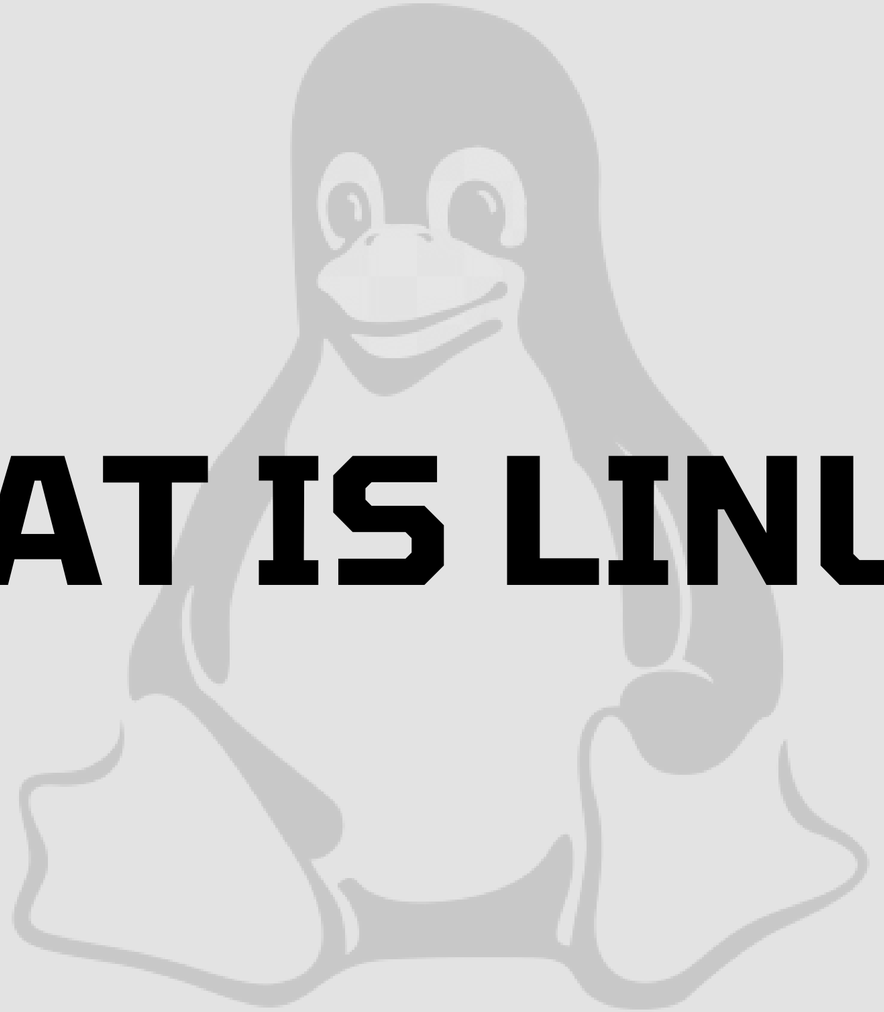

LINUX



# WHAT IS LINUX ?

- Just like Windows, iOS, and Mac OS, Linux is an operating system. In fact, one of the most popular platforms , Android, is powered by the Linux operating system. .

- To put it simply, the operating system manages the communication between your software and your hardware.

- Linux, a popular version of the UNIX Operating System, is an open-source OS because its source code is freely available i.e., it is free to use.

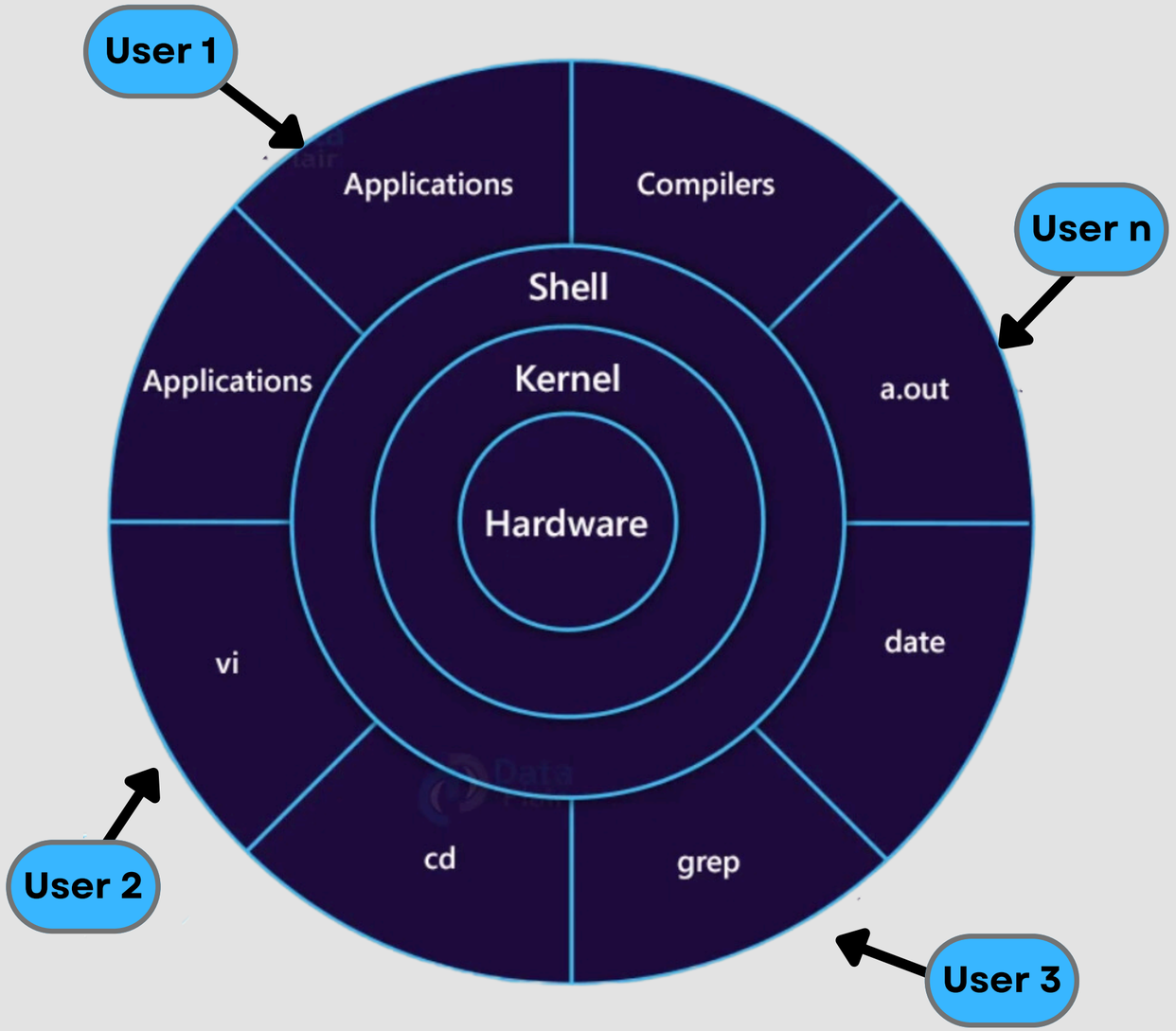
- It is compatible with UNIX and has similar functions to UNIX. It is the fastest-growing OS, used from phones to supercomputers.

- Linux was mainly programmed in C and Assembly languages.

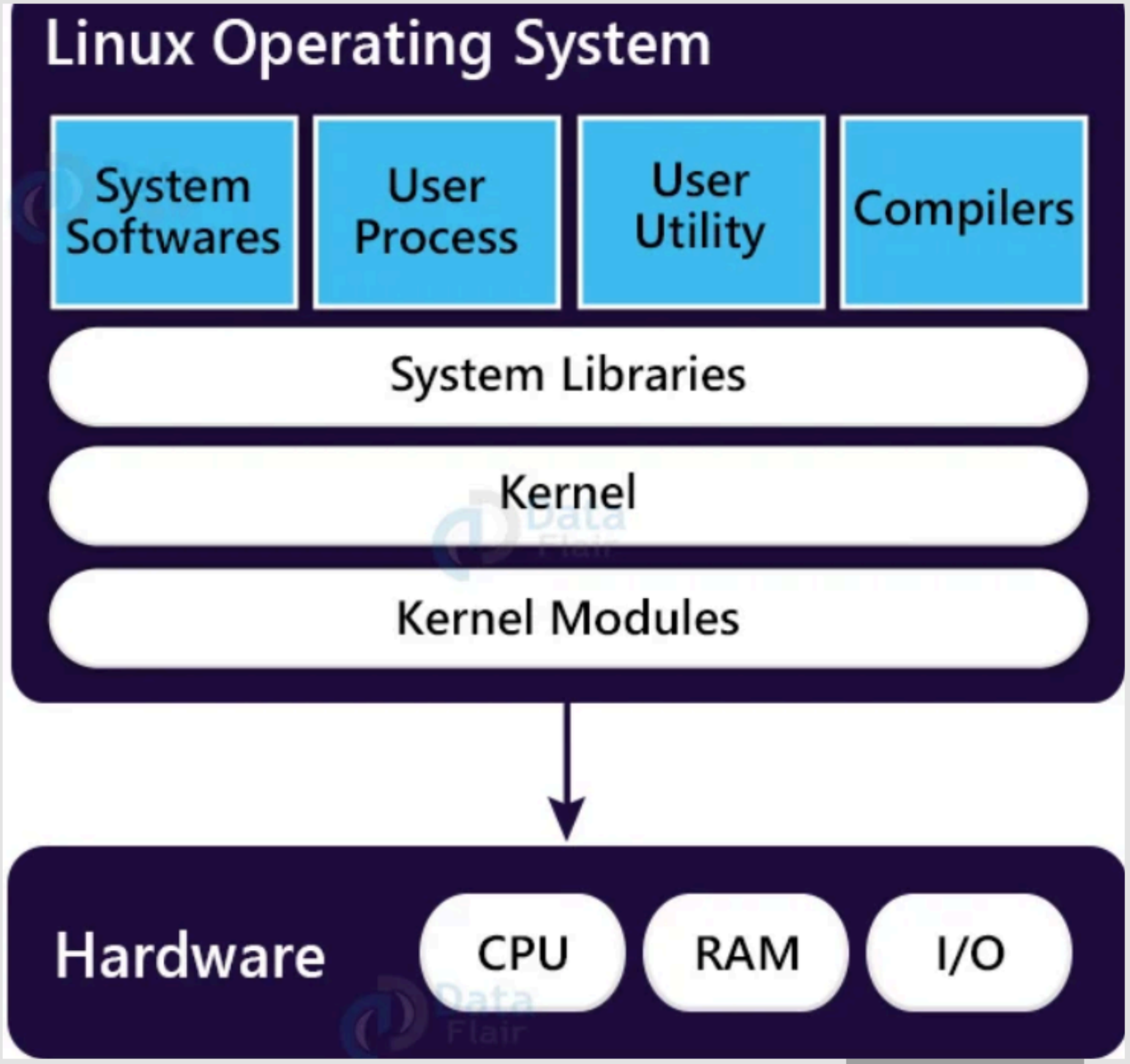
- The development of Linux started in the early 80's when Richard Stallman started the GNU project.

- Linux was first originally developed as a free operating system for Intel x86 processors. Today more than 90% of the worlds fastest super computers use a version of Linux as the OS.

# Architecture of Linux



# Components of Linux System



# WHAT IS UNIX ?

UNIX is a powerful, multiuser, multitasking operating system originally developed in the late 1960s and early 1970s at Bell Labs, the research and development subsidiary of AT&T.

***UNIX Philosophy***

Highlight key aspects of the UNIX philosophy:
Simplicity
Modularity
"Do one thing, and do it well" principle

***Architecture and Components***

Discuss the fundamental architecture of UNIX:
Client-Server model
Hierarchical file system
Command-line interpreter (shell)
Kernel managing system resources and processes

***Portability and Variants***

Emphasize the portability of UNIX due to its C programming language implementation.
Briefly mention major UNIX variants like BSD, AIX, HP-UX, and Solaris.

***Impact and Legacy***

Contribution to internet development
Widespread use in server environments
Influence on modern operating systems like Linux and macOS

# WHAT IS GNU PROJECT ?

- **Origins (1983):**

1. The GNU Project was launched in 1983 by Richard Stallman to develop a free and open-source Unix-like operating system.
2. It aimed to provide users with the freedom to run, study, modify, and distribute software.

- **Key Components:**

1. The GNU Project has developed critical components of a free software stack, including the GNU Compiler Collection (GCC), GNU Emacs, and the GNU Debugger (GDB).

- **GNU General Public License (GPL):**

1. The GPL, created by the GNU Project, is a widely used open-source license.
2. It ensures that software remains free and open by requiring derivative works to be distributed under the same terms.

- **GNU/Linux Operating System:**

1. While the GNU Project aimed to create a complete operating system, it lacked a kernel. Linux, developed by Linus Torvalds in 1991, became the missing piece, leading to the creation of the GNU/Linux operating system.

- **Collaboration with Linux:**

1. The combination of the GNU userland tools and the Linux kernel resulted in a complete and functional operating system. This collaboration demonstrated the effectiveness of combining different free software projects.

# Why Linux?

- Open Source.
- Community support.
- Heavily customizable.
- Most Servers runs on Linux.
- DevOps most of the tools implements on Linux only.
- Automation
- Secure.

# Diffrent Linux distros

## → Popular Desktop Linux OS

- Ubuntu - 2004
- Linux Mint - 2006
- Fedora - 2003
- openSUSE - 1994
- Manjaro - 2011
- Debian - 1993
- elementary OS - 2011
- Zorin OS - 2009
- Kubuntu - 2005
- Kali Linux - 2013
- Solus - 2015

## → Popular Server Linux OS

- Ubuntu Server - 2004
- CentOS - 2004
- Debian Server - 1993
- Red Hat Enterprise Linux (RHEL) - 1995
- openSUSE Leap Server - 1994
- Fedora Server - 2003
- Arch Linux (often used for servers) - 2002
- Oracle Linux - 2006
- Alma Linux - 2021
- Rocky Linux - 2021

# Linux Commands

# Basic Linux Commands

- File Handling
- Text Processing
- File permission
- System Administration
- Process Management
- Archival
- Network
- File Systems
- Advanced Commands
- Editor commands

# sources of learning commands

- **man** <command> - shows all information about the command

- <command> **--help**-shows the available options for that command

# File Handling

# Commands

- **ls**: List files and directories.

  Example: ls

- **cp**: Copy files or directories.

  Example: cp file.txt /destination

- **mv**: Move or rename files and directories.

  Example: mv file.txt newfile.txt

- **rm**: Remove files or directories.

  Example: rm file.txt

- **mkdir**: Create a new directory.

  Example: mkdir new_directory

- **rmdir**: Remove an empty directory.

  Example: rmdir empty_directory

# Text Processing

- **cat**: Display the content of a file.
  Example: cat file.txt

- **grep**: Search for patterns in files.
  Example: grep "pattern" file.txt

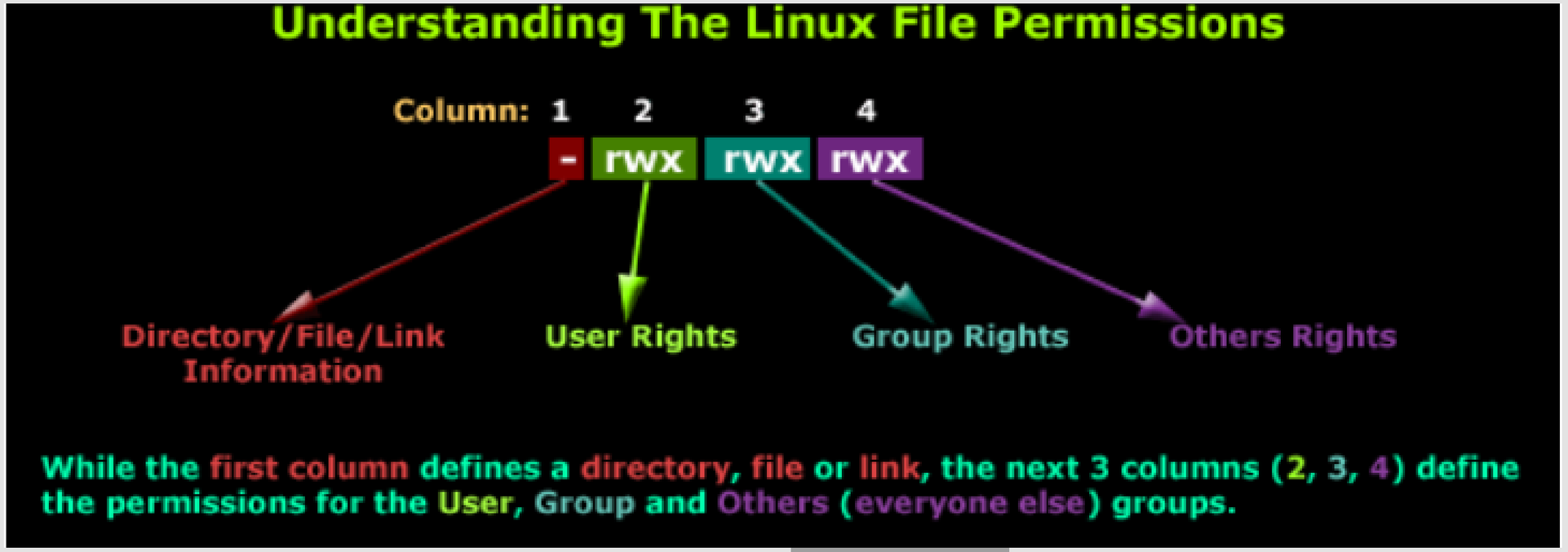
- **sed**: Stream editor for filtering and transforming text.
  Example: sed 's/old/new/' file.txt

- **awk**: A powerful programming language for pattern scanning and text processing.
  Example: awk '{print $1}' file.txt
- **echo:** used for printing messages or displaying the values of variables
  Example: echo "Hello, World!"

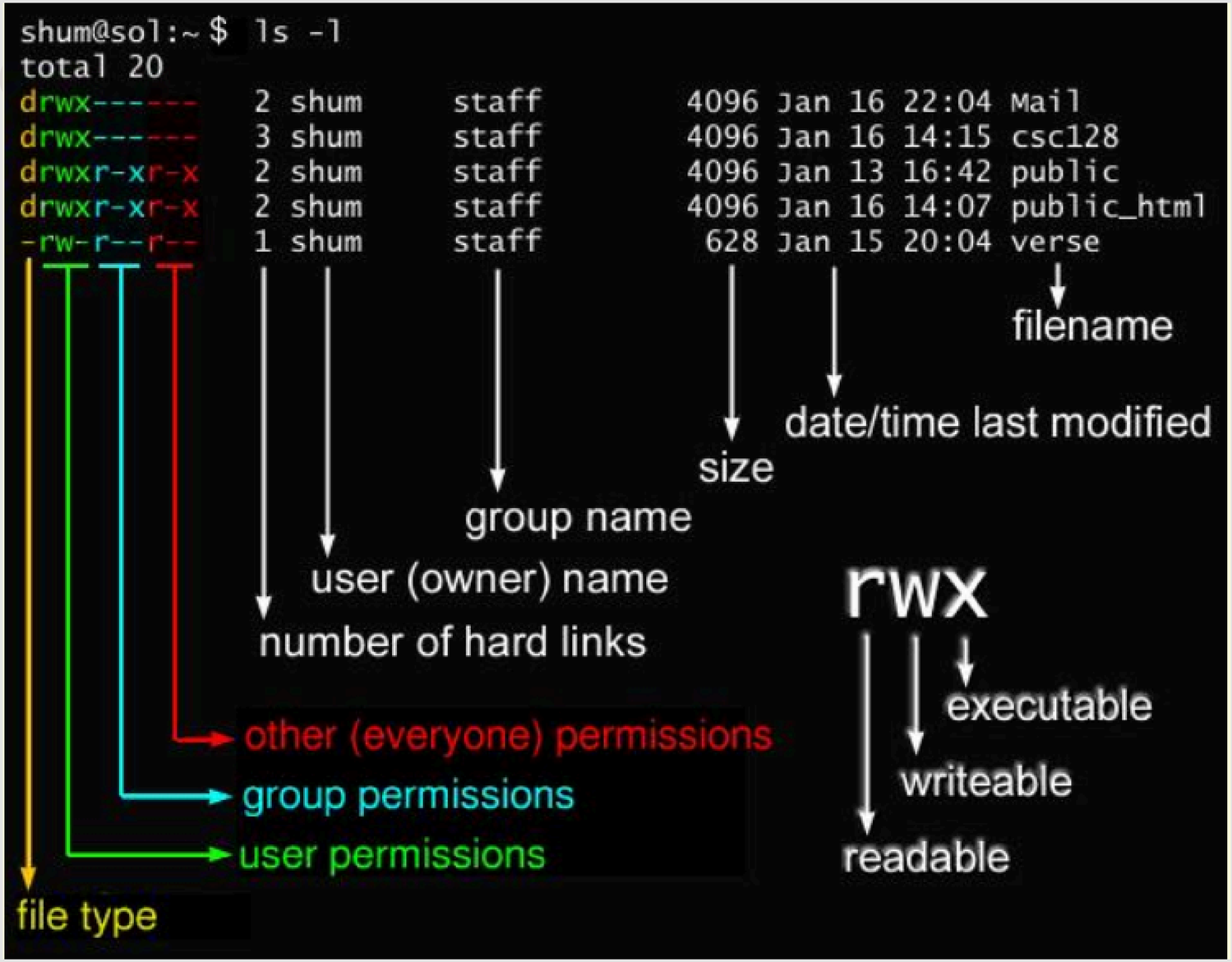
# Linux File Permissions

- 3 types of file permissions - read, write, execute
- 10 bit format from 'ls -l' command

| 1 | 2 3 4 | 5 6 7 | 8 9 10 |
| --- | --- | --- | --- |
| file type | owner | group | others |



# The ls -l command displays a lot of information about the files in the directory:



# System Administration

- **sudo**: Execute a command with administrative privileges.
  Example: sudo command

- **chmod**: Change file permissions.
  Example: chmod +x script.sh

- **chown**: Change file owner and group.
  Example: chown user:group file.txt

- **df**: Display disk space usage.
  Example: df -h

- **du**: Display file and directory space usage.
  Example: du -h

# Process Management

- **ps**: Display information about processes.
  Example: ps aux
- **kill**: Terminate a process.
  Example: kill PID
- **top**: Display real-time system information and processes.
  Example: top

# Archival

- **tar**: Archive files.
  Example: tar -cvf archive.tar file1 file2
- **gzip**: Compress or decompress files.
  Example: gzip file.txt
- **zip:** command is used to compress files and directories into a compressed archive.
  Example: zip compressed_file.zip file.txt
- **unzip:** command is used to decompress files from a compressed archive.
  Example: unzip compressed_file.zip

# Network

- **ping**: Check network connectivity.

  Example: ping example.com

- **ifconfig** or **ip**: Display network configuration.

  Example: ifconfig or ip addr

- **netstat**: Display network statistics.

  Example: netstat -an

# File Systems

- **fdisk**: Display or manipulate disk partition table.
  Example: fdisk -l

- **mkfs**: Create a file system on a device.
  Example: mkfs.ext4 /dev/sdb1

# Advanced Commands

- **find**: Search for files in a directory hierarchy.
  Example: find /path -name "file.txt"

- **grep**: Search for patterns in files.
  Example: grep "pattern" file.txt

- **ssh**: Securely connect to a remote server.
  Example: ssh user@hostname

- **wget or curl**: Download files from the internet.
  Example: wget url or curl -O url

- **alias**: Create an alias for a command.
  Example: alias ll='ls -al'

# Editor commands

- **Nano:** Nano is a simple and user-friendly text editor that runs in the terminal.
  Example: nano filename

- **vim -** Vi Improved, a programmers text editor
  Example: vi hello.c

- **gedit -** A text Editor. Used to create and edit files.
  Example:  gedit hello.c