Born2beroot notes

These notes aim to offer a perpective on operating systems and system administration. It can also serve as a cheatsheet to 42's Born2beroot project, but is not a substitute for a formal tutorial. Each section corresponds to a keyword. They are not necessarily in a relevant order.

1 Kernel

The kernel is the core part of any operating system. It is what actually interacts with the hardware of the computer: one can think of it as the bridge between software (such as applications) and hardware (such as the CPU, memory and storage). It is responsible for:

- resource management: ensures efficient use of CPU, memory and other system resources;
- process management: manages the execution and scheduling of tasks or processes;
- memory management: allocates and monitors memory use, essential for multitasking;
- device drivers: interfaces with hardware;
- file system management: controls access to data storage and manages file systems;
- security and permissions: enforces user permissions and controls access to resources.

2 Operating system

The operating system (OS) is the software that manages computer hardware and resources, providing an interface and environment for users and applications to interact with the system. It is organized in multiple layers:

1. the kernel, the foundation providing core services;
2. system libraries and services (daemons) that offer higher-level functions;
3. the shell and GUI provide user interaction;
4. applications sit at the top, relying on the underlying layers to function.

3 Unix

Unix is a powerful, multiuser, multitasking operating system originally developed in the 1970s at Bell Labs. It set the foundation for many modern operating systems, including Linux and macOS. Known for its simplicity, stability, and portability, Unix introduced key concepts like a hierarchical file system, process management, and the use of plain text for configuration files. Unix has influenced a whole family of OSs called Unix-like systems, which follow its principles and design philosophy. These include popular systems like Linux, BSD (Berkeley Software Distribution), and macOS.

4 Linux

Linux is an open-source operating system based on the Unix principles. When we refer to "Linux", we typically mean the Linux operating system, which is a combination of the Linux kernel with various other tools and software to create a complete environment that can be used for daily tasks, development, server management and more.

5 Debian

Debian is a free open-source operating system based on the Linux kernel. It was started by Ian Murdock in 1993 and has since been developed by a large community around the world. It is one of the oldest Linux distributions.

6 Rocky

Rocky Linux is a free, open-source Linux distribution designed as a downstream, 1:1 binary-compatible replacement for Red Hat Enterprise Linux (RHEL). It was created by Gregory Kurtzer, one of the original co-founders of CentOS, in response to Red Hat's shift in CentOS strategy from CentOS Linux to CentOS Stream, which no longer provided the same stable, production-ready version of RHEL. That would make it designed with enterprise use in mind, but can also be used by other users.

7 BIOS

BIOS (Basic Input/Output System) is a firmware stored on a small memory chip on the motherboard. It is the first software that runs when you turn on your computer, initializing hardware and loading the operating system. It has mainly three functions:

1. POST (Power-On Self-Test): when the computer is powered, checks if the hardware components are functioning properly;
2. bootstrapping: after POST, looks for a bootable device and loads the boot loader or operating system;
3. hardware configuration: provides a setup utility to configure hardware settings, set the boot order, and manage hardware components.

8 UEFI

UEFI (Unified Extensible Firmware Interface) is a modern replacement for BIOS, designed to overcome its limitations. It serves a similar purpose in initializing hardware and loading the operating system but offers more features and flexibility. For instance, UEFI is graphical and mouse-supportive, whereas BIOS is text-based.

9 File system

A file system is a method and structure used by an operating system to manage, store, retrieve, and organize data on a storage device, like a hard drive, SSD, or USB drive. It acts as an organizational layer that enables the OS to locate and access files efficiently. Its key functions are:

1. file organization: structures data in directories and subdirectories for easy access;
2. storage management: allocates and tracks space on the storage device;
3. metadata: maintains information about files, such as names, sizes, and timestamps;
4. access control: defines permissions to control who can access files.

Examples of file systems are NFTS (Windows), ext4 (Linux), APFS (macOS), FAT32 (removable media).

10 Partitions

A partition is a defined section of a computer's storage drive (like an HDD or SSD) that acts as a separate, independent storage area. When you partition a drive, you divide it into multiple logical sections, each of which can function like a separate drive. Partitions help organize data, manage multiple operating systems, or isolate files for different purposes (*e.g.* system files *vs.* personal files). There are three types of partitions.

1. Primary partitions: can hold bootable files and is usually where operating systems are installed. A drive can have up to four primary partitions.
2. Extended partition: used to bypass the primary partition limit by allowing you to create multiple logical partitions within it.
3. Logical partitions: partitions within an extended partition that can hold files or other data but are not directly bootable.

Each partition can be formatted with a different file system. Primary partitions can be booted directly by the computer's BIOS/UEFI, which means you can install an operating system that the system can directly boot from, whereas logical partitions cannot be booted directly. Instead, they require a boot manager or boot loader to be installed that can recognize and load the operating system installed on the logical partition.

11 Logical Volume Manager

LVM (Logical Volume Manager) is a powerful and flexible disk management system used primarily in Linux environments. It allows for the management of disk space in a more dynamic way than traditional partitioning methods. Instead of being restrict to fixed partitions, LVM allows you to allocate disk space more flexibly, making it easier to manage and adjust storage needs as they evolve. It has three key components.

1. Physical volume: a disk or partition on a disk that is initialized for use by LVM. It is the basic building block of LVM. You can add one or more physical volumes to the LVM setup.
2. Volume group: a pool of storage that combines multiple physical volumes into a single logical storage unit. They allow you to aggregate disk space from multiple disks or partitions, providing more flexibility in how that space is allocated.
3. Logical volume: created from the space available in a volume group, it can be thought of as a virtual partition. They can be resized dynamically and can also be formatted with a file system and used like a regular partition.

LVM is particularly beneficial in server environments where disk usage can change frequently.

12 Virtual machine

A virtual machine (VM) is a software-based emulation of a physical computer. It runs an operating system (guest OS) and applications in an isolated environment on top of a host system.

Common commands

Change directory to `/file/path`

```
cd /file/path
```

List contents of working directory

```
ls
```

`-l`	long list format
`-h`	human readable format
`-t`	sort by time