CSCI 211 UNIX Lab

Basic Unix Commands (2)

Dr. Jiang Li



Jiang Li, Ph.D. Department of Computer Science

Today's Focus

- Create, remove, copy, move files and directories
 - -mkdir, rmdir, cp, mv, rm
- The 'cat', 'more' and 'echo' command
- Output redirection (>, >>)
- Pipe (|)



Make/Remove Directories

- mkdir
 - MaKe DIRectory (create a new directory)
 - Syntax: mkdir <directories>
 - Example: mkdir directory1 directory2
- rmdir
 - ReMove DIRectory, which must be empty directory at first
 - Syntax: rmdir <directories>



Copy/Move Files

- cp
 - CoPies a file, preserving the original file
 - Syntax:cp <flags> <sources> <destination>
 - Example: cp tutorial.txt tutorial.txt.bak
 - Copy directory (-r):
 - cp -r sourcedir destdir/
- mv
 - MoVes or renames a file, destroying the original file
 - Syntax:mv <flags> <sources> <destination>
 - Examples:
 - mv tutorial.txt tutorial.txt.bak (rename)
 - mv tutorial.txt tutorial-slides.ppt backups/

Department of Computer Science



Remove Files/Directories

• Remove files

- ReMoves a file, without a possibility of "undelete!"
- Syntax:rm <flags> <file(s)>
- Example: rm file1.txt backups/file2.txt
- -f flag: forces the system to remove without prompting

Note: All of these commands over-write or remove existing files without warning you! Use the –i flag.

Remove directories

- Syntax: rm -r <directories>
- Example: rm -rf backups/



Wildcard Character – *

• Asterisk sign ' * ' is a special character

 It means match any characters with zero to multiple times

- Example:
 - rm file*
 - Remove all files with the name begin with 'file'
 - rm *xyz
 - Remove all files with the name end with 'xyz'
 - rm *
 - Remove all files in current directory



The cat and more Command

• cat and more command show the content of a file

cat [option] [file]...
more [option] [file] ...

- more shows the content screen by screen
 - Useful when you read a large file
 - Press the space bar to switch to next screen
 - Press 'q' to quit
- echo display a line of text or an environment variable
 - echo hello: this command displays the string hello
 - echo \$PATH: this command displays the content of PATH env variable



Output Redirection (1)

- Redirect the standard output (stdout) to a file instead of the screen
- Symbols: > and >>
 - -> overwrites

(output: the old data is replaced by the new data)

->> appends

(output: the new data is appended at the end of the old data)

• Examples

cat file1 > file2

Redirect the content of file1 to file2

echo 'Hello World!' >> file2

— Redirect the output string to file file2 (with append mode)

Output Redirection (2)

- Redirect the standard error (stderr) to a file instead of the screen
- Symbols: 2> and 2>>
 - 2> overwrites

(output: the old data is replaced by the new data)

- 2>> appends

(output: the new data is appended at the end of the old data)

- Examples:
 - x 2> t
 - x 2>> t



Output Redirection (3)

- Redirect both stdout and stderr to a file instead of the screen
- Syntax:

command > file-name 2>&1

• Example:

find /usr/home -name .profile 2>&1



Pipe

- A pipe connects the standard output of the previous command directly with the standard input of the later command.
- The symbol for a **pipe** is a vertical bar ().
- Examples

cat file | more ls -l | more

