

# Linux基础

---

## 准备部分

材料: vmware、centos7镜像、SecureCRT

内容: 演示linux安装

IP配置

1.修改/etc/sysconfig/network-scripts/ifcfg-ens33;

2.重启网络服务 service network restart)

文件传送方法:FTP

## Part1基本命令

### 一、基本命令

#### 1.文件权限

例: -rw-r--

第一位代表文件类型, d为目录, -为文件

第二位代表创建者读权限

第三位代表创建者写权限

第四位代表创建者执行权限

第5~7位代表用户组读写可执行权限

第8~10代表其他用户权限

赋予权限命令: chmod 777 文件名

#### 2.建立目录、文件和链接

建立目录

`mkdir` 目录名

`mkdir -p` 路径 递归建立

建立文件 `touch` 文件名

建立连接 创建软链接 `ln -s` 源文件 目标文件 创建硬链接 `ln` 源文件 目标文件

#### 3.切换目录

```
cd dirname
cd -  返回上一目录
cd .. 返回上一级目录
cd ~  家目录
```

相对路径:以当前目录为起点

绝对路径:以根目录为起点

#### 4.查看当前路径

```
pwd
```

#### 5.删除指令

```
rm name
rm -rf name 强制删除
```

#### 6.复制

```
cp -r 复制文件夹和文件
    -d 复制链接
    -p 复制文件属性
    -a 相当于pdr
```

#### 7.剪切

```
mv 源路径 目标路径（两个路径相同为改名）
```

#### 8.查看文件

```
cat 文件名 | head -10
cat 文件名 | tail -10
```

#### 9.查看文件夹下内容

```
ls -a 所有的
    -l 详细
    -h 人类可读的
```

## 10.alias

更改命令别名

## 11.查看文件夹大小

```
du -h/m/g --max-depth=1
```

## 12.挂载点查看

```
df -h
```

## 二常用目录

1. / 根目录
2. /bin 普通用户指令
3. /sbin root用户指令
4. /usr/bin 系统软件资源
5. /usr/sbin 系统软件资源
6. boot 启动目录
7. dev 硬件目录
8. etc 配置文件目录
9. tmp 临时文件
10. lib 函数库
11. misc media mnt 空目录，用于挂载新硬件
12. proc /sys 内存挂载点
13. var 文档

## 三、文件搜索命令

### 1.locate 文件名；

特点:搜索速度快  
缺点:不在索引库里的文件搜索不到  
updatedb 更新索引库

## 2.find 遍历搜索

```
find 路径 -name 关键字
      -iname 不区分大小写
      -user 按照所有者搜索
      -mtime 修改时间
      -mmin 按照分钟查找
      -atime 访问时间
      -ctime 改变文件属性时间（+代表指定时间之前，-指定时间之内，无加减号
等于指定时间，单位天）
      -size 文件大小（默认为块，M为兆，k为千字节）
      -a 与
      -o 或
      -exec 命令 {} \;
```

## 3.Grep字符串搜索命令

```
grep 字符串 文件名
      -v 不包含忽略大小写
      -n 显示行号
      -i 忽略大小写
```

## 4.帮助命令

```
man 命令
```

## 5.查找命令是否存在

```
whereis 命令
```

# 四、压缩与解压

## 文件类型

```
.zip .gz .bz2 .tar.gz .tar.bz2
```

1. zip -r 压缩文件名 源目录
2. unzip 压缩文件

3. gzip 文件名（压缩后删除原始文件） -c 保留原始文件 -r 压缩目录下文件
4. gunzip 文件名 解压
5. bzip2 源文件 （压缩后不保留原始文件） -k 压缩后保存原始文件 -d 解压
6. bunzip 解压

#### 7. tar 命令

参数：

```
-x 解包  
-c 打包  
-v 显示过程  
-f 指定文档名  
-z 压缩  
-t 查看内容
```

例子：

```
tar -cvf 文档名.tar 源文件  
-cvf 打包不压缩  
-zcvf 压缩为.tar.gz  
-zxvf 解压.tar.gz  
-jcvf 压缩为.tar.bz2  
-jxvf 解压.tar.bz2
```

## 五、关机，后台执行

### 1. 关机、重启

```
shutdown -c 取消前一条关机命令  
-r 重启  
-h 关机
```

### 2. 后台执行

```
nohup 命令 &  
jobs #查看后台运行程序  
fg job_id #后台程序调到前台  
bg job_id #后台
```

## PART2 shell基础

定义: shell是命令行解释器, 连接人机沟通

## 一、 输出命令

### 1.echo [选项] 输出内容

-e 输出特殊字符

特殊字符(需要加引号):

\b 退格

\t 制表符

\n 换行

\e[背景色;前景色m 文本 \e[0m

echo -e "\e[41;37m hello \e[0m"

## 二、 Shell脚本格式

```
#!/bin/bash #声名
```

```
# ‘#’是注释符号
```

```
#print word
```

```
echo "firest program"
```

## 三、 Shell执行

方法1

```
chmod 755 first.bash
```

```
./first.bash
```

方法2

```
sh ./first.bash
```

## 四、 别名的赋予,历史命令查看

### 1.alias 别名='原名'

存储于环境变量配置文件 ~/.bashrc

### 2. history

历史命令查看

-c清空历史命令

-w保存缓存中的历史命令到 `.bash_history`

## 五、输出重定向

命令 > 文件 将屏幕输出的正确运行内容输出到文件（覆盖原始文件内容）  
命令 >> 文件 将屏幕输出的正确运行内容输出到文件（追加到原始文件内容）  
命令 2>> 文件 将屏幕报错内容输入到文件  
命令 &>> 文件 正确错误内容全部追加到文件中  
命令 >> 文件 2>&1 正确错误内容全部追加到文件中  
命令 >> /dev/null 将内容丢入回收站

## 六、字符统计

wc 参数 文件名  
-c统计字节数  
-w统计单词数  
-l 统计行数

## 七、命令连接符

命令1; 命令2 连接命令，两条命令没有逻辑关系  
命令1&&命令2 与 只有命令1正确执行才会执行命令2  
命令1||命令2 或 只有命令1执行错误才执行命令2；命令1正确执行，不执行命令2  
命令1 | 命令2 管道符 命令1的输出作为命令2的输入

## 八、通配符

\* 匹配任意字符  
? 匹配一个字符  
[ ] 匹配括号中的内容（[^]反选）

## 九、其他符号

'': 内特殊符号无含义  
"": 中除了\$, \, ' 外特殊符号无含义  
`命令` \$(命令): 先执行（）内内容  
#: shell中注释标识  
\$ 调用变量  
\ 转义字符

## PART3 变量

### 一、自定义变量（局部变量）

#### 1.定义

```
变量名=变量值
```

#### 2.调用

```
$变量
```

#### 3.变量叠加

```
例 x=123 x="$x"123 x=${x}123
```

#### 4.变量查看

```
set
```

#### 5.变量删除

```
unset 变量名
```

### 二、环境变量（全局变量）

定义 `export 变量名=变量值` `export 变量名`

显示环境变量 `env`

### 三、位置参数

```
$n 命令变量 $0为命令本身 $1.....为对应参数 ${10}  
$* 所有参数输出（一个字符串）  
$@ 分别列出参数  
$# 参数个数
```

### 四、预定义变量



```
$? 正确执行为0，不正确执行为非0数
$$ 当前PID
$! 最后一个后台进程PID
```

## 五、输入指令

read 参数: -p “提示信息” #赋值给REPLY -t 限制时间 -s 不显示输入 -n 指定参数长度

## PART4 正则表达式

目的: 处理文件中的字符串

### 一、基础的正则表达式

*	符号前的字母重复n任意次则匹配
.	匹配任何一个字符
^	代表行首
\$	代表行尾
^\$	匹配空白行
[ ]	与通配符作用相同

### 二、字符处理

#### 1.cut

```
-f 提取列数
-d 指定分隔符
```

#### 2.awk

```
awk “条件1{动作1}条件2{动作2}”
-F “,” 指定分隔
cat temp.csv | awk -F "," '{print $1 "," $2}'
cat temp.csv | awk -F "," '$1==168085761{print $1 "," $2}'
```

#### 3.sed

选项 -n 被操作的行显示 -e 允许输入多条动作 (-e间隔) cat temp.csv | sed -e '1,5s/110/bbb/' -e '7,9s/110/aaa/' -i 直接修改文件内容 动作 a 追加到下一行 c 行替换 cat temp.csv | sed '1caaa'

```

i 插入到上一行
cat temp.csv | sed '1iaaa'

d 删除
sed '2,$d' example （删除第二行以后所有）

p 打印
cat temp.csv | sed -n '1,2p'

s 字符串替换
替换行数/替换的目标字符串/替换的字符串/g g代表替换该行所有符合条件内容
cat temp.csv | sed -e '1,5s/1/bbb/g' -e '7,9s/1/aaa/'

```

## Part5 计划任务与系统管理

### 1.计划任务

分	小时	日	月	星期	执行命令
0-59	0-23	1-31	1-12	0-6	command (0表示周日)

指令 查看crontab -l 修改crontab -e

特殊符号的含义: "\*"代表取值范围内的数字, "/"代表"每", "-"代表从某个数字到某个数字, ","分开几个离散的数字

计划任务写法样例:

```
* * * * * date &>>/home/test.log
```

每月10号及20号的3: 30执行ls命令[注: ", "用来连接多个不连续的时段]

```
30 3 10,20 * * ls /
```

每天8-11点的第25分钟执行ls命令[注: "-"用来连接连续的时段]

```
25 8-11 * * * ls /
```

每15分钟执行一次ls命令 [即每个小时的第0 15 30 45 60分钟执行ls命令 ]

```
*/15 * * * * ls /
```