

Linux入门

学习目标

1. 能够独立搭建Linux环境
2. 能够安装SSH Secure客户端连接Linux
3. 能够使用Linux进行目录操作的命令
4. 能够使用Linux进行文件操作的命令
5. 能够使用Linux进行目录文件压缩和解压的命令
6. 能够使用Linux进行目录文件权限的命令
7. 能够使用其它常用的Linux命令

第1章 学习Linux的目的

对于windows操作系统而言，大家应该不陌生，这里我列举一些windows的不足：

1. 个人用户正版windows需要收费
2. 系统长时间运行不稳定，变慢，容易死机
3. windows常招到病毒攻击

相反，上述windows的不足，恰好是另一款操作系统Linux的优势所在，这里我也列举一些Linux的优点：

1. 个人用户正版Linux不需要收费或安装特殊功能需要收费
2. 系统长时间运行稳定，不会变慢，不容易死机
3. Linux不常招到病毒攻击，也很难攻击

总之，做为一个后端javaEE程序员，在windows中开发完程序后，得部署到一个相对比较安全稳定的服务器中运行，这台服务器上安装的不是windows操作系统，而是Linux操作系统。

为了顺利的操作Linux操作系统，我们得学会：安装Linux操作系统，对目录，文件，权限等基本操作命令，所以学会对Linux操作系统的基本操作是后端javaEE程序员的必修课。

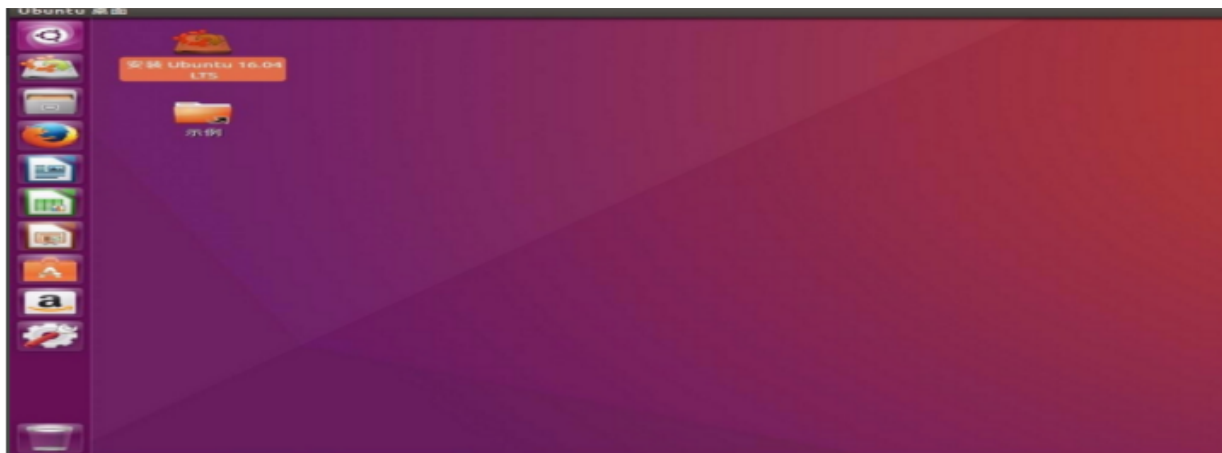
第2章 Linux的概念

Linux是基于Unix的开源免费，多用户，用任务的操作系统，由于系统的稳定性和安全性几乎成为程序代码运行的最佳系统环境。

第3章 Linux的分类

3.1根据市场需求不同，分为两个方向

1. 图形化界面版：注重用户体验，但目前成熟度不够。



2. 服务器版：没有好看的界面，是以在控制台窗口中输入命令操作系统的，是我们架设服务器的最佳选择，类似于DOS界面。

```
CentOS release 6.5 (Final)
Kernel 2.6.32-431.el6.i686 on an i686

JavaEE63 login: root
Password:
Last login: Wed Jul 19 05:34:54 on tty1
[root@JavaEE63 ~]# cd /
[root@JavaEE63 /]# ls
bin    cgroup  etc     lib          media  mnt    opt    root  selinux  sys  usr
boot  dev     home   lost+found  misc   net    proc   sbin  srv      tmp  var
[root@JavaEE63 /]# _
```

3.2 根据原生程度不同，分为两个方向

1. 内核版本：在Linux之父领导下的内核小组开发维护的系统内核的版本号。
2. 发行版本：一些个人/组织/公司在内核版基础上进行二次开发而重新发行的版本号。

3.3 根据发行不同，分为多个版本

1. ubuntu
2. redhat
3. centos
4. lubuntu
5. freeBSD

6. 等等



第4章 Linux的安装

我们今天安装的是：服务器版 + 发行版 + CentOS版

4.1 虚拟机简介

虚拟机是一个软件，它可以使你在一台真实PC机器上同时运行二个或更多window / linux等操作系统。它可以模拟一个标准PC环境。这个环境和真实的计算机一样，都有芯片组、CPU、内存、显卡、声卡、网卡、软驱、硬盘、光驱、串口、并口、USB控制器。

目前市场上流行的虚拟机有两种：

1. VMware（威睿）公司的虚拟机软件，功能强大，收费产品，有30天试用期
2. VirtualBox（甲骨文）公司的虚拟机软件，免费商品

4.2 安装Virtual Box虚拟机

参见<< 01_安装VirtualBox步骤.pdf >>

参见<< 02_在Virtual Box中创建虚拟机步骤.pdf >>

4.3 安装CentOS 7.0系统

参见<< 03_在虚拟机中安装CentOS 7.0步骤.pdf >>

第5章 客户端工具使用

5.1 SSH Secure简介

在实际开发中，Linux服务器都在其他的地方，我们要通过windows客户端工具远程去连接Linux并操作它，连接Linux的windows客户端工具有很多，企业中常用的有Putty、secureCRT、SSH Secure等。

课程中我们使用SSH Secure客户端工具进行连接Linux操作系统，该工具是免费的图形化界面和命令行窗口集一身客户端工具。

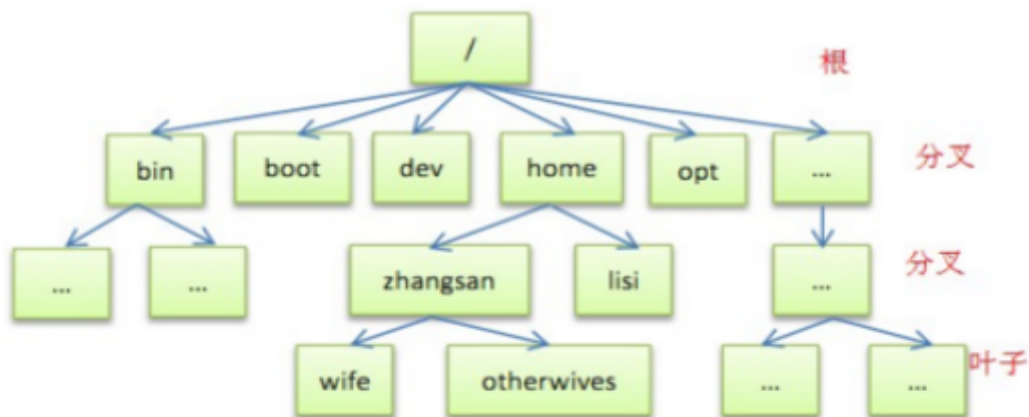
5.2 安装SSH Secure

参见<< 04_安装SSH Secure 步骤.pdf>>

5.3 用SSH Secure连接CentOS

参见<< 05_SSH Secure客户端连接CentOS步骤.pdf >>

第6章 Linux 的目录结构



/bin 二进制可执行命令

/dev 设备特殊文件

/etc 系统管理和配置文件

/etc/rc.d 启动的配置文件和脚本

/home 用户主目录的基点，比如用户user的主目录就是/home/user，可以用~user表示

/lib 标准程序设计库，又叫动态链接共享库，作用类似windows里的.dll文件

/sbin 超级管理命令，这里存放的是系统管理员使用的管理程序

/tmp 公共的临时文件存储点

/root 系统管理员的主目录

/mnt 系统提供这个目录是让用户临时挂载其他的文件系统

/lost+found这个目录平时是空的，系统非正常关机而留下“无家可归”的文件（windows下叫什么.chk）就在这里

/proc 虚拟的目录，是系统内存的映射。可直接访问这个目录来获取系统信息。

/var 某些大文件的溢出区，比方说各种服务的日志文件

/usr 最庞大的目录，要用到的应用程序和文件几乎都在这个目录，其中包含：

上述Linux目录结构中，请大家关注：

1. root目录：超级管理员所在的目录，用~表示
2. home目录：普通用户所在的目录
3. usr目录：安装用户文件所在的目录
4. etc目录：Linux系统管理和配置文件所在的目录

第7章 目录操作

7.1 增加

语法: mkdir 目录

创建目录

命令:

cd /	定位于根目录
mkdir aaa	在当前目录下创建aaa目录, 相对路径
mkdir ./bbb	在当前目录下创建bbb目录, 相对路径
mkdir /ccc	在根目录下创建ccc目录, 绝对路径

```
[root@localhost ~]# cd /
[root@localhost ~]# mkdir aaa
[root@localhost ~]# mkdir ./bbb
[root@localhost ~]# mkdir /ccc
[root@localhost ~]# ls
aaa  bin  ccc  etc  lib  media  opt  root  sbin  sys  usr
bbb  boot  dev  home  lib64  mnt  proc  run  srv  tmp  var
```

7.2 查看

语法: ls [参数]

查看目录下的内容

参数解释:

1. -l 以详细形式查询当前目录下所有文件和目录, 不包括隐藏文件和目录
2. -a 以精简形式查询当前目录下所有文件和目录, 包括隐藏文件和目录

命令:

cd /	
ls	以精简形式查询当前目录下的内容
ls -l	以详细形式查询当前目录下的内容, d表示目录, -表示文件
ls -a	以精简形式查询当前目录下的隐藏内容
ls -la	以详细形式查询当前目录下的隐藏内容, la和al都可
clear	清屏



```
[root@localhost /]# ls -l
total 16
drwxr-xr-x.  2 root root    6 Feb  1 20:05 aaa
drwxr-xr-x.  2 root root    6 Feb  1 20:05 bbb
lrwxrwxrwx.  1 root root    7 Jan 27 23:13 bin -> usr/bin
dr-xr-xr-x.  5 root root 4096 Jan 27 23:23 boot
drwxr-xr-x.  2 root root    6 Feb  1 20:05 ccc
drwxr-xr-x. 19 root root 3040 Feb  1 19:41 dev
drwxr-xr-x. 75 root root 8192 Feb  1 19:41 etc
drwxr-xr-x.  2 root root    6 Nov  5  2016 home
lrwxrwxrwx.  1 root root    7 Jan 27 23:13 lib -> usr/lib
lrwxrwxrwx.  1 root root    9 Jan 27 23:13 lib64 -> usr/lib64
drwxr-xr-x.  2 root root    6 Nov  5  2016 media
drwxr-xr-x.  2 root root    6 Nov  5  2016 mnt
drwxr-xr-x.  2 root root    6 Nov  5  2016 opt
dr-xr-xr-x. 97 root root    0 Feb  1 19:41 proc
dr-xr-xr-x.  2 root root  135 Jan 27 23:27 root
drwxr-xr-x. 23 root root   720 Feb  1 19:41 run
lrwxrwxrwx.  1 root root    8 Jan 27 23:13/sbin -> usr/sbin
drwxr-xr-x.  2 root root    6 Nov  5  2016 srv
dr-xr-xr-x. 13 root root    0 Feb  1 19:41 sys
drwxrwxrwt.  8 root root   211 Feb  1 19:41 tmp
drwxr-xr-x. 13 root root   155 Jan 27 23:13 usr
drwxr-xr-x. 19 root root   267 Jan 27 23:23 var
```

d表示目录
表示快捷方式

7.3 搜索

语法: find 目录 -name '条件'

根据条件, 搜索目录下的内容

命令:

```
cd /
find / -name 'b'      查询根目录下 (包括子目录), 名为b的目录和文件
find / -name 'b*'     查询根目录下 (包括子目录), 名以b开头的目录和文件
在Linux中, 按ctrl+c  强行终止程序执行
```

```
[root@localhost /]# find / -name 'b'
/var/lib/yum/yumdb/b
/usr/share/terminfo/b
```

7.4 修改

语法: mv 旧目录 新目录

对目录进行重命名

命令:

```
cd /
mv aaa aaaa          将根目录下的aaa目录修改为aaaa
```



```
[root@localhost /]# ls
aaa  bin  ccc  etc  lib  media  opt  root  sbin  sys  usr
bbb  boot  dev  home  lib64  mnt  proc  run  srv  tmp  var
[root@localhost /]# mv aaa aaaa
[root@localhost /]# ls
aaaa  bin  ccc  etc  lib  media  opt  root  sbin  sys  usr
bbb  boot  dev  home  lib64  mnt  proc  run  srv  tmp  var
```

旧目录名

新目录名

7.5 剪切

语法：mv 原目录位置 新目录的位置

对目录进行移动

命令：

```
cd /
mv /aaaa /bbb      将根目录下的aaaa目录，移动到bbb目录下，在bbb目录下也叫aaaa目录
mv bbbb usr/bbbbb  将当前目录下的bbbb目录，移动到usr目录下，且修改名为bbbbbb
```

```
[root@localhost /]# ls
aaaa  bin  ccc  etc  lib  media  opt  root  sbin  sys  usr
bbb  boot  dev  home  lib64  mnt  proc  run  srv  tmp  var
[root@localhost /]# mv /aaaa /bbb
[root@localhost /]# cd /bbb
[root@localhost bbb]# ls
aaaa
```

aaaa原来在/目录下

aaaa现在在/bbb目录下

7.6 拷贝

语法：cp [参数] 目录 目录的新位置

对目录进行复制

参数解释：

1. -r 递归处理，将指定目录下的文件与子目录一并处理

命令：

```
cd /      定位于/目录
cp -r /aaa /bbb      将/目录下的aaa目录复制到/bbb目录下，在/bbb目录下名字为aaa
cp -r /aaa /bbb/aaaa  将/目录下的aaa目录复制到/bbb目录下，且修改名为aaaa
```

```
[root@localhost /]# ls
aaa  bin  ccc  etc  lib  media  opt  root  sbin  sys  usr
bbb  boot  dev  home  lib64  mnt  proc  run  srv  tmp  var
[root@localhost /]# cp -r /aaa /bbb
[root@localhost /]# cd /bbb
[root@localhost bbb]# ls
aaa
```

aaa在/目录下

aaa也复制到bbb目录下

7.7 删除

语法: rm [参数] 目录

对目录进行删除

参数解释:

1. -r 递归处理, 将指定目录下的所有文件及子目录一并处理, 它会询问你, y表示是删除, n表示不删除
2. -f 强制删除目录, 它不会询问你

命令:

```
cd /
rm -r /aaa    询问式删除/目录下的aaa目录, y表示确认, n表示取消
               如果aaa目录中还有子目录, 则进入aaa目录下的所有子目录, 依次询问
rm -rf /bbb   强行式删除/目录下的bbb目录
               如果bbb目录中还有子目录, 也会被强行式删除
```

```
[root@localhost /]# ls
aaa  bin  ccc  etc  lib  media  opt  root  sbin  sys  usr
bbb  boot dev  home lib64 mnt  proc  run  srv  tmp  var
[root@localhost /]# rm -r aaa
rm: remove directory 'aaa'? y
[root@localhost /]# rm -rf bbb
```

询问式删除/目录下的aaa

强行式删除/目录下的bbb

第8章 文件操作

8.1 增加

语法: touch 文件

在指定的目录中, 创建文件

命令:

```
cd /aaaa      定位于/aaaa目录
touch a.java   在aaaa目录中创建a.java文件
touch b.txt    在aaaa目录中创建b.txt文件
touch c.xml    在aaaa目录中创建c.xml文件
```

```
[root@localhost /]# mkdir aaa
[root@localhost /]# ls
aaa  boot  etc  lib  media  opt  root  sbin  sys  usr
bin  dev  home lib64 mnt  proc  run  srv  tmp  var
[root@localhost /]# cd aaa
[root@localhost aaa]# touch a.java
[root@localhost aaa]# ls
a.java
[root@localhost aaa]#
```

在aaa目录下, 创建a.java文件

8.2 查看

语法: cat/more/head/tail 文件

以只读方式查看文件的内容

命令：

```
cd /aaaa
cat a.java          查看a.java文件的最后一页内容
more a.java         从第一页开始查看a.java文件内容，按回车键一行一行的看，按空格键一页一页的看
head a.java         查看a.java文件的前10行内容
tail a.java         查看a.java文件的后10行内容
head -n 7 a.java    查看a.java文件的前7行内容
tail -n 7 a.java    查看a.java文件的后7行内容
```

```
[root@localhost aaa]# more a.java
hello
how
are
you
zhao
jun
1
2
3
4
5
```

以只读方式
查询/aaa目录
下的a.java
文件的内容

8.3 删除

语法：rm [参数] 文件

从目录中删除文件

参数解释：

1. -r 递归处理，对于文件删除，这个参数可带，也可不带，它会询问你，y表示是删除，n表示不删除
2. -f 强制删除文件，它不会询问你

命令：

```
cd /aaaa
rm -rf a.java      强行式删除当前目录下的a.java文件
rm -rf ./a*        强行式删除当前目录下以a开头的所有文件
rm -rf ./          强行式删除当前目录下所有文件
```

```
[root@localhost aaa]# ls
a.java
[root@localhost aaa]# rm -r a.java
rm: remove regular file 'a.java'? y
```

删除a.java文件，
必须位于a.java
文件所在的目录

8.4 修改

修改已存在文件的内容

8.4.1 vi编辑器

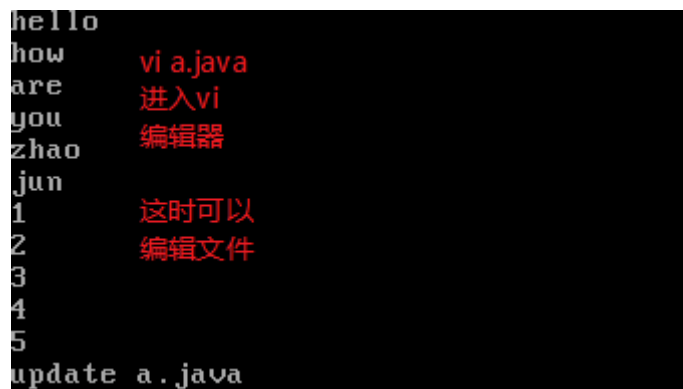
linux 下的vi 是一种文字编辑器，后来的升级版本是vim。vi 分为三种模式：一般模式、编辑模式、命令命令模式。

8.4.2 vi编辑器的使用

语法：vim 文件

命令：

```
vim a.java          用vi编辑器打开/创建a.java文件
按i键，进入编辑模式
输入Java代码
按esc键，进入一般模式
按 Shift : 进行命令模式
输入wq回车，表示存盘退出，输入q!回车，表示不存盘强行退出
```



第9章 压缩

windows的压缩文件的扩展名：.zip或.rar

Linux中的打包文件扩展名：.tar

Linux中的压缩文件扩展名：.gz

Linux中打包压缩的文件扩展名：.tar.gz

9.1 打包压缩包

压缩语法：tar [参数] 打包压缩后的文件名 要打包压缩的文件，支持通配符*号

将多个文件打包并压缩成一个特定的文件

参数解释：

1. -c 创建新的文件（必选项）
2. -x 取出文件中的内容（必选项）
3. -f 文件名由命令台设置（必选项）
4. -v 输出文件清单（可选项）
5. -z 自动识别压缩或解压（可选项）

命令：



```
cd /          定位于/目录
mkdir aaa     创建aaa目录
cd    aaa     定位于aaa目录
touch a.java  创建a.java文件
touch b.java  创建b.java文件
touch c.java  创建c.java文件
tar -zcvf zhaojun.tar.gz a.java b.java c.java
或
tar -zcvf zhaojun.tar.gz ./*  将当前目录下的所有文件打包压缩成zhaojun.tar.gz文件
```

```
[root@localhost aaal# ls
[root@localhost aaal# touch a.java
[root@localhost aaal# touch b.java
[root@localhost aaal# touch c.java
[root@localhost aaal# ls
a.java b.java c.java
[root@localhost aaal# tar -zcvf zhaojun.tar.gz ./*
./a.java          打包压缩aaa目录下面所有文件为：
./b.java          zhaojun.tar.gz
./c.java
[root@localhost aaal# ls
a.java b.java c.java zhaojun.tar.gz
```

9.2 解压压缩包

解压语法: tar [参数] 压缩文件

解压后的文件在当前目录中

解压语法: tar [参数] 压缩文件 -C/usr/local

解压后的文件放入/usr/local目录中，注意-C是大写字母C，后面不用接空格符

将一个特定的文件解包成多个文件，放在指定的目录下

参数解释:

1. -c 创建新的文件 (必选项)
2. -x 取出文件中的内容 (必选项)
3. -f 文件名由命令台设置 (必选项)
4. -v 输出文件清单 (可选项)
5. -z 自动识别压缩或解压 (可选项)

命令:

```
cd /soft      定位于/目录
rm -rf ./*.java 强制式删除当前目录下所有扩展名为java的文件
tar -xvf zhaojun.tar.gz 解压zhaojun.tar.gz这个压缩文件，里面的文件释放到当前目录下
tar -xvf zhaojun.tar.gz -C/usr/local 解压zhaojun.tar.gz这个压缩文件，里面的文件释放到/usr/local目录下
```



```
[root@localhost aaa]# ls
zhaojun.tar.gz
[root@localhost aaa]# tar -xvf zhaojun.tar.gz
./a.java      将aaa目录下的zhaojun.tar.gz这个打包压缩文件进行解压，
./b.java      解压后的文件就放在aaa目录下
./c.java
[root@localhost aaa]# ls
a.java b.java c.java zhaojun.tar.gz
```

第10章 其他

10.1 查看当前工作路面

语法：pwd

命令：

```
cd /
pwd
cd /usr/local
pwd
```

```
[root@localhost /]# pwd
/
[root@localhost /]# cd bin
[root@localhost bin]# pwd
/bin
[root@localhost bin]# cd /usr/bin
[root@localhost bin]# pwd
/usr/bin
```

当前工作目录

10.2 查看进程

语法：ps [参数]

进程，正在执行的一个应用程序

参数解释：

1. 无 显示当前用户通过终端启动的所有进程
2. -a 显示所有用户通过终端启动的所有进程
3. -u 显示所有用户通过终端启动的所有进程详细信息
4. -x 显示所有用户通过终端和非终端启动的所有进程详细信息，内容相当多，少用

命令：

```
cd /
ps      显示当前用户通过终端启动的所有进程
ps a    显示所有用户通过终端启动的所有进程
ps au   显示所有用户通过终端启动的所有进程详细信息，包括进程号，作者，CPU占用率等，常用
ps aux  显示所有用户通过终端和非终端启动的所有进程详细信息，不常用
```

```
[root@localhost ~]# ps
  PID TTY          TIME CMD
 1128 tty1        00:00:02 bash
 3756 tty1        00:00:00 ps
[root@localhost ~]# ps -a
  PID TTY          TIME CMD
 3759 tty1        00:00:00 ps
[root@localhost ~]# ps -au
USER      PID %CPU %MEM    USZ    RSS TTY      STAT START   TIME COMMAND
root      1128  0.0  0.2 115524 2216 tty1    Ss   19:41   0:02 -bash
root      3770  0.0  0.1 151064 1796 tty1    R+   20:58   0:00 ps -au
```

当前用户进程

所有用户进程

10.3 杀死进程

语法: kill [参数] 进程号

最好只删当前用户在终端中启动的进程，而不要删除其它用户的进程，尤其是root用户的进程。

先用ps au来确认一下当前需要杀死进程号，再用kill 进程号将这个进程杀死

参数解释

1. 无 普通式删除进程号对应的进程
2. -9 强制式删除进程号对应的进程

命令:

```
ps au          显示所有用户通过终端启动的所有进程详细信息
kill 1114       杀死1114进程号，如果这个1114的进程杀不死，用下面这个命令:
kill -9 1114    强制性杀死1114进程号
```

```
[root@localhost ~]# ps -au
USER      PID %CPU %MEM    USZ    RSS TTY      STAT START   TIME COMMAND
root      1128  0.0  0.2 115524 2216 tty1    Ss   19:41   0:02 -bash
root      3898  0.0  0.1 151064 1796 tty1    R+   21:01   0:00 ps -au
[root@localhost ~]# kill -9 1128
```

强行删除1128进程

10.4 搜索

语法: grep [参数] 搜索字符串 文件名

grep是Linux中强大的文本搜索命令。

参数解释:

1. -n 显示匹配行及行号，大小写敏感
2. -v 显示不匹配行及行号，大小写敏感
3. -i 搜索时不区分大小写，大小写不敏感

命令:

```

cd /
grep as 123.txt          在123.txt文件中搜索as字符串，大小写敏感，显示行
grep -n as 123.txt       在123.txt文件中搜索as字符串，大小写敏感，显示行及行号
grep -v as 123.txt       在123.txt文件中搜索as字符串，大小写敏感，显示没搜索到行
grep -i as 123.txt       在123.txt文件中搜索as字符串，大小写不敏感，显示行
grep -ni "Q q" 123.txt    在123.txt文件中搜索Q q字符串，大小写不敏感，显示行及行号
    
```

```

[root@localhost aal]# ls
a.java b.java c.java zhaojun.tar.gz
[root@localhost aal]#
[root@localhost aal]# grep how a.java
how
[root@localhost aal]# grep -n how a.java
2:how
[root@localhost aal]# grep -ni how a.java
2:how
    
```

在a.java文件中搜索
内容为"how"的
字符串

10.5 管道

语法：

一个命令的输出，可以作为另一个命令的输入，至少有二个命令参与执行。

常用的需要使用管道命令有more，grep。

命令：

```

cd /
ps aux | more          分屏显示Linux中所有用户通过终端和非终端启动的所有进程的详细信息
ps aux | grep -i a      在ps aux的结果中，搜索字符串a，大小写不敏感，显示对应的行
    
```

```

USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START
root         1  0.0  0.6 128164 6808 ?        Ss   19:40
d/systemd --switched-root --system --deserialize 21
root         2  0.0  0.0      0     0 ?        S    19:40
root         3  0.0  0.0      0     0 ?        S    19:40
root         5  0.0  0.0      0     0 ?        S<   19:40
root         7  0.0  0.0      0     0 ?        S    19:40
root         8  0.0  0.0      0     0 ?        S    19:40
root         9  0.0  0.0      0     0 ?        R    19:40
root        10  0.0  0.0      0     0 ?        S    19:40
root        12  0.0  0.0      0     0 ?        S    19:40
root        13  0.0  0.0      0     0 ?        S<   19:40
root        14  0.0  0.0      0     0 ?        S    19:40
root        15  0.0  0.0      0     0 ?        S<   19:40
root        16  0.0  0.0      0     0 ?        S<   19:40
root        17  0.0  0.0      0     0 ?        S<   19:40
root        18  0.0  0.0      0     0 ?        S<   19:40
root        19  0.0  0.0      0     0 ?        S<   19:40
root        25  0.0  0.0      0     0 ?        S    19:40
root        26  0.0  0.0      0     0 ?        SN   19:40
root        27  0.0  0.0      0     0 ?        SN   19:40
root        28  0.0  0.0      0     0 ?        S<   19:40
root        36  0.0  0.0      0     0 ?        S<   19:40
root        37  0.0  0.0      0     0 ?        S    19:40
--More--
    
```

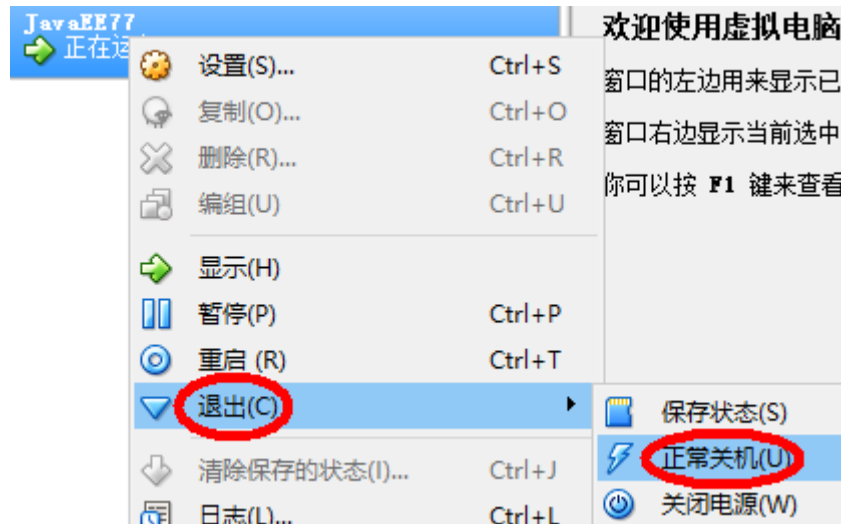
ps aux | grep

10.6 关机

语法: halt

命令:

```
cd /
halt    正常关闭安装Linux的虚拟机
```



10.7 重启

语法: reboot

命令:

```
cd /
reboot    重新启动安装Linux的虚拟机
```

```
CentOS Linux 7 (Core)
Kernel 3.10.0-693.el7.x86_64 on an x86_64

localhost login: root
                123456
```

10.8 文件权限

10.8.1 权限的简介

Linux中对每个目录和文件都做了规定，只能由满足条件的用户才能操作，这个规定叫权限。



属主 (user)			属组 (group)			其他用户		
r	w	x	r	w	x	r	w	x
4	2	1	4	2	1	4	2	1

10.8.2 操作权限

10.8.2.1 添加权限

语法: `chmod + [参数] 目录/文件`

+: 所有用户添加权限

参数解释:

1. -r 只读
2. -w 只写
3. -x 只执行

命令:

```
chmod +x a.java    增加所有用户对a.java文件的执行权限
```

```

[root@localhost aaal# ls -l
total 8
-rw-r--r--. 1 root root  27 Feb  1 21:04 a.java
-rw-r--r--. 1 root root   0 Feb  1 20:51 b.java
-rw-r--r--. 1 root root   0 Feb  1 20:51 c.java
-rw-r--r--. 1 root root 135 Feb  1 20:51 zhaojun.tar.gz
[root@localhost aaal# chmod +x a.java
[root@localhost aaal# ls -l
total 8
-rwxr-xr-x. 1 root root  27 Feb  1 21:04 a.java
-rw-r--r--. 1 root root   0 Feb  1 20:51 b.java
-rw-r--r--. 1 root root   0 Feb  1 20:51 c.java
-rw-r--r--. 1 root root 135 Feb  1 20:51 zhaojun.tar.gz
    
```

(Diagram annotations: Red arrows point from the text 'a.java文件原来无x权限' to the first 'r' in the first line of the first ls output, and from 'a.java文件现在有x权限' to the first 'x' in the first line of the second ls output.)

10.8.2.2 修改权限

语法: chmod [参数] 目录/文件

可以针对不同的用户修改权限

参数解释:

1. u 作者
2. g 组员
3. o 它人
4. -r 只读
5. -w 只写
6. -x 只执行

命令:

```
chmod u=rwx,g=rx,o=x a.java
```

修改a.java文件的权限

作者具有rwx权限, 组员具有rx权限, 其它人具有x权限

```
root@localhost aaal# ls -l
total 8
-rw-r--r--. 1 root root 27 Feb 1 21:04 a.java
-rw-r--r--. 1 root root 0 Feb 1 20:51 b.java
-rw-r--r--. 1 root root 0 Feb 1 20:51 c.java
-rw-r--r--. 1 root root 135 Feb 1 20:51 zhaojun.tar.gz
root@localhost aaal# chmod u=rwx,g=rx,o=x a.java
root@localhost aaal# ls -l
total 8
-rwxr-x--x. 1 root root 27 Feb 1 21:04 a.java
-rw-r--r--. 1 root root 0 Feb 1 20:51 b.java
-rw-r--r--. 1 root root 0 Feb 1 20:51 c.java
-rw-r--r--. 1 root root 135 Feb 1 20:51 zhaojun.tar.gz
```

a.java文件权限如下:
作者: 读, 写, 执行
组员: 读, 执行
它人: 执行

10.8.2.3 删除权限

语法: chmod - [参数] 目录/文件

-: 所有用户减少权限

参数解释:

1. -r 只读
2. -w 只写
3. -x 只执行

命令:

```
chmod -x a.java
```

减少所有用户对a.java文件的执行权限



```
root@localhost aaal# ls -l
total 8
-rwxr-xr-x 1 root root 27 Feb 1 21:04 a.java
-rw-r--r-- 1 root root 0 Feb 1 20:51 b.java
-rw-r--r-- 1 root root 0 Feb 1 20:51 c.java
-rw-r--r-- 1 root root 135 Feb 1 20:51 zhaojun.tar.gz
root@localhost aaal# chmod -x a.java
root@localhost aaal# ls -l
total 8
-rw-r--r-- 1 root root 27 Feb 1 21:04 a.java
-rw-r--r-- 1 root root 0 Feb 1 20:51 b.java
-rw-r--r-- 1 root root 0 Feb 1 20:51 c.java
-rw-r--r-- 1 root root 135 Feb 1 20:51 zhaojun.tar.gz
```

a.java文件原来有x权限

减少x权限后，
a.java文件所有
用户权限中没有
x权限了

