第四课 用户管理和软件安装

# 用户和权限

系统通过用户和密码识别用户权限

用户和组帐号概述

* Linux基于用户身份对资源访问进行控制
  + 用户帐号：
    - 超级用户root (0)
    - 普通用户 (500--)
    - 程序用户 (1-499)
  + 组帐号：
    - 基本组(私有组) (唯一性，用户只能加入一个基本组，而且必须要加入)
    - 附加组（公共组）（任意性）
  + UID和GID：
    - UID（User Identity，用户标识号）
    - GID（Group Identify，组标识号）

Linux基于用户的UID数字识别用户

UID 用户

0 代表管理员

1—499 程序用户（一般不需管理，安装软件包时会自动创建，不用于系统登录，只用于管理某个服务

>=500 代表普通用户（管理员创建的可以分配给其他用户以登录系统的）

7系统中，0为管理员，1—499为系统用户，500—999为程序用户，1000及以上的普通用户

GUI登录界面，只显示普通用户，其他用户不显示，需要手工输入

组：分为基本组和附加组

把用户添加到组时，必须有且只能有一个基本组（唯一性），可以有任意个附加组（任意性）

基本组的属性会捆绑到用户，用户创建文件时，默认用户为拥有组，其基本组为拥有组。建议人员少的为基本组，多的为附加组，以强化安全性。

* useradd命令
  + 格式：useradd [选项]... 用户名
* 常用命令选项
  + -u：指定 UID 标记号（默认按系统最大的递增）
  + -d：指定宿主目录，缺省为 /home/用户名
  + -e：指定帐号失效时间,（默认永不过期）
  + -g：指定用户的基本组名（或UID号）（默认为用户同名的组）
  + -G：指定用户的附加组名（或GID号）（默认无）
  + -M：不为用户建立并初始化宿主目录
  + -s：指定用户的登录Shell（用户通过shell控制服务器，服务器通过shell反馈结果给用户，cat /etc/shells可以查看到系统支持的shell，/bin/bash是linux中默认使用的shell。

/sbin/nologin为非交互式shell，使用这种shell，用户登录后会自动退出，用于系统中的大部分程序用户及远程访问本地服务的用户（该部分用户不需要使用ssh本地登录，如ftp服务的用户）

实例：

useradd zhangsan

useradd -u 1000 -g bin -G mail -s /sbin/nologin -d /lisi -e 2018-10-1 lisi

-g -G指定组名必须是已存在的组。

id zhangsan显示用户基本信息

默认，一个用户的UID是唯一的

cat /etc/passwd 可以查看所有用户信息

man 5 passwd可以查看passwd这个配置文件的详细说明

* userdel命令
  + 格式：userdel [-r] 用户名
  + 添加 -r 选项时，表示连用户的宿主目录和邮件文件一并删除

/etc/shadow 这个是记录了系统中所有账户的密码信息，该文件以:为分隔符，划分了九段内容，每段可以参考man 5 shadow有详解

* groupadd命令
  + 格式：groupadd [-g GID] 组帐号名
* groupdel命令
  + 格式：groupdel 组帐号名，如有用户以该组为基本组，则不能删除

Linux中只有root是管理员，其他用户添加到root组，只能查看root组对应的文件，很多操作权限还是没有（如修改系统IP），如需要添加多个管理员，可以创建一个用户，设置密码，然后把用户的UID改为0即可，此时用户登录后显示为root。

UID原则上是唯一的，无法使用命令修改，需要直接编辑/etc/passwd配置文件（每行中的第二列）

权限管理

* 访问权限（对文件只有三个权限）
  + 读取r：允许查看文件内容、显示目录列表
  + 写入w：允许修改文件内容，允许在目录中新建、移动、删除文件或子目录
  + 可执行x：允许运行程序、切换目录。x权限一般只有文件夹才具备，可以使用cd命令或者鼠标双击打开文件夹，如果没有x权限直接报错，文件不需要x权限，只有把文件当做程序运行（脚本）才需要x权限
* 归属（所有权）
  + 属主（UID）：拥有该文件或目录的用户帐号
  + 属组（GID）：拥有该文件或目录的组帐号

ls -l

可以看到文件的UID和GID。如用户UID与文件UID同，则是文件的属主(owner)；否则，如用户GID（可能有多个）包含了文件的GID，则该用户归属于文件的属组(group)；否则为其他人（other）。

ls -l 显示的前10个字符分别为

权限可使用3位8进制数表示ugo三种用户的权限，r(4),w(2),x(1)，

touch test.txt 创建测试文件

ls test.txt

chmod 777 test.txt

ls test.txt

chmod 000 test.txt

ls test.txt

chmod 123 test.txt

ls test.txt

chmod 456 test.txt

ls test.txt

chmod u+w test.txt

ls test.txt

chmod g-x test.txt

ls test.txt

chmod o= test.txt

ls test.txt

umask 查看当前用户反掩码，当用户创建文件或文件夹的时候默认以最大权限减去umask数值就得到新建文件或文件夹的权限，文件夹最大权限是777 ，文件则是666

chown 修改拥有者或拥有组，具体格式：

1.chown 用户名 文件或文件夹 修改属主

2.chown :组名 文件或文件夹 修改属组

3.chown 用户名:组名 文件或文件夹 同时修改拥有者和拥有组

chmod和chown都支持-R选项，文件夹中递归，不仅会修改文件夹自身的属性同时也会将所有子目录或文件全部修改

软件包安装

rpm 红帽linux提供的包管理，把编译的工作已预先做好，而不是使用源码包进行配置、编译和安装。安装只需要一条命令即可。其他类型的linux也多数支持rpm

安装光盘中包含了绝大多数的RPM包。

mkdir /media/cdrom

mount /dev/cdrom /media/cdrom 挂载光驱到指定目录

ls /media/cdrom/Packages 该目录是光盘中所有的RPM包的位置

rpm -q vsftpd 查询vsftpd软件是否已安装

rpm -ivh /media/cdrom/Packages/vsftpd-\* 安装软件包

rpm -q vsftpd

rpm -e vsftpd 卸载软件包

RPM包安装过程中可能会涉及依赖关系，被依赖的软件包需要先安装，如果依赖关系比较复杂，安装软件变得非常困难，可能需要花费大量时间在寻找依赖关系上。yum可以自动帮助用户解决依赖关系

CentOS系统默认情况下yum可用，会通过互联网连接CentOS官方网站下载软件更新，也可以换其他更新源，如果服务器不能上网

本地更新源配置：

mkdir /etc/yum.repos.d/bak 创建一个目录放置以前更新源

mv /etc/yum.repos.d/Cent\* /etc/yum.repos.d/bak 移动以前更新源文件至备份目录下

自己创建本地更新源文件

vim /etc/yum.repos.d/local.repo 位置固定，名称任意，必须以.repo结尾

写入下面内容：

[local] 内容随意，本地记录唯一即可

name=This is my repo 描述信息，随意写

baseurl=file:///media/cdrom 定义更新源位置，光驱就是一个更新源，也可以把光盘文件复制到硬盘中，或使用ftp、http等协议指定源

enabled=1 是否启用该记录，1为启用0为禁用

gpgcheck=0 是否验证签名，0为禁用，1为启用，如果是1的话需要再定义gpgkey

yum install gcc-c++ 可以自动解决依赖关系

cat /etc/yum.conf 记录了yum主配置文件，可以看到更新源应该放置的位置，最后两行