



LINUX用户管理

信息工程学院

- 单用户单任务操作系统
DOS等
- 单用户多任务操作系统
Windows95, Windows98, Windows XP等
- 多用户多任务操作系统
Win 7, Win 10, Linux, Unix等

- **Linux**系统是个多任务、多用户系统，允许在同一时刻有多个用户同时访问系统，因此一定要有文件权限控制机制。
- **Linux**系统的权限控制机制和**Windows**的权限控制机制有着很大的差别
 - **Linux**的文件或目录被一个用户拥有时，这个用户称为文件的拥有者（或属主）；
 - 同时文件还被指定的用户组所拥有，这个用户组称为文件所属组；
 - 文件的权限由权限标志来决定，权限标志决定了文件的拥有者、文件的所属组、其他用户对文件访问的权限。

- 在linux系统中，使用用户和组的概念来区分和组合不同的用户，它们通过uid和gid识别，并且根据这两个概念给文件系统赋予权限。
- 用户：一定资源的使用者，可以创建和管理文件以及访问其他用户文件。uid即user ID，系统给用户分配的标识号，uid是唯一的。
- 用户组：由一定数量的对某些文件具有相同操作权限的用户组成的小组。gid即group ID，也就是用户所在的组的标识号。
 - 一个用户可以从属于多个用户组
 - 一个用户组可以拥有多个用户

Linux用户分为三种：

- 超级用户 (**root, UID=0**)
 - 系统用户 (**UID=1-499/999, 65534**)
 - 普通用户 (**UID=1000-2³²-1**)
-
- **id**命令可以查看当前用户**UID**号,
 - **id + 用户名**可以查指定用户**UID**号

```
[root@CM00 ~]# id
uid=0(root) gid=0(root) 组=0(root) 环境=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@CM00 ~]# su wlw
[wlw@CM00 root]$ id
uid=500(wlw) gid=500(wlw) 组=500(wlw) 环境=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

```
[root@CM00 ~]# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
usbmuxd:x:113:113:usbmuxd user:/:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
rtkit:x:499:499:RealtimeKit:/proc:/sbin/nologin
avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin
vcsa:x:69:69:virtual console memory owner:/dev:/sbin/nologin
abrt:x:173:173:/:etc/abrt:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
haldaemon:x:68:68:HAL daemon:/:/sbin/nologin
ntp:x:38:38:/:etc/ntp:/sbin/nologin
apache:x:48:48:Apache:/var/www:/sbin/nologin
saslauth:x:498:76:Saslauthd user:/var/empty/saslauth:/sbin/nologin
postfix:x:89:89:/:var/spool/postfix:/sbin/nologin
gdm:x:42:42:/:var/lib/gdm:/sbin/nologin
pulse:x:497:496:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72:/:/sbin/nologin
dhcpd:x:177:177:DHCP server:/:/sbin/nologin
wlw:x:500:500:/:home/wlw:/bin/bash
jsj:x:501:501:/:home/jsj:/bin/bash
wlwks1:x:502:502:/:home/wlwks1:/bin/bash
```

- **超级用户root (UID=0)**
 - 系统最高权限的拥有者。
 - 是系统管理唯一的控制者；
 - 可以操作任何系统文件和命令。
 - 通常，应尽量避免直接用**root**用户进行登录。
- 如果把/etc/passwd中的某用户的**UID**的值改为**0**，则该用户则会被认为是**root**用户，从而可以进行所有的**root**操作。
- **UID**的值为**0**的是超级用户，而不是用户名为**root**的是超级用户

```
[root@CM00 ~]# cat /etc/passwd  
root:x:0:0:root:/root:/bin/bash
```

● 系统用户 (UID=1~499/1~999)

- 通常不需要或无法登录系统，是不能与人交互的特殊账号；
- 与系统和程序服务相关，是系统运行不可缺少的用户，故也被称为虚拟用户或系统用户，
- 有的是系统安装时自行建立的，有的是用户自定义的。

⑩ bin、daemon、shutdown、halt等，任何Linux系统默认都有这些系统用户

⑩ mail、news、games、apache、ftp、mysql及sshd等，与Linux系统的进程有关

```
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
```

```
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
```

关于UID用户标识号

- 普通用户（**UID=500-60000/1000-2³²-1**）
 - 这类用户可以登录系统，
 - 但只能操作属于自己目录的内容，权限有限，
 - 这类用户都是系统管理员自行添加的。

```
wlw:x:500:500::/home/wlw:/bin/bash  
jsj:x:501:501::/home/jsj:/bin/bash  
wlwks1:x:502:502::/home/wlwks1:/bin/bash  
wlwks2:x:503:503::/home/wlwks2:/bin/bash  
wlwks3:x:504:504::/home/wlwks3:/bin/bash  
wlwks4:x:505:505::/home/wlwks4:/bin/bash
```

◆ **UID**在系统中的唯一性关系到系统的安全。



su命令切换用户



```
[wlw@CM00 ~]$ su - root
```

密码:

```
[root@CM00 ~]#
```

#: 代表超级用户

#: 代表普通用户

```
[root@CM00 ~]# su wlw
```

```
[wlw@CM00 root]$ su
```

密码:

```
[root@CM00 ~]#
```

su 用户名 直接切换到所指定的用户下
su 不加任何参数时默认为切换到root用户

- 普通用户切换到其它任何用户时都需要密码验证
- 超级用户root向其他用户切换时不需要密码

pwd命令查看当前路径

```
[root@CM00 ~]# pwd  
/root
```

切换用户后，执行**pwd**查看是否完全切换？

```
[w1w@CM00 ~]$ pwd  
/home/w1w  
[w1w@CM00 ~]$  
[w1w@CM00 ~]$ su root  
密码：  
[root@CM00 w1w]# pwd  
/home/w1w
```

```
[root@CM00 w1w]# su w1w  
[w1w@CM00 ~]$  
[w1w@CM00 ~]$ su - root  
密码：  
[root@CM00 ~]# pwd  
/root  
[root@CM00 ~]# █
```

- **su**后不加**-**,表示不完全切换，仍保留原用户的一些特性.

- **su**后加**-**,表示完全切换，相当于完全登录新用户.

- 管理用户信息最主要的两个文件是：
 - 用户信息文件: **/etc/passwd**
 - 用户密码文件: **/etc/shadow**

- 管理用户组信息最主要的文件是：
 - 用户组信息文件: **/etc/group**
 - 用户组密码文件: **/etc/gshadow**

cat 命令查看文件内容

cat /etc/passwd 查看用户信息文件

- 此文件好比花名册，系统所有的用户都在这里登录的记载。
- 此文件中每一行表示的是一个用户的信息。一行有七个段位，每个段位用：分隔。

登录名：密码：用户标志号：组标志号：用户全名或其他描述：主目录：登录shell

- 下面是超级用户**root**和普通用户在此文件中对应的行：

```
[root@CM00 ~]# cat /etc/passwd  
root:x:0:0:root:/root:/bin/bash
```

```
wlw:x:500:500:~/home/wlw:/bin/bash  
jsj:x:501:501:~/home/jsj:/bin/bash
```

注：此处存放的口令**x**为屏蔽字符，真正密码保存在**/etc/shadow**文件中。



/etc/passwd用户信息文件



```
root@CM00 ~]# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
usbmuxd:x:113:113:usbmuxd user:/:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
rtkit:x:499:499:RealtimeKit:/proc:/sbin/nologin
avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin
vcsa:x:69:69:virtual console memory owner:/dev:/sbin/nologin
abrt:x:173:173:/:etc/abrt:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
haldaemon:x:68:68:HAL daemon:/:/sbin/nologin
ntp:x:38:38:/:etc/ntp:/sbin/nologin
apache:x:48:48:Apache:/var/www:/sbin/nologin
saslauth:x:498:76:Saslauthd user:/var/empty/saslauth:/sbin/nologin
postfix:x:89:89:/:var/spool/postfix:/sbin/nologin
gdm:x:42:42:/:var/lib/gdm:/sbin/nologin
pulse:x:497:496:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72:/:/sbin/nologin
dhcpd:x:177:177:DHCP server:/:/sbin/nologin
wlw:x:500:500:~/home/wlw:/bin/bash
jsj:x:501:501:~/home/jsj:/bin/bash
wlwksl:x:502:502:~/home/wlwksl:/bin/bash
```

最后一列的类型：

/bin/bash

说明系统使用**bash**，表示此用户可以登录系统，可以收邮件，可以登录**FTP**。

/sbin/nologin

表示此用户不能登录系统，可以收邮件，可以登录**FTP**。

/bin/false

表示此用户不能登录系统，可以收邮件，不能登录**FTP**



与/etc/passwd文件对应的shadow文件



```
[root@CM00 ~]# cat /etc/shadow
root:$6$jQQVCWC6$dImXN6Ges98RxYigxhFU3mBomxLyyv8AAkCarhSByNkeartZd.RvmbogqSQ30B5
rQVhsmGdwGmuekT8.kxpk00:19108:0:99999:7:::
bin:!:17246:0:99999:7:::
daemon:!:17246:0:99999:7:::
adm:!:17246:0:99999:7:::
lp:!:17246:0:99999:7:::
sync:!:17246:0:99999:7:::
shutdown:!:17246:0:99999:7:::
halt:!:17246:0:99999:7:::
mail:!:17246:0:99999:7:::
uucp:!:17246:0:99999:7:::
operator:!:17246:0:99999:7:::
games:!:17246:0:99999:7:::
gopher:!:17246:0:99999:7:::
ftp:!:17246:0:99999:7:::
nobody:!:17246:0:99999:7:::
dbus:!!:18379:!:!:!:
usbmuxd:!!:18379:!:!:!:
rpc:!!:18379:0:99999:7:::
rtkit:!!:18379:!:!:!:
avahi-autoipd:!!:18379:!:!:!:
vcsa:!!:18379:!:!:!:
abrt:!!:18379:!:!:!:
rpcuser:!!:18379:!:!:!:
nfsnobody:!!:18379:!:!:!:
haldaemon:!!:18379:!:!:!:
ntp:!!:18379:!:!:!:
apache:!!:18379:!:!:!:
ssslauth:!!:18379:!:!:!:
postfix:!!:18379:!:!:!:
gdm:!!:18379:!:!:!:
pulse:!!:18379:!:!:!:
sshd:!!:18379:!:!:!:
tcpdump:!!:18379:!:!:!:
dhcpcd:!!:18419:!:!:!:
wlv:$6$kWwrdLPv$j8HZ0YLy7Uh/CTwU1uP2h6LJUtp3J2qKFbBYIDrT.0HME/77rAM7jgNWQAXzWEf8
Apu004ERrrQUuHSue0bC4/:19108:0:99999:7:::
jsj:$6$ysyis0lP$ZodByZnBipbGpaFqNXzE3zPJEjcwwxHTC6Uow7R4KGTpEEENkETkTl0rVRyjDkrk
rUp.YD5kEEpXFF44ao1Ke1:19107:0:99999:7:::
```

/etc/shadow 用户密码文件

- 此文件只能在 **root** 用户下查看。
- 此文件为每个用户提供一条记录，它除了包括 **passwd** 文件中的相关信息外，还包括用户被加密的密码和一些时间期限。一行有九个段位，每个段位用：分隔。

用户名：密码：最后修改天数：最小时间间隔：最大时间间隔：警告时间：闲置时间：失效时间：保留

下面是该文件中 **root** 用户的密码文件信息行：

```
[root@CM00 ~]# cat /etc/shadow  
root:$6$jQQVCWC6$dImXN6Ges98RxYigxhFU3mBomxLyyv8AAkCarhSByNkeartZd.RvmbogqSQ30B5rQVhsmGdwGmuekT8.kxpk00:19108:0:99999:7:::
```

- **\$6\$jQQVCWC6\$dImXN6Ges98RxYigxhFU3mBomxLyyv8AAkCarhSByNkeartZd.RvmbogqSQ30B5rQVhsmGdwGmuekT8.kxpk00** 是加密后的密码
- **19108** 该口令修改后已过去的天数，从 **1970.1.1** 开始计算，
- **0** 需要再过多少天这个口令可以被修改
- **99999** 口令的有效期
- **7** 口令失效多少天前发出警告

- 对系统而言，创建一个用户账号需要完成以下步骤：

- ① 添加一个记录到/etc/passwd文件。
- ② 创建用户的主目录。
- ③ 在用户的主目录中设置用户的默认配置文件（如.bashrc）。

- 配置文件/etc/login.defs和/etc/default/useradd来保存创建用户时使用的默认参数。

```
[root@CM00 ~]# cat /etc/default/useradd
# useradd defaults file
GROUP=100
HOME=/home
INACTIVE=-1
EXPIRE=
SHELL=/bin/bash
SKEL=/etc/skel
CREATE_MAIL_SPOOL=yes
```

```
[root@CM00 ~]# cat /etc/login.defs
#
# Please note that the parameters in this configuration file control the
# behavior of the tools from the shadow-utils component. None of these
# tools uses the PAM mechanism, and the utilities that use PAM (such as the
# passwd command) should therefore be configured elsewhere. Refer to
# /etc/pam.d/system-auth for more information.
#
# "REQUIRED"
#   Directory where mailboxes reside. _or_ name of file, relative to the
#   home directory. If you _do_ define both, MAIL_DIR takes precedence.
#   QMAIL_DIR is for Qmail
#
MAIL_DIR      Maildir
MAIL_DIR      /var/spool/mail
MAIL_FILE     .mail
#
# Password aging controls:
#
#   PASS_MAX_DAYS   Maximum number of days a password may be used.
#   PASS_MIN_DAYS   Minimum number of days allowed between password changes.
#   PASS_MIN_LEN     Minimum acceptable password length.
#   PASS_WARN_AGE   Number of days warning given before a password expires.
#
PASS_MAX_DAYS 99999
PASS_MIN_DAYS 0
PASS_MIN_LEN 5
PASS_WARN_AGE 7
#
# Min/max values for automatic uid selection in useradd
#
UID_MIN        500
UID_MAX        60000
#
# Min/max values for automatic gid selection in groupadd
#
GID_MIN        500
GID_MAX        60000
#
# If defined, this command is run when removing a user.
# It should remove any at/cron/print jobs etc. owned by
# the user to be removed (passed as the first argument).
#
USERDEL_CMD    /usr/sbin/userdel_local
#
# If useradd should create home directories for users by default
# On RH systems, we do. This option is overridden with the -m flag on
# useradd command line.
#
CREATE_HOME    yes
#
# The permission mask is initialized to this value. If not specified,
# the permission mask will be initialized to 022.
UMASK          077
#
# This enables userdel to remove user groups if no members exist.
#
USERGROUPS_ENAB yes
#
# Use SHA512 to encrypt password.
ENCRYPT_METHOD  SHA512
```



常用用户管理命令



- 增加用户 **useradd/adduser**
- 删除用户 **userdel/deluser**
- 修改用户信息 **usermod**
- 设置用户口令 **passwd**

- 查看当前登录用户 **who**
- 查看用户号 **id**
- 查看用户详细信息 **finger**

1) 添加新用户——只能由root用户完成

格式: **useradd** [参数] 用户名

参数:

- **-u UID** //指定用户的UID值
- **-g 组名** //指定用户所属的默认组（该组名已存在）
- **-c 注释名** //指定用户的注释
- **-d 路径** //指定用户的家目录即“/home/用户登录名”
- **-e 时间** //指定用户帐号失效日期(YYYY-MM-DD)
- **-s shell类型** //指定默认的shell类型

◆ 特别说明: 会在主目录/home自动生成同名的目录, 当切换到该用户时, 会以“~”显示。

2) 删除用户——只能由root用户完成

格式: **userdel** [参数] 用户名

参数: **-r** // 同时删除用户主目录

实例: **# userdel -r u3**

- ◆ 特别说明: 如果不加参数**-r**, 主目录/**home**和**/var/spool/mail**目录下还会存在该用户的目录, 删除不彻底, 以后如果要添加同名的用户时会报错不允许, 因为主目录已存在。

- 如果遇到报 “**userdel: user *** is currently used by process *****” 时，因为**stud**已处于登录状态，要用**exit**退出该用户的登录状态后，再进入**root**删除。
- **useradd**和**userdel**都要在**root**用户下进行，如果在其他普通用户下，就要使用**sudo**命令来执行。
- 查看当前用户的登录状态用**who**或**w**命令。

3) 修改用户信息

格式: **usermod** [参数] 用户名

- **-l** 新用户名 当前用户名 //更改用户名
- **-d** 路径 //更改用户主目录
- **-G** 组名 //修改附加组
- **-L** 用户帐号名 //锁定用户帐号(不能登录)
- **-U** 用户帐号名 //解锁用户帐号

例: **# usermod -l stu -d home/stu -g test may**
//将用户may登录名改为stu加入到test组, 目录改为/home/stu

4) 设置用户口令

格式: **passwd** [用户名]

- **whoami** 查看当前登录的用户名

- 同id命令

- **who** 查看当前登录的所有用户信息

- 不同版本可看到的情况不同，**centos7**能把所有切换登录的用户都列出，**centos8**则不会，因为有些用户的登录是同一个用户打开的不同终端。

- **w** 查看当前登录的所有用户详细信息

```
[root@CM00 ~]# whoami
root
[root@CM00 ~]#
[root@CM00 ~]# who
wlw      tty1      2023-03-08 08:14 (:0)
wlw      pts/0     2023-03-08 08:14 (:0.0)
[root@CM00 ~]#
[root@CM00 ~]# w
08:50:46 up 39 min,  2 users,  load average: 0.07, 0.03, 0.01
USER      TTY      FROM          LOGIN@   IDLE   JCPU   PCPU WHAT
wlw       tty1      :0             08:14    39:11 10.09s  0.26s pam: gdm-password
wlw       pts/0    _              08:14    0.00s  0.22s  3.08s gnome-terminal
```



禁用和恢复用户



□ 禁用

#usermod -L username

#passwd -l username

□ 恢复

#usermod -U username

#passwd -u username

- 具有某种共同特征的用户集合起来就是用户组。
- 每个用户至少属于一个组，**Linux**系统可以对一个用户组中的用户进行集中管理。
- 一个用户可以从属于多个组。
- 和用户组相关的一个重要文件是：
 - **/etc/group**



/etc/group组数据文件



- 系统中的每个组都对应文件/etc/group中的一行记录。
- 每行记录的形式如下：
组名：密码：组标志号GID：用户列表User_list

例如：

root:x:0:

bin:x:1:

daemon:x:2:

...

may:x:100:

常用用户组管理命令

- 增加用户组 **groupadd/addgroup**
- 删除用户组 **groupdel/delgroup**
- 修改用户组信息 **groupmod**
- 设置组密码&管理组成员 **gpasswd**
- 查看当前用户所在组 **group**
- 查看用户组号 **id**
- 修改文件所属组 **chgrp**

常用的用户组管理命令

1) 建立组

格式: **groupadd** [参数] 组名

参数:

- **-g GID** //指定新建组的**GID**值
- **-r** //建立伪用户组 (**1--499**)

2) 删除组

格式: **groupdel** 组名

3) 修改组的信息

格式: **groupmod** [参数] 组名

参数:

- **-n** 新组名 原组名 //修改组的名称
- **-g** GID //修改组的GID

4) 设置组密码及管理组内成员

格式: **gpasswd** [参数] 组名

参数:

- **-a** 用户名 //向指定组添加用户
- **-d** 用户名 //从指定组中删除用户
- **-A** 设置用户组管理员
- **-r** 删除用户组密码
- **-R** 禁止用户切换为该组

5) 查看当前用户加入哪些组

格式: **groups** [用户名]

实例:

- ◆ **#groups** (显示当前用户所属组)
- ◆ **#groups root** (显示root用户的所属组)

◆ 特别说明: groupadd和groupdel都要在root用户下进行, 如果在其他普通用户下, 就要使用sudo命令来执行。

- 查看当前用户所属组用groups命令
- 查看操作后的用户组信息看文件/etc/group。

用户组管理的一般用法举例

- 添加一个用户组test输入 **#groupadd test**

说明：新添用户的uid号会自动接到已有用户之后。

- 将用户may加入用户组test输入 **#gpasswd -a may test**

- 删除已有用户组test输入 **#groupdel test**

说明：删除用户组时，要先删除用户组成员



用户组权限示例



授权用户jack和mary对目录/software有写权限

```
# groupadd softadm           //先添加组softadm
# usermod -G softadm jack    //修改jack加入softadm组
# gpasswd -a mary softadm    //再添加mary加入softadm组
# chgrp softadm /software    //修改目录所属组也为softadm
# chmod g+w /software        //修改对该目录所属组有写入权限
```

```
# ls -ld /software
```

```
Drwxrwxr-x 2 root softadm 512 Jul 14 06:17 /software
```

```
# grep softadm /etc/group
```

```
softadm::100:jack,mary
```

- **Linux**系统中规定了**3**种不同类型的用户：
 - 文件属主 (**user**)
 - 同组用户 (**group**)
 - 可以访问系统的其他用户 (**others**)
- 访问权限规定**3**种访问文件或目录的方式：
 - 读 (**r**)
 - 写 (**w**)
 - 可执行或查找 (**x**)



sudo给普通用户提升权限



- **su命令**是启用**root**账户，容易造成破坏性后果。在实际应用中，很多管理员为了避免误操作，也会不使用**root**，而使用普通用户。
- **sudo命令**是临时使用**root**权限，如果普通用户想做**root**用户才有权限做的管理操作时使用，执行完毕后自动返回普通用户状态。

- 例：root授权普通用户teacher所有权限
- 此做法安全隐患大，不推荐

```
[root@CM00 ~]# useradd teach
[root@CM00 ~]# passwd teach
更改用户 teach 的密码 。
新的 密码：
无效的密码： WAY 过短
无效的密码： 过于简单
重新输入新的 密码：
passwd： 所有的身份验证令牌已经成功更新。
[root@CM00 ~]# █
```

```
[root@CM00 ~]# su - teach
[teach@CM00 ~]$ touch text
[teach@CM00 ~]$ ls
text
[teach@CM00 ~]$ touch /root/text
touch: 无法创建"/root/text": 权限不够
[teach@CM00 ~]$ sudo touch /root/text
[sudo] password for teach:
teach 不在 sudoers 文件中。此事将被报告。
[teach@CM00 ~]$ █
```

```
[root@CM00 ~]#vi /etc/sudoers
```

```
##
```

```
.....
```

```
root          ALL=(ALL)          ALL
```

```
teacher      ALL=(ALL)          ALL
```

//在文档中增加该行，复制root行到新用户名行

```
.....
```

- 此文档只读，执行wq!强制保存。

```
[root@CM00 ~]# su - teach
```

```
[teach@CM00 ~]$ touch /root/teacher
```

touch: 无法创建"/root/teacher": 权限不够

```
[teach@CM00 ~]$ sudo touch /root/teacher
```

```
[sudo] password for teach:
```

```
[teach@CM00 ~]$
```

如何安全的给普通用户提升权限

● 实际应用尽量考虑权限最小化

```
[root@CM00 ~]#vi /etc/sudoers
##
```

```
.....
```

```
root          ALL=(ALL)          ALL
```

```
teacher      ALL=/usr/bin touch /root/teacher
```

```
// 此时teacher用户只能执行sudo touch命令，  
   且只能在/root/teacher目录下执行
```

```
.....
```

```
[root@CM00 ~]#vi /etc/sudoers
```

```
##
```

```
.....
```

```
root          ALL=(ALL)          ALL
```

```
teacher      ALL=/usr/sbin/route,/usr/sbin/ifconfig
```

```
// 此时teacher用户只能执行sudo route和ifconfig命令
```

```
.....
```

```
[root@CM00 ~]#vi /etc/sudoers
```

```
##
```

```
.....
```

```
Cmnd_Alias NETWORKING=/usr/sbin/route,/usr/sbin/ifconfig
```

```
// 多条命令可定义一个别名，把别名授权给用户即可
```

```
root          ALL=(ALL)          ALL
```

```
teacher      ALL=NETWORKING
```

```
.....
```