**File Display Commands**

* **more**

to view a text file one page at a time, press spacebar to go to the next page

**more filename :**show the document one page at a time

**more -num filename** : show the document page few lines as specified bu (-num)

example : **more -10 filename** will show 10 lines for every page

.

* **less**

is much the same as more command except:

a) You can navigate the page up/down using the less command and not possible in more command.

b) You can search a string in less command. (use /keywordto search)

c) “more” was fairly limited, and additional development on “more” had stopped

d) it uses same functions as vi editor

the usage : **less filename**

.

* **Head**

displays the first ten lines of a file, unless otherwise stated.

Examples

**head myfile.txt** – Would display the first ten lines of myfile.txt.

**head -15 myfile.txt** – Would display the first fifteen lines of myfile.txt.

* **Tail**

display the last part of the file

usage : **tail filename**

**tail -n filename** : display the last n lines of the file

**File Permission and Ownership command :-**

**1. chown Command:**

Purpose: Changes the owner of a file or directory.

Syntax:

chown [OPTIONS] USER[:GROUP] FILE

Example:

chown john:users myfile.txt

Explanation:

john: New owner of the file.

users: New group ownership (optional).

myfile.txt: Name of the file to be modified.

**2. chgrp Command:**

Purpose: Changes the group ownership of a file or directory.

Syntax:

chgrp [OPTIONS] GROUP FILE

Example:

chgrp developers myfolder

Explanation:

developers: New group owner.

myfolder: Name of the directory to be modified.

**3. chmod Command:**

Purpose: Changes the permissions of a file or directory.

Syntax:

chmod [OPTIONS] PERMISSIONS FILE

Example:

chmod +x script.sh

Explanation:

+x: Adds execute permission.

script.sh: Name of the script to be modified.

Permission Notation:

u (user/owner)

g (group)

o (others)

a (all)

+ (add permission)

- (remove permission)

= (set exact permission)

Example:

chmod u+rwx,go=rx myfile

Explanation:

u+rwx: Gives read, write, and execute permissions to the owner.

go=rx: Removes read and execute permissions from the group and others.

Second Method:

rwx = 111 in binary = 7

rw- = 110 in binary = 6

r-x = 101 in binary = 5

r-- = 100 in binary = 4

example:

**chmod 600 some\_file**

(rw——-) The owner may read and write a file. All others have no rights. A common setting for data files that the owner wants to keep private.

These commands are essential for managing file and directory permissions in a Linux environment. Use them carefully to ensure the security and integrity of your system.