



LINUX OPERATING SYSTEM

Chapter 3



IT - INFORMATION SECURITY AND CYBERSECURITY INSTRUCTOR - FACULTY OF
INFORMATION TECHNOLOGY

Philadelphia University

NOVEMBER 15, 2024
Athari Al-Natsheh

Linux Commands

➤ **PWD:** to know which path you are on (Print Working Directory).

Ex: \$pwd ----->/home/username/


➤ **LS:** to list the files in the directory to see (List) ----->**\$ ls [options] [path]**

Ex:\$ls----->Desktop Documents Download Music

Some important options for the list:

- ✓ **\$ ls -l** -----> View files and folders in detail, showing permissions, owner name, file changes, last modified map, and file name.
- ✓ **\$ ls -a** -----> Contains all files including hidden files (that start with a dot ., like .athari).
- ✓ **\$ ls -lh** -----> It displays details like -l, but uses a "human readable" format for file size, such as K for kilobytes and M for megabytes, to make it easier to read.
- ✓ **\$ ls -R** -----> Displays files and folders recursively; that is, displays the contents within all subfolders.
- ✓ **\$ ls -lt** -----> Sorts files by modification date from newest to oldest.
- ✓ **\$ ls -ls** -----> Sorts files by size from largest to smallest.

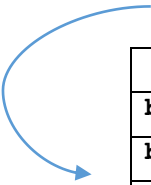
Ex: \$ ls -alh



```
drwxr-xr-x 2 username username 4.0K Nov 10 13:05 .config
drwxr-xr-x 2 username username 4.0K Nov 10 12:00 Documents
-rw-r--r-- 1 username username 1.2M Nov 10 12:30 largefile.mp4
-rw-r--r-- 1 username username 1.5K Nov 10 12:10 smallfile.txt
```

➤ / : System path (root user)

Ex: \$ cd / ls



Folders	Description
bin	
boot	
dev	
etc	
root	
home	
usr	
tmp	
var	

➤ ~ : user home (normal user)

➤ **echo** : to print

Ex: \$ **echo** ~ ----->/home/username

➤ **clear** : to clear the terminal

➤ **--help**: to open the help page for any tools if not working use -help or man + tool name

➤ **cd** : to enter the directory(Change Directory)

✓ **Absolute Path**: Starts with / and indicates the full path

Ex: \$ **cd** /home/username/Documents

✓ **Relative Path**: It indicates the path to the current folder.

Ex: \$ **cd** Documents

Some important options for the Change Directory:

✓ **cd /var/log**: To navigate to a specific folder using an absolute path

- ✓ **cd Documents:** If you are in the **/home/username** folder and want to move to the **Documents** folder inside it
- ✓ **cd or cd ~:** Without any path it takes you back to your personal folder.
- ✓ **cd - :** To roll back to the folder you were in before the last operation
- ✓ **cd / :** To enter the root folder of the system
- ✓ **cd.. :** To move to the top (parent) folder in the hierarchy

Ex: \$cd /Athari/path -----> bash: cd: /Athari/path: No such file or directory

If you type the path with errors like forgetting the slash / or spaces, the command will not work.

➤ **mkdir :** to create a folder (Make Directory) -----> **mkdir [options] Name-of-Folder**

Some important options for the Make Directory:

- ✓ **-p (Parent):** Nested folders will be created with the writer of the full path if it does not exist.

Ex:\$mkdir -p /home/username/Documents/2024/November

- ✓ **-v (Verbose):** A message confirming the creation of each folder is displayed, which helps in achieving the correct success.

Ex:\$mkdir -v athari -----> mkdir: created directory 'athari'

Ex:\$mkdir -pv /home/username/Projects/2024/November

mkdir: created directory '/home/username/Projects'

mkdir: created directory '/home/username/Projects/2024'

mkdir: created directory '/home/username/Projects/2024/November'

➤ **touch** : to create a file -----> \$touch example.txt

Some important options for the Make Directory:

✓ **Create multiple files at once:** touch file1.txt file2.txt file3.txt

✓ **To specify a specific time and date for the timestamp:** touch -t 202411151200 report.txt

✓ **If you want to update the timestamp only if the file exists, without creating it if it doesn't exist:** touch -c myfile.txt

✓ **Update only the access time without affecting the modification time:** touch -a myfile.txt

➤ **rm** : to delete a file or a director(Remove)-----> \$rm example.txt or \$rm file1.txt file2.txt file3.txt

Some important options for the Remove:

✓ **-f (Force):** The system forces files to be deleted without displaying warnings, even if they are protected.-----> \$rm -f example.txt

✓ **-r or -R (Recursive):** Used to delete folders and their contents repeatedly.

\$rm -r example.txt or \$rm -r *



rm: remove regular empty file 'example.txt' ? y

✓ **-i (Interactive):** Request deletion confirmation for each file. -----> \$rm -i example.txt

✓ **-v (Verbose):** View details while deleting, such as the names of deleted files. -----> `$rm -v myfile.txt`

✓ **mv** : to change the name and change the file location

✓ **change the name**-----> `$mv data.csv info.csv`

✓ **change the file location**-----> `$mv report.txt /home/username/Docs/`

✓ **to change the name and change the file location**----->

`$mv oldname.txt /tmp/newname.txt`

✓ **Move multiple files at once to a specific folder**----->

`$mv file1.txt file2.txt file3.txt /home/username/Documents/`

✓ **To merge the contents of folder1/ into folder2/**----->

`$mv folder1/* folder2`

➤ **chmod** :(Change File Permissions)The `chmod` command is used in Linux/UNIX systems to change the permissions of files or folders. These permissions control who can read, write, or execute the file/folder./

----->`$chmod [option] [Permissions] namefile/folder`

→ **To specify additional behavior for the command**

How to specify permissions (symbolic or numeric) ←

The name or path of the file/folder./ for which you want to change permissions. ←

✓ **Understanding Permissions in Linux:**

(Every file or folder in Linux has three categories of permissions.)

1. **User (User - u):** The owner of the file

2. **Group (Group - g):** The group of users who use the file.

3. **Others (Others - o): other users.**
4. **All (All -a): everyone**

✓ **Each category can get permissions:**

1. **R: Reading (reading) -----> 4**
2. **W: Write (write) ----->3**
3. **X: Execute (Execute) ----->1**
4. **-:NO permissions ----->0**

✓ **View permissions :** -----> \$ls -l

-rwxr-xr ---

1. **-: Indicates the type of item (file or folder).**
2. **rwx: User permissions.**
3. **r-x: Group permissions.**
4. **r-: Others' Permissions**

✓ **Grant permission:** ----->\$chmod 644 myfile.txt OR \$chmod g+r myfile.txt

+: Grant permission----->\$chmod g+r myfile.txt (منح إذن القراءة للمجموعة)

-: Remove permission----->\$chmod o-w myfile.txt (زالة إذن الكتابة عن الآخرين)

=: Set specific permission----->\$chmod a=rwx myfile.txt (تعيين أذونات كاملة للجميع)

✓ **Make file executable :**----->\$chmod +x script.py

➤ **nano :** to open the file in edit mode to type or edit in it-----> \$nano test.py

✓ **To save the file you are working on**----->Press Ctrl+O, then press Enter to confirm saving.

✓ **Exit**----->Press Ctrl+X