

EXPERIMENT 2

LINUX COMMANDS

- REDIRECTION
- PIPES & FILTERS
- JOB CONTROL
- FILE OWNERSHIP
- FILE PERMISSIONS
- LINKS

REDIRECTION

- Method of directing data between files or programs in useful ways
- **Output redirection:**

The o/p of a program is saved in another file using > operator
- **Input redirection:**

We can read data from a file and give it to another file using < operator

PIPING

- The output from the program on the left is fed as input to the program on the right
- The operator we use is |
- Eg :

\$who

#shows no of logged in users

user :0

2017-02-15 16:45 (:0)

user pts/0

2017-02-15 21:38 (:0)

\$who | wc -l

#shows no of lines of output generated by who

FILTERS

A filter is a program that accepts textual data and then transforms it in a particular way. They take raw data, either produced by another program, or stored in a file, and manipulate it to be displayed in a more suitable way.

- Eg:

```
$ ls
barry.txt  bob  example.png  firstfile  first1
$ ls | head -3
```

```
barry.txt
bob
example.png
```

Filter commands

- **head**
view the first n lines of data.
- **tail**
view the last n lines of data.
- **sort**
organise the data into order.
- **nl**
print line numbers before data.
- **wc**
print a count of lines, words and characters.

JOB CONTROL

A job is defined as a task or command that has started running but not yet finished what it is doing.

- jobs - List all the jobs that are running or suspended.
- fg - Bring the job to the foreground.
- bg - Send the job to the background.
- stop or Ctrl + z - Suspend the job.
- kill or Ctrl + c - Terminate the job.

FILE PERMISSION

- Linux is a multiuser OS
- For example, if your computer is attached to a network, or the Internet, remote users can log in via telnet or ssh (secure shell) and operate the computer

Chmod

The chmod command is used to change the permissions of a file or directory

It is easy to think of the permission settings as a series of bits (which is how the computer thinks about them). Here's how it works:

`rwX rwX rwX` = 111 111 111

`rw- rw- rw-` = 110 110 110

`rwX --- ---` = 111 000 000

and so on...

`rwX` = 111 in binary = 7

`rw-` = 110 in binary = 6

`r-x` = 101 in binary = 5

`r--` = 100 in binary = 4

777 (rwxrwxrwx) : No restrictions on permissions.

755 (rwxr-xr-x) : The directory owner has full access. All others may list the directory, but cannot create files nor delete them.

eg: \$ chmod 600 some_file

(600 stands for rw-----)

FILE OWNERSHIP

You can change the owner of a file by using the chown command

eg: \$ su

Password:

\$ chown new some_file

(suppose new stands for the new owner)

FILE LINKS

- A link is a connection between a file name and the actual data on the disk, ie we can have access to file using multiple names by creating a link.
- Two types of links
 - symbolic links:** Refer to a symbolic path indicating the abstract location of another file
 - hard links :** Refer to the specific location of physical data

Hard links

- Hard link to a file named test.sh is given by command
`$ln test.sh test.sh.save`
- The file test.sh.save is the hard link to the file test.sh. This means if we change the content of anyone of the file then the other one will also change.
- If we accidentally delete test.sh file then other file test.sh.save still exists , ie only one of the links is cut off. The data on the disk is removed only if no links remain to that file.
- Hard links cannot be made to directories.
- Even if original file is moved or deleted, the link exists.

Symbolic links

- Hard links cannot be used for directories. So we use symbolic links. It can be used for files as well.
- Here if you delete or move the original file, the link will break.
- It acts as a shortcut or pointer to original file.
- Symbolic link to a file named test.sh is given by
\$ ln -s test.sh direc , where direc is a directory to which the link is made.

File System Hierarchy

- The linux file system is structured as a tree. The leaf of the tree are ordinary files. The internal nodes of the tree are directories. A subdirectory is the child of another directory.
- File system break files down into two logical categories.

shareable vs unshareable files

- Shareable files are those that can be accessed locally and by remoter hosts, while unsharable files are only available locally.

variable vs static files

- Variable files like documents can be changed at any time; while static files like binaries do not change without an action from the system administrator.

File System Hierarchy Standard Organisation

- The `/boot/` directory
 - Contains static files required to boot the system.
- The `/dev/` directory
 - Contains file system entries which represent devices that are attached to the system. These files are essential for the system to function properly.
- The `/etc/` directory
 - Reserved for configuration files that are local to the machine.
 - The `X11/` and `skel/` are subdirectories of this directory.
- The `/lib/` directory
 - Contain only those libraries needed to execute the binaries in `/bin/` and `/sbin/`.
 - These shared library images are important for booting the system and executing the commands within the root file system.

- The `/mnt/` directory
 - This directory is for temporarily mounted file systems, such as CD-ROMs
- The `/opt/` directory
 - Provides storage for large, static application software packages.
- The `/proc/` directory
 - Contains special files that either extract information from or send information to the kernel
 - This directory can be used to communicate with the kernel
- The `/sbin/` directory
 - Stores executables used by the root user, which are only used at boot time and perform system recovery operations.
- The `/usr/` directory
 - This directory is for files that can be shared across multiple machines. It is read only. Often on its own partition.
 - There are many subdirectories like `|- bin/` , `|- doc/` , `|- tmp` → `/var/tmp`

- The `/usr/local` directory
 - It is for use by the system administrator when installing software locally.
 - It have many subdirectories like `|-bin/` ,`|-doc/` ,`|-games/` etc similar to `/usr/` directory.
- The `/var/` directory
 - Any program that write log files or need `spool/` or `/lock` should write them to `/var/` directory.
 - `/var/` is for variable data files.
 - There are many subdirectories found within this directory
 - `/var`
 - `|- accout/`
 - `|- arpwatch/`
 - `+ - spool/`
 - `|- at/`
 - `|- news/`
 - `|- tmp/`

- The /usr/local directory

- It is for use by the system administrator when installing software locally.
- It have many subdirectories like |-bin/ ,|-doc/ ,|-games/ etc similar to /usr/ directory.

- The /var/ directory

- Any program that write log files or need spool/ or /lock should write them to /var/ directory.
- /var/ is for variable data files.
- The are many subdirectories found within this directory

/var

|- accout/

|- arpwatch/

+ - spool/

|- at/

|- news/

|- tmp/