OPEN SOURCE OS (LINUX)

برمجة مفتوحة المصدر

LECTURE 4

BY SCHOLAR PHD. EDRIS HUSSAIN MOHAMMED

DEPARTMENT OF INFORMATION SYSTEMS

EDRUIS.HESSAN@GMAIL.COM

2025

Lecture 4:

- Managing ownership
- How ownership works
- Using chown
- Using chgrp
- File Permissions
- File and Directory Access Permissions
- Files common settings

Managing ownership

Anytime a user creates a new file or directory, his or her user account is assigned as that file or directory's "owner." and the group corresponding to that user will be the file or directory's "group owner."

For example, suppose the azad user logs in to her Linux system and creates a file named azadfile1 in home directory. Because he created this file, azad is automatically assigned ownership of azadfile1.

How ownership works

- You can specify a different user and/or group as the owner of a given file or directory. To change the user who owns a file, you must use superuser privileges with sudo command.
- Using chown
  - ✓Using chgrp
  - ✓You can also view file ownership from the

command line using the ls –l command

Using chown

• The chown utility can be used to change the user

or group that owns a file or directory.

Syntax chown user file or directory.

Example: If I wanted to change the file's owner to the azad user, I would enter

chown azad myfile

Note: You can use the –R option with chown to change ownership on many files at once recursively.

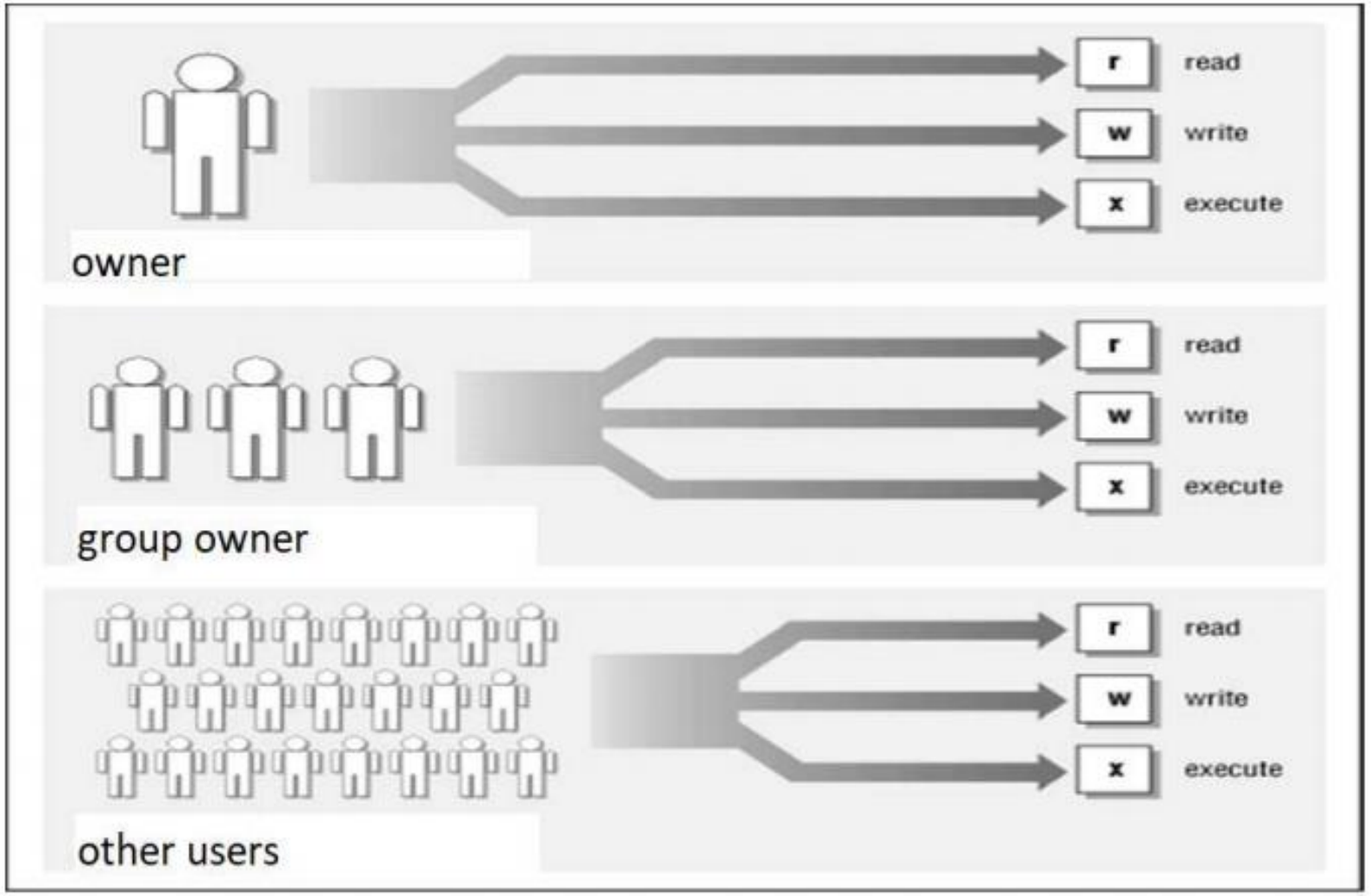
Using chgrp

- In addition to chown, you can also use chgrp to change the group that owns a file or directory.
- Syntax: chgrp group file (or directory)
- Example: chgrp student myfile

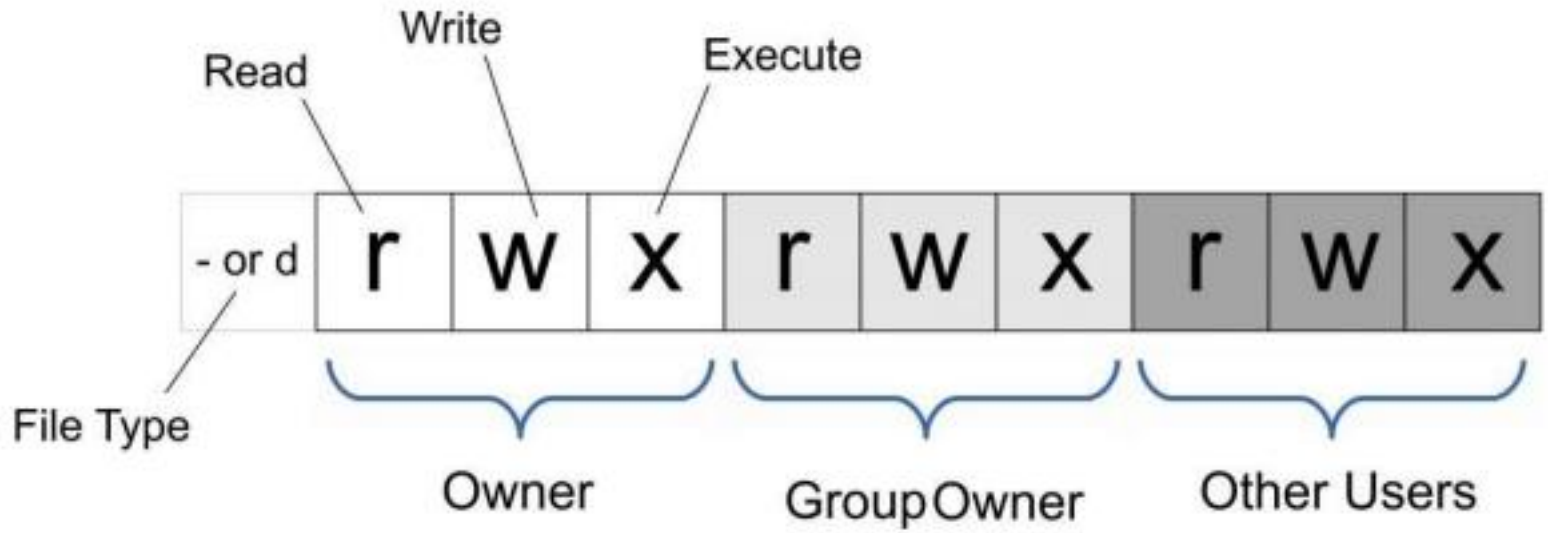
File Permissions

- On a Linux system, each file and directory is assigned access rights for the owner of the file, the members of a group of related users, and everybody else. Rights can be assigned to read a file, to write a file, and to execute a file (i.e., run the file as a program).
- Linux controls access to files on the computer through a system of “permissions.”
- **Permissions** are settings configured to control exactly how files on your computer are accessed and used.
- To see the permission settings for a file, we can use the ls -l command.

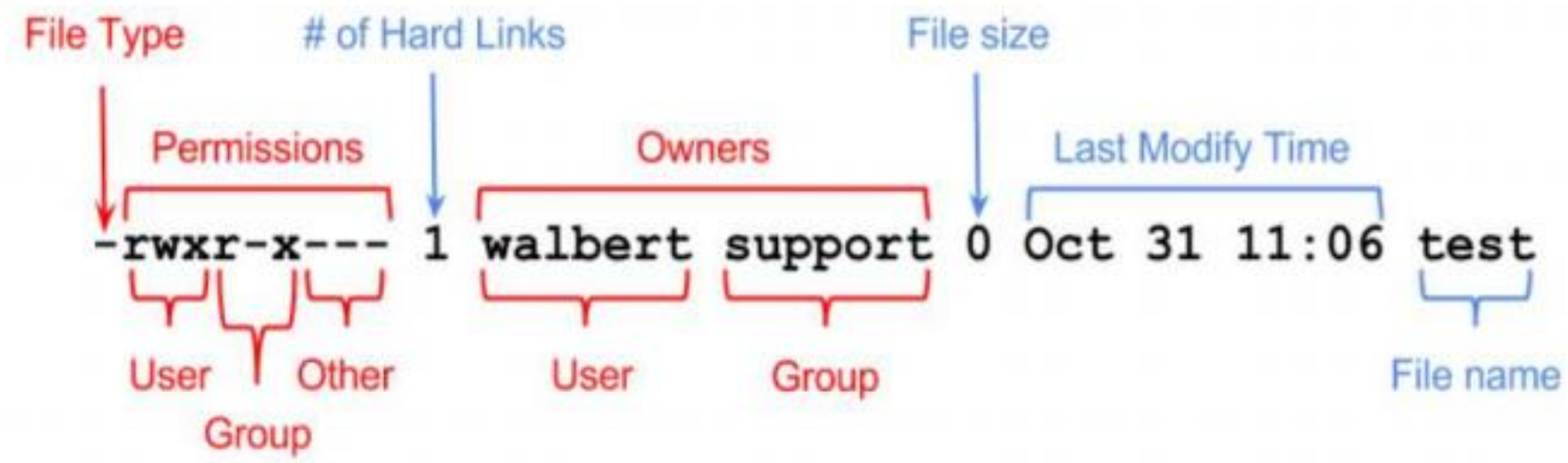
File Permissions



File Permissions



File and Directory Access Permissions



chmod – changing file permissions

- The chmod command is used to change the permissions of a file or directory. To use it, you specify the desired permission settings and the file or files that you wish to modify. There are two ways to specify the permissions. In this lesson we will focus on one of these, called the octal notation method.
- Here's how it works:

  rwx rwx rwx = 111 111 111
  rw- rw- rw- = 110 110 110
  rwx --- --- = 111 000 000

- and so on... rwx = 111 in binary = 7 rw- = 110 in binary = 6 r-x =

  101 in binary = 5 r-- = 100 in binary = 4

- Now, if you represent each of the three sets of permissions (owner, group, and other) as a single digit, you have a pretty convenient way of expressing the possible permissions settings. For example, if we wanted to set myfile to have read and write permission for the owner, but wanted to keep the file private from others, we would:

  sudo chmod 600 myfile

Files common settings

Value	Meaning
777	(rwxrwxrwx) No restrictions on permissions. Anybody may do anything. Generally not a desirable setting.
755	(rwxr-xr-x) The file's owner may read, write, and execute the file. All others may read and execute the file. This setting is common for programs that are used by all users.
700	(rwx------) The file's owner may read, write, and execute the file. Nobody else has any rights. This setting is useful for programs that only the owner may use and must be kept private from others.
666	(rw-rw-rw-) All users may read and write the file.
644	(rw-r--r--) The owner may read and write a file, while all others may only read the file. A common setting for data files that everybody may read, but only the owner may change.
600	(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.

Directory Permissions

- The chmod command can also be used to control the access permissions for directories. Again, we can use the octal notation to set permissions, but the meaning of the **r, w,** and **x** attributes is different:
- **r -** Allows the contents of the directory to be listed if the x attribute is also set.
- **w -** Allows files within the directory to be created, deleted, or renamed if the x attribute is also set.
- **x -** Allows a directory to be entered (i.e. cd dir).

The end