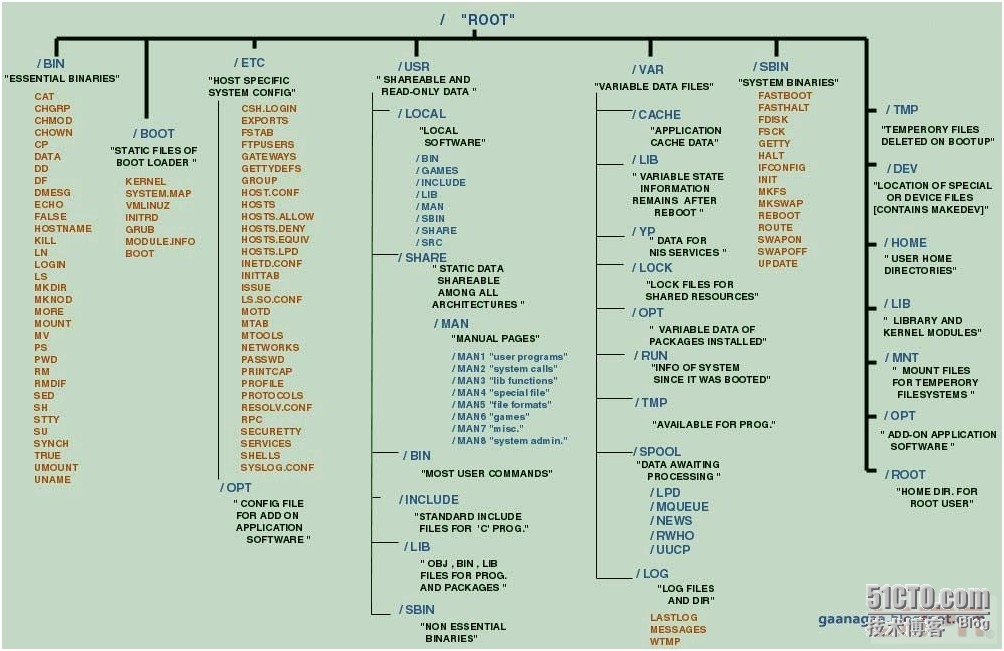
**linux的目录结构：**

**1、树状目录结构图**

**、树状目录结构图**

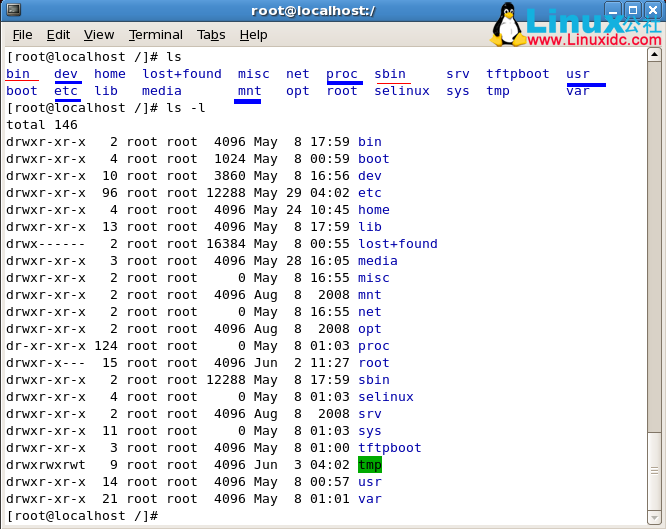
[](http://img1.51cto.com/attachment/201309/230817424.jpg" \t "_blank)

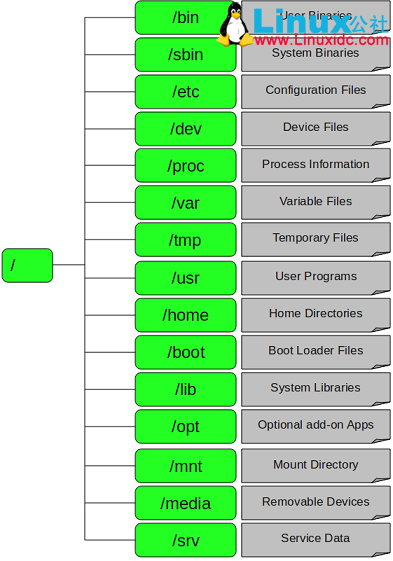
**让我们回顾一下Linux的文件系统结构，并理解各上层目录的含义。**

**蓝色：表示目录**

**青色：表示链接**

**黑色：表示文件**





**因为根目录与开机有关，开机过程中仅有根目录会被挂载， 其他分区则是在开机完成之后才会持续的进行挂载的行为。就是因为如此，因此根目录下与开机过程有关的目录， 就不能够与根目录放到不同的分区去。那哪些目录不可与根目录分开呢？有底下这些：**

**/etc：配置文件**

**/bin：重要执行命令**

**/dev：所需要的装置文件**

**/lib：执行档所需的函式库与核心所需的模块**

**/sbin：重要的系统执行文件**

**这五个目录千万不可与根目录分开在不同的分区。请背下来啊。**

**Linux 登录后，配置文件执行顺序为(Debian Serials Capable)：/etc/environment -> /etc/profile -> (~/.bash\_profile | ~/.bash\_login | ~/.profile) -> ~/.bashrc -> /etc/bashrc -> ~/.bash\_logout**

**可用ls –a 查看隐藏的文件，一般可看到/etc/profile , ~/.profile, ~/.bashrc,/etc/bashrc**

**关于各个文件的作用，在LinuxSir上找到了以下说明：**

**（1）/etc/environment：此配置文件设置基本的PATH变量，及系统当前语言变量，虽然比较短，但却在系统启动中占据举足轻重的作用，比如如下是我的系统中的内容：**

**PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games"**

**LANGUAGE="zh\_CN:zh:en\_US:en"**

**LANG="zh\_CN.UTF-8"**

**（2）/etc/profile： 此文件为系统的每个用户设置环境信息,当用户第一次登录时,该文件被执行. 并从/etc/profile.d目录的配置文件中搜集shell的设置。**

**（3）/etc/bashrc: 为每一个运行bash shell的用户执行此文件.当bash shell被打开时,该文件被读取。此配置文件可体现用户自己的特色，如给一些常用的命令取一个别名等。如 alias cp=’cp –i’**

**Alias rm=’rm –i’**

**（4）~/.bash\_profile: 每个用户都可使用该文件输入专用于自己使用的shell信息,当用户登录时,该文件仅仅执行一次!默认情况下,他设置一些环境变量,执行用户的.bashrc文件。**

**（5）~/.bashrc: 该文件包含专用于你的bash shell的bash信息,当登录时以及每次打开新的shell时,该该文件被读取。**

**（6）~/.bash\_logout: 当每次退出系统(退出bash shell)时,执行该文件. 另外,/etc/profile中设定的变量(全局)的可以作用于任何用户,而~/.bashrc等中设定的变量(局部)只能继承 /etc/profile中的变量,他们是"父子"关系。**

**（7）~/.bash\_profile 是交互式、login 方式进入 bash 运行的~/.bashrc 是交互式 non-login 方式进入 bash 运行的通常二者设置大致相同，所以通常前者会调用后者。**

1、**/**- 根  
每一个文件和目录从根目录开始。  
只有root用户具有该目录下的写权限。请注意，/root是root用户的主目录，这与/.不一样

2、**/bin**中 - 用户二进制文件  
包含二进制可执行文件。  
在单用户模式下，你需要使用的常见Linux命令都位于此目录下。系统的所有用户使用的命令都设在这里。  
例如：ps、ls、ping、grep、cp

3、**/sbin**目录 - 系统二进制文件  
就像/bin，/sbin同样也包含二进制可执行文件。  
但是，在这个目录下的linux命令通常由系统管理员使用，对系统进行维护。例如：iptables、reboot、fdisk、ifconfig、swapon命令

4、**/etc** - 配置文件  
包含所有程序所需的配置文件。  
也包含了用于启动/停止单个程序的启动和关闭shell脚本。例如：/etc/resolv.conf、/etc/logrotate.conf

9、/usr - 用户程序

包含二进制文件、库文件、文档和二级程序的源代码。

/usr/bin中包含用户程序的二进制文件。如果你在/bin中找不到用户二进制文件，到/usr/bin目录看看。例如：at、awk、cc、less、scp。

/usr/sbin中包含系统管理员的二进制文件。如果你在/sbin中找不到系统二进制文件，到/usr/sbin目录看看。例如：atd、cron、sshd、useradd、userdel。

/usr/lib中包含了/usr/bin和/usr/sbin用到的库。

/usr/local中包含了从源安装的用户程序。例如，当你从源安装Apache，它会在/usr/local/apache2中。

依据FHS的基本定义，/usr里面放置的数据属于可分享的与不可变动的(shareable, static)， /usr不是user的缩写，其实usr是Unix Software Resource的缩写， 也就是Unix操作系统软件资源所放置的目录，而不是用户的数据啦。这点要注意。 FHS建议所有软件开发者，应该将他们的数据合理的分别放置到这个目录下的次目录，而不要自行建立该软件自己独立的目录。

因为是所有系统默认的软件(distribution发布者提供的软件)都会放置到/usr底下，因此这个目录有点类似Windows 系统的C:\Windows\ + C:\Program files\这两个目录的综合体，系统刚安装完毕时，这个目录会占用最多的硬盘容量。 一般来说，/usr的次目录建议有底下这些：

|  |  |
| --- | --- |
| 目录 | 应放置文件内容 |
| /usr/X11R6/ | 为X Window System重要数据所放置的目录，之所以取名为X11R6是因为最后的X版本为第11版，且该版的第6次释出之意。 |
| /usr/bin/ | 绝大部分的用户可使用指令都放在这里。请注意到他与/bin的不同之处。(是否与开机过程有关) |
| /usr/include/ | c/c++等程序语言的档头(header)与包含档(include)放置处，当我们以tarball方式 (\*.tar.gz 的方式安装软件)安装某些数据时，会使用到里头的许多包含档。 |
| /usr/lib/ | 包含各应用软件的函式库、目标文件(object file)，以及不被一般使用者惯用的执行档或脚本(script)。 某些软件会提供一些特殊的指令来进行服务器的设定，这些指令也不会经常被系统管理员操作， 那就会被摆放到这个目录下啦。要注意的是，如果你使用的是X86\_64的Linux系统， 那可能会有/usr/lib64/目录产生 |
| /usr/local/ | 统管理员在本机自行安装自己下载的软件(非distribution默认提供者)，建议安装到此目录， 这样会比较便于管理。举例来说，你的distribution提供的软件较旧，你想安装较新的软件但又不想移除旧版， 此时你可以将新版软件安装于/usr/local/目录下，可与原先的旧版软件有分别啦。 你可以自行到/usr/local去看看，该目录下也是具有bin, etc, include, lib...的次目录 |
| /usr/sbin/ | 非系统正常运作所需要的系统指令。最常见的就是某些网络服务器软件的服务指令(daemon) |
| /usr/share/ | 放置共享文件的地方，在这个目录下放置的数据几乎是不分硬件架构均可读取的数据， 因为几乎都是文本文件嘛。在此目录下常见的还有这些次目录：/usr/share/man：联机帮助文件  /usr/share/doc：软件杂项的文件说明  /usr/share/zoneinfo：与时区有关的时区文件 |
| /usr/src/ | 一般原始码建议放置到这里，src有source的意思。至于核心原始码则建议放置到/usr/src/linux/目录下。 |

**五. 绝对路径与相对路径**

除了需要特别注意的FHS目录配置外，在文件名部分我们也要特别注意。因为根据档名写法的不同，也可将所谓的路径(path)定义为绝对路径(absolute)与相对路径(relative)。 这两种文件名/路径的写法依据是这样的：

绝对路径：

由根目录(/)开始写起的文件名或目录名称， 例如 /home/dmtsai/.bashrc；

相对路径：

相对于目前路径的文件名写法。 例如 ./home/dmtsai 或 http://www.cnblogs.com/home/dmtsai/ 等等。反正开头不是 / 就属于相对路径的写法

而你必须要了解，相对路径是以你当前所在路径的相对位置来表示的。举例来说，你目前在 /home 这个目录下， 如果想要进入 /var/log 这个目录时，可以怎么写呢？

复制代码

代码如下:

cd /var/log (absolute)  
cd ../var/log (relative)

因为你在 /home 底下，所以要回到上一层 (../) 之后，才能继续往 /var 来移动的，特别注意这两个特殊的目录：

.  ：代表当前的目录，也可以使用 ./ 来表示；

.. ：代表上一层目录，也可以 ../ 来代表。

这个 . 与 .. 目录概念是很重要的，你常常会看到 cd .. 或 ./command 之类的指令下达方式， 就是代表上一层与目前所在目录的工作状态。

实例1：如何先进入/var/spool/mail/目录，再进入到/var/spool/cron/目录内？

命令：

复制代码

代码如下:

cd /var/spool/mail  
cd ../cron

“man hier”，通过这个命令的输出，就知道“根目录”中所子目录的作用了