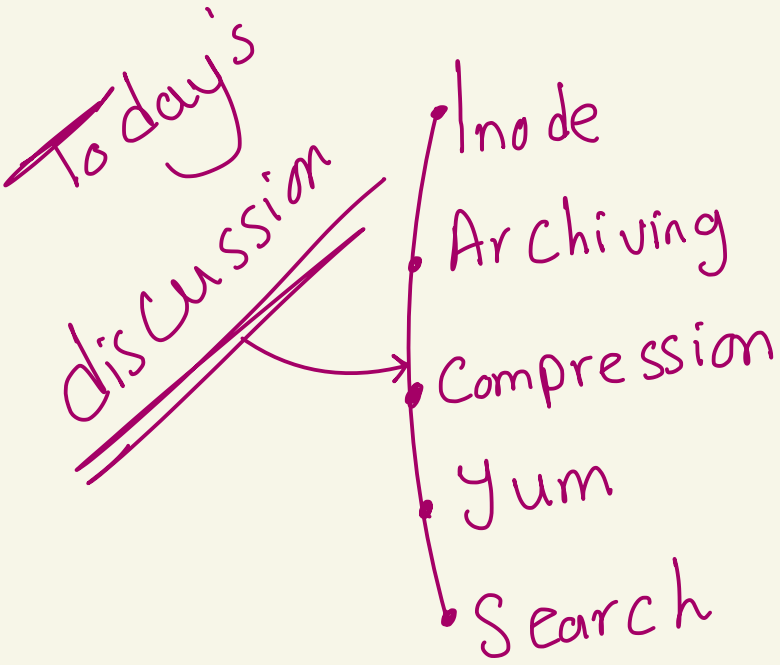
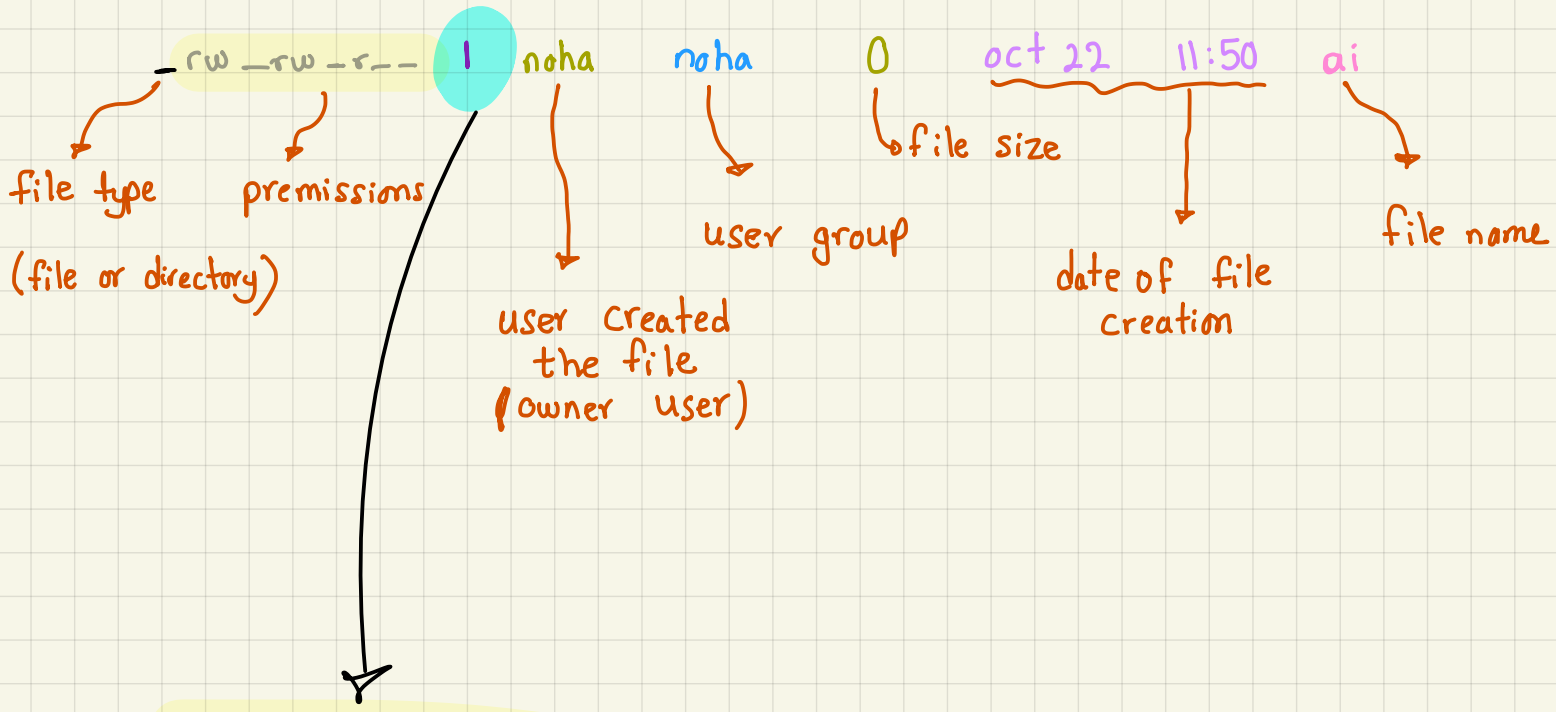
LEC6



Created by : Israa Abdelghany
LinkedIn : www.linkedin.com/in/israa-abdelghany-4872b0222
GitHub : https://github.com/IsraaAbdelghany9

ls -l → show all data about all files or folders in the folder I am in



Today's discussion

(Number of links)

Inode

Index node

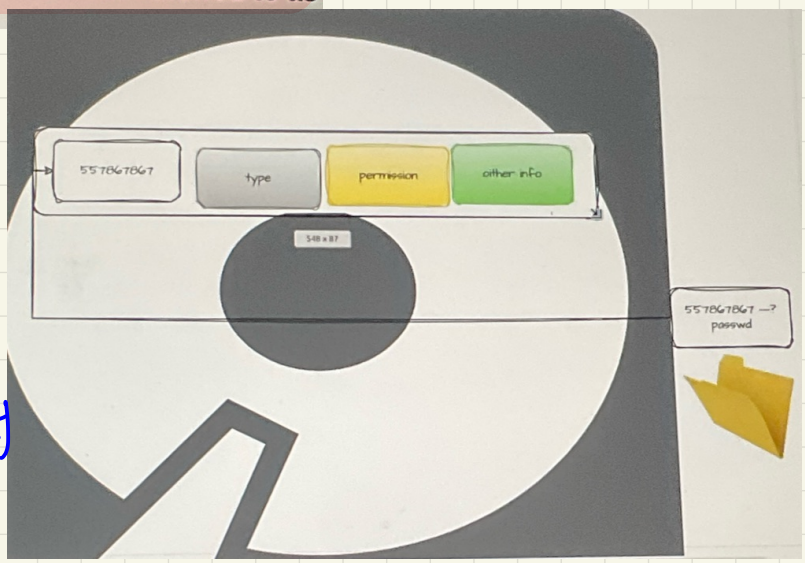
- The information stored in this table for each entry includes the following:
  1. The type of file.
  2. The file's permissions.
  3. The number of links.
  4. The file owner's user ID.
  5. The group owner's GID.
  6. When the file was last changed.
  7. When the file was last accessed.
  8. Where the file is on the media.

Inode Contain:-

Inode

- But It does not contain the file name or file content.
- Names are stored in the directory.
- Each file name knows which inode it has to address to access further file information.
- An inode does not know which name it has; It just know how many names are associated with the inode,These names are referred to as hard links.

file has many names



* ls -i

- all inodes of files
- Folder has logical existance (like Egypt) not physical like file
- inode table contains all info about inodes (media) including:- type, names, owner, group, access ----etc but name is from the file
- inode or inode tables don't have names

* the address is an Inode number
  ↳ (Index node)

* Linux stores the administrative data in (Inode)
  - premissions of the file
  - Address

* each folder has Inode table
* Inode table has info about files
  - it don't Contain Content of the file or its name

* cd .. → change to parent

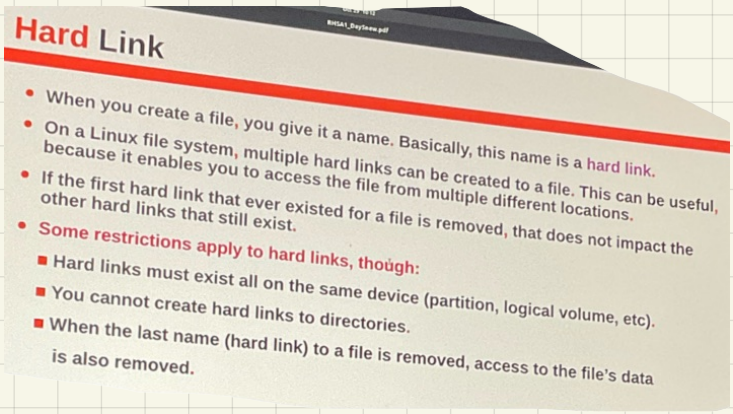
* ls -i → inode number + name of the
* each file has its own inode which have full info about it excep the name of the file and can acces its content.
  the folder have an Inode table that have the ~~name~~ inode of the file

ls -i ⟹ give me data of inode from media
ls -l → all info from inode table and inode of the file
ls -il → both
ls -id → inode of directories only

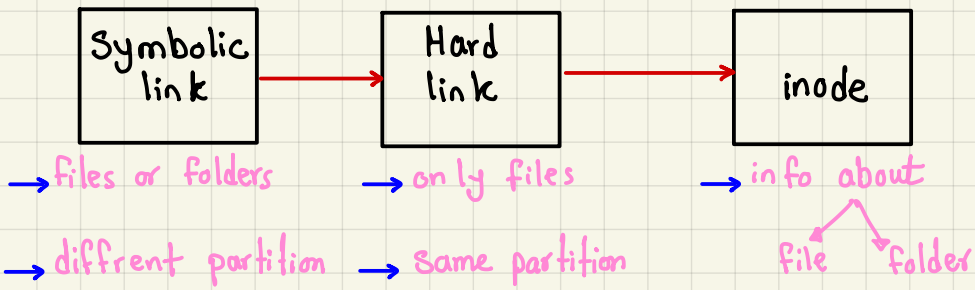
* when you creat a file you give it a name basically hard link

ntf and fat32 formats

* ln → to create a link
* ln duck duckline
* ls -li duck*
* cat duck = cat duck line
* echo "iti" >> duck.line
* cat duck = cat duckline
* ls -l ⇒ will show that the number corrected from 1 to 2



* meta data is data about data
  → part from the media
* ln duckline duck
* copy creates a new copy (physical) for the file with new inode table and not changing the number of links
* hard link is the name of the file
* mv change only the hard link to rename the file



ls -li duck*

ln -s duck duck symbolic

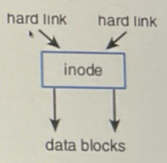
ls -l duck symbolic

↓

lrwx rwx rwx

- To create hard link:
  - ln source-file targetfile or directory
  - ln /home/fatma/myfile hardlinkfile
  - ls -i /home/fatma/myfile hardlinkfile

  11272876 myfile 11272876 hardlinkfile
- To be able to create hard links, you must be the owner of the item that you want to link to .



- simbolic links don't change the number of links

* tar -cvf mydata.tar duck moved duck ⟶ files in .tar
* tar -xvf mydata.tar ⟶ extract data

v → viewing

f → specify which file to put in or extract from

c → copy

x → extract

* tar -rvf mydata.tar myfile → add myfile to mydata.tar
* tar -xvf mydata.tar duck → will extract only duck file not all
* tar -uvf mydata.tar byee passwd ⇒ will update the only file that has been changed (updated)

* tar -tvf mydata.tar → will view the content (may be in 2 files but if extract will be only 1 file)

t → list

*if folder has folder inside it bzip2 and gzip what to use?

Compress comand?

Find vs locate

* package manager ⇒ apt

```
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$ touch duck
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$ ls -li duck
18612234 -rw-rw-r-- 1 noha noha 0 Oct 29 10:15 duck
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$ gedit duck
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$ ls -li duck
18612278 -rw-rw-r-- 1 noha noha 52 Oct 29 10:16 duck
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$ ln duck ducklink
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$ ls -l duck*
-rw-rw-r-- 2 noha noha 52 Oct 29 10:16 duck
-rw-rw-r-- 2 noha noha 52 Oct 29 10:16 ducklink
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$ ls -li duck*
18612278 -rw-rw-r-- 2 noha noha 52 Oct 29 10:16 duck
18612278 -rw-rw-r-- 2 noha noha 52 Oct 29 10:16 ducklink
```

Same inode

File Manipulation

- The cp command:
- Allocates a new inode number for the copy, placing a new entry in the inode table.
- Creates a directory entry, referencing the file name to the inode number within that directory.

```
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$ ls -li
total 24
18612426 -rwxrwxrwx 1 noha noha   22 Oct 26 12:09 abbass
18612366 dr-x------ 3 noha noha 4096 Oct 26 09:57 abc
18612545 -rw-rw-r-- 1 noha noha   15 Oct 26 10:21 abcd
18612278 -rw-rw-r-- 2 noha noha   56 Oct 29 10:29 duck
18612278 -rw-rw-r-- 2 noha noha   56 Oct 29 10:29 ducklink
18612493 -rw-r--r-- 1 noha noha 2655 Oct 26 11:14 mypasswd
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$ cp ducklink duckcpy
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$ ls -li duck*
18612278 -rw-rw-r-- 2 noha noha 56 Oct 29 10:29 duck
18612284 -rw-rw-r-- 1 noha noha 56 Oct 29 11:01 duckcpy
18612278 -rw-rw-r-- 2 noha noha 56 Oct 29 10:29 ducklink
```

cp don't affect number of links

File Manipulation

- The mv command:
- If the destination is on the same file system as the source:
- mv creates a new directory entry with the new file name.
- Example:
  - ls -i f1

    1196100 f1
  - mv f1 f2
  - ls -i f2

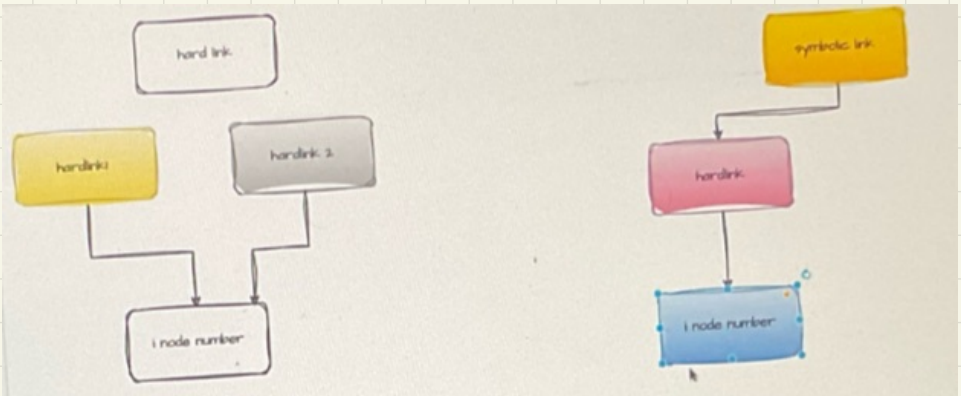
    1196100 f2

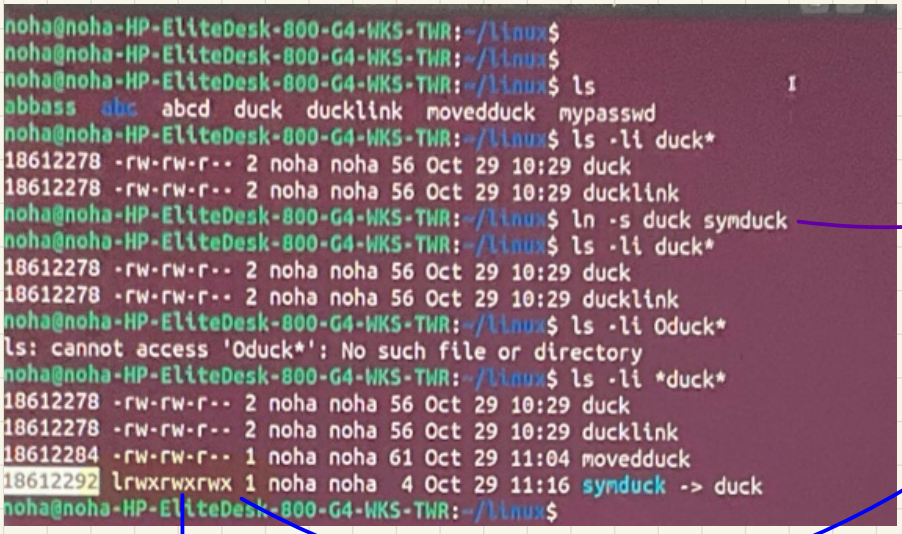
```
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$ ls -li duck*
18612278 -rw-rw-r-- 2 noha noha 56 Oct 29 10:29 duck
18612284 -rw-rw-r-- 1 noha noha 61 Oct 29 11:04 duckcpy
18612278 -rw-rw-r-- 2 noha noha 56 Oct 29 10:29 ducklink
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$ mv duckcpy movedduck
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$ ls -li
total 28
18612426 -rwxrwxrwx 1 noha noha   22 Oct 26 12:09 abbass
18612366 dr-x------ 3 noha noha 4096 Oct 26 09:57 abc
18612545 -rw-rw-r-- 1 noha noha   15 Oct 26 10:21 abcd
18612278 -rw-rw-r-- 2 noha noha   56 Oct 29 10:29 duck
18612278 -rw-rw-r-- 2 noha noha   56 Oct 29 10:29 ducklink
18612284 -rw-rw-r-- 1 noha noha   61 Oct 29 11:04 movedduck
18612493 -rw-r--r-- 1 noha noha 2655 Oct 26 11:14 mypasswd
noha@noha-HP-EliteDesk-800-G4-WKS-TWR:~/linux$
```

symbolic name (link) ⇒ reffer to file (بيشاور عليه)

Symbolic Links

- New entry is made to the inode table for the link The content of this entry is the path to the original file.
- This allows you to use symbolic links across partition boundaries.
- The advantage of symbolic links is that they can link to files on other devices, as well as on directories.
- But when the original file is removed, the symbolic link becomes invalid and does not work any longer.





create Symbolic Link

Note that:-
premissions of link
not of file

note that any Symbolic link has lrwx rwx rwx premissions

★ create symbolic link doesn't increase the number of links associated with the file.

★ the symbolic link has same content as the main file

Archiving

Archiving

- To safeguard your files and directories, you can create a copy, or archive, of the files and directories on a removable medium, such as a cartridge tape. You can use the archived copies to retrieve lost deleted, or damaged files.

Managing Archives with tar

- The Tape Archiver (tar) utility is used to archive files. It designed to stream files to a backup tape.
- To put files on the directory, you need at least read permissions to the file and execute permissions on the directory the file resides in.
- To create an archive:
  - tar -cvf archivename.tar file1 file2 file3
    - c: create a new tar file.
    - v: verbose mode. → will put something
    - f: specify the archive file.

Red Hat

Managing Archives with tar

- To see the contents of the tar archive:
  - tar -tvf /root/homes.tar
    - t: List table of content.
- To extract the contents of an archive:
  - tar -xvf /root/homes.tar
    - x: Extracts files from the tar command.
  - tar -xvf /root/homes.tar -C /tmp
    - C: To specify the target directory you want to extract the file to.

-rw- rw- r-- → tar premissions

Managing Archives with tar

- To add a file to an existing archive or to update an archive:
  - tar -cvf /root/homes.tar /home
  - tar -rvf /root/homes.tar /etc/hosts
    - r: Appends files to an archive.
  - tar -uvf /root/homes.tar /home
    - u: updates an archive, only newer files will be written to the archive.

zcat

③ -mtime

find /etc -u

⑥ -type d

⑧ file abc ⇒ what is the type of abc

⑩ cmp → byte // edge

diff

⑫ if same partition it will be okay if no it will not