Linux Terminal Commands – Detailed Explanation

Below is a **comprehensive breakdown** of each Linux command, including its purpose, common options, and practical examples.

1. FILE & DIRECTORY OPERATIONS

ls **– List Directory Contents**

Purpose: Lists files and directories in the current or specified location.
Common Options:

- -l → Long format (detailed info)

- -a → Show hidden files (starting with .)

- -h → Human-readable file sizes (KB, MB, GB)
  Example:

bash

ls -lha /home/user *# Lists all files (including hidden) with details*

cd **– Change Directory**

Purpose: Navigates between directories.
Special Paths:

- cd ~ → Go to home directory

- cd .. → Move up one directory

- cd - → Switch to the previous directory
  Example:

bash

cd /var/log *# Moves to /var/log*

pwd **– Print Working Directory**

Purpose: Shows the current directory path.
Example:

bash

pwd *# Output: /home/user/Documents*

mkdir **– Create Directory**

Purpose: Creates a new directory.
Common Options:

- -p → Create parent directories if they don't exist
  Example:

bash

mkdir -p projects/{src,bin,doc} *# Creates nested directories*

touch **– Create Empty File**

Purpose: Creates a new file or updates the timestamp of an existing one.
Example:

bash

touch file.txt *# Creates file.txt if it doesn't exist*

cp **– Copy Files/Directories**

Purpose: Copies files or directories.
Common Options:

- -r → Recursive (for directories)

- -i → Prompt before overwriting
  Example:

bash

cp -r dir1/ dir2/ *# Copies dir1 into dir2 recursively*

mv **– Move or Rename Files**

Purpose: Moves files/directories or renames them.
Example:

bash

mv old.txt new.txt *# Renames old.txt to new.txt*

mv file.txt ~/Documents/ *# Moves file.txt to Documents*

rm **– Remove Files/Directories**

Purpose: Deletes files or directories (**use with caution**).
Common Options:

- -r → Recursive (for directories)

- -f → Force (no confirmation)
  Example:

bash

rm -rf temp/ *# Deletes the 'temp' directory and its contents*

find **– Search Files/Directories**

Purpose: Searches for files/directories based on criteria.
Common Usage:

bash

find /home -name "*.txt" *# Finds all .txt files in /home*

find / -type f -size +10M *# Finds files larger than 10MB*

2. FILE VIEWING/EDITING

cat **– Concatenate/Display Files**

Purpose: Displays file contents or combines files.
Example:

bash

cat file1.txt file2.txt > combined.txt *# Merges two files*

less **– View File Page-by-Page**

Purpose: Displays large files interactively (scroll with ↑/↓).
Example:

bash

less /var/log/syslog *# View system logs*

head **– Show First Lines of File**

Purpose: Displays the first few lines (default: 10).
Example:

bash

head -n 20 file.log *# Shows first 20 lines*

tail **– Show Last Lines of File**

Purpose: Displays the last few lines (useful for logs).
Common Options:

- -f → Follow (live updates)
  Example:

bash

tail -f /var/log/nginx/access.log *# Monitor live log entries*

nano **– Simple Text Editor**

Purpose: Easy-to-use terminal text editor.
Example:

bash

nano notes.txt *# Opens notes.txt for editing*

grep **– Search Text Patterns**

Purpose: Searches for text in files.
Common Options:

- -i → Case-insensitive

- -r → Recursive (search subdirectories)
  Example:

bash

grep -i "error" /var/log/* *# Searches for "error" in all logs*

3. PERMISSIONS & OWNERSHIP

chmod **– Change File Permissions**

Purpose: Modifies read (r), write (w), execute (x) permissions.
Numeric Mode:

- 7 → rwx (Owner)

- 5 → r-x (Group)

- 4 → r-- (Others)
  Example:

bash

chmod 755 script.sh *# Owner: rwx, Group/Others: rx*

chown **– Change File Owner**

Purpose: Changes file/directory ownership.
Example:

bash

chown user:group file.txt *# Sets owner and group*

sudo **– Execute as Superuser**

Purpose: Runs commands with root privileges.
Example:

bash

sudo apt update *# Updates package lists as root*

4. SYSTEM MONITORING

top **– Dynamic Process Viewer**

Purpose: Displays running processes (CPU, memory usage).
Example:

bash

top *# Interactive process monitor*

htop **– Enhanced** top **(Install First)**

Purpose: Colorful, interactive process viewer.
Installation:

bash

sudo apt install htop *# Debian/Ubuntu*

sudo yum install htop *# Red Hat/CentOS*

df **– Disk Space Usage**

Purpose: Shows disk space on mounted filesystems.
Example:

bash

df -h *# Human-readable format*

du **– Directory Space Usage**

Purpose: Estimates file/directory space usage.
Example:

bash

du -sh ~/Downloads *# Shows total size of Downloads*

free **– Memory Usage**

Purpose: Displays RAM and swap usage.
Example:

bash

free -m *# Shows memory in MB*

ps **– Snapshot of Processes**

Purpose: Lists running processes.
Example:

bash

ps aux | grep nginx *# Finds Nginx processes*

5. NETWORKING

ping **– Test Network Connectivity**

Purpose: Checks if a host is reachable.
Example:

bash

ping google.com *# Sends ICMP packets to Google*

curl **– Transfer Data from URLs**

Purpose: Downloads/uploads data via URLs.
Example:

bash

curl -O https://example.com/file.zip *# Downloads file*

wget **– Download Files**

Purpose: Downloads files from the web.
Example:

bash

wget -c https://example.com/large.iso *# Resumes interrupted download*

ssh **– Secure Remote Login**

Purpose: Connects to a remote server securely.
Example:

bash

ssh user@192.168.1.10 *# Logs into remote server*

scp **– Secure File Copy**

Purpose: Copies files between hosts over SSH.
Example:

bash

scp file.txt user@remote:/path/ *# Uploads file to remote*

ifconfig **(Deprecated) vs.** ip

Purpose: Configures network interfaces.
Example:

bash

ip addr show *# Modern alternative to ifconfig*

netstat **– Network Statistics**

Purpose: Displays network connections.
Example:

bash

netstat -tuln *# Lists listening ports*

6. PACKAGE MANAGEMENT

Debian/Ubuntu (APT)

bash

sudo apt update   *# Updates package lists*

sudo apt upgrade   *# Upgrades installed packages*

sudo apt install nano   *# Installs Nano editor*

sudo apt remove nano   *# Removes Nano*

Red Hat/CentOS (DNF/YUM)

bash

sudo dnf update   *# Updates packages (DNF)*

sudo yum install nano *# Installs Nano (YUM)*

7. ARCHIVES & COMPRESSION

tar **– Archive Files**

Purpose: Combines files into a single archive.
Example:

bash

tar -cvf archive.tar dir/ *# Creates archive*

tar -xvf archive.tar   *# Extracts archive*

gzip **/** gunzip **– Compress/Decompress**

Purpose: Compresses files (.gz format).
Example:

bash

gzip file.txt   *# Compresses to file.txt.gz*

gunzip file.txt.gz *# Decompresses*

zip **/** unzip **– ZIP Archives**

Purpose: Creates/extracts ZIP files.
Example:

bash

zip -r archive.zip dir/ *# Creates ZIP*

unzip archive.zip   *# Extracts ZIP*

8. PROCESS CONTROL

kill **– Terminate Process**

Purpose: Stops a running process by PID.
Example:

bash

kill -9 1234 *# Force-kills process with PID 1234*

killall **– Kill by Name**

Purpose: Stops all processes with a given name.
Example:

bash

killall nginx *# Kills all Nginx processes*

bg **/** fg **/** jobs **– Job Control**

Purpose: Manages background/foreground jobs.
Example:

bash

sleep 100 & *# Runs in background*

jobs   *# Lists jobs*

fg %1   *# Brings job 1 to foreground*

9. TEXT PROCESSING

echo **– Print Text**

Purpose: Displays text or variables.
Example:

bash

echo "Hello, World!" >> file.txt *# Appends text to file*

sed **– Stream Editor**

Purpose: Edits text in files or streams.
Example:

bash

sed 's/foo/bar/g' file.txt *# Replaces "foo" with "bar"*

awk **– Pattern Scanning**

Purpose: Processes text line-by-line.
Example:

bash

awk '{print $1}' data.txt *# Prints first column*

sort **/** uniq **– Sorting & Deduplication**

Purpose: Sorts lines and removes duplicates.
Example:

bash

sort names.txt | uniq -c *# Counts unique occurrences*

10. SYSTEM CONTROL

shutdown **– Power Off/Restart**

Purpose: Shuts down or reboots the system.
Example:

bash

shutdown -r now *# Reboots immediately*

reboot **– Restart System**

Purpose: Reboots the machine.
Example:

bash

sudo reboot *# Restarts the system*

date **– Show/Set Date & Time**

Purpose: Displays or modifies system time.
Example:

bash

date "+%Y-%m-%d %H:%M:%S" *# Custom format*

crontab **– Schedule Tasks**

Purpose: Manages cron jobs (automated tasks).
Example:

bash

crontab -e *# Opens cron editor*

*# Add: 0 3 * * * /path/to/backup.sh # Runs daily at 3 AM*

ESSENTIAL SHORTCUTS

Shortcut	Description
Ctrl+C	Kill current command

Shortcut	Description
Ctrl+Z	Suspend process (fg to resume)
Ctrl+D	End terminal session
Ctrl+R	Reverse command search
!!	Repeat last command
!$	Last argument of previous command

Final Notes

- **Use** man <command> for detailed documentation (e.g., man grep).
- **Be cautious with** rm -rf **and** sudo – irreversible damage is possible.
- **Combine commands with pipes (|)** for powerful workflows.