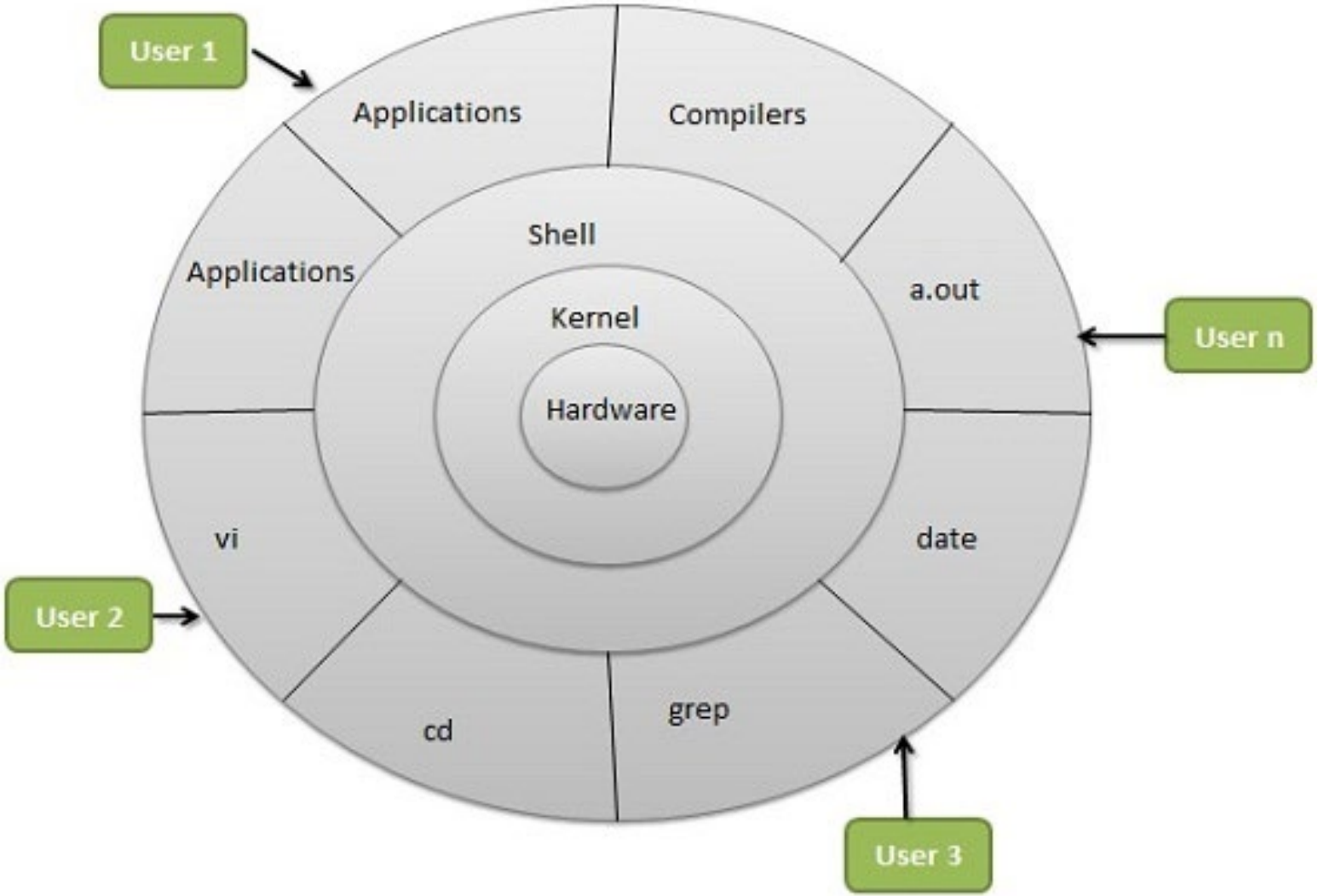
Linux Commands for Beginners

Supplement Note

Linux Operation System

Linux is one of popular version of UNIX operating System. It is open source as its source code is freely available. It is free to use. Linux was designed considering UNIX compatibility. It's functionality list is quite similar to that of UNIX. Linux began in 1991 with the commencement of a personal project by Finnish student Linus Torvalds to create a new free operating system kernel. Since then, the resulting Linux kernel has been marked by constant growth throughout its history.

Linux OS Architecture



System Info Commands

man *command* – show the manual for *command*

date – show the current date and time

uptime – show current uptime

w – display who is online

who - show who is logged on

whoami – who you are logged in as

uname – show kernel information

uptime – Show how long the system has been running + load

hostname – system's host name

hostname -i – Display the IP address of the host. (Linux only)

System Info Commands

dmesg – Detected hardware and boot messages

cat /proc/cpuinfo – cpu information

cat /proc/meminfo – memory information

df – show disk usage

du – show directory space usage

whereis *app* – show possible locations of *app*

which *app* – show which *app* will be run by default

last – show last logins on the system

history – display the command history list

UNIX File System

UNIX treats everything as a file... Directories and devices like the hard disk, DVD-ROM, and printer are files to UNIX.

Three types of files

- Ordinary file

  Also known as a regular file, contains only data as a stream of characters.

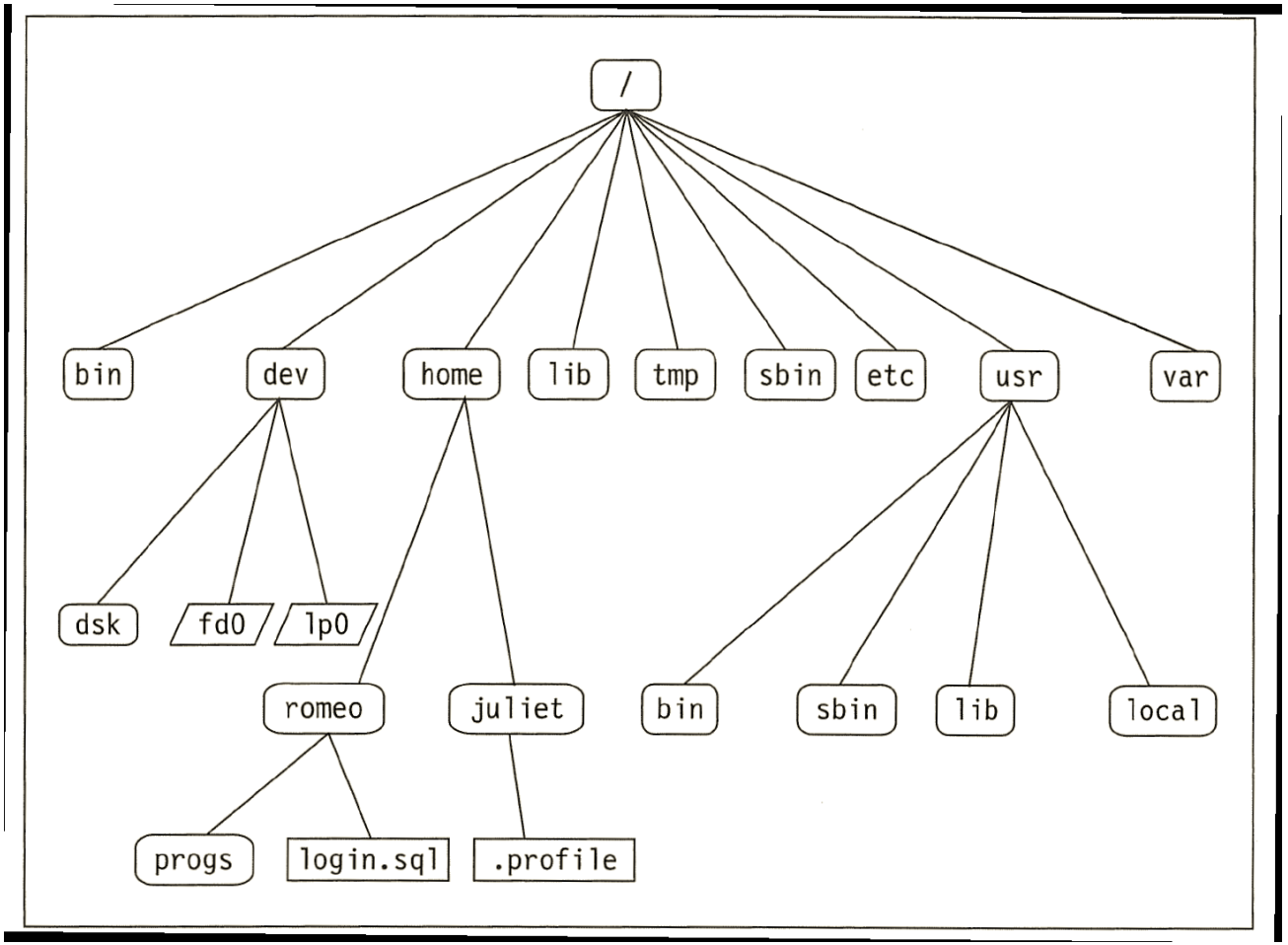
- Directory file

  A folder containing the names of other files and subdirectories.

- Device file

  Represents a device or peripheral.

UNIX Fife System Tree



man hier to view the description of the file system hierarchy

File Commands

ls – directory listing

ls -al – formatted listing with hidden files

cd *dir* - change directory to *dir*

cd – change to home

pwd – show current directory

mkdir *dir* – create a directory *dir*

rm *file* – delete *file*

rm -r *dir* – delete directory *dir*

rm -f *file* – force remove *file*

rm -rf *dir* – force remove directory *dir* * *(be careful with this command)*

File Commands

cp *file1 file2* – copy *file1* to *file2*

cp -r *dir1 dir2* – copy *dir1* to *dir2*; create *dir2* if it

doesn't exist

mv *file1 file2* – rename or move *file1* to *file2,*if *file2* is an existing directory, moves *file1* into directory *file2*

more or less *file* – output the contents of *file*

head *file* – output the first 10 lines of *file*

tail -f *file* – output the contents of *file* as it grows, starting with the last 10 lines

diff – compare files line by line

File Permissions

chmod *octal file* – change the permissions of *file* to *octal*, which can be found separately for user, group, and world by adding:

- 4 – read (r)
- 2 – write (w)
- 1 – execute (x)

Examples:

chmod 777 – read, write, execute for all

chmod 755 – rwx for owner, rx for group and world

Octal Permissions

Octal	Permissions	Significance
0	`---`	No permissions
1	`--x`	Execute only
2	`-w-`	Writable only
3	`-wx`	Writable and executable
4	`r--`	Read only
5	`r-x`	Readable and executable
6	`rw-`	Readable and writable
7	`rwx`	Readable, writable, and executable

More or Less

- Displays files one page at a time
- Use more for large files
- less is more more
  - Allows movement backwards in a file
  - Faster than most standard text editors
- man by default uses more (or less)

A Few Internal Commands of more and less

- Spacebar or f -- One page forward
- b -- One page backward
- [Enter] or j -- One line forward
- [Enter] or k -- One line backward
- /foo -- Searches forward for expression foo

Redirection

Redirect command output to a file

- Think of redirection characters as arrows.
- e.g., ls /dev > test.txt
  - redirects the output of ls into a new file named test.txt

Append command output to the end of a file

- e.g., ls /proc >> test.txt

pipes

|

- As the name implies, a pipe (|) takes the output of one command to the input of another command
  - e.g., ls /usr/sbin | less; ls /etc | sort –nk 5
- We can actually write short “programs” by stringing pipes together.

tee and script

tee

- Splits a data stream so it flows both into a specified file and continues as tee's stdout
- Useful for saving intermediate steps in a long string of pipes.
  - e.g., ls /usr/sbin | tee processes.txt | less

script

- Script makes a typescript of everything printed on your terminal.
- script can save all dialogue in a file
  - e.g., script mylog.txt

Archive and Compression

tar czf *file.tar.gz files* – create a tar with Gzip compression

tar xzf *file.tar.gz* – extract a tar using Gzip

Using tar to copy folder:

tar cf - . | (cd /target; tar xfp -)

tar cjf *file.tar.bz2* – create a tar with Bzip2 compression

tar xjf *file.tar.bz2* – extract a tar using Bzip2

gzip *file* – compresses *file* and renames it to *file.gz*

gzip -d *file.gz* – decompresses *file.gz* back to *file*

Process Management

ps – display your currently active processes

top – display all running processes

kill *pid* – kill process id *pid*

killall *proc* – kill all processes named *proc* *

bg – resume a stopped job in the background

fg – brings the most recent job to foreground

Network Related Commands

ping *host* – ping *host* and output results

wget *file* – download *file*

ssh *user@host* – connect to *host* as *user*

scp file user@host:path - *secure copy (remote file copy program)*

ifconfig – configure a network interface

netstat -rn – view route table

mount – mount a filesystem

mount -t cifs -o user=IEPCLAN/[your PCLAN account] //ieug0.ie.cuhk.edu.hk/homes /mnt

Searching

grep *pattern files* – search for *pattern* in *files*

grep -r *pattern dir* – search recursively for *pattern* in *dir*

command* | grep *pattern – search for *pattern* in the output of *command*

locate *file* – find all instances of *file*

find - search for files in a directory hierarchy

Shortcuts

Ctrl+C – halts the current command
Ctrl+Z – stops the current command, resume with **fg** in the foreground or **bg** in the background
Ctrl+D – log out of current session, similar to **exit**
Ctrl+W – erases one word in the current line
Ctrl+U – erases the whole line
Ctrl+R – type to bring up a recent command
!! - repeats the last command
exit – log out of current session

Customize ~/.bashrc

The ~/.bashrc file determines the behavior of interactive bash shells
Example of a ~/.bashrc

```
#User specific aliases and functions
alias rm='rm -i'
alias cp='cp -i'
alias mv='mv -i'
alias dir="ls -laF"
alias ls="ls -aF"

# set display prompt and LANG variables
export PS1="\h:\w>"
export LANG=en_US.iso885915
```

References

Linux Documentation Project

http://www.tldp.org/

Tutorials for Linux / UNIX Commands

http://www.cyberciti.biz/tips/linux-unix-commands-cheat-sheets.html