第3章 用户和组

本章学习目标：

- 了解多用户系统的概念
- 掌握用户和组的概念
- 掌握管理用户和组的方法
- 了解用户登录过程和环境变量的设置

Linux操作系统是多用户、多任务系统，即允许多个用户同时登录Linux系统并启动多个任务(有的用户远程登录)。用户账号和用户组是进行身份鉴别和权限控制的基础，身份鉴别的目的是规定哪些人可以进入系统，而权限控制的目的则是规定进入系统的用户能做哪些操作。

3.1 多用户系统

一个安装好的但是没有启动的Linux系统是静态的系统，静态的Linux系统一般由根分区上的文件、目录和交换分区组成，内容不会发生改变；而启动的Linux系统称为动态的系统，动态的Linux系统一般由根分区上的文件、目录和虚拟内存(含交换区和物理内存)中的进程组成，动态系统里的内容会时时发生变化。

如果Linux系统被引导到多用户目标(默认引导目标是graphical.target，它就是多用户的，这方面的内容请参考第9章)，那么就允许已经注册的用户登录，例如图3.1所示的动

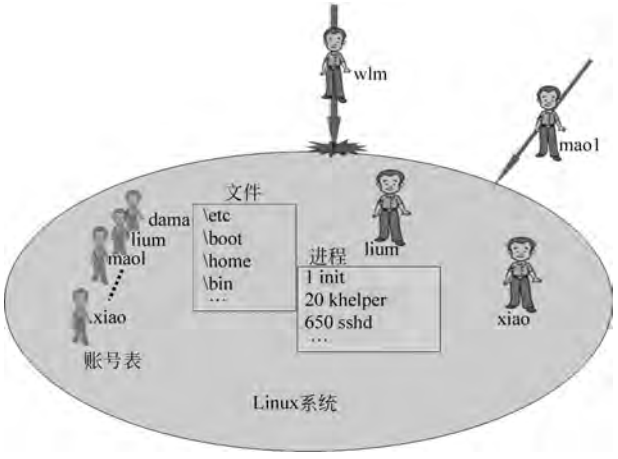

图3.1 多用户系统

态系统中有一个已注册用户的账号表，同时已经登录了两个用户 lium 和 xiao，另外两个人正在使用账号 wlm 和 mao1 登录，由于账号 wlm 不在系统的账号表中，所以这个用户登录一定失败，用户持账号 mao1 登录时必须输入正确的密码，否则系统拒绝用户登录。

所谓的多用户系统就是指一台计算机启动后，允许多个用户同时登录并使用计算机，最常见的例子是很多用户通过网络远程登录到一台运行 Linux 的计算机。远程登录的内容请参见 5.6 节。

3.2 用户和组的概念

3.2.1 用户的概念

有的人喜欢说“用户”，另一些人喜欢用“账号”，在 Linux 系统里，用户和账号是指同一个概念：使用 Linux 系统的人，他的信息必须事先在 Linux 系统里登记，登记之后才允许登录。创建(即登记)一个用户时需要提供如下信息。

(1) 用户名：也叫账号，正规的账号由 A～Z，a～z，0～9，-，_组成，账号长度介于 1～32，如 alice123、zsan 等，尽管允许使用中文，但建议不要用，因为字符登录界面往往无法输入中文。如果实在要在账号中加入其他字符，如空格、星号等，在创建用户时带上--badname 参数即可。在 Linux 系统中用户名是唯一的，用户名主要用于身份鉴别。

(2) 口令：或称密码，主要用于身份鉴别。一个好的口令最好同时包含大小写字母、数字和其他符号，长度建议大于 6。取口令的一种好的方法是对应古诗词中的一句话，例如 Y4yhL9t 对应“疑似银河落九天”，10LctwyyC 对应“十里长亭望眼欲穿”，这样的密码自己好记，别人难猜。

注意：密码中字母的大小写是有区分的。

(3) 用户 ID 号：简称为 UID，犹如人的身份证号码，但允许不唯一，也就是说允许多个不同的用户拥有相同的 UID，有点类似一个人拥有多个称呼，例如学名、绰号、小名等。UID 主要用于权限控制，由此可知，具有相同 UID 的用户具有相同的权限。

(4) 属组：每一个用户只能归属于一个主要组群，但是可以同时归属于多个附加组群。给用户分组主要是便于管理同一类用户的权限，例如赋予一个组某种权限，那么隶属于这个组的所有用户自动拥有该权限。主要组群和附加组群类似学校的班级和社团，一个学生只能属于一个班级，但可以加入若干社团。

(5) 家目录：用户登录后默认进入的目录。如果不特别指定，用户的家目录就是 /home/<账号>，例如创建用户 zsan，那么默认的家目录就是/home/zsan。root 用户的家目录有点特别，默认是/root。

(6) 登录 Shell：用户登录 Linux 的过程中，会自动执行一系列的程序，其中最后执行的那个程序称为 Shell 程序。Shell 意为“壳”，可想象为包裹在 Linux 系统外面的“壳”，用户登录后一直在这个“壳”中——与这个壳进行交互，用户输入的任何命令都由这个“壳”代为执行。目前常用的 Shell 程序有 Bash、tcsh、dash 等，其中 Bash 是默认的 Shell 程序，是最流行的“壳”。另外还有一些特殊的 Shell 程序，如 nologin、false，这两个 Shell 程序其实除了立即退出系统之外什么也不做，即当一个用户的登录 Shell 是这二者之一时，这个用户是不能登

录的,因为一登录就马上退出来了。自己编写的应用程序也可以作为登录 Shell,例如编写一个简单的关机、重启、修改系统时间、终止进程的程序,把这个程序设置成一个普通管理员的登录 Shell,登录后通过选择菜单可以完成这几个简单的任务。

(7) 备注:对用户的描述,这个可以省略。

登记的用户信息主要保存在文件/etc/passwd 和/etc/group 中,加密后的密码保存在文件/etc/shadow 中。/etc/passwd 每一行对应一个用户,一行的格式如下。

```
用户名:密码:UID:GID:备注:家目录:登录 Shell
```

各个参数之间用":"分开,其中的"密码"都用 x 代替(真正的密码保存在/etc/shadow 文件中),GID 是该用户的主要组群的组号。例如/etc/passwd 文件中有如下一行:

```
alice123:x:1000:500:Alice 的账号:/home/alice123:/bin/bash
```

从上面这一行可以获得如下信息:用户名是 alice123,密码的位置出现 x 只是占位符,没有任何意义,用户的 ID 是 1000,隶属于主要组群 500,用户的家目录是/home/alice123,登录 Shell 是/bin/bash,备注信息是"Alice 的账号"。

同样/etc/shadow 也是一行对应一个用户,格式如下:

```
账号:密码:最后一次更改密码的日期:密码有效期最少天数:密码有效期最多天数:密码警告时间段:
密码禁用期:账户过期日期:保留字段
```

(1) 密码:经过加密后的密文。这种加密算法是不可逆的,也就是说不能从密文反推算出原始密码。在用户登录校验密码时,Linux 系统采用相同的加密方法对用户输入的密码进行加密得到密文,然后通过比较两份密文是否相同来判断密码输入是否正确。

(2) 最后一次更改密码的日期:具体表示为从 1970 年 1 月 1 日以来的天数。例如在 2019 年 8 月 10 日修改过密码,那么这里就是 18117。如果是 0,那么下次用户登录时必须修改密码。

(3) 密码有效期最少天数:即自上次修改密码之后要过多少天后才允许再次修改密码。如果为 0 或者空表示没有限制,即可随时修改密码。

(4) 密码有效期最多天数:即多少天前必须修改密码。如果过了有效期最多天数还没有修改密码,那么当下一次用户登录时提示用户必须修改密码;为空表示没有限制,同时也没有密码警告时间段,没有密码禁用期;如果密码有效期最多天数小于密码有效期最少天数,那么用户不能修改密码。

(5) 密码警告时间段:即开始不断地通知用户要修改密码,如果为 0 或者空则不通知。

(6) 密码禁用期:过了密码有效期最多天数如果仍然没有修改密码,则进入密码禁用期。在禁用期内,用户登录时强制要求修改密码。过了禁用期,账号就完全冻结了,冻结的账户经过解冻之后可以继续使用。

(7) 账户过期日期:表示为从 1970 年 1 月 1 日以来的天数,为空则没有限制。账户过期后不能再用了。例如打算让账号在 2024 年 10 月 1 日失效,那么这里的值就是 19997。

这些参数之间的关系可以用图 3.2 来表示。

例如/etc/shadow 文件中有如下一行:

```
wochi: $6$87wjcyRC $J2rPOb.SQw:15142:10:20:3:5:19997:
```

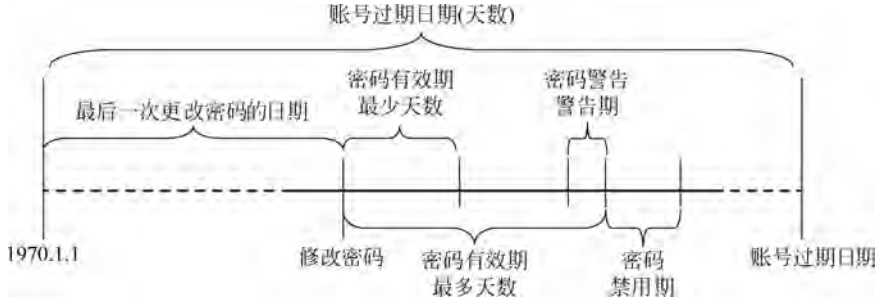

图 3.2 密码的老化过程

可以用图 3.3 来表示此密码的老化过程。

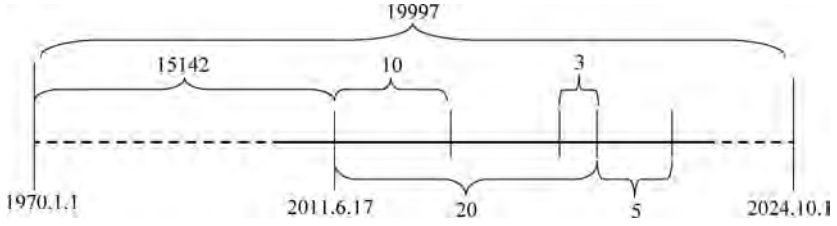

图 3.3 密码的老化过程举例

Linux 系统的用户分为三类,分别是超级用户 root、系统用户和普通用户。在安装系统时默认创建超级用户 root,root 用户的权力没有限制,它的 UID 和 GID 都是 0。超级用户的作用是管理系统,例如创建用户、给硬盘分区、配置网络等。系统用户主要用来启动服务或者用作一些特殊权限控制,系统用户的权限受到限制,系统用户也是在安装 Linux 或者应用软件时自动创建的,它们的 UID 小于 1000,系统用户不能登录。普通用户是由超级用户创建的并分配给 Linux 系统的使用者,权限受到限制,使用者用普通用户登录以完成他们的日常工作,普通用户的 UID 一般大于或等于 1000。下面的内容摘自/etc/passwd 文件:

```
root:x:0:0:root:/root:/bin/bash
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
moodisk:x:1000:1000:Teacher Wang:/home/moodisk:/bin/bash
```

其中,mail、www-data 和 backup 是系统用户,都是在安装软件时自动创建的,它们的登录 Shell 都是/usr/sbin/nologin,执行这个程序就是拒绝登录。

3.2.2 组的概念

登记一个组群需要提供三个信息:组群名、组群 ID 和该组群的成员用户,这些信息主要保存在文件/etc/group 中,每一行对应一个组群,组群名必须唯一。文件/etc/group 中的一行格式如下:

组群名:密码:组号:该组的用户成员

密码是一个 x,加密后的密码存放在/etc/gshadow 文件中,对于组群的密码,没有多大意义。组号为 0 表示超级用户组群,组号 1～999 表示系统组群,1000 及以后的数字表示普通组群。组号允许不唯一,也即多个组群名允许拥有相同的组号。

例如/etc/group 文件中有一行:

```
admin:x:121:john,zsan,alice
```

表示组名为 admin,组号为 121,组中的成员包括用户 john、zsan、alice。如果组 x 是用户 y 的主要组群,那么 y 不会列在 x 的成员列表中,也即组的成员列表只是以此组为附加组群的用户,如上面的例子,admin 是 john、zsan 和 alice 三个用户的附加组群。

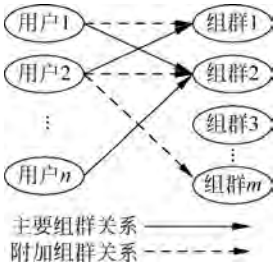

图 3.4 用户和组群的关系

用户和组群的关系可以用图 3.4 表示。

从图 3.4 可以看出,一个用户必须且只能属于一个主要组群,但是可以属于 0 个或者多个附加组群,一个组群可以包含 0 个或者多个用户。类似在校大学生,一个学生只能属于一个班级(类似主要组群),但是他可以同时加入很多社团(类似附加组群)。

3.3 用户和组管理

用户和组的管理包括创建、删除、修改属性、修改密码等。具体操作可以采用可视化的图形界面方式,也可以采用命令方式,这里重点介绍后者。Linux 下的命令语法大致如下:

```
<命令> <选项> <目标>
```

“命令”是用来操纵“目标”的,到底怎么操纵呢? 这是由“选项”规定的。由于可能存在多个选项,因此选项采用“-<一个字母或数字> [参数]”或者“--<多个字母或数字>[参数]”的形式,例如下面创建用户 alice 的命令:

```
useradd -u 1001 -d /home/zbc --shell /bin/bash alice
```

其中,命令是 useradd,目标是 alice,选项是“-u 1001 -d /home/zbc --shell /bin/bash”。命令一般都是某个程序对应的文件名。命令、选项、目标以及各个参数之间用空格符分开,但允许出现多个空格。例如下面两个命令是等价的:

```
useradd -u 1002 john
useradd          -u  1002                                    john
```

文件/etc/login.defs 和/etc/default/useradd 定义了组群和用户的默认属性,在创建用户和组的时候,如果没有给出相应的参数,那么就取默认值。其中一些主要参数如表 3.1 所示。

表 3.1 文件/etc/login.defs 定义的参数

序号	参 数	默 认 值	说 明
1	PASS_MAX_DAYS	99999	密码有效期最多天数
2	PASS_MIN_DAYS	0	密码有效期最少天数
3	PASS_WARN_AGE	7	密码警告时间段,密码到期前 7 天开始发警告
4	UID_MIN	1000	普通用户 UID 的最小值
5	UID_MAX	60000	普通用户 UID 的最大值
6	GID_MIN	1000	普通组 GID 的最小值
7	GID_MAX	60000	普通组 GID 的最大值
8	CREATE_HOME	yes(红帽),no(Debian)	是否创建家目录

而/etc/default/useradd 文件,不同的 Linux 发行版本会有所不同,主要定义默认的登录 Shell、家目录的父目录以及用户的模板目录(即 skel),模板目录中包含一些用户登录/登出的配置文件。主流 Linux 发行版的模板目录都是/etc/skel,创建用户时,会把模板目录复制成家目录。

在图形桌面上可以采用文本编辑器浏览此文件内容,启动文本编辑器的方法是:单击左上角的“活动”,再单击图标(即“显示应用程序”),然后找到并单击“文本编辑器”启动它,最后打开需要浏览的文件。

注意:下面的操作除特别声明外,都需要超级用户权限,在具体实验时最好用 root 用户登录,或者切换到 root 用户。需要注意的是这些例子只是案例教学,并不代表在实际的操作过程中一定要这样做,例如创建组的第一条命令 groupadd class1 用于创建一个新组 class1,也可以创建其他的组,例如 sales、itx86 组等。

参考图 2.9 打开一个命令窗口,然后在命令窗口中输入 su -可切换到超级用户,以后执行组和用户的命令都在超级用户的命令窗口中输入即可,参考图 3.5。命令 sudo -s 也可以切换到超级用户状态。

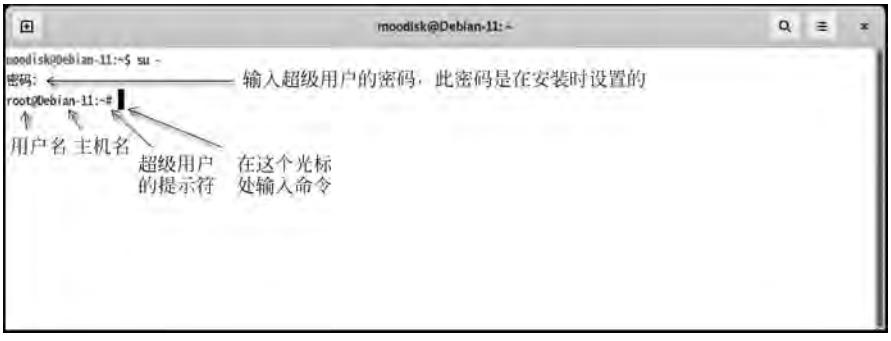

图 3.5 切换到超级用户命令窗口

3.3.1 组管理

1. 创建组

创建组的命令是 groupadd,使用语法是:

```
groupadd      [选项]      组名
-g,--gid GID  定义组号
-o,--non-unique 允许组号不唯一
```

在语法中,出现在中括号内的内容表示可以省略,单字符参数-g 和多字符参数--gid 是等价的,使用任何一个都行,-o 和--non-unique 也是等价的,以后的命令语法都是一样的理解。

(1) 创建组群 class1(参考图 3.6)。

图 3.6 创建组群

为了节约篇幅,以后只列出命令。

(2) 创建组群 grade2 且指定 GID 为 555:

```
groupadd -g 555 grade2
```

(3) 创建已存在组群 root 的别名组群 administrators:

```
groupadd -g 0 -o administrators
```

新建的组信息保存在/etc/group 中,可以采用 more 或者 cat 命令查看,例如:

```
cat /etc/group
```

2. 删除组群

删除组群 class1:

```
groupdel class1
```

注意:只能删除已经存在的空组群,空组群也就是组里没有用户成员。

3. 修改组群属性

修改组的命令是 groupmod,使用语法是:

```
groupmod      [选项]      组名
-g,--gid GID  定义组号
-o,--non-unique 允许组号不唯一
-n, --new-name NEW_NAME 修改组名
```

(1) 修改组群 sales 的组号(GID)为 1650：

```
groupmod -g 1650 sales
```

(2) 修改组群 sales 的组群名为 sales1：

```
groupmod -n sales1 sales
```

(3) 用一条命令完成上面两个任务：

```
groupmod -g 1650 -n sales1 sales
```

注意：只能修改已经存在的组群属性。

4. 查看组群信息

查看文件/etc/group 的内容即可，可以采用 cat、more、tail 等命令查看。

(1) 查看文件/etc/group 的末尾 10 行：

```
tail /etc/group
```

(2) 翻页显示/etc/group 信息：

```
more /etc/group
```

按 Space 键查看下一页，直到结束，如果想中途退出 more 命令，按 Q 键即可。

3.3.2 用户管理

1. 创建用户

创建用户的命令是 useradd，它的语法如下：

useradd [选项] 用户名

```
--badname 创建非标用户，即允许用户名中出现中文、空格、*等非字母数字字符
-c, --comment COMMENT  备注
-d, --home-dir HOME_DIR 家目录
-g, --gid GROUP  主要组群
-G, --groups GROUP1[,GROUP2,...]  若干附加组群
-m, --create-home  创建家目录，Debian默认是不创建的
-M, --no-create-home  不创建家目录，红帽默认是创建的
-p, --password PASSWORD  密码
-r, --system  创建系统用户。系统用户没有家目录，且id小于1000
-s, --shell SHELL  登录Shell
-u, --uid UID  指定用户ID
```

关于一些不太常用的参数，可以查看在线文档，即执行命令 man useradd 可获得 useradd 的在线文档。下面举例说明。

(1) 创建用户 alice：

```
useradd alice
```

没有选项，全部采用默认值：UID 取最小的可用的普通用户号，家目录为/home/alice，登录 Shell 为/bin/bash(红帽)或者/bin/sh(Debian)，归属主要组群 alice(组群 alice 会自动创建)。但是在 Debian 11，不会创建家目录，而在红帽系统中，会创建家目录。

(2) 创建用户 john,用户 ID 为 1688,-m 参数表明要创建家目录:

```
useradd -u 1688 -m john
```

(3) 指明用户 id、家目录、主要组群和登录 Shell 程序:

```
useradd -u 1005 -d /home/lisihome -g sales -s /bin/bash lisi
```

众多参数之间的前后次序没有关系,但是参数和值必须毗连,例如参数-u 后面的 1005 就不能放到其他地方。因此上面的命令与下面的命令等价:

```
useradd --home-dir /home/lisihome -s /bin/bash -g sales -u 1005 lisi
```

由 useradd 的语法可知,单字符参数-d 与多字符参数--home-dir 是等价的。

(4) 创建用户 limusheng,加入两个附加组群 mail 和 class2:

```
useradd -c "这是李木生的账号" -G mail,class2 limusheng
```

注意:mail、class2 是参数-G 的值,里面不能出现空格,而备注里面允许出现空格,但必须用单引号或双引号括起来。例如 useradd -c "hello Mr. Li" lisi。

(5) 创建超级用户 root 的别名,以后用 Administrator 登录,权限与 root 一样:

```
useradd -u 0 -o Administrator
```

(6) 创建非标用户 Mr. Wang&:

```
useradd --badname -d /home/mrwang -m "Mr.Wang&"
```

用户名中出现了.、空格和 & 字符。非标用户唯一的好处是别人不容易猜出用户名,从而提高了安全性。

2. 删除用户

删除用户采用 userdel 命令,其语法如下:

```
userdel    [选项]    用户名

-f, --force   强制删除用户(即使已经登录),包括家目录、邮件以及同名组群
-r, --remove  删除用户和他的家目录、邮件目录
```

如果省略参数,那么只删除用户。下面举例说明。

(1) 删除 alice,其家目录、邮件目录等都保留:

```
userdel alice
```

(2) 彻底删除 john,即使他已经登录:

```
userdel -f john
```

上述命令不会删除他保存在其他目录中的文件。已经登录的用户不会马上被踢出。

(3) 删除非标用户 Mr. Wang&,如果此用户目前已经登录,就不删:

```
userdel --remove "Mr.Wang&"
```

3. 修改用户

采用 usermod 命令修改用户，其语法为：

```
usermod [选项] 用户名

-a, --append 额外加入到由-G参数指定的组群中
--badname 非标用户，即允许用户名中出现中文、空格、*等非字母数字字符
-c, --comment COMMENT 备注
-d, --home-dir HOME_DIR 新家目录
-g, --gid GROUP 新的主要组群
-G, --groups GROUP1[,GROUP2,...] 若干附加组群
-l, --login NEW_LOGIN 新的用户名
-m, --move-home 旧家目录移到新家目录（由-d参数指定）
-p, --password PASSWORD 密码
-s, --shell SHELL 新的登录Shell
-u, --uid UID 新用户ID
```

其他一些不常用的参数请参考在线文档。

注意：使用-a 参数时，必须同时使用-G 参数指定额外的附加组群，类似的，使用-m 参数时必须同时使用-d 参数。下面举例说明。

(1) 把 alice 的 id 号改为 1234：

```
usermod -u 1234 alice
```

(2) 修改 john 的家目录和登录 Shell 程序，同时旧家目录中的文件被移到新家目录中：

```
usermod -d /opt/zsan -s /bin/sh -m john
```

(3) 主要组群改为 grade1，同时额外加入 class2 附加组群，之前的附加组群继续保留：

```
usermod -a -g grade1 -G class2 lisi
```

(4) alice 改名为非标用户 Hello World：

```
usermod --badname -login "Hello World" alice
```

4. 修改用户密码

采用 passwd 命令修改用户的密码，它的语法如下：

```
passwd [选项] [用户名]

-d, --delete 删除密码，被删除密码的用户不能再登录了
-e, --expire 使密码失效，下次登录时被强制要求修改密码
-l, --lock 锁住密码，以后不能使用密码登录，但允许以其他方式登录，如公钥
-u, --unlock 解锁密码
-S, --status 查看密码状态
-n, --mindays MIN_DAYS 密码有效期最少天数
-x, --maxdays MAX_DAYS 密码有效期最多天数
-w, --warndays WARN_DAYS 密码警告时间段
-i, --inactive INACTIVE 密码禁用期
```

超级用户 root 能修改任何用户的密码，而普通用户只能修改自己的密码。下面举例说明。

(1) 修改自己的密码：

```
passwd
```

如果是 root 用户，不用输入旧密码，普通用户必须先输入旧的密码。

注意：输入密码时，屏幕上没有任何显示。

(2) 修改 alice 的密码，只有 root 用户才能修改其他用户的密码：

```
passwd alice
```

(3) 锁住 john 的密码：

```
passwd --lock john
```

锁住密码，并不代表锁住用户，此用户还可以使用其他方式登录，例如采用 SSH 的公钥远程登录。

(4) 查看用户 moodisk 的密码状态：

```
passwd -S moodisk
```

显示的状态格式类：

```
moodisk P 01/12/2022 0 99999 7 -1
```

上述用空格分开的七部分分别表示：用户名为 moodisk，密码状态为 P(P 表示密码有效，L 表示密码被锁，NP 表示无密码)，最近修改密码的日期为 01/12/2022，密码有效期最少天数为 0，密码有效期最多天数为 99999，密码警告时间段为 7，密码禁用期为－1。关于密码老化过程请参考图 3.2。

(5) 修改用户 zsan1 的密码老化时间：

```
passwd -n 10 -x 20 -w 3 -i 5 zsan1
```

上述表明密码有效期最少天数为 10 天，最大天数为 20 天，过期前 3 天会发警告通知，密码禁用期为 5 天。

3.4 登录过程和环境变量

这里着重讲解字符界面的用户登录过程。

3.4.1 用户登录过程

图 3.7 完整地描述了用户登录过程，图中的 F*N* 和 tty*N* 中的 *N* 为 2～6 的一个数字，表示从第 *N* 个虚拟屏幕登录系统，例如同时按下 Ctrl、Alt 和 F3 三个键切换到第三个虚屏，另外图中的< user >代表输入的用户名，“～”表示用户的家目录。

从图 3.7 中可以发现用户成功登录后，最后自动执行登录 Shell 程序(例如/bin/bash)，此后 Bash 进程显示命令行提示符＃或＄(超级用户的命令提示符是＃，普通用户是＄)，并等待用户输入命令。一旦用户输完命令并按 Enter 键后，Bash 读取输入信息并执行用户输入的命令，命令执行完毕后又显示命令行提示符，等待用户输入下一个命令，直到用户输入命令 exit 退出系统为止。

如果要让所有用户登录时都执行一些代码，就把这些代码写入一个以 sh 为扩展名的文件中，并放在/etc/profile.d 目录下。相反如果只让某个用户登录时执行，那么就直接加在

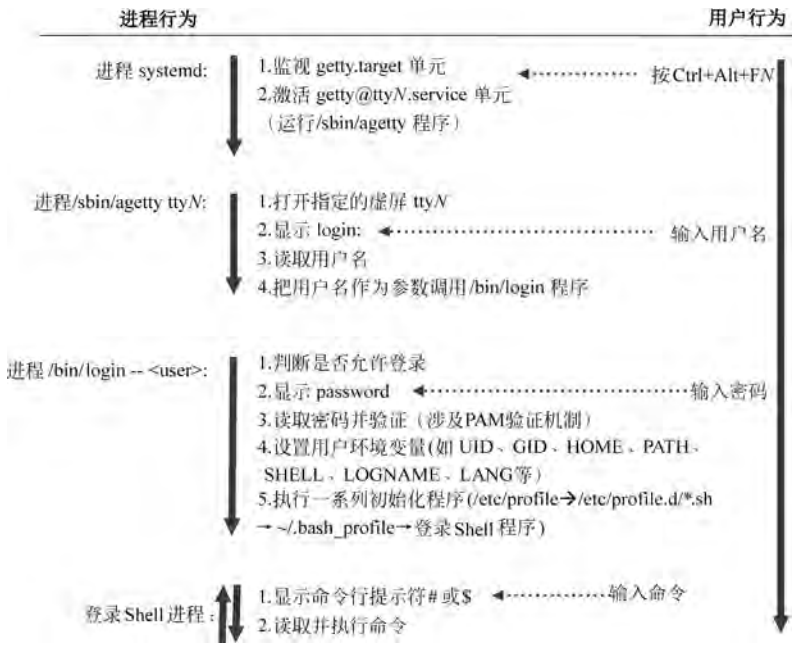

图 3.7　用户登录过程

此用户家目录中的.bash_profile 文件中(即～/.bash_profile)。

3.4.2　用户环境变量

用户登录 Linux 系统时，操作系统会自动为他配置好工作环境——语言、家目录、邮箱目录、命令搜索路径、终端类型、用户名、命令提示符等。用户的工作环境由一系列的环境变量定义，环境变量的格式如下：

环境变量名 = 值

“环境变量名”由大小写字母、_、数字组成，以字母开头，但是建议一般用大写字母，如 LOGNAME、HOME 等，“值”可以由任意字符组成，如果包含空格，则要用引号括起来。表 3.2 所示是一些常见的用户环境变量。

表 3.2　常见的用户环境变量

序号	环 境 变 量	说　　明
1	LANG=zh_CN.UTF-8	语言定义为中文 UTF-8
2	HOME=/home/zsan	用户家目录是/home/zsan
3	LOGNAME=zsan	用户名为 zsan
4	PATH =/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin	定义命令搜索路径，即 Bash 在这些路径中查找用户输入的外部命令所对应的程序，然后执行这个程序(关于内外部命令请参考第 5 章)
5	SHELL=/bin/bash	用户登录 Shell 为/bin/bash
6	PWD=/home/zsan	动态跟踪用户的当前目录，如果用户改变目录，那么这个变量的值也发生相应的改变

显示用户环境变量采用命令 env 或者 echo $环境变量名,前者显示全部的用户环境变量,后者显示一个特定的环境变量的值,如 echo $HOME;设置用户环境变量采用命令 export:

```
export 变量名 = 值
```

(1) 定义语言为英语:

```
export LANG = C
```

(2) 定义新变量 HELLO:

```
export HELLO = "Hello World!"
```

(3) 重新定义变量 PATH:

```
export PATH = $PATH:/opt/chrome/bin
```

本例中变量的值又引用了其他一个变量,即采用 $XXX 的形式引用变量 XXX 的值。删除用户环境变量采用命令 unset,如:

```
unset HELLO
```

与用户环境变量类似的概念是 Shell 变量,每个登录 Shell 都拥有一套默认的 Shell 变量集,而用户环境变量是采用 export 命令导出的 Shell 变量,是 Shell 变量的子集。Shell 变量直接采用赋值的形式(如 xyz=123)或在前面加上 set(如 set xyz=123),使用命令 unset 删除。一些常见的 Shell 变量如表 3.3 所示。

表 3.3 常见的 Shell 变量

序号	Shell 变 量	说 明
1	PS1,PS2,PS3,PS4	这 4 个 Shell 变量定义了命令行提示符号格式
2	HISTFILE=/root/.bash_history	定义记录历史命令的文件
3	HISTFILESIZE=1000	记录最近输入的 1000 条历史命令,这些命令保存在由 HISTFILE 定义的文件中。如果为 0 则表示不记录历史命令,为-1 则表示记录全部的历史命名
4	HISTTIMEFORMAT='%F %T '	同时记录历史命名执行的时间
5	HOSTTYPE=x86_64	定义了 CPU 架构
6	IFS=$'\t\n'	定义分隔符为空格和 Tab 键
7	LINES=30,COLUMNS=114	定义了字符屏幕敞口的大小(行列数)
8	MACHTYPE=x86_64-redhat-linux-gnu	定义了机器类型
9	HOSTNAME=Debian-11	定义了主机名
10	EUID=1000	有效用户 ID 号,EUID 有别于 UID,EUID 用于权限控制

直接运行命令 set 显示所有的 Shell 变量。例如下面的命令修改 HISTFILESIZE 变量的值为 5000,表明要记录最近输入的 5000 个历史命令:

```
set HISTFILESIZE = 5000
```

再如定义 pi 的命令：

```
pi = 3.14159
```

删除 ABC 变量的命令：

```
unset ABC
```

注意：使用 export 和 set 命令定义的变量是临时性的，在用户注销或者重启计算机后就没有了。如果希望定义的变量永久生效，那么就要把定义变量的命令加到用户登录时自动执行的脚本程序中(参见图 3.7)，通常选择加在～/.bash_profile 文件的末尾(对特定用户起作用)，如果要对全部的用户起作用，就在/etc/profile.d/目录中增加一个扩展名为 sh 的文件。例如用文本编辑器编辑新文件/etc/profile.d/alls.sh，文件内容为：

```
export HISTFILESIZE = 5000
```

这样每一个用户登录时都会执行/etc/profile.d/alls.sh，所以此后登录的用户都具有用户环境变量 HISTFILESIZE，值是 5000。采用下面的命令可以查看此用户环境变量的值。

```
echo $HISTFILESIZE
```

另一个类似的变量是 HISTSIZE，它定义运行 history 命令显示的历史命令条数，例如 HISTSIZE＝20，那么运行下面的命令就把记录历史命令的文件的末尾 20 行显示出来。

```
history
```

如果使生产环境，最好设置 HISTTIMEFORMAT＝'%F %T '，以便同时记录每条命令执行的日期和时间，这样出问题时可以追溯责任。

3.4.3 用户切换

用户登录后可以切换到另一个用户，普通用户需要知道被切换用户的密码，而超级用户不要密码就可以切换到任何用户。切换用户的命令有 sudo、su。

1. sudo 命令

以另一个用户身份执行命令，默认是 root，假设目前以普通用户登录系统，普通用户没有权限删除其他用户，所以需要以 root 身份执行删除用户的命令，如下所示：

```
sudo userdel alice
```

会提示输入当前普通用户的密码。

以 moodisk 用户身份执行命令：

```
sudo -u moodisk ls /home/moodisk
```

切换到 root 用户(此后可采用 exit 命令退出 root 用户)：

```
sudo -s
```

为了系统安全，平时一般以普通用户登录系统完成日常工作，当需要超级用户权限执行

某个命令时,采用 sudo 命令临时获取 root 权限,这是个好习惯。

注意:用户只有属于 sudo 组群(Debian 操作系统)或 wheel 组群(红帽操作系统)时,才能使用 sudo 命令。例如下面的命令把 alice 加入 sudo 组群:

```
usermod -a -G sudo alice
```

2. su 命令

su 命令类似与上面的 sudo -s 命令,但是 su 有两种切换方式:一是做彻底切换,把全部环境变量改为切换后的用户的环境变量;二是修改少量环境变量。现举例如下。

彻底切换到 root 用户:

```
su -root
```

上面的命令与 su -等价。下面的命令不彻底切换到 moodisk 用户:

```
su moodisk
```

需要注意的是 su 切换用户时需要输入被切换用户的密码,而 sudo 需要输入当前用户的密码,如图 3.8 所示。

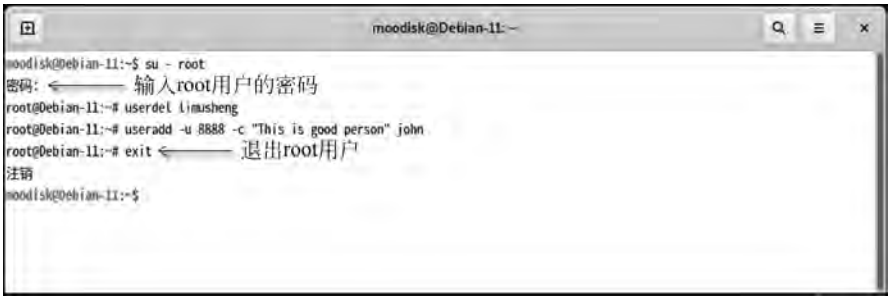

图 3.8 切换到 root 用户

课堂实操

创建组 lessen。新建用户 jack,要求主要组群是 lessen,附加组群为 sudo(如果是 Debian)或者 wheel(如果是红帽),同时创建家目录。给 jack 用户设置密码 A1b2C3。在第 4 个虚屏上登录 jack 用户。查看环境变量有哪些,各有什么值。切换到 root 用户。从 root 用户退出。

3.5 知识拓展和作业

3.5.1 知识拓展

(1) 了解身份鉴别机制:PAM。

(2) 掌握实际用户号 UID 和有效用户号 EUID 的区别。

(3) 了解 sudo 插件。

3.5.2 作业

(1) 创建用户 wang,该用户具有如下属性:家目录为/var/home/wang,登录 Shell 是/bin/sh,归属主要组群为 mail,同时属于附加组群 users 和 fuse 的成员,初始密码为123456,并在用户首次登录时提示修改密码,要求该用户每隔 90 天修改一次密码。请写出命令序列。

(2) 假设用户 zsan 已经存在,家目录是/home/zsan,登录 shell 是/bin/bash,请帮助该用户设置永久有效的用户环境变量(其他用户不受影响):

```
HELLO = "I am fine"
NAME = "Zhan san"
```

(3) 文件/etc/passwd 中的一行信息如下:

```
jack:x:501:1001:The superman:/home/jackhome:/bin/bash
```

请解释各个字段的含义。

(4) 文件/etc/shadow 中有如下一行:

```
woman: $6$87wjcyRCGHJ2rPOb.SQw:15142:10:20:3:5:16253:
```

请解释各个字段的含义。