An operating system (OS) is the low-level software that manages resources, controls peripherals, and provides basic services to other software. In Linux, there are 6 distinct stages in the typical booting process.

### ****1. BIOS****

BIOS stands for Basic Input/Output System. In simple terms, the BIOS loads and executes the Master Boot Record (MBR) boot loader.

When you first turn on your computer, the BIOS first performs some integrity checks of the HDD or SSD.

Then, the BIOS searches for, loads, and executes the boot loader program, which can be found in the Master Boot Record (MBR). The MBR is sometimes on a USB stick or CD-ROM such as with a live installation of Linux.

Once the boot loader program is detected, it's then loaded into memory and the BIOS gives control of the system to it.

### ****2. MBR****

MBR stands for Master Boot Record, and is responsible for loading and executing the GRUB boot loader.

The MBR is located in the 1st sector of the bootable disk, which is typically /dev/hda, or /dev/sda, depending on your hardware. The MBR also contains information about GRUB, or LILO in very old systems.

### ****3. GRUB****

Sometimes called GNU GRUB, which is short for GNU GRand Unified Bootloader, is the typical boot loader for most modern Linux systems.

The GRUB splash screen is often the first thing you see when you boot your computer. It has a simple menu where you can select some options. If you have multiple kernel images installed, you can use your keyboard to select the one you want your system to boot with. By default, the latest kernel image is selected.

The splash screen will wait a few seconds for you to select and option. If you don't, it will load the default kernel image.

In many systems you can find the GRUB configuration file at /boot/grub/grub.conf or /etc/grub.conf.

**4. Kernel**

The kernel is often referred to as the core of any operating system, Linux included. It has complete control over everything in your system.

In this stage of the boot process, the kernel that was selected by GRUB first mounts the root file system that's specified in the grub.conf file. Then it executes the /sbin/init program, which is always the first program to be executed. You can confirm this with its process id (PID), which should always be 1.

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

### ****5. Init****

At this point, your system executes runlevel programs. At one point it would look for an init file, usually found at /etc/inittab to decide the Linux run level.

The kernel finally loads **Systemd**, which is the replacement of the old **SysV** init. **Systemd** is the mother of all Linux processes and manages among other things mounting of file systems, starting and stopping services to mention just a few.

Systemd uses the **/etc/systemd/system/default.target** file to determine the state or target that the Linux system should boot into.

* For a desktop workstation (with a GUI) the default target value is 5 which is the equivalent of run level 5 for the old SystemV init.
* For a server, the default target is **multi-user.target** which corresponds to run level 3 in SysV init.

Modern Linux systems use systemd to choose a run level instead. these are the available run levels:

* **poweroff.target (runlevel 0)**: Poweroff or Shutdown the system.
* **rescue.target (runlevel 1)**: launches a rescue shell session.
* **multi-user.target (runlevel 2,3,4)**: Configures the system to a non-graphical (console) multi-user system.
* **graphical.target (runlevel 5)**: Set the system to use a graphical multi-user interface with network services.
* **reboot.target (runlevel 6)**: reboots the system.  
  **Emergency** is matched by **emergency.target**.

systemd will then begin executing runlevel programs.

To check the current target on your system, run the command:

$ systemctl get-default

To set the default target, run the command below.

# systemctl set-default multi-user.target

While the system is running, you can switch the target (run level), meaning only services as well as units defined under that target will now run on the system.

To switch to **runlevel 3**, run the following command.

# systemctl isolate multi-user.target

To change the system to **runlevel 5**, type the command below.

# systemctl isolate graphical.target

### ****6. Runlevel programs****

Depending on which Linux distribution you have installed, you may be able to see different services getting started. For example, you might catch starting sendmail …. OK.

These are known as runlevel programs, and are executed from different directories depending on your run level. Each of the 6 runlevels described above has its own directory:

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

Note that the exact location of these directories varies from distribution to distribution.

If you look in the different run level directories, you'll find programs that start with either an "S" or "K" for startup and kill, respectively. Startup programs are executed during system startup, and kill programs during shutdown.