1. Linux简介
2. Linux系统启动过程

Linux 系统的启动过程大体上可分为五部分：内核的引导；运行 init；系统初始化；建立终端 ；用户登录系统。

1. 内核引导

当计算机打开电源后，首先是 BIOS 开机自检，按照 BIOS 中设置的启动设备（通常是硬盘）来启动。紧接着由启动设备上的 grub 程序开始引导 linux，当引导程序成功完成引导任务后，Linux 从它们手中接管了 CPU 的控制权，然后 CPU 就开始执行 Linux 的核心映象代码，开始了 Linux 启动过程。也就是所谓的内核引导开始了，在内核引导过程中其实是很复杂的，我们就当它是一个黑匣子，反正是 linux 内核做了一些列工作，最后内核调用加载了 init 程序，至此内核引导的工作就完成了。交给了下一个主角 init。

1. 运行init

init 进程是系统所有进程的起点，你可以把它比拟成系统所有进程的老祖宗，没有这个进程，系统中任何进程都不会启动。init 程序首先是需要读取配置文件 /etc/inittab。inittab 是一个不可执行的文本文件，它有若干行指令所组成。

1. 系统初始化

在 init 的配置文件中有这么一行： si::sysinit:/etc/rc.d/rc.sysinit　它调用执行了/etc/rc.d/rc.sysinit，而 rc.sysinit是一个bash shell的脚本，它主要是完成一些系统初始化的工作，rc.sysinit 是每一个运行级别都要首先运行的重要脚本。它主要完成的工作有：激活交换分区，检查磁盘，加载硬件模块以及其它一些需要优先执行任务。

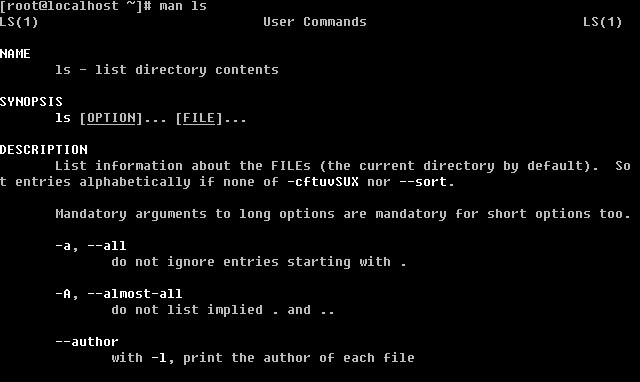
1. 建立终端

rc 执行完毕后，返回 init。这时基本系统环境已经设置好了，各种守护进程也已经启动了。init 接下来会打开 6 个终端，以便用户登录系统。

1. 用户登录系统

对于运行级别为 5 的图形方式用户来说，他们的登录是通过一个图形化的登录界面。登录成功后可以直接进入KDE、Gnome 等窗口管理器。而本文主要讲的还是文本方式登录的情况：当我们看到 mingetty 的登录界面时，我们就可以输入用户名和密码来登录系统了。

1. Linux快捷键
   1. Ctrl + C：这个是用来终止当前命令的快捷键，当然你也可以输入一大串字符，不想让它运行直接 Ctrl + C，光标就会跳入下一行。
   2. Tab： 这个键是最有用的键了，也是笔者敲击概率最高的一个键。因为当你打一个命令打一半时，它会帮你补全的。不光是命令，当你打一个目录时，同样可以补全，不信你试试。
   3. Ctrl + D： 退出当前终端，同样你也可以输入 exit。
   4. Ctrl + Z： 暂停当前进程，比如你正运行一个命令，突然觉得有点问题想暂停一下，就可以使用这个快捷键。暂停后，可以使用 fg 恢复它。
   5. Ctrl + L： 清屏，使光标移动到第一行。
2. 查询帮助文档命令 – man



输入 man ls 其实格式为 man + 命令。从命令的介绍到命令的参数以及用法介绍的都非常详细的。

1. Linux系统目录结构

http://wiki.jikexueyuan.com/project/linux/images/ls.png

以下是对这些目录的解释：

/bin bin 是 Binary 的缩写。这个目录存放着最经常使用的命令。

/boot 这里存放的是启动 Linux 时使用的一些核心文件，包括一些连接文件以及镜像文件。

/dev dev 是 Device(设备)的缩写。该目录下存放的是 Linux 的外部设备，在 Linux 中访问设备的方式和访问文件的方式是相同的。

/etc 这个目录用来存放所有的系统管理所需要的配置文件和子目录。

/home 用户的主目录，在 Linux 中，每个用户都有一个自己的目录，一般该目录名是以用户的账号命名的。

/lib 这个目录里存放着系统最基本的动态连接共享库，其作用类似于 Windows 里的 DLL 文件。几乎所有的应用程序都需要用到这些共享库。

/lost+found 这个目录一般情况下是空的，当系统非法关机后，这里就存放了一些文件。

/media linux 系统会自动识别一些设备，例如U盘、光驱等等，当识别后，linux 会把识别的设备挂载到这个目录下。

/mnt 系统提供该目录是为了让用户临时挂载别的文件系统的，我们可以将光驱挂载在 /mnt/ 上，然后进入该目录就可以查看光驱里的内容了。

/opt 这是给主机额外安装软件所摆放的目录。比如你安装一个 ORACLE 数据库则就可以放到这个目录下。默认是空的。

/proc 这个目录是一个虚拟的目录，它是系统内存的映射，我们可以通过直接访问这个目录来获取系统信息。这个目录的内容不在硬盘上而是在内存里，我们也可以直接修改里面的某些文件，比如可以通过下面的命令来屏蔽主机的 ping 命令，使别人无法 ping 你的机器：

echo 1 > /proc/sys/net/ipv4/icmp\_echo\_ignore\_all。

/root 该目录为系统管理员，也称作超级权限者的用户主目录。

/sbin s 就是 Super User 的意思，这里存放的是系统管理员使用的系统管理程序。

/selinux 这个目录是 Redhat/CentOS 所特有的目录，Selinux 是一个安全机制，类似于 Windows 的防火墙，但是这套机制比较复杂，这个目录就是存放 selinux 相关的文件的。

/srv 该目录存放一些服务启动之后需要提取的数据。

/sys 这是 linux2.6 内核的一个很大的变化。该目录下安装了 2.6 内核中新出现的一个文件系统 sysfs ，sysfs 文件系统集成了下面3种文件系统的信息：针对进程信息的 proc 文件系统、针对设备的 devfs 文件系统以及针对伪终端的 devpts 文件系统。该文件系统是内核设备树的一个直观反映。当一个内核对象被创建的时候，对应的文件和目录也在内核对象子系统种被创建。

/tmp 这个目录是用来存放一些临时文件的。

/usr 这是一个非常重要的目录，用户的很多应用程序和文件都放在这个目录下，类似与 windows 下的 program files目录。

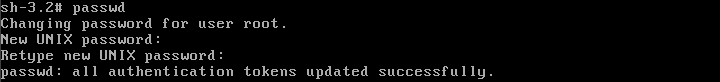
/usr/bin：系统用户使用的应用程序。

/usr/sbin：超级用户使用的比较高级的管理程序和系统守护程序。

/usr/src：内核源代码默认的放置目录。

/var 这个目录中存放着在不断扩充着的东西，我们习惯将那些经常被修改的目录放在这个目录下。包括各种日志文件。

1. 在 linux 系统中，有几个目录是比较重要的，平时需要注意不要误删除或者随意更改内部文件。/etc： 上边也提到了，这个是系统中的配置文件，如果你更改了该目录下的某个文件可能会导致系统不能启动。/bin, /sbin, /usr/bin, /usr/sbin: 这是系统预设的执行文件的放置目录，比如 ls 就是在 /bin/ls 目录下的。值得提出的是，/bin, /usr/bin 是给系统用户使用的指令（除 root 外的通用户），而 /sbin, /usr/sbin 则是给 root 使用的指令。 /var： 这是一个非常重要的目录，系统上跑了很多程序，那么每个程序都会有相应的日志产生，而这些日志就被记录到这个目录下，具体在 /var/log 目录下，另外 mail 的预设放置也是在这里。
2. 正确关机Linux系统
3. 关机流程为：sysnc ? shutdown ? reboot ? halt
   1. sync 将数据由内存同步到硬盘中。
   2. shutdown 关机指令，你可以man shutdown 来看一下帮助文档。例如你可以运行如下命令关机：
   3. shutdown –h 10 ‘This server will shutdown after 10 mins’ 这个命令告诉大家，计算机将在10分钟后关机，并且会显示在登陆用户的当前屏幕中。
   4. Shutdown –h now 立马关机
   5. Shutdown –h 20:25 系统会在今天20:25关机
   6. Shutdown –h +10 十分钟后关机
   7. Shutdown –r now 系统立马重启
   8. Shutdown –r +10 系统十分钟后重启
   9. reboot 就是重启，等同于 shutdown –r now
   10. halt 关闭系统，等同于shutdown –h now 和 poweroff
   11. 最后总结一下，不管是重启系统还是关闭系统，首先要运行 sync 命令，把内存中的数据写到磁盘中。关机的命令有 shutdown –h now halt poweroff 和 init 0 , 重启系统的命令有 shutdown –r now reboot init 6。
4. 忘记root密码
5. 重启 linux 系统，3 秒之内要按一下回车，然后输入 e。
6. 在第二行最后边输入 single，有一个空格。具体方法为按向下尖头移动到第二行，按“e”进入编辑模式
7. 最后按“b”启动，启动后就进入了单用户模式了
8. 此时已经进入到单用户模式了，你可以更改 root 密码了。更密码的命令为 passwd



1. Linux系统的远程登录 – 通过SSH服务 访问Linux服务器
2. windows系统下安装shell相关软件
3. 输入服务器Linux的主机IP
4. 输入用户名和密码
5. Linux的密钥认证登录

SSH服务支持一种安全认证机制，即密钥认证。所谓的密钥认证，实际上是使用一对加密字符串，一个称为公钥(public key)，任何人都可以看到其内容，用于加密；另一个称为密钥(private key)，只有拥有者才能看到，用于解密。 通过公钥加密过的密文使用密钥可以轻松解密，但根据公钥来猜测密钥却十分困难。 ssh 的密钥认证就是使用了这一特性。

二、Linux系统文件和目录管理

1. 文件和目录管理--基本命令
2. pwd这个命令打印出当前所在目录。
3. cd 进入到某一个目录，/表示根目录；./表示当前目录；../表示父级目录。
4. mkdir(make directory)创建一个目录，其中-m , –p 为其选项.

-m：这个参数用来指定要创建目录的权限，该参数不常用；

-p：递归创建目录，即使上级目录不存在。如果创建的目录存在，加上就不会报错。

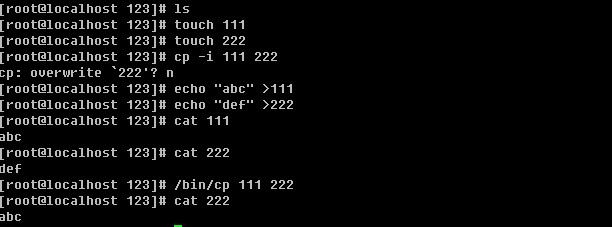
1. rmdir (rmove directory)删除一个目录，不能删除文件。其只有一个选项-p 类似与mkdir命令，这个参数的作用是将上级目录一起删除。
2. rm删除目录或者文件 -- rmdir不能删除文件
3. -f 强制的意思，如果不加这个选项，当删除一个不存在的文件时会报错。
4. -i 这个选项的作用是，当用户删除一个文件时会提示用户是否真的删除。如果删除，输入y 否则输入 n
5. -r 当删除目录时，加该选项，如果不加这个选项会报错。rm是可以删除不为空的目录的。
6. which 用来查找一个命令的绝对路径，这个命令笔者不详细介绍，因为平时笔者只用来查找一个命令的绝对路径。
7. alias用来设置指令的别名。语法：alias[别名]=[指令名称]，例如 alias rm='rm -i' ，即当我们使用rm命令时，实际上是使用的是rm –i ，而用绝对路径的/bin/rm 则不会被alias。
8. 环境变量PATH -- echo $PATH可以打印出环境变量

作用：直接打出命令，而不用打出命令所在目录的绝对路径

添加环境变量：”$PATH”:/root

绝对路径执行命令：/root/ls ； ./ls

1. ls命令用来查看某个目录或者文件。 -- man ls 查看详细帮助信息
2. -a 全部的档案都列出，包括隐藏的。隐藏文件的文件名是以.开头的。
3. -l 详细列出文件的属性信息，包括大小、创建日期、所属主所属组等等。ll 这个命令等同于ls –l 。
4. -color=never/always/auto never即不要显示颜色，always 即总显示颜色，auto 是由系统自行判断。
5. -d 后边跟目录，如果不加这个选项则列出目录下的文件，加上后只列车目录本身。
6. cp copy的简写，即拷贝。格式为 cp [选项] [来源文件] [目的文件] ，例如我想把test1 拷贝成test2 ，这样即可 cp test1 test2。
7. -d 这里涉及到一个“连接”的概念。连接分为软连接和硬连接。软连接和快捷方式有点像。如果不加这个-d 则拷贝软连接时会把软连接的目标文件拷贝过去，而加上后，其实只是拷贝了一个连接文件（即快捷方式）。
8. -r 如果你要拷贝一个目录，必须要加-r选项，否则你是拷贝不了目录的。
9. -i 如果遇到一个存在的文件，会问是否覆盖。在Redhat/CentOS系统中，我们使用的cp其实是cp –i
10. -u 该选项仅当目标文件存在时才会生效，如果源文件比目标文件新才会拷贝，否则不做任何动作。



touch命令：如果有这个文件，则会改变文件的访问时间，如果没有这个文件就会创建这个文件。

echo命令：打印，其写入作用的就是这个大于号”>” 在linux中这叫做重定向，即把前面产生的输出写入到后面的文件中。

cat命令：c读一个文件，并把读出的内容打印到当前屏幕上。

1. mv 命令(move移动)，格式为 mv [ 选项 ] [源文件] [目标文件]
2. -i 和cp的-i 一样，当目标文件存在时会问用户是否要覆盖。在Redhat/CentOS系统中，我们使用的mv其实是mv –i
3. -u 和上边cp 命令的-u选项一个作用，当目标文件存在时才会生效，如果源文件比目标文件新才会移动，否则不做任何动作。
4. cat 比较常用的一个命令，即查看一个文件的内容并显示在屏幕上。
5. -n 查看文件时，把行号也显示到屏幕上。
6. -A 显示所有东西出来，包括特殊字符
7. ”>>”这个符号跟前面介绍的”>”的作用都是重定向，即把前面输出的东西输入到后边的文件中，只是”>>”是追加的意思，而用”>”，如果文件中有内容则会删除文件中内容，而”>>”则不会。
8. 其他命令
9. tac命令是cat的反写，同样的功能也是反向打印文件的内容到屏幕上。
10. more命令是用来查看一个文件的内容，分屏查看，按空格键翻页。
11. less作用跟more一样，但比more好在可以上翻，下翻。 j键向上移动，k键向下移动。
12. head命令后直接跟文件名，则显示文件的前十行。如果加 –n 选项则显示文件前n行。
13. tail 和head一样，后面直接跟文件名，则显示文件最后十行。如果加-n 选项则显示文件最后n行。

-f 动态显示文件的最后十行，如果文件是不断增加的，则用-f 选项。如：tail -f /var/log/messages

tail命令可以用来查看日志文件 。

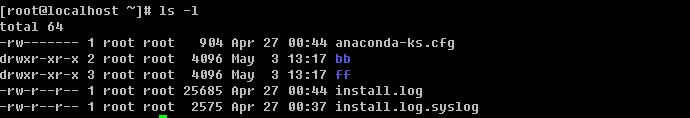
1. 文件的所属主以及所属组
2. 概念

一个linux目录或者文件，都会有一个所属主和所属组。所属主，即文件的拥有者，而所属组，即该文件所属主所在的一个组。

1. 用户权限

Linux文件属性不仅规定了所属主和所属组，还规定了所属主（user）、所属组(group)以及其他用户（others）对该文件的权限。你可以通过ls -l 来查看这些属性。

1. Lixnux文件属性 -- 总共9列内容，空格划分列



1. 第1列，包含的东西有该文件类型和所属主、所属组以及其他用户对该文件的权限。

第一位描述文件类型，

d 表示该文件为目录。

- 表示该文件为普通文件。

l 表示该文件为连接文件（linux file）。

b 表示该文件为块设备文件，比如磁盘分区。

c 表示该文件为串行端口设备，例如键盘、鼠标。

s 表示该文件为套接字文件（socket），用于进程间通信。

后边的9位，每三个为一组。均为rwx 三个参数的组合。其中r 代表可读，w代表可写，x代表可执行。前三位为所属主（user）的权限，中间三位为所属组（group）的权限，最后三位为其他非本群组（others）的权限。

1. 第2列，表示为连接占用的节点（inode），若为目录时，通常与该目录地下还有多少目录有关系，关于连接（link）在以后章节详细介绍。
2. 第3列，表示该文件的所属主。
3. 第4列，表示该文件的所属组。
4. 第5列，表示该文件的大小。
5. 第6列、第7列和第8列为该文件的创建日期或者最近的修改日期，分别为月份日期以及时间。
6. 第9列，文件名。如果前面有一个. 则表示该文件为隐藏文件。
7. 更改文件权限
8. 更改文件的权限，也就是更改所属主、所属组以及他们对应的读写执行权限。
9. 更改所属组 chgrp，语法：chgrp [组名] [文件名]

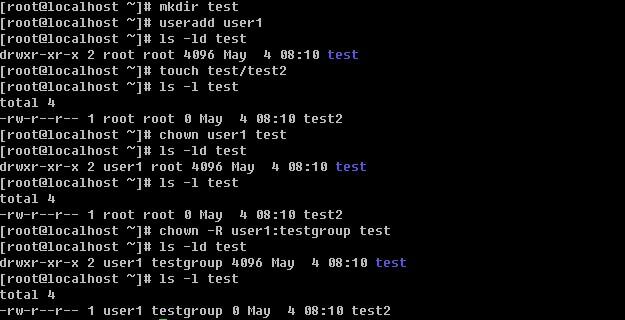
chgrp testgroup test1 将test1所属组改为testgroup

1. 更改文件的所属主 chown

语法：chown [ -R ] 账户名文件名

chown [ -R ] 账户名：组名文件名

这里的-R选项只作用于目录，作用是级联更改，即不仅更改当前目录，连目录里的目录或者文件全部更改。



1. 改变用户对文件的读写执行权限 chmod
2. 在linux中为了方便更改这些权限，linux使用数字去代替rwx ，具体规则为r: 4 w:2 x:1 -:0 举个例子，-rwxrwx---用数字表示就是 770，具体是这样来的：

rwx = 4+2+1=7; rwx= 4+2+1=7; --- = 0+0+0=0

1. chmod 语法： chmod [-R] xyz 文件名（这里的xyz，表示数字）

-R 选项作用同chown，级联更改。

**值得提一下的是**，在linux系统中，默认一个目录的权限为 755，而一个文件的默认权限为644.

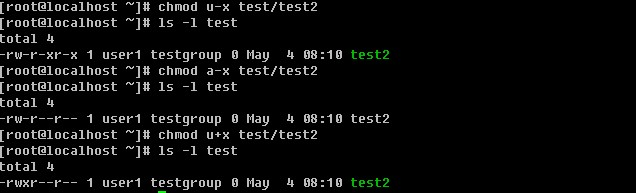
1. 权限分为三群，user，group，others

6_87.png.jpg

+：加上 -：减去 =：设定

chmod u=rwx,og=rx 文件名

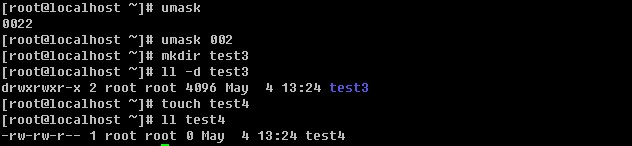
chmod u-x 文件名



1. umask -- 关于默认权限

查看umask值只要输入umask然后回车。 umask预设是0022，其代表什么含义？先看一下下面的规则：

1. 若用户建立为普通文件，则预设“没有可执行权限”，只有rw两个权限。最大为666（-rw-rw-rw-）
2. 若用户建立为目录，则预设所有权限均开放，即777（drwxrwxrwx）
3. umask数值代表的含义为，上边两条规则中的默认值（文件为666，目录为777）需要减掉的权限。所以目录的权限为(rwxrwxrwx) – (----w--w-) = (rwxr-xr-x)，普通文件的权限为(rw-rw-rw-) – (----w--w-) = (rw-r--r--)。umask的值是可以自定义的，比如设定umask 为 002，你再创建目录或者文件时，默认权限分别为(rwxrwxrwx) – (-------w-) = (rwxrwxr-x)和(rw-rw-rw-) – (-------w-) = (rw-rw-r--)。
4. umask 可以在/etc/bashrc里面更改，预设情况下，root的umask为022，而一般使用者则为002，因为可写的权限非常重要，因此预设会去掉写权限。



1. chattr 修改文件的特殊属性 -- 设置的属性是级联生效的

语法： chattr [+-=][ASaci [文件或者目录名]

+-= ：分别为增加、减少、设定

A：增加该属性后，文件或目录的atime将不可被修改；

S：增加该属性后，会将数据同步写入磁盘中；

a：增加该属性后，只能追加不能删除，非root用户不能设定该属性；

c：自动压缩该文件，读取时会自动解压；

i：增加后，使文件/目录不能被删除、重命名、设定连接、写入、新增数据；

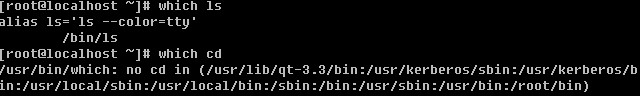
1. lsattr 列出文件/目录的特殊属性
2. 语法： lsattr [-aR] [文件/目录名]

-a：类似与ls 的-a 选项，即连同隐藏文件一同列出；

-R：连同子目录的数据一同列出

1. 在 linux 下搜索一个文件
2. which用来查找可执行文件的绝对路径。

需要注意的一点是，which只能用来查找PATH环境变量中出现的路径下的可执行文件。当查找的文件在PATH变量中并没有时，就会报错。



1. whereis 通过预先生成的一个文件列表库去查找跟给出的文件名相关的文件。

语法： whereis [-bmsu] [文件名称]

-b：只找binary 文件

-m：只找在说明文件manual路径下的文件

-s：只找source来源文件

-u：没有说明档的文件



1. find --- **最重要的**

语法： find [路径] [参数] 下面介绍几个笔者经常用的参数

-atime +n ：访问或执行时间大于n天的文件

-ctime +n ：写入、更改inode属性（例如更改所有者、权限或者连接）时间大于n天的文件

-mtime +n ：写入时间大于n天的文件

1. 文件的 Access time，atime 是在读取文件或者执行文件时更改的。

文件的 Modified time，mtime 是在写入文件时随文件内容的更改而更改的。文件的 Create time，ctime 是在写入文件、更改所有者、权限或链接设置时随 Inode 的内容更改而更改的。

1. ls -l 命令可用来列出文件的 atime、ctime 和 mtime。

ls -lc filename 列出文件的 ctime

ls -lu filename 列出文件的 atime

ls -l filename 列出文件的 mtime

1. -name filename 直接查找该文件名的文件，这个使用最多了。

find /root -name test3 //查找root文件夹下文件名为test3的文件

1. -type type ：通过文件类型查找。type 包含了 f, b, c, d, l, s 等等。

find ./file1 -type d

find ./file1 -type f

1. linux 的文件系统
2. Ext3文件系统 -- 日志功能

Ext3文件系统是直接从Ext2文件系统发展而来，Ext3文件系统带有日志功能，可以跟踪记录文件系统的变化，并将变化内容写入日志，写操作首先是对日志记录文件进行操作，若整个写操作由于某种原因 (如系统掉电) 而中断，系统重启时，会根据日志记录来恢复中断前的写操作，而且这个过程费时极短。目前Ext3文件系统已经非常稳定可靠。它完全兼容Ext2文件系统。用户可以平滑地过渡到一个日志功能健全的文件系统中来。

1. ReiserFS系统

ReiserFS 的优势在于，它是基于 B\*Tree 快速平衡树这种高效算法的文件系统，例如在处理小于 1k 的文件比 Ext3 快 10 倍。再一个就是 ReiserFS 空间浪费较少，它不会对一些小文件分配 inode，而是打包存放在同一个磁盘块 (簇) 中，Ext2/Ext3 是把它们单独存放在不同的簇上，如簇大小为 4k，那么 2 个 100 字节的文件会占用 2 个簇，ReiserFS 则只占用一个。当然 ReiserFS 也有缺点，就是每升级一个版本，都要将磁盘重新格式化一次。

1. linux 文件类型
2. 正规文件（regular file）：就是一般类型的文件，当用ls –l 查看某个目录时，**第一个属性为”-“的文件就是正规文件，或者叫普通文件。**正规文件又可分成纯文字文件（ascii）和二进制文件（binary）。纯文本文件是可以通过cat, more, less等工具直接查看内容的，而二进制文件并不能。例如我们用的命令/bin/ls 这就是一个二进制文件。
3. 目录（directory）：这个很容易理解，就是目录，跟windows下的文件夹一个意思，只不过在linux中我们不叫文件夹，而是叫做目录。ls –l 查看**第一个属性为”d”。**
4. 连接档（link）：ls –l 查看**第一个属性为 “l”**，类似windows下的快捷方式。这种文件在linux中很常见，而且笔者在日常的系统运维工作中用的很多，所以你要特意留意一下这种类型的文件。
5. 设备档（device）：与系统周边相关的一些档案，通常都集中在 /dev 这个目录之下！**区块 (block) 设备档**：就是一些储存数据，以提供系统存取的接口设备，简单的说就是硬盘啦！例如你的一号硬盘的代码是 /dev/hda1 等等的档案啦！**第一个属性为 “ b “；**

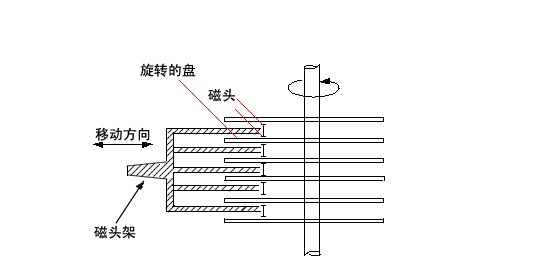
**字符 (character) 设备档**：亦即是一些串行端口的接口设备，例如键盘、鼠标等等！**第一个属性为 “ c “。**

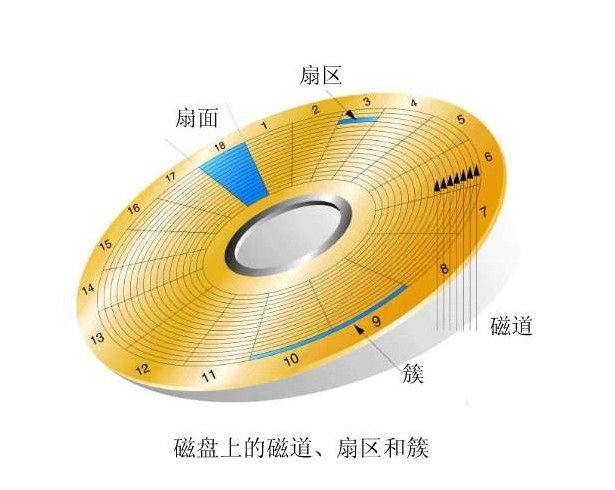
1. linux 文件后缀名

在linux系统中，文件的后缀名并没有具体意义，也就是说，你加或者不加，都无所谓。但是为了容易区分，一般都会加上后缀名用来当标志。

1. linux 中的连接档
2. inode概念

磁盘的整体构造。磁盘是有多个盘片（类似与光盘）重叠在一起构成的，而每个盘片上会有一个可以移动的磁头，这个磁头的作用就是用来读写数据的。磁头并不是一直在动，当磁头固定时，盘片转一圈，这一圈就是一个磁道了。很多个盘片同半径的那一圈的磁道总和称为磁柱。而由圆心向外画出直线，可以得到一个个扇区，如图二所示，一个扇区的物理量大约是 512 bytes ( 约 0.5K )。





1. 硬盘分割

在进行硬盘分割的时候，最小都是以磁柱为单位进行分割的，那么分割完成之后自然就是格式化( format )。在 Linux 里面我们在进行格式化的时候必须要考虑到 Block 与 inode 的信息，block 是我们磁盘可以记录的最小单位，是由数个 sector 所组成的，所以他的大小通常为 n\*512 bytes ，例如 4K 。Block 是记录“档案内容数据”的地区，而 inode 则是记录“该档案的属性、及该档案放置在哪一个 Block 之内”的信息。所以，每个档案都会占用到至少一个 inode 。

当我们 Linux 系统要找到这个档案时，他会先去搜寻 inode table 找到这个档案的属性及数据放置的地区，然后再到数据去找到数据存放的 Block 进而将数据取出利用。这个 inode 数目在一开始就会被设定好，他的设定方式通常是利用 ( 硬盘大小 / 一个容量 )，这个容量至少应该比 Block 要大一些较佳，例如刚刚的 Block 订为 4K ，那么 inode 可以订为 8K 左右。而一个 inode 的大小为 128 bytes 这么大！这么一来的话，我们就可以清楚的知道了，那就是一个 partition 格式化为一个 filesystem 之后，基本上，他一定会有 inode table 与 data area 两个区块，一个用来记录档案的信息与该档案放置的 block 区块，一个用来记录档案的内容！

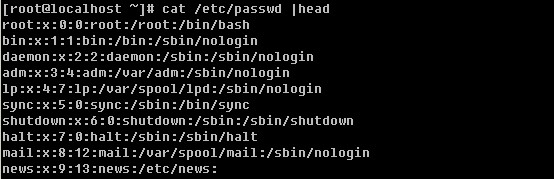
1. ln 建立连接档 --连接档分为两种，硬连接（hard link）和软连接（symbolic link）。
2. **Hard Links**：上面内容中说过，当系统要读取一个文件时，就会先去读inode table，然后再去根据inode中的信息到块区域去将数据取出来。而hard link 是直接再建立一个inode连接到文件放置的块区域。也就是说，进行hard link的时候实际上该文件内容没有任何变化，只是增加了一个指到这个文件的inode，不过这样一来就会有个问题，因为增加的inode会连接到块区域，而目录本身仅仅消耗inode而已，那么hard link就不能连接目录了。请你记住，hard link 有两个限制：1 不能跨文件系统，因为不通的文件系统有不同的inode table； 2 不能连接目录。
3. **Symbolic Links**：跟hard link不同，这个是建立一个独立的文件，而这个文件的作用是当读取这个连接文件时，它会把读取的行为转发到该文件所link的文件上。这样讲，也许比较绕口，那么就来举一个例子。现在有文件a，我们做了一个软连接文件b（只是一个连接文件，非常小），b指向了文件a。当读取b时，那么b就会把读取的动作转发到a上，这样就读取到了文件a。所以，当你删除文件a时，文件b并不会被删除，但是再读取b时，会提示无法打开文件。而，当你删除b时，a是不会有任何影响的。
4. ln 语法： ln [-s] [来源文件] [目的文件]

ln 常用的选项就一个-s ，如果不加就是建立硬连接，加上就建立软连接。

三、linux 系统用户以及用户组管理

/etc/passwd 和 /etc/shadow 两个文件是管理用户登录的文件

1. /etc/passwd 文件



/etc/passwd由’:’分割成7个字段，每个字段的具体含义是：

1）用户名（如第一行中的root就是用户名），代表用户账号的字符串。用户名字符可以是大小写字母、数字、减号（不能出现在首位）、点以及下划线，其他字符不合法。虽然用户名中可以出现点，但不建议使用，尤其是首位为点时，另外减号也不建议使用，因为容易造成混淆。

2）存放的就是该账号的口令，为什么是’x’呢？早期的unix系统口令确实是存放在这里，但基于安全因素，后来就将其存放到/etc/shadow中了，在这里只用一个’x’代替。

3）这个数字代表用户标识号，也叫做uid。系统识别用户身份就是通过这个数字来的，0就是root，也就是说你可以修改test用户的uid为0，那么系统会认为root和test为同一个账户。通常uid的取值范围是0~65535，0是超级用户（root）的标识号，1~499由系统保留，作为管理账号，普通用户的标识号从500开始，如果我们自定义建立一个普通用户，你会看到该账户的标识号是大于或等于500的。

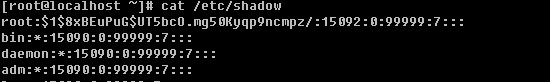
4）表示组标识号，也叫做gid。这个字段对应着/etc/group 中的一条记录，其实/etc/group和/etc/passwd基本上类似。

5）注释说明，该字段没有实际意义，通常记录该用户的一些属性，例如姓名、电话、地址等等。不过，当你使用finger的功能时就会显示这些信息的（稍后做介绍）。

6）用户的家目录，当用户登录时就处在这个目录下。root的家目录是/root，普通用户的家目录则为/home/username，这个字段是可以自定义的，比如你建立一个普通用户test1，要想让test1的家目录在/data目录下，只要修改/etc/passwd文件中test1那行中的该字段为/data即可。

7）shell，用户登录后要启动一个进程，用来将用户下达的指令传给内核，这就是shell。Linux的shell有很多种sh, csh, ksh, tcsh, bash等，而Redhat/CentOS的shell就是bash。查看/etc/passwd文件，该字段中除了/bin/bash外还有/sbin/nologin比较多，它表示不允许该账号登录。如果你想建立一个账号不让他登录，那么就可以把该字段改成/sbin/nologin，默认是/bin/bash。

1. /etc/shadow这个文件，和/etc/passwd类似，用”:”分割成9个字段。



1）用户名，跟/etc/passwd对应。

2）用户密码，这个才是该账号的真正的密码，不过这个密码已经加密过了，但是有些黑客还是能够解密的。所以为了安全，该文件属性设置为600，只允许root读写。

3）上次更改密码的日期，这个数字是这样计算得来的，距离1970年1月1日到上次更改密码的日期，例如上次更改密码的日期为2012年1月1日，则这个值就是365\*（2012-1970）+1=15331。

4）要过多少天才可以更改密码，默认是0，即不限制。

5）密码多少天后到期。即在多少天内必须更改密码，例如这里设置成30，则30天内必须更改一次密码，否则将不能登录系统，默认是99999，可以理解为永远不需要改。

6）密码到期前的警告期限，若这个值设置成7，则表示当7天后密码过期时，系统就发出警告告诉用户，提醒用户他的密码将在7天后到期。

7）账号失效期限。你可以这样理解，如果设置这个值为3，则表示：密码已经到期，然而用户并没有在到期前修改密码，那么再过3天，则这个账号就失效了，即锁定了。

8）账号的生命周期，跟第三段一样，是按距离1970年1月1日多少天算的。它表示的含义是，账号在这个日期前可以使用，到期后账号作废。

9）作为保留用的，没有什么意义。

1. 新增/删除用户和用户组
2. 新增一个组 groupadd [-g GID] groupname

不加-g 则按照系统默认的gid创建组，跟用户一样，gid也是从500开始的



-g选项可以自定义gid

groupadd -g 505 grptest2

1. 删除组 gropudel groupname

groupdel grptest1

groupdel 组名

1. 增加用户 useradd [-u UID] [-g GID] [-d HOME] [-M] [-s]

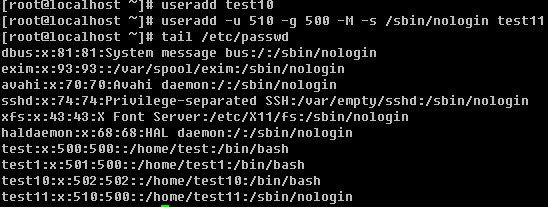
-u 自定义UID

-g 使其属于已经存在的某个GID

-d 自定义用户的家目录

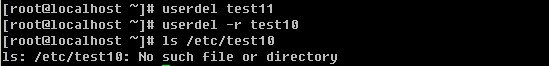
-M 不建立家目录

-s 自定义shell



1. 删除用户 userdel [-r] username

-r 选项的作用是删除用户时，连同用户的家目录一起删除。



1. 创建/修改一个用户的密码 “passwd [username]”

创建密码：passwd

修改密码：passwd test

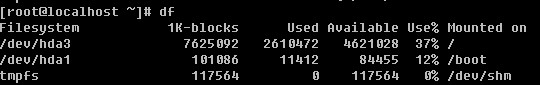
1. 用户身份切换
2. su - 就可以切换成root身份，前提是知道root的密码。
3. 使用echo $LOGNAME来查看当前登录的用户名。
4. su 的语法为： su [-] username

后面可以跟”-”也可以不跟，普通用户su不加username时就是切换到root用户，当然root用户同样可以su到普通用户。

1. 使用sudo执行一个root才能执行的命令是可以办到的，但是需要输入密码，这个密码并不是root的密码而是用户自己的密码。默认只有root用户能使用sudo命令，普通用户想要使用sudo，是需要root预先设定的，即，使用visudo命令去编辑相关的配置文件/etc/sudoers。如果没有visudo这个命令，请使用” yum install -y sudo”安装。

四、Linux 磁盘管理

1. df 命令，可以查看已挂载磁盘的总容量、使用容量、剩余容量等，可以不加任何参数，默认是按k为单位显示的。



-i 使用inodes 显示结果。

-h 使用合适的单位显示，例如G。

-k -m 分别为使用K，M为单位显示。

1. Filesystem 表示扇区，也就是你划分磁盘时所分的区；1K-blocks/1M-blocks表示以1K/1M为单位；Used 和 Available 分别是已使用和剩余；Use% 就是已经使用的百分比，如果这个值大于90% 那么你就应该注意了，磁盘很有可能马上就会变满的；Mounted on 则表示该分区（扇区）所挂载的地方。
2. du 命令，用来查看某个目录所占空间大小

语法：du [-abckmsh] [文件或者目录名]

-a：全部文件与目录大小都列出来。如果不加任何选项和参数只列出目录（包含子目录）大小。

-b：列出的值以bytes为单位输出，默认是以Kbytes

-c：最后加总

-k：以KB为单位输出

-m：以MB为单位输出

-s：只列出总和

-h：系统自动调节单位，例如文件太小可能就几K，那么就以K为单位显示，如果大到几G，则就以G为单位显示。笔者习惯用 du –sh filename 这样的形式。

1. 磁盘的分区和格式化 -- fdisk linux下的硬盘分区工具 **重要**

语法： fdisk [-l ] [设备名称]

1. -l ：后边不跟设备名会直接列出系统中所有的磁盘设备以及分区表，加上设备名会列出该设备的分区表。

fdisk -l

1. 如果不加-l 则进入另一个模式，在该模式下，可以对磁盘进行分区操作。

刚进入该模式下，会有一个提示Command (m for help): 按m则会打印出帮助列表。

P：打印当前磁盘的分区情况。

n：重新建立一个新的分区。

w：保存操作。

q：退出。

d：删除一个分区

1. mkfs.ext2 / mkfs.ext3 /mke2fs 格式化linux硬盘分区

当用man查询这三个命令的帮助文档时，你会发现我们看到了同一个帮助文档，这说明三个命令是一样的。常用的选项有：

-b：分区时设定每个数据区块占用空间大小，目前支持1024, 2048 以及4096 bytes每个块。

-i：设定inode大小

-N：设定inode数量，有时使用默认的inode数不够用，所以要自定设定inode数量。

-c：在格式化前先检测一下磁盘是否有问题，加上这个选项后会非常慢

-L：预设该分区的标签label

-j：建立ext3格式的分区，如果使用mkfs.ext3 就不用加这个选项了

1. e2label 用来查看或者修改分区的标签（label）

当想要修改标签名时，分区编号后边跟想要的标签名即可。

1. fsck 检查硬盘有没有坏道

语法： fsck [-Aar] [分区]

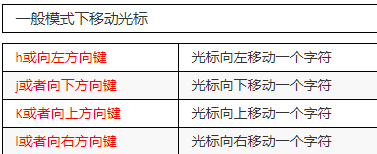
-A ：加该参数时，后不需要跟分区名作为参数。它会自动检查/etc/fstab 文件下的所有分区（开机过程中就会执行一次该操作）；

-a ：自动修复检查到有问题的分区；

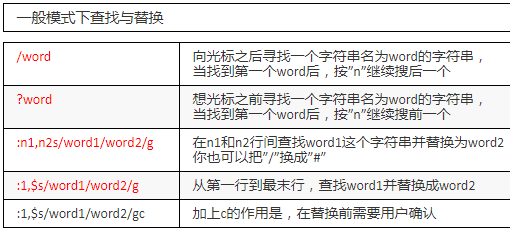
-r ：当检查到有坏道的分区时会让用户决定是否修复。

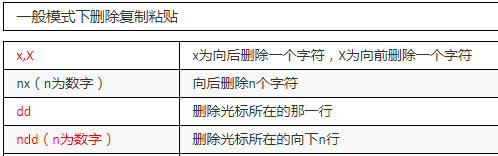
五、文本编辑工具 vim

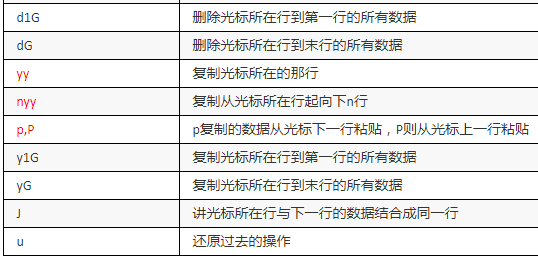
1. vim的三种模式：一般模式、编辑模式、命令模式。
2. 一般模式：当你vim filename 编辑一个文件时，一进入该文件就是一般模式了。在这个模式下，你可以做的操作有，上下移动光标；删除某个字符；删除某行；复制、粘贴一行或者多行。 **如** ：**vim test.txt**
3. 编辑模式：一般模式下，是不可以修改某一个字符的，只能到编辑模式了。从一般模式进入编辑模式，只需你按一个键即可（i,I,a,A,o,O,r, R）。当进入编辑模式时，会在屏幕的最下一行出现“INSERT或REPLACE”的字样。从编辑模式回到一般模式只需要按一下键盘左上方的ESC键即可。
4. 命令模式：在一般模式下，输入”:”或者”/”即可进入命令模式。在该模式下，你可以搜索某个字符或者字符串，也可以保存、替换、退出、显示行号等等。

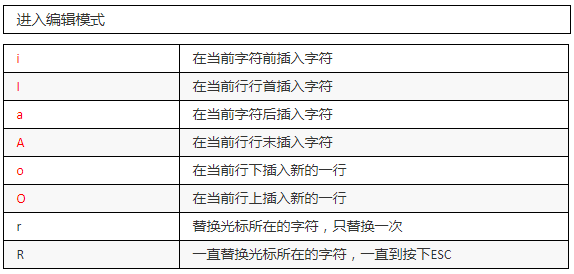














六、文档的压缩与打包

1. 在linux下最常见的压缩文件通常都是以.tar.gz 为结尾的，除此之外还有.tar, .gz, .bz2, .zip等等。以前也介绍过linux系统中的后缀名其实要不要无所谓，但是对于压缩文件来讲必须要带上。这是为了判断压缩文件是由哪种压缩工具所压缩，而后才能去正确的解压缩这个文件。

.gz gzip 压缩工具压缩的文件

.bz2 bzip2 压缩工具压缩的文件

.tar tar 打包程序打包的文件(tar并没有压缩功能，只是把一个目录合并成一个文件)

.tar.gz 可以理解为先用tar打包，然后再gzip压缩

.tar.bz2 同上，先用tar打包，然后再bzip2压缩

1. gzip -- gzip不可以压缩目录

语法： gzip [-d#] filename 其中#为1-9的数字

-d ：解压缩时使用

-# ：压缩等级，1压缩最差，9压缩最好，6为默认

1. bzip2 命令

语法：bzip2 [-dz] filename

-d ：解压缩

-z ：压缩

1. tar 命令

语法：tar [-zjxcvfpP] filename

-z ：是否同时用gzip压缩

-j ：是否同时用bzip2压缩

-x ：解包或者解压缩

-t ：查看tar包里面的文件

-c ：建立一个tar包或者压缩文件包

-v ：可视化

-f ：后面跟文件名，压缩时跟-f文件名，意思是压缩后的文件名为filename，解压时跟

-f文件名，意思是解压filename。请注意，如果是多个参数组合的情况下带有-f，请把f写到最后面。

-p ：使用原文件的属性，压缩前什么属性压缩后还什么属性。（不常用）

-P ：可以使用绝对路径。（不常用）

--exclude filename ：在打包或者压缩时，不要将filename文件包括在内。（不常用）

七、shell脚本基础

1. shell概念

简单点理解，就是系统跟计算机硬件交互时使用的中间介质，它只是系统的一个工具。实际上，在shell和计算机硬件之间还有一层东西那就是系统内核了。打个比方，如果把计算机硬件比作一个人的躯体，而系统内核则是人的大脑，至于shell，把它比作人的五官似乎更加贴切些。回到计算机上来，用户直接面对的不是计算机硬件而是shell，用户把指令告诉shell，然后shell再传输给系统内核，接着内核再去支配计算机硬件去执行各种操作。

1. Bash shell

笔者接触的linux发布版本（Redhat/CentOS）系统默认安装的shell叫做bash，即Bourne Again Shell，它是sh（Bourne Shell）的增强版本。

1. 记录命令历史

我们敲过的命令，linux是会有记录的，预设可以记录1000条历史命令。这些命令保存在用户的家目录中的.bash\_history文件中。有一点需要你知道的是，只有当用户正常退出当前shell时，在当前shell中运行的命令才会保存至.bash\_history文件中。

1. !! （连续两个”!”），表示执行上一条指令；
2. !n（这里的n是数字），表示执行命令历史中第n条指令，例如”!100”表示执行命令历史中第100个命令；
3. !字符串（字符串大于等于1），例如!ta，表示执行命令历史中最近一次以ta为开头的指令。
4. 指令和命令行补全
5. 按一次tab键补全一个指令，也可以补全一个路径或者一个文件名。
6. 连续按两次tab键，系统则会把所有的指令或者文件名都列出来。
7. 别名
8. alias把一个常用的并且很长的指令别名一个简洁易记的指令。

alias语法很简单，alias [命令别名]=[’具体的命令’]。

1. 可以用unalias解除别名功能。
2. 直接敲alias会看到目前系统预设的alias 。
3. 通配符

在bash下，可以使用\*来匹配零个或多个字符，而用?匹配一个字符。

12_8.png.jpg

1. 输入输出重定向

输入重定向用于改变命令的输入，输出重定向用于改变命令的输出。

1. 输入重定向的命令是<。
2. 输出重定向的命令是>。
3. 错误重定向2>。
4. 追加重定向>>。

6）管道符 | -- 把前面的命令运行的结果丢给后面的命令。

7）作业控制。

当运行一个进程时，你可以使它暂停（按Ctrl+z），然后使用fg命令恢复它，利用

bg命令使他到后台运行，你也可以使它终止（按Ctrl+c）。

1. 变量

环境变量PATH，这个环境变量就是shell预设的一个变量，通常shell预设的变量都是大写的。变量，说简单点就是使用一个较简单的字符串来替代某些具有特殊意义的设定以及数据。PATH就代替了所有常用命令的绝对路径的设定。

* 使用env命令即可全部列出系统预设的全部系统变量（全部环境变量）。不过登录的用户不一样这些环境变量的值也不一样。当前显示的就是root这个账户的环境变量了。
  1. PATH 决定了shell将到哪些目录中寻找命令或程序
  2. HOME 当前用户主目录
  3. HISTSIZE 历史记录数
  4. LOGNAME 当前用户的登录名
  5. HOSTNAME 指主机的名称
  6. SHELL 前用户Shell类型
  7. LANG 语言相关的环境变量，多语言可以修改此环境变量
  8. MAIL 当前用户的邮件存放目录
  9. PWD 当前目录
* set命令把系统预设的全部变量都显示出来，不仅可以显示系统预设的变量，也可以连同用户自定义的变量显示出来。

用户自定义的变量只能在当前shell生效，对其他shell不生效。使用bash命令打开一个新shell，exit退出当前shell。

1. 让用户自定义变量一直生效，不受其他影响。
2. 系统内所有用户登录后都能使用该变量

需要在/etc/profile文件最末行加入 “export myname=Aming” 然后运行”source /etc/profile”就可以生效了。此时你再运行bash命令或者直接su - test账户看看。

1. 只让当前用户使用该变量，即使注销再登录也能用。

需要在用户主目录下的.bashrc文件最后一行加入“export myname=Aming” 然后运行”source .bashrc”就可以生效了。这时候再登录test账户，myname变量则不会生效了。上面用的source命令的作用是，讲目前设定的配置刷新，即不用注销再登录也能生效。

1. 用户设置自定义变量的方式
2. 设定变量的格式为”a=b”，其中a为变量名，b为变量的内容，**等号两边不能有空格。**
3. 变量名只能由英、数字以及下划线组成，而且不能以数字开头。
4. 当变量内容带有特殊字符（如空格）时，需要加上单引号。有一种情况，需要你注意，就是变量内容中本身带有单引号，这就需要用到双引号了。
5. 如果变量内容中需要用到其他命令运行结果则可以使用反引号。
6. 变量内容可以累加其他变量的内容，需要加双引号；
7. 单引号和双引号的区别：用双引号时不会取消掉里面出现的特殊字符的本身作用（这里的$），而使用单引号则里面的特殊字符全部失去它本身的作用。
8. 其他命令
9. bash命令，如果在当前shell中运行bash指令后，则会进入一个新的shell，这个shell就是原来shell的子shell。
10. pstree这个指令会把linux系统中所有进程通过树形结构打印出来。
11. unset命令+变量名，可以取消变量的值。
12. export命令其实就是声明一下这个变量的意思，让该shell的子shell也知道变量abc的值是123.如果export后面不加任何变量名，则它会声明所有的变量。
13. 系统环境变量与个人环境变量的配置文件
14. /etc/profile：这个文件预设了几个重要的变量，例如PATH, USER, LOGNAME, MAIL, INPUTRC, HOSTNAME, HISTSIZE, umas等等。
15. /etc/bashrc ：这个文件主要预设umask以及PS1。这个PS1就是我们在敲命令时，前面那串字符了，例如笔者的linux系统PS1就是 [root@localhost ~]# ，你不妨看一下PS1的值。
16. 除了两个系统级别的配置文件外，每个用户的主目录下还有几个这样的隐藏文件：
17. .bash\_profile：定义了用户的个人化路径与环境变量的文件名称。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。
18. .bashrc：该文件包含专用于你的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。例如你可以将用户自定义的alias或者自定义变量写到这个文件中。
19. .bash\_history：记录命令历史用的。
20. .bash\_logout ：当退出shell时，会执行该文件。可以把一些清理的工作放到这个文件中。
21. linux shell中的特殊符号
22. \* 通配符：代表零个或多个字符或数字。
23. ? ：只代表一个任意的字符
24. # ：这个符号在linux中表示**注释**说明的意思，即”#”后面的内容linux忽略掉。