FILE PERMISSIONS - LINUX NOTES (CHEAT SHEET)

1. FILE PERMISSIONS

- Linux permissions control who can read, write, or execute a file.

Types:

r = read

w = write

x = execute

Permission order:

owner | group | others

Example: rwxr-xr-- (750)

2. MODIFYING PERMISSIONS

- chmod command changes permissions.

Examples:

chmod 755 file : rwxr-xr-x

chmod u+x file : add execute for owner

chmod g-w file : remove write for group

chmod o=r file : set read-only for others

Numeric modes:

7 = rwx

6 = rw-

5 = r-x

4 = r--

3. OWNERSHIP PERMISSIONS

- chown changes file owner.

Examples:

chown user file

chown user:group file

- chgrp changes the group:

chgrp group file

4. UMASK

- umask sets default permissions for new files.

Default file creation:

File: 666 - umask

Directory: 777 - umask

Example:

umask 022 → default file perms = 644 (rw-r--r--)

5. SETUID

- When set on a file, the file runs with the owner's privileges.

Symbol: s in owner execute field.

Example:

-rwsr-xr-x

Used by commands like passwd.

6. SETGID

- File runs with group privileges.

Symbol: s in group execute field.

Example:

-rwxr-sr-x

- For directories: files created inside inherit group.

7. PROCESS PERMISSIONS

- A process inherits user ID and permissions of the user running it.

- real UID = actual user

- effective UID = used for permission checks

8. THE STICKY BIT

- Protects files inside a directory.

- Only the owner can delete their files.

Symbol: t in others execute field.

Example:

drwxrwxrwt (common in /tmp)