## 查看文件信息：ls

Linux文件或者目录名称最长可以有265个字符，“.”代表当前目录，“..”代表上一级目录，以“.”开头的文件为隐藏文件，需要用 -a 参数才能显示。

### 语法说明

ls [options] [路径] 如果没路径 就代表显示当前所在的文件

|  |
| --- |
| 命令名称：ls  命令英文原意：list  命令所在路径：/bin/ls  执行权限：所有用户  功能描述：显示目录文件和文件夹  语法：ls 选项[-ald] [文件或目录] 可以通过 ls --help  -a 显示指定目录下所有子目录与文件，包括隐藏文件  -l 以列表方式显示文件的详细信息  -h 配合 -l 以人性化的方式显示文件大小  -d 可以查看当前目录的属性  案例  ls -a 显示当前目录下的所有文件包含隐藏文件  ls -al 显示当前目录下的所有文件包含隐藏文件以列表方式显示文件的详细信息  ls -alh 显示当前目录下的所有文件包含隐藏文件以列表方式显示文件的详细信息和大小  ls -dl 查看当前文件夹的属性  ll = ls -l显示当前文件夹下面的所有内容 |

### 通配符

| **通配符** | **含义** |
| --- | --- |
| \* | 文件代表文件名中所有字符 |
| ls te\* | 查找以te开头的文件 |
| ls \*html | 查找结尾为html的文件 |
| ？ | 代表文件名中任意一个字符 |
| ls ?.c | 只找第一个字符任意，后缀为.c的文件 |
| ls a.? | 只找只有3个字符，前2字符为a.，最后一个字符任意的文件 |
| [] | [”和“]”将字符组括起来，表示可以匹配字符组中的任意一个。“-”用于表示字符范围。 |
| [abc] | 匹配a、b、c中的任意一个 |
| [a-f] | 匹配从a到f范围内的的任意一个字符 |
| ls [a-f]\* | 找到从a到f范围内的的任意一个字符开头的文件 |
| ls a-f | 查找文件名为a-f的文件,当“-”处于方括号之外失去通配符的作用 |
| \ | 如果要使通配符作为普通字符使用，可以在其前面加上转义字符。“?”和“\*”处于方括号内时不用使用转义字符就失去通配符的作用。 |
| ls \\*a | 查找文件名为\*a的文件 |

## cd

|  |
| --- |
| cd / 进入用户的主目录。  cd test 进入当前目录下的test目录 【这个是相对目录】  cd /root/test 进入root目录下的test目录 【这个是绝对目录】  cd ~ 切换到当前用户的主目录(/home/用户目录)  cd . 切换到当前目录  cd .. 切换到上级目录  cd - 可进入上次所在的目录 |

## 7，拷贝文件：cp

cp命令的功能是将给出的文件或目录复制到另一个文件或目录中，相当于DOS下的copy命令。  
常用选项说明：

|  |  |
| --- | --- |
| **选项** | **含义** |
| -a | 该选项通常在复制目录时使用，它保留链接、文件属性，并递归地复制目录，简单而言，保持文件原有属性。 |
| -n | 已经存在的目标文件而不提示 |
| -i | 交互式复制，在覆盖目标文件之前将给出提示要求用户确认 |
| -r | 若给出的源文件是目录文件，则cp将递归复制该目录下的所有子目录和文件，目标文件必须为一个目录名。 |
| -v | 显示拷贝进度 |

### 语法说明

命令名称：cp

命令英文原意：copy

命令所在路径：/bin/cp

执行权限：所有用户

语法：cp -R [源文件或目录] [目的目录]

-R 复制目录

功能描述：复制文件或目录 ,默认的cp只能复制文件，不能是目录 如果是目录要使用-r

案例

cp -r /root/mywork /root/helloworld 把mywork复制到helloworld里面

cp - /root/install.log /root/helloworld 把install.log文件复制到helloworld里面

cp -rvif /root/install.log /root/helloworld 把install.log文件复制到helloworld里面 并显示进度

### 特别注意

如果是拷贝文件夹 必须加 -r的选项

## 8，移动文件：mv

用户可以使用mv命令来移动文件或目录，也可以给文件或目录重命名。

### 常用选项说明：

| **选项** | **含义** |
| --- | --- |
| -f | 禁止交互式操作，如有覆盖也不会给出提示 |
| -v | 显示移动进度 |

### 语法说明

|  |
| --- |
| 命令名称：mv  命令英文原意：move  命令所在路径：/bin/mv  执行权限：所有用户  语法：mv [源文件或目录] [目的目录]  功能描述：移动文件、更名  案例  mv /root/install.log /root/mywork/install.log 把install.log移动到mywork里面  mv /root/install.log /root/ins.log 把install.log改成ins.log |

## 查看或合并文件：cat

cat 命令用于连接文件并打印到标准输出设备上。

常用参数及含义如下表所示：

|  |  |
| --- | --- |
| **参数** | **含义** |
| -n | 显示文件里数据的行号 |

|  |
| --- |
| 命令名称：cat  命令英文原意：concatenate and display files  命令所在路径：/bin/cat  执行权限：所有用户  语法：cat [文件名]  功能描述：显示文件内容  范例：  cat install.log 查看当前目录下的install.log文件  cat /root/install.log 查看绝度目录下的文件  cat test1.txt test2.txt >>test3.txt 把test1.txt test2.txt里面的内容追加到test3里面  cat test1.txt test2.txt > test3.txt 先清空test3.txt再把test1.txt test2.txt里面的内容追加到test3里面  > >>的区别是前者先清空再追加 后者直接追加 |

## 15，建立链接文件：ln

Linux链接文件类似于Windows下的快捷方式。

链接文件分为软链接和硬链接。

### 区别：

软链接：软链接不占用磁盘空间，源文件删除则软链接失效。

硬链接：硬链接只能链接普通文件，不能链接目录，会同步更新。

### 使用格式：

~~ln 源文件 链接文件~~

ln -s 源文件 链接文件 --就是一个wind里面快捷方式

如果没有-s选项代表建立一个硬链接文件，两个文件占用相同大小的硬盘空间，即使删除了源文件，链接文件还是存在，所以-s选项是更常见的形式。

注意：如果软链接文件和源文件不在同一个目录，源文件要使用绝对路径，不能使用相对路径。

### 语法说明

命令名称：ln

命令英文原意：link

命令所在路径：/bin/ln

执行权限：所有用户

语法：ln -s [源文件] [目标文件]

-s 创建软链接

功能描述：产生链接文件

案例

ln /etc/issue /root/etc/issue.bak.hard 硬链接

|--如果删除了issue文件 issue.bak.hard可以使用

可以使用命令echo 'www.whsxt.com' >> /etc/issue去修改issue文件

看看issue.bak.hard里面内容是否更新了

ln -s /etc/issue /root/etc/issue.soft 软链接

|--如果删除了etc/issue文件，那么issue.soft就失效了

ls -ilh可以查看文件的硬盘地址，可以看到硬链接的地址的原文件的地址一样

这也就是为什么能同步更新的原因了

## 18，文件搜索命令：find

### 作用：查找文件

find命令功能非常强大，通常用来在特定的目录下搜索符合条件的文件，也可以用来搜索特定用户属主的文件。

### 常用用法：

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | **命令** | **含义** | | --- | --- | | find  test.sh | 查找当前目录下所有名为test.sh的文件 | | find  \*.sh | 查找当前目录下所有后缀为.sh的文件 | | find  [A-Z]\* | 查找当前目录下所有以字母开头的文件 | | find /tmp -size 2M | 查找在/tmp 目录下等于2M的文件 | | find /tmp -size +2M | 查找在/tmp 目录下大于2M的文件 | | find /tmp -size -2M | 查找在/tmp 目录下小于2M的文件 | | find -size +4k -size -5M | 查找当前目录下大于4k，小于5M的文件 | | find  -perm 777 | 查找当前目录下权限为 777 的文件或目录 | |

### 语法说明

|  |
| --- |
| 命令名称：find  命令所在路径：/usr/bin/find  执行权限：所有用户  语法：find [搜索路径] [搜寻关键字]  功能描述：查找文件或目录  $ find /etc -name init 在目录/etc中查找文件init  $ find / -size +204800 在根目录下查找大于100MB的文件  $ find / -user sam 在根目录下查找所有者为sam的文件  $ find /etc -ctime -1 在/etc下查找24小时内被修改过属性的文件和目录  $ find /etc -size +163840 -a -size -204800 在/etc下查找大于80MB小于100MB 的文件  $ find /etc -name inittab -exec ls -l {} \; 在/etc下查找inittab文件并显示其详细信息 |

## 19，文件搜索命令：grep

### 搜索文件里面的内容的命令

Linux系统中grep命令是一种强大的文本搜索工具，grep允许对文本文件进行模式查找。如果找到匹配模式， grep打印包含模式的所有行。

grep一般格式为：

grep [-选项] ‘搜索内容串’文件名

在grep命令中输入字符串参数时，最好引号或双引号括起来。例如：grep‘a ’1.txt。

常用选项说明：

|  |  |
| --- | --- |
| 选项 | 含义 |
| -v | 显示不包含匹配文本的所有行（相当于求反） |
| -n | 显示匹配行及行号 |
| -i | 忽略大小写 |

grep搜索内容串可以是正则表达式。

正则表达式是对字符串操作的一种逻辑公式，就是用事先定义好的一些特定字符、及这些特定字符的组合，组成一个“规则字符串”，这个“规则字符串”用来表达对字符串的一种过滤逻辑。

### grep常用正则表达式

|  |  |
| --- | --- |
| **参数** | **含义** |
| ^a | 行首,搜寻以 a 开头的行；grep -n '^a' 1.txt |
| ke$ | 行尾,搜寻以 ke 结束的行；grep -n 'ke$' 1.txt |
| [Ss]igna[Ll] | 匹配 [] 里中一系列字符中的一个；搜寻匹配单词signal、signaL、Signal、SignaL的行；grep -n '[Ss]igna[Ll]' 1.txt |
| . | (点)匹配一个非换行符的字符；匹配 e 和 e 之间有任意一个字符，可以匹配 eee，eae，eve，但是不匹配 ee，eaae；grep -n 'e.e' 1.txt |

### 语法说明

命令名称：grep

命令所在路径：/bin/grep

执行权限：所有用户

语法：grep [指定字串] [源文件

功能描述：在文件中搜寻字串匹配的行并输出

范例：# grep ftp /etc/services

|--在/etc/services文件里面找有没有ftp这个一个字符串

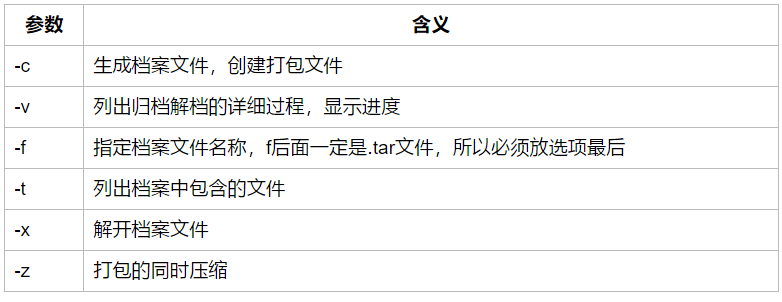
## 20，归档管理：tar 【重点掌握】

计算机中的数据经常需要备份，tar是Unix/Linux中最常用的备份工具，此命令可以把一系列文件归档到一个大文件中，也可以把档案文件解开以恢复数据。

tar使用格式 tar [参数] 打包文件名 文件

tar命令很特殊，其参数前面可以使用“-”，也可以不使用。

### 常用参数：



注意：除了f需要放在参数的最后，其它参数的顺序任意。

### 语法说明

|  |
| --- |
| 命令名称：tar  命令所在路径：/bin/tar  执行权限：所有用户  语法：tar 选项[cvf] [目录]  -c 产生.tar打包文件  -v 显示详细信息  -f 指定压缩后的文件名  -z 打包同时压缩  -x 解压  功能描述：打包目录  压缩后文件格式：.tar.gz  案例  tar -zcvf test.tar.gz test 把test文件或文件夹压缩成test.tar.gz  tar -zxvf test.tar.gz 解压test.tar.gz  tar -zxvf xxx.tar.gz -C ./dir1 #解压到指定目录 |

# 10【掌握】Linux命令-用户、权限管理

## 查看当前用户：whoami

## 退出登录账户： exit

如果是切换后的登陆用户，退出则返回上一个登陆账号。

## 添加用户账号：useradd

在Unix/Linux中添加用户账号可以使用adduser或useradd命令，因为adduser命令是指向useradd命令的一个链接，因此，这两个命令的使用格式完全一样。

useradd命令的使用格式如下： useradd [参数] 新建用户账号

| **参数** | **含义** |
| --- | --- |
| -d | 指定用户登录系统时的主目录，如果不使用该参数，系统自动在/home目录下建立与用户名同名目录为主目录 |
| -m | 自动建立目录 |
| -g | 指定组名称 |

### 相关说明：

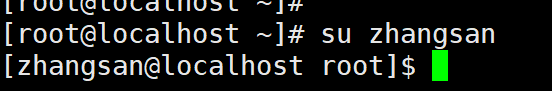
* Linux每个用户都要有一个主目录，主目录就是第一次登陆系统，用户的默认当前目录(/home/用户)；
* 每一个用户必须有一个主目录，所以用useradd创建用户的时候，一定给用户指定一个主目录；
* 用户的主目录一般要放到根目录的home目录下，用户的主目录和用户名是相同的；
* 如果创建用户的时候，不指定组名，那么系统会自动创建一个和用户名一样的组名。

| **命令** | **含义** |
| --- | --- |
| useradd -d /home/abc abc -m | 创建abc用户，如果/home/abc目录不存在，就自动创建这个目录，同时用户属于abc组 |
| useradd -d /home/a a -g test -m | 创建一个用户名字叫a，主目录在/home/a，如果主目录不存在，就自动创建主目录，同时用户属于test组 |
| cat /etc/passwd | 查看系统当前用户名 |

## 设置用户密码：passwd

## 切换用户：su

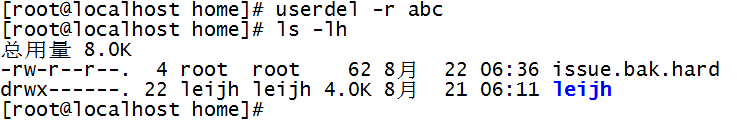
可以通过su命令切换用户，su后面可以加“-”。su和su –命令不同之处在于，su -切换到对应的用户时会将当前的工作目录自动转换到切换后的用户主目录：



| **命令** | **含义** |
| --- | --- |
| su | 切换到root用户 |
| su root | 切换到root用户 |
| su - | 切换到root用户，同时切换目录到/root |
| su - root | 切换到root用户，同时切换目录到/root |
| su 普通用户 | 切换到普通用户 |
| su - 普通用户 | 切换到普通用户，同时切换普通用户所在的目录 |

## 8，删除用户：userdel

| **命令** | **含义** |
| --- | --- |
| userdel abc(用户名) | 删除abc用户，但不会自动删除用户的主目录 |
| userdel -r abc(用户名) | 删除用户，同时删除用户的主目录 |



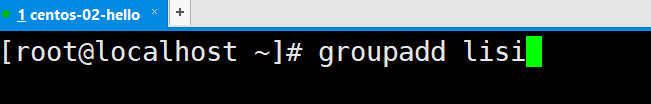
## 9，查看有哪些用户组

命令

cat /etc/group

## 10，添加、删除组账号：groupadd、groupdel

### groupadd 添加组

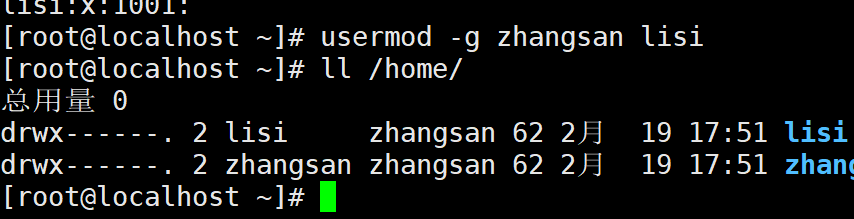


### groupdel 删除组



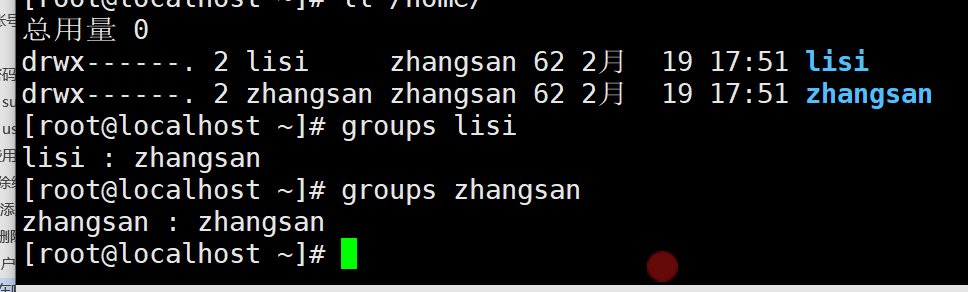
### 11，修改用户所在组：usermod

使用方法：usermod -g 用户组 用户名



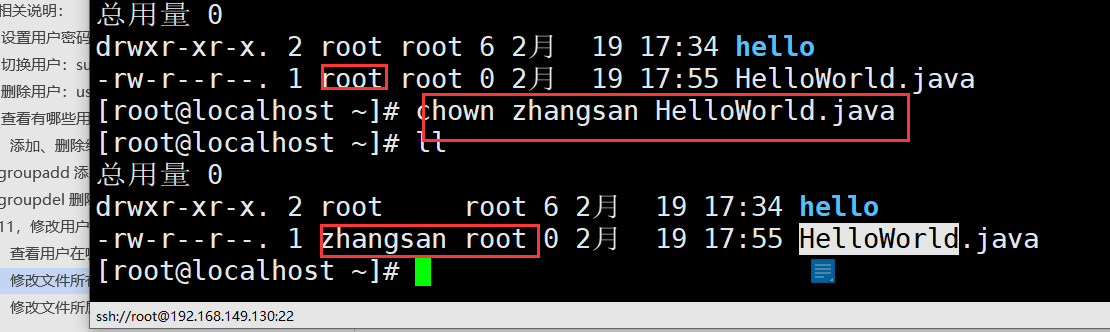
## 12，查看用户在哪些组

groups 用户名

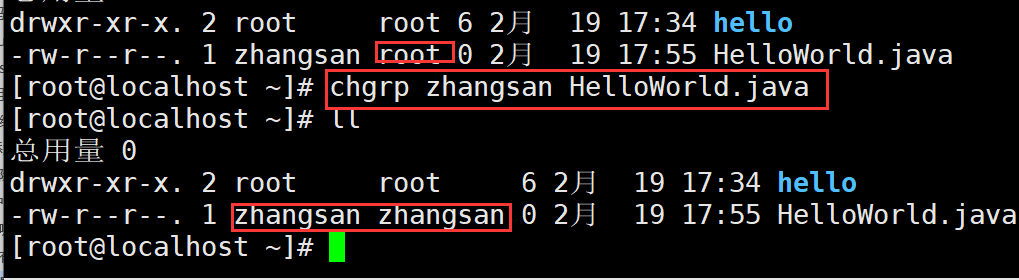


## 13，修改文件所有者：chown

注意切换到root用户



## 14，修改文件所属组：chgrp



## 15，修改文件权限：chmod



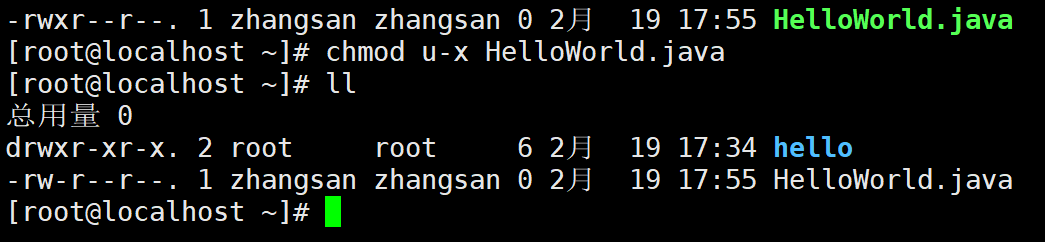
chmod 修改文件权限有两种使用格式：字母法与数字法。

### 字母法：chmod u/g/o/a +/-/= rwx 文件

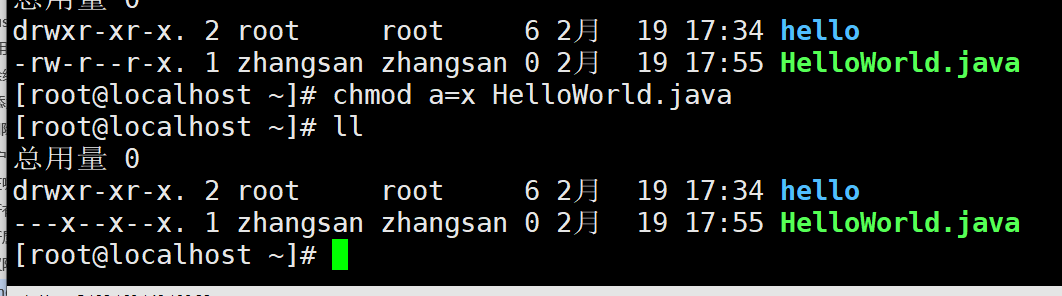
| **[ u/g/o/a ]** | **含义** |
| --- | --- |
| u | user 表示该文件的所有者 |
| g | group 表示与该文件的所有者属于同一组( group )者，即用户组 |
| o | other 表示其他以外的人 |
| a | all 表示这三者皆是 |

| **[ +-= ]** | **含义** |
| --- | --- |
| + | 增加权限 |
| - | 撤销权限 |
| = | 设定权限 |

| **rwx** | **含义** |
| --- | --- |
| r | read 表示可读取，对于一个目录，如果没有r权限，那么就意味着不能通过ls查看这个目录的内容。 |
| w | write 表示可写入，对于一个目录，如果没有w权限，那么就意味着不能在目录下创建新的文件。 |
| x | excute 表示可执行，对于一个目录，如果没有x权限，那么就意味着不能通过cd进入这个目录。 |



chmod u-x HelloWorld.java 给HelloWorld.java文件的用户去掉一个x权限



### 数字法：“rwx” 这些权限也可以用数字来代替

| **字母** | **说明** |
| --- | --- |
| r | 读取权限，数字代号为 "4" |
| w | 写入权限，数字代号为 "2" |
| x | 执行权限，数字代号为 "1" |
| - | 不具任何权限，数字代号为 "0" |

chmod 751 file：

* 文件所有者：读、写、执行权限
* 同组用户：读、执行的权限
* 其它用户：执行的权限

语法

chmod 777 HelloWorld.java

把HelloWorld.java的权限改 rwxrwxrwx

chmod 121 HelloWorld.java

把HelloWorld.java的权限改 –x-w---x

chmod 721 HelloWorld.java

把HelloWorld.java的权限改rwx-w---x

# 11【掌握】Linux命令-系统管理命令

## 2，显示或设置时间：date

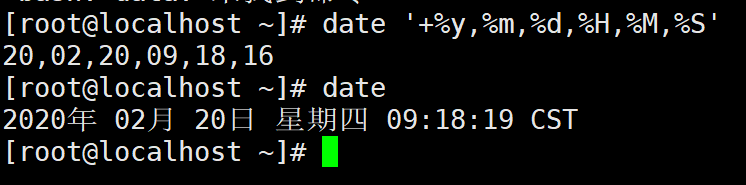
设置时间格式（需要管理员权限）：

date [MMDDhhmm[[CC]YY][.ss]] +format

CC为年前两位yy为年的后两位，前两位的mm为月，后两位的mm为分钟，dd为天，hh为小时，ss为秒。如： date 010203042016.55。

显示时间格式（date '+%y,%m,%d,%H,%M,%S'）：

| **format格式** | **含义** |
| --- | --- |
| %Y，%y | 年 |
| %m | 月 |
| %d | 日 |
| %H | 时 |
| %M | 分 |
| %S | 秒 |



## 3，查看进程信息：ps

进程是一个具有一定独立功能的程序，它是操作系统动态执行的基本单元。

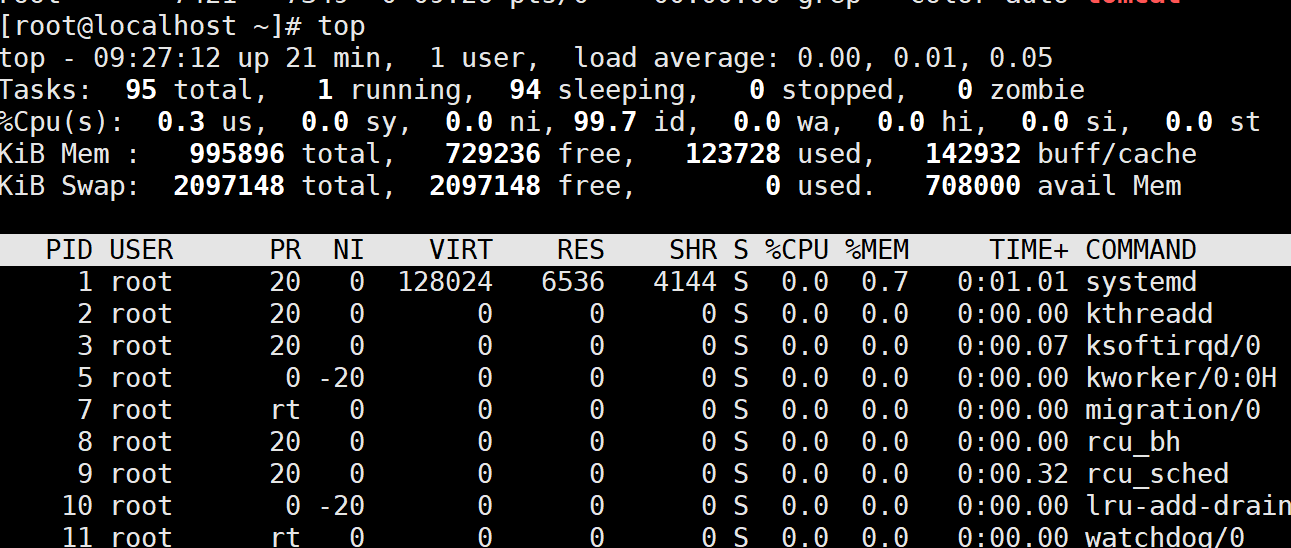
ps命令可以查看进程的详细状况，常用选项(选项可以不加“-”)如下：

| **选项** | **含义** |
| --- | --- |
| -a | 显示终端上的所有进程，包括其他用户的进程 |
| -u | 显示进程的详细状态 |
| -x | 显示没有控制终端的进程 |
| -w | 显示加宽，以便显示更多的信息 |
| -r | 只显示正在运行的进程 |

## 4，动态显示进程top

加上-d 来指定显示信息更新的时间间隔。

| **按键** | **含义** |
| --- | --- |
| M | 根据内存使用量来排序 |
| P | 根据CPU占有率来排序 |
| T | 根据进程运行时间的长短来排序 |
| U | 可以根据后面输入的用户名来筛选进程 |
| K | 可以根据后面输入的PID来杀死进程。 |
| q | 退出 |
| h | 获得帮助 |



### 如何查询进程ID

ps -ef|grep java

top

ps -a

jps 查询所有的java进程

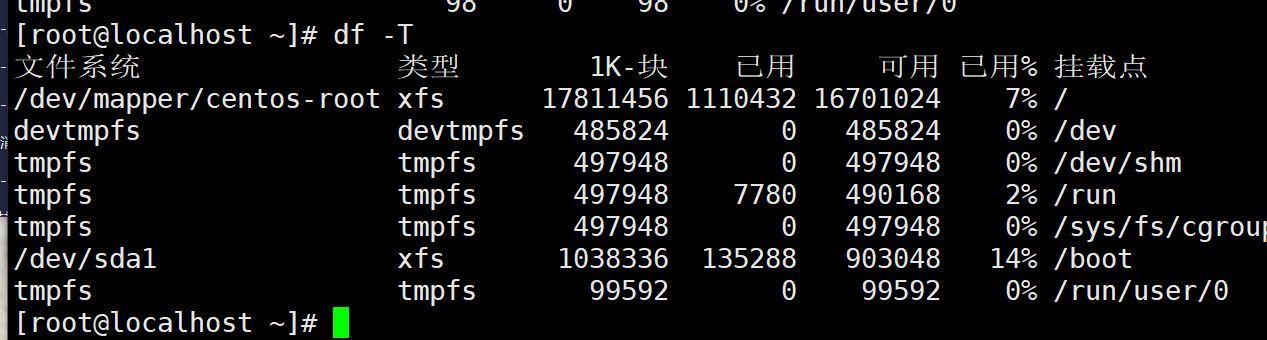
## 关机重启：reboot、shutdown、init

| **命令** | **含义** |
| --- | --- |
| reboot | 重新启动操作系统 |
| shutdown –r now | 重新启动操作系统，shutdown会给别的用户提示 |
| shutdown -h now | 立刻关机，其中now相当于时间为0的状态 |
| shutdown -h 20:25 | 系统在今天的20:25 会关机 |
| shutdown -h +10 | 系统再过十分钟后自动关机 |
| init 0 | 关机 |
| init 6 | 重启 |

## 7，检测磁盘空间：df

df命令用于检测文件系统的磁盘空间占用和空余情况，可以显示所有文件系统对节点和磁盘块的使用情况。

| **选项** | **含义** |
| --- | --- |
| -a | 显示所有文件系统的磁盘使用情况 |
| -m | 以1024字节为单位显示 |
| -t | 显示各指定文件系统的磁盘空间使用情况 |
| -T | 显示文件系统 |



# 12【掌握】Linux命令-网络及组合命令

### yum 在线安装某些软件

yum update package1 更新指定程序包package1

yum check-update 检查可更新的程序

yum list 显示所有已经安装和可以安装的程序包

yum search jdk 在软件库查询jdk的安装包

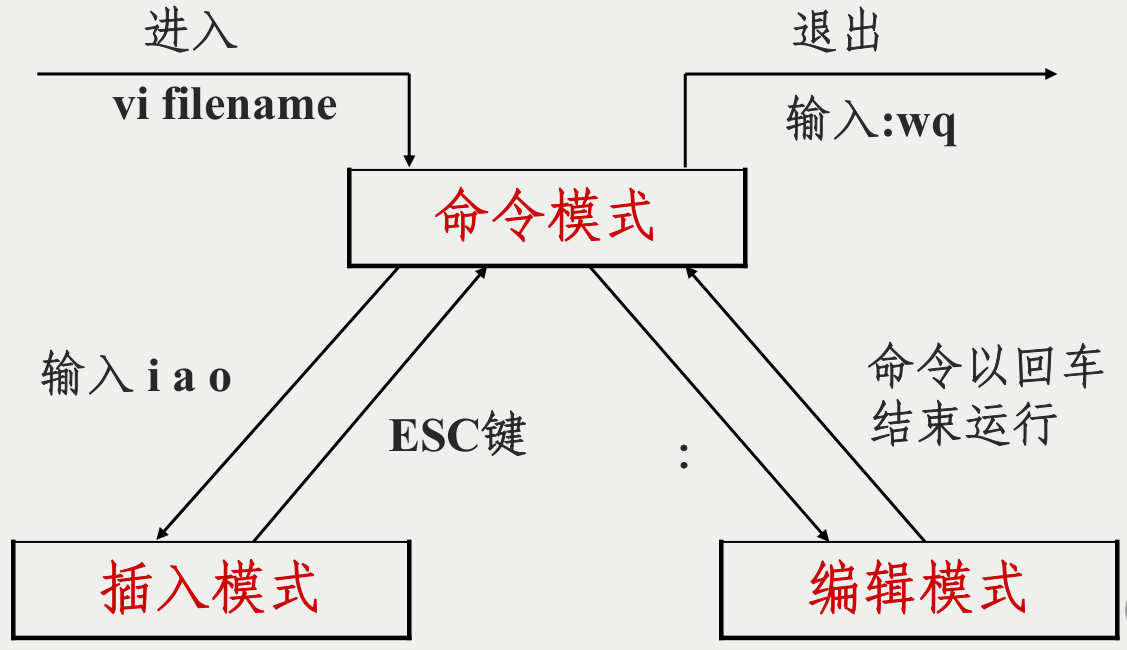
# VI/VIM编辑器的使用

## Vim/Vi 简介

### vim和vi的区别

Vim 编辑文件本xml文本时会有高亮显示

## 2，Vim/Vi 工作模式



## 3，插入命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| a | 在光标后附加文本 |
| A（shift + a） | 在本行行末附加文本   行尾 |
| i | 在光标前插入文本 |
| I(shift+i) | 在本行开始插入文本 行首 |
| o | 在光标下插入新行 |
| O(shift+o) | 在光标上插入新行 |

## 4，定位命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| :set nu | 设置行号 |
| :setnonu | 取消行号 |
| gg  G | 到第一行  到最后一行 |
| nG | 到第n行 |
| :n | 到第n行 |

|  |  |
| --- | --- |
| $ | 移至行尾 |
| 0 | 移至行首 |
| h 或者方向左键 | 左移一个字符 |
| j 或者方向下键 | 下移一行 |
| k 或者方向上键 | 上移一行 |
| l 或者方向右键 | 右移一个字符 |
| **H** | **移至屏幕上端** |
| **M** | **移至屏幕中央** |
| **L** | **移至屏幕下端** |

## 5，删除命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| x | 删除光标所在处字符  nx 删除光标所在处后n个字符 |
| dd | 删除光标所在行，ndd删除n行 |
| :n1,n2d | 删除指定范围的行（eg :1,3d   删除了123这三行） |
| dG | 删除光标所在行到末尾的内容 |
| D | 删除从光标所在处到行尾 |

## 6，复制和剪切命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| yy、Y | 复制当前行 |
| nyy、nY | 复制当前行以下n行 |
| dd | 剪切当前行 |
| ndd | 剪切当前行以下n行 |
| p、P | 粘贴在当前光标所在行下  或行上 |

## 7，替换和取消命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| r | 取代光标所在处字符 |
| R(shift + r) | 从光标所在处开始替换字符，按Esc结束 |
| u | undo,取消上一步操作 |

## 8，搜索和替换命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| /string | 向后搜索指定字符串   搜索时忽略大小写 :set ic |
| ?string | 向前搜索指定字符串 |
| :%s/old/new/g | 全文替换指定字符串 |
| :n1,n2s/old/new/g | 在一定范围内替换指定字符串 |

% 指全文，s 指开始，g 指全局替换

    起始行，终止行s/要替换的字符串/替换的新的字符串/g c 询问确认

|  |
| --- |
| eg: :%s/ftp/yang/g 全局替换，把ftp替换为yang  eg: :41,44s/yang/lee/c 从41行到44行，把yang替换为lee,询问是否替换  eg: :41,44s/yang/lee/g 同上，不询问，直接替换  eg: :%s/\/root/\/ROOT/g 把/root替换为/ROOT  eg: :%s#/bin/bash#/bin/ksh/g 把/bin/bash全部替换为/bin/ksh |

## 9，保存和退出命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| :w new\_filename | 另存为指定文件 |
| :w >> a.txt | 内容追加到a.txt文件中 文件需存在 |
| :wq | 保存修改并退出 |
| shift+zz（ZZ） | 快捷键，保存修改并退出 |