LPIC-1 TRAINING COURSE

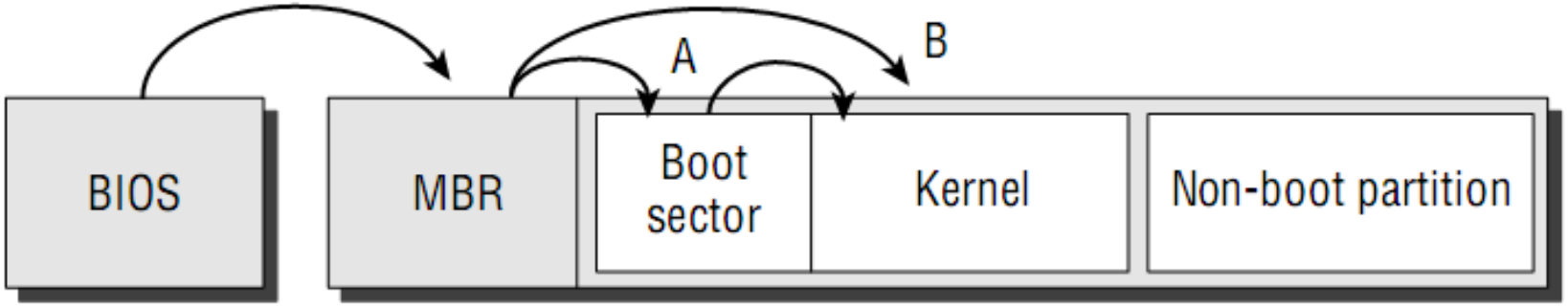
Topic 101: Booting Linux

Objectives

- Able to guide the system through the booting process
- Understand boot manager
- Manage the runlevel of the system

Boot Loader Principles

1. Boot process begins with the BIOS
2. BIOS load coad from the MBR (***primary boot loader*** code)
3. ***Primary boot loader*** either:
   A. Loads the boot sector from ***bootable primary partition*** (***secondary boot loader***), which continues the process by loading an OS kernel. DOS and Windows use this.
   B. Load the OS kernel directly (bypass the ***secondary boot loader***). Linux supports this.



Linux boot loader

- **LILO** (**LI**nux **LO**ader) or **GRUB** (**GR**and **U**nified **B**ootloader)
- Can be installed in MBR or boot sector of primary bootable partition.
- Trouble comes if your system is dual-boot (Windows + Linux)
  - Install LILO/GRUB in MBR: can be wiped out if you re-installing Windows
  - Install LILO/GRUB in boot sector: remain intact, although Windows might configure the system to bypass it.
    - This partition must be a primary partion
    - You must you **FDISK** to re-mark the Linux partition as the boot partitoin

LILO vs GRUB

	LILO	GRUB
Configuration file	/etc/lilo.conf	/boot/grub/grub.conf or /boot/grub/menu.lst
Support booting from a network	No	Yes
Command to install	**lilo**	**grub-install**
Require re-install after changing configuration file	Yes	No

Linux boot process

- System booting process:
  1. System is power on, CPU runs the BIOS
  2. BIOS load the primary boot loader in MBR
  3. Primary boot loader load secondary loader in boot sector
  4. Boot loader load the Linux kernel
  5. Linux kernel initializing devices, mounting root partition and run **/sbin/init**
  6. **/sbin/init** read **/etc/inittab** to determine what other program to run based on default *runlevel*
- Extracting information about the boot process:

```
dmesg | less
less /var/log/messages
```

Runlevel

Runlevel	Purpose
0	A transitional runlevel, meaning that it's used to shift the system from one state to another. Specifically, it shuts down the system. On modern hardware, the system should completely power down. If not, you're expected to either reboot the computer manually or power it off.
1, s, or S	Single-user mode. What services, if any, are started at this runlevel varies by distribution. It's typically used for low-level system maintenance that may be impaired by normal system operation, such as resizing partitions.
2	On Debian and its derivatives, a full multi-user mode with X running and a graphical login. Most other distributions leave this runlevel undefined.
3	On Fedora, Mandriva, Red Hat, and most other distributions, a full multi-user mode with a console (non-graphical) login screen.
4	Usually undefined by default and therefore available for customization.
5	On Fedora, Mandriva, Red Hat, and most other distributions, the same behavior as runlevel 3 with the addition of having X run with an XDM (graphical) login.
6	Used to reboot the system. This runlevel is also a transitional runlevel. Your system is completely shut down, and then the computer reboots automatically.

/etc/inittab

- Syntax:
  id:runlevels:action:process
  - **id** 1-4 characters that identifies its function
  - **runlevels** applicable runlevel
  - **action** action to be taken (**wait**, **respawn**…)
  - **process** process to run
- Default runlevel: describe in line contains **initdefault**
- Example:

```
id:5:initdefault:
l0:0:wait:/etc/init.d/rc 0
l1:1:wait:/etc/init.d/rc 1
l2:2:wait:/etc/init.d/rc 2
l3:3:wait:/etc/init.d/rc 3
l4:4:wait:/etc/init.d/rc 4
l5:5:wait:/etc/init.d/rc 5
l6:6:wait:/etc/init.d/rc 6
```

The SysV Startup Scripts

- Runlevel specific scripts are stored in **/etc/rc.d/rc?.d** or **/etc/init.d/rc?.d**
- All scripts's name are begin with **S** or **K**
  - Example: **S10network**, **K35smb**
  - Scripts are actually symbolic links to main scripts in **/etc/rc.d** or **/etc/init.d/**
  - When entering a runlevel, **rc**:
    - pass **`start`** parameter to **S*** scripts
    - pass **`stop`** parameter to **K*** scripts

Managing Runlevel

- View default runlevel
  `grep :initdefault: /etc/inittab`
- View current runlevel:
  `runlevel`
- Change to another runlevel:
  `telinit runlevel`
- List the services and their applicable runlevels:
  `chkconfig --list [servicename]`
- Modify the runlevels in which a services run
  `chkconfig --level runlevels servicename {on|off|reset}`
  - Example: **`chkconfig --level 345 nfs-common on`**
- Text-based menu driven tools:
  `ntsysv`

Exercise

1. Login as **root** in graphical mode
2. Learn your current runlevel with **`runlevel`**
3. If your system reports it's in runlevel 5, type **`telinit 3`** to switch to runlevel 3. See what's difference?
4. Return to your orginal runlevel with **`telinit 5`**
5. Edit **/etc/inittab** and change the default runlevel to 3
6. Reboot the computer by typing **`reboot now`** or **`shutdown -r now`**
7. Login as **root** again and type **`runlevel`** to verify that you're running in the runlevel you specified in step 5
8. Edit **/etc/inittab** to restore it to its original state
9. Type **`telinit 6`**. This enters runlevel 6, which reboot the system.

Thank You !

BACKUP SLIDES

Example of lilo.conf

```
image=/boot/vmlinuz-2.6.31-14-generic
        label="Lin 2.6.31-14"
        initrd=/boot/initrd.img-2.6.31-14-generic
        read-only

image=/boot/vmlinuz-2.6.31-20-generic
        label="Lin 2.6.31-20"
        initrd=/boot/initrd.img-2.6.31-20-generic
        read-only

image=/boot/memtest86+.bin
        label="Memory Test+"
        read-only

# If you have another OS on this machine (say DOS),
# you can boot if by uncommenting the following lines
# (Of course, change /dev/sda1 to wherever your DOS partition is.)
other=/dev/sda6
        label="Fedora 8"

 other=/dev/sda1
   label="Windows XP"
```

Example of grub.conf

```
default=1
timeout=10
splashimage=(hd0,5)/boot/grub/splash.xpm.gz
#hiddenmenu
password --md5 $1$RW1VW/$4XGAklxB7/GJk0uO47Srx1
title Upgrade to Fedora 11 (Leonidas)
        kernel /boot/upgrade/vmlinuz preupgrade \
      repo=hd::/var/cache/yum/preupgrade stage2=\
      hd:UUID=8b4c62e7-2022-4288-8995-5eda92cd149b:/boot/upgrade/install.img \
      ks=hd:UUID=8b4c62e7-2022-4288-8995-5eda92cd149b:/boot/upgrade/ks.cfg
        initrd /boot/upgrade/initrd.img
title Fedora (2.6.26.8-57.fc8)
        root (hd0,5)
        kernel /boot/vmlinuz-2.6.26.8-57.fc8 ro root=LABEL=FEDORA8 rhgb quiet
        initrd /boot/initrd-2.6.26.8-57.fc8.img
title Fedora (2.6.26.6-49.fc8)
        root (hd0,5)
        kernel /boot/vmlinuz-2.6.26.6-49.fc8 ro root=LABEL=FEDORA8 rhgb quiet
        initrd /boot/initrd-2.6.26.6-49.fc8.img
title GRUB Menu
        rootnoverify (hd0,1)
        chainloader +1
title Windows
        rootnoverify (hd0,0)
        chainloader +1
```