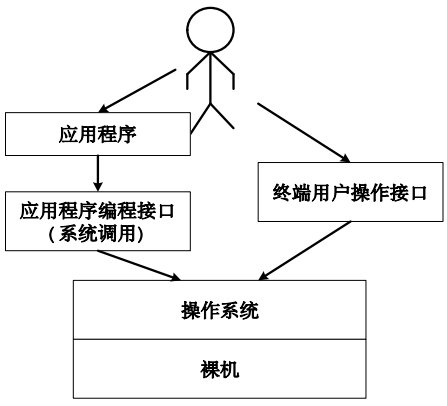
Linux基础知识介绍

Fundamental Knowledge of Linux

Operating System

主要功能：

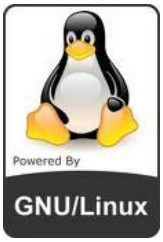
1. CPU管理
2. 存储管理
3. 设备管理
4. 文件管理
5. 网络与通信管理
6. 用户接口

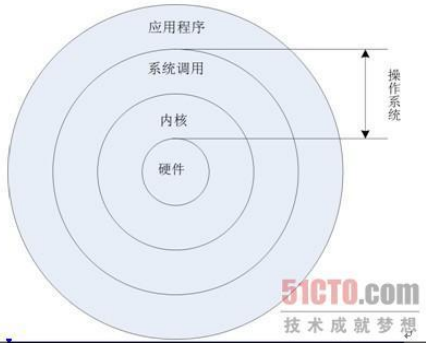


POSIX表示可移植操作系统接口(Portable Operating System Interface)是为解决应用程序平台移植性提出的一种标准。

What is Linux?

- Linux是一个由几百万行源代码组成的庞大、复杂的程序，任何人都能从www.kernel.org上下载。
- Linux是一套免费的、源代码开放的、符合POSIX标准规范的类Unix操作系统 。
- 严格来说，Linux只包含下图中内核与系统调用接口那两层。

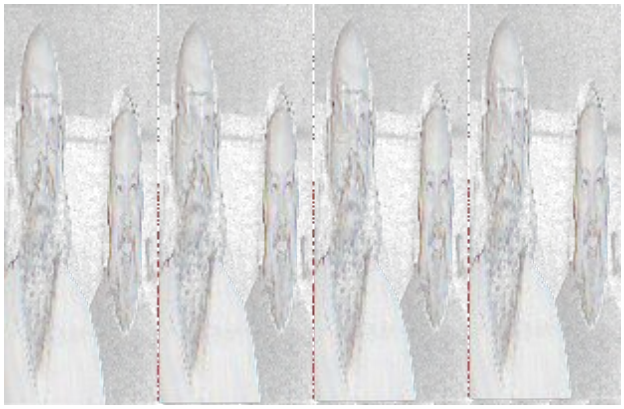




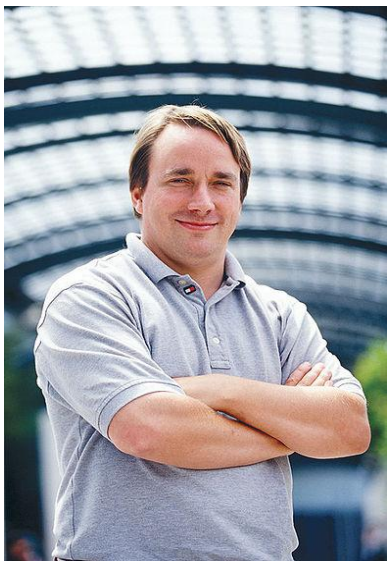
History of Linux

- 1973： Thompson、Ritchie等写出第一个正式UNIX内核
- 1984： Richard Stallman的GNU项目与FSF基金会
- 1985： Richard Stallman撰写了GNU GPL。
- 1991： 芬兰赫尔辛基大学21岁的Linus发布了第一版的Linux内核。
- 1992： 在GNU GPL下Linux内核被重新授权
- 1996： Linux2.0版内核发布
- 1999： Linux2.2版内核发布
- 2001： Linux2.4版内核发布
- 2003： Linux2.6版内核发布
- 至今： RedHat、Fedora、Ubuntu、Debian、Slackware、OpenSUSE、Knoppix等众多发行版

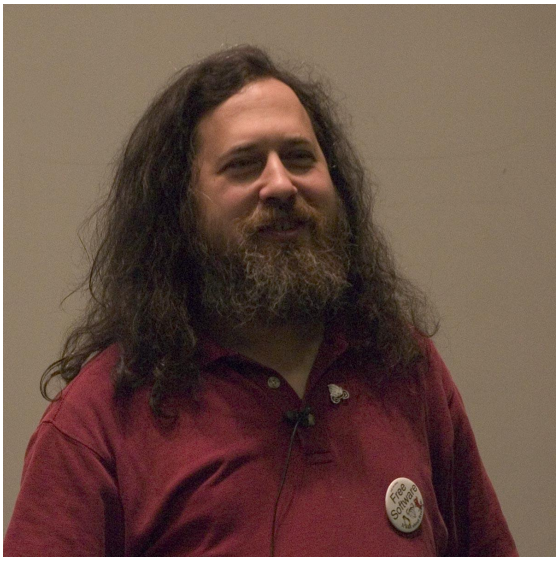
Dennis Ritchie and Ken Thompson



Linus Torvalds



Richard Stallman



GPL License

GPL（GNU General Public License）规定了软件使用自由度的下限。一个软件挂上了GPL版权声明之后，它自然就成了自由软件，具有如下特性：

1. 在发行软件时必须同时发布软件的源码
2. 复制：可以自由复制该软件
3. 修改：可以将获取的源码进行修改，使之适合自己的工作
4. 再发行：您可以将修改过的程序再度自由发行
5. 回馈：您应该将您修改过的程序代码回馈于社会
6. 不能修改授权：一个GPL授权的自由软件，在您修改后，不能取消GPL授权（传染性）
7. 不能单纯销售：您不能单纯销售自由软件

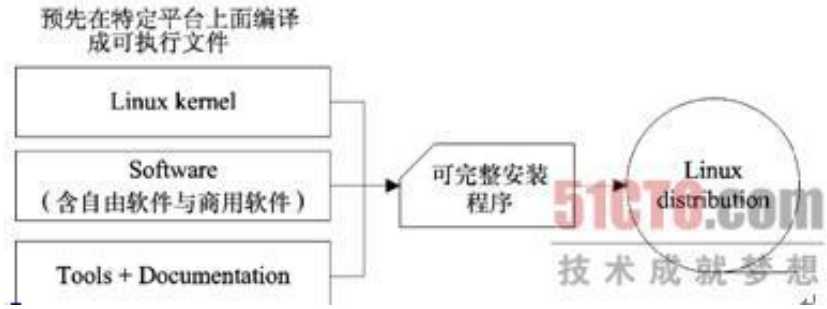
Linux Kernel Versions

- 版本号构成：**主版本.次版本.释出版本-修改版本**
- 最新的内核版本：stable：4.15.7 2018-02-28
- mainline:4.16-rc4 2018-03-04
- 次版本为奇数：开发中版本（development）
- 次版本为偶数：稳定版本（stable）
- https://www.kernel.org/

Linux Distributions

- www.kernel.org上的Linux是源代码，要将这些源码移植到特定的硬件平台，对于普通用户来讲太“高深了”
- www.kernel.org上的Linux仅具有Kernel与Kernel提供的工具，“OS + 各种应用软件”才能构成一个完整的能够被普通用户接受的操作系统。
- 为了能让普通用户接触到Linux，很多的商业公司或非营利团体就将Linux Kernel与可运行的软件集成起来，加上自己具有创意的工具程序，这个工具程序可以让用户以光盘、DVD或者通过网络直接安装。这个"Kernel + Softwares + Tools"的可完全安装的系统，我们称之为Linux distribution，一般中文翻译成可完全安装套件
- 为了让所有的Linux distributions开发不至于差异太大，规定了Linux Standard Base（LSB）和目录架构的File system Hierarchy Standard（FHS）标准规范来规范开发者

Linux Distributions



主要的Linux Distributions

- Red Hat: http://www.redhat.com
- Fedora: http://fedoraproject.org/
- Mandriva: http://www.mandriva.com
- Novell SuSE: http://www.novell.com/linux/
- Debian: http://www.debian.org/
- Slackware: http://www.slackware.com/
- Gentoo: http://www.gentoo.org/
- Ubuntu: http://www.ubuntu.com/
- CentOS: http://www.centos.org/
- KNOPPIX: http://www.knoppix.net/
- Raspbian: http://www.raspbian.org/ 专门针对树莓派的发行版

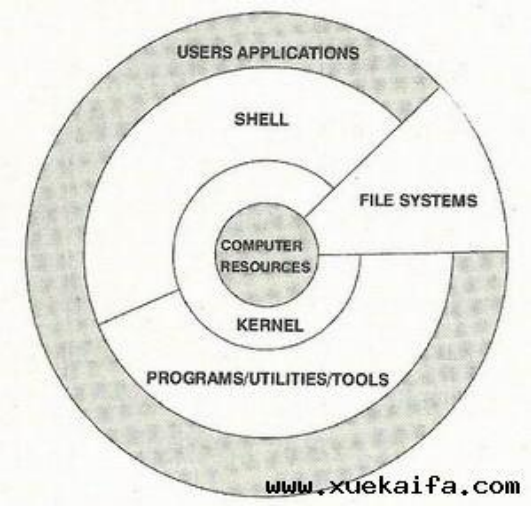
Advantages of Linux

1. 稳定的系统
2. 免费或少许费用
3. 开源社区的广泛支持
4. 安全性、漏洞的快速修补
5. 可以移植到多种硬件平台
6. 多任务、多用户的支持
7. 相对比较不耗资源的系统、性能优异
8. 内核容易裁剪、定制，适合嵌入式系统
9. 良好的网络支持

Applications of Linux

1. 桌面应用： 网络浏览、文字处理、图片编辑、电子邮件、OpenOffice等。
2. 服务器应用： Linux的应用主要集中于服务器市场,linux服务器端的应用软件主要集中在Web服务、邮件系统、文件传输系统和数据库等基础软件。
3. 工作站应用： 数值仿真、数据处理
4. 嵌入式系统： 手机、PDA、机顶盒。
5. 集群计算机： 集群计算机是一种计算机系统,它通过软件和硬件把多台计算机以特殊的方式连接起来,协作完成制定的任务。

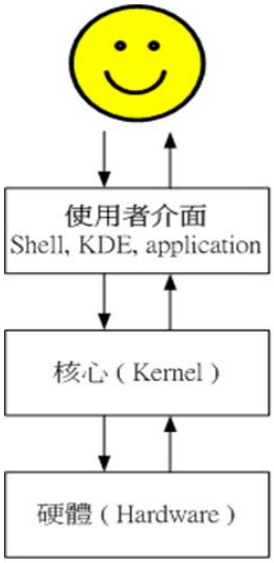
Components of Linux



Linux Shell

- Linux Shell：Shell是系统的用户界面，提供用户与内核进行交互操作的一种接口。
- Shell是一个命令解释器，拥有自己内建的命令集，它解释由用户输入的命令并且把它们送到内核执行。
- Shell也能被系统中其他的应用程序调用。
- Shell支持一种解释型的程序设计语言(Shell Script)，该语言中支持高级语言中所能见到的绝大多数程序控制结构，如循环、函数、变量和数组。
- 目前常见的Shell有Bourne Shell（sh）、Korn Shell（ksh）、C Shell（csh）、Bourne-again Shell（bash）。

Linux Shell



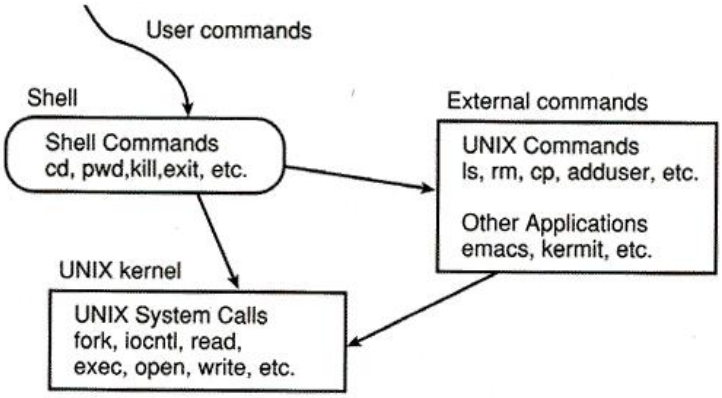
用户利用键盘将命令输入到Shell

如果是内部命令，Shell负责执行相应的动作。

如果是外部命令，Shell会在默认的路径下查找同名的应用程序，然后执行该应用程序。

Linux Shell

内部命令，外部命令，和 系统调用



System Call and API

Bash Shell

- 命令记忆功能。（上下箭头）
- 命令与文件名称自动补全（Tab）
- 命令别名设置
  - alias lm='ls –al'
- 作业控制，前台、后台控制
- Shell变量、环境变量
  - echo、export、set、unset
  - 配置文件：/etc/profile 、~/.bash_profile
- Shell Script 命令脚本

Shell Script

```
[root@www ~]# mkdir scripts; cd scripts
[root@www scripts]# vi sh01.sh
#!/bin/bash
# Program:
#       This program shows "Hello World!" in your screen.
# History:
# 2005/08/23    VBird   First release
PATH=/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin:~/bin
export PATH
echo -e "Hello World! \a \n"
exit 0
```

```
[root@www scripts]# sh sh01.sh
Hello World !
```

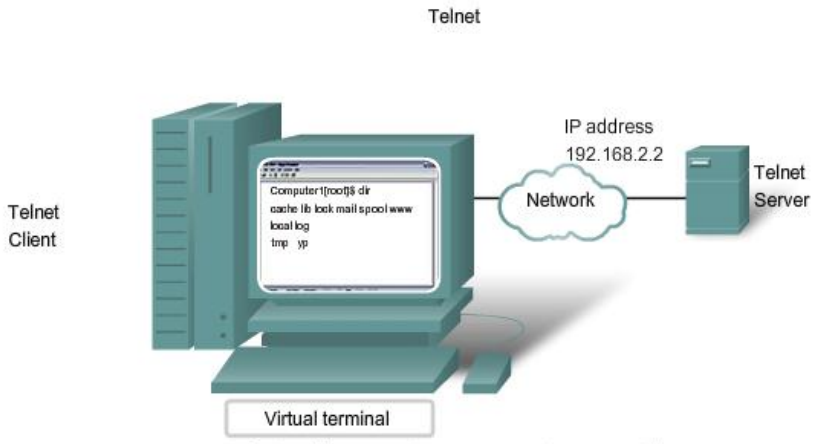
Shell Script

```
[root@www scripts]# vi sh09.sh
#!/bin/bash
# Program:
#       Check $1 is equal to "hello"
# History:
# 2005/08/28    VBird   First release
PATH=/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin:~/bin
export PATH

if [ "$1" == "hello" ]; then
        echo "Hello, how are you ?"
elif [ "$1" == "" ]; then
        echo "You MUST input parameters, ex> {$0 someword}"
else
        echo "The only parameter is 'hello', ex> {$0 hello}"
fi
```

Shell Script已经发展成一门编程语言。

Telnet



Telnet provides a way to use a computer, connected via the network, to access a network device as if the keyboard and monitor were directly connected to the device.

Telnet

1. 用户利用Telnet客户端连接到服务器端。

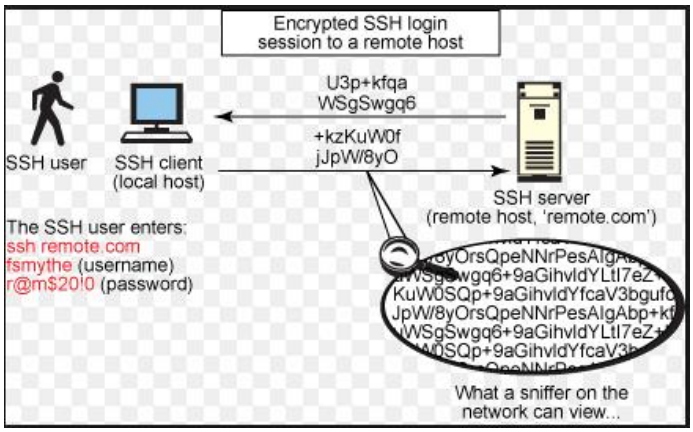
2. 用户在Telnet程序中输入命令，这些命令被传送到服务器上运行，就像直接在服务器的控制台上输入一样。

3. 命令执行结果被回传到客户端。

4. Telnet是常用的远程控制服务器的方法。

SSH: Secure Shell

SSH是Telnet的升级版，客户端与服务器之间的通信内容都被加密。



APP Programs for Linux

- Linux应用程序：标准的Linux系统都有一套称为应用程序的程序集，包括文本编辑器（vim）、GCC编译工具链、X Window、办公套件、Internet工具、数据库等。当然，还可以有用户自己编写的具有特定功能的应用程序。

File System of Linux

- Linux文件系统：文件系统是数据存放在磁盘等存储设备上的组织方法。通常将数据组成文件，多个文件组织成目录，目录再按层次方式进行组织。每个目录可以包括多个子目录以及文件，系统以“/”为根目录。
- Linux文件系统与Windows文件系统的在逻辑概念上的区别：
  - 树形结构 VS 森林结构
  - 磁盘分区与目录树的关系不同

图1 Linux文件系统

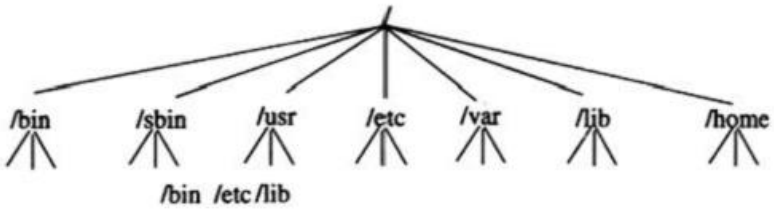
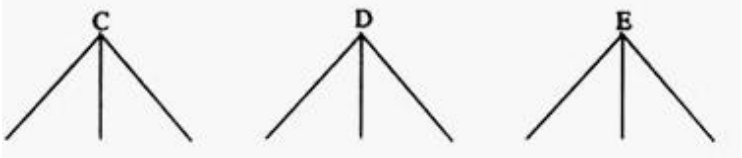
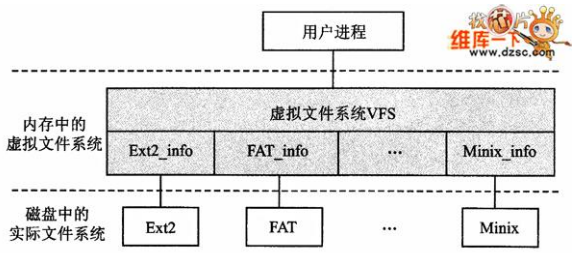


图2 DOS文件系统



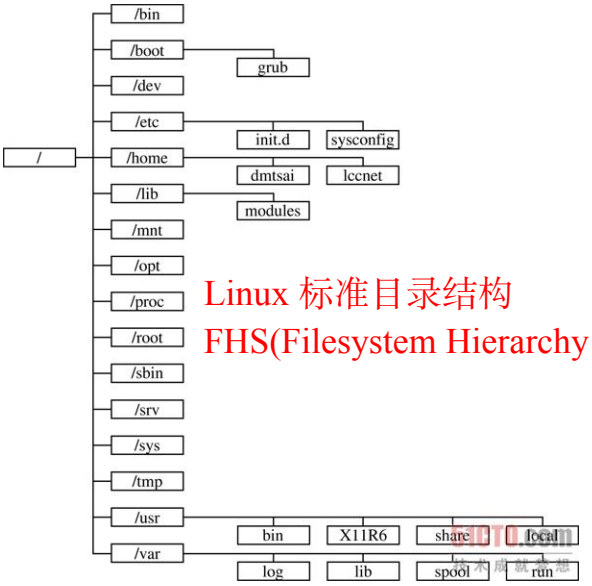
Linux 的虚拟文件系统

- Linux虚拟文件系统：虚拟文件系统VFS是一个统一的、抽象的、虚拟的文件操作系统。可以认为它是应用层与驱动层之间的一个中间层，对上提供一组标准的接口open/close/read/write/lseek，对下则又根据不同的文件系统类型调用不同的驱动程序提供的接口完成对具体设备的操作。
- 不同的存储器有不同的特点，因此有不同的数据读写、组织和索引方式，也就对应不同的文件系统类型



常用的Linux文件系统类型

- 磁盘文件系统类型：
  - Ext2、Ext3、Ext4、ReiserFS、Swap、Vfat、NTFS
  - 区别：访问速度、空间利用率、可靠性
- 光盘文件系统类型：ISO9660
- Flash存储器文件系统类型：
  - NorFlash：JFFS、JFFS2
  - NandFlash: YAFFS和YAFFS2
  - CramFS:专门针对闪存设计的只读压缩的文件系统
- 内存文件系统类型： RamFS、TmpFS
- 网络文件系统：NFS、SMB
- 伪文件系统：proc、sys两个特殊的子目录



Linux 标准目录结构

FHS(Filesystem Hierarchy Standard)

磁盘分区

- **磁盘原理：柱面、磁头、磁道、扇区**
- **主引导记录(Main Boot Record)**
- **主分区、扩展分区、逻辑分区**
- **扩展引导记录(Extended Boot Record)**
- **Linux为每一个设备在dev子目录下都创建一个设备文件，**应用程序对设备文件进行读写就相当于对设备进行读写。
- **硬盘设备文件命名方式：hda1、hda2、sda1…**

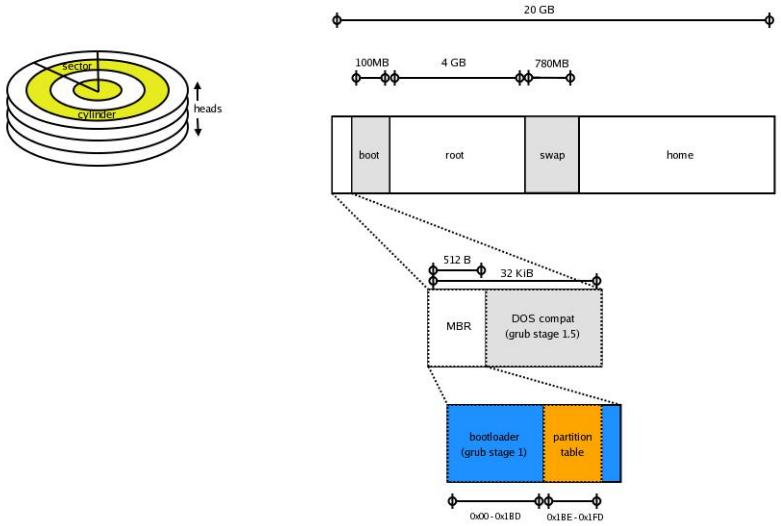
hd：IDE硬盘

sd：SATA、SCSI、USB硬盘

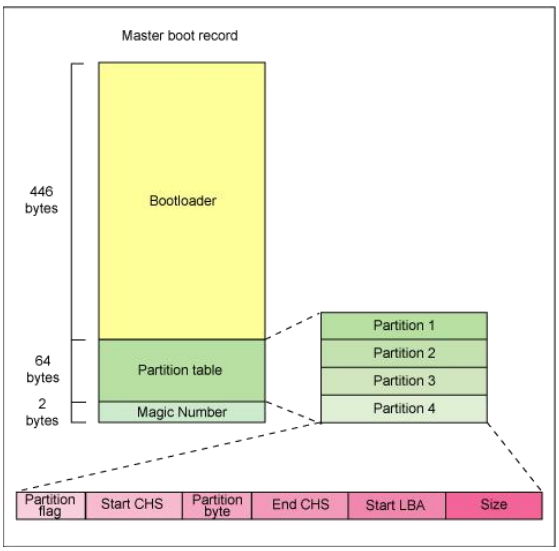
a: 第一块硬盘。如果是第二块硬盘，则为b，依此类推c,d.....

1: 主分区。其中1，2，3，4都是主分区，从第5开始为逻辑分区，最大到16。

硬盘的主引导记录(MBR)



硬盘的主引导记录(MBR)



硬盘的主引导记录(MBR)

标准 MBR 结构

地址			描述		长度（字节）
Hex	Oct	Dec			
0000	0000	0	代码区		**440** (最大 446)
01B8	0670	440	选用磁盘标志		4
01BC	0674	444	一般为空值; 0x0000		2
01BE	0676	446	**标准 MBR 分区表规划** (四个16 byte的主分区表入口)		**64**
01FE	0776	510	55h	MBR 有效标志: 0x55AA	**2**
01FF	0777	511	AAh		
MBR, 总大小: 446 + 64 + 2 =					**512**

主引导记录格式

Structure of a classical generic MBR

Address		Description		Size in bytes
Hex	Dec			
+000h	+0	Bootstrap code area		446
+1BEh	+446	Partition entry #1	*Partition table* (for primary partitions)	16
+1CEh	+462	Partition entry #2		16
+1DEh	+478	Partition entry #3		16
+1EEh	+494	Partition entry #4		16
+1FEh	+510	55h	*Boot signature*[nb 1]	2
+1FFh	+511	AAh		
Total size: 446 + 4*16 + 2				**512**

硬盘的主引导记录(MBR)

硬盘分区结构信息

偏移	长度(字节)	意义
00H	1	分区状态：00-->非活动分区；80--> 活动分区； 其它数值没有意义
01H	1	分区起始磁头号(HEAD)，用到全部8位
02H	2	分区起始扇区号(SECTOR)，占据02H的位0－5； 该分区的起始磁柱号(CYLINDER)，占据 02H的位6－7和03H的全部8位
04H	1	文件系统标志位
05H	1	分区结束磁头号(HEAD)，用到全部8位
06H	2	分区结束扇区号(SECTOR)，占据06H的位0－5； 该分区的起始磁柱号(CYLINDER)，占据 06H的位6－7和07H的全部8位
08H	4	分区起始相对扇区号
0CH	4	分区总的扇区数

扩展引导记录

Common Structure of Extended Boot Records:

Offsets within EBR sectors		Contents	Size
Hex	Dec		bytes
000 – 1BD	000 – 445	Generally unused; normally filled with zeroes; may contain another boot loader i.e. a partition boot record, for example in conjunction with Advanced Active Partitions	446
1BE – 1CD	446 – 461	Partition table's **first entry**	16
1CE – 1DD	462 – 477	Partition table's **second entry**	16
1DE – 1ED	478 – 493	Unused[3] **third entry** filled with zeroes	16
1EE – 1FD	494 – 509	Unused[3] **fourth entry** filled with zeroes	16
1FE – 1FF	510 - 511	Signature 55AAh in big-endian network order, same as little-endian 0xAA55. On disk: 0x55 at offset 510 and 0xAA at offset 511.	2
		EBR, total size: 446 +(4×16) +2 =	512

将磁盘分区挂接到子目录上

- 例1：

  # mkdir /mnt/WinF

  # mount -t ntfs /dev/hda7 /mnt/WinF

- 例2：

  # mkdir /mnt/usb

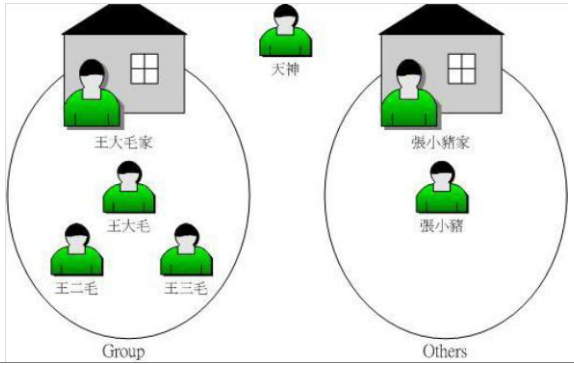
  # mount –t vfat /dev/sdb1 /mnt/usb

  常用命令格式

  mount -t type device dir

Linux 的用户和用户组

- Linux是一个多用户、多任务操作系统，可以同时为多个用户服务。
- 每个用户都属于一个或多个用户组。
- 所用用户的账号都放在/etc/passwd中，对应的密码都放在/etc/shadow中，用户组信息都放在/etc/group中



Linux 的用户管理命令

- 增加用户： **useradd** [选项] 用户名

  例1：#useradd –d /usr/sam -m sam

  例2：#useradd -g group1 –G adm,root user5

- 删除用户： **userdel [选项] 用户名**
- 设置用户口令： **passwd** [选项] 用户名
- 增加用户组： **groupadd** [选项] 用户组
- 切换当前登录用户：

  su [选项参数] [用户]

  例子： su -l root

Linux 的文件权限

- 文件和目录的3种属性：r、w、x
- 文件所有者的权限
- 同用户组的权限
- 其他非本用户组的权限

```
[root@www ~]# ls -al
total 156
drwxr-x---  4  root  root  4096  Sep  8 14:06 .
drwxr-xr-x 23  root  root  4096  Sep  8 14:21 ..
-rw-------  1  root  root  1474  Sep  4 18:27 anaconda-ks.cfg
-rw-------  1  root  root   199  Sep  8 17:14 .bash_history
-rw-r--r--  1  root  root    24  Jan  6  2007 .bash_logout
-rw-r--r--  1  root  root   191  Jan  6  2007 .bash_profile
-rw-r--r--  1  root  root   176  Jan  6  2007 .bashrc
-rw-r--r--  1  root  root   100  Jan  6  2007 .cshrc
drwx------  3  root  root  4096  Sep  5 10:37 .gconf     <=范例说明处
```

Linux 的文件权限

Linux 的文件类型

1. 普通文件：数据文件、文本文件、可执行文件。
2. 目录文件
3. 设备文件：字符设备、块设备
4. 符号链接文件
5. 管道设备文件
6. 套接口文件

一个设计原则：一切皆文件

Linux 的sudo命令

- sudo命令可让普通用户以root管理员身份来执行预先指定的特权命令。
- 用户使用sudo时，必须先输入密码(当前用户的，而非root用户的密码)，之后有5分钟的有效期限，超过期限则必须重新输入密码。
- 用法：sudo [-u <用户>] [命令]
- 在/etc/sudoers中设置了可执行sudo的用户组列表，以及这些用户可以执行的特权指令和可以获得的临时权限。
- /etc/sudoers文件的例子：

  userGroupA ALL=/bin/more

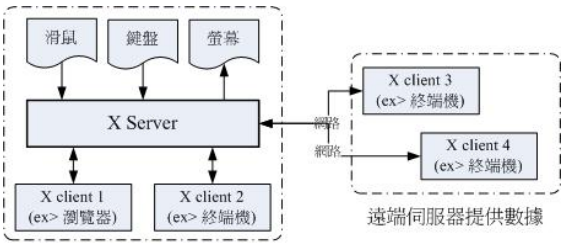
  userGroupB ALL=/usr/sbin/*, /sbin/*, !/usr/sbin/fdisk

守候进程daemon、服务

- 守候进程是一种常驻内存的进程，用来提供一些系统或网络服务，有时也被称为后台服务。
- 守候进程独立于任何控制终端，并且周期性地执行某种任务或等待处理某些发生的事件。
- 守护进程一般在系统启动时开始运行(设置在/etc/rc.d或/etc/init.d配置文件中)，除非强行终止，否则保持运行直到系统关机。
- 守护进程经常以超级用户（root）权限运行，因为它们要使用特殊的端口（1-1024）或访问某些特殊的资源。
- 守护进程的名称通常以d结尾，比如httpd、ftpd、sshd、syslogd、mysqld、lqd、xinetd等
- Linux守护进程又可分为独立启动，和按需启动两种类型。

X Window系统

- X Windows系统是Linux上的一个应用程序，提供GUI操作接口
- 为什么叫X？在开发X之前就已经在另外一个视窗系统上工作了。那个系统的名字叫做“W”(“Window”)。X是W的后一个字母。
- X Window系统的构架：X Server/Client
- 两类窗口管理器、窗口风格：KDE和GNOME



用戶端的 X 架構

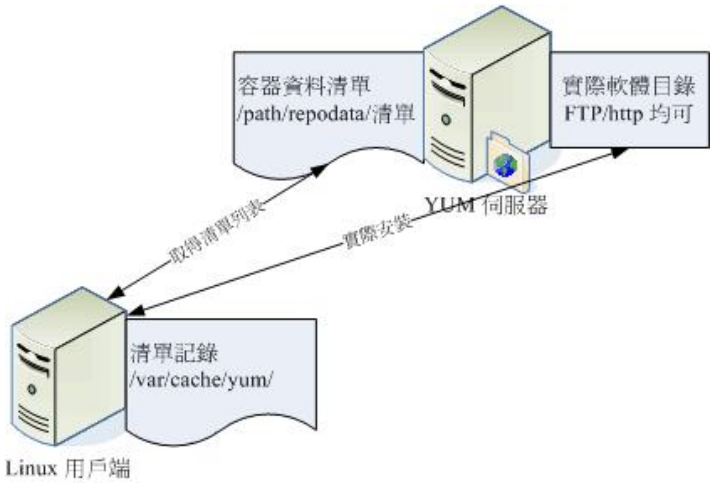
Linux下的软件安装与升级

1. 基于源码与Tarball的安装：(难度大)
   - 取得源文件包，并利用tarball工具解压
   - 运行configure检测装载环境生成正确的Makefile文件
   - make clean 清除已有的编译结果
   - make 编译整个工程
   - make install 将编译结构安装的当前计算机
   - 利用patch更新源码，并重复上述过程
   - 难点：操作系统版本问题、编译器版本问题、函数库路径/版本问题

Linux下的软件安装与升级

2. 基于RPM的安装、基于yum的在线升级
   - RedHat Package Manager
   - 被Redhat, Fedora, CentOS, SuSE采用
   - 安装：rpm –i rp-pppoe-3.5-32.1.i386.rpm
   - 删除：rpm – e appname
   - 可以rpm命令查询、升级和删除本计算机上已经安装的应用程序。
   - 利用yum可以实现在线安装、升级和删除应用程序。
   - 安装：yum install bochs
   - 本地安装：yum localinstall ur.rpm
   - 升级：yum update bochs
   - 删除：yum erase bochs / yum remove bochs

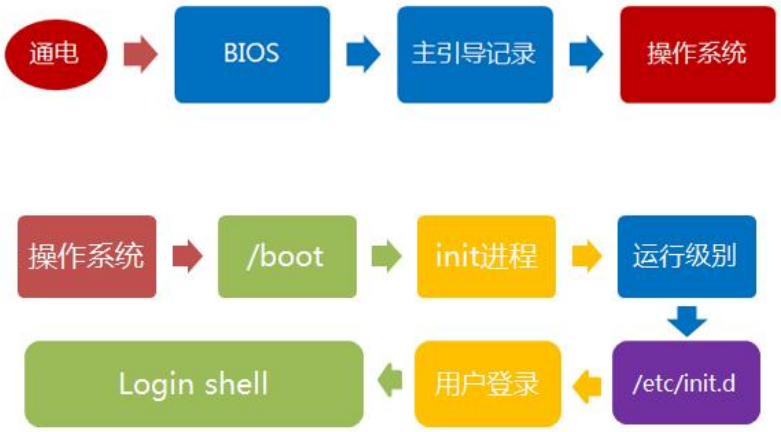
Linux下的软件安装与升级



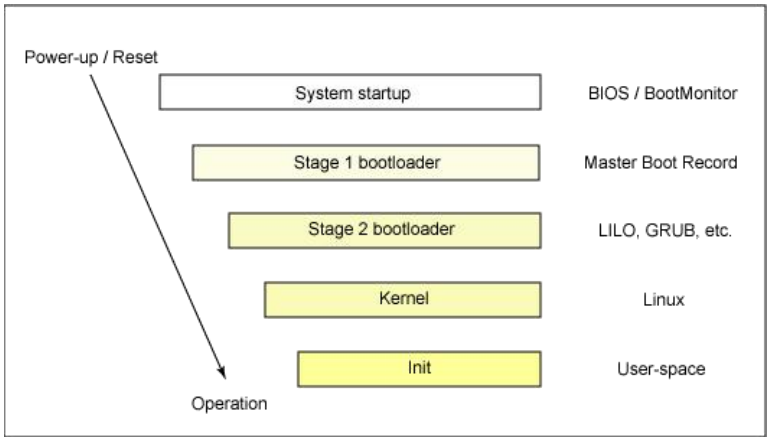
Linux下的软件安装与升级

3. 基于DPKG的安装、基于apt-get的在线升级
   - 被 Debian, Ubuntu采用
   - 安装：dpkg -i package.deb
   - 删除：dpkg -r package
   - 查询：dpkg -l package
   - 利用apt-get可以实现在线安装、升级和删除应用程序
   - 安装：apt-get install packagename
   - 升级：apt-get update packagename
   - 删除：apt-get remove packagename

Linux的启动流程(PC机)



Linux的启动流程



Linux的启动流程(PC机)

- 加电后最先执行的是主板上的BIOS
- BIOS负责加载主引导记录中的引导程序
- 再加载GRUB(统一多重引导管理器)引导程序
- GRUB负责加载/boot子目录下的Linux内核文件
- Linux内核文件加载运行以后，就开始运行第一个应用程序/sbin/init，它的作用是初始化系统环境。
- init进程的一大任务就是运行一些开机启动的守候进程或服务。但是，不同的应用场合需要启动不同的程序，因此Linux定义了“运行级别”(Run Level)的概念。init根据"运行级别"，确定要运行哪些程序。
- /etc/inittab文件中规定了默认运行级别。
- /etc/rc0.d、/etc/rc1.d、/etc/rc2.d …子目录中的配置文件分别定义了不同运行级别下的运行脚本

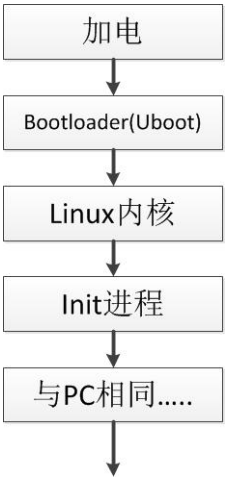
Linux的运行级别

	级别符号	RHEL 阵营	Ubuntu 阵营
1	0	停机（从其他级别切换到 0 级别时表示关机）	
2	1	单用户模式（或称维护模式），运行 rc.sysinit 和 rc1.d 目录下的脚本	
3	2	多用户字符模式（无网络）	多用户图形模式（有网络），默认级别是 2
4	3	多用户字符模式（有网络）	
5	4	用户自定义级别	
6	5	多用户图形模式（有网络）（默认）	
7	6	重启	
8	S	单用户模式（或称维护模式），只运行 rc.sysinit 脚本	
9	s		

启动过程中重要的配置文件

- /etc/grub.conf
- /etc/inittab
- /etc/sysconfig/modules
- /etc/init.d
- /etc/rc0.d、 /etc/rc1.d、 /etc/rc2.d …
- /etc/profile
- ~/.bash_profile
- ~/.profile

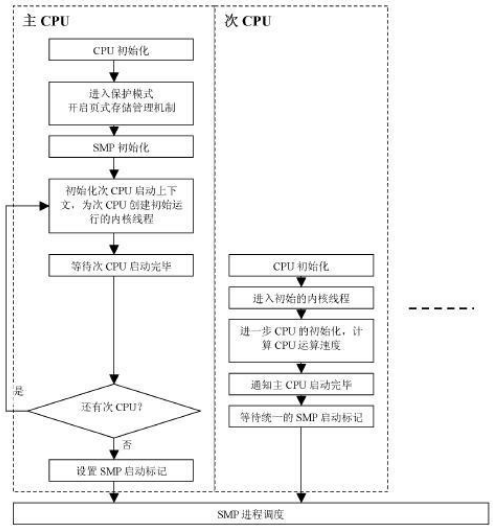
嵌入式Linux启动流程



Linux内核的特性

1. 真正的多用户、多任务管理
2. 完善的虚拟内存管理和运行保护机制
3. 虚拟文件系统
4. 支持POSIX标准的系统调用
5. 动态内核模块加载
6. 网络功能
7. 支持SMP（Symmetric MultiProcessor）
8. 可移植性

SMP



Linux2.6内核的新特性

1. 可抢占式内核
2. 完全公平调度算法
3. 统一设备模型
4. PnP支持
5. 内核模块改变
6. 线程模型
7. …