第十课 MYSQL

Linux中权限默认是ugo，可以满足绝大多数的用户需求，如果希望单独设置某用户的权限，只能通过修改属主和属组然后再分配ug的权限，Windows系统中默认是acl管理权限，在Linux也支持acl的方式来单独控制某个用户或组的权限

getfacl 文件或文件夹名称 可以查看某个文件或文件夹的权限设置

setfacl -m u:test:rw test.txt 为test.txt文件增加test用户读写的权限

ls -ld test.txt 当文件或文件夹设置了acl权限后，ls查看的权限将会和之前权限有所不同（前三位属主权限，中间三位是mask权限，最后三位其他人权限，而且后面不是以.结尾而是以+结尾）

getfacl 文件或文件夹名 对于ls -l权限位有+的必须使用getfacl来获取完整的权限列表

setfacl -m g:wheel:r test.txt 赋予组访问权限

setfacl -m g::rw test.txt 修改属组的权限

setfacl -x g:wheel test.txt 删除权限

如果目标是文件夹可以使用-R递归

mask 代表权限掩码，是文件或文件夹上允许出现最大权限

setfacl -m m:r test.txt 设置目标的权限最大只能是r，临时限制用户访问的最大权限

default acl（设置目录的默认权限继承，即在本目录下创建文件或文件夹时默认的ACL）

setfacl -m u:a:rwx test

setfacl -d -m u:b:rwx test 只有文件夹能够设置,default acl不对自己生效，是创建的子对象自动应用default acl的配置

setfacl -m d:u:b:rwx test 效果和-d（上一行命令）相同

acl 备份与恢复

getfacl -R test > test.acl

setfacl -Rb test 清空所有acl配置

setfacl --restore=test.acl 还原ACL配置

getfacl -R test

注意：在CentOS6系统中默认新建的分区自动挂载不支持ACL

mount **-o acl** /dev/md127p1 /mnt/data/ 手动挂载加载ACL

自动挂载方法：修改/etc/fstab文件,增加以下一行

/dev/md127p1 /mnt/data ext4 defaults,acl 0 0 增加acl支持，否则不支持setfacl命令

**MySQL数据库**：非常有名的开源数据库软件，拥有大量的著名客户（苹果、谷歌、维基百科等），MySQL被Oracle收购，面临闭源风险，大量用户都转型其他数据库，mariadb和MySQL操作基本上无区别

yum install mysql-server mysql mysql只是客户端工具，客户端安装可以执行mysql命令，mysql-server才是真正的服务端软件

service mysqld start 开启MySQL数据库软件

chkconfig mysqld on 保证开机启动

默认情况下mysql的root用户是空密码，为了数据库安全考虑，第一件事就是设置密码

mysql的root用户密码和用户是保存在mysql数据库表中的一条记录，和系统的root是不同

mysql 由于是空密码可以直接连接

mysqladmin -u root -p password 123456 设置root的密码为123456，-u指定用户名默认就是root，-p是当用户有密码需要提示用户输入密码（如果空密码也可以不需要-p），password设置密码的命令，123456是要设置的密码值，建议使用安全密码

mysql -u root -p -u指定用户名，如果是root用户可以不用-u，-p如果有密码必须使用

netstat -an | grep 3306 查看mysql数据库侦听的端口信息，默认是所有地址的3306端口，网络中是允许其他主机的客户访问

vim /etc/my.cnf 修改mysql主配置文件

bind-address=127.0.0.1 将这一行添加在[mysqld]下面任何位置新建一行

service mysqld restart

netstat -an | grep 3306 发现只侦听127.0.0.1，只允许自己访问mysql服务

数据库管理命令

mysql> show databases; 列出当前mysql服务器中有哪些数据库，命令最后要有；结尾

mysql> use mysql; 切换至mysql数据库，后续执行命令默认都是在mysql这个库中执行

mysql> show tables; 查看当前库（mysq）中有哪些表

mysql> describe user; 显示当前库中user表的结构

mysql> describe mysql.user; 显示mysql库中user表的结构（类似于Linux的绝对路径），任何库中都能执行

mysql> create database zwm; 创建以库名为zwm

mysql> show databases; 验证新增库

mysql> use zwm; 切换数据库

mysql> show tables; 查看库中的表，默认是空白的没有表信息

mysql> create table users(username char(30),password char(20)); 在当前库中创建一个名为users的表，含有两列：username字符类型最多30个，password也是字符类型20个

mysql> describe users; 查看表结构

mysql> insert into zwm.users values('zhangsan','123456'); 表中插入一行记录

mysql> insert into zwm.users values('lisi',encrypt('123456')); 插入一条记录，密码采用加密方式

mysql> select \* from zwm.users; 查看表内容

mysql> select \* from zwm.users where username='lisi'; 查询username是lisi的记录，lisi是字符类型，使用''

mysql> update zwm.users set password=encrypt('123456') where username='zhangsan'; 更新一条记录

mysql> select \* from zwm.users; 查看表内容

mysql> delete from zwm.users where username='zhangsan'; 删除username是zhangsan的记录

mysql> select \* from zwm.users; 查看表内容

mysql> drop table users; 删除表

mysql> show tables;

mysql> drop database zwm; 删除库

mysql> show databases;

mysql数据库备份：

默认所有的数据库文件放置在/var/lib/mysql文件夹下（可以看/etc/my.cnf文件有记录）

第一种备份方式可以直接复制文件夹，到时候还原文件夹即可，每个数据库就是一个文件夹

第二种备份方式可以使用mysqldump命令进行备份

mysqldump -u root -p auth > auth.sql 将auth数据库备份至当前目录下auth.sql文件中，这个文件信息就是由T\_SQL语句写的

mysqldump -u root -p --all-databases > all.sql 备份所有数据库到all.sql文件中

mysql -u root -p auth < auth.sql 还原auth数据库

mysql> grant select on auth.users to zhangsan@'localhost' identified by 'redhat'; zhangsan只能本地查询auth库

mysql> grant all on auth.\* to lisi@'localhost' identified by 'redhat'; lisi只能本地访问auth数据库的所有表

mysql> grant all on \*.\* to admin@'loclahost' identified by 'redhat'; 赋予admin用户从任何主机访问任何的所有权限

mysql> flush privileges; 刷新权限，新用户生效

mysql> show grants for admin@localhost;