

THE BOOT PROCESS

BIOS [1]

- When a computer is booted, the processor looks for BIOS program at the end of system memory and runs it.
- Once loaded, the BIOS tests the system, looks for and checks peripherals, and then locates a valid device with which to boot the system (CDROM, Hard Disk) based on the BIOS boot sequence setting.
- The BIOS then loads into memory from *Master Boot Record* or *MBR*.
- MBR (512 bytes in Size) contains machine code instructions for booting machine (boot loader, and partition table)

Boot Loader and Kernel

Below are the basic stages of the boot process for an x86 system [1]:

1. The system BIOS checks the system and launches the first stage boot loader on the MBR of the primary hard disk.
2. The first stage boot loader loads itself into memory and launches the second stage boot loader from the `/boot/` partition.
3. The second stage boot loader loads the kernel into memory, which in turn loads any necessary modules and mounts the root partition read-only.
4. The kernel transfers control of the boot process to the `/sbin/init` program.
5. The `/sbin/init` program loads all services and user-space tools, and mounts all partitions listed in `/etc/fstab`.
6. The user is presented with a login screen for the freshly booted Linux system.

`/sbin/init` Program

- When the `init` command starts, it becomes the parent or grandparent of all of the processes that start up automatically on the system.
 - First, it runs the `/etc/rc.d/rc.sysinit` script, which sets the environment path, starts swap, checks the file systems, and executes all other steps required for system initialization.
 - The `init` command then runs the `/etc/inittab` script, which describes how the system should be set up in each SysV `init` runlevel.
 - Next, the `init` command sets the source function library, `/etc/rc.d/init.d/functions`, for the system, which configures how to start, kill, and determine the PID of a program.

- The init program starts all of the background processes by looking in the appropriate rc directory for the runlevel specified as the default in /etc/inittab.
- The rc directories are numbered to correspond to the runlevel they represent. For instance, /etc/rc.d/rc5.d/ is the directory for runlevel 5.
- One of the last things the init program executes is the /etc/rc.d/rc.local file.
- In runlevel 5, the /etc/inittab runs a script called /etc/X11/prefdm.
The prefdm script executes the preferred X display manager gdm, kdm, or xdm, depending on the contents of the /etc/sysconfig/desktop file.
- Once finished, the system operates on runlevel 5 and displays a login screen.

Source:

[1] https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/5/html/Installation_Guide/s1-boot-init-shutdown-process.html#s2-boot-init-shutdown-bios

Further Study:

<http://www.thegeekstuff.com/2011/02/linux-boot-process/>