

# Linux 常见操作总结

-- by 张太瑞

/\*\*\*\*\*

root:管理员()当命令文件存放于以下两个文件夹 则该命令只有管理员能执行

/sbin

/usr/sbin

**注:如 sbin s 开头的命令一般只有超级管理员 root 有操作权限**

all users:所有用户()以下 2 个文件夹存放的命令文件所有用户都可以执行

/bin

/usr/bin

**注:如 bin 没有 s 开头的命令一般所有用户都有操作权限;这里单指系统命令,不包含其他用户创建文件;**

缩写:bin 原意:binary 描述:二进制文件

缩写:usr 原意:user 描述: 用户

缩写:sbin-super binary 含义描述:超级二进制文件

\*\*\*\*\*/

## 权限

```
drwxr-xr-x  2      root    root      4096      12-01 20:52      bin
```

硬链接数或目录子目录数   所有者   所属组   文件大小   文件创建或最后修改的时间   文件

**drwxr-xr-x** 中的 **d** 表示文件类型 其中常见文件类型三种 **d** 、 **-** 、 **l**;

**d** 表示目录 directory

**-** 表示二进制文件

**l** 表示软连接文件 link

**drwxr-xr-x** 中的 **rwXr-xr-x** 表示文件权限,含义如下:

r-read 读	w-write 写	x-execute 执行
rwX	r-x	r-x
所有者(user or onwer)	所属组 g(group)	其他人 o(others)

注\*:linux 中表示文件大小 用数据单位:数据块(最小存贮单位) block 512;

## Linux 文件命名注意事项

- 1.文件命名规则 除了/之外,所有的字符合法; [注解:文件命名式,不要以/命名,/表示根目录]
- 2.有些字符最好不用,如空格符,制表符; [注解:linux 命名与参数之间以空格隔开,使用空格容易产生歧义,linux 无法识别]
- 3.便面使用.作为普通文件名的第一个字符;[注解: .和..为隐藏文件,以.或..开始命名表示隐藏文件;]
- 4.大小写敏感

## 命令格式

命令格式: 命令 -选项 参数

示例: `ls -la /etc`

说明:

当有多个选项的时候, 可以写在一起.

两个特殊的目录 `.` 和 `..` 分别表示当前目录和当前目录的父目录 示例: `./` `../`

## 常见命令--文件

{1} 命令名称: `ls`

命令英文原意: `list`

选项(属性): {

`-a` `all` 显示所有文件, 包括隐藏文件

`-l` `long` 详细信息显示

`-d` 查看目录属性

`-i` `i-inode` `i` 节点 数字标识 每个文件必须有一个 `i` 节点

`-ld` 查看目录的权限

}

功能描述: 显示目录文件

执行权限: 所有用户

{2} 命令名称: `cd`

命令英文原意: `change directory`

执行权限: 所有用户

语法: `cd [目录]`, 用户切换目录

示例: `cd /` 切换到根目录

`cd ..` 回到上一级目录

{3} 命令名称: `pwd`

命令英文原意: `print working directory`

执行权限: 所有用户

语法: `pwd`

功能描述: 显示当前所在的工作目录

{4} 命令名称: `touch`

执行权限: 所有用户

语法: `touch 文件名`

功能描述: 创建空文件

示例: `touch newfile`

{5} 命令名称: `mkdir`

英文原意: **make** **directory**

选项{

**-p** **--parents** 递归创建

**-m** **--mode=模式**创建目录的同时也指定目录的权限

}

执行权限:所有用户

语法:mkdir 目录名称

功能描述:创建新目录

示例:mkdir newdir 创建一个名为 newdir 的文件

mkdir -p a/b/c 递归创建 c 文件

mkdir -m 777 test 创建目录 test 指定权限为 777

{6} 命令名称:cp

英文原意:**copy**

选项{

**-R** 复制目录

**-p** 保留时间值 不改变时间值

}

执行权限:所有用户

语法:cp -R [源文件或目录][目的目录]

**-R** 赋值目录

功能描述:复制文件或者目录

示例:.

cp file1 file2 dir1 将文件 file1 file2 复制到目录 dir1

cp -R dir1 dir2 将 dir1 下的所有文件及子目录复制到 dir2

{7} 命令名称:mv

命令英文原意:**move**

执行权限:所有所用

语法:mv [源文件或目录][目的目录]

功能描述:移动文件, 更名

示例:mv file1 file3 将当前目录下文件 file1 改名为 file3

mv file2 dir2 将文件 file2 移动到目录 dir2 下

可以组合使用, 在移动的同时并改变文件的名称

mv /test/test.php /tangqi/testOne.php

{8} 命令名称:rm

选项{

**-r** 删除目录

}

命令英文原意:**remove**

执行权限:所有用户

语法:rm -r[文件或目录]

功能描述:删除文件

扩展:rmdir 删除空目录

rm -rf [目录名] 强制删除 屏蔽询问确认语句,直接删除要操作的目录

示例:rm file3[文件名]

删除文件 file3

rm -r dir1[目录名]

删除目录 dir1

{9} 命令名称:cat

命令英文原意:con**cat**enate and display files

执行权限:所有用户

语法:cat[文件名]

功能描述:显示文件内容

示例:cat /d/php1/ThinkPHP4/Admin/Tpl/Index/index.html

{10} 命令名称:more

执行权限:所有用户

语法:more[文件名]

(空格)或 f 显示下一页

(Enter) 显示下一行

Q 或 q 退出

功能描述:分页显示文件内容

示例:more 文件名或者路径

{11} 命令名称:head

执行权限:所有用户

语法:head -num[文件名]

-num 显示文件的前 num 行

功能描述:查看文件的前几行

示例:head -20 文件名

{12} 命令名称:tail

执行权限:所有用户

语法与 head 相似

-f 动态显示文件末尾的内容

功能描述:查看文件的后几行

{13} 命令名称:ln

命令英文原意:**l**ink

执行权限:所有用户

语法:ln -s[源文件][目标文件]

-s 创建软连接

ln [源文件][目标文件] 硬连接与源文件有相同的 i 节点  
功能描述:产生链接文件  
软链接文件的权限:rwxrwxrwx -->指向到源文件 类似于 windows 中桌面快捷方式  
硬链接: 类似于复制 但是硬链接文件与源文件同步更新 硬链接不能跨文件系  
统

## 常见命令—用户操作

su -[用户名] 切换用户  
useradd [用户名] linux 下添加用户

## 常见命令--权限处理:

{14} 命令名称: chmod  
命令英文原意:change the permissions mode of a file  
执行权限:所有用户  
语法:chmod u-所有者 +r  
          g-所属组 -w  
          o-其他人 =x  
rwx 可读可写可执行 chmod u+wx chmod 0-rx chmod g=rwx  
功能描述:改变文件或者目录的权限  
r-4 w-2 x-1  
示例: rwxr-xr-- 754  
      rw-r-x--x 651  
      752 rwxr-x-w  
语法格式:chmod 权限[777] 文件名

文件目录权限总结:

代表字符	权限	对文件的含义	对目录的含义
r	读权限	可以查看文件的内容	可以列出目录中的内容
w	写权限	可以修改文件的内容	可以在目录中创建,删除文件
x	执行权限	可以执行文件	可以进入目录

### 文件

r - cat, more, hrad, tail  
w - echo, vi  
x - 命令, 脚本

### 目录

r - ls

w - 删除一个文件前提是必须有该文件所在目录的写权限-touch, mkdir, rm

x - 可以进入目录 cd

{15} 命令名称: chown

命令英文原意: **ch**ange file **ow**nership

执行权限: 所有用户

语法: chown[用户][文件或目录] : 用户必须为系统存在的用户

功能描述: 改变文件或目录的所有者

示例: chown nobody file 改变文件 file 的所有者为 nobody

{16} 命令名称: chgrp

命令英文原意: **ch**ange file **gr**oup ownership

执行权限: 所有用户

语法: chgrp[用户组][文件或目录]

功能描述: 改变文件或目录的所属组

示例: chgrp adm file 改变文件 file 的所属组为 adm

umask -S 查看文件或目录的系统默认权限

umask 0000

0 - 特殊权限位

000 - 用户权限位, 权限掩码值 777-000

linux 默认权限规则

不能授予可执行 x 权限

更改系统默认权限

umask 777-750=027

umask 027[掩码值]

## 常见命令--文件搜索

{17} 命令名称: which/where

执行权限: 所有用户

语法: which/where[命令名称]

功能描述: 显示系统命令所在目录

示例: which ls/where ls

{18} 命令名称: find

执行权限: 所有用户

语法: find[所搜路径][搜寻关键字]

功能描述: 查找文件或目录

## 缩小查找范围, 尽量不要在跟目录下查找

选项:find /etc

1, 根据文件名查找

-name init 在目录/etc 中查找文件 init

\* 匹配任意字符 init\* 查找所有以 init 开头的文件

? 匹配单个字符 init???

example:find /d/php1 -name init\*

find /d/php1 -name \*init\*

find /d/php1 -name init???

2, 根据文件大小查找

-size 文件大小 block 数据块 512 字节

大于+ 小于-

example:find /d/php1 -size +204800

find /d/php1 -size -102400

find /d/php1 -size +102400 -a -size -204800

3 根据文件所有者查找

-user [用户名]

example:find /d/php1 -user 用户名

4 根据时间查找

(1) 以天为单位 ctime atime mtime

(2) 以分钟为单位 cmin amin mmin

c- change 改变 表示文件属性被修改过, 所有者, 所属组, 权限

a-access 访问

m-modify 修改 表示文件的类容被修改过

-表示多长时间之内, +表示超过多少时间

example:find /d/php1 -mmin -120 查找 2 个小时之内被修改过

的文件

-type 文件类型 f

软连接文件 l

目录 d

1, 连接符 -a and 逻辑与 -o or 逻辑或

example:find /d/php1 -name init\* -a -size +102400

find /d/php1 -name init\* -a -type l

2, 连接执行符

-exec 命令 {} \;

花括号 {} 表示 find 查询的结果 \转义符 使符号命令使用本身

的含义

example:find /d/php1 -name init??? exec ls -l {} \;

## 常见命令—压缩解压

(19) 压缩解压命令:

命令名称: gzip

命令英文原意: gun zip



执行权限: 所有用户

语法: gzip 选项 [文件]

功能描述: 压缩文件

压缩后的文件格式: .gz

gzip 文件名 只能压缩文件 不能压缩目录 不保留原文件

示例: gzip test.php 生成压缩文件 test.php.gz 不保留原文件

(20) 压缩解压命令: gunzip

命令英文原意: gun unzip

执行权限: 所有用户

语法: gunzip 选项 [文件名]

功能描述: 解压缩 .gz 的压缩文件

示例: gunzip test.php.gz

gzip -d test.php.gz

(21) 压缩解压命令: 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称

-z 打包同时压缩 或 解压缩

-x 解包 .tar 文件

功能描述: 打包目录

压缩后的文件格式: .tar.gz

(23) file [文件名] 判断文件类型

(24) 命令名称: bzip2

执行权限: 所有用户

语法:

bzip2 选项[-k] 文件

不加 选项 -k 只能 压缩文件 不保留原来的文件

解压相同

压缩后的文件格式: .bz2

## 常见命令—网络

(25) 网络通信指令:

指令名称: write

执行权限: 所有用户

语法:write 用户名

功能描述:向另外一个用户发信息,以 Ctrl+D 为结束

(26)

指令名称:wall

执行权限:所有用户

语法:wall [信息]

功能描述:向所有用户发送信息

(27)

命令名称:ping

执行权限:root

语法:ping ip 地址

功能描述:测试网络连通性

示例: ping 192.168.1.1          以 Ctrl+C 结束

ping -c 3 ip 地址    定义 ping 多少次

ping -s 60000 定义 发送包的大小          icmp

(28)

命令名称:ifconfig

执行权限:root

语法:

ifconfig 选项[-a]显示所有网卡信息

功能描述:查看网络设置信息

示例:ifconfig -a

更改 ip 地址

## 常见命令—系统操作

(29)

命令名称:shutdown

执行权限:root

语法:shutdown

功能描述:关机

示例: shutdown -h now

(30)

命令所在:reboot

执行权限:root

功能描述:重启系统

## Shell 应用技巧

<Tab> 命令补全 文件名称

clear 清屏 ctrl+l

ctrl +u 删除当前命令行的字符

上下左右键 命令历史记录

定义命令别名

alias copy=cp

alias xr="rm -r"

查看别名信息: alias

删除别名命令 unalias 别名

## 文本编辑器:Vim/Vi

### 插入命令

a 在光标后附加文本

A 在本行行末附加文本

i 在光标前插入文本

I 在本行开始插入文本

o 在光标下插入新行

O 在光标上插入新行

### 定位命令:

h 方向左键

j 方向下键

k	方向上键
i	方向右键
\$	移至行尾
0	移至行首
H	移至屏幕上端
M	移动至屏幕中央
L	移动至屏幕下端
:set nu	设置行号
:set nonu	取消行号
gg	到第一行
G	到最后一行 i
nG	到第 n 行 8G
:n	到第 n 行 :8

## 删除命令:

x	删除光标所在处的字符
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
dG	删除光标所在行到末尾的内容
D	删除光标所在处到行尾
:n1, n2d	删除指定范围的行

## 复制和剪切命令:

yy Y 复制当前行

nyy nY 复制当前行一下 n 行

dd 剪切当前行

ndd 剪切当前行一下 n 行

p(下) P(上) 粘贴在当前光标所在行下火行上

## 替换取消命令

r 替换当前字符

R 替换当前字符及其后的字符，直至按 ESC 键

u 取消上一步操作

## 搜索替换命令:

/string

向前搜索指定字符串搜索时忽略大小写:set ic

n 指定搜索 字符串的下一个出现位置

:%/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

:wq=ZZ 等效

:wq! 强行保存退出 只有文件的所有者 或者是管理员 root

:q! 不保存退出

:w 保存

:r !data 把当前系统时间导入光标所在位置 把命令的执行结果导入文件

:map 要定义的快捷键(ctrl+v+p) 要触发的命令

:map ^p I#<ESC>

`/ .vimrc

## linux 引导流程解析

加载固件 firmware (CMOS/BIOS) -> POST 加电自检

自举程序 BootLoader (GRUB) -> 载入内核

载入内核 Kernel

启动进程 init

读取执行配置文件/etc/inittab

固件设置

hwclock 固件时间 hwclock --help 查看选项

date linux 软件时间命令

date -s "2015-10-14 10:46:20"

同步软件时间 和固件时间

hwclock --hctosys 以硬件时间为标准修改软件时间

    --systohc 按照当前的系统时间(软件时间)修改硬件时间

    --set 修改硬件时间(自定义)

hwclock --set --date="9/22/96 16:45:05" 修改硬件时间

date 121511302010.38 修改软件时间

/etc/grub.conf 配置文件

kernel(内核:linux 内核存放位置, 压缩文件)/vmlinuz-2.6.18.194.el5 ro  
root=LABEL=/ rhgb quiet 载入内核

2            主版本号  
6            次版本号 如果为奇数 则表示该内核为测试版 偶数则为正式版  
18           末版本号

关于内核:

驱动 引进

启动进程 init(inittab:初始化) PID 恒为 1 ps -le | more init

是所有进程的父进程 当一个进程的父进程死亡 系统将自动将该进程的父进程只  
init

ps -le 查看进程

内核调度器 PID 为 0 获得 CPU 时间

| 管道 将一个命令的输出作为另一个命令的输入

init 配置文件:etc/inittab

## linux 运行级别

0.halt                    关机  
1.Single user mode    单用户模式 类似 windows 的安全模式 只有 root 可以登录  
2.Multiuser without NFS 类似 win 下的网上邻居  
3.Full multiuser mode  
4.unused                没有在使用 用户可以自定义  
5.X11                    系统缺省的, 默认的运行级别  
6.redboot               重启

查看当前的运行级别

runlevel

grep -v "^#" /etc/inittab | more

inittab 文件剖析

id:runlevels:action:process

id: 标示符, 一般为两位数字或者是字母

runlevels: 指定运行级别(数字对应), 如果为空 则所有运行级别都执行

action:指定运行状态

process:指定运行的脚本或者命令

etc/rc.d/rc3.d

S -start

K-kill

数字表示启动顺序

## linux 软件包管理

二进制软件包管理(RPM YUM)

源代码包安装

脚本安装(Shell 或 java 脚本)

sudo-1.7.2p1-5.e15.i386.rpm

软件名(sudo), 版本号(1.7.2p1), 发行号(5.e15), 硬件平台(i386)

(卸载) `rpm -e sudo`(软件名称)

注意:如果其他软件包有依赖关系, 卸载时产生提示信息, 可使用 `--nodeps`

安装 RPM 软件包

(安装)

`rpm -ivh sudo-1.7.2p1-5.e15.i386.rpm` (软件包名称)

其他安装选项

`-- excludedocs`(不安装软件包中的文档文件)



-- prefix PATH (将软件包安装到由 PATH 指定的路径下)  
-- test (只对安装进行测试, 并不实际安装)

--replacepkgs (覆盖安装)  
--replacefiles (替换安装)  
-nodeps (强制安装)

(升级)

rpm -Uvh sudo-1.8.2p1-5.el5.i386.rpm

(查询)

rpm -q sudo (查看软件包是否已经安装 +软件名称)

rpm -qa | grep samba (软件包名称关键字)

-a 查询所有已安装的软件包  
-f 查询文件所属软件包  
-p 查询软件包  
-i 显示软件包信息  
-l 显示软件包中的文件列表  
-d 显示被标注为文档的文件列表  
-c 显示被标注为配置文件得的文件列表

实例

rpm -qf 查询文件隶属的软件包  
rpm -qi (已安装)  
rpm -qip (未安装) 查询软件包信息  
rpm -ql (已安装)  
rpm -qlp (未安装) 查看软件包安装文件  
rpm -qd 查询软件包帮助文档  
rpm -qc 查看软件包配置文件

挂载光盘 mount /dev/cdrom/mnt/cdrom

## YUM 包管理

引用 yum 的好处

自动解决软件包依赖关系

方便软件包升级

安装	yum install sudo-1.7.2p1-5.el5.i386.rpm
检测升级	yum check-update sudo-1.7.2p1-5.el5.i386.rpm
升级	yum update sudo
软件包查询	yum list
软件包信息	yum info

卸载	<code>yum remove</code>
帮助	<code>tum -help man yum</code>

## linux 安装 ftp

```
yum install vsftpd
```

```
ftp 开启/停止/重启 /sbin/service vsftpd start/stop/restart
```

```
设置开机启动 vsftpd ftp 服务 chkconfig vsftpd on
```

## 什么是 vsftpd

vsftpd 是一款在 Linux 发行版中最受推崇的 FTP 服务器程序。特点是小巧轻快，安全易用。

vsftpd 的名字代表”very secure FTP daemon”，安全是它的开发者 Chris Evans 考虑的首要问题之一。在这个 FTP 服务器设计开发的最开始的时候，高安全性就是一个目标。

安装 vsftpd

1、以管理员（root）身份执行以下命令

```
yum install vsftpd
```

2、设置开机启动 vsftpd ftp 服务

```
chkconfig vsftpd on
```

3、启动 vsftpd 服务

```
service vsftpd start
```

管理 vsftpd 相关命令：

有一台 CentOS 的服务器，原来建的一个用于 ftp 的用户帐号，制定一个特定的目录，现在需要修改这个用户的 home\_dir。

只需要使用 `usermod -d /new/home_dir -U username` 命令就可以更改了。

下面补充一下 usermod 更多功能：

usermod 不仅能改用户的 SHELL 类型，所归属的用户组，也能改用户密码的有效期，还能改登录名。usermod 如此看来就是能做到用户帐号大转移；比如我把用户 A 改为新用户 B；

```
usermod [-u uid [-o]] [-g group] [-G group,...]  
        [-d 主目录 [-m]] [-s shell] [-c 注释] [-l 新名称]  
        [-f 失效日] [-e 过期日] [-p 密码] [-L|-U] 用户名
```

usermod 命令会参照你命令列上指定的部份修改系统帐号档。下列为 usermod 可选用的参数。

-c comment 更新用户帐号 password 档中的注解栏，一般是使用 chfn(1)来修改。

-d home\_dir 更新用户新的登入目录。如果给定-m 选项，用户旧目录会搬到新的目录去，如旧目录不存在则建个新的。

-e expire\_date 加上用户帐号停止日期。日期格式为 MM/DD/YY。

-f inactive\_days 帐号过期几日后永久停权。当值为 0 时帐号则立刻被停权。而当值为-1 时则关闭此功能。预设值为-1。

-g initial\_group 更新用户新的起始登入用户组。用户组名须已存在。用户组 ID 必须参照既有的用户组。用户组 ID 预设值为 1。

-G group, [...] 定义用户为一堆 groups 的成员。每个用户组使用", "区格开来，不可以夹杂空白字元。用户组名同-g 选项的限制。如果用户现在的用户组不再此列，则将用户由该用户组中移除。

-l login\_name 变更用户 login 时的名称为 login\_name。其它不变。特别是，用户目录名应该也会跟着更动成新的登入名。

-s shell 指定新登入 shell。如此栏留白，系统将选用系统预设 shell。

-u uid 用户 ID 值。必须为唯一的 ID 值，除非用-o 选项。数字不可为负值。预设值为最小不得小于/etc/login.defs 中定义的 UID\_MIN 值。0 到 UID\_MIN 值之间是传统上保留给系统帐号使用。用户目录树下所有的档案目录其 userID 会自动改变。放在用户目录外的档案则要自行手动更动。

**警告:**usermod 不允许你改变正在线上的用户帐号名称。当 usermod 用来改变 userID，必须确认这名 user 没在电脑上执行任何程序。你需手动 更改用户的 crontab 档。也需手动更改用户的 at 工作档。采用 NISserver 须在 server 上更动相关的 NIS 设定。

举个简单的例子，我们在前面说了关于 useradd 的工具，而 usermod 工具和 useradd 的参数差不多；两者不同之处在于 useradd 是添加，usermod 是修改。

停止 vsftpd: `service vsftpd stop`

重启 vsftpd: `service vsftpd restart`

## 配置防火墙

打开/etc/sysconfig/iptables 文件

`vi /etc/sysconfig/iptables`

在 REJECT 行之前添加如下代码

```
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 21 -j ACCEPT
```

保存和关闭文件，重启防火墙

`service iptables start`

配置 vsftpd 服务器

默认的配置文件是/etc/vsftpd/vsftpd.conf，你可以用文本编辑器打开。

`vi /etc/vsftpd/vsftpd.conf`

## 添加 ftp 用户

下面是添加 ftpuser 用户，设置根目录为/home/wwwroot/ftpuser, 禁止此用户登录 SSH 的权限，并限制其访问其它目录。

1、修改/etc/vsftpd/vsftpd.conf

将底下三行

```
#chroot_list_enable=YES
```

```
# (default follows)
```

```
#chroot_list_file=/etc/vsftpd.chroot_list
```

改为

```
chroot_list_enable=YES
```

```
# (default follows)
```

```
chroot_list_file=/etc/vsftpd/chroot_list
```

3、增加用户 ftpuser，指向目录/home/wwwroot/ftpuser, 禁止登录 SSH 权限。

```
useradd -d /home/wwwroot/ftpuser -g ftp -s /sbin/nologin ftpuser
```

4、设置用户口令

```
passwd ftpuser
```

5、编辑文件 chroot\_list:

```
vi /etc/vsftpd/chroot_list
```

内容为 ftp 用户名, 每个用户占一行, 如:

```
peter
```

```
john
```

## 6、重新启动 vsftpd

安装 sftp 服务软件: `yum -y install vsftpd`

增加组: `groupadd www`

增加用户: `useradd -g www -d /data0/htdocs/www -s /bin/false -M user_name`

?

1

2

3

4

参数含义 -g 指定用户所属的组

-d 指定用户的目录

-s 禁用用某项服务

-M 指定用户名

/bin/false 和/sbin/nologin 的差别:

?

1

2

3

4

5

6

/bin/false 是最严格的禁止 login 选项, 一切服务都不能用。

/sbin/nologin 只是不允许 login 系统

如果想要用 false 在禁止 login 的同时允许 ftp, 则必须在/etc/shells 里增加一行  
/bin/false,

如果要限制用户在主目录, 则需作如下设置:

1、修改/etc/vsftpd/vsftpd.conf

将底下三行

```
#chroot_list_enable=YES
```

```
# (default follows)
```

```
#chroot_list_file=/etc/vsftpd.chroot_list
```

改为

```
chroot_list_enable=YES
```

```
# (default follows)
```

```
chroot_list_file=/etc/vsftpd/chroot_list
```

2、编辑文件: /etc/vsftpd/

```
chroot_list
```

内容为 ftp 用户名, 每个用户占一行, 如:

```
?
```

```
1
```

```
2
```

```
www
```

```
john
```

3、重新启动 vsftpd

```
[root@nginx conf]# /etc/init.d/vsftpd restart
```

## linux 用户管理

/etc/passwd 文件位置 man 5 passwd

文件配置

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

用户名:密码位:用户的 UID(用户的标示符):缺省组标示符(GID 组 ID):注释性描述(示例:存放用户全名等信息):宿主目录(用户登陆系统后的缺省目录):命令解释器(用户使用的 shell, 默认为 bash)

root: 表示用户名(不要太长 不要超过 8 位字母)

x : 密码位

linux 用户分为 3 种:

1. 超级用户 示例::root UID 为 0

2. 普通用户 UID500--60000

3. 伪用户 UID1-499

伪用户 与系统和程序服务相关 与 linux 系统进程相关 不需要登陆无法登陆 可以没有宿主目录

用户组

每个用户都至少属于一个用户组

每个用户组可以包括多个用户

同一用户组的用户享有该组共有权限

用户信息文件

etc/shadow 文件位置

```
root:$6$T6OnQVvJ$1HWIG4GLI4cL5zIH9ZhXEsp12yKsPj7R.K6S4sLRUIjUWUw.WkHexERfD1
PUHTToRlXKyhdY8opn37XoSxJxG.:16324:0:99999:7:::
```

用户名:加密密码:最后一次修改时间:最小时间间隔(两次修改密码之间的最小天数):  
最大时间间隔(密码保持有效的最多天数)

## ---小技巧---

### 1. 查找系统中所有的空文件

```
find / -size 0 -type f > /var/zerofile.log
```

### 2. df

-i 查看 inode 的详细信息

-Th 所有分区详情

使用 `ls -li` 命令, 可以看到文件名对应的 inode 号码:

```
ls -li example.txt
```

在一台配置较低的 Linux 服务器（内存、硬盘比较小）的/data 分区内创建文件时，系统提示磁盘空间不足，用 `df -h` 命令查看了一下磁盘使用情况，发现/data 分区只使用了 66%，还有 12G 的剩余空间，按理说不会出现这种问题。后来用 `df -li` 查看了一下/data 分区的索引节点(inode)，发现已经用满(IUsed=100%)，导致系统无法创建新目录和文件。

查找原因：

/data/cache 目录中存在数量非常多的小字节缓存文件，占用的 Block 不多，但是占用了大量的 inode。

解决方案：

- 1、删除/data/cache 目录中的部分文件，释放出/data 分区的一部分 inode。
- 2、用软连接将空闲分区/opt 中的 newcache 目录连接到/data/cache，使用/opt 分区的 inode 来缓解/data 分区 inode 不足的问题：

```
ln -s /opt/newcache /data/cache
```

## 环境变量

1

“/etc/profile” 此文件为系统的每个用户设置环境信息，当用户第一次登录时，该文件被执行。并从“/etc/profile.d”目录的配置文件中搜集 shell 的设置。所以如果你有对/etc/profile 有修改的话必须得重启你的修改才会生效，此修改对每个用户都生效。

2

“/etc/bashrc” 为每一个运行 bash shell 的用户执行此文件。当 bash shell 被打开时，该文件被读取。如果你想对所有的使用 bash 的用户修改某个配置并在以后打开的 bash 都生效的话可以修改这个文件，修改这个文件不用重启，重新打开一个 bash 即可生效。

3

“~/.bash\_profile” 每个用户都可使用该文件输入专用于自己使用的 shell 信息，当用户登录时，该文件仅仅执行一次！默认情况下，它设置一些环境变量，执行用户的 .bashrc 文件。此文件类似于/etc/profile，也是需要重启才会生效，/etc/profile 对所有用户生效，~/.bash\_profile 只对当前用户生效。

4

“~/.bashrc” 该文件包含专用于你的 bash shell 的 bash 信息，当登录时以及每次打开新的 shell 时，该文件被读取。（每个用户都有一个 .bashrc 文件，在用户目录下）。此文件类似于/etc/bashrc，不需要重启生效，重新打开一个 bash 即可生效，/etc/bashrc 对所有用户新打开的 bash 都生效，但~/.bashrc 只对当前用户新打开的 bash 生效。

5

“~/.bash\_logout” 当每次退出系统（退出 bash shell）时，执行该文件。

在文件末尾加上如下两行代码

```
PATH=/usr/local/webserver/php/bin:$PATH
export PATH
```

要是刚才的修改马上生效，需要执行以下代码

```
[root@CentOS ~]# source /etc/profile
```

```
echo $PATH
```



## 查看 linux 系统信息

输入“`uname -a`”,可显示电脑以及操作系统的相关信息。

输入“`cat /proc/version`”,说明正在运行的内核版本。

输入“`cat /etc/issue`”,显示的是发行版本信息

`lsb_release -a` (适用于所有的 linux, 包括 Redhat、SuSE、Debian 等发行版, 但是在 debian 下要安装 lsb)

查看 linux CPU 信息

查看物理 CPU 的个数

```
cat /proc/cpuinfo | grep 'physical id' | sort | uniq | wc -l
```

查看逻辑 CPU 的个数

```
cat /proc/cpuinfo | grep 'processor' | wc -l
```

查看 CPU 是几核

```
cat /proc/cpuinfo | grep 'cores' | uniq
```

## Linux 系统用户管理

用户账号的管理工作主要涉及到用户账号的添加、修改和删除。

添加用户账号就是在系统中创建一个新账号,然后为新账号分配用户号、用户组、主目录和登录 Shell 等资源。刚添加的账号是被锁定的,无法使用。

### 添加新的用户

代码:

```
useradd 选项 用户名
```

选项含义:

代码:

`-c comment` 指定一段注释性描述。

`-d 目录` 指定用户主目录,如果此目录不存在,则同时使用`-m`选项,可以创建主目录。

`-g 用户组` 指定用户所属的用户组。

-G 用户组, 用户组 指定用户所属的附加组。  
-s Shell 文件 指定用户的登录 Shell。  
-u 用户号 指定用户的用户号, 如果同时有-o 选项, 则可以重复使用其他用户的标识号。

用户名 指定新账号的登录名。

示例 1.:

代码:

```
# useradd -d /usr/sam -m sam
```

此命令创建了一个用户 sam,

其中-d 和-m 选项用来为登录名 sam 产生一个主目录/usr/sam (/usr 为默认的用户主目录所在的父目录)。

示例 2:

代码:

```
# useradd -s /bin/sh -g group -G adm,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 group 用户组, 同时又属于 adm 和 root 用户组, 其中 group 用户组是其主组。

这里可能新建组: #groupadd group 及 groupadd adm

增加用户账号就是在/etc/passwd 文件中为新用户增加一条记录, 同时更新其他系统文件如/etc/shadow, /etc/group 等。

Linux 提供了集成的系统管理工具 userconf, 它可以用来对用户账号进行统一管理。

## 删除用户

如果一个用户的账号不再使用, 可以从系统中删除。删除用户账号就是要将/etc/passwd 等系统文件中的该用户记录删除, 必要时还删除用户的主目录。删除一个已有的用户账号使用 userdel 命令, 其格式如下:

代码:

```
userdel 选项 用户名
```

常用的选项是-r, 它的作用是把用户的主目录一起删除。

示例:

代码:

```
# userdel sam
```

此命令删除用户 sam 在系统文件中（主要是/etc/passwd, /etc/shadow, /etc/group 等）的记录，同时删除用户的主目录。

## 修改用户

修改用户账号就是根据实际情况更改用户的有关属性，如用户号、主目录、用户组、登录 Shell 等。

修改已有用户的信息使用 usermod 命令，其格式如下：

代码：

```
usermod 选项 用户名
```

常用的选项包括-c, -d, -m, -g, -G, -s, -u 以及-o 等，这些选项的意义与 useradd 命令中的选项一样，可以为用户指定新的资源值。另外，有些系统可以使用如下选项：

代码：

```
-l 新用户名
```

这个选项指定一个新的账号，即将原来的用户名改为新的用户名。

示例：

代码：

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

此命令将用户 sam 的登录 Shell 修改为 ksh，主目录改为/home/z，用户组改为 developer。

## 用户口令的管理

用户管理的一项重要内容是用户口令的管理。用户账号刚创建时没有口令，但是被系统锁定，无法使用，必须为其指定口令后才可以使用，即使是指定空口令。指定和修改用户口令的 Shell 命令是 passwd。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。命令的格式为：

代码：

```
passwd 选项 用户名
```

可使用的选项：

```
-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。
```

如果默认用户名，则修改当前用户的口令。

示例:，假设当前用户是 sam，则下面的命令修改该用户自己的口令：

代码：

```
$ passwd
Old password:*****
New password:*****
Re-enter new password:*****
```

如果是超级用户，可以用下列形式指定任何用户的口令：

代码：

```
# passwd sam
New password:*****
Re-enter new password:*****
```

普通用户修改自己的口令时，passwd 命令会先询问原口令，验证后再要求用户输入两遍新口令，如果两次输入的口令一致，则将这个口令指定给用户；而超级用户为用户指定口令时，就不需要知道原口令。

为了系统安全起见，用户应该选择比较复杂的口令，例如最好使用 8 位长的口令，口令中包含有大写、小写字母和数字，并且应该与姓名、生日等不相同。

为用户指定空口令时，执行下列形式的命令：

代码：

```
# passwd -d sam
```

此命令将用户 sam 的口令删除，这样用户 sam 下一次登录时，系统就不再询问口令。

passwd 命令还可以用 -l (lock) 选项锁定某一用户，使其不能登录，示例::

代码：

```
# passwd -l sam
新建用户异常：
useradd -d /usr/hadoop -u 586 -m hadoop -g hadoop
1 Creating mailbox file: 文件已存在
  删除即可 rm -rf /var/spool/mail/用户名
2 useradd: invalid numeric argument 'hadoop'
  这是由于 hadoop 组不存在 请先建 hadoop 组
通过 cat /etc/passwd 可以查看用户的 pass
cat /etc/shadow 可以查看用户名
```

`cat /etc/group` 可以查看 组

# linux 下创建用户

## Linux 系统用户组的管理

每个用户都有一个用户组，系统可以对一个用户组中的所有用户进行集中管理。不同 Linux 系统对用户组的规定有所不同，如 Linux 下的用户属于与它同名的用户组，这个用户组在创建用户时同时创建。用户组的管理涉及用户组的添加、删除和修改。组的增加、删除和修改实际上就是对/etc/group 文件的更新。

1、增加一个新的用户组使用 `groupadd` 命令。 其格式如下：

代码：

```
groupadd 选项 用户组
```

可以使用的选项有：

代码：

`-g GID` 指定新用户组的组标识号（GID）。

`-o` 一般与 `-g` 选项同时使用，表示新用户组的 GID 可以与系统已有用户组的 GID 相同。

示例 1：

代码：

```
# groupadd group1
```

此命令向系统中增加了一个新组 `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。

示例 2：

代码：

```
#groupadd -g 101 group2
```

此命令向系统中增加了一个新组 `group2`，同时指定新组的组标识号是 101。

2、如果要删除一个已有的用户组，使用 `groupdel` 命令， 其格式如下：

代码：

```
groupdel 用户组
```

示例：

代码：

```
#groupdel group1
```

此命令从系统中删除组 group1。

3、修改用户组的属性使用 groupmod 命令。 其语法如下：

代码：

```
groupmod 选项 用户组
```

常用的选项有：

代码：

-g GID 为用户组指定新的组标识号。

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

-n 新用户组 将用户组的名字改为新名字

示例 1：

代码：

```
# groupmod -g 102 group2
```

此命令将组 group2 的组标识号修改为 102。

示例 2：

代码：

```
# groupmod -g 10000 -n group3 group2
```

此命令将组 group2 的标识号改为 10000，组名修改为 group3。

4、如果一个用户同时属于多个用户组，那么用户可以在用户组之间切换，以便具有其他用户组的权限。用户可以在登录后，使用命令 newgrp 切换到其他用户组，这个命令的参数就是目的用户组。 示例::

代码：

```
$ newgrp root
```

这条命令将当前用户切换到 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。类似于用户账号的管理，用户组的管理也可以通过集成的系统管理工具来完成。

## 权限分配

分配权限

```
chown -R hadoop:hadoop /usr/hadoop/
```

让普通用户拥有 root 的权限

1. root 登录
2. adduser 用户名
3. passwd 用户名  
    确定密码
4. 修改/etc/passwd 即可，把用户名的 ID 和 ID 组修改成 0。