# Boot Loader

•A boot loader is a small program stored in the **MBR** or **GUID** partition table that helps to load an operating system into memory.

•Without a boot loader, your operating system can not be loaded into memory.

•When you turn on your machine, immediately after **POST** (**Power On Self Test**) is completed successfully, the **BIOS** locates the configured bootable media, and reads some instructions from the master boot record (**MBR**) or **GUID** partition table which is the first **512** bytes of the bootable media.

•A boot loader allows you to manage multiple operating systems on your machine and select which one to use at a particular time, without it, your machine can not load the kernel and the rest of the operating system files.

# GRUB (GRand Unified Bootlader)

•**GRUB** is a popular and probably the most used multiboot Linux boot loader available, based on the original **GRUB** (**GRand Unified Bootlader**) which was created by **Eirch Stefan Broleyn**.

•It comes with several improvements, new features and bug fixes as enhancements of the original GRUB program.

•Importantly, **GRUB 2** has now replaced the **GRUB**. And notably, the name **GRUB** was renamed to **GRUB Legacy** and is not actively developed, however, it can be used for booting older systems since bug fixes are still on going.

**GRUB** has the following prominent features:

•Supports multiboot

•Supports multiple hardware architectures and operating systems such as Linux and Windows

•Offers a Bash-like interactive command line interface for users to run GRUB commands as well interact with configuration files

•Enables access to GRUB editor

•Supports setting of passwords with encryption for security

•Supports booting from a network combined with several other minor features

## LILO (Linux Loader)

- **LILO** is a simple yet powerful and stable Linux boot loader. W
- ith the growing popularity and use of GRUB, which has come with numerous improvements and powerful features, **LILO** has become less popular among Linux users.
- While it loads, the word "**LILO**" is displayed on the screen and each letter appears before or after a particular event has occurred.
- However, the development of LILO was stopped in December 2015,
- it has a number of features as listed below:
- Does not offer an interactive command line interface
- Supports several error codes
- Offers no support for booting from a network
- All its files are stored in the first 1024 cylinders of a drive

**Syslinux**

•**Syslinux** is an assortment of light weight boot loaders that enable booting from CD-ROMs, from a network and so on.

•It supports filesystems such as FAT for MS-DOS, and ext2, ext3, ext4 for Linux.

•It as well supports uncompressed single-device Btrfs.

•Note that Syslinux only accesses files in its own partition, therefore, it does not offer multi-filesystem boot capabilities.

# Introduction to Linux System Administration:

- Linux is a major strength in computing technology.
- Most of the webserver, mobile phones, personal computers, supercomputers, and cloud-servers are powered by Linux.
- The job of a Linux systems administrator is to manage the operations of a computer system like maintain, enhance, create user account/report, taking backups using Linux tools and command-line interface tools.
- Most computing devices are powered by Linux because of its high stability, high security, and open-source environment.
- There are some of the things that a Linux system administrator should know and understand:
- Linux File Systems
- File System Hierarchy
- Managing Root/super User
- Basic Bash Command
- Handling File, Directories and Users

## Task of a Linux Administrator:

- System Administration has become a solid criterion for an organization and institute that requires a solid IT foundation.
- Hence, the need for efficient Linux administrators is the requirement of the time.
- The job profile might change from each organization as there may be added responsibilities and duties to the role.
- The main role of the **Linux Systems Administrator** is to manage the operations like install, observe the software and hardware systems and taking backup.
- And also have a good ability to describe an In-depth understanding of technical knowledge.
- Below are some duties of a Linux Administrator:

•Maintain all internet requests inclusive to DNS, RADIUS, Apache, MySQL, PHP.

•Taking regular back up of data, create new stored procedures and listing back-up is one of the duties.

•Analyzing all error logs and fixing along with providing excellent customer support for Webhosting, ISP and LAN Customers .

•Communicating with the staff, vendors, and customers in a professional manner at all times has to be one of his characteristics.

•Enhance, maintain and creating the tools for the Linux environment and its users.

•Detecting and solving the service problems ranging from disaster recovery to login problems.

•Installing the necessary systems and security tools.

•Working with the Data Network Engineer and other personnel/departments to analyze hardware requirements and makes acquiring recommendations.

•Troubleshoot, when the problem occurs in the server.

# Identifying Linux File

As a system administrator you will mostly work with regular files, directories block and character devices.
There is only 1 command you need to know, which will help you to identify and categorize different file types found on the Linux system.

**ls** command will show the file type

$ ls -ld <file name>

## Regular file

The regular file is a most common file type found on the Linux system. It governs all different files such us text files, images, binary files, shared libraries, etc. You can create a regular file with the **touch** command: $ touch linuxcareer.com

## Directory

Directory is second most common file type found in Linux. Directory can be created with the **mkdir** command:

$ mkdir FileTypes
$ ls -ld FileTypes/

## Character device

Character and block device files allow users and programs to communicate with hardware peripheral devices. For example:  $ ls -ld /dev/vmmon
In this case the character device is the vmware module device.

## Block Device

Block devices are similar to character devices. They mostly govern hardware as hard drives, memory, etc.  e,.g. $ ls -ld /dev/sda

## Local domain sockets

Local domain sockets are used for communication between processes. Generally, they are used by services such as X windows, syslog and etc.
$ ls -ld /dev/log

## Configuration and log files

•Linux system administrators often need to look at log files for troubleshooting purposes. In fact, this is the first thing any sysadmin would do.

•Linux and the applications that run on it can generate all different types of messages, which are recorded in various log files.

•Linux uses a set of configuration files, directories, programs, commands and daemons to create, store and recycle these log messages.

•Knowing where the system keeps its log files  help save valuable time during troubleshooting.

**Default Log File Location**

The default location for log files in Linux is /var/log.

You can view the list of log files in this directory with a simple ls -l /var/log command.

```
[root@TestLinux ~]# ls -l /var/log
```

```
total 1472
-rw-------. 1 root root   4524 Nov 15 16:04 anaconda.ifcfg.log
-rw-------. 1 root root  59041 Nov 15 16:04 anaconda.log
-rw-------. 1 root root  42763 Nov 15 16:04 anaconda.program.log
-rw-------. 1 root root 299910 Nov 15 16:04 anaconda.storage.log
-rw-------. 1 root root  40669 Nov 15 16:04 anaconda.syslog
-rw-------. 1 root root  57061 Nov 15 16:04 anaconda.xlog
-rw-------. 1 root root   1829 Nov 15 16:04 anaconda.yum.log
drwxr-x---. 2 root root   4096 Nov 15 16:11 audit
-rw-r--r--  1 root root   2252 Dec  9 10:27 boot.log
-rw-------  1 root utmp    384 Dec  9 10:31 btmp
-rw-------. 1 root utmp   1920 Nov 28 09:28 btmp-20131202
drwxr-xr-x  2 root root   4096 Nov 29 15:47 ConsoleKit
-rw-------  1 root root   2288 Dec  9 11:01 cron
-rw-------. 1 root root   8809 Dec  2 17:09 cron-20131202
-rw-r--r--  1 root root  21510 Dec  9 10:27 dmesg
-rw-r--r--  1 root root  21351 Dec  6 16:37 dmesg.old
-rw-r--r--. 1 root root 165665 Nov 15 16:04 dracut.log
-rw-r--r--. 1 root root 146876 Dec  9 10:44 lastlog
-rw-------  1 root root    950 Dec  9 10:27 maillog
-rw-------. 1 root root   4609 Dec  2 17:00 maillog-20131202
-rw-------  1 root root 123174 Dec  9 10:27 messages
-rw-------. 1 root root 458481 Dec  2 17:00 messages-20131202
-rw-------  1 root root   2644 Dec  9 10:44 secure
-rw-------. 1 root root  15984 Dec  2 17:00 secure-20131202
-rw-------  1 root root      0 Dec  2 17:09 spooler
-rw-------. 1 root root      0 Nov 15 16:02 spooler-20131202
-rw-------. 1 root root      0 Nov 15 16:02 tallylog
-rw-rw-r--. 1 root utmp  89856 Dec  9 10:44 wtmp
-rw-------  1 root root   3778 Dec  6 16:48 yum.log
```

# Viewing Log File Contents

Here are some common log files you will find under /var/log:
wtmp
utmp
dmesg
messages
maillog or mail.log
spooler
auth.log or secure

e.g. The wtmp and utmp files keep track of users logging in and out of the system.

You cannot directly read the contents of these files using cat– there are specific commands for that.

To see who is currently logged in to the Linux server, simply use the who command.
This command gets its values from the /var/run/utmp file (for CentOS and Debian) or /run/utmp (for Ubuntu).

```
[root@TestLinux ~]# who
```

```
root     tty1         2013-12-09 10:44
root      pts/0        2013-12-09 10:29 (10.0.2.2)
sysadmin pts/1        2013-12-09 10:31 (10.0.2.2)
joeblog  pts/2        2013-12-09 10:39 (10.0.2.2)
```

The last command tells us the login history of users:

```
[root@TestLinux ~]# last | grep sysadmin
```

```
sysadmin pts/1        10.0.2.2         Mon Dec  9 10:31   still logged in
sysadmin pts/0        10.0.2.2         Fri Nov 29 15:42 - crash  (00:01)
sysadmin pts/0        10.0.2.2         Thu Nov 28 17:06 - 17:13  (00:06)
sysadmin pts/0        10.0.2.2         Thu Nov 28 16:17 - 17:05  (00:48)
sysadmin pts/0        10.0.2.2         Thu Nov 28 09:29 - crash  (06:04)
sysadmin pts/0        10.0.2.2         Wed Nov 27 16:37 - down   (00:29)
sysadmin tty1                          Wed Nov 27 14:05 - down   (00:36)
sysadmin tty1                          Wed Nov 27 13:49 - 14:04  (00:15)
```

To find out when was the system last rebooted, we can run the following command:

```
[root@TestLinux ~]# last reboot
```

The result may look like this

```
reboot   system boot  2.6.32-358.el6.x Mon Dec  9 10:27 - 10:47  (00:19)
reboot   system boot  2.6.32-358.el6.x Fri Dec  6 16:37 - 10:47 (2+18:10)
reboot   system boot  2.6.32-358.el6.x Fri Dec  6 16:28 - 16:36  (00:08)
reboot   system boot  2.6.32-358.el6.x Mon Dec  2 17:00 - 16:36 (3+23:36)
reboot   system boot  2.6.32-358.el6.x Fri Nov 29 16:01 - 16:36 (7+00:34)
reboot   system boot  2.6.32-358.el6.x Fri Nov 29 15:43 - 16:36 (7+00:53)
...
...
wtmp begins Fri Nov 15 16:11:54 2013
```

To see when did someone last log in to the system, use lastlog:

```
[root@TestLinux ~]# lastlog
```

In my system, the output looked like this:

```
Username              Port            From                    Latest
root                  tty1                                    Mon Dec  9 10:44:36
bin                                      **Never logged in**
daemon                                   **Never logged in**
adm                                      **Never logged in**
lp                                       **Never logged in**
sync                                     **Never logged in**
shutdown                                 **Never logged in**
halt                                     **Never logged in**
mail                                     **Never logged in**
uucp                                     **Never logged in**
operator                                 **Never logged in**
games                                    **Never logged in**
gopher                                   **Never logged in**
ftp                                      **Never logged in**
nobody                                   **Never logged in**
vcsa                                     **Never logged in**
saslauth                                 **Never logged in**
postfix                                  **Never logged in**
sshd                                     **Never logged in**
sysadmin        pts/1    10.0.2.2        Mon Dec  9 10:31:50 +1100 2013
dbus                                     **Never logged in**
joeblog         pts/2    10.0.2.2        Mon Dec  9 10:39:24 +1100 2013
```

# The rsyslog Daemon

- This service is responsible for listening to log messages from different parts of a Linux system and routing the message to an appropriate log file in the /var/log directory.
- It can also forward log messages to another Linux server.
- The rsyslog daemon gets its configuration information from the rsyslog.conf file.
- The file is located under the /etc directory.

# chkconfig command in Linux

•**chkconfig** command is used to list all available services and view or update their run level settings.

•it is used to list current startup information of services or any particular service.

•It updating *runlevel* settings of service and adding or removing service from management.

```
chkconfig --list [name]
chkconfig --add name
chkconfig --del name
chkconfig --override name
chkconfig [--level levels] name
chkconfig [--level levels] name
```

**To List current status of all system services.**

$chkconfig --list

```
[root@localhost ~]# chkconfig --list

Note: This output shows SysV services only and does not include native
      systemd services. SysV configuration data might be overridden by native
      systemd configuration.

      If you want to list systemd services use 'systemctl list-unit-files'.
      To see services enabled on particular target use
      'systemctl list-dependencies [target]'.

netconsole      0:off   1:off   2:off   3:off   4:off   5:off   6:off
network         0:off   1:off   2:on    3:on    4:on    5:on    6:off
rhnsd           0:off   1:off   2:on    3:on    4:on    5:on    6:off
[root@localhost ~]#
```

**To View current status of a particular services. $**chkconfig --list rhnsd

**To Delete a Service - $**chkconfig del rhnsd

**To add a Service - $**chkconfig add rhnsd

**For Disabling a Service -** By default 2 3 4 5 run levels are affected by this command to disable certain run levels only. $ chkconfig rhnsd off

**Enabling a Service:** By default 2 3 4 5 run levels are affected by this command to enable certain run levels only. $ chkconfig rhnsd on

## What is so special about the system administrator account?

•The root account has full (unrestricted) access, so he/she can do anything with system. For example, root can remove critical system files.

•In addition, there is no way you can recover file except using tape backup or disk based backup systems.

•Many tasks for system administration can be automated using Perl/Python or bash shell scripts.

•For example:

1. Create new users.
2. Resetting user passwords.
3. Lock/unlock user accounts.
4. Monitor server security and special services such as DNS, Apache, Nginx, Postfix and more.

**The system administrator is responsible for following things:**

- User administration (setup and maintaining account)
- Maintaining system
- Verify that peripherals are working properly
- Quickly arrange repair for hardware in occasion of hardware failure
- Monitor system performance
- Create file systems
- Install software using tools such as apt command/apt-get command, dnf command/yum command, zypper command, apk command and others.
- Patching firmware and software
- Create a backup and recover policy (disaster recovery [DR])
- Monitor network communication
- Update system as soon as new version of OS and application software comes out
- Implement the policies for the use of the computer system and network

•Setup security policies for users. A sysadmin must have a strong grasp of computer security (e.g. firewalls and intrusion detection systems. You must know how to use tools such as wireshark and nmap command)

•Password and identity management

•Network administration

•Database administration

•How to view and troubleshoot with Unix and Linux log files

•Setting up cron jobs on your Unix and Linux system using the crontab command

# What is SELinux (Security-Enhanced Linux)

•Security-Enhanced Linux (SELinux) is a security architecture for Linux® systems that allows administrators to have more control over who can access the system.

•It was originally developed by the United States National Security Agency (NSA) as a series of patches to the Linux kernel using Linux Security Modules (LSM).

•SELinux was released to the open source community in 2000, and was integrated into the upstream Linux kernel in 2003.

•SELinux defines access controls for the applications, processes, and files on a system.

•It uses security policies, which are a set of rules that tell SELinux what can or can't be accessed, to enforce the access allowed by a policy.

# What Does X Window System Mean?

•The X Window System (X11) is an open source, cross platform, client-server computer software system that provides a GUI in a distributed network environment.

•Used primarily on Unix variants, X versions are also available for other operating systems. Features of the X window system include network transparency, the ability to link to different networks, and customizable graphical capabilities.

•The X window system was first developed in 1984, as part of project Athena, a collaboration between Stanford University and MIT.

•X.Org Foundation, an open group, manages the development and standardization of the X window system.

•The X Window System is also known simply as X, X11 or X Windows.

# configuration apache and mysql in redhat linux

https://youtu.be/qVdlM7mXqWU