| **Legend** |  |
| --- | --- |
| **Blue** | **Terminals Commands** |
| **Black** | **Theory** |
| **Green** | **Codes** |

# **Linux Basics**

## What is Linux?

Linux is an open-source operating system kernel first created by Linus Torvalds in 1991. It forms the foundation for many operating systems known as Linux distributions.

## Key Characteristics

* Multi-user operating system: Multiple users can work simultaneously
* Multi-tasking: Can run multiple programs at once
* Open-source: Source code is freely available
* Secure: Built-in security features and regular updates

## Directory Structure

* **/bin** - Essential command binaries
* **/etc** - System configuration files
* **/home** - User home directories
* **/root** - Root user's home directory
* **/var** - Variable data files

## Basic Commands

| **Command** | **Description** |
| --- | --- |
| pwd | Print working directory |
| ls | List directory contents |
| cd | Change directory |
| mkdir | Create directory |
| rm | Remove files/directories |
| cp | Copy files/directories |
| mv | Move/rename files |

## File System Hierarchy

Linux follows a hierarchical structure starting from the root directory (/)

* Everything in Linux is treated as a file, including hardware devices
* File names are case-sensitive
* Hidden files start with a dot (.)

## Process Management

| **Command** | **Purpose** |
| --- | --- |
| ps | Show running processes |
| top | Display system tasks |
| kill | Terminate processes |

These fundamentals form the basis for working with Linux systems and understanding their operation.

A Program under execution is called a Process

# **How to use vim: INSERT MODE COMMANDS**:

| **Command** | **Description** |
| --- | --- |
| :q | Quit editor |
| :w | Save/write file |
| :wq | Save and quit |
| :q! | Force quit without saving |
| ! | Force any command |
| dd | Delete current line |
| yy | Copy current line |
| p | Paste |
| gg | Move cursor to top |
| u | Undo |
| shift+G | Move cursor to bottom |

# **redirections**:

### input redirections

creates file if not exists

echo 'hello' > file1 it overwrites the old content

echo 'everyone' >> file1 it appends the data to the file

### output redirections

the command output will be stored in a file.

cal -3 > test2 overwrites the old content

cal -3 >> test2 appends the data to the file

### error redirections

redirects the error generated by a command to a file

2> overwrite    2>> append

ls test 2> file1  [file 'test' doesn't exist, so the error message will be written in file 'file1']

ls test 2>> file1 [appends the error message to the file]

### Error and output redirection

&> &>>

### output in one file and error in another file:

Example

grep devops jenkins file1.txt \*\*>\*\* ./output.txt \*\*2>\*\* ./error.txt

# **Command separators**

denoted by

* ; (semicolon),
* – (double hyphen)
* && (double &)

# **grep command:**

The grep command is used to search for specific patterns or keywords in files.

### Basic Syntax

grep string filename

### Common Options

* grep root /etc/passwd - Basic search for "root" in passwd file
* grep -c root /etc/passwd - Count occurrences of "root"
* grep -i root /etc/passwd - Case-insensitive search
* grep -r root /etc/ - Recursive search through directories

# **lsattr and chattr command:**

**lsattr :** know the attributes of a certain file

**chattr: a**dd attributes to a certain file

**for example :-**

[root@devops ~]# lsattr file1.txt --------------------- file1.txt [root@devops ~]# man chattr [root@devops ~]# chattr +a ./file1.txt [root@devops ~]# lsattr file1.txt ----a---------------- file1.txt

* a – can only append by redirection
* i – cannot be deleted by the root user

# **File Viewing Commands**

### head and tail

**head command** - Displays first n lines of content

**tail command** - Displays last n lines of content

**Pipeline (|)** - Used to combine commands and filter content

* Example: For a 40-line file:
  + head -20 | tail -10 shows lines 11-20
  + head /etc/passwd | tail -2 shows second-to-last two lines

# **File Linking**

### Types of Links

**Soft Link (Symbolic)** ln -s /root/test1 Videos/test1

**Hard Link** ln /root/test2 Videos/test1

* Key differences:
  + Soft links break if source file is deleted
  + Hard links preserve access even if one file is deleted
  + Check inode number using: ls -i filename

# **User Management**

### User Types

| **Admin / Root** | **UID – 0** |
| --- | --- |
| Regular User | UID – 1000 to 60000 |
| System User | UID – 1 to 999 |

### User Commands

useradd - Create new user id - Display user IDs

* chage -l root - View password aging info

User data stored in: /etc/passwd

# **File Permissions**

### Permission Types

* Read (r) = 4
* Write (w) = 2
* Execute (x) = 1

### User Categories

* user (u)
* group (g)
* other (o)

### Commands

ls -l - List file permissions ls -ld - List directory permissions

# **Access Control Lists (ACL)**

Enhanced permission management:

* setfacl -m u:user100:rwx /tata - Set specific permissions
* getfacl /tata - View access list
* setfacl -b /dir - Remove all ACL permissions