

TRAINING DAY 8 REPORT:

- **Introduction to Command Line**

The **command line** is a **text-based application** used for **viewing, handling, and manipulating files** on your computer without using the mouse.

Other Common Names:

cmd , CLI (Command Line Interface) , Prompt , Console ,Terminal.

It acts as a...

- **User to Kernel interface** – It connects **you (the user)** with the **core (kernel)** of the operating system.

- **Interface to use operating system services** – You can **manage files, run programs, install software**, and more using commands.

Where does it get input from?

- From **input devices** like **keyboard** (when you type commands)
- Or from **files** (like shell scripts)

When does the shell start?

- The **shell starts automatically** when you **log in** to the system

Or when you **open a terminal window**

Why it matters:

Using the command line is essential for:

1. System administration
2. Ethical hacking
3. Programming
4. Automation
5. Linux operation

- **Windows Command Line, Command Shell & PowerShell**

Windows Command Line (CMD)

Also called: **Command Prompt** or **cmd.exe**

- It is the **original command-line interface** in Windows.
- Used to perform **basic tasks** like navigating folders, copying files, checking IP settings, etc.
- Syntax is **based on MS-DOS** (older style).
- Open by typing **cmd** in the Run box or Start menu.

Example commands:

dir → Lists files

cd folder → Changes directory

ipconfig → Shows network info

copy file1 file2 → Copies files

Command Shell

- A "**shell**" is the program that **accepts commands from the user** and sends them to the operating system.
- In Windows, **Command Prompt is a type of shell**.
- It acts as a **user-to-kernel interface**, like the Linux terminal.
- Reads commands **from keyboard or script files**.

PowerShell

- A **more advanced command-line tool** introduced by Microsoft.
- Built on **.NET framework**, supports both command-line and **scripting**.
- More powerful than CMD — used for **automation, system management, and advanced scripting**.

• Linux Command Line

In **Linux**, the **command line** is a **text-based interface** used to **interact with the operating system** directly by typing commands instead of clicking.

In Kali Linux:

- The command line runs inside a **terminal emulator**.
- It uses a **shell** called **Bash** – short for **Bourne Again SHell**.
- **Bash** was created for the **GNU project** as a **free, open-source shell**.
- Bash is the **default shell** in most Linux distributions, including **Kali Linux**.

• Linux File Systems

Linux file system, which is the way files and folders are organized and managed in Linux-based systems like **Kali Linux**.

What is a File System?

A **file system** is the structure that an operating system uses to **organize, store, and manage data** on storage devices like hard drives or SSDs.

In Linux, everything is treated as a **file** — even **hardware** (like keyboard, disk drives) and **processes** are accessed as files!

Key Features of Linux File System:

- It is **hierarchical**, meaning it starts from a **single root directory /** and branches out like a tree.
- There are **no drive letters** (like C:\ in Windows); all drives and devices are mounted under directories.
- It supports **file permissions** and **ownership** for security.

Common Linux File Systems:

File System	Description
ext2	Older file system (no journaling)
ext3	ext2 + journaling support
ext4	Most commonly used, fast and reliable
XFS	High-performance, used in servers
Btrfs	Newer, advanced file system with snapshots
Swap	Not a file system for data; used as virtual memory

Important Directories in Linux:

Directory	Purpose
/	Root of the file system
/home	Contains user folders (e.g., /home/yourname)

Directory	Purpose
/bin	Essential binaries (commands like ls, cp)
/etc	Configuration files
/var	Logs and variable data
/usr	User programs and libraries
/boot	Bootloader files (like GRUB)
/dev	Device files (like USB, hard disk)
/proc	Info about running processes
/tmp	Temporary files

• NTFS, FAT32, EXT4

Types of **file systems** used by operating systems to store and manage files on storage devices like hard drives, SSDs, and USB drives.

1. NTFS (New Technology File System)

- Developed by: **Microsoft**

Used in: **Windows operating systems**

Supports: **Large file sizes** (more than 4GB)

- **File permissions, encryption, compression**

- **Journaling** (helps recover after crashes)

Best for: **Windows internal drives** and **dual booting with Linux**

Linux can read/write NTFS with additional support.

2. FAT32 (File Allocation Table 32)

- Developed by: **Microsoft**

Used in: **Older systems, USB drives, SD cards**

Supports: **Simple structure**, compatible with almost all OS

- **Max file size: 4 GB**

- **Max partition size: 2 TB**

Best for: **USB drives, memory cards, and data transfer between systems**

No file permissions or journaling support.

3. EXT4 (Fourth Extended File System)

- Developed by: **Linux community**
Used in: **Linux distributions** (Ubuntu, Kali, etc.)
Supports: **Large files and partitions**

- **Journaling, faster access, less fragmentation**

- **File permissions and advanced security features**
Best for: **Linux system drives**

Not natively supported by Windows (read/write requires special software).

In summary:

- **NTFS** is great for **Windows**.
- **FAT32** is best for **universal compatibility** but limited.
- **EXT4** is the most powerful for **Linux systems**.