The Linux Booting Process Explained 🐧

ravindra bagale · Follow
4 min read · Sep 18, 2022

Listen Share

When you press the system's power button, you will see the login screen on your screen and the system will begin working.

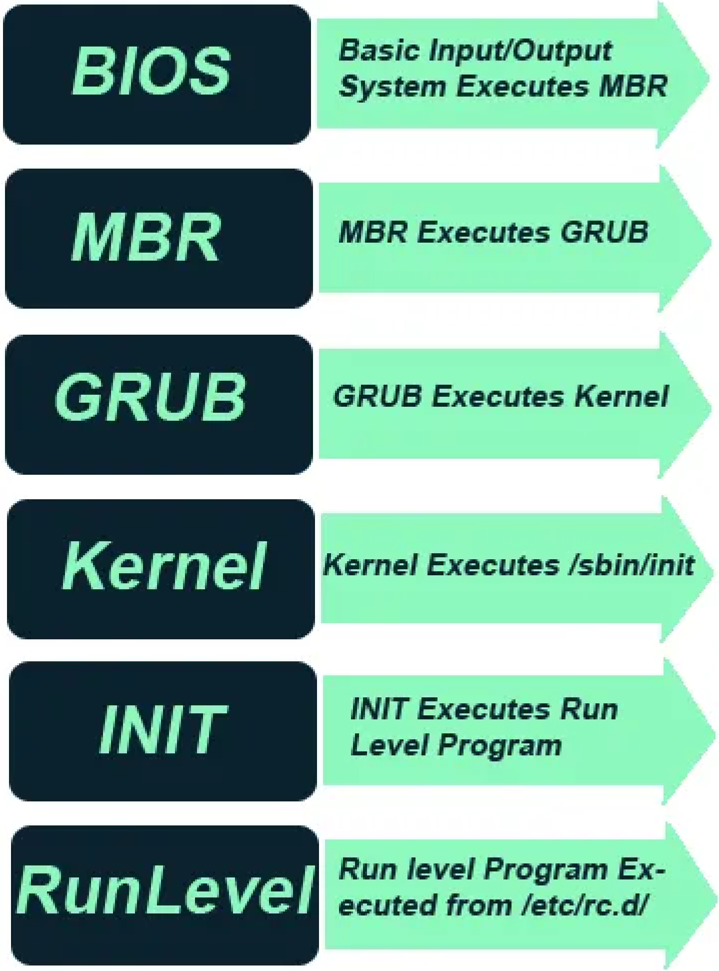
Have you ever wondered what happens between pressing the power button and seeing the login screen show up?

For any operating system, there is a boot process that is executed one by one until you reach the operating system's login screen.

In this thread, we'll go through the Linux boot process step by step.

The Linux boot process is divided into six high-level stages:

- BIOS – Basic Input/Output System executes MBR
- MBR – Master Boot Record execute GRUB
- GRUB – Grand Unified Bootloader executes Kernel
- Kernel – Kernel executes /sbin/init
- Init – Init executes runlevel programs
- Runlevel – Runlevel programs are executed from /etc/rc.d/rc*.d/


BIOS
Basic Input/Output System Executes MBR
MBR
MBR Executes GRUB
GRUB
GRUB Executes Kernel
Kernel
Kernel Executes /sbin/init
INIT
INIT Executes Run Level Program
RunLevel
Run level Program Executed from /etc/rc.d/

Stage 1: BIOS

BIOS is an abbreviation for Basic Input/Output System. In layman's terms, the BIOS's responsibilities are to locate and execute a boot loader (GRUB, LILO) program.

When you initially power on your computer, the BIOS examines the HDD or SSD for integrity.

The BIOS then searches for, loads, and runs the boot loader program stored in the Master Boot Record (MBR). The MBR is sometimes kept on a USB stick or CD-ROM, such as during a live Linux installation.

When the boot loader program is detected, it is loaded into memory and the BIOS gives it control of the computer.

Stage 2: MBR

MBR is an abbreviation for Master Boot Record. It's responsibility is to load and run the GRUB boot loader.

MBR is found in the first sector of the bootable drive. In most cases, /dev/had or /dev/sda.

MBR is 512 bytes in size and consists of three components.
The primary boot loader information is stored in the first 446 bytes.
The next 64 bytes include partition table information.
The last two bytes contain an MBR validation check.

In older systems, the MBR also stores information about GRUB, or LILO.

Stage 3: GRUB

GRUB is an abbreviation for Grand Unified Bootloader.

It contains all information about the operating system image that will be loaded and executed.

If you have more than one operating system, all of them will be listed in this GRUB file, and you can choose which one to use as the default.

On system boot, GRUB displays a splash screen and waits for user input to select an operating system; if you do not type anything, it loads the default kernel image supplied in the grub configuration file.

Grub is familiar with your operating system's filesystem.The GRUB configuration file can be found in "/boot/grub/grub.conf.

Stage 4: Kernel

The kernel is frequently referred to as the heart of every operating system, including Linux. It has complete control over your entire system.

During this stage of the boot process, the GRUB-selected kernel initially mounts the root file system defined in the grub.conf file. Then it runs the /sbin/init program, which is always the first one to run. This is confirmed by its process id (PID), which should always be 1.

The kernel then uses Initial RAM Disk (initrd) to create a temporary root file system until the real file system is mounted.

Stage 5: Init

At this moment, your system executes runlevel applications. To determine the Linux run level, it would look for an init file, normally found at /etc/inittab.

Instead, modern Linux systems use systemd to select a run level. The Linux operating system includes seven (7) different run levels.

- 0 – halt
- 1 – single-user mode
- 2 – Multiuser, without NFS
- 3 – Full multiuser mode
- 4 – unused
- 5 – X11
- 6 – reboot

Init will examine and identify the default initlevel from the file "/etc/inittab" in order to load all appropriate programs for the run level.

You may determine your system's default run level by running the following command:

```
$ grep initdefault /etc/inittab
```

Stage 6: Runlevel

There is only one default run level; it will execute all essential programs for that run level.

The system will examine and execute run-level programs from the folders listed below:

- Run level 0 – /etc/rc.d/rc0.d/
- Run level 1 – /etc/rc.d/rc1.d/
- Run level 2 – /etc/rc.d/rc2.d/
- Run level 3 – /etc/rc.d/rc3.d/
- Run level 4 – /etc/rc.d/rc4.d/
- Run level 5 – /etc/rc.d/rc5.d/
- Run level 6 – /etc/rc.d/rc6.d/

It should be noted that the actual location of these directories vary from distribution to distribution.

You'll find programs that begin with a "S" or "K" for startup or kill in the various run level directories.

Startup programs are run during system startup, and kill programs on system shutdown.

Linux Bootcamp Grub Operating Systems

Follow

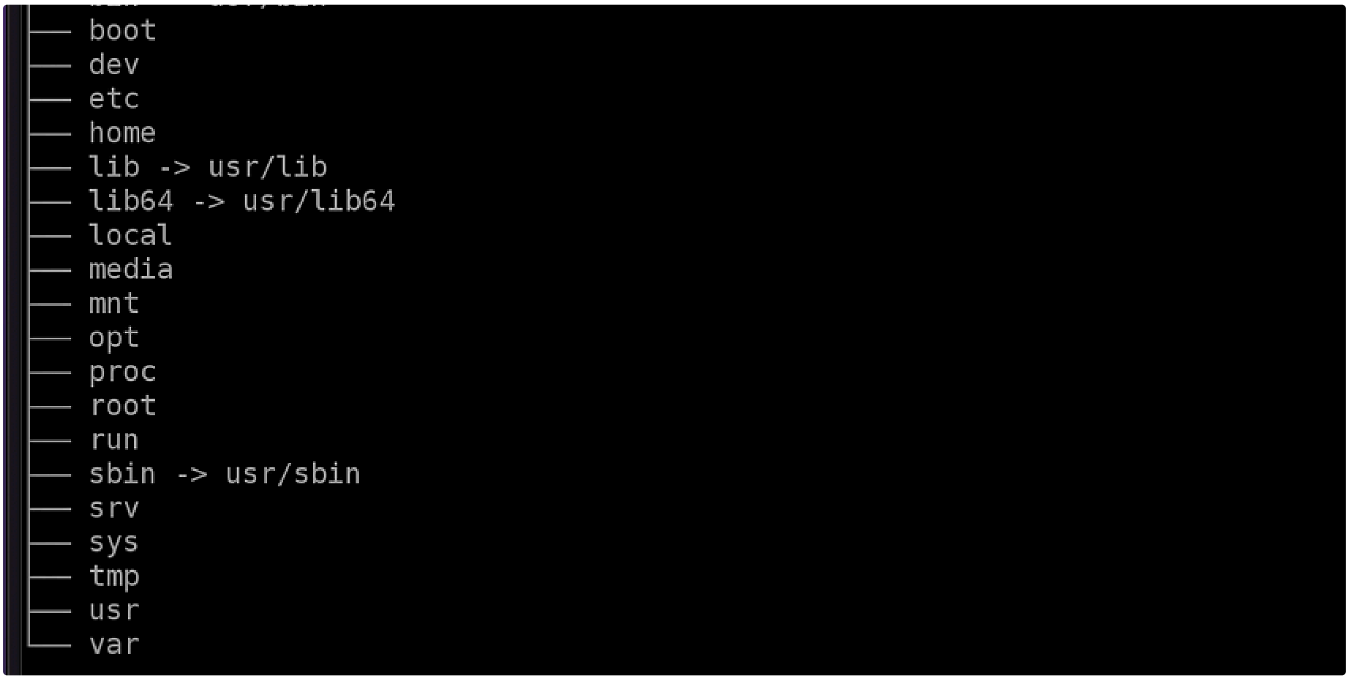
Written by ravindra bagale

31 Followers

10+ years in Cloud Architecture.... Devops Enginner and Technology lover from India

Open in app ↗ Sign up Sign in

Medium Search



```
├── boot
├── dev
├── etc
├── home
├── lib -> usr/lib
├── lib64 -> usr/lib64
├── local
├── media
├── mnt
├── opt
├── proc
├── root
├── run
├── sbin -> usr/sbin
├── srv
├── sys
├── tmp
├── usr
└── var
```

ravindra bagale

Linux Directory Structure