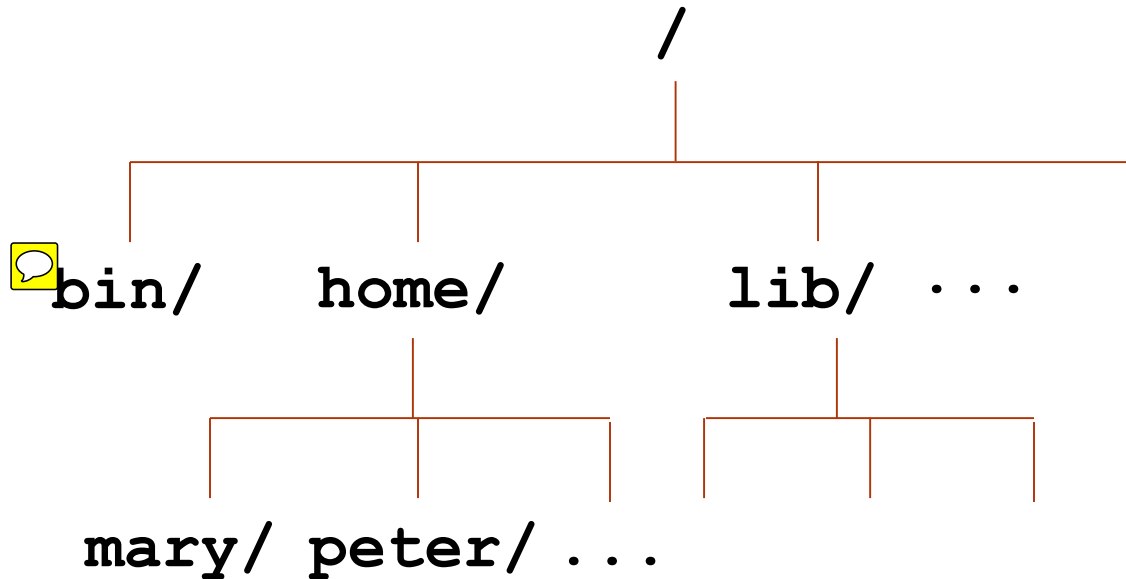
Linux

Change Directory

- Basic command: `cd pathname`
 - E.g., `cd /usr/bin`
typical path name format
- Special characters for directories
 - root directory: `/`
 - home directory: `~`
 - Linux is a multi-user operating system. It is the “home directory” of you.
 - current directory: `.`
 - parent directory: `..`

Aside: Root Directory

- Directory in Linux is organized as a tree
- The topmost directory is root directory “/”



List Contents of a Directory

- Basic command: `ls directory`
 - E.g., `ls /home`
- `ls` (i.e., “`ls`” alone): list the current working directory



Options

- `ls -l [directory]`: list in long format
- `ls -a [directory]`: list all files including the hidden files
 - Hidden files: file name begin with a dot. E.g., “.bash_history”
- In Linux, options can be combined together.
 - “`ls -la`” or “`ls -l -a`”

Aside: Long Format of File Information

- `ls -l`

group modification time

-rw-----	1	john	john	576	Apr 17 1998	weather.txt
drwxr-xr-x	6	john	john	1024	Oct 9 1999	web_page
-rw-rw-r--	1	john	john	276480	Feb 11 20:41	web_site.tar
-rwx-----	1	john	john	5743	Dec 16 1998	my_app

permission

owner

file size
(in bytes)

file name

- File permission
 - First character: '-' regular file; 'd' directory
 - Next three: read, write, execution permission of the owner
 - Next three: read, write, execution permission of the group
 - Final three: read, write, execution permission of everyone else

Manipulating Files/Directories

- Create directories: **mkdir** dir
- Delete directories: **rmdir** dir
 - Can only remove **empty** directory
- Create an empty file: **touch** file

Copy Files/Directories

- Basic command: `cp source dest`
- Variations
 - `cp file1 file2`: copy the content of file1 into file2
 - `cp file1 dir`: copy file into a directory
 - `cp file1 file2 dir`
 - `cp file* dir`
 - *: wildcard, matches any character
 - `cp -r dir1 dir2`: If dir2 does not exist, copy dir1 as dir2. If dir2 exists, copy dir1 inside dir2

Question: How do you list all the “.cpp” file in the current directory?

Rename/Move a File

- Basic command: `mv source dest`
- Variations
 - `mv file1 file2`: rename file1 as file2
 - `mv file1 dir`: move file into a directory
 - `mv dir1 dir2`: If dir2 does not exist, then rename dir1 as dir2. If dir2 exists, then move dir1 inside dir2

Delete Files/Directories

- Basic command: `rm file`
- Variations
 - `rm file`: delete file
 - `rm file1 file2`: delete file1 and file2
 - `rm -r dir`: delete dir along with its contents
- Useful options `-i`: prompt before every removal
 - To use: alias `rm='rm -i'`;
 - Put it into `~/.bashrc`

Edit/Show a File

- Edit file: `nano file` `gedit file`
 - advanced editor: vim, emacs
- Show file content
 - `cat file`
 - `less file`
 - quit 'less': press 'q'
 - go to the end: press 'G' (shift + g)
 - go to the beginning: press 'g'
 - search: press '/', then enter the thing to be searched
 - press 'n' for the next match; press 'N' for the previous match.

I/O Redirection

- Most command line programs display their results on the **standard output**.
 - By default, standard output is our display.
- We can redirect from standard output to a file by using '>'.
 - E.g., `ls -l > ls_rst.txt`: the “ls” result is now in `ls_rst.txt`

I/O Redirection

- Many commands can accept input from a facility called **standard input**.
 - By default, standard input is our keyboard.
- We can redirect standard input from a file instead of keyboard by using '<'.
 - One application: testing
 - E.g., `my_add < input.txt`
my_add is a program taking two inputs from keyboard and output their sum on screen
- Question: what does the following command mean?
 - `my_add < input.txt > output.txt`

Other Commands

- Auto completion: type a few characters; then press ‘Tab’
 - If there is a single match, Linux completes the remaining.
 - If there are multiple matches, hit the second time, Linux show the candidates.
- Compare two files: `diff file1 file2`
 - If files are the same, no output
 - If there are differences: lines after “<” are from the first file; lines after “>” are from the second file
 - In a summary line: ‘c’: change; ‘a’: add; ‘d’: delete
 - Useful option “-w”: ignore white spaces (space, tab)

Other Commands

- Install a program: `sudo apt-get install program`
 - E.g., `sudo apt-get install emacs`
 - `sudo command`: execute command as a superuser
 - Need you to type your password
- Remove a program: `sudo apt-get autoremove program`
- Looking for help? `man command` E.g., `man ls`
 - Browse the manual using the same command as for 'less'

Reference

- <http://linuxcommand.org/>