

Self-Learning Guide: `chmod` and `chown` in Linux

1 Basic Understanding: Files & Permissions in Linux

In Linux, every file and directory has permissions that decide:

- Who can read it
- Who can write (edit) it
- Who can execute it

Linux divides users into **three categories**:

Category	Meaning
Owner	The user who created the file
Group	Users belonging to the same group
Others	Everyone else

2 Types of Permissions

There are **three types** of permissions:

Permission	Symbol	Meaning
Read	r	View file content
Write	w	Modify file
Execute	x	Run file / access directory

3 Understanding `ls -l` Output (Very Important)

Run:

```
ls -l
```

Example output:

```
-rwxr-xr--
```

How to read this:

Part	Meaning
-	File (<code>d</code> means directory)
rwx	Owner permissions
r-x	Group permissions
r--	Others permissions

4 What is `chmod`?

📌 Meaning

`chmod` = change mode

It is used to change file or directory permissions.

5 Two Ways to Use `chmod`

✓ Method 1: Symbolic Mode (Easy for Beginners)

Format:

```
chmod who+permission filename
```

Symbol	Meaning
u	user (owner)
g	group
o	others
a	all

◆ Examples (Read Slowly)

Give read permission to owner:

```
touch file.txt
ls -l
chmod u+r file.txt
ls -l
```

Remove write permission from group:

```
chmod g-w file.txt
ls -l
```

Give execute permission to everyone:

```
chmod a+x script.sh  
ls -l
```

✓ Method 2: Numeric Mode (Important for Interviews)

Number	Permission
4	read
2	write
1	execute

Combine numbers:

Value	Meaning
7	rwx
6	rw-
5	r-x
4	r--

Example:

```
chmod 755 file.txt  
ls -l
```

Means:

- Owner → rwx
- Group → r-x
- Others → r-x

6 Directory Permissions (Special Case)

For directories:

- r → list files
- w → create/delete files
- x → enter directory

👉 Without x, you **cannot enter** a directory even if r is present.

7 What is chown?

📌 Meaning

chown = change owner

Used to change:

- File owner
- File group

8 Basic chown Syntax

```
chown user filename
```

Example:

```
chown john file.txt  
ls -l
```

Now john becomes the owner.

Change Owner and Group Together

```
chown user:group filename
```

Example:

```
chown john:developers file.txt  
ls -l
```

9 Changing Group Only

```
chown :groupname filename
```

Example:

```
chown :admins report.txt  
ls -l
```

10 Recursive Ownership Change (Very Important)

For directories:

```
chown -R user:group directory_name
```

`-R` = recursive (applies to all files inside)

1 1 Practice Flow for Students

Students should follow this order:

1. Create files and directories
2. Check permissions using `ls -l`
3. Apply `chmod` (symbolic first)
4. Apply `chmod` (numeric)
5. Change ownership using `chown`
6. Verify changes again using `ls -l`