* Linux预设提供了六个命令窗口终端机，分别为tty1,tty2 … tty6。按Ctrl + Alt + F1 ~ F6 切换。默认第一个窗口，也就是tty1
* 使用 man [命令] 查看各个命令的使用文档，如 ：man cp。
* 我的CentOS IP：180.76.150.17
* user1的口令是1-8
* ftp端口22

## 根目录

/ 根目录

* /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：

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中被创建。

* /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，类似于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：

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

* /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。
* 如果一个目录或文件名以一个点 . 开始，表示这个目录或文件是一个隐藏目录或文件(如：.bashrc)。即以默认方式查找时，不显示该目录或文件。

## Vi/Vim

vi 是老式的字处理器，所有的 Unix Like 系统都会内建 vi 编辑器。

Vim是从 vi 发展出来的一个文本编辑器。代码补完、编译及错误跳转等方便编程的功能特别丰富，在程序员中被广泛使用。更偏向于一个程序开发工具而不是文字处理软件。

* 普通模式normal mode

vi file1

如果file1存在，则打开；如果不存在，则创建并打开

打开后进入普通模式

~符号表示没有任何东西

i 进入输入模式，在当前光标字符前插入，insert

a 进入输入模式，在当前光标字符后插入，append

I 进入输入模式，在当前行首插入

A 进入输入模式，在当前行尾插入

o 进入输入模式，在下方开一新行，插入。效果相当于在本行行尾按了enter

O 进入输入模式，在上方开一新行，插入，open

kjhl 分别是上下左右，移动光标。但是用小键盘上下左右也行，我为什么要用这个？

: 切换到命令行模式

v 进入可视模式，选中当前光标字符，

V 进入可视模式，选中当前行Visual Line

Ctrl-v 进入可视模式，能选中一个矩形块Visual Block

u 撤销

p 粘贴

x 删除当前光标处的字符

p 粘贴于当前光标之后paste

P 粘贴于当前光标之前

在普通模式下，有的修改文本的操作需要操作+移动

这样的操作分为

c 删除，并进入插入模式

d 删除

y 复制

移动是上下左右hjkl

dh

删除左边一个字符

d2h

删除左边两个字符，因为单个数字不能构成移动

二连击变成行操作，比如dd删除整行

这就是用来炫技的，复制粘贴整段删除用visual模式就OK

* 输入模式 insert mode

ESC 退出输入模式，切换到普通模式

出问题了，输入模式下小键盘任何数字，都不是代表本身的数字，而是一堆奇奇怪怪的东西。只能避免用小键盘了。

:wq 保存并退出

:q! 不保存退出

* 命令行模式command-line mode

在普通模式下按 : 就进入命令行模式

在命令行模式中，基本的命令有（已经省略了冒号）：

q 退出程序

w 保存文件

ESC 退出命令行模式，回到普通模式

* 可视模式Visual Mode

对于选中的文本：

d 剪切操作，并回到普通模式

y 复制操作，并回到普通模式

p 粘贴操作，并回到普通模式

s 删除当前已选择的内容，并进入插入模式

* Visual、Normal模式都能用

0 跳到行首

^ 跳到行首

$ 跳到行尾

gg 跳到首行

G 跳到末行

w 跳到下一个词首

b 跳到上一个词首

e 跳到下一个词尾

ge 跳到上一个词尾

6l 光标向右移动6个位置。6能换成任意数字，l能换成hjkl上下左右中的按键。直接按6和l既可，不用按enter，底下的命令行也不会有任何显示。

* Visual、Normal、Insert模式都能用

HOME/END 移动光标到行首/行尾

Page Up 向上翻页

Page Down 向下翻页

ctrl+f 向上翻页

ctrl+b 向下翻页

Ctrl + u 向上翻半页

Ctrl + d 向下翻半页

-/+ 光标移动到非空格符的上/下一行

## 常用命令

cat f.txt

查看文件内容

cat -n f1.txt

nl f1.txt

查看f1.txt文件的内容，并且对所有输出行进行编号。

cat -b f1.txt

查看f1.txt文件的内容，用法与-n相似，只不过对于空白行不编号。

cat f1.txt f2.txt

同时显示f1.txt和f2.txt文件内容，文件名之间以空格

more /etc/man\_db.config

--More--(28%) 光标会等待命令

space 向下翻一页

Enter 向下翻一行

/string 在这个显示的内容当中，向下搜寻string这个关键字；

:f 立刻显示出档名以及目前显示的行数；

q 离开 more ，不再显示该文件内容。

b 或 [ctrl]-b 往回翻页

less /etc/man.config

space 向下翻一页

[pagedown] 向下翻一页

[pageup] 向上翻一页

/string 向下搜寻string

? string 向上搜寻string

n 重复前一个搜寻

N 反向的重复前一个搜寻

q 离开 less 这个程序；

* ll或者ls –l

显示文件的属性以及文件所属的用户和组。形如

drwxr-xr-x 2 root root 4096 Dec 8 0213 tempDirectory

第一个字母d表示是一个目录

* d 目录
* - 文件
* l 链接文档(link file)；
* b 装置文件里面的可供储存的接口设备(可随机存取装置)
* c 装置文件里面的串行端口设备，例如键盘、鼠标(一次性读取装置)

第二个rwx确定所有者的同组用户拥有该文件的权限

第三个rwx确定组外其他用户拥有该文件的权限。

r读w写x执行

r4 w2 x1

有权限用字母表示，没有权限用"-"字符表示。

对于 root 用户来说，一般情况下，文件的权限对其不起作用。

chgrp groupname filename

更改属组

chgrp -R groupname filename

递归更改文件属组。在更改某个目录文件的属组时，该目录下的所有文件的属组都会更改。

chown [–R] 属主名 文件名

chown [-R] 属主名：属组名 文件名

更改文件属主，或同时更改属主和属组

进入 /root 目录（~）将install.log的拥有者改为bin这个账号：

等会儿，如果这个属主本来就不是这个属组里边的用户，能瞎jr改么

chmod [-R] 775 文件或目录

更改权限

775表示对于属主是7，即4+2+1，rwx权限都有。对于组外其他用户是5，即4+1，有rx权限。

-R还是表示递归更改。加上后目录下所有文件都变更

chmod u=rwx,g=rx,o=r 文件名

分别给属主user、属组group、other其他用户设置权限。另一种语义化的写法

* ls:

列出目录

ls [options] 目录名称

后面也可以不加目录名称，显示当前目录信息。

-a 列出全部文件，连同隐藏档(开头为 . 的文件)

-d 仅列出目录本身，而不是列出目录内的文件数据。ls -d返回一个 .

-l 列出文件的属性与权限等数据

可以连用多个参数

ls -al ~

* cd：切换目录
* pwd：显示目前的目录

有-P参数，和连结link连结档有关。不懂

* touch test 创建文件test
* mkdir

创建一个新的目录

mkdir -p test1/test2/test3/test4

mkdir -m 711 test2

-p 创建多层目录

-m 配置属性

rmdir a

删除一个空目录

加入有目录结构a🡪b🡪c🡪d🡪e，进入a目录，

rmdir -p b/c/d/e

可以把bcde目录都删掉。当然前提是除了一个子目录外每个目录不含其它文件

* cp:

复制文件或目录

cp aaa tmp

将aaa复制到tmp目录下

cp aaa tmp/bbb

将aaa复制到tmp目录下重命名为bbb

* 也就是说如果destination存在，并且是一个目录，则复制到目录下；如果destination存在且是一个文件，则会询问是否覆盖
* 如果destination不存在，例如形如dir1/dir2，如果dir1不存在，则报错；如果dir1存在，则复制过去并重命名；

-a 即-pdr

-d 若来源档为连结档的属性(link file)，则复制连结档属性而非文件本身；

-i 若destination已经存在，在覆盖时会先询问。默认应该已经有了

-l 进行硬式连结(hard link)的连结档创建，而非复制文件本身；

-p 连同文件的属性一起复制过去，而非使用默认属性。默认应该也是。。。

-r 递归复制，用于复制目录

* rm

移除文件或目录

rm [-fir] 文件或目录

-f force，忽略不存在的文件，不会出现警告信息

-i 互动模式，在删除前会询问使用者是否动作

-r 递归删除，删除目录时用。

* -f存在的时候-i不起作用了。
* 在dir目录下，使用rm -rf，什么都不会发生。得加文件名
* 不能执行rm -rf ./。得移到上一层再删
* rm ./\*删除当前目录所有文件
* mv

移动文件与目录，或修改名称

mv source source2 source3 destination

-f 如果destination已经存在，不询问而直接覆盖；

-i 如果destination已经存在，询问是否覆盖

-u 若目标文件已经存在，且 source 比较新，才会升级 (update)

复制一文件，创建一目录，将文件移动到目录中

mv test test2

也可以用于重命名，如果test2不存在

* tar

tar命令用于备份文件。或者将备份文件还原

tar -xzvf test.tar.gz

解压.tar.gz文件

-x 或--extract或--get，从备份文件中还原文件。

-v 或--verbose 显示指令执行过程。

-f file 或--file=file，指定备份文件。

-z 或--gzip或--ungzip，通过gzip指令处理备份文件。

tar是打包程序，使用tar程序打出来的包称为tar包，tar包文件通常都是以.tar结尾的。

生成tar包后，就可以用其它的程序来进行压缩。

nfboy@ubuntu:$ **ps**

PID TTY TIME CMD

2035 pts/3 00:00:00 bash

2046 pts/3 00:00:00 ps

PID: 运行着的命令(CMD)的进程编号

TTY: 命令所运行的位置（终端）

TIME: 运行着的该命令所占用的CPU处理时间

CMD: 该进程所运行的命令

$ ps -ax

PID TTY STAT TIME COMMAND

1 ? Ss 0:03 /sbin/init auto noprompt

2 ? S 0:00 [kthreadd]

$ ps -ax | less

ps -ef|grep 1181

## 用户

添加新的用户账号

useradd [options] username

-c 注释

-d 指定用户主目录

-m 创建用户主目录

-g 指定用户所属的用户组。可以是gid

-G 指定用户所属的附加组。可以是gid

-s 指定用户的登录Shell。不懂

-u 指定用户的用户号。-o与-u同时使用，表示可以重复使用其他用户的标识号。

useradd –d /home/sam -m sam

创建一个用户sam，其中-d和-m选项用来为sam产生一个主目录 /home/sam。

home为默认用户主目录所在的父目录，所以这一命令等于

useradd sam

-m参数表示如果用户的主目录不存在，则创建它。skeleton目录(/etc/skel)中包含的文件将被复制到主目录。在CREATE\_HOME为no时，则创建用户不会创建主目录。但是我的CentOS默认CREATE\_HOME为yes，自动创建主目录。CREATE\_HOME在/etc/login.defs中设置。可能在其他系统中CREATE\_HOME参数默认为no，加上-m保险一点。

useradd -s /bin/sh -g group1 –G group2,root gem

此命令新建了一个用户gem，该用户的登录Shell是 /bin/sh

它属于group1用户组，同时又属于group2和root用户组，其中group1用户组是其主组。

group1和group2必须存在

用户标识号是一个整数，系统内部用它来标识用户。一般情况下它与用户名一一对应。如果几个用户名对应的用户标识号是一样的，系统内部将把它们视为同一个用户，但是它们可以有不同的口令、不同的主目录以及不同的登录Shell等。

userdel -r sam

删除一个用户

-r表示连同用户主目录也删除

usermod -s /bin/ksh -d /home/z –g developer sam

修改用户属性。

创建的时候的属性都可以修改。有些系统不能改名

passwd [options] [username]

-l 锁定口令，即禁用账号。

-u 口令解锁。

-d 使账号无口令。

-f 强迫用户下次登录时修改口令。

如果不写用户名，则修改当前用户的口令。

用户账号刚创建时没有口令，被系统锁定，无法使用。必须为其指定口令后才可以使用

超级用户可以为自己和其他用户指定口令，普通用户只能用它修改自己的口令

设置完口令之后，就能用Xshell新建连接以用户的身份登录啦！[root@instance-5hguojbs home]就变成[user1@instance-5hguojbs home]

当前用户user1修改自己的口令：

passwd

超级用户指定任何用户的口令：

passwd sam

普通用户修改自己的口令时，passwd命令会先询问原口令；而超级用户为用户指定口令时，不需要知道原口令。

passwd -d sam

将用户 sam 的口令删除，这样用户 sam 下一次登录时，系统就不再允许该用户登录了。

passwd -l sam

用 -l(lock) 选项锁定某一用户，使其不能登录

groupadd [options] groupname

-g 指定新用户组的组标识号GID。

-o 与-g同时使用，表示新用户组的GID可以与系统已有用户组的GID相同。

groupadd group1

增加用户组group1。新组的组标识号是在当前已有的最大组标识号的基础上加1。

groupadd -g 101 group2

增加用户组group2，同时指定组标识号是101。

groupdel group1

删除组group1

groupmod [options] groupname

修改用户组的属性

-g 为用户组指定新的组标识号。

-o 与-g同时使用，表示用户组的新GID可以与系统已有用户组的GID相同。

-n 重命名用户组

groupmod -g 102 group2

将group2的标识号修改为102。

groupmod –g 10000 -n group3 group2

将group2的组标识号改为10000，组名修改为group3。

切换用户组

newgrp root

将当前用户切换到root用户组，前提条件是root用户组确实是该用户的主组或附加组

* /etc/passwd中一行记录对应着一个用户，形如

root:x:0:0:root:/root:/bin/bash

user1:x:1000:1000::/home/user1:/bin/bash

* 每行记录被冒号(:)分隔为7个字段，具体含义如下：

username:口令:uid:gid:注释性描述:主目录:登录Shell

* 当一个用户同时是多个组中的成员时，在/etc/passwd文件中记录的是用户所属的主组，也就是登录时所属的默认组，而其他组称为附加组。
* 用户要访问属于附加组的文件时，必须首先使用newgrp命令使自己成为所要访问的组中的成员。
* /etc/passwd文件对所有用户都可读，为了安全，/etc/passwd里的口令字段都是x
* 主目录，也就是用户的起始工作目录，即用户在登录到系统之后所处的目录。各用户对自己的主目录有读、写、执行（搜索）权限，其他用户对此目录的访问权限则根据具体情况设置。
* 用户登录后，要启动一个进程，负责将用户的操作传给内核，这个进程是用户登录到系统后运行的命令解释器或某个特定的程序，即Shell。Shell是用户与Linux系统之间的接口。Linux的Shell有许多种，每种都有不同的特点。常用的有sh(Bourne Shell), csh(C Shell), ksh(Korn Shell), tcsh(TENEX/TOPS-20 type C Shell), bash(Bourne Again Shell)等。说了半天，我还是不懂。系统管理员可以根据系统情况和用户习惯为用户指定某个Shell。如果不指定Shell，那么系统使用sh为默认的登录Shell，即这个字段的值为/bin/sh。
* 用户组的所有信息都存放在/etc/group文件中。每个组对应一行记录，形如

root:x:0:

user1:x:1000:

groupname:口令:gid:组内用户列表

* 一般Linux 系统的用户组都没有口令，即这个字段一般为空，或者是\*。我的centOS是x
* "组内用户列表"是属于这个组的所有用户的列表，不同用户之间用逗号(,)分隔。这个用户组可能是用户的主组，也可能是附加组。我的centOS没有组内用户列表
* /etc/shadow存的是真正的口令，但是也加密过了
* Creating mailbox file: File exists

添加用户后,会在系统里自动加一个邮箱(系统邮箱),路径是/var/spool/mail/用户名.

如果删除user时没有使用-r命令，则这个文件会保存下来。没删干净

如果再次添加同名的user就会出现这个错误

## 磁盘

检查文件系统的磁盘空间占用情况。可以利用该命令来获取硬盘被占用了多少空间，目前还剩下多少空间等信息。

df [-ahikHTm] [目录或文件名]

-a 列出所有的文件系统，包括系统特有的 /proc 等文件系统；

-k 以 KBytes 的容量显示各文件系统；

-m 以 MBytes 的容量显示各文件系统；

-h 以人们较易阅读的 GBytes, MBytes, KBytes 等格式自行显示；

-H 以 M=1000K 取代 M=1024K 的进位方式；

-T 显示文件系统类型, 连同该 partition 的 filesystem 名称 (例如 ext3) 也列出；

-i 不用硬盘容量，而以 inode 的数量来显示

df

df -h

df -aT

df -h /etc

du：检查磁盘空间使用量

fdisk：用于磁盘分区

有这么几种用法，列出来我也看不懂，也就不记了，应该用不到。反正知道有这么一个命令能查看磁盘状态就行了

## yum命令

* yum Yellow dog Updater是一个Shell前端软件包管理器
* 基于RPM包管理，能够从指定的服务器自动下载RPM包并且安装，可以自动处理依赖性关系，并且一次安装所有依赖的软体包
* yum提供了查找、安装、删除某一个、一组甚至全部软件包的命令

yum [options] [command] [package ...]

options包括-h（帮助），-y（当安装过程提示选择全部为"yes"），-q（不显示安装的过程）等等。

command：要进行的操作。

package操作的对象。

* 列出所有可更新的软件清单命令：yum check-update
* 更新所有软件命令：yum update
* 仅安装指定的软件命令：yum install <package\_name>
* 仅更新指定的软件命令：yum update <package\_name>
* 列出所有可安裝的软件清单命令：yum list
* 删除软件包命令：yum remove <package\_name>
* 查找软件包 命令：yum search <keyword>

清除缓存命令:

* yum clean packages: 清除缓存目录下的软件包
* yum clean headers: 清除缓存目录下的 headers
* yum clean oldheaders: 清除缓存目录下旧的 headers
* yum clean, yum clean all (= yum clean packages; yum clean oldheaders) :清除缓存目录下的软件包及旧的headers

## centOS

查看版本

rpm -q centos-release

centos-release-7-3.1611.el7.centos.x86\_64

cat /etc/redhat-release

CentOS Linux release 7.3.1611 (Core)

## 远程登录

通过SSH Secure Shell服务实现

SSH服务默认22端口。这tm不是sftp的端口

通过SSH，可以把所有传输的数据进行加密，也能够防止DNS欺骗和IP欺骗。

SSH之所以能够保证安全，原因在于它采用了非对称加密技术(RSA)加密了所有传输的数据

第一种，基于口令的安全验证：只要你知道自己帐号和口令，就可以登录到远程主机。

所有传输的数据都会被加密，但是有可能受到中间人攻击，有假货冒充真正的服务器。

第二种，基于密钥的安全验证

* 首先为自己创建一对密钥
* 把公钥放在需要访问的服务器上。
* 如果要连接到SSH服务器上，客户端就会向服务器发出请求，请求用你的密钥进行安全验证。
* 服务器收到请求之后，先在该服务器上你的主目录下寻找你的公钥，然后把它和你发送过来的公钥进行比较。
* 如果两个密钥一致，服务器就用公钥加密“质询”(challenge)并把它发送给客户端软件。客户端软件收到“质询”之后就可以用你的私钥在本地解密再把它发送给服务器完成登录。

具体步骤：

* 打开Xshell，点击工具——新建用户密钥生成向导
* 密钥类型RSA，密钥长度2048位。
* 一直点下一步，到设置密钥名称，默认即可。
* 设置给用户密钥加密的密码，这个相当于本地的，和传输没有关系，设为空也行。和服务器上root的密码也没有关系，感觉相当于手机锁屏密码。
* 然后保存这个公钥。
* 在SSh服务器上注册这个公钥。此时还没有完成密钥验证，怎么登录服务器呢？当然是用第一种，基于口令的安全验证啦
* 新建连接，SSH协议，创建链接——接下来会提示你输入用户名密码，用户名是root
* 进入后打开/root/.ssh/authorized\_keys，把刚刚保存的公钥全部复制进去。
* 然后返回xshell，新建连接，用密钥登录就OK。以后链接就不用输密码了