* The /lib and /lib64 Directories

-> /lib contains libraries for essential program in /bin and /sbin.

-> These library filename either starts with ld or lib.

/lib/libncurses.so.5.9

-> most of these are what is known as dynamically loaded libraries.

-> Some Linux distributions there exists a /lib64 directories containing 64-bit libraries.

-> while /lib contains 32-bit versions.

* Additional Directories under /

Directory Name	Usage
/opt	optional application software pkg
/sys	virtual Pseudo-file system giving information about the system and the hardware, -> can use to alter system parameter.
/srv	site-specific data served up by system seldom used
/tmp	Temporary files on some distribution erased ~~a~~ ~~reags~~
/usr	multi user application utilities ~~and~~ data.

* Comparing Files with diff

→ Now that you know about the file system and it's structure, let's learn how to manage files and directories.

→ diff is used to compare files and directories.

diff options	usage
-c	Provides a listing of difference that include three lines of context before and after the lines differing in content.
-r	→ used to recursively compare subdirectories as well as the current directories
-i	ignore the case of letters
-w	ignore diffrence in spaces and tabs
-q	Be quite : only report if files are diffrent without listing the diffrence

Syntax

↳ diff [options] <filename1> <filename2>

→ diff is meant to be used for text files, binary files, one can use cmp.

* using diff3 and patch

→ You can compare three files at once using diff3, which usage one file as reference basis for other two.

→ The syntax for diff3

$ diff3 my-file comman-file your-file.

→ A patch file contains the deltas require to update an older version of a file to the new one.

syntax

$ diff -Nur originalfile newfile > patchfile

→ Distributing just the patch is more concise and efficient than distributing the entire file

→ For example, if only one line needs to change in file that contains 1000 line, the patch will jus a few line long.

$ patch -p1 < patchfile

$ patch originalfile patchfile.

* Using rsync

-> rsync is very powerful utility. For example, useful way to back up Project.

$ rsync -r Project -X archive-machine:archives/
Project-X

-> Note that rsync can be very destructive.

-> It is highly recommended that you first test your rsync command using --dry-run option.

-> To use rsync at the command line type,

rsync sourcefile destinationfile.

-> where file can be on local file or network file

* Compressing data.

-> File data is often compressed to save disk space and reduce the time it take to transmit over the network.

Command	usage
gzip	The most frequently used linux compression utility
bzip2	Produce file significantly smaller than those produce by gzip
XZ	→ The most space-efficient compression utility used in linux.
zip	→ is often required to examine and decompress archives from othe O.S.

* Compress data using gzip

→ gzip has historically been the most widely used linux compression utility.

→ It compress well and very fast.

Command	usage
gzip *	Compress all files in the current directory each file compressed and renamed a .gz extension.
gzip -r Project X	Compress all files in the project X directory, along with all files in all directory in project X.
gunzip foo	De-compress foo found in the file foo.gz

gunzip = gzip -d

* Compressing Data using bzip2

→ It has same syntax as gzip but usage diffrent algorithms and produces significantly smaller files.

Command	usage
bzip2 *	→ Compress all of the files in the current directories and replace each file with file name with ~~a.bip~~ a.bz2 extension.
bzip2 * .bz2	→ Decompress all of the files with an extension of .bz2 in current directory.

bzip2 -d = bunzip2

Note :- bzip2 has lately deprecated due to lack of maintenace.

* Compressing data using XZ.

Command	Usage
xz *	compresses all of the files in the current directories. and replace each file with a.xz extension
xz foo	compress foo into foo.xz
xz -dk bar.xz	Decompresses bar.xz into bar and does not remove bar.xz.
xz -dcf a.txt b.txt.xz > abcd.txt	Decompresses a mix of compressed and uncompressed file to standard output.
xz -d *.xz	decompresses the file compressed usin xz.

* Handling File using zip

Command	Usage
zip backup *	compresses all files in the current directory and place them in the backup.zip
zip -r backup.zip ~	Archives your login directories (~) and all files and directories under it in backup.zip
unzip backup.zip	extracts all files in backup.zip and place them in current directories.