**Linux Operating System Commands**

**What is Shell**

The shell is a program that takes keyboard commands and passes them to the operating system to carry out action.

Almost all Linux distributions supply a shell program from the GNU Project called bash.

The name “bash” is an acronym for “Bourne Again Shell”

**Terminal**

Terminal is used to access the Shell to enter the commands

When we first log in to our system our current working directory is set to our home directory.

After terminal lunch, we should see prompt like below

username@computer\_name:pwd$

it will usually include your username@machinename, followed by the current working directory and a dollar sign if normal user or # sign for root user.

after the $ sign we will enter Linux commands.

echo $PS1

ps1\_old =”$PS1”

$PS1 = new value look for more option at page no. 181

**Linux Hierarchy**

/ The root directory, where everything begins

/bin contain programs which is required for system to boot

/boot contain Linux Kernel

/dev contain devices

/etc contain system wise configuration

/home contain user home directory

/lib contain shared library

/list+found lost files

/media contain mount point for USB, CD etc

/mnt same like media but used in older version of OS

/opt contain optional installed software.

/proc contain system related files

/root contain home directory for root user

/sbin contain program which perform vital tasks. Root user have access only

/tmp contain temporary files

/usr contain program used by all users

/var contain variable type of data file

/var/log contain different logs files

**Commands History**

we can recall entered commands by pressing up & down arrows keys from keyboard

or use history command to show previous entered commands. To execute the previous command from history, issue command **!number**

**Simple Commands**

who i am find about login user information

date this will show the date

cal this will show current month calendar

hostname show system name

uname Operating System Name

uname -a Operating System name and other detail

id show information about login

**Hard Disk Related Commands**

df -h this will show disk information

free this will show RAM information

**Navigation / File System commands**

pwd print working directory (current location)

cd change directory. only cd command change to

cd .. or cd - both command have same result, takes to one level up

cd ~ change to home directory

cd ~username change to home directory of mentioned user

chmod change permission on file or directory

mkdir create directory or Folder

Absolute Path start from root directory up to target directory

Relative Path two special symbols (.) and (..) are used. The (.)

is used for current working directory where (..) is refer to one level up from current directory

ls list directory contents

clear (Ctrl + l ) clear the screen

history show all previous entered commands

Ctrl + a beginning of line

Ctrl + e end of line

Ctrl + f forward one character

Ctrl + b backward one character

**Symbolic Links**

Symbolic links allow use to give multiple name to files. There two type of links

Soft links

Hard Links

ln create hard and symbolic links

ln file link this will create hard link

ln –s file link this will create soft link

Note: hard link has two major limitations:

It cannot create link with file lying in different partition

It cannot create link with directory

**Working With Files & Directories**

file file\_name determine file type. for example, command file

file\_name will show type of file mean text file,

picture etc.

less view file contents. read text file. use **/word** to

search , n from keyboard for find next ,

q to quit

space to page down

/word to search word

n to search next occurrence of work

G to move to end of file

g to move to beginning of file

h to show help on screen

cp copy files only

cp -r copy directory

mv move / rename files and directories

mkdir create directory

rmdir remove directories

rm remove files only

rm -rf remove files and directories

stat show file or directory statistic

**Wildcards**

\* Matches any Characters

? Matches single characters

[characters] matches any ch member of set

[!characters] matches any ch not member of set

[[:class:]] matches any ch member of specified class

[:alnum:] match any alphanumeric characters

[:alpha:] match any alphabetic characters

[:digit:] match an digit

[:lower:] match any lowercase letters

[:upper:] match any uppercase letters

**Reading / handling Text Files**

cat concatenate files, same like less command. Used to concate multiple files.

sort sort text lines

uniq omit repeated lines

grep print line matching pattern ( ls –l /usr/bin | grep zip )

wc word count

head output first part of file ( head –n 5 text\_file.txt )

tail output last part of file ( tail –n 5 text\_file.txt)

tee read from standard input and write to standard output

> used for redirection e.g. ( ls –l > ls\_output.txt)

> file\_name make the file empty, means delete the content. Or if file not exist, create new.

Commands &> output.txt this command record both output and error both

ls -l /user/bin 2> /dev/null /dev/null is called bit bucket in Linux. Mean trash.

echo display text message on screen (echo “hello”) but if pass \* to echo like echo \* will

display all directories

echo \* display all directories

echo d\* display all directories staring with d

echo [[:upper:]]\* display all directories start with upper characters

echo ~ show home directory for user

printenv | less show all environment variable for user

**Searching in Linux**

history | grep “tail” this command will search tail in history

**Getting Help for commands**

type show the type of commands (type ls , type cp )

which show the location of commands

man show help about command

apropos display appropriate command

info show help about command

whatis show help about command

help show help regarding command (help cp)

--help also give help regarding command ( mkdir –help)

alias used to create our own command (alias p=’pwd’ ) or (unalias p ) for deleting

info coreutils help for core utilities

**Permission**

id display user identity ( id command will show user info, group etc)

chmod change a file mode

umask set the default file permissions

su run shell as different user

sudo excuate command as different user

chown change file owner

chgrp change file group membership

passwd change user password (passwd username )

use ls –l command to see the directory listing

**Output**

**Type Permission user group size date & time file / directory name**

d rwx rwx r-x 2 vagrant vagran 4096 2008-01-10 16:40 dir1

owner grp world





**r=4 w=2 r=1**

**first digit is equal to 1 , 2nd digit is equal to 2 and 3rd is equal to 4**

**0 0 0 = 0 (0 + 0 + 0)**

**0 0 1 = 1 (0 + 0 + 1)**

**0 1 0 = 2 (0 + 2 + 0)**

**0 1 1 = 3 (0 + 2 + 1)**

**1 0 0 = 4 (0 + 2 + 1)**

**1 0 1 = 5 (0 + 2 + 1)**

**1 1 0 = 6 (4 + 2 + 0)**

**1 1 1 = 7 (4 + 2 + 1)**

**when user owns a file or directory, user have full control over it’s access.**

**User accounts are define in /etc/passwd file**

**User groups are define in /etc/group file**

**User passwords are define in /etc/shadow file**

**Only file owner or root user can change file mode**

To assign permission to file issue below commands

chmod 644 filename

Changing Identities

su switch user

sudo running command with root user

Changing file owner and Group

chown this command is used to change the owner or group of file or directory

sudo chown new\_owner: file name or path to file

**Processes**

ps report a snapshot of current process ( command ps , ps x

top display task

jobs display active jobs

bg place a job in background

fg place a job in foreground

kill send a signal to a process

killall kill process by name

shutdown shutdown or reboot system

**The Environment Setup**

printenv print all environment ( this command is used to see all environemt)

set set shell option ( same as above )

export Export Environment

alias create new commands based on existing commands

There are two type of variables

Environment variables

Shell Variables values are set by bash

Below are the files, mainly used in environment setup

/etc/profile/

~/.bash\_profile

~/bash\_login

~/.profle

~/.bashrc

/etc/bash\_bashrc

**VIM Text Editor**

**Starting vim**

To start vim , enter vi command and to exit enter :q command

**Entering text**

To insert text , press I from keyboard

To insert line above line O and for below lowercase o

**Save File**

To save text , press :w from keyboard

**Delete Line**

To delete the current line press dd from keyboard

d$ from current location to end of line

d0 from current location to beginging of line

dG from current location to end of file.

**Undo**

for undo, command

**copy & Paste**

press dd from keyboard to cut and p to paste the line

press y is used for copy and p is used for paste is well

**Searching**

f is used for searching single word

/phrase is used for searching

press n for next occurance

**Replacing**

:%s /searchingword/replacebyword/g g mean globally

:%s /searchingword/replacebyword/gc g mean globally , c will ask for confirmation

Moving in multiple open files

:n to move to next time

:N to move back to first file

:buffers will show all open files

:e filename this command will open another file for editing

:buffer number will open file with buffer number

**Packages**

To install high level software in Ubuntu we use apt-get install software\_name ( apt-get remove name)

To install low level software in Ubuntu we use dpkg --install software\_name

To install high level software in Redhat we use yum install software\_name (yum erase name)

To install low level software in Redhat we use rpm -i software\_name

**Storage Media**

We can handle all type of storage in Linux, like

Physical storage (hard disks)

Network storage

Virtual Storage (RAID)

Logical Volume Manager (LVM)

mount mount file system ( This command will show mount points )

umount unmount file system

fsck check and repair file system

fdisk partition table

mkfs create file system

fdformat format floppy disk

dd write block level data directly to device

genisoimage (mkisofs) create ISO 9660 image file

wodim (cdrecord) write data to optical storage media

md5sum calculate an md5 checksum

Managing Storage

1st step to manage storage is to attach storage device to file system tree. This process is call mount.

file fstab located at /etc/fstab shows all devices ( harddisk partitions)





Device Name mount point FS Opt frequency Order

mount to show all mount points

umount /dev/hdc to unmount the CD-ROM

mount /dev/hdc /mnt/cdrom

above command will mount CD-ROM on cdrom directory created by me.

Create new file system / formatting the drive

Connect USB drive to system, then run following command

fdisk

after finishing partition using fdisk , then use mkfs command to create file system

sudo mkfs –t ext3 /dev/sdb1

**Networking**

We can build all types of networking system using Linux, like

Firewalls

routers

name servers

NAS (Network Attached Storage)

etc etc

ping for connectivity checking

traceroute print route

netstat print network info

ftp file transfer protocol

wget non-interactive downloader

ssh for connection

**Creating Play Ground**

create a directory with name playground in home directory.

mkdir playground

create two more directories dir1 and dir2 inside playground

cd playground

mkdir dir1 dir2

copy passwd file from etc directory to playground directory

cp /etc/passwd .

cp -i /etc/passwd . (copy file and give warring if file exist )

cp -v /etc/passwd . (show messages on screen)

rename the file passwd to fun

mv passwd fun

now move fun file to other directory

mv fun dir1

mv fun dir2

create hard links

ln fun fun-hard

ln fun dir1/fun-hard

ln fun dir2/fun-hard

create soft link

ln –s fun fun-sym

ln –s ../fun dir1/fun-sym

delete hard linklink

rm fun-hard

use ls –l command to see the directory listing

**Output**

**Type Permission user group size date & time file / directory name**

d rwxrwxr-x 2 vagrant vagrant 4096 2008-01-10 16:40 dir1