# Basic to Advanced Linux Commands

### Basic File and Directory Management

1. **ls**

This command is used to list directory contents. It displays the files and directories within a specified location.

* + **Important flags:**
    - -l: Provides a detailed listing, including permissions, owner, size, and date.
    - -a: Shows all files, including hidden files (those starting with a dot).
    - -h: Displays file sizes in a human-readable format (e.g., KB, MB, GB).
  + Open your terminal and type ls and press Enter.
    - ls -l
    - ls -a
    - ls -lh

1. **pwd**
   * This command is used to print the current working directory. It displays the full path of the directory you are currently in.
   * Type pwd in your terminal and press Enter.
2. **cd**
   * This command is used to change the current directory. It allows you to navigate the file system.
   * **Important flags:**
     + ~: Represents the user's home directory.
     + ..: Represents the parent directory.
   * To go to your home directory: cd or cd ~
     + To go up one directory: cd ..
     + To go to a specific directory: cd /home/user/Documents (replace with an actual directory). Then use pwd to confirm.
3. **mkdir**
   * This command is used to create a new directory.
   * **Important flags:**
     + -p: Creates parent directories if they don't exist (e.g., mkdir -p /home/user/newdir/subdir).
   * To create a directory named "testdir", type mkdir testdir and press Enter. Then, use ls to see the newly created directory.
4. **rmdir**
   * This command is used to remove an empty directory.
   * First, create an empty directory: mkdir emptydir. Then, remove it with rmdir emptydir. Use ls to confirm.
5. **touch**
   * This command is used to create an empty file or update the timestamp of an existing file.
   * To create an empty file named "myfile.txt", type touch myfile.txt and press Enter. Use ls to see the new file.
6. **rm**
   * This command is used to remove files and directories. USE WITH EXTREME CAUTION.
   * **Important flags:**
     + -r: Removes directories and their contents recursively.
     + -f: Forces removal (be very careful with this).
   * To remove a file: rm myfile.txt (replace with a file you are willing to delete).
     + To remove a directory and its contents: rm -r testdir (replace testdir, and be *very* careful).
     + To remove a directory and its contents forcefully: rm -rf testdir (BE EXTREMELY CAREFUL). Use ls to verify.
7. **cp**
   * This command is used to copy files and directories.
   * **Important flags:**
     + -r: Copies directories recursively.
   * To copy "myfile.txt" to "mynewfile.txt": cp myfile.txt mynewfile.txt.
     + To copy a directory: cp -r sourcedir destdir
     + Use ls to see the copied file.
8. **mv**
   * This command is used to move or rename files and directories.
   * To rename: mv oldname.txt newname.txt
     + To move: mv file.txt /path/to/new/location
     + Use ls to verify.
9. **cat**
   * This command is used to display the content of a file.
   * Create a file: echo "Hello world" > hello.txt
     + Show content: cat hello.txt

### File Content and Manipulation

1. **less**
   * This command is used to display file content, allowing scrolling. It's very useful for long files.
   * less /var/log/syslog (or any long file). Use the arrow keys to navigate, and 'q' to quit.
2. **head**
   * This command is used to display the beginning of a file.
   * **Important flags:**
     + -n: Specifies the number of lines to display.
   * head /var/log/syslog (shows the first 10 lines). head -n 5 /var/log/syslog (first 5 lines).
3. **tail**
   * This command is used to display the end of a file.
   * **Important flags:**
     + -n: Specifies the number of lines to display.
     + -f: Follow the file as it grows (useful for log files).
   * tail /var/log/syslog (last 10 lines). tail -n 5 /var/log/syslog (last 5 lines). tail -f /var/log/syslog (follow the file as it grows).
4. **echo**
   * This command is used to display a line of text.
   * echo "Hello, world!"
5. **> and >>**
   * These are redirection operators. > overwrites a file, and >> appends to a file.
   * echo "New content" > file.txt (overwrites file.txt)
     + echo "More content" >> file.txt (adds to file.txt)
     + cat file.txt to see the result.
6. **| (pipe)**
   * This operator sends the output of one command to the input of another command.
   * ls -l | grep "myfile" (lists files, then filters for "myfile").
7. **grep**
   * This command is used to search for a pattern in files.
   * **Important flags:**
     + -i: Ignores case.
     + -r or -R: Recursively searches directories.
     + -v: Inverts the search (shows lines that *don't* match).
   * grep "error" /var/log/syslog (searches for "error" in the syslog).
8. **sed**
   * This command is used for stream editing of text. It can perform powerful find-and-replace operations.
   * sed 's/old/new/g' file.txt (replaces "old" with "new" in file.txt). You can redirect the output to a new file: sed 's/old/new/g' file.txt > newfile.txt
9. **awk**
   * This is a powerful text processing tool, often used for manipulating data in columns.
   * awk '{print $1}' file.txt (prints the first column of file.txt). Create a file with space-separated columns to see this in action.
10. **diff**
    * This command is used to compare files and show the differences between them.
    * Create file1.txt with "line 1\nline 2\nline 3"
    * Create file2.txt with "line 1\nline 4\nline 3"
    * Run diff file1.txt file2.txt

### System Information and Management

1. **uname**
   * This command is used to print system information.
   * **Important flags:**
     + -a: Prints all system information.
   * uname (shows the operating system)
     + uname -a (shows all information)
2. **uptime**
   * This command is used to show how long the system has been running.
   * Type uptime in your terminal.
3. **whoami**
   * This command is used to print the current user.
   * Type whoami in your terminal.
4. **w**
   * This command is used to show who is logged on and what they are doing.
   * Type w in your terminal.
5. **df**
   * This command is used to show disk space usage.
   * **Important flags:**
     + -h: Displays sizes in human-readable format.
   * df -h (shows in human-readable format).
6. **du**
   * This command is used to show directory space usage.
   * **Important flags:**
     + -s: Displays the total size.
     + -h: Displays sizes in human-readable format.
   * du -sh directoryname (shows the total size of a directory).
7. **free**
   * This command is used to show memory usage.
   * **Important flags:**
     + -h: Displays sizes in human-readable format.
   * free -h (shows in human-readable format).
8. **top**
   * This command is used to display running processes in real-time.
   * Type top. Press 'q' to quit.
9. **ps**
   * This command is used to display running processes.
   * **Important flags:**
     + aux: Shows all processes for all users.
   * ps (shows processes for the current terminal)
     + ps aux (shows all processes)
10. **kill**
    * This command is used to send a signal to a process (often to terminate it).
    * **Important signals:**
      + -9: KILL signal (forceful termination).
    * Use ps aux to find the PID (Process ID) of a process you want to kill (be careful!).
      + kill PID (sends the TERM signal).
      + kill -9 PID (sends the KILL signal - use as a last resort).

### User and Permissions

1. **chmod**
   * This command is used to change file permissions.
   * **Important formats:**
     + Symbolic mode (e.g., chmod +x file.sh).
     + Numeric mode (e.g., chmod 755 file.txt).
   * chmod +x script.sh (makes script.sh executable).
     + chmod 755 file.txt (sets rwxr-xr-x permissions).
     + Use ls -l to see the permission changes.
2. **chown**
   * This command is used to change the owner of a file.
   * chown user file.txt (changes owner to "user"). Use ls -l to verify. You may need sudo.
3. **chgrp**
   * This command is used to change the group associated with a file.
   * chgrp group file.txt (changes group to "group"). Use ls -l to verify. You may need sudo.
4. **useradd**
   * This command is used to add a new user.
   * **Important flags:**
     + -m: Create the user's home directory.
   * sudo useradd newuser. You'll then likely want to set a password with sudo passwd newuser. The user will be added to /etc/passwd.
5. **userdel**
   * This command is used to delete a user.
   * **Important flags:**
     + -r: Removes the user's home directory.
   * sudo userdel newuser. Add the -r flag to remove the user's home directory as well: sudo userdel -r newuser.
6. **passwd**
   * This command is used to change a user's password.
   * passwd (changes your own password). sudo passwd otheruser (changes another user's password).
7. **groupadd**
   * This command is used to add a new group.
   * sudo groupadd newgroup. Groups are stored in /etc/group
8. **groupdel**
   * This command is used to delete a group
   * sudo groupdel newgroup

### Networking

1. **ifconfig or ip addr**
   * These commands are used to display network interface configuration. ifconfig is older, ip addr is newer and more powerful.
   * ifconfig or ip addr.
2. **ping**
   * This command is used to test network connectivity.
   * ping google.com. Press Ctrl+C to stop.
3. **netstat**
   * This command is used to display network connections.
   * **Important flags**
     + -t: TCP connections
     + -u: UDP connections
     + -l: Listening sockets
     + -n: Don't resolve names.
     + -p: Show PID/Program name
   * netstat -tulnp (shows listening ports).
4. **ssh**
   * This command is used for Secure Shell remote login.
   * ssh user@hostname (replace with a real username and hostname). The output will depend on the remote system.
5. **wget**
   * This command is used to download files from the web.
   * wget https://example.com/file.txt (downloads file.txt).
6. **curl**
   * This command is used to transfer data from or to a server
   * curl https://example.com (shows the HTML of the page)

### System Management

1. **sudo**
   * This command is used to execute a command as the superuser (root).
   * sudo ls /root (lists the contents of the root directory). You'll be prompted for your password.
2. **apt or yum or dnf**
   * These are package managers. apt is for Debian/Ubuntu, yum is for older Red Hat/CentOS, and dnf is for newer Red Hat/Fedora.
   * sudo apt update (updates package lists on Debian/Ubuntu)
     + sudo yum update (updates packages on older Red Hat/CentOS)
     + sudo dnf update (updates packages on newer Red Hat/Fedora)
3. **systemctl**
   * This command is used to manage systemd services.
   * sudo systemctl status ssh (shows the status of the SSH service)
     + sudo systemctl start ssh (starts the SSH service)
     + sudo systemctl stop ssh (stops the SSH service)
4. **reboot**
   * This command is used to reboot the system.
   * sudo reboot (reboots the system). The system will shut down and restart.
5. **shutdown**
   * This command is used to shut down the system.
   * sudo shutdown -h now (shuts down immediately).
6. **history**
   * This command is used to display the command history.
   * Type history in your terminal.