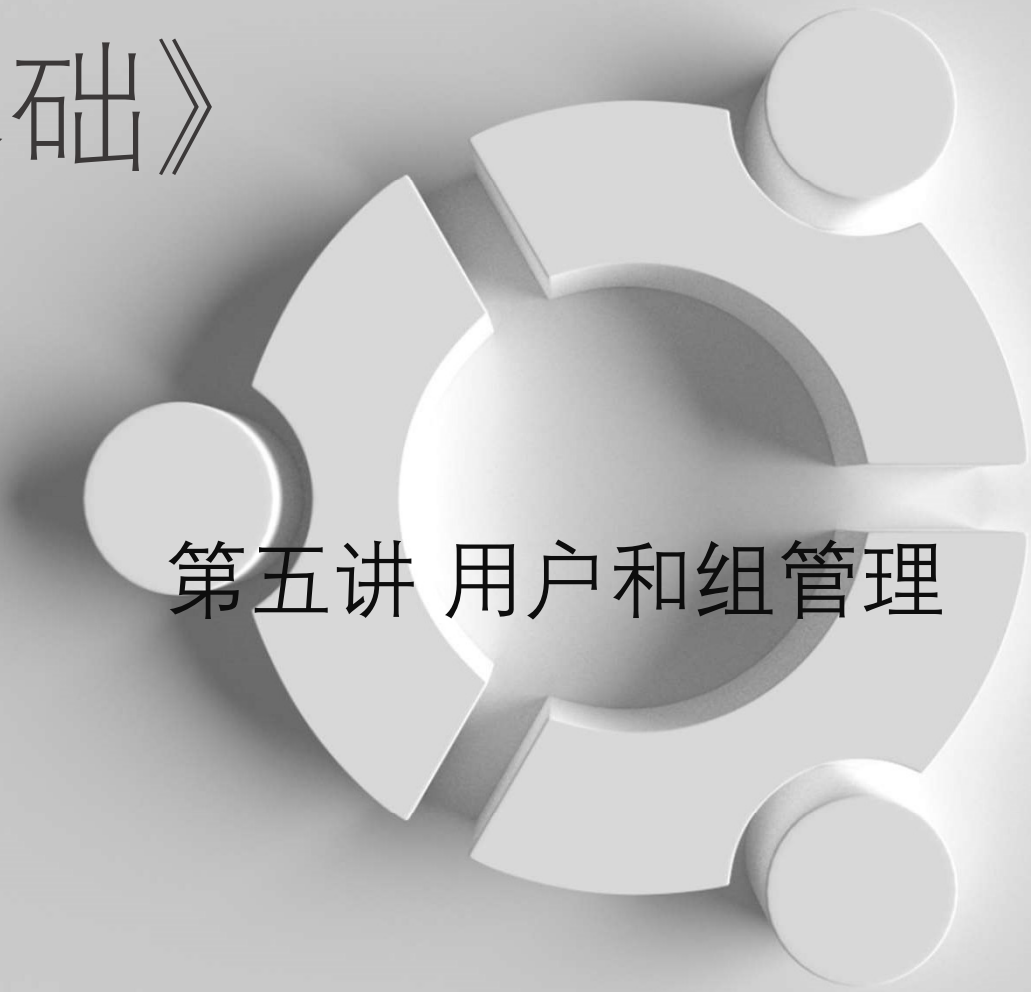
河北师范大学 软件学院
Software College of Hebei Normal University

《Linux基础》



第五讲 用户和组管理

用户与权限的问题

- 可读写权限在Windows上是一个经常被忽略的问题。
- Windows对使用者的影响：Windows的设计使得用户并不关心文件的所有者，文件权限等问题。使用者也很少涉及到文件权限带来的问题，而事实上糟糕的设计会带来很多安全问题，尤其在部署开发软件的时候，通常并不需要考虑目录以及文件的可写权限，程序运行直接就具备可写的权限，如果存在恶意程序或者是由于bug、被攻击等情况很容易造成数据损坏。

- 在Linux上，用户权限是一个经常遇到的问题，用户需要具备root权限才可以执行关键操作：添加/删除用户、修改密码、更改系统配置、安装/卸载软件等。而某些关键的文件，可读权限都是受限的，只有root用户才可以。

- 本次课程重点：用户与组的配置信息文件，如何管理用户与组。

用户的概述

- Linux是一个多用户多任务的系统，它基于用户身份对资源访问进行控制。

- Linux中的用户分以下三类：

  超级用户：

  root

  普通用户：

  系统安装时创建的用户及后期使用中由用户创建的用户

  系统用户：

  系统及服务运行时必须存在的用户，但与真实的普通用户有所不同，默认情况下是不能登录系统的，它们的存在主要是满足系统进程对文件属主的需求。一般用于系统服务。

用户信息文件 passwd

- 用户信息保存在/etc/passwd文件中，每一行对应一个用户的帐号记录，可以使用 cat /etc/passwd 命令查看其中保存的信息。
- 文件中每行的格式为:

  登录名:口令:用户标识号:组标识号:注释性描述:主目录:登录后启动的SHELL。

```
sshd:x:110:65534::/var/run/sshd:/usr/sbin/nologin
wy:x:1000:1000:bravewang,,,:/home/wy:/bin/bash
```

用户密码信息文件 shadow

- /etc/shadow 文件保存的是用户密码加密后的数据，每一行对应一个用户的密码记录，每个用户对应/etc/passwd中的用户，只有root权限才可以读取。
- shadow文件字段用：分隔，依次为：
  登录名
  加密的密码：！表示无密码，*表示系统用户，不能登录
  最近一次修改密码的时间：距离1970年1月1日的天数
  密码的最短有效天数：默认为0，表示无限制
  密码的最长有效天数：默认为99999
  提前多少天警告用户口令将过期：默认为7
- 此文件不要手动更改，应该由程序去操作。

```
sshd:*:17587:0:99999:7:::
wy:$6$YPGWdo05$wLTLjAflIWqwvlhNPXTk3bJwMKCgSvw5u4Jd
71:17587:0:99999:7:::
```

组的概念

- 按照不同的角度，Linux中的组可以有不同的分法。
- 第一种分为超级用户组（root group）、系统组（system group）和用户组（user group）。超级用户组是超级用户所属的组，系统组是系统用户所属的组，用户组是普通用户所属的组。
- 第二种分为基本组和附加组。用户所属组中的第一个组称为基本组，基本组在 /etc/passwd 文件中指定；用户所在的其他组为附加组，附加组在 /etc/group 文件中指定。不可以把用户从基本组中删除，但是可以从附加组中删除。一个用户可以属于多个附加组，但是一个用户只能有一个基本组。
- 第三种分为私有组和公共组。建立账户时，若没有指定账户所属的组，系统会建立一个和用户名相同的组，这个组就是私有组，这个组只容纳了一个用户。而公共组可以容纳多个用户。
- 属于多个组的用户所拥有的权限是它所在的组的权限之和。

组信息文件 group

- /etc/group 文件保存系统中所有组的信息。
- 第一个字段是组名；第二个字段是组密码，同样显示密码占位符x，真正的密码已经加密存放在 /etc/gshadow文件中；第三个字段是组标识号；第四个字段是以此组为附加组的用户列表。
- 查看用户所属的组：groups 查看当前用户所属的组；groups [user] 查看[user]所属的组；id [user] 同时查看用户信息和组信息

```
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,wy
tty:x:5:
```

root用户

- root用户具有最高权限，它的UID是0。在Ubuntu上以root用户登录时提示符会变成#，其他用户提示符是$。

- 经常有人把root和Windows下的administrator做对比，表面上看二者都是系统最高级别管理员，但是它们其实是有区别的。Windows下有SYSTEM用户，SYSTEM才是最高权限用户，但仅限系统自己使用，administrator的权限也没有SYSTEM的权限大。而Linux下的root用户可以做一切事情，甚至可以直接毁掉整个系统。

- 可以修改 /etc/passwd 文件中的uid为0，使普通用户获得和root一样的权限。

SU

- su：(switch user切换用户)，可让一个普通用户切换为超级用户或其他用户，并可临时拥有所切换用户的权限，切换时需输入要切换用户的密码；也可以让超级用户切换为普通用户，临时以低权限身份处理事务，切换时无需输入密码。
- 用法： su [选项] [用户名] 。后边不带 username 使用时，su 默认会切换到超级用户。
- 带 -，-l，--login 选项可以切换到其他用户，示例：

  su - oklinux
- su root和su一样，切换后以root身份执行命令，但当前工作目录不变。

sudo

- sudo允许程序临时以root身份运行。sudo默认是以root身份运行命令，但是使用-u [username]可以以其他用户身份运行命令。
- sudo是受限制的su，它通过一个配置文件，授权某些用户可以临时具有root用户才有的权限。（5分钟）
- sudo读取 /etc/sudoers 文件的信息以判断当前用户是否有权限运行sudo。运行sudo输入的是当前用户的密码，这样使用授权的方式杜绝了root密码的泄露，同时可以根据需要进行用户授权。

- 如果是root用户，不需要使用sudo

- 示例(获取软件更新)： sudo apt update

设置密码

- passwd用于设置用户密码： sudo passwd [username]
- 在Ubuntu上，root用户默认是没有密码的，安装过程也不会设置。如果想要设置root用户的密码，可在安装完成后，运行命令：

  sudo passwd root

- sudo -i表示以root身份登录，主目录也切换为root的主目录。为了频繁地执行某些只有超级用户才能执行的命令而不用每次输入密码，可以使用该命令。提示输入密码时该密码为当前账户的密码。没有时间限制。执行该命令后提示符变为“#”而不是“$”。想退回普通账户时可以执行“exit”或“logout”。

添加用户：adduser

- 此命令默认会创建主目录，创建的是普通用户，但是可通过选项创建不能登录的系统用户。
- 用法：adduser [--home DIR] [--shell SHELL] [--no-create-home]
  [--uid ID] [--ingroup GROUP | --gid ID]
  [--disabled-password] [--disabled-login] user
- 示例：

```
sudo adduser --shell /bin/bash oklinux //创建hellolinux用户，默认登录shell是bash

sudo adduser --shell /bin/bash ubuntu1 --gid 1001 //指定要添加的组

sudo adduser --shell /usr/sbin/nologin --no-create-home --system --disabled-password --disabled-login mysql //创建系统用户mysql
```

删除用户：deluser

- sudo deluser [username] 此操作不会删除主目录。
- sudo deluser --remove-home [username] 删除用户并删除主目录。
- sudo deluser --remove-all-files [username] 删除用户以及系统中一切属于此用户的文件。

创建组以及删除组

- sudo addgroup [--gid ID] [group]
  --gid ID 手动指定组ID
- 另一种创建组的方法： sudo adduser --group [--gid ID] [group]

- 删除组：sudo delgroup [group]

其他命令示例

- 给oklinux用户添加brave组：
  sudo usermod -G brave -a oklinux

- 从brave组中移除oklinux用户：
  sudo gpasswd -d oklinux brave

本节课任务

- 创建用户组：genius
- 创建用户：brain，要求用户属于genius用户组
- 切换到brain用户，并创建目录task
- 运行命令：id 并且把结果保存到 task/a