Linux150个基础命令

1.mkdir 创建目录 \*

主要选项：-p 递归创建目录

主要案列：mkdir /jingfeng

mkdir -p /jingfeng/jf

mkdir -p /jingfeng/jf{1,2} #创建多个目录

2.cd 切换路径 \*

主要案例：cd或cd ~ #切换至家目录

cd - #在上一个目录和当前目录来回切换

cd ~jingfeng #切换至某用户的家目录

cd .. #切换至上一级目录

3.ls 查看目录或文件信息 \*

主要选项：-l 列出目录或者文件的详细信息。比如权限、修改时间等等

-a 列出当前目录下所有文件，包括隐藏文件（已点开头的都是隐藏文件）

-A 列出除.和..之外的所有文件

-h 已人类能查看的方式，列出文件的大小

-d 只列出目录的详细信息

-F给不同文件结尾加不同标识

-p给目录结尾加斜线

-t按修改时间排序

-r反转排序

-i查看inode节点

--time-style=long-iso 格式化属性时间

-R 递归

主要案例：1、ls -l #列出目录或者文件的详细信息

-rw-r--r-- 1 root root 1 Aug 5 21:24 jingfeng.txt

1. ls -la #列出当前目录下所有文件，包括隐藏文件

-rw------- 1 root root 18029 Aug 11 22:05 .bash\_history

-rw-r--r-- 1 root root 18 Dec 8 2016 .bash\_logout

-rw-r--r-- 1 root root 176 Dec 8 2016 .bash\_profile

1. ls -lh #已人类能查看的方式，列出文件的大小

-rw-r--r-- 1 root root 22K Dec 8 2016 install.log

4、ls -ld jingfeng #只列出目录的详细信息

drwxr-xr-x 5 root root 4096 Aug 11 23:00 jingfeng

5、ls -lrt #已时间倒叙的方式来列出

-rw-r--r-- 1 root root 205 Feb 5 2017 jingfeng.txt.bz2

-rw-r--r-- 1 root root 1 Aug 5 21:24 jingfeng.txt

drwxr-xr-x 5 root root 4096 Aug 11 23:00 jingfeng

1. pwd 列出当前目录 \*

主要选项：-L 大写的L，显示的逻辑路径 /etc/init.d

-P 大写的P，显示的是实际物理路径 /etc/rc.d/init.d 这两个参数了解即可

主要案例：[root@db02 ~]# pwd

/root

5.touch 创建一个空白文件，假如当前目录有同样的文件，则会更新文件的时间戳 \*

主要选项：-a 修改access（访问）时间

-m 修改modify（修改）时间 这两个参数了解即可

主要案例：touch jingfeng.txt

1. cat 查看文件 \*

主要选项：-n 显示行号

-b 只显示非空行的内容（显示行号）

-E 在每行结束处显示$

-s 压缩空行

-T 把tab键字符显示为^I

主要案例：cat -n jingfeng.txt

cat -b jingfeng.txt

cat -E jingfeng.txt

cat -T jingfeng.txt

cat >>jingfeng.txt<<EOF #cat的高级用法，配合重定向

Jingfeng

Woaini

EOF

7.vi 简单的编辑命令 \*

主要案列：vi jingfeng.txt

8.vim 复杂的编辑器 \*（使用方法参考linux基础里面的vim篇章）

使用方法同上

1. echo 打印文件内容或编辑文件内容 \*

主要选项：-n 不换行输出

-e 可以使用转义字符（\n回车，\t tab键）

主要案列：echo “I am studying linux”>>jingfeng.txt 追加文件尾部内容

echo $? 假如返回值为0的时候，表示上一次命令成功。假如是1到255的话，则是失败

echo -e “wo\tshi\tshei”> jingfeng.txt

10.xargs 从标准输入（管道或stdin、输入重定向）获取数据。并将数据转换成命令行的参数 \*

简单理解就是find 管道|后面接文件，就需要xargs。假如| 前面输入的是文字，字符串就不需要接xargs

主要选项：-n 数字分组

-i 将xargs的每项名称，一行一行赋值给{}，可以用{}代替

主要案列：xargs -n 2 < jingfeng.txt

find /etc -type f -name “\*.txt”|xargs rm -f

find /etc -type f -name “\*.txt”|xargs -i mv {} /tmp

1. cp 复制文件或目录 \*

主要选项：-a 相当于-pdr

-f 强制复制，不提示

主要案列：cp -a jingfeng.txt /jf

简写方式 cp /etc/ssh/sshd\_config {,orz}

cp -a {/root/jingfeng.txt,/tmp/1.txt}

1. mv 移动文件或目录，并且可以更改文件或目录的名字 \*

主要选项：-f 强制移动，不提示

主要案列：mv jingfeng.txt /jingfeng xj.txt

13.rm 删除文件或者目录，这个命令在日常工作中一定要慎用 \*

主要选项：-r 递归删除文件或者目录

-f 强制删除不提示

主要案列：rm -f jingfeng.txt

1. find 详细查找文件或者目录 \*\*\*\*\*

主要选项：-type 文件类型 f=文件 d=目录 c=字符类型 b=块设备类型

-name 文件名

-iname 文件名不区分大小写

已天为单位：

-mtime 修改文件内容 按修改时间查找 +7 7天以前 7 第7天 -7 7天以内

-atime 根据访问时间查找

-ctime 根据创建时间查找

已分钟为单位：

-amin 根据访问时间查找

-mmin 根据修改时间查找

-cmin 根据创建时间查找

-user 根据用户名查找

-group 根据组用户查找

-uid 根据用户的uid查找文件

-gid 根据用户的gid查找文件

-size 根据文件的大小来查找文件（默认单位：KB,MB,GB）

1024：什么都不加，小于文件大小-1或等于

+1024：大于等于文件大小

-1024 小于等于文件大小

-perm 根据权限查找

644：什么都不加，精确文件权限匹配

/644： 任何一类用户（u,g,o）的权限中的任何一位（r,w,x）符合条件即满足

-644： 每一类用户（u,g,o）权限中的每一位（r,w,x）同时符合条件即满足

！取反

-o 或者

-exec COMMAND {} \; 对查找到的每个文件执行由COMMAND表示的命令

注意：find传递查找到的文件路径至后面的命令时，是先查找出所有符合条件的文件路径，并一次性传递给后面的命令。但是有些命令不能接受过长的参数，此时命令执行会失败。所以需要另一种方式来规避问题：find |xargs COMMAND

主要案例：find jingfeng/ -type f -name "\*.txt" | xargs rm -f

find jingfeng/ -type f -name "jingfeng.txt" -exec rm {} \;

find jingfeng/ -type f -name "\*txt" -mtime -7 | xargs rm -f

这条命令解析：查找jingfeng目录下的为文件、名字为\*txt、最近7天之内修改的文件通过管道命令，再通过xargs排序 删除文件

find /jingfeng -type f -name "\*.txt" | xargs sed -i s#jingfeng#momo#

或者sed -i "s#shei#jingfeng#g" $(find /jf/ -type f -name "\*.txt")

再或者find /jf/ -type f -name "\*.txt" -exec sed -i "s#shei#jingfeng#g" {} \;

find /jingfeng -type f -name "\*.txt" -ok rm -f {} \;这种方式，当你删除的时候，会询问你是否需要删除该文件

find /backup -type f \( -name "\*.log" -o -name "\*.tar.gz" \) 同时查找多个名字的文件

find /data -type f -name “.txt” -mtime +7|xargs -i mv {} /tmp

1. grep 过滤器 \*\*\*\*\*

主要选项：-v 排除内容

-E 可以同时实现两条命令的使用，当然也要加|

--color=auto 对过滤的字符串加颜色

-i 不区分大小写

-w 按单词搜索

-B 除了显示匹配的一行之外，并显示该行之前的N行

-A 除了显示匹配的一行之外，并显示该行之后的N行

-C 除了显示匹配的一行之外，并显示该行之前后各N行

-o 只显示匹配到的内容

-n 显示行号

主要案列：grep -v old jingfeng.txt grep -Po '(?<=addr:)\S+' 取ip地址

grep -n “.\*” jingfeng.txt grep -w “jf” jingfeng.txt

grep -E “root|nginx” /etc/passwd

16.egrep 支持扩展正则表达式的grep命令 \*\*\*\*\*

主要参数： -v 排除内容

--color=auto 对过滤的字符串加颜色

-i 不区分大小写

-o 只显示匹配到的内容

-B 除了显示匹配的一行之外，并显示该行之前的N行

-A 除了显示匹配的一行之外，并显示该行之后的N行

-C 除了显示匹配的一行之外，并显示该行之前后各N行

主要案列：egrep -v old jingfeng.txt egrep -Po '(?<=addr:)\S+' 取ip地址

egrep -w “jf” jingfeng.txt

1. head 查看文件的前N行 \*

主要选项：-n 后面接数字，表示查看前几行（也可以不加）

注意：head默认查看10行

主要案列：head -5 /etc/passwd

head -10 /etc/passwd|tail -5 查看passwd文件中6到10行

1. tail 查看文件的后N行 \*

主要选项：-n 后面接数字，表示查看前几行（也可以不加）

-f（循环读取）监视filename文件的尾部内容（默认10行，相当于增加参数-n 10）

刷新显示在屏幕上，退出按下ctrl+c

注意：tail默认查看10行

主要案列：tail -5 /etc/passwd

1. alias 查看和设置别名 \*

主要案列：查看别名，直接输入alias。

设置别名：alias cp='cp -i'

alias rm='echo “this command does not allow to ues”'

别名的配置文件：/etc/profile或者/etc/bashrc 全局生效

~/.bashrc #当前用户生效

设置别名文件时命令必须是/bin下的

1. unalias 取消别名

主要案例：unalias cp

1. seq 生成序列 \*（编写shell时需要）

主要选项：-s 指定分隔符

-w 输出同宽数列，不足的位数用0补齐

主要案例：[root@db02 ~]# seq -s : 10

1:2:3:4:5:6:7:8:9:10

[root@jf ~]# seq 100 >jingfeng.txt

[root@jf ~]# cat jingfeng.txt

1

2

....100

1. sed 流编辑器，实现对文件的增删改查 \*\*\*\*\*

编辑命令：

d：删除 #sed ‘2,3’d /etc/passwd

p：打印输出

a：在行后面追加文本，支持使用\n实现多行追加

i：在行前面插入文本，支持使用\n实现多行插入

c：把匹配到的行替换为此处指定的文本

w：保存模式空间匹配到的行至指定的文件中

r：读取指定文件的内容至当前文件被模式匹配到的行后面并且文件合并

!：条件取反

s///：查找替换，其分隔符可自行指定，常用的有s@@@,s###等

替换标记：g 全局替换

主要选项：-n 取消默认输出，把我想输出的内容输出到屏幕上

-i 修改文件内容并且会破坏软链接和硬链接

-e 多项编辑

-f 每行一个编辑命令

-r 支持使用扩展正则表达式

主要案例：sed -n ‘20,30’p /etc/passwd

sed -i "s#xj#jingfeng1#g" jingfeng.txt

sed -nr '2s#^.\*addr:(.\*) Bca.\*$#\1#gp' 取IP

sed -i '3a '$(sed -n '2p' 1.txt)'\n'$(sed -n '3p' 1.txt) 2.txt

注意：sed 's#内容#替换的内容#g jingfeng.txt 2.sed里面有小p，前面必须接参数-n

1. awk 过滤，输出内容 \*\*\*\*\*

区域：$就是取什么什么引用 $1 表示第一列，$2 第二列，依次类推。但是$0表示整行

FS==区域分隔符，默认是空格

RS==每个记录读入的时候的分隔符（行的结束标识）

NR==行号，记录的数。

NF==NF表示记录中的区域（列）数量，$NF表示取最后一列

awk默认已空格为分隔符

BEGIN模块在awk读取文件之前就执行,一般用来定义我们的内置变量（预定义变量：FS，RS）

END模块就是END在awk读取完所有文件的时候，再执行END模块，一般用来输出一个结果（累加，数组的结果）

awk编程思想：

1.先处理，最后再END模块输出

2.BEGIN模块用于awk内置变量FS，RS的赋值，在awk执行前

3.END模块用来最后输出，统计信息

awk几种运算表达式：a=a+1或者a++ a=a+2或者a+=2 a+=$0

主要参数：-F 指定分隔符 比如说指定：为分隔符，那么：相当于一把菜刀，把数据切成一块一块，这个就是-F的作用

在linux里面==等于等号。一个=在linux里面相当于赋值

主要案例：awk 'NR>19 && NR<31' jingfeng.txt

awk 'NR==20,NR==30' jingfeng.txt

awk -F ":" 'NR==1 {print $6}' /etc/passwd

awk -F "[ :]+" '{print $3,$5,$7}' 加号表示取一个整体

cat /etc/passwd|awk -F "[ :]+" 'NR==10,NR==20 {print $3}'

awk 'BEGIN{RS=":"} {print NR,$0}' passwd

awk 'BEGIN{RS=":"} {print NR,$0}' passwd

awk 'BEGIN{RS="[:/0-9 |\n]+"}{print $0}' passwd | sort | uniq -c | sort -rn 统计单词数

awk -F "[ /]+" '$1~/^(ftp|http|ssh)$/{print $1,$2}' /etc/services

awk -F "[:]+" '$3>15{a=a+1}END{print a}' awkfile.txt

awk 'BEGIN{jf["a"]="jingfeng";olboy["b"]="xiaoyu";jf["c"]="wo"

awk '{print NR,$0}' jingfeng.txt

>print jf["a"]

>print oldbot["b"]

>print jf["c"]}' 这个案列引用了数组的概念：1.awk数组的元素名（苹果名）可以是字符串

2.字符串要使用双引号引起来

echo jf{01..10}|xargs -n 1|awk '{print "echo "$0" >>/tmp/jf.txt && useradd "$0" && pa=$(mkpasswd|cut -c 1-5) && echo $pa|passwd --stdin "$0"|echo $pa >>/tmp/jf.txt"}'|bash 批量创建用户和随机密码

awk -F "[: ]+" '/Dan/{print $5}' reg.txt 已Dan开头的行，并且匹配第五列

awk -F "[ :]+" '$3~/^138/{print $1,$3}' reg.txt 已138开头的手机号码，取人的名字和号码（这里的~表示匹配的意思）

awk -F "[ :]+" '$3!~/^138/{print $1,$2":"$3}' reg.txt 匹配不已138开头的手机号码和人的名字

awk -F "[ :]+" '!/138/{print $1,$2":"$3}' reg.txt 匹配不已138开头的手机号码和人的名字

awk -F "[ :]+" '$1~/C|E/{print $1,$2}' reg.txt 已C或E开头的人名

awk -F "[ :]+" '$3~/(1|5)$/{print $1,$2}' reg.txt 已结尾为1或者5的手机数字，然后取对应的人名

awk 'NR==2||NR==3||NR==9{print NR,$0}' jingfeng.txt或者awk 'NR==2;NR==3;NR==9{print NR,$0}'

jingfeng.txt

2 bin:x:1:1:bin:/bin:/sbin/nologin

3 daemon:x:2:2:daemon:/sbin:/sbin/nologin

9 mail:x:8:12:mail:/var/spool/mail:/sbin/nologin 取第二行，第三行，第九行

awk 'BEGIN{a="abcd";print a}' awk中字母会被认为是变量，如果真的要给一个变量赋值，使用双引号

awk '/^$/{i=i+1}END{print i}' /etc/services 统计空行数量

seq 100|awk '{s=s+$0}'END'{print s}'

awk 'BEGIN{jf["a"]="oldgirl";jf["b"]="jingfeng";jf["c"]="oldbaby"

> for(police in jf)

> print police,jf[police]

> }'

a oldgirl

b jingfeng

c oldbaby （awk数组）

awk -F "[/]+" '{ceshi[$2]=ceshi[$2]+1}'END'{for(police in ceshi)print police,ceshi[police]}' ceshi.txt （数组）

mp3.etiantian.org 1

post.etiantian.org 2

www.etiantian.org 3

awk '{a[$1]+=$2}END{for(i in a)print i,a[i]}' shuzu.txt “a[$1]=a[$1]+$2”=”a[$1]+=$2”

awk小结：1.NR行号，存放着每个记录的号，读取新行时候自动+1

2.RS是输入数据的记录的分隔符，简单理解就是可以指定每个记录的结尾标志

3.用RS替换\n

4.RS作用就是表示一个记录的结束

5.FS标识着每个区域的结束

6.awk只有模式（条件）的时候，会默认输出整行（$0）

7.$1~/正则表达式-开始/,$3~/正则结束/ 行，记录

8.awk中字母会被认为是变量，如果真的要给一个变量赋值，使用双引号

1. rmdir 删除空目录

主要选项：-p 删除某目录后，如果其父目录为空，则一并删除

主要案列：rmdir /jf

1. tree 显示目录树 \*

主要选项：-a 显示所有文件和目录

-d 只显示目录

-L 后面接上数字，表示只显示几层树结构

-f 为每个文件和目录显示完整的路径

-F 可以用来区分文件和目录

-i 不以阶梯状列出文件或目录名称

主要案例：[root@db02 ~]# tree -d -L 1 /application/

/application/

├── apache2.2.17

├── mysql -> /application/mysql-5.6.35/

└── mysql-5.6.35

3 directories

1. man 帮助命令 \*

当要搜索某个关键字的时候，输入/，然后在输入关键字搜索

其中n这个快捷键是向下翻所查找的关键字的内容，大写N是向上查找

主要案列：man cp

27.--help 简单的帮助命令 \*（外部命令）

主要案列：cp --help

1. uname 查看系统 \*

主要选项：-m 查看系统是几位操作系统

-r 查看系统的内核版本

-a 查看详细的系统内核版本和系统的操作系统

主要案列：[root@db02 ~]# uname -a

Linux db02 2.6.32-573.el6.x86\_64 #1 SMP Thu Jul 23 15:44:03 UTC 2015 x86\_64 x86\_64 x86\_64 GNU/Linux

1. stat 详细查看文件和目录的信息 \*

主要案例：[root@db02 ~]# stat jingfeng.txt

File: `jingfeng.txt'

Size: 310 Blocks: 8 IO Block: 4096 regular file

Device: 803h/2051d Inode: 916335 Links: 1

Access: (0644/-rw-r--r--) Uid: ( 0/ root) Gid: ( 0/ root)

Access: 2017-08-14 16:53:53.668136237 +0800

Modify: 2017-08-14 16:53:48.144137021 +0800

Change: 2017-08-14 16:53:48.144137021 +0800

30.rz 上传命令 \*

假如系统里面没有这个命令，则使用yum install lrzsz -y来安装次命令

主要参数：-y 覆盖

主要案例：直接输入rz，就可以上传文件

注意：新版的xshell支持直接从windows把文件拖进xshell里，但是前提必须安装了rz

1. sz 下载命令 \*

主要参数：-y 覆盖

主要案例：sz -y jingfeng.txt

注意：1.只能上传文件，目录不行。假如一定要上传目录的话，可以打包上传，但必须是zip格式

2.下载的文件到linux中，是当前路径

1. useradd 创建普通用户 \*

主要选项：-u 指定uid 列子：useradd -u 504 jingfeng （重要）

-c 新账号password的说明栏

-d 设置新账号每次登入时使用的家目录

-e 账号终止日期。日期格式为MM/DD/YY

-g 指定属于的主组

-G 定义此用户为多个不同的组成员，每个用户组使用，逗号来分隔

-s 指定登录shell （重要）

-m 创建家目录

-M 不创建用户家目录

-r 创建系统用户

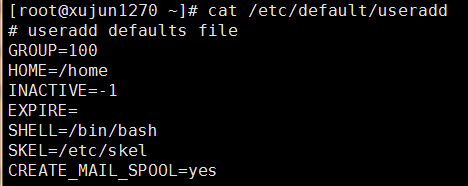
-D 显示或修改用户的默认配置 修改的结果保存与/etc/default/useradd文件中

主要案例：useradd -s /sbin/nologin -M jingfeng

useradd -u 504 jingfeng

注意：创建用户时的诸多默认设定配置文件为/etc/login.defs

/etc/default/useradd这个配置文件，可以默认更改useradd的一些参数



1. passwd 创建用户密码或修改密码 \*

主要选项：--stdin 非交互式设置密码

-n 指定最短使用期限

-x 最大使用期限

-w 提前多少天开始警告

-i 非活动期限

-e 过期期限

主要案例：echo “123456”|passwd --stdin jingfeng

1. su 切换用户 \*

主要选项：-c 变更账号为USER的使用者，并执行指令后再变回原来的使用者

- 加载环境变量

主要案例：su - jingfeng

1. hostname 显示和设置主机名 \*

主要案列：hostname 直接输入，显示主机名

hostname jingfeng 更改jingfeng为主机名 但这个只是暂时的更改，重启就没有了

假如需要永久性的更改主机名，需要到配置文件/etc/sysconfig/network

1. init 切换运行级别

主要案例：init 0 关机

init 1 单用户

init 2 没有NFS网络的多用户模式

init 3 有NFS网络的多用户模式

init 4 没有用户

init 5 x window

init 6 重启

1. runlevel 查看当前系统的运行级别

主要案例：直接输入runlevel

1. shutdown 关机

主要选项：-h 关机

-r 重启

-c 终止关机或重启的动作

主要案例：shutdown -h now

shutdown -r now

1. reboot 重启

主要案例：直接输入reboot

1. history 查看命令历史记录 \*

登陆shell时，会读取命令历史文件中记录下的命令（~.bash\_history）

登陆进shell后新执行的命令只会记录在缓存中，这些命令会用户退出时追加至~.bash\_history

!#：再一次执行历史列表中的第#条命令

!!：再一次执行上一条命令

!STRING：再一次执行已STRING开头的命令

控制命令历史记录的方式：HISTCONTROL=ignoredups（忽略重复的命令），ignorespace（忽略已空格开头的命令），ignoreboth（以上两者同时生效）

主要选项：-a 追加本次会话新执行的命令历史列表至历史文件中

-c 清空所有

-d 删除指定历史内容

主要案例：如何调用历史记录，使用！后面接命令的字符 比如：！Touch

history -c

History -d 5 删除指定历史内容

1. mount 磁盘挂载 \*

主要选项：-t 后面可以接文件系统ext2 ext3等等

-r 只读挂载

-L 挂载时以卷标的方式指明设备

-U 挂载时以uuid的方式指明设备

-a 可自动挂载定义在/etc/fstab文件中的所支持自动挂载的设备

-o 挂载选项：

sync/async：同步/异步操作

atime/noatime：文件或目录在被访问时是否更新其访问时间戳

diratime/nodiratime：目录在被访问时是否更新其访问时间戳

remount/：重新挂载

acl：支持使用访问控制列表

dev/nodev：此设备上是否允许创建设备文件

exec/noexec：是否允许允许此设备上的程序文件

user/nouser：是否允许普通用户挂载此文件系统

suid/nosuid：是否允许程序文件上的suid和sgind特殊权限生效

使用技巧：可以实现将目录绑定至另一个目录上，作为其临时访问入口

mount --bind 源目录 目标目录：mount --bind /etc /mnt

主要案列：mount -t ext3 /dev/cdrom /mnt

lsof /mnt 当umount /mnt无法卸载的时候，可以使用这个命令，来查看是否有进程在使用中

1. which 详细查看命令的路径 \*

主要选项：--skip-alias 不显示别名

主要案列：which ls

[root@m01 ~]# which --skip-alias ls

/bin/ls

43.dmesg 查看系统的故障详细情况

主要案例：dmesg

1. ifup和ifdown启动和停止网卡，后面接网卡名 \*

主要案例：ifdown eth0

ifup eth0

1. nl 显示行号

主要案例：nl jingfeng.txt

46.less 查看当前文档内容，相比cat，less的使用方式更加的灵活多变，它可以实现上下翻页等功能 \*

主要选项：-N 显示行号

主要案列：less jingfeng.txt

1. more 按页查看，不能回退，灵活上不如less \*

主要案例：more jingfeng.txt

1. wc 统计文件的字数和行数，字节数 \*

主要选项：-l 显示总行数，常用于查看进程是否启动

-L 显示最长一行的字符数

-c 显示字节数

-m 显示字符数

-w 显示单词数

主要案例：wc -l jingfeng.txt

[root@web01 ~]# ps -ef|grep "/sbin/nginx"|grep -v grep|wc -l

1 工作常用这个来查看进程是否存在，假如为1，就是存在的。

要是0的话，就表示该进程不存在

1. chkconfig 设置开机自启动 \*

主要选项：--list 查看启动服务

--level 设置运行级别

--add 添加服务

--del 删除服务

注意：正常级别下，最后启动的一个服务S99local没有链接至/etc/init.d/下的脚本，而是链接了/etc/rc.d/rc.local脚本。因此，不便或不需写为服务脚本的程序要想开机自启动，则需要放置此文件中即可。

主要案列：chkconfig --level 5 sshd off（on）

chkconfig --list sshd （查看指定服务是否开启还是关闭）

chkconfig --list|grep "3:on"|egrep -v "crond|sshd|network|rsyslog|sysstat"|awk '{print "chkconfig",$1,"off"}'|bash

假如我写了一个脚本（命令的集合），怎么让chkconfig管理并运行呢

1.脚本必须放在/etc/init.d下

2.脚本要有执行权限（x）

3.脚本的前几行必须要有

#chkconfig：2345 08 92 #这个必须要有

#description：starts，stop and saves iptables firewall

4.通过chkconfig --add添加

1. dumpe2fs 显示ext2/ext3/ext4文件系统信息

主要案列：dumpe2fs /dev/sda1

1. df 列出linux下所有的磁盘容量 \*

主要选项：-h 已人类可读的方式（MB,GB,KB）显示出来

-i 列出inode剩余的总量和使用量

-T 查看当前挂载的文件系统

-l 仅显示本地文件的相关信息

主要案例：df -h

1. file 查看文件类型

主要案例：file jingfeng.txt

1. 查看用户的最近登陆信息，对应的数据文件是/var/log/wtmp \*

主要案例：[root@db02 yum.repos.d]# last

root pts/3 10.0.0.1 Fri Aug 18 11:19 still logged in

root pts/1 10.0.0.1 Mon Aug 14 22:37 - 13:15 (3+14:37)

root pts/4 10.0.0.1 Mon Aug 14 17:00 - 17:02 (00:02)

root pts/3 10.0.0.1 Fri Aug 11 22:56 - 19:40 (2+20:44)

54.lastlog：显示最近登陆的所有用户信息。对应的数据文件是/var/log/lastlog

主要案例：[root@web01 ~]# lastlog

Username Port From Latest

root pts/0 192.168.203.1 Tue Mar 14 22:12:43 +0800 2017

bin \*\*Never logged in\*\*

daemon \*\*Never logged in\*\*

adm \*\*Never logged in\*\*

lp \*\*Never logged in\*\*

sync \*\*Never logged in\*\*

55.ln 链接文件 \*

1.硬链接：可以看做是一种备份，当原文件写入，备份文件也会自动写入。当原文件被删除的时候，备份文件还能继续使用。

2.软链接：可以看做是windows下的快捷方式。此链接方式的特点可以概括如下：原文件删除，备份文件不能使用。

主要选项：-s 如果不加-s，就是硬链接。相反加了-s，就是软链接

主要案例：ln -s /application/nginx-5.6.3.0 /application/nginx

1. tar 打包命令 \*

主要选项：-j:通过bzip2的支持进行压缩和解压缩。此时文件名最好为：.tar.bz2

-c：新建打包文件

-f：这个参数后面要接被出来的文件名，建议 -f 单独写一个参数

-t：查看打包文件

-z：通过gzip的支持进行压缩和解压缩。此时文件名最好为：.tar.gz

--exclude 这个参数可以过滤你不想打包的目录内容 tar -zcvf /tmp/etc.tar.gz etc --exclude /etc/services

-C:这个参数用在解压缩时，若要在特定目录解压缩，可以使用这个参数

-x：解压缩打包文件

-h：打包的时候，假如目录里面有软链接文件，这个参数会打包软链接对应的真实目录

或文件

-X：从文件里面排除不需要的文件或者目录

-u：添加改变了和现有的文件到已经存在的压缩文件里

主要案例：压缩：tar -zcv -f /tmp/etc.tar.gz /etc

查询：tar -t -f /tmp/etc.tar.gz

解压缩：tar -x -f /tmp/etc.tar.gz /tmp

tar -zcf /backup/html\_`date +%F-%H`.tar.gz www/html/

1. tr 转换字符串 \*

主要选项：-d 删除

主要案例：转换 tr 'abc' 'XYZ' < jingfeng.txt (a-X,b-Y,c-z)

1. id 查看用户和用户组的信息 \*

主要选项：-u 仅显示用户的uid

-g 仅显示用户的gid

-G 仅显示用户所属的所有组ID

-n 仅显示名字而非ID

主要案例：[root@db02 ~]# id john

uid=511(john) gid=511(john) groups=511(john)

59.du 查看linux下目录的容量 \*

主要选项：-s 列出目录容量，而不列出每个各别的目录占用容量

-h 表示系统自动调节单位

-a或-all 为每个指定文件显示磁盘使用情况，或者为目录中每个文件显示各自磁盘使用情况

–max-depth=<目录层数> 超过指定层数的目录后，予以忽略

主要案例：[root@db02 ~]# du -sh /etc

30M /etc

1. date 设置和显示时间 \*

主要选项：-s 修改并设置时间

-d 可以显示以前和未来的时间

%H 小时

%M 分钟

%S 秒

%X 相当于%H %M %S

%F 显示当前所有时间属性

%Y 完整年份

%m 月

%d 日

%A 星期的全称

主要案例：date +%F

date -s “2017-03-09 18:56:00”

echo `date +%F` 工作中会用到的

echo $(date +%F) 工作中会用到的

date +%F -d “-2day” 2天以前的时间

tar zcf etc.tar$(date +%F).gz /etc

1. whereis 查找文件帮助、源代码、命令

主要案列：whereis ifconfig

1. locate 根据数据库里内容查找文件

主要案列：locate ifconfig

1. readlink 查看软链接的源文件在哪

主要案列：readlink 后面接软链接文件

64.rename 重命名软件 \*

主要案列：rename "\_finished" "" \*

1. basename：basename和dirname命令通常用于shell脚本中的命令替换来指定和指定的输入文件名称有所差异的输出文件名称。

主要案例：basename /root/jingfeng1270.txt

jingfeng1270.txt

1. dirname：去除文件名中的非目录部分，仅显示与目录有关的内容

主要案例： dirname /root/jingfeng1270.txt

/root

1. md5sum 计算和校验文件的md5值 \*

主要选项：-c 从指定文件中读取MD5校验和，并进行校验

主要案例：md5sum jf.txt

md5sum jf.txt >> jingfeng.txt

md5sum -c jingfeng.txt

jf.txt: OK

1. chmod 改变文件权限 \*

主要选项：-R 递归改变目录下所有子目录和文件的权限

主要案列：-rwxr-xr-x root root

数字方式：r=4 w=2 x=1

使用方法：chmod 777 jingfeng.txt

ls -l jingfeng.txt

-rwxrwxrwx

1. chown 改变文件和目录的所有者 \*

主要选项：-R 递归改变目录下所有子目录和文件的所有者

主要案列：ls -l jingfeng.txt

-rwxr-xr-x root root

chown jingfeng jingfeng.txt

ls -l jingfeng.txt

-rwxr-xr-x jingfeng root

chown 用户：组 文件或目录

1. umask 修改umask值

当umask值都为偶数，直接相减

如果有奇数，对应为加一

注意：之所以文件用666去减，表示文件默认不能拥有执行权限。如果减得的结果中有执行权限，则需要将其加一。所以这个对目录而言，是没有任何效果的

主要案列:umask 022

1. chattr 增加隐藏属性 \*

主要选项：-a 可以增加内容，但是不能减少或删除文件

+i 锁定文件，让其不能增加，删除，修改

主要案例：chattr +i jingfeng.txt

chattr -i jingfeng.txt #取消锁定

1. lsattr 查看隐藏属性 \*

主要选项：lsattr jingfeng.txt

1. groupadd 创建一个组用户 \*

主要选项：-g 创建组的gid

-r 创建系统组

主要案例：groupadd -g 504 jf

1. usermod 更改用户属性 \*

主要选项：-c 注释

-d 更新用户新的家目录。如果给定-m选项，用户旧的家目录会搬到新的家目录去，如旧的家目录不存在则建个新的

-e 加上用户账号停止日期

-g 更新用户新的起始登入用户组

-G 定义用户为多个用户组

-s 指定新登入的shell

-u 指定用户的UID

-a 和-G配合使用，表示追加新的用户组，不覆盖

-m只能与-d选项一同使用，用于将原来的家目录移动为新的家目录

主要案例：usermod -c "Handsomeboy" -u 806 -G root,jf -s /bin/sh -md /jf6 jf6

1. userdel 删除用户 \*

主要选项：-r 删除用户的家目录（此选项慎用）

主要案例：userdel -r jingfeng

1. crontab 定时任务 \*

主要选项：-l 查看crontab文件内容

-e 编辑crontab文件内容

主要案例：crontab -e

注意：假如直接用命令行去编辑的时候，后面不要加>/dev/null 2>&1

1. tac 倒叙的方式查看文件内容

主要案列：tac jf.txt

1. rev 将文件中的每行内容已字符为单位反向输出，即第一个字符最后输出，最后一个字符最先输出

主要案例：rev jf.txt

1. cut 用来显示行中的指定部分，删除文件中指定字段 \*

主要选项：-b 用字节来切割

-c 用字符来切割

-d 指定字段的分隔符，默认的字段分隔符为“TAB”

-f 显示指定字段的内容（一般配合-d使用）

--output-delimiter=“ ” 可以把指定的分隔符替换成其他字段，比如说空格符

主要案例：ifconfig eth0|grep "inet addr:"|cut -c 21-33

192.168.203.8

1. chage 修改用户密码有效期限 管理/etc/shadow \*

主要选项：-l 查看当前的设置

-E 将账户过期时间设为“过期日期”。日期写法“2017-3-15”

主要案例：[root@db02 ~]# chage -l jingfeng

Last password change : Dec 31, 2016

Password expires : never

Password inactive : never

Account expires : never

Minimum number of days between password change : 0

Maximum number of days between password change : 99999

Number of days of warning before password expires : 7

1. sudo 用来以其他身份来执行命令，预设的身份为root。在/etc/sudoers中设置了可执行sudo指令的用户 \*

主要选项：-l 查看当前用户可以执行的sudo命令

主要案例：[jf@web01 ~]$ sudo ifconfig

1. visudo 配置sudo权限的编辑命令 \*

主要案例：visudo 进入编辑模式之后，在第98行左右，编辑用户和命令（98gg）

root ALL=(ALL) ALL

%jingfeng ALL=(ALL) /usr/sbin/useradd 用户组前面一定加百分号

jf ALL=(ALL) /usr/sbin/useradd

定义多个别名：用户和用户组和多条命令

#用户别名分类

User\_Alias ADMINS = jsmith, mikem

#岗位命令分类

Cmnd\_Alias SOFTWARE = /bin/rpm, /usr/bin/up2date, /usr/bin/yum

Cmnd\_Alias SERVICES = /sbin/service, /sbin/chkconfig

Cmnd\_Alias LOCATE = /usr/bin/updatedb

Cmnd\_Alias STORAGE = /sbin/fdisk, /sbin/sfdisk, /sbin/parted, /sbin/partprobe, /bin/mount, /bin/umount

Cmnd\_Alias DELEGATING = /usr/sbin/visudo, /bin/chown, /bin/chmod, /bin/chgrp

#角色

Runas\_Alias OP1 = root, jf

Runas\_Alias OP2 = oldgirl

ADMINS ALL=(OP1) SOFTWARE

1. groupdel 删除用户组 \*

注意：删除有用户的组，需要先删除用户，然后在删除组

主要案例：[root@db02 ~]# groupdel john

1. w 详细查看当前用户的登陆情况 \*

主要案例：[root@web01 ~]# w

11:06:21 up 3 days, 23:25, 2 users, load average: 0.00, 0.00, 0.00

#第一行从左至右显示的信息依次为：时间、系统运行时间、登陆用户数、平均负载：平均负载里面的数字表示，1分钟内，5分钟内，15分钟内，当数值超过当前系统内核数量的时候，表示系统已经有了压力

USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT

root pts/0 192.168.203.1 Tue22 0.00s 0.00s 0.00s w

root pts/1 192.168.203.1 Mon21 16:01m 0.39s 0.39s -bash

1. who 查看当前用户登陆情况

主要案例：[root@web01 ~]# who

root pts/0 2017-03-14 22:12 (192.168.203.1)

root pts/1 2017-03-13 21:25 (192.168.203.1)

1. whoami 查看当前用户是什么 \*

主要案例：[root@web01 ~]# whoami

root

1. free 可以显示当前系统未使用的和已使用的内存数目，还可以显示被内核使用的内存缓冲区 \*

主要选项：-m 已mb为单位显示

-h 以人类可读的方式显示

主要案例：[root@web01 ~]# free -m

total used free shared buffers cached

Mem: 981 476 505 0 62 298

-/+ buffers/cache: 114 866 （工作看内存大小，主要看这里）

Swap: 1023 0 1023

提示：1）linux系统的特性是将系统不用的物理内存做为缓存或缓冲使用，因此，505不是系统真实的内存

2）系统真正系统可用内存是866M

3）buffers为写入缓冲区，sync将缓冲区数据写入磁盘

4）cache为读取数据的缓存区

5）硬盘是机械的，无论是写入还是读取都太慢了，所以读取和写入都是用了缓存技术

6）门户网站架构都会用缓存技术，来让用户写入读取尽可能不接触硬盘

1. od 查看二进制文件

主要案列：od /bin/ls

1. dd 用于复制文件并对原文件的内容进行转换和格式化处理

主要案列：dd if=/dev/zero of=/root/jingfeng.txt bs=1M count=1 创建一个1M的文件

1. fdisk 用于观察硬盘实体使用情况，也可对硬盘分区 \*

主要选项：-l 列出分表情况

主要案例：fdisk /dev/sdb 为第二块硬盘进行分区

Command (m for help): m

Command action

a toggle a bootable flag

b edit bsd disklabel

c toggle the dos compatibility flag

d delete a partition 删除分区

l list known partition types

m print this menu

n add a new partition 创建一个新的分区

o create a new empty DOS partition table

p print the partition table 列出磁盘目前的分区情况

q quit without saving changes 直接退出不修改分区

s create a new empty Sun disklabel

t change a partition's system id

u change display/entry units

v verify the partition table

w write table to disk and exit 保存分区表并退出（这个选项慎用）

x extra functionality (experts only)

使用物理分区构建swap

fdisk /dev/sdb 分成一个分区

mkswap /dev/sdb1

free -m

swapon /dev/sdb1

free -m

fdisk之后需要通知内核强制重读磁盘分区表：

Centos5:partprobe[device]

Centos6和7:partx,kpartx

partx -a [device]

Kpartx -af [device]

1. parted 磁盘分区管理工具，比fdisk更加灵活。当磁盘大于2T的时候，需要用到这个命令，并且这个命令分区的时候，是实时生效的，所以分区时一定要慎重

主要案列：[root@localhost ~]# parted /dev/sdb GNU Parted Copyright (C) 1998 - 2004 free Software Foundation, Inc. This program is free software, covered by the GNU General Public License. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

使用/dev/sdb

(parted)mklabel gpt

(parted)mkpart primary 0 2048（单位是M）

(parted)Ignore

(parted)quit

1. mkfs 创建文件系统 \*

主要选项：-t 指定文件系统格式

主要案列：mkfs -t ext4 /dev/sdb

1. fsck 磁盘检验

主要选项：-t 指定文件系统格式

主要案列：fsck -t ext4 /dev/sda6 没坏的磁盘一定不要使用这个命令

94.partprobe 把分区表的修改变化通知内核

一般配合fdisk和parted使用

主要案例：partprobe /dev/sdb

1. mkswap 格式化swap分区

主要选项：-L 指明卷标

-f 强制

主要案列：mkswap /dev/sdb1

1. swapon 使用swap分区

主要案列：swapon /dev/sdb1

97.swapoff：不使用swap分区

主要案例：swapoff /dev/sdb1

98.tune2fs：修改文件系统信息 （很少用到）

主要参数：-i 查看超级块的内容

-j 修改文件系统

-L 修改卷标

-m 调整预留空间百分比

-O 开启或关闭某种特性

-o 开启或关闭某种默认挂载选项

主要案例：tune2fs -c 30 /dev/hda1 设置强制检查前文件系统可以挂载的次数

tune2fs -c -l /dev/hda1 关闭强制检查挂载次数限制。

tune2fs -i 10 /dev/hda1 10天后检查

tune2fs -i 1d /dev/hda1 1天后检查

tune2fs -i 3w /dev/hda1 3周后检查

tune2fs -i 6m /dev/hda1 半年后检查

tune2fs -i 0 /dev/hda1 禁用时间检查

tune2fs -j /dev/hda1 添加日志功能，将ext2转换成ext3文件系统

tune2fs -r 40000 /dev/hda1 调整/dev/hda1分区的保留空间为40000个磁盘块

tune2fs -o acl,user\_xattr /dev/hda1 设置/dev/hda1挂载选项，启用Posix Access Control Lists和用户指定的扩展属性

99.megacli：查看raid信息

主要案例：megacli

100.ipmitools：查看硬件信息工具

主要案列：ipmitools

1. resize2fs：调整文件系统大小（LVM，brbd）

主要案列：resize2fs

1. split：按照指定的大小或行数分割文件

主要选项：-l 指定行数

-a 指定生成文件后缀长度 split -l 10 -a 4 jingfeng.txt jingfeng\_(默认是2字节)

-d 指定文件后缀为数字

-b 指定文件大小

主要案例：split -l 10 /etc/inittab new\_

1. paste：把每个文件以列对列的方式，一列列地加以合并

主要选项：-d 指定分隔符

-s 已行来指定文件

主要案例：paste -d ：jingfeng.txt jingfeng1.txt

paste -sd '=\n' svn.txt

1. sort：将文件的每一行作为一个单位，然后升序排列（默认按照accll码排序，升序，从小到大）\*

主要选项：-n 按照数值排序 sort -n jingfeng.txt

-r 倒叙排列数值 sort -r jingfeng.txt

-u 去除重复行

-t 指定分隔符

-k 指定区域

-f 忽略字符大小写

主要案例： awk 'BEGIN{RS="[:/0-9 |\n]+"}{print $0}' 、passwd | sort | uniq -c | sort -rn 统计单词数

head -10 /etc/passwd|tail -5|sort -t : -k 3 -n|cut -d : -f1 #以冒号分隔，取出/etc/passwd文件的第6至第10行，并将这些信息按第三个字段的数值大小进行排序，最后仅显示各自的第一个字段

1. dos2unix：windows格式转换成unix格式 \*

主要案列：dos2unix jingfeng.txt

1. diff：用于比较文件或目录的内容，特别是比较两个版本不同的文件以找到改动的地方 \*

提示符：

a add 增加

c chage 改变

d delate 删除

主要选项：-u 显示行修改的上下文，默认为上下3行

主要案例：diff jingfeng.txt jingfeng1（比较两个文件内容有什么不同）

diff /etc/rc3.d/ /etc/rc6.d/

107.vimdiff：可视化的比较工具，和diff一样，就是多了vim可视化（可以比较4个文件） \*

主要案列：vimdiff jingfeng.txt jingfeng1 jingfeng2 jingfeng3

1. tr：转换删除字符串 \*

主要选项：-d 删除

主要案列：转换 tr 'abc' 'XYZ' < jingfeng.txt (a-X,b-Y,c-z)

1. tee：多重定向

主要选项：-a 追加重定向

主要案例：ls|tee /tmp/ls.txt

cat /etc/passwd|tee /tmp/passwd.txt|tr ‘a-z’ ‘A-Z’ 把cat的数据通过管道输入到tr命令并且保存到/tmp/passwd.txt文件中

1. dig：解析域名 \*

主要选项：-t 正向解析，直接跳过本地的hosts文件，解析DNS配置的IP

-x 反向解析

主要案例：dig www.baidu.com (这样解析不是很完美)

dig @8.8.8.8 www.baidu.com +trace （完美解析）

dig -t A www.baidu.com

111.nslookup：DNS服务诊断工具

主要案例：nslookup

>baidu.com

112.host：域名解析工具

主要案例：host www.baidu.com

1. ping：网络服务诊断工具 \*

主要选项：-c 指定发送的ping包个数

-w 指定ping命令超时时长

-W 一次ping操作中，等待对方响应的超时时长

-s 指定ping包报文大小

主要案列：ping www.baidu.com

114.route:配置默认网关和网络静态路由 \*

主要选项：-n 查看网关信息

-net 到一个网络的路由表

主要案例:route add default gw 网关（添加网关，但是是临时生效）

route del default gw 网关（删除网关）

route add -net 192.168.0.0/24 gw 10.10.10.2 （添加192.168.0.0这个网段，并从10.10.10.2这个网关出去）

1. ifconfig：显示网卡信息 \*

主要选项：-a 显示所有接口

主要案例：ifconfig eth0 10.0.0.100/24 up

ifconfig eth0 （只查看一块网卡）

ifconfig eth0:0 192.168.1.1/24 up（配置在一块网卡上配置多个IP，网卡或者系统重启失效）

ifconfig eth0 192.168.1.100 netmask 255.255.255.0 up

ifconfig eth0:0 down （删除IP）

1. ip：查看网卡信息（未来趋势）\*

主要选项：ip address 查看网卡信息

ip address add：添加ip地址

ip address del：删除ip地址

ip address show：指定查看接口的信息

ip address flush：清空端口的ip地址

ip route 查看路由

ip route add：添加新的路由

ip route change：修改路由

ip route replace：修改或添加新的

ip route delete：删除路由

ip route show：列出路由表

ip link 查看不同网络接口的统计数据

注意：ip命令大部分选项都可以简写

主要案例：ip addr add 10.0.0.13/24 dev eth0:3（配置在一块网卡上配置多个IP，网卡或者系统重启失效）

ip addr del 10.0.0.13/24 dev eth0:3 （删除IP）

ip link show eth1 #指定查看接口的信息

ip address show #查看IP地址（同ifconfig）

ip address flush dev eth0 #清空eth0的ip地址

ip route add 192.168.0.0/24 via 10.0.0.2 dev eth1

ip route add default via 10.0.0.2

注意：配置子接口，需要在/etc/sysconfig/network-scripts/里配置一个ifcfg-eth0:0，这样才能永久有效

117.netstat：查看进程或者端口 \*

主要参数：-a 查看所有

-p 显示正在运行的进程

-u 显示udp协议相关的进程或者端口

-t 显示tcp协议相关的进程或者端口

-n 已数字显示

-l 仅列出有在 Listen (监听) 的服務状态

-r 显示路由表

主要案列：netstat -lntup|grep 80

netstat -rn

118.lsof：查看端口 \*

主要参数：-i 列出符合条件的进程。（4、6、协议、:端口、 @ip ）

主要案例：lsof -i :80

lsof |grep del 查看被删除的进程

注意：lsof /mnt 当umount /mnt无法卸载的时候，可以使用这个命令，来查看是否有进程在使用中

119.traceroute：用于追踪网络数据包在网络上传输时所经过的路径 \*

主要参数：-n：直接使用IP地址而非主机名称

主要案例：traceroute www.baidu.com

120.tcpdump：抓包工具 \*

主要参数：-i 指定端口

-n 指定协议

-t 在输出的每一行不打印时间戳

-S 将tcp的序列号以绝对值形式输出，而不是相对值

-c 在收到指定的数量的分组后，tcpdump就会停止

主要案列：tcpdump -n icmp -i eth0

tcpdump tcp -i eth1 -t -s 0 -c 100 and dst port ! 22 and src net 192.168.1.0/24 -w ./target.cap

(1)tcp: ip icmp arp rarp 和 tcp、udp、icmp这些选项等都要放到第一个参数的位置，用来过滤数据报的类型  
(2)-i eth1 : 只抓经过接口eth1的包  
(3)-t : 不显示时间戳  
(4)-s 0 : 抓取数据包时默认抓取长度为68字节。加上-S 0 后可以抓到完整的数据包  
(5)-c 100 : 只抓取100个数据包  
(6)dst port ! 22 : 不抓取目标端口是22的数据包  
(7)src net 192.168.1.0/24 : 数据包的源网络地址为192.168.1.0/24  
(8)-w ./target.cap : 保存成cap文件，方便用ethereal(即wireshark)分析

1. uniq：用于去除或忽略文件中的重复行，一般和sort结合使用 \*

主要选项：-c 在每列旁边显示该行重复出现的次数

-d 仅显示重复的行

-u 仅显示不曾重复的行

主要案例：sort file.txt | uniq -c 统计各行在文件中出现的次数

awk 'BEGIN{RS="[ :/0-9\n]+"}{print $0}' jingfeng.txt|sort -n|uniq -c

3 adm

1 bash

5 bin

2 daemon

3 halt

2 lp

1 lpd

3 mail

6 nologin

3 root

12 sbin

122.column：有序的排序

主要参数：-t 判断输入行的列数来创建一个表。分隔符是使用在-s中指定的字符。如果没有指定分隔符，默认是空格

主要案例：column -t

1. rsync：备份工具 \*

主要选项：-v 详细模式输出，传输时的进度等信息

-z 传输时进行压缩以提高传输效率

-a 归档模式，表示以递归方式传输文件，并保持所有文件属性，等于-rtopgdl

--bwlimit=10 限制传输速度

--exclude=PATH：表示指定排除不需要传输的文件，等号后面跟文件名，可以是万用字符模式（如\*.txt）

--delete：表示删除DST中SRC没有的文件

-L：会把软链接指向的目标文件复制到对端目录中

-e：指定使用ssh还是其他方式来数据同步

主要案例：主要案列：本地模式 rsync -avz /tmp/ /jf/

使用远程ssh来传递文件：rsync -avz /etc/hosts -e 'ssh -p 22' [jf@192.168.229.101:/home/jf/](mailto:oldboy@192.168.229.101:/home/oldboy/) ssh key （免密钥的方式）

1. mail：邮件服务

主要选项：-s 指点邮件主题

vim /etc/mail/rc 编辑邮件服务配置文件

set from=jingfeng1270@163.com

set smtp=smtp.163.com

set smtp-auth-user=jingfeng1270@163.com

set smtp-auth-password=jingfeng66207383

set smtp-auth=login

主要案列：mail -s “标题” 邮件地址<文件

mial -s [“jf”837367121@qq.com](mailto:) </etc/hosts

echo “正文”|mail -s “jf” 837367121@qq.com

1. umount：卸载挂载服务

主要选项：-f 强制卸载

-l 并不是马上umount，而是在该目录空闲后再umount

主要案例：umount -lf /mnt

lsof /mnt 当umount /mnt无法卸载的时候，可以使用这个命令，来查看是否有进程在使用中

1. showmount：查看挂载

主要选项：-e 显示NFS服务输出的目录列表

主要案例：showmount -e 10.0.0.14

127.inotifywait：在被监控的文件或目录上等待特定文件系统时间（open、close、delete）发生，执行后处于阻塞状态，适合在shell脚本中使用

主要参数：-r 递归查询目录

-q 打印监控事件的信息

-m 始终保持事件监听状态

--excludei 排除文件或目录时，不区分大小写

--timefmt 指定时间输出的格式

--format 打印使用指定的输出类似格式字符串

-e 通过此参数可以指定需要监控的时间，如下所示：

access 文件或目录被读取

modify 文件或目录内容被修改

attrib 文件或目录属性被改变

close 文件或目录封闭，无论读/写模式

open 文件或目录被打开

moved\_to 文件或目录被移动至另外一个目录

move 文件或目录被移动另一个目录或从另一个目录移动至当前目录

create 文件或目录被创建在当前目录

delete 文件或目录被删除

unmount 文件系统被卸载

主要案列：inotifywait -mrq --format ‘%w%f’ -e create,delete /backup 实时监控backup某个事件的状态

1. sersync：实时同步工具 \*

主要选项：-d 启用守护进程模式

-r 在监控前，将监控目录与远程主机用rsync命令推送一遍

-n 指定开启守护进程的数量，默认为10个

-o 指定配置文件，默认使用confxml.xml文件

主要案例：/application/sersync\_installdir\_64bit/sersync/bin/sersync -d -r -n 8 -o /application/sersync\_installdir\_64bit/sersync/conf/confxml.xml

1. watch：周期性执行程序，打印到目录，默认2秒

主要选项：-n 指定秒

-d或--differences  用-d或--differences 选项watch 会高亮显示变化的区域。 而-d=cumulative选项会把变动过的地方(不管最近的那次有没有变动)都高亮显示出来。

主要案例：watch -n 1 -d netstat -ant

1. ssh：linux ssh客户端工具 \*

主要选项：-p 指定端口

-t 打开终端

主要案例：ssh -p22 jingfeng@172.16.1.52

1. scp：远程复制 \*

主要选项：-P 指定端口

-r 复制目录

-p 保持属性

-l 限速

主要案例：scp -P22 -rp /etc jingfeng@192.168.203.61:/tmp

132.help：内置命令的帮助 \*

主要案列：help cd

1. hash：shell搜寻到的外部命令的路径结果会缓存至内存缓存中

主要选项：-d 指定删除某个命令

-r 清空所有

主要案例：hash -d help

hash -r

1. wget：下载命令 \*

主要选项：-q 不显示下载过程

-O 指定文件名

-b 在后台执行下载操作

-c 断点续传

--spider 爬虫，检查网站是不是好的，不会下载，只会检查

-T, –timeout=SECONDS 设定响应超时的秒数

--tries=2 指定重试的次数

--limit-rate=amount 以指定的速率传输文件

主要案例：wget -O /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-6.repo

135.curl 下载工具 \*

主要参数：-I 查看响应header信息

-s 安静的

-w 获取状态码

-o 把输出写到该文件中

主要案例：[root@m01 ~]# curl www.etiantian.org

www

[root@m01 ~]# curl bbs.etiantian.org

bbs

curl -o /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-5.repo

136.rpm ：RPM是RPM软件包的管理命令 \*

Rpm包的命名格式：redis-3.0.2-1.centos7.x64.rpm

3.0.2-1是版本号

Centos7是适用于哪个平台

x64是32位还是64位系统

主要选项：-a 查看所有软件包

-f 查看指定文件的软件包

-q 使用询问模式，当遇到任何问题时。Rpm会先询问用户

-l 显示软件包的文件列表

--nodeps 忽略软件包的依赖关系强行安装

-e 删除指定的软件包

--force  忽略软件包及文件的冲突

-h 软件包安装时列出标记

-v 打印输出

-U 升级指定的软件包

-i 安装软件包

-c 查询指定的程序提供的配置文件

-i 程序包相关的信息，版本号，大小，所属的包组等

-d 查询指定的帮助文档

-R 查询指定包的依赖关系

--replacepkgs 重新安装

--noslgnature 不检查包签名信息，不检查来源合法性

--nodigest 不检查包完整性信息

--force 强制升级

--allmatches 卸载所有匹配指定名称的程序包的各版本

--whatrequires 查询指定的命令或者文件被哪个包所依赖

主要案例：

－ivh：安装显示安装进度--install--verbose--hash  
－Uvh：升级软件包--Update；  
－qpl： 列出RPM软件包内的文件信息[Query Package list]；  
－qpi：列出RPM软件包的描述信息[Query Package install package(s)]；  
－qf：查找指定文件属于哪个RPM软件包[Query File]；  
－Va：校验所有的 RPM软件包，查找丢失的文件[View Lost]；  
－e：删除包

rpm -q samba //查询程序是否安装

rpm -ivh /media/cdrom/RedHat/RPMS/samba-3.0.10-1.4E.i386.rpm //按路径安装并显示进度  
rpm -ivh --relocate /=/opt/gaim gaim-1.3.0-1.fc4.i386.rpm    //指定安装目录

rpm -ivh --test gaim-1.3.0-1.fc4.i386.rpm　　　 //用来检查依赖关系；并不是真正的安装；  
rpm -Uvh --oldpackage gaim-1.3.0-1.fc4.i386.rpm //新版本降级为旧版本

rpm -qa | grep httpd　　　　　 ＃[搜索指定rpm包是否安装]--all搜索\*httpd\*  
rpm -ql httpd　　　　　　　　　＃[搜索rpm包]--list所有文件安装目录

rpm -qpi Linux-1.4-6.i368.rpm　＃[查看rpm包]--query--package--install package信息  
rpm -qpf Linux-1.4-6.i368.rpm　＃[查看rpm包]--file  
rpm -qpR file.rpm　　　　　　　＃[查看包]依赖关系  
rpm2cpio file.rpm |cpio -div    ＃[抽出文件]

rpm -ivh file.rpm 　＃[安装新的rpm]--install--verbose--hash  
rpm -ivh http://mirrors.kernel.org/fedora/core/4/i386/os/Fedora/RPMS/gaim-1.3.0-1.fc4.i386.rpm  
rpm -Uvh file.rpm    ＃[升级一个rpm]--upgrade  
rpm -e file.rpm      ＃[删除一个rpm包]--erase

rpm -ivh --aid \*.rpm #--aid参数解决RPM包循环依赖的问题

rpm -qf $(which inotifywait) #查找命令所在的软件包

rpm -ivh --replacepkgs 安装包 #重新安装

137.yum：linux包管理器 \*

主要选项：-y（当安装过程提示选择全部为"yes"）

--nogpgcheck 禁止进行gpg check

-q 静默模式

--disablerepo=repoidglob 临时禁用此处指定的repo

--enablerepo=repoidglob 临时启用此处指定的repo

--noplugins 禁用所有插件

显示仓库列表：repolist [all|enabled|disabled]

显示程序包：list

# yum list [all | glob\_exp1] [glob\_exp2] [...]

# yum list {available|installed|updates} [glob\_exp1] [...]

安装程序包：install package1 [package2] [...]

reinstall package1 [package2] [...] (重新安装)

升级程序包：update [package1] [package2] [...]

downgrade package1 [package2] [...] (降级)

检查可用升级：check-update

卸载程序包：remove | erase package1 [package2] [...]

查看程序包information：info [...]

查看指定的特性(可以是某文件)是由哪个程序包所提供：provides | whatprovides feature1 [feature2] [...]

清理本地缓存：clean [ packages | metadata | expire-cache | rpmdb | plugins | all ]

搜索：search string1 [string2] [...] 以指定的关键字搜索程序包名及summary信息；

查看指定包所依赖的关系：deplist package1 [package2] [...]

查看yum的事务历史：yum history

[info|list|packages-list|packages-info|summary|addon-info|redo|undo|rollback|new|sync|stats]

注意：yum -y update 升级所有包同时也升级软件和系统内核

yum -y upgrade 只升级所有包，不升级软件和系统内核

主要案例：yum -y install tree

怎么使用YUM命令升级所有RPM软件包？

yum update

怎么使用YUM命令升级单个RPM软件包

yum update 软件名称

使用YUM安装RPM软件包

yum install 软件名称

yum repolist 查看当前系统使用的源

yum list 查看yum源所有已安装的和未安装的包

yum grouplist 列出所有安装过的和未安装的套件

yum provides COMMAND

1. kill 结束进程，但是后面只能加上pid进程的号码 \*

常用信号：1 无须关闭进程而让其重读配置文件

2 终止正在运行的进程，相当于Ctrl+c

3 退出正在运行的进程

9 杀死运行中的进程（慎用）

15 终止运行中的进程

18 继续

19 暂停

主要案例：kill 3268

kill -9 pid 很危险，无论这个进程是否存在，都会杀死

139.killall命令使用进程的名称来杀死进程

主要案例：killall rsync

140.pkill 可以杀掉指定名字的进程，还可以根据各种属性杀掉进程（进程名可以不是全名，但必须是唯一） \*

主要案例：pkill rsync

pkill rsyn

1. pstree 把进程已树状结构列出 \*

主要案例：[root@db02 ~]# pstree

init─┬─atd

├─crond

├─6\*[mingetty]

├─mysqld\_safe───mysqld───21\*[{mysqld}]

├─rsyslogd───3\*[{rsyslogd}]

├─sh───mysqld───23\*[{mysqld}]

├─sshd───sshd───bash───pstree

└─udevd───2\*[udevd]

142.ps 显示当前运行的进程 \*

主要选项：a 所有与终端相关的进程

x 所有与终端无关的进程

u 以用户为中心组织进程状态信息显示

o 自定义要显示的字段列表，以逗号分隔

常用的field：pid，ni，pri，psr，pcpu，stat，comm，tty，ppid

-e 显示所有进程

-f 显示完整格式的进程信息

-F 显示更加完整格式的进程信息

-H 以层级结构显示进程的相关信息

主要案例：常用组合之一：aux

常用组合之二：-ef

常用组合之三：-eFH

常用组合之四：-eo或者axo

1. pidof 根据进程名，取其pid

主要案例：pidof sshd

1. top 实时显示系统中各个进程的资源占用情况 \*

主要选项：-d 指定刷新时间间隔，默认为3秒

-b 以批次方式显示

-n 显示多少批次

主要案例：top显示，按q退出

top -b -n 1

1. **htop 比top更加好的命令**

注意：htop是一个新的命令，所以linux系统中是没有，需要yum install htop -y，但是yum之前，需要更新一下epol源wget -O /etc/yum.repos.d/epel.repo <http://mirrors.aliyun.com/repo/epel-6.repo>

主要选项：-d 指定延迟时间间隔

-u 仅显示指定用户的进程

-s 以指定字段进行排序

主要案列：htop

1. vmstat 显示系统中各个进程的资源占用情况

主要选项：-s 显示内存统计数据

主要案例：[root@db02 ~]# vmstat 2 #每隔两秒进行一次监控

procs -----------memory---------- ---swap-- -----io---- --system-- -----cpu-----

r b swpd free buff cache si so bi bo in cs us sy id wa st

0 0 451020 64936 23864 148060 0 2 3 2 16 64 0 0 100 0 0

vmstat每一列的解释：

procs：

r：等待运行的进程的个数。CPU上等待运行的任务的队列长度

b：处于不可中断睡眠态的进程个数，被阻塞的任务队列的长度

memory：

swpd：交换内存使用总量

free：空闲的物理内存总量

buffer:用于buffer的内存总量

cache：用于cache的内存总量

swap：

si：数据进入swap中的数据速率

so：数据离开swap的速率

lo：

bi：从块设备读入数据到系统的速度

bo：保存数据至块设备的速率

system：

in：中断速率

cs：上下文，切换的速率

cpu：

us：user space

sy：system

id：idle

wa：wait

st：stolen

147.glances 实时显示系统中各个进程的资源占用情况

主要选项：-b 以byte为单位显示网上数据速率

-d 关闭磁盘I/O模块

-m 关闭mount模块

-n 关闭network模块

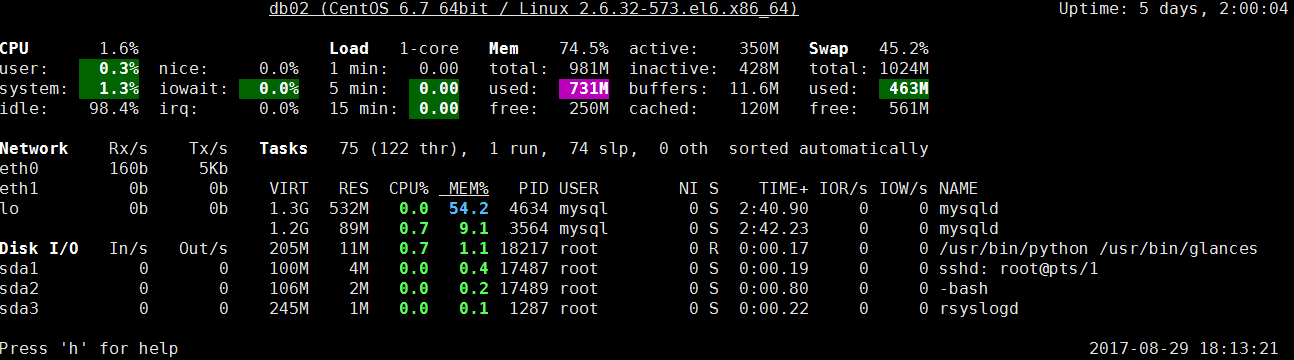
-t 刷新时间间隔

-1 每个cpu的相关数据单独显示

-o 输出格式

-f 设定输出文件的保存目录

主要案例：glances



148.dstat 实时显示系统中各个进程的资源占用情况 \*

主要选项：-c 显示cpu的相关信息

-d 显示磁盘的相关信息

-g 显示page相关的统计数据

-m 显示内存的相关统计数据

-n 显示网络的相关统计数据

-p 显示进程的相关统计数据

-r 显示io请求的相关的统计数据

-s 显示swqpped的相关统计数据

--tcp

--udp

--raw

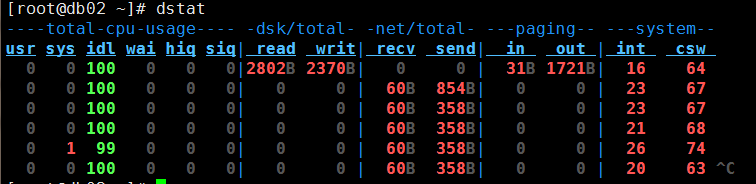
--socket

--top-cpu 显示最占用cpu的进程

--top-io 显示最占用io的进程

--top-mem 显示最占用内存的进程

--top-lantency 显示延迟最大的进程

主要案列：

1. fpm 简易的rpm打包软件 \*

主要选项：-s 指定源类型

-t 指定目标类型

-n 指定包的名字

-v 指定包的版本号

-C 指定打包的相对路径

-d 指定依赖于哪些包

-f 第二次打包时如果有同名安装包存在，则覆盖

-p 输出的安装包的目录，不想放在当前目录下就需要指定

-post-install 软件包安装完成之后所要运行的脚本

-pre-install 软件包安装完成之前所要运行的脚本

-post-uninstall 软件包卸载完成之后所要运行的脚本

-pre-uninstall 软件包卸载完成之前所要运行的脚本

主要案列：fpm -s dir -t rpm -n nginx -v 1.6.3 -d 'pcre-devel,openssl-devel' --post-install /sercer/scripts/nginx\_rpm.sh -f /application/nginx-1.6.3/

150.ulimit 用来限制系统用户对shell资源的访问。如果不懂什么意思，下面一段内容可以帮助你理解： 假设有这样一种情况，当一台 Linux 主机上同时登陆了 10 个人，在系统资源无限制的情况下，这 10 个用户同时打开了 500 个文档，而假设每个文档的大小有 10M，这时系统的内存资源就会受到巨大的挑战。 而实际应用的环境要比这种假设复杂的多，例如在一个嵌入式开发环境中，各方面的资源都是非常紧缺的，对于开启文件描述符的数量，分配堆栈的大 小，CPU 时间，虚拟内存大小，等等，都有非常严格的要求。资源的合理限制和分配，不仅仅是保证系统可用性的必要条件，也与系统上软件运行的性能有着密不可分的联 系。这时，ulimit 可以起到很大的作用，它是一种简单并且有效的实现资源限制的方式。 ulimit 用于限制 shell 启动进程所占用的资源，支持以下各种类型的限制：所创建的内核文件的大小、进程数据块的大小、Shell 进程创建文件的大小、内存锁住的大小、常驻内存集的大小、打开文件描述符的数量、分配堆栈的最大大小、CPU 时间、单个用户的最大线程数、Shell 进程所能使用的最大虚拟内存。同时，它支持硬资源和软资源的限制。 作为临时限制，ulimit 可以作用于通过使用其命令登录的 shell 会话，在会话终止时便结束限制，并不影响于其他 shell 会话。而对于长期的固定限制，ulimit 命令语句又可以被添加到由登录 shell 读取的文件中，作用于特定的 shell 用户。

主要参数：-a：显示目前资源限制的设定；

-H：设定资源的硬性限制，也就是管理员所设下的限制；

-n <文件数目>：指定同一时间最多可开启的文件数；

-S：设定资源的弹性限制；

主要案列：ulimit -SHn 65535

1. hwclock 显示硬件时钟

主要选项：-s 让系统时间同步硬件时间

-w 让硬件时间同步系统时间

主要案列：hwclock

152.groupmod 组属性修改

主要参数：-n 修改组名

-g 新的GID

主要案例：groupmod -g 1000 jf

153.gpasswd 组密码修改 \*

主要参数：-a 将用户添加至指定组中

-d 从组中移除用户

-A 设置有管理权限的用户列表（设置之后，在sudo管理文件可以看见）

-M 添加多个用户到一个组

主要案例：gpasswd -a jf test

gpasswd -A jf oldgirl root

154.newgrp 临时切换基本组

如果用户不属于此组，则需要组密码

主要参数：- 会模拟用户重新登陆以实现重新初始化其工作环境

主要案例：newgrp test

1. iconv 用来转换文件的编码方式

**156.ss 网络状态工具**（比netstat更加强大的工具） \*

主要参数：-h, --help 帮助信息

-n, --numeric 不解析服务名称

-r, --resolve 解析主机名

-o, --options 显示计时器信息

-e, --extended 显示详细的套接字（sockets）信息

-p, --processes 显示使用套接字（socket）的进程

-i, --info 显示 TCP内部信息

-s, --summary 显示套接字（socket）使用概况

-t, --tcp 仅显示 TCP套接字（sockets）

-u, --udp 仅显示 UCP套接字（sockets）

-d, --dccp 仅显示 DCCP套接字（sockets）

-l 监听状态的连接

-a 所有状态的连接

-m 内存用量

主要案例：ss -lntup|grep rsync

tcp LISTEN 0 5 :::873 :::\* users:(("rsync",1329,5))

tcp LISTEN 0 5 \*:873 \*:\* users:(("rsync",1329,3))

157.sshpass 非交互的ssh操作

主要参数：-p 端口

主要案列：sshpass -p66207383 ssh 172.16.1.7 hostname

158.set 设置shell \*

主要参数：-C 禁止使用输出重定向覆盖已存在的文件 如果想强制覆盖输出，可使用：>|

主要案例：set -C 禁止使用输出重定向覆盖已存在的文件

set +C 关闭上诉功能

159.install 安装或升级软件或备份数据，它的使用权限是所有用户。install命令和cp命令类似，都可以将文件/目录拷贝到指定的地点。但是，install允许你控制目标文件的属性。install通常用于程序的makefile，使用它来将程序拷贝到目标（安装）目录。

主要参数：-m 设定目标文件权限，默认为755

-o 设定目标文件属主

-g 设定目标文件属组

-d 创建目录

主要案列：install -m 644 /etc/passwd /tmp/

install -m 644 -o jingfeng -g jf /etc/passwd /tmp/

install -d jingfeng/

160.mktemp 创建一个临时的随机命名的文件或者目录 注意：一个月之后就会被删除

主要参数：-d 创建临时目录

主要案列：mktemp /tmp/jingfeng.xxx

1. uptime 查看系统平均负载 \*

主要案例：[root@db02 ~]# uptime

14:02:28 up 6 days, 4:06, 1 user, load average: 0.00, 0.00, 0.00

1. 当前时间为：14:02:28
2. 系统已运行的时间：6 days, 4:06
3. 当前在线用户：1 user
4. 平均负载：0.00, 0.00, 0.00，最近1分钟、5分钟、15分钟系统的负载
5. export 查看环境变量 \*

主要案例：[root@db02 ~]# export

declare -x MAIL="/var/spool/mail/root"

declare -x MODULEPATH="/usr/share/Modules/modulefiles:/etc/modulefiles"

declare -x MODULESHOME="/usr/share/Modules"

declare -x OLDPWD

declare -x PATH="/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin:/root/bin"

declare -x PWD="/root"

declare -x SHELL="/bin/bash"

declare -x SHLVL="1"

declare -x SSH\_CLIENT="10.0.0.1 62252 22"

declare -x SSH\_CONNECTION="10.0.0.1 62252 10.0.0.52 22"

declare -x SSH\_TTY="/dev/pts/1"

declare -x TERM="xterm"

declare -x USER="root"

1. readonly 只读变量

注意：只读变量无法重新赋值，并且不支持撤销。存活时间为当前shell进程的生命周期，随shell进程终止而终止

主要案例：readonly name

164.patch 向文件打补丁

主要选项：-i 读取指定的修补文件

主要案列：patch jingfeng.new jingfeng.txt

165.访问控制列表facl：文件的额外赋权机制，在原来的u，g，o之外，另一层让普通用户能控制赋权给另外用户或组的赋权机制

getfacl命令：查看访问控制列表

getfacl FILE.......

setfacl命令：设置访问控制列表

赋权给用户：

setfacl -m u:USERNAME:MODE FILE......

赋权给组：

setfacl -m g:GROUPNAME:MODE FILE.....

撤销赋权：

setfacl -x u:USERNAME FILE.....

setfacl -x g:GROUPNAME FILE.....

166.mke2fs:ext系列文件系统专用管理工具

主要选项：-t 指定要创建的文件系统类型

-b 指定文件系统的块大小

-L 指明卷标

-J 创建有日志功能的文件系统

-i 指定inode与字节的比率。即每多少字节创建一个inode

-N 直接指明要给此文件系统创建的inode数量

-O 已指定的特性创建目标文件系统

-m 指定预留的空间

主要案例：mke2fs -t ext4 /mnt/jingfeng/

167.e2label：卷标的查看与设定

查看：e2label device

设定：e2label device LABEL

168.blkid：查看文件系统信息

主要选项：-L 根据LABEL定位设备

-U 根据UUID定位设备

主要案例：blkid /dev/sda3

169.watch：实时监控每一条命令执行的结果动态变化

主要选项：-n #：刷新间隔，单位是秒

主要案例：watch -n 2 `ifconfig`

170.pvdisplay：显示卷组中一个或多个物理卷的信息（LVM专用）

171.pvcreate：创建一个可以被卷组使用的物理卷（LVM专用）

172.vgdisplay 显示卷组信息（LVM专用）

173.vgcreate 创建一个卷组（LVM专用）

174.vgremove 从系统上删除一个或多个卷组的定（LVM专用）

175.vgextend 通过添加物理卷扩展一个卷组（LVM专用）

176.vgreduce 通过删除一个或多个物理卷减小卷组（LVM专用）

177lvdisplay 显示逻辑卷的信息（LVM专用）

178.lvcreate 在卷组中创建一个逻辑卷（LVM专用）

主要参数：-L 指定逻辑卷大小

-s 创建快照（类似于虚拟机上面的快照）

-n 指明逻辑卷的名字

-p 指明快照的权限（r，w）

主要案例：lvcreate -L #[mMgGtT] -p r -s -n snapshot\_lv\_name original\_lv\_name

179.lvextend 增加分配给逻辑卷的物理区域数（LVM专用）

主要参数：-L 指定逻辑卷大小

180.lvreduce 减少分配给逻辑卷的物理区域数（LVM专用）

主要参数：-L 指定逻辑卷大小

181.lvremove 从卷组中删除一个或多个逻辑卷（LVM专用）

主要案例：lvremove /dev/VG\_NAME/LV\_NAME

182.gzip 压缩工具（一般压缩以后，文件的后缀名是gz）

主要参数：-d: 解压缩，相当于gunzip

-c: 将结果输出至标准输出；

-#：1-9，指定压缩比，默认为6；

主要案例：gzip [OPTION]... FILE ...

183.zcat：可以直接查看gz结尾的压缩文件

主要案例：zcat jingfeng.gz

184.bgzip2：也是一个压缩工具（后缀名是bz2）

主要参数：-k: keep, 保留原文件；

-d：解压缩

-#：1-9，压缩比，默认为6；

主要案例：bzip2 [OPTION]... FILE ...

185.bzcat：可以直接查看bz2结尾的压缩文件

主要案例：bzcat jingfeng.bz2

186.xz：压缩工具（后缀名是xz）

主要参数：-k: keep, 保留原文件；

-d：解压缩

-#：1-9，压缩比，默认为6；

主要案例：xz [OPTION]... FILE ...

187.xzcat：可以直接查看xzcat结尾的压缩文件

主要案列：xzcat jingfeng.xz

188.at：一次性任务计划（这条命令要启用，需要开启进程atd，开启用/etc/init.d/atd start or restart

TIME：（时间格式，这里可以定义出什么时候要进行at这项任务的时间，选项有以下几项）

HH:MM：[YYYY-MM-DD]

noon,midnight,teatime

tomorrow

now+

主要参数：-l 查看作业队列

-f<文件> 从指定文件读入任务而不是从标准输入读入

-d 删除指定的作业

-c 查看指定作业的具体内容

-q 指明队列

主要案例：# at now+2min

at> echo "wo ai ni"

at> ls -l /root

退出按ctrl+d

注意：作业执行结果是以邮件方式发送给用户

1. lsmod：列出内核已载入模块的状态（几乎用不到）

lsmod以美观的方式列出/proc/modules的内容

主要案例：lsmod

1. modinfo：显示内核模块的信息（几乎用不到）

主要选项：-F 仅显示指定字段的信息

-n 显示文件路径

主要案例：modinfo [ -0 ] [ -F field] [modulename | filename ... ]

1. modprobe：linux内核添加删除模块（几乎用不到）

主要案例：modprobe [-r] module\_name

模块的动态装载：modprobe module\_name

动态卸载：modprobe -r module\_name

192.depmod：分析可加载模块的依赖性，生成modules.dep文件和映射文件（几乎用不到，只是了解一下）

1. insmod：向linux内核中插入一个模块（几乎用不到）

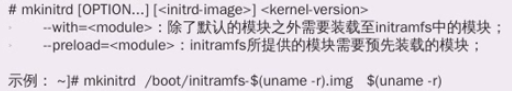
主要案例：insmod [filename] [modue options ...]

1. rmmod：删除内核中的一些模块（几乎用不到）

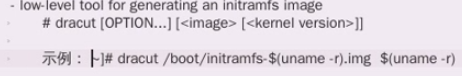
主要案例：rmmod [ -f ] [ -w ] [ -s ] [ -v ] [ modulename ]

1. mkinitrd：为当前使用中的内核重新制作ramdisk文件（几乎用不到，只是了解）

主要案例：



1. dracut命令（几乎用不到，只是了解）



1. nload 查看网卡流量

安装过程：1.yum install -y epel-release

2.yum install -y nload

主要案列：nload

