

# 实验 7 Linux中的用户管理

## 实验目的

掌握Linux中管理用户和组的基本命令。

## 实验内容

- 1.添加新的用户账号
- 2.删除帐号
- 3.修改帐号
- 4.用户口令的管理
- 5.增加一个新的用户组
- 6.删除一个已有的用户组
- 7.修改用户组的属性
- 8.用户在用户组之间切换
- 9.批量添加用户

## 实验前提

请先克隆 ubuntu server 1604 虚拟机并使用克隆版ubuntu server 进行实验。以防原ubuntu linux 失效。

## 实验步骤

### 一.添加新的用户账号

- 1.使用useradd命令添加用户账户。

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 /home/sam（/home为默认的用户主目录所在的父目录）。

```
useradd -d /home/sam -m sam
```

2.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。

```
# useradd -s /bin/sh -g group -G adm,root gem
```

## 二.删除帐号

使用下列命令删除刚才建立的用户账户**sam**。

```
userdel -r 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。

## 三.修改帐号

使用下列命令将用户**sam**（若无此账户，参考实验内容1建立）的登录Shell修改为**ksh**，主目录改为**/home/z**，用户组改为**developer**。

```
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**。

## 四.用户口令的管理

1.使用下列命令将用户**sam**（若无此账户，参考实验内容1建立）修改该用户的口令：

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

2.将账户**sam**的口令设置为空。

```
# passwd -d sam
<p class="mume-header " id="passwd-d-sam"></p>
```

此命令将用户 **sam** 的口令删除，这样用户 **sam** 下一次登录时，系统就不再允许该用户登录了。

3.使用下列命令锁定账户。

```
# passwd -l sam
<p class="mume-header " id="passwd-l-sam"></p>
```

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

## 五.增加一个新的用户组

1.使用下列命令建立用户组group。

```
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。

2.使用下列命令建立developer用户组。

```
groupadd -g 1001 developer
```

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

## 六.删除一个已有的用户组

使用下列命令删除group1

```
groupdel group1，此命令从系统中删除组group1。
```

## 七.修改用户组的属性

1.使用下列命令将组group2的组标识号修改为102。

```
groupmod -g 1002 group2
```

2.使用下列命令将组group2的标识号改为10000，组名修改为group3

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

注：上述组若不存在，则请自行增加组。

## 八.用户在用户组之间切换

使用命令将当前用户切换到root组

```
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。

## 九.批量添加用户

1.先编辑一个有10行的文本用户文件。每一行按照/etc/passwd密码文件的格式书写，要注意每个用户的用户名、UID、宿主目录都不可以相同，其中密码栏可以留做空白或输入x号。

范例文件user.txt内容如下：

```
user001::600:100:user:/home/user001:/bin/bash
user002::601:100:user:/home/user002:/bin/bash
user003::602:100:user:/home/user003:/bin/bash
user004::603:100:user:/home/user004:/bin/bash
user005::604:100:user:/home/user005:/bin/bash
user006::605:100:user:/home/user006:/bin/bash
```

注意：在Debian Linux/Ubuntu中可能会存在一个Bug，即无法使用newusers批量增加用户，有时会报出“Error in `newusers': free(): invalid next size (fast)”等错误。纠正方法暂时未知，但如果将上述文件中“user00\*.”后加上密码，例如“123456”，可能不报出错误。例如先在文本文件中写两个用户：

```
user001:123456:600:100:user:/home/user001:/bin/bash
user002:123456:601:100:user:/home/user002:/bin/bash
```

2.以root身份执行命令 /usr/sbin/newusers，从刚创建的用户文件user.txt中导入数据，创建用户：

```
newusers < user.txt
```

3.执行命令/usr/sbin/pwunconv。

将 /etc/shadow 产生的 shadow 密码解码，然后回写到 /etc/passwd 中，并将/etc/shadow的shadow密码栏删掉。这是为了方便下一步的密码转换工作，即先取消 shadow password 功能。

```
pwunconv
```

4.编辑每个用户的密码对照文件。

范例文件 passwd.txt 内容如下:

```
user001:密码
user002:密码
user003:密码
user004:密码
user005:密码
user006:密码
```

5.以root身份执行命令/usr/sbin/chpasswd

创建用户密码，chpasswd 会将经过 /usr/bin/passwd 命令编码过的密码写入 /etc/passwd 的密码栏。

```
chpasswd < passwd.txt
```

6.确定密码经编码写入/etc/passwd的密码栏后。

执行命令 /usr/sbin/pwconv 将密码编码为 shadow password，并将结果写入 /etc/shadow。

```
pwconv
```

这样就完成了大量用户的创建了，之后您可以到/home下检查这些用户宿主目录的权限设置是否都正确，并登录验证用户密码是否正确。