第 3 章　用户和组管理

Linux 是个多用户多任务的分时操作系统，所有要使用系统资源的用户都必须先向系统管理员申请一个账号，然后以这个账号的身份进入系统。用户的账号一方面能帮助系统管理员对使用系统的用户进行跟踪，并控制他们对系统资源的访问；另一方面也能帮助用户组织文件，并为用户提供安全性保护。每个用户账号都拥有一个唯一的用户名和用户密码。用户在登录时输入正确的用户名和密码后，才能进入系统和自己的主目录。

本章要点：

(1) 掌握实现用户账号的管理方法。

(2) 掌握用户账号的添加、删除和修改。

(3) 掌握用户密码的管理。

(4) 掌握用户组的管理。

3.1　用 户 和 组

在 Linux 系统中每个用户都拥有一个唯一的标识符，称为用户 ID(UID)，每个用户对应一个账号。Linux 系统把具有相似属性的多个用户分配到一个称为用户分组的组中，每个用户至少属于一个组。系统安装完毕后，已创建了一些特殊用户，它们具有特殊的意义，其中最重要的是超级用户，即 root。用户分组是由系统管理员建立的，一个用户分组内包含若干个用户，一个用户也可以归属于不同的分组。用户分组也有一个唯一的标识符，称为组 ID(GID)。对某个文件的访问都是以文件的用户 ID 和分组 ID 为基础的。同时可以根据用户和分组信息控制如何授权用户访问系统，以及被允许访问后用户可以进行的操作权限。

根据用户的权限用户可以定义为普通用户、系统用户和超级用户。普通用户只能访问自己的文件和其他有权限执行的文件，而超级用户权限最大，可以访问系统的全部文件并执行任何操作。超级用户也被称为根用户，一般系统管理员使用的是超级用户 root 的权限，有了这个权限，管理员可以突破系统的一切限制，方便地维护系统。普通用户也可以用 su 命令使自己转变为超级用户。而系统用户是指系统内置的、执行特定任务的用户，不具有登录系统的能力。

系统的这种安全机制有效地防止了普通用户对系统的破坏。例如：存放于/dev 目录下的设备文件分别对应于硬盘驱动器、打印机、光盘驱动器等硬件设备，系统通过对这些

文件设置用户访问权限，使得普通用户无法通过覆盖硬盘而破坏整个系统，从而保护了系统。

在 Linux 中可以利用用户配置文件，以及用户查询和管理的控制工具来进行用户管理，用户管理主要通过修改用户配置文件完成。用户管理控制工具最终的目的也是为了修改用户配置文件，所以在进行用户管理的时候，直接修改用户配置文件同样可以达到用户管理的目的。

3.1.1　用户账号文件

/etc/passwd 是系统识别用户的一个文件，用来保存用户的账号数据等信息，又称为密码文件。系统所有的用户都在此文件中有记载。例如：当用户以 zhang 这个账号登录时，系统首先会查阅/etc/passwd 文件，看是否有 zhang 这个账号，然后确定 zhang 的 UID，通过 UID 来确认用户和身份。如果存在，则读取/etc/shadow 影子文件中所对应的 zhang 的密码，如果密码核实无误，则登录系统并读取用户的配置文件。

用户登录进入系统后都有一个属于自己的操作环境，可以执行 cat 命令查看完整的系统账号文件。假设当前用超级用户身份登录，执行下列命令：

```
#cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
adm:x:2:2:daemon:/sbin:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
⋮
zhang:x:500:500::/home/zhang:/bin/bash
```

在/etc/passwd 中，每一行都表示一个用户的信息，一行有 7 个段位，每个段位用“:”号分隔，其格式如下：

```
username:password:User ID:Group ID:comment:home directory:shell
```

字段含义如下所示。

(1) username：用户名，它唯一地标识了一个用户账号，用户在登录时使用的就是它。通常长度不超过 8 个字符，可由大小写字母(区分大小写)、下画线、句点或数字等组成。用户名中不能有冒号，因为冒号在这里是分隔符。为了兼容起见，在创建用户时，用户名中最好不要包含点字符“.”，并且不使用连字符“-”和加号“+”打头。

(2) password：该账号的密码，passwd 文件中存放的密码是经过加密处理的，一般采用的是不可逆的加密算法。当用户登录输入密码后，系统会对用户输入的密码进行加密，再把加密的密码与机器中存放的用户密码进行比较。如果这两个加密数据匹配，则允许用户进入系统。目前许多 Linux 系统都使用了 shadow 技术，把真正的加密后的用户密码字存放到/etc/shadow 文件中，而在/etc/passwd 文件的密码字段中只存放一个特别的字符，例如“x”或“*”。Linux 的加密算法很严密，其中的密码很难被破解。盗用账号的人一般都借助专门的黑客程序，构造出无数个密码，然后使用同样的加密算法将其加密，

再和本字段进行比较，如果相同，就代表构造出的密码是正确的。因此，建议不要使用生日、常用单词等作为密码，它们在黑客程序面前几乎是不堪一击的。特别是对那些直接连入较大网络的系统来说，系统安全性显得尤为重要。

(3) User ID：用户识别码，简称 UID。此字段非常重要，Linux 系统内部使用 UID 来标识用户，而不是用户名。在系统中每个用户的 UID 的值是唯一的，更确切地说，每个用户都要对应一个唯一的 UID。一般情况下 UID 和用户名是一一对应的，如果几个用户名对应的用户标识号是相同的，系统内部将把他们视为同一个用户，不过他们能有不同的密码、不同的主目录及不同的登录 shell 等。通常 UID 的取值范围是 0～65535 的整数(UID 的最大值可以在文件/etc/login. gefs 中查到，一般 Linux 发行版约定为 60000)。其中，0 是超级用户 root 的标识号，1～499 作为管理账号，普通用户的标识号从 500 开始。

(4) Group ID：用户组识别码，简称 GID。不同的用户可以属于同一个用户组，享有该用户组共有的权限。与 UID 类似，GID 唯一地标识了一个用户组。

(5) comment：这是给用户账号做的注解，它一般是用户真实姓名、电话号码、住址等，当然也可以是空的。这个字段并没有什么实际的用途。在不同的 Linux 系统中，这个字段的格式并没有统一。在许多 Linux 系统中，这个字段存放的是一段任意的注释性描述文字，用做 finger 命令的输出。

(6) home directory：主目录，系统为每个用户配置的单独使用环境，即用户登录系统后最初所在的目录，在这个目录中，用户不仅可以保存自己的配置文件，还可以保存自己日常工作中的各种文件。一般来说，root 账号的主目录是/root，其他账号的主目录都在/home 目录下，并且和用户名同名。各用户对自己的主目录有读、写、执行(搜索)权限，其他用户对此目录的访问权限则根据具体情况设置。用户可以在账号文件中更改用户登录目录。

(7) login command：用户登录后，要启动一个进程，负责将用户的操作传给内核，这个进程是用户登录到系统后运行的命令解释器或某个特定的命令，即 shell。shell 是用户和 Linux 系统之间的接口。Linux 的 shell 有许多种，每种都有不同的特点。系统管理员能根据系统情况和用户习惯为用户指定某个 shell。如果不指定 shell，那么系统使用 sh 为默认的登录 shell，即这个字段的值为/bin/sh。

用户的登录 shell 也可以指定为某个特定的程序(此程序不是命令解释器)。利用这一特点，能限制用户只能运行指定的应用程序，在该应用程序运行结束后，用户就自动退出了系统。有些 Linux 系统要求只有那些在系统中登记了的程序才能出现在目前这个字段中。系统中有一类用户称为伪用户(pseudo users)，这些用户在/etc/passwd 文件中也占有一条记录，不过不能登录，因为他们的登录 shell 为空。他们的存在主要是方便系统管理，满足相应的系统进程对文件属主的需求。常见的伪用户有 bin(拥有可执行的用户命令文件)、sys(拥有系统文件)、adm(拥有账户文件)等。

除了上面列出的伪用户外，还有许多标准的伪用户，例如：audit、cron、mail、usenet 等，它们也都各自为相关的进程和文件所需要。由于/etc/passwd 文件是所有用户都可读的，如果用户的密码太简单或规律比较明显，一台普通的计算机就能够非常容易地将它

破解，因此对安全性要求较高的 Linux 系统把加密后的密码字分离出来，独立存放在一个文件中，这个文件是/etc/shadow 文件。只有超级用户才拥有该文件的读权限，这就确保了用户密码的安全性。

3.1.2　用户影子文件

Linux 使用了不可逆算法来加密登录密码，所以黑客从密文得不到明文。但由于任何用户都有权限读取/etc/passwd 文件，用户密码保存在这个文件中是极不安全的。针对这种安全问题，许多 Linux 的发行版本引入了影子文件/etc/shadow 来提高密码的安全性。使用影子文件是将用户的加密密码从/etc/passwd 中移出，保存在只有超级用户 root 才有权限读取的/etc/shadow 中，/etc/passwd 中的密码域显示一个“x”。

/etc/shadow 文件是/etc/passwd 的影子文件，这个文件并不是由/etc/passwd 产生的，这两个文件是对应互补的。shadow 内容包括用户、被加密的密码，以及其他/etc/passwd 不能包括的信息，比如用户的有效期限等。

/etc/shadow 文件的内容包括 9 个字段，每个字段之间用“：”号分隔。用户可以输入命令“cat /etc/shadow”来查看影子文件的内容，如下所示。

```
#cat /etc/shadow |more
root:$6$M9sgi327sdggd62hjH5Fdsrthjk&68fgdsd43$hgk&jgdsf2kjb@jhghfhgh5jfds
6ffd768h%jggh(khhhvh%hgYgg6kjUgff.:14997:0:99999:7:::
bin:*:14790:0:99999:7:::
daemon:*:17790:0:99999:7:::
lp:*:14790:0:99999:7:::
⋮
zhang: *: $6 $fg7DUHGgrtjrsuutc548hxdsahfe289hjgfd $68gcx # uhjgcg% hfgffse
h67765hgdshju%hhkk*hkhbjgj%hghgjgkk/:14997:0:99999:7:::
```

(1) 用户名(也被称为登录名)。在/etc/shadow 中，用户名和/etc/passwd 是相同的，这样就把 passwd 和 shadow 中的用户记录联系在一起。这个字段是非空的。

(2) 密码(已被加密)。如果有些用户在这段是“*”，表示这个用户不能登录到系统；这个字段是非空的，带有 1 个“!”表示账户被锁定，带有 2 个“!”表示密码被锁定。

(3) 上次修改密码的时间。这个时间是从 1970 年 1 月 1 日算起到最近一次修改密码的时间间隔(天数)，管理员可以通过 passwd 来修改用户的密码，然后查看/etc/shadow 中此字段的变化。

(4) 两次修改密码间隔最少的天数。也就是说用户必须经过多少天才能修改其密码。如果配置为 0，则禁用此功能。此项功能用处不是太大，默认值通过/etc/login. defs 文件中的 PASS_MIN_DAYS 进行定义。

(5) 两次修改密码间隔最多的天数。这个字段可以增强管理员管理用户密码的时效性，也增强了系统的安全性。如果是系统默认值，则在添加用户时由/etc/login. defs 文件中的 PASS_MAX_DAYS 进行定义。

(6) 提前多少天警告用户密码将过期。如果满足条件，则当用户登录系统后，系统登

录程序提醒用户密码将要作废；系统默认值在添加用户时由/etc/login.defs 文件中的 PASS_WARN_AGE 进行定义。

(7) 在密码过期多少天之后禁用此用户。此字段表示用户密码作废多少天后，系统会禁用此用户，也就是说系统不会再让此用户登录，也不会提示用户过期，是完全禁用。

(8) 用户过期日期；此字段指定了用户作废的天数(从 1970 年 1 月 1 日开始的天数)，如果这个字段的值为空，则账号长久可用。

(9) 保留字段，目前为空，以备将来 Linux 发展之用。

3.1.3 组账号文件

具有某种共同特征的用户集合起来就是用户组(group)。用户组的设置主要是为了方便检查、设置文件或目录的访问权限。每个用户组都有唯一的用户组号 GID。

/etc/group 文件是用户组的配置文件，内容包括用户和用户组，并且能显示出用户归属哪个用户组或哪几个用户组。同一用户组的用户之间具有相似的特征，比如把某一用户加入到 info 用户组，那么这个用户就可以浏览 info 用户登录目录的文件。如果 info 用户把某个文件的读写执行权限放开，info 用户组的所有用户都可以修改此文件，如果是可执行的文件(比如脚本)，info 用户组的用户也是可以执行的。

/etc/group 的内容包括用户组名、用户组密码、GID 及该用户组所包含的用户，每个用户组使用一条记录。格式如下：

```
group_name:passwd:GID:user_list
```

/etc/group 中的每条记录分 4 个字段。第 1 字段：用户组名称；第 2 字段：用户组密码；第 3 字段：GID；第 4 字段：用户列表，每个用户之间用逗号(,)分隔，本字段可以为空，如果字段为空表示用户组为 GID 的全部用户。

通过执行“cat /etc/group”命令，可以得到/etc/group 文件的内容，如下所示。

```
#cat /etc/group|more
root:*:0:root
bin:*:1:root,bin,daemon
deamon:*:2:root,bin,daemon
lp:*:7:daemon,adm
⋮
zhang:*:500:
```

其中，第 2 行 root:x:0:root 的含义为：用户组名为 root，x 是已加密的密码段，GID 是 0，root 用户组下包括 root 用户。

GID 和 UID 类似，是一个从 0 开始的正整数。root 用户组的 GID 为 0。系统会预留一些较靠前的 GID 给系统虚拟用户组用。

对照/etc/passwd 和/etc/group 两个文件，会发现在/etc/passwd 中的每条用户记录有用户默认的 GID。在/etc/group 中，也会发现每个用户组下有多少个用户。在创建目录和文件时会使用默认的用户组。

需要注意的是，判断用户的访问权限时，默认的 GID 并不重要，只要一个目录让同组用户具有可以访问的权限，那么同组用户就可以拥有该目录的访问权限。

3.1.4　用户组影子文件

与/etc/shadow 文件一样，考虑到组信息文件中密码的安全性，引入相应的组密码影子文件/etc/gshadow。

/etc/gshadow 是/etc/group 的加密文件，比如用户组管理密码就存放在这个文件中。/etc/gshadow 和/etc/group 是互补的两个文件。对于大型服务器，针对很多用户和组，定制一些关系结构比较复杂的权限模型，设置用户组密码是极有必要的。例如，如果不想让一些非用户组成员永久拥有用户组的权限和特性，这时就可以通过密码验证的方式来让某些用户临时拥有一些用户组特征，这时就要用到用户组密码。

/etc/gshadow 格式如下，每个用户组独占一行。

```
groupname:passwd:admin1,admin2,...:member1,member2,...
```

第 1 字段：用户组；第 2 字段：用户组密码，这个字段可以是空的或“!”，如果是空的或“!”，表示没有密码；第 3 字段：用户组管理者，这个字段也可为空，如果有多个用户组管理者，用“,”分隔；第 4 字段：组成员，如果有多个成员，用“,”分隔。

执行“cat /etc/gshadow”命令，可以查看用户组影子文件的内容，如下所示。

```
#cat /etc/gshadow|more
root:::root
bin:::root,bin,daemon
daemon:::root,bin,daemon
sys:::root,bin,adm
⋮
zhang:!!::
```

其中一行 daemon:::root,bin,daemon 的含义为：用户组名为 daemon，没有设置密码，该用户没有用户组管理者，组成员有 root、bin 和 daemon。

3.1.5　与用户和组管理相关的文件和目录

1. /etc/skel

/etc/skel 目录一般存放用于初始化用户启动文件的目录，这个目录由 root 权限控制。一般来说，每个用户都有自己的主目录，用户成功登录后就处于自己的主目录下。当用 useradd 命令添加用户时，这个目录下的文件自动复制到新添加的用户的目录下。/etc/skel目录下的文件都是隐藏文件，也就是类似“.file”格式的；可通过修改、添加、删除/etc/skel 目录下的文件，来为用户提供一个统一、标准的、默认的用户环境。典型的/etc/skel内容如下：

```
#ls -a /etc/skel/
.  ..  .bash_logout  .bash_profile  .bashrc  .gnome2  .mozilia
```

2. /etc/login.defs 配置文件

/etc/login.defs 文件用于创建用户账号时进行的一些规划，比如创建用户时，是否需要创建用户家目录、用户的 UID 和 GID 的范围、用户的期限等，这个文件是可以通过 root 来定义的。典型的/etc/login.defs 文件内容如下。

```
#cat /etc/login.defs
⋮
MAIL_DIR        /var/spool/mail     //创建用户时,用户邮箱所在的目录
⋮
PASS_MAX_DAYS   99999     //账户密码的最长有效天数
PASS_MIN_DAYS   0         //账户密码的最短有效天数,允许更改密码的最短天数
PASS_MIN_LEN    5         //密码最小长度
PASS_WARN_AGE   7         //密码过期前提前警告的天数
⋮
UID_MIN         500       //建立用户时,自动产生的最小 UID 值,也就是 UID 是从此值开始
UID_MAX         60000     //建立用户时,最大的 UID 值
⋮
GID_MIN         500       //建立用户时,自动产生的最小 GID 值
GID_MAX         60000     //建立用户时,自动产生的最大 GID 值
⋮
CREATE_HOME     yes       //建立用户时,是否创建用户家目录
⋮
UMASK           077       //默认创建文件和目录的权限
⋮
USERGROUPS_ENAB  yes      //创建用户时是否创建用户主群组
⋮
ENCRYPT_METHOD  SHA512    //用户的口令使用 SHA512 加密算法加密
```

3. /etc/default/useradd 文件

该文件是通过 useradd 命令新建用户时的规则文件，其内容如下。

```
#more /etc/default/useradd
GROUP=100                 //默认用户群组 ID
HOME=/home                //把用户的家目录建在/home 中
INACTIVE=-1               //是否启用账号过期停权,-1 表示不启用
EXPIRE=                   //账号终止日期,不设置表示不启用
SHELL=/bin/bash           //默认登录 SHELL 的类型
SKEL=/etc/skel            //存放用于初始化用户文件的目录;当使用 adduser 添加用户时,
                            用户家目录下的文件都是从这个目录中复制过去的
CREATE_MAIL_SPOOL=yes     //是否主动帮使用者建立邮件信箱
```

3.2　用户账号的管理

3.2.1　用户账号管理

用户账号的管理主要涉及用户账号的添加、删除和修改等。

1. 添加账号

添加用户账号就是在系统中创建一个新账号，可以同时为新账号分配用户号、用户组、主目录和登录 shell 等资源。如果没有给刚添加的账号设置密码，则该账号是被锁定的，无法使用。

添加新的用户账号使用 useradd 命令，语法如下：

```
useradd [选项] 用户名
```

其中常用选项含义如下。

-c comment：指定一段注释性描述。

-d home_dir：指定用户主目录，如果此目录不存在，则同时使用“-m”选项创建主目录。

-m：若主目录不存在，则创建它。

-M：不创建主目录。

-g group：指定用户初始所属的用户组名或组 ID。该用户组名或组 ID 在指定时必须已存在。

-G 用户组列表：指定用户所属的附加组，各组用逗号隔开。

-s Shell：指定用户的登录 shell，默认为/bin/bash。

-u userID：指定新用户的用户号，该值必须唯一且大于 499，如果同时有-o 选项，则能重复使用其他用户的标识号。

-p password：为新建用户指定登录密码。此处的 password 是对登录密码经 md5 加密后所得到的密码值，不是真实密码原文，因此实际应用中使用较少。

例 1：

```
#useradd -d /tmp/wuli -m wuli
```

此命令创建了一个用户 wuli，其中-d 和-m 选项用来为登录名 wuli 产生一个主目录/tmp/wuli(/tmp 为当前用户主目录所在的父目录)。

例 2：

```
#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。

增加用户账号就是在/etc/passwd 文件中增加了一条新用户的记录，同时会更新其他系统文件，如/etc/shadow、/etc/group 等。如果要查看系统在创建用户时默认的参数，可以使用如下命令：

```
#useradd -D
```

2. 删除账号

如果一个用户账号不再使用，要能从系统中删除。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，必要时还要删除用户的主目录。删除一个已有的用户账号可以使用 userdel 命令，格式如下：

```
userdel [选项] 用户名
```

常用的选项是-r，其作用是删除用户账号的同时把该用户的主目录一起删除。

例如：

```
#userdel -r wuli
```

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

3. 修改账号

修改用户账号就是根据实际情况更改用户（chgrp 是针对文件而言）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可以使用 usermod 命令，格式如下：

```
#usermod [选项] 用户名
```

常用的选项包括-c、-d、-m、-g、-G、-s、-u、-o 等，这些选项的意义和 useradd 命令中的相同，能为用户指定新的资源值。下面按用途介绍几个选项。

(1) 改变用户账号名

格式：

```
usermod -l 新用户名 原用户名
```

-l 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。

例如：

```
#usermod -l zhang zhao              //将用户 zhao 改名为 zhang
```

(2) 锁定账号

若要临时禁止用户登录，可将该用户账户锁定。其格式为：

```
usermod -L 用户名
```

Linux 锁定账户，也可直接在密码文件 shadow 的密码字段前加“!”来实现。

(3) 解锁账户

格式：

```
usermod -U 用户名
```

-U 选项是将指定的账户解锁，以便可以正常使用。

(4) 将用户加入其他组

格式：

```
usermod -G 组名或 GID 用户名
```

例如：

```
#usermod -G cheng tom          //将用户 tom 追加到 cheng 这个组
```

其他选项应用如下例：

```
#usermod -s /bin/ksh -d /home/zh -g developer wuli
```

此命令将用户 wuli 的登录 shell 修改为 ksh，主目录改为/home/zh，用户组改为 developer。

4. 查看账号属性

格式：

```
id [选项] [用户]
```

此命令是显示有效用户的 uid 和 gid，默认为当前用户的 id 信息。

常用的选项如下所示。-g 或--group 表示只显示用户所属群组的 ID；-G 或--groups 表示显示用户所属附加群组的 ID；-n 或--name 表示显示用户、所属群组或附加群组的名称；-r 或--real 表示显示实际 ID；-u 或--user 表示只显示用户 ID；--help 表示显示帮助；--version 表示显示版本信息。

此外，利用 groups [用户]命令可以显示用户所在的组，默认为当前用户所在的组信息。

3.2.2　用户密码管理

用户管理的另一项重要内容是用户密码的管理。用户账号刚创建时没有密码，是被系统锁定的，无法使用，必须为其指定密码后才能使用，即使是空密码。

1. 设置用户登录密码

指定和修改用户密码的 shell 命令是 passwd。超级用户能为自己和其他用户指定密码，普通用户只能修改自己的密码。命令的格式为：

```
passwd [选项] 用户名
```

可使用的选项如下所示。