Linux Startup Process

Table of Contents

- Importance of understanding Linux startup
- Main process during Linux startup
- Some commands to manipulate Linux runlevel

Importance of understanding Linux startup

- Good for knowledge
- Being able to configure and resolve startup issues

Main process during Linux startup

Two main sequences during a Linux distro startup

- Boot
  - ==> When the computer is turned on, and completed when the kernel is initialized and systemd is launched.
- Startup
  - ==> When the booting sequence is over and it launches all the process necessary of making the computer operational for the user.

boot and **startup** sequences are composed of 6 steps :

1. BIOS (POST)
2. MBR
3. Bootloader (GRUB2)
4. Kernel (Linux)
5. Init (Systemd)
6. Runlevel and scripts

BIOS (**B**asic **I**nput **O**utput **S**ystem)

- Stored in EEPROM (**E**lectrically-**E**rasable **P**rogrammable **R**ead-**o**nly **M**emory).
- Written in Assembly Language.
- First interaction with the physical material.
- Loads and executes the 512 bytes of the disk (MBR).

> Nowadays the BIOS is replaced by UEFI (Unified Extensible Firmware Interface)

MBR (Master Boot Record)

- Contains Bootstrap code which contains information about the boot loader (446 bytes).
- Partitions table to index all the partitions of the disk (64 bytes).
- Boot signature to check if the disk is bootable or not (2 bytes).

GRUB (GRand Unified Bootloader)

- Loads all the available operating system or other boot loaders.
- Loads and executes automatically the default Linux kernel (vmlinuz) and initrd (inital ramdisk) images.
- Contains all the additional modules and drivers for the kernel.

Kernel

- The Linux kernel first mounts the root file system set in `grub.conf` in the line `root=`.
- Then executes the `/sbin/init` program as the fisrt program with root privileges which executes some others scripts.

> Init has the PID (Process IDentifier) of 1.

- Establishes a temporary root file system with initrd until the real file system is mounted. It also contains necessary drivers compiled inside.

Init

- init program reads its initialization files which are in /etc/init.d/ (/etc/inittab before with SysV).
- It sets everything the system needs for its initialization.
  Then it set the default run level.

There are 7 run level from 0 to 1 :

Level	Description
0	Halt
1	Single user mode
2	Multi-user
3	Full multi-user mode
4	Unused
5	X11 (Full multi-user graphical mode)
6	Reboot

Modern Linux systems use systemd which refers with this :

Level	Target
0	poweroff.target
1	rescue.target
2,3,4	multi-user.target
5	graphical.target
6	reboot.target

Runlevels and scripts

- The scripts in /etc/init.d are not directly executed by the init process.
- Each of the directories /etc/rc0.d through /etc/rc6.d contain symbolic links to scripts in the /etc/init.d directory.

> "S" stands for "start" and the "K" stands for "kill"

Some commands to manipulate Linux runlevel

Commands

Current runlevel of the system `sudo runlevel`

> "N" means has not changed since the boot.

Default runlevel `systemctl get-default`

Current loaded targets `systemctl list-units --type target`

Change runlevel `sudo systemctl set-default runlevel.target`

Thank you !