Basic Linux Commands

by
B.Prathibha
prathibhab@cdac.in

15th September,2010

Centre for Development of Advanced Computing,Chennai

cal - Command to see calender for any specific month or a complete year

usage: cal [month] [year]

eg: cal sep 2010

date –prints the date and time

date +"%D %H: %M: %S"

09/14/10 14: 19: 43

Options:

d - The date of the month (1-31)

y - The last two digits of the year

H,M,S - Hour Minute and second respectively

D - the date in mm/dd/yy

echo: Print message on the terminal

usage: echo “<message>”

eg: echo "Welcome to the workshop"

o/p: Welcome to the workshop

passwd - allows you to change your password

man displays the documentation for a command

usage: man <command name>

eg: man mkdir

Linux File System

Standard directory structure

/ - the topmost

/dev - all the devices are accessible as files

/var - “variable” data such as mails, log files, databases

/usr - almost all the packages installed

/etc - configuration files

/home - home directories for all the users

/root - home directory of the privileged user root

/mnt - used to mount other directories/partitions.

- Metacharacters -These are special characters that are recognised by the shell.
- * - matches 0 or more characters.

  eg: ls *.c
- ? - matches any single character

  eg: ls ab?.c
- [] - This will match any single character in the range.

  eg: ls tut[0-9].m

  This will find files such as tut0.m, tut9.m etc.,
- > - Redirect standard output to a file.

  echo “hello world” > hello.txt

- >> -Appends standard output to a file.

  eg: echo “Hello Again” >> hello.txt

- < - Takes standard input from a file
- | - This is pipe character. Sends the output of first command as input for the second command.

- mkdir – make directory

  usage: mkdir <dirname>

  eg: mkdir -p path/test/test1

  -p --> no error if existing, make parent directories as needed

- cd - change directories

  Use cd to change directories. Type cd followed by the name of a directory to access that directory.

- mv- change the name of a directory

  Type mv followed by the current name of a directory and the new name of the directory.

  Ex: mv testdir newnamedir

- cp - copy files and directories

  usage: cp source destination

  cp -i myfile yourfile

  With the "-i" option, if the file "yourfile" exists, you will be prompted before it is overwritten.

  cp -r srcdir destdir

  Copy all files from the directory "srcdir" to the directory "destdir" recursively.

- rmdir - Remove an existing directory
- rm - remove files or directories

  Usage: rm -r name

  Removes directories and files within the directories recursively.

File Handling Commands

- cat - used to display the contents of a small file on terminal

  usage: cat <file name>

  cat when supplied with more than one file will concatenate the files without any header information

- more and less commands are used to view large files one page at a time

  usage: more <file name>

  usage: less <file name>

- wc command is used to count lines, words and characters, depending on the option used.

  usage: wc [options] [file name]

  You can just print number of lines, number of words or number of charcters by using following options:

  -l : Number of lines

  -w : Number of words

  -c : Number of characters

Simple Filters

- Filters are commands which accept data from standard input, manupulate it and write the results to standard output.
- Head - displays the lines at the top of the file

  - when used without any option it will display first 10 lines of the file

  usage: head filename

  -n --> print the first N lines instead of the first 10
- tail - displays the lines at the end of the file. By default it will display last 10 lines of the file

  usage: tail filename

- cut command can be used to cut the columns from a file with -c option.

  eg: cut -c 1,3-5 /etc/passwd

- With -f option you can cut the feilds delemited by some character

  eg: cut -d':' -f2 /etc/passwd

  -d option is used to specify the delimiter and -f option used to specify the feild number

- paste command will paste the contents of the file side by side

  eg: paste a.txt b.txt

- sort - re-orders lines in the file

  whitespaces first, then numerals, uppercase and finally lowercase

  you can sort the file based on a field by using -t and -k option.

  Eg: sort -t" " -k 2 students.txt

  sorts the file based on the second field using the delimiter as space

  -r --> reverses the result

- grep - scans its input for a pattern, displays the line containing that pattern

  usage: grep options pattern filename(s)

- searching for a text string in one

  grep 'boss' /etc/passwd

  searches for the pattern boss in the /etc/passwd file

- searching for a text string in multiple files

  grep ‘root’ *.txt

- Case-insensitive file searching with the Unix grep command

  grep -i ‘hello’ hello.txt

- Reversing the meaning of a grep search

  grep -v ‘boss’ /etc/passwd

  Displays all the lines that do not contain the specified pattern

- Using grep in a Unix/Linux command pipeline

  ls -al | grep ‘^d’

  print the lines that starts with d

- Linux grep command to search for multiple patterns at one time

  egrep ‘boss|root’ /etc/passwd

- grep pattern matching and regular expressions (regex patterns)

  grep '[FG]oo' *

  grep '[0-9][0-9][0-9]' *

  grep '^fred' /etc/passwd

- sed - editor for filtering and transforming text

  -i --> edit the files in place

- sed -i '1,10d' hello.txt

  deleted the first 10 lines from hello.txt

- sed -i ‘‘2’i hai’ hello.txt

  Inserts the text ‘hai’ in the second line

- sed -i '/hello/d' hello.txt

  Deleted the line containing the pattern hello.

- sed 's/hello/world/' hello.txt

  Replaces the first occurrence of hello on each line to world.

- sed 's/hello/world/g' hello.txt

  Replaces all the occurrences of hello on each line to world.

- pwd - print working directory

  will show you the full path to the directory you are currently in.

- shred - overwrite a file to hide its contents

  The result is that your file is so thoroughly deleted it is very unlikely to ever be retrieved again.

- ln -s test symlink

  Creates a symbolic link named symlink that points to the file test

- free - Displays the amount of used and free system memory.

w – show who is logged on and what they are doing

usage: w

who -show who is logged in

usage: who

who -b --> last system boot time

whoami – prints the effective user id.

whereis ls - Locates binaries and manual pages for the ls command.

cat – displays the contents of the file on the screen.

- df – report file system disk space usage

  Usage: df -h

  -h --> print sizes in human readable format

- du - summarize disk usage of each file, recursively for directories.

  Usage: du -h

- find - Find locations of files/directories quickly across entire filesystem

  Usage: find / -name appname -type d -xdev

  - type d - search for the directory named appname

  -xdev Don't descend directories on other filesystems.

  - search against all directories below / for the appname found in directories but only on the existing filesystem.

- Command to find and remove files

  find . -name "FILE-TO-FIND"-exec rm -rf { } \;

- lspci - a utility for displaying information about PCI buses in the system and devices connected to them.

  -v – displays a detailed information.

- lsusb – a utility for displaying information about USB buses in the system and the devices connected to them.

  -v – displays a detailed information.

- lshw - list the hardware
- hwinfo – probs for the hardware
- cat /proc/cpuinfo – gives information about cpu
- cat /proc/meminfo - gives information about memory

- ps (i.e., process status) command is used to provide information about the currently running processes, including their process identification numbers (PIDs).

  ps – lists all the processes

  usage: ps -aux

- kill – to kill a process

  ps is most often used to obtain the PID of a malfunctioning process in order to terminate it with the kill command

  usage: kill -9 pid

  where pid – process id of the process to be killed.

- Cron is the name of program that enables linux users to execute commands or scripts (groups of commands) automatically at a specified time/date.
- You can setup setup commands or scripts, which will repeatedly run at a set time.
- The cron service (daemon) runs in the background and constantly checks the /etc/crontab file, /etc/cron.*/ directories.
- It also checks the /var/spool/cron/ directory.
- To edit the crontab file, type the following command at the Linux shell prompt:

  crontab -e

- Syntax of crontab (Field Description)

  m h dom mon dow /path/to/command arg1 arg2

  where

m: Minute (0-59)

h: Hours (0-23)

dom: Date (0-31)

mon: Month (0-12 [12 == December])

dow: week days(0-7 [0 or 7 sunday])

/path/to/command - Script or command name to schedule

If you wished to have a script named /root/backup.sh run every day at 3am, your crontab entry would look like as follows:

```
0 3  * * * /root/backup.sh
```

Execute every minute

```
* *  * * * /bin/script.sh
```

This script is being executed every minute.

Execute every Friday 1AM

To schedule the script to run at 1AM every Friday, we would need the following cronjob:

```
0 1 * * 5 /bin/execute/this/script.sh
```

The script is now being executed when the system clock hits:

1. minute: 0
2. of hour: 1
3. of day of month: * (every day of month)
4. of month: * (every month)
5. and weekday: 5 (=Friday)

Execute on workdays 1AM

To schedule the script to run from Monday to Friday at 1 AM, we would need the following cronjob:

0 1 * * 1-5 /bin/script.sh

The script is now being executed when the system clock hits:

1. minute: 0
2. of hour: 1
3. of day of month: * (every day of month)
4. of month: * (every month)
5. and weekday: 1-5 (=Monday till Friday)

Execute 10 past after every hour on the 1st of every month

10 * 1 * * /bin/script.sh

if you want to run something every 10 minutes:

0,10,20,30,40,50 * * * * /bin/script.sh

or

*/10 * * * * /bin/script.sh

Special Words

If you use the first (minute) field, you can also put in a keyword instead of a number:

@reboot Run once, at startup

@yearly Run once a year "0 0 1 1 *"

@annually (same as @yearly)

@monthly Run once a month "0 0 1 * *"

@weekly Run once a week "0 0 * * 0"

@daily Run once a day "0 0 * * *"

@midnight (same as @daily)

@hourly Run once an hour "0 * * * *

Eg: @daily /bin/script.sh

Storing the crontab output

To store the output in a separate logfile. Here's how:

*/10 * * * * /bin/script.sh 2>&1 >> /var/log/script_output.log

- Starting vi

  You may use vi to open an already existing file by typing

  vi filename

- vi Modes

  vi has two modes:

  * command mode

  * insert mode

  In command mode, the letters of the keyboard perform editing functions (like moving the cursor, deleting text, etc.). To enter command mode, press the escape <Esc> key.

  In insert mode, the letters you type form words and sentences. Unlike many word processors, vi starts up in command mode.

- Entering Text

  In order to begin entering text in this empty file, you must change from command mode to insert mode. To do this, type

  I

- Deleting Words

  To delete a word, move the cursor to the first letter of the word, and type

  dw

  This command deletes the word and the space following it.

  To delete three words type

  3dw

- Deleting Lines

To delete a whole line, type

dd

The cursor does not have to be at the beginning of the line. Typing dd deletes the entire line containing the cursor and places the cursor at the start of the next line. To delete two lines, type

2dd

To delete from the cursor position to the end of the line, type

D (uppercase)

- Replacing Characters

  To replace one character with another:

  1. Move the cursor to the character to be replaced.

  2. Type r

  3. Type the replacement character.

  The new character will appear, and you will still be in command mode.

- Replacing Words

  To replace one word with another, move to the start of the incorrect word and type

  cw

You are now in insert mode and may type the replacement. The new text does not need to be the same length as the original. Press <Esc> to get back to command mode. To replace three words, type

3cw

- Replacing Lines

To change text from the cursor position to the end of the line:

1. Type C (uppercase).
2. Type the replacement text.
3. Press <Esc>.

Moving around in a file

H	to top line of screen
M	to middle line of screen
L	to last line of screen
G	to last line of file
1G	to first line of file

- Moving by Searching

To move quickly by searching for text, while in command mode:

1. Type / (slash).
2. Enter the text to search for.
3. Press <Return>.

The cursor moves to the first occurrence of that text.

To repeat the search in a forward direction, type

n

To repeat the search in a backward direction, type

N

To save the edits you have made, but leave vi running and your file open:

1. Press <Esc>.
2. Type :w
3. Press <Return>.

To quit vi, and discard any changes your have made since last saving:

1. Press <Esc>.
2. Type :q!
3. Press <Return>.

Thank You