## Linux操作系统基础

linux的基本原则：

1、有目的单一的小程序组成、组合小程序完成复杂任务

2、一切皆文件

3、尽量避免捕获用户接口

4、配置文件保存为存文本格式

GUI接口：

CLI接口： 命令提示符，prompt,bash(sheel)

#:root

$:普通用户

命令：

命令格式

命令 选项 参数

选项 ：

短选项：-

多个选项可以组合：-a -b=-ab

长选项：--

参数：命令的作用对象，不可组合使用

虚拟终端（terminal） ：Ctrl+Art+F1-F6 init 5 |3

GUI

Gnome:C

KDE:C++

XFace:

CLI

sh

bash

cah

zsh

ksh

Tcsh

## 常用技巧

### 取消当前输入：ctrl+c

### 清屏：clear 或者ctrl+l

### 自动补全：tab

## 命名规则

1. 长度不能超过255个字符
2. 不能使用/当文件名
3. 严格区分大小写

### Bash特性

bash支持的引号

``:命令替换

“”:弱引用，实现变量替换

‘’:强引用，不完成变量替换

站在用户的角度来说shell的类型：

登录式shell:

正常通过某终端登录

Su - USERNAME

读取配置文件

/etc/prfile-->/etc/profile.d/\*.sh-->~/.bash\_profile-->~/.bashrc-->/etc/bashrc

非登录式shell:

Su SUERNAME

图形终端下打开命令窗口

自动执行shell脚本

读取配置文件：

~/.bashrc-->/etc/basrc-->/etc/profile.d/\*.sh

Bash的配置文件

全局配置

/etc/profile,/etc/profile.d/\*.sh,/ect/bashrc

个人配置

~/.bash\_profile,~/.bashrc

profile类的文件

设定环境变量

运行命令或脚本

Bashrc类的文件

设定本地变量

定义命令别名

1. 命令行编辑

命令光标跳转：

Ctrl+a:调到命令行首

Ctrl+e:调到命令行尾

Ctrl+k:删除光标至命令行尾的内容

Ctrl+u:删除光标至命令行首的内容

Ctrl+l:清屏

Shift+pageup|pagedown:向上翻屏|向下翻屏

1. 命令历史

# History：查看命令历史

-c:清空命令历史

-d:删除指定位置的命令

-w:将缓冲区的的命令历史保存再历史文件中，方便下次使用

#HISTSIZE:保存的命令历史的大小 #echo HISTSIZE 修改：etc/profile

命令历史的使用技巧

!n:执行命令历史中第n条命令

!-n:执行命令历史中的倒数n条命令

!!:执行上一条命令

!string:执行命令历史中最近一个以指定字符串开头的命令

!$:引用前一个命令的最近的一个最后一个参数

类似用法：esc+.,alt+

1. 命令补全：

tab键

连按两次tab键显示所有相关的命令

1. 路径补全：

tab键

连按两次tab键显示所有相关的命令

1. 命令别名

#alias 查看所有别名

# alias 命令 别名

在shell中定义的别名仅在当前的shell中的生命周期中有效

使用回命令本身：\CMD

1. 命令替换

把命令中某个子命令替换为其执行结果的过程

使用：$(command)、反引号`command`

1. 文件名通配（\*、？、[]）

查看所用通配符 ：man glob

正则表达式通配

\*:任意长度的任意字符

?:任意单个字符

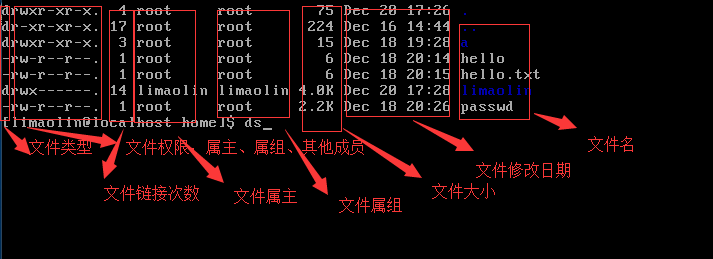
[]:匹配指定范围内的任意单个字符

## 常用命令

### su:switch user

# su [l] 用户名

### Ls:list列表



Ls:

-l:

1. 长格式

文件类型

-：普通文件（f）

d:目录文件

b:块设备文件(block)

c:字条设备文件（character）

l：符号链接文件（symbolic link file）

p:命令管道文件（pipe）

s:套接字文件（socket）

1. 文件权限：9位，每三位一组，每一组：rwx-(读、写、执行、无权限）
2. 文件硬链接次数
3. 文件的属主（owner）
4. 文件的属组（group）
5. 文件大小（size）,单位字节
6. 时间戳（timestamp）:最近一次修改的时间

访问：access

修改：modify

改变：change,metadata,元数据

-h:做单位转换

-a:显示以.开头的隐藏文件

. 表示当前目录

..表示父目录

A:不显示..

-d:显示目录自身属性

-i：index node ,inode

-r:逆序显示

-R:递归（recursive）显示

### Cd:change directory

cd~username:进入指定用户的家目录

cd -:在当前目录和前一次所在目录之间来回切换

### Type:显示指定是属于哪种类型

命令类型：

内置命令（shell内置），内部，内建

查看命令：help command

外部命令：在文件系统的某个路径下有一个与命令名称相应的可执行文件

Command --help

### 命令手册：man command

Whatis command

man分章节：

1. 用户命令（/bin,/usr/bin,/usr/local/bin）
2. 系统调用
3. 库用户
4. 特殊文件（设备文件）
5. 文件格式（配置文件的语法）
6. 游戏
7. 杂项
8. 管理命令（/sbin,/usr/sbin,/usr/local/sbin）

MAN:

Name:命令名称及功能简要说明

System：用法说明，包括可用的选项

Description：命令功能的详尽说明，包括每一个选项的意义

Files:此命令相关的配置文件

Bugs:

Examples:使用示例

See also:另外参照

q:退出

在线文档：

Info command

### Data:时间管理

系统时间：date

硬件时间：clock|hwclock(时间转换)

### Cal:日历

## 文件系统

### 根文件系统

#### /boot:

系统启动相关的文件，如内核、initrd,以及grub(bootloader)

#### /dev:

设备文件

设备文件：

块设备：随机访问，数据块

字符设备：线性访问，按字符为单位

设备号：主设备号（major）和次设备号（minor）

#### /etc:

配置文件

#### /home:

用户的家目录，每一个用户的家目录通常默认为/home/username

#### /root:

管理员的家目录,不能单独分区

#### /lib:

库文件

静态库，.a

动态库，.all,.so(shared)

/lib/modules:内核模块文件

#### /Media:

挂载点目录,移动设备

#### /mnt:

挂载点目录，额外的临时文件挂载点

#### /opt:

可选目录第三方程序的安装目录（之前）

#### /proc:

伪文件系统，内核映射文件

#### /sys:

伪文件系统，跟硬件设备相关的属性映射文件

#### /tmp:

临时文件，/var/tmp

#### /var:

可变化的文件

#### /bin:

可执行文件,用户命令

#### /sbin：

管理命令

#### /usr:

shared,read-only只读文件

/usr/bin

/usr/sbin

/usr/lib

/usr/local:第三方

/usr/local/bin

/usr/local/sbin

/usr/local/lin

### 文件查找

Locate:

非实时

#### Find:

实时、精确、支持众多查找标准

遍历指定目录中的所有文件完成查找，速度慢

查找路径：默认为当前目录

查找标准：默认为指定目录下

处理运作：默认为显示

匹配标准

-name ‘文件名’，对文件名称进行精确匹配

文件名通配：\*，？[]

-iname :文件名匹配不区分大小写

-regex:基于正则表达式查找

-user ‘用户名’:根据属主查找

-group ‘组名’：根据属组进行查找

-uid UID:根据UID查找

-gid GID:根据GID查找

-nouser:查找没有属主的文件

-nogroup:查找没有属组的文件

-type:根据文件类型查找

-size:根据文件大小查找

参数，[+|-]k,m,c 例如 10k,

组合条件

-a:and

-o:or

-not:

-atime:access time 文件访问或执行时间

-ctime:change time文件状态改变时间

-mtime:modify time 文件内容被修改

-amin:access time 文件访问或执行时间

-cmin:change time文件状态改变时间

-mmin:modify time 文件内容被修改

+ #：表示修改的时间发生在#天以前或者更早

-#：最后一次修改发生在n天以内，距离当前时间为n\*24小时以内

#：最后一次修改发生在距离当前时间n\*24小时至(n+1)\*24 小时

-prem Mode:权限匹配

/MODE:任意一位匹配即满足条件

-MODE:文件权限能完全包含此mode时才匹配

运作：

-print :显示

-ls:类似ls -l的形式显示每一文件的详细

-ok command {} \;每一次操作都需要用户确认

-exec command {} \:找出文件并执行命令

### 文本处理

#### cut:文本剪切

-d:指定分隔符，默认是空格

-f:fields指定要显示的字段按域切分

-f 1,3

-f 1-3

#### tr:转换或删除字符

-d：删除出现在字符集中的所有字符

<重定向文件

#### sort:文件排序

-n:数值排序

-r:降序排序

-t:字段分隔符

-k:以哪个字段为关键字为进行排序

-u:排序后相同的行只显示一次

-f:排序时忽略大小写

#### uniq:限制重复行

-c:显示文件中行重复的次数

-d:只显示重复的行

#### wc:文本统计

-l:显示新的行数

-w:显示单词

-c:显示字节

-L:显示最长行的大小

### 文件操作

#### Vim:编辑器

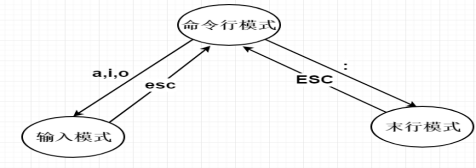
1. 打开文件

# vim /path/to/somefile

Vim +#:打开文件，并定位于第#行

Vim +:打开文件，定位于最后一行

vim模式：



编辑模式（命令模式）（默认）

输入模式

末行模式

模式转换：

命令模式-->输入模式

i:在当前光标所在字符的前面，转为输入模式

a:在当前光标所在字符的后面，转为输入模式

O:在当前光标所在字符的下方，新建一行，转为输入模式

I:在当前光标所在字符的前面，转为输入模式

A:在当前光标所在字符的后面，转为输入模式

O:在当前光标所在字符的下方，新建一行，转为输入模式

输入模式-->命令行模式

ESC

编辑模式-->末行模式

:

末行模式-->编辑模式

ESC,ESC

1. 关闭文件

末行模式：

：w:保存

：q:退出

：q！:强制退出

：wq-->x:保存并退出

：w!:强行保存

命令行模式

ZZ:直接保存退出

1. 移动光标（命令行模式）
   1. 逐字符移动

h：左

l:右

J:下

K:下

* 1. 以单词为单位移动

w:移动至下一个单词的词首

e：跳至当前或下一个单词的词尾

b：跳至当前或前一个单词的词首

* 1. 行内跳转

0：绝对行首

^:行首的第一个非空字符

$:绝对行尾

* 1. 行间跳转

#G:跳转至#行

G:最后一行

末行模式下直接给出行号即可

1. 翻屏（命令行模式）

Ctrl+f:向下翻一屏

Ctrl+b:向上翻一屏

Ctrl+d:向下翻半屏

Ctrl+u:向下翻一屏

1. 删除（命令行模式）

「x」：每按一次，删除光标所在位置的“后面”一个字符。

「#x」：例如，「6x」表示删除光标所在位置的“后面”6个字符。

「X」：大写的X，每按一次，删除光标所在位置的“前面”一个字符。

「#X」：例如，「20X」表示删除光标所在位置的“前面”20个字符。

「dd」：删除光标所在行。

「#dd」：从光标所在行开始删除#行

末行模式下：

：1,8d:删除1-8行

：.d:表示当前行

：+#：表示向下的#行

1. 复制

「yw」：将光标所在之处到字尾的字符复制到缓冲区中。

「#yw」：复制#个字到缓冲区

「yy」：复制光标所在行到缓冲区。

「#yy」：例如，「6yy」表示拷贝从光标所在的该行“往下数”6行文字。

「p」：将缓冲区内的字符贴到光标所在位置。注意：所有与“y”有关的复制命 令都必须与“p”配合才能完成复制与粘贴功能。

1. 替换

「r」：替换光标所在处的字符。

「R」：替换光标所到之处的字符，直到按下「ESC」键为止。

在命令行模式下使用s命令

起始地址，结束地址 s 分隔符（/）匹配模式 分隔符（/等） 替换字符 支持gi

%：表示全文查找

1. 恢复/撤消/还原上一次操作

「u」：如果误执行一个命令，可以马上按下「u」，撤消上一个操作。按多次 “u”可以执行多次撤消。

还原最后一次撤销操作 ctrl+r

1. 更改

「cw」：更改光标所在处的字到字尾处

「c#w」：例如，「c3w」表示更改3个字

1. Vim编辑多个文件

Vim file1 file2

:next 切换至下一个文件

:prev 切换至前一个文件

:last 切换至最后一个文件

:first 切换至第一文件

q退出 qa全部退出

1. Vim分屏显示一个窗口

Ctrl+w,s：水平拆分窗口

Ctrl+w,v:垂直拆分窗口

切换光标

Ctrl+上下箭头

：qa关闭所有窗口

1. Vim分窗口显示多个文件

Vim -o：水平分割显示

Vim -O: 垂直分割显示

1. Vim将当前文件中部分内容另存为另外一个文件

在末行模式下使用w命令

：w

: [地址] /path/to/somewhere

1. Vim将另外一个文件的内容附加到当前文件

: r /path/to/somewhere

1. Vim跟shell交互

：！Command

1. Vim高级话题
2. 显示行号

:Set number

:set nonumber

1. 显示或忽略大小写

:Set ignorecase

:set noignorecase

1. 设定自动缩进

：set autoindent

: set noautoindet

1. 查找到的文本高亮显示或取消

：set hlsearch

: set nohlsearch

5.语法高亮

：set hsyntax on

: set syntax off

1. Vim配置

全局：/etc/vimrc

局部：~/.vimrc

#### cat:连接并打开

-e:显示文件$换行符

-n:显示行数

#### more:分屏显示

翻屏：

向后翻一屏：空格键

向前翻一屏：b

向后翻一行：enter

向前翻一行：k

#### less:分屏显示(支持查找)

查找：

/keyword：向后查看

/？Keyword:向前查看

n:下一个

N:前一个

#### head:查看前n行

-n:行数

#### tail:查看后n行

f:查看文件尾部，不退出等待显示后续追加至此文件的新内容，可跟踪日志

#### grep:文本查找

# grep [-acinv] [--color=auto] '搜寻字符串' filename

-a ：将 binary 文件以 text 文件的方式搜寻数据

-c ：计算找到 '搜寻字符串' 的次数

-i ：忽略大小写的不同，所以大小写视为相同

-n ：顺便输出行号

-v ：反向选择，亦即显示出没有 '搜寻字符串' 内容的那一行！

-E:使用正则表达式=egrep

-o:只显示被匹配模式的字符串

--color=auto ：可以将找到的关键词部分加上颜色的显示喔！

正则表达式：

元字符：

.：匹配任意单个字符

[]:匹配指定范围内的单个任意字符

[^]:匹配指定范围外的单个任意字符

匹配次数：

\*：匹配其前面的字符任意次

.\*：匹配任意长度的任意字符

\？：匹配其前面的字符一次或零次（可有可无），需要反斜杠使用\

\{m,n\}:匹配前面的字符至少m次，至多n次

位置锚定：

^：锚定行首，次字符后面的任意内容必须出现在行首

$：锚定行首，次字符前面的任意内容必须出现在行首

^$：空白行

#### Basename:取出路径的基名

#### Hostname:主机名

### I/O重定向

Linux:

>: 输出

>>:追加输出

2>:重定向错误输出

2>>:追加方式

&>：重定向标准输出和错误输出

&>>:追加方式

<:输入重定向

<<:在此处生成文档

管道：前一个命令的输出，作为后一个命令的输入

#### set:设置shell

-c:保护文件在使用重新动向的时候不被覆盖

+c:关闭上述功能

### 文件压缩

#### Gzip:.gz（不可压缩目录）

gzip /path/to/somefile.gz 压缩完成后会删除原文件

-d:解压缩

-#：1-9，指定压缩比，默认是6

#### Gunzip:解压缩.Gz

Gunzip /path/to/somefile: 解压完成后会删除原文件

#### Zcat：临时查看文件内容.Gz

#### Bzip2：压缩.bz2

用法相似，比gzip具有更大的压缩比

#### Xz：压缩.xz文件（最新）

用法相似

-k:不删除源文件

#### Zip：展开文件（可压缩目录）

Zip newfile.zip file1,file2

Unzip :解压缩

#### tar：归档工具（只归档不压缩）.tar

-c:创建归档文件

-f file.tar:操作的归档文件（必须）

-x:还原归档

--xattrs:归档时，保留文件的扩展属性信息

-t:不展开归档，直接查看文件

-zcf:归档并调用gzip压缩

-zxf:调用gzip解压缩并展开归档,-z选项可省略，自动判断

类型

-jcf：bzip2

-jxf:

-Jcf:xz

-Jxf:

## 目录系统

### 常用操作

#### mkdir:创建空目录

-p:递归创建

-v:显示创建的详细信息

{}：命令行展开 mkdir /home/test{a/c,b}

# tree:查看目录树及相关便捷操作（需安装）

#### rmdir:删除目录

只能删除空目录

#### touch:创建文件（修改时间）

-a

-m

-t

-c

#### stat:查看文件状态

#### rm:删除文件

f:强制删除（递归）

-r:删除目录

#### cp:复制

i:显示提示

p:保留文件的相关信息

a:归档复制，常用语备份

-f:强制复制

-r:递归复制

#### mv:移动文件

属性同上

## 权限系统

权限：

r,w,x

0 000 ---:无权限

1 001 --x:执行

2 010 -w-:写

3 011 -wx:写和执行

4 100 --x:只读

5 101 ---:读和执行

6 110 ---:读写

7 111 --x:读写执行

文件：

r:可读，使用类似cat等命令查看文件

w:可写,可以编辑或删除此文件

x:可执行

目录：

r:可以对此目录执行ls以列出所有内部文件

w:可以在此目录创建文件

x:可以使用cd切换此目录，也可以查看内部文件的详细信息

用户：UID ,/ect/passwd=>真实地址：/etc/shadow

组：GID,/etc/group=>真实地址：/etc/gshadow

用户类别

管理员：0

普通用户：1-65535

系统用户：1-499

一般用户：500-600000

用户组类别

管理组：

普通组：

系统组：

一般组：

其他分法：

私有组：创建用户组时，如果没有为其制定所属的组，系统会自动为其创建一个与用户名同名的组

基本组：用户的默认组

附加组：额外组，默认组以外的其他组

加密方法:

对称加密:加密和解密使用同一个密码

公钥加密：每个密码都成对出现，一个为私钥，一个为公钥

单项加密：散列加密，提取数据特征码

1. 雪崩效应
2. 定长输出

Md5:128定长输出

Sha1:哈希加密，160位的定长输出

三类用户：

u:属主

g:属组

o:其他用户

a:全部用户

#### chown：改变文件属主（只有管理员可以使用此命令）

# chown USERNAME FILE,...

-R:修改目录及其内部文件的属主

--reference：引用参考 /path/to/somefile file...将文件file的属性改为somefile

#### chgrp：改变文件属组（只有管理员可以使用此命令）

-R:修改目录及其内部文件的属主

--reference：引用参考 /path/to/somefile file...将文件file的属性改为somefile

#### chmod：改变文件的权限

1. 修改三类用户的权限:

Chmod MODE file,...

-R:

--reference：

1. 修改某类用户或某些类用户权限（u,g,o,a）

Chmod 用户类别=MODE file,...

1. 修改某类用户的某位或某些位权限

Chmod 用户类别+|-MODE file,...

### 特殊权限：

#### SUID:

运行某程序是，相应进程的属主是程序文件自身的属主，而不是启动者:

Chmod u+s file

Chmod u-s file

#### **SGID:**

运行某程序时，相应进程的属组是程序文件自身的属组，而不是启动者所属的基本组

Chmod g+s file

Chmod g-s file

#### Sticky:

在一个公共目录，每个都可以创建文件，删除自己的文件，但不能动别人的文件

Chmod o+t dir

Chmod o-t dir

例：chmod 5755 /backup/test

#### 文件系统访问控制列表:

利用文件扩展保存额外的访问控制权限

#### Setfacl:设置文件的额外控制列表

-m:设定

u:UID :perm

g:GID:perm

-x:取消

#### getfacl:查看文件的额外控制列表

### 用户管理：

Useradd,userdel,usermod,passwd,chsh,chfn,finger,id,chage.

#### useradd [option] USERNAME创建用户

-u: UID

-g：GID（基本组）组名必须存在

-G:GID,...(附加组)可以多个，逗号隔开

-c:注释信息

-d:/path/to/somedirectory 指定某个家目录路径（默认下是home）

-s:指定shell路径

-m:-k不为用户创建家目录

-r:添加系统目录

#### userdel:[option] USERNAME删除用户

-r:同时删除用户的家目录

#### id:查看用户的账号属性信息

-g或--group 　 显示用户所属群组的ID。

-G或--groups 显示用户所属附加群组的ID。

-n或--name 　 显示用户，所属群组或附加群组的名称。

-r或--real 　 显示实际ID。

-u或--user 　 显示用户ID。

#### finger:查看用户的账号信息

-l：列出该用户的帐号名称，真实姓名，用户专属目录，登入所用的Shell，登入时 间，转信地址，电子邮件状态，还有计划文件和方案文件内容；

-m：排除查找用户的真实姓名；

-s：列出该用户的帐号名称，真实姓名，登入终端机，闲置时间，登入时间以及 地址和电话；

-p：列出该用户的帐号名称，真实姓名，用户专属目录，登入所用的Shell，登入 时间，转信地址，电子邮件状态，但不显示该用户的计划文件和方案文件内容。

#### usermod:修改用户的账号属性

-c<备注>：修改用户帐号的备注文字；

-d<登入目录>：修改用户登入时的目录；

-e<有效期限>：修改帐号的有效期限；

-f<缓冲天数>：修改在密码过期后多少天即关闭该帐号；

-g<群组>：修改用户所属的群组；

-G<群组>；修改用户所属的附加群组；

-l<帐号名称>：修改用户帐号名称；

-L：锁定用户密码，使密码无效；

-s：修改用户登入后所使用的shell；

-u：修改用户ID；

-U:解除密码锁定。

#### chsh:修改用户的默认shell

#### chfn:修改注释信息

#### passwd:密码管理

-d：删除密码，仅有系统管理者才能使用；

-f：强制执行；

-k：设置只有在密码过期失效后，方能更新；

-l：锁住密码；

-s：列出密码的相关信息，仅有系统管理者才能使用；

-u：解开已上锁的帐号。

#### pwck:检出用户完整性

#### Whoami:查看用户

#### who:显示最近登录的用户

#### w:显示最近登录的用户和在做什么

#### last:显示最近登录的用户列表及系统重启历史

#### Lastb:显示用户错误的登录尝试

#### Lastlog:显示每一个用户最近的成功登录信息

### 组管理

#### groupadd:创建组

-g：指定新建工作组的id；

-r：创建系统工作组，系统工作组的组ID小于500；

-K：覆盖配置文件“/ect/login.defs”；

-o：允许添加组ID号不唯一的工作组。

#### groupmod:更改组

-g<群组识别码>：设置欲使用的群组识别码；

-o：重复使用群组识别码；

-n<新群组名称>：设置欲使用的群组名称。

#### groupdel:删除组

#### gpasswd:为组设定密码

#### chage:修改有效期限

-m：密码可更改的最小天数。为零时代表任何时候都可以更改密码。

-M：密码保持有效的最大天数。

-w：用户密码到期前，提前收到警告信息的天数。

-E：帐号到期的日期。过了这天，此帐号将不可用。

-d：上一次更改的日期。

-i：停滞时期。如果一个密码已过期这些天，那么此帐号将不可用。

-l：例出当前的设置。由非特权用户来确定他们的密码或帐号何时过期。

## 磁盘管理：

#### 文件链接

Ln [-s -v] src dest

硬件连接：

1. 只能对文件创建，不能应用于目录
2. 不能跨文件系统
3. 创建硬链接会增加文件被链接的次数

软链接（符号链接）

1. 可应用于目录
2. 可以跨文件系统
3. 不会增加被链接文件的链接次数
4. 其大小为指定的路径所包含的字符个数

#### Du:查看目录所占的磁盘大小

-s:

#### Df:显示整个磁盘分区的使用情况

## 网络配置：

## 软件管理：

程序的组成部分

1. 二进制程序（bin）
2. 库(lib)
3. 配置文件(etc)
4. 帮助文件(share/man)

/etc、/bin 、/sbin, /lib: 系统启动就需要用到的程序，这些目录不能挂载额外的分区，必须在（/）根文件系统的分区上。

第三方软件安装位置:

/usr/local

bin,lib,etc,etc,sbin等

#### Rpm:

#### yum: