**Basic Permission Management in Linux**

**1. What are permissions?**

Every file and folder in Linux has rules about **who can do what**:

* **Read (r)** → view contents
* **Write (w)** → modify/delete
* **Execute (x)** → run (if it’s a program/script)

Permissions apply to:

1. **User (u)** → the file owner
2. **Group (g)** → people in the same group
3. **Others (o)** → everyone else

**2. Viewing permissions**

Command:

ls -l

Example output:

-rwxr-xr-- 1 student student 1200 Sep 7 notes.txt

Breakdown:

* -rwxr-xr-- → permissions
  + **rwx** = user (owner) can read, write, execute
  + **r-x** = group can read, execute
  + **r--** = others can only read

**3. Changing permissions with chmod**

* **Symbolic method**:

chmod u+x file.txt # give execute to user

chmod g-w file.txt # remove write from group

chmod o=r file.txt # set others to read-only

* **Numeric method** (octal):
  + r = 4, w = 2, x = 1
  + Add them up:
    - 7 = rwx, 6 = rw-, 5 = r-x, 4 = r--

Example:

chmod 755 script.sh

Now:

* User = rwx (full)
* Group = r-x (read + run)
* Others = r-x (read + run)

**4. Changing ownership with chown**

The **owner** can be changed with:

sudo chown newuser:newgroup file.txt

Example:

sudo chown alice:developers notes.txt

Owner is now alice, group is developers.

**5. Common Permission Settings**

| **Code** | **Meaning** | **Use Case** |
| --- | --- | --- |
| 644 | rw-r--r-- | Text files (owner can edit, others read-only) |
| 755 | rwxr-xr-x | Scripts/programs (owner full, others can run) |
| 700 | rwx------ | Private files (only owner access) |
| 777 | rwxrwxrwx | Everyone can do everything *Not safe* |

**In short**

* Permissions = rules for user, group, others.
* ls -l = check permissions.
* chmod = change permissions.
* chown = change owner.