Capgemini

UNIX

Lesson 03 : Basic UNIX commands

Lesson Objectives

- In this lesson, you will learn:
  - Basic UNIX Commands

3.1: Commands
man Command

man command:

- The on line help provided by the **man** command includes brief description, options, and examples.
- Example:

```
$man <command>
```

cat command

cat :concatenate files & print on the standard output.
cat file_name
The above command is used to display the contents of the specified file.
cat file1 > file2
The above command is used to copy the contents of file1 to file2.

cat command (Continued)

cat file1 >> file2

The above command is used to append the contents of file2 from file1.

cat file1 file2

The cat command can be used to vertically concatenate the contents of more than 1 file

cat -b filename - *shows line numbers*

cat file1 file2 > file3 - *adds file1 and file2 to make file3*

Directory command

ls

The ls command is used to display the names of the files & sub directories in a directory.

```
$ ls -l
total 2
-rw-r--r--    1 gilberg   staff      12 May 17 08:45 f-t
-rw-r--r--    1 gilberg   staff      12 May 17 08:45 f1t
```

File Type | Permissions | Links | Owner | Group | File Size | Last Mod | Filename

File and Directory Commands

- 'ls'
  - ls - *show contents of working directory*
  - ls file - *list file, if it exists in working directory*
  - ls dir - *show contents of the directory dir*
  - ls -a - *shows all your files, including hidden ones*
  - ls -al - *give detailed listing of contents*
  - ls -F - *mark directories with "/" and executable files with "*"*
  - ls *.doc - *show all files with suffix ".doc"*
  - ls -lt - *Time of last modification.*
  - ls -lut - *Time of last access*

mkdir Command

The mkdir command creates a directory.

- Example: 1:

```
$ mkdir doc
```

- Example 2:

```
$ mkdir doc    doc/example        doc/data
```

- Example 3:

```
$ mkdir        doc/example        doc
```

  - It will give error - Order important.

cp command

copies files. Will overwrite unless otherwise specified. Must also have write permission in the destination directory.

Syntax:

cp [options] <sourcefile/s] <destination directory/file>

Examples:

cp file1 file2

cp command

Options	Function
-i	Prompts before overwriting
-r	Copies the entire directory

Example:
cp –r dir1 dir2
The above command copies the dir1 directory and all its files and sub directories to the dir2 directory. If the directory dir2 exists all the contents are put inside that directory, otherwise dir2 is created in the current working directory.

cp (Continued)

- Examples:
  - cp sample.f sample2.f - *copies sample.f to sample2.f*
  - cp -R dir1 dir2 - *copies contents of directory dir1 to dir2*
  - cp -i file.1 file.new - *prompts if file.new will be overwritten*
  - cp *.txt chapt1 - *copies all files with .txt suffix to directory chapt1*
  - cp /usr/doc/README ~ - *copies file to your home directory*
  - cp ~betty/index . - *copies the file "index" from user betty's home directory to current directory*

    cp f1 f2 f3 f4

    cp f1 dir

mv command

- mv
  - moves files. Will overwrite unless otherwise specified. Must also have write permission in the destination directory.
  - Example:
    - mv sample.f sample2.f - *moves sample.f to sample2.f*
    - mv dir1 newdir/dir2 - *moves contents of directory dir1 to newdir/dir2*
    - mv -i file.1 file.new - *prompts if file.new will be overwritten*
    - mv *.txt chapt1 - *moves all files with .txt suffix to directory chapt1*

rm command

- rm
- deletes/removes files or directories if file permissions permit.
- Example
  - rm sample.f - *deletes sample.f*
  - rm chap?.txt - *deletes all files with chap as the first four characters of their name and with .txt as the last four characters of their name*
  - rm -i * - *deletes all files in current directory but asks first for each file*
  - rm -r / olddir - *recursively removes all files in the directory olddir, including the directory*

The –r option is used to remove a directory along with its sub directories. This option is sometimes preferred over the rmdir command since in the case of rmdir, you can only delete an empty directory.

rmdir Command

Example 1:

```
$ rmdir doc
```

Example 2:

```
$ rmdir doc/example doc
```

Example 3:

```
$ rmdir doc  doc/example
```

- It will give error.

cal

cal month year
Example: cal 9 2003
It will display calendar of September 2003.
cal year
cal 2008
It will display calendar of entire year 2008.
cal
It will display the calendar of current month.

date Command

date command:

- The **date** command is used to see current date and time.
- Date can be displayed in different formats
- Example:

```
$ date
```

- **Output:** Fri Apr 6 11:14:46 IST 2001

```
$ date “+%T”          -- %t is used to display only time
```

- **Output:** 11:15:20

```
$ date "+ %d %h“      -- To display date and month name
```

- **Output:** 6 Apr

uptime

It prints the amount of time the system has been “up”—the amount of time from the last unix boot.

uptime also gives the current time and the load average. The load average is the average number of jobs waiting to run in a certain time period.

$ uptime

07:40:49 up 7 days, 21:29, 100 users, load average: 0.21, 0.61, 0.85

Few More Commands

Locate a command.

which [filename] ...

Filename Name of the file you are looking for.

which perl

Would locate the executable location of the perl command.

whereis

Locate a binary, source, and manual page files for a command.

whereis [-bmsu] [-BMS directory... -f] filename ...

-b Search only for binaries.

-m Search only for manual sections.

-s Search only for sources.

alias & unalias

Defined a new name for a command

$ alias

- with no arguments lists currently active aliases

$ alias newcommand oldcommand

- defines a newcommand

$ alias cl cal 2003

$ cl

Unalias Removes alias

Requires an argument.

$ unalias cl

uname

Print name of current system.

uname [-a] [-i] [-m] [-n] [-p] [-r] [-s] [-v] [-X] [-S systemname]

-a Print basic information currently available from the system.

-I Print the name of the hardware implementation (platform).

-m Print the machine hardware name (class). Use of this option is discouraged; use uname -p instead.

-n Print the nodename (the nodename is the name by which the system is known to a communications network).

-p Print the current host's ISA or processor type.

-r Print the operating system release level.

-s Print the name of the operating system. This is the default.

-v Print the operating system version.

-X Print expanded system information, one information element per line, as expected by SCO Unix. The displayed information includes: system name, node, release, version, machine, and number of CPUs.

BusType, Serial, and Users (set to "unknown" in Solaris)

OEM# and Origin# (set to 0 and 1, respectively)

lp Command

lp command:

- The **lp** command is used for printing files.
  - Example:

```
$lp myfile.txt
```

```
$lp –n 10 myfile.txt
```

```
$lpq
```

```
$lprm -Pps99  11042
```

nl Command

nl command:

- The **nl** command is used to print file contents along with line numbers.
- Options:
  - -w : width of the number
  - -v : Indicate first line number
  - -i : increment line number by

- Example:

```
$ nl myfile.txt
1  line one
2  line two
```

tty Command

tty Command:

- Unix treats a terminal also as a file. In order to display the device name of a terminal, the **tty** (teletype) command is used.
- print the file name of the terminal connected to standard input
- Example: Using tty command

```
$ tty
  /dev/ttyp3
```

2.4: File Related Commands
pwd Command

The pwd command checks current directory.

```
$ pwd
```

- **Output:** /usr/Kumar

cd Command

The cd command changes directories to specified directory

The directory name can be specified by using absolute path (Full Path) or relative path

```
$ pwd
```

- **Output:** /usr/kumar

```
$ cd Prog
$ pwd
```

- **Output:** /usr/kumar/Prog

cd Command

- Moving one level up:

```
$ cd ..
```

- Switching to home directory:

```
$ cd
```

- Switching to /usr/sharma:

```
$ cd /usr/Sharma
```

- Switching to root directory:

```
$ cd /
```

logname Command

The logname command checks the login directory.

```
$ logname
```

Output: Kumar

cmp Command

cmp Command:

```
$ cmp file1.txt file2.txt
file1.txt file2.txt differ: char 41, line 2
$ cmp file1.txt file1.txt
```

comm Command

comm Command:

- The comm command compares two sorted files. It gives a 3 columnar output:
  - First column contains lines unique to the first file.
  - Second column contains lines unique to the second file.
  - Third column displays the common lines.

comm Command

```
$ cat cfile1.lst
A
G
K
X
```

```
$ cat cfile2.lst
A
F
K
W
X
Z
```

```
$ comm cfile1.lst cfile2.lst
                A
        F
G
                K
        W
                X
        Z
```

```
$ comm -12 cfile1.lst cfile2.lst
A
K
X
```

diff Command

The diff command is used to display the file differences. It tells the lines of one file that need to be changed to make the two files identical.

- Example:

```
$ diff cfile1.lst cfile2.lst
2c2
< G
> F
3a4
> W
4a6
> Z
```

tr Command

The tr command accepts i/p from standard input.
This command takes two arguments which specify two character sets.
The first character set is replaced by the equivalent member in the second character set.
The –s option is used to squeeze several occurrences of a character to one character.

tr Command

Example 1: To squeeze number of spaces by single space:

```
$ tr –s “ ”  < file1.txt
```

Example 2: To convert small case into capital case:

```
$ tr “[a-z]” “[A-Z]” < file1.txt
ONE
TWO
THREE
FOUR
```

more Command

The more command, from the University of California, Berkeley, is a paging tool.

The more command is used to view one page at a time. It is particularly useful for viewing large files.

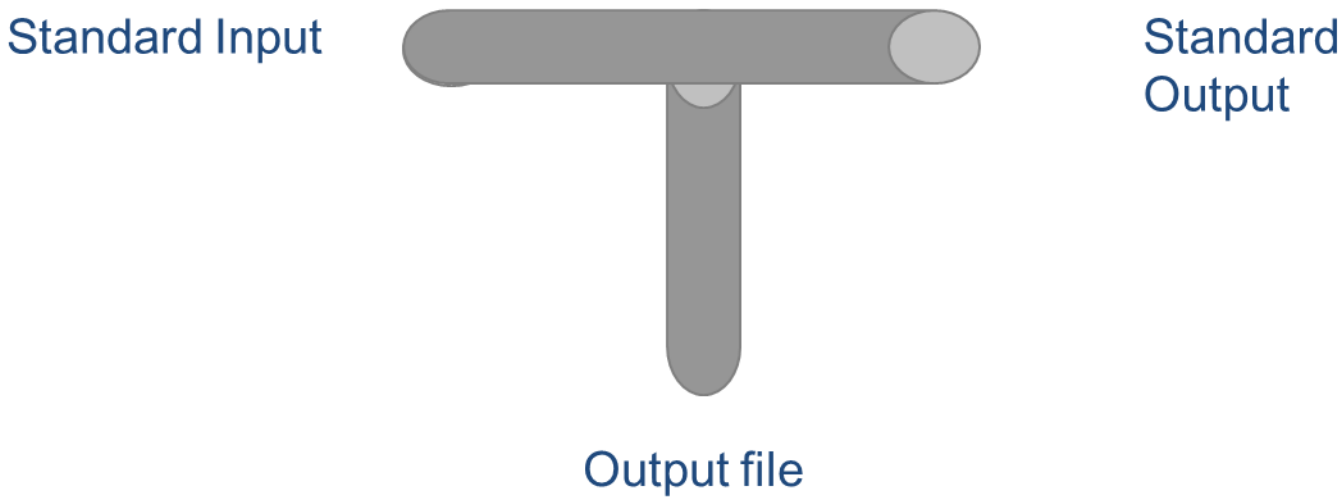
Syntax for more command is as follows:

```
more <options> <+linenumber> <+/pattern> <filename(s)>
```

Example: To display file1.txt one screenful at a time

```
$ more file1.txt
```

tee Command



- To display contents of file employee on screen as well as save it in the file:

```
$ tee user.txt < employee
```

SUMMARY

- In this lesson, you have learnt:
  - Commands to create files and directories
  - Commands to delete files and directories
  - Browsing file
  - Comparing files

Review Questions

- Question 1: ___ command to display directory listing on screen as well as store it in dirlist.lst.
- Question 2: ___command is used to read a directory?
- Question 3: Display date in format of 11th July 16?