1. Mysql数据库中的备份

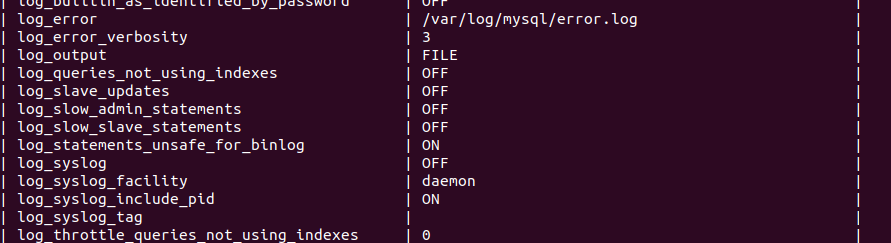
日志备份:

首先执行命令查找log文件的存储位置:

2019-04-22 09-36-21 的屏幕截图

在结果表格中找到log\_error,这里我只开启了错误日志,实际上Mysql有三种日志:general, error, query, slow query, 事务等日志.

结果如图:



从图中可以看到, 错误日志被保存了/var/log/mysql/error.log文件中. 那么对于错误日志备份,我们只需要保存相应的文件即可.这种备份方式称为冷备份,也就是说只能在系统关闭的时候不会发生数据丢失. 我们可以手动拷贝该日志文件实现备份.

数据备份

数据备份也分为热备份和冷备份, 因为数据是重要的,所以一般情况下不允许冷备份,尽量使用热备份的方式. 在本次实验中, 两种方法我均会介绍.

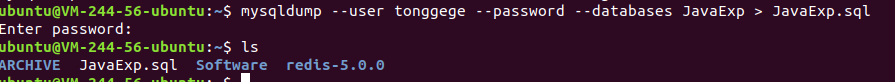
热备份的方式是使用工具mysqldump

备份所有数据库: mysqldump --all-databases > dump.sql

备份指定数据库:mysqldump --databases db1 db2 db3 > dump.sql

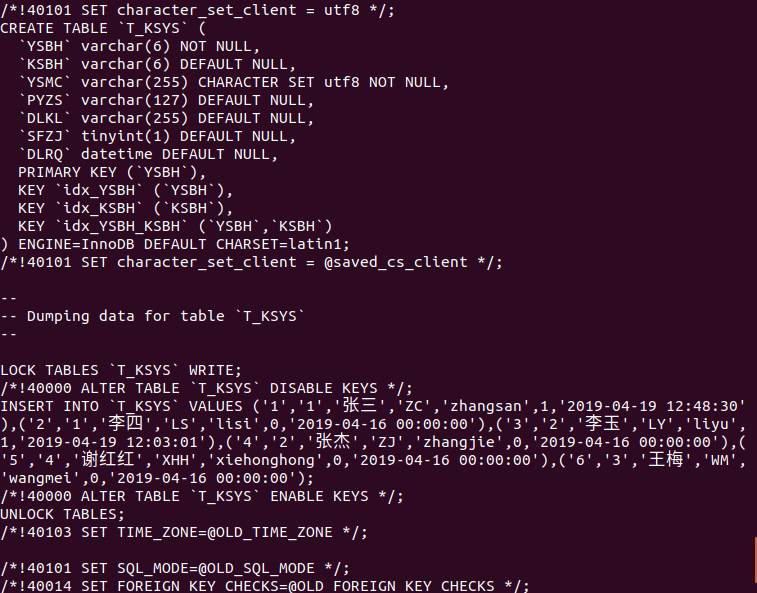
备份某个表:mysqldump --user [username] --password=[password] [database name] [table name] table\_name.sql

本次示例中,演示备份JavaExp数据库的备份,这个是我们JAVA实验使用到的数据库.



从上图中可以知道,备份文件已经成功保存到~目录下的JavaExp.sql中

如何恢复呢?我们首先打开.sql文件阅读一下, 使用命令cat JavaExp.sql, 得到结果如图:



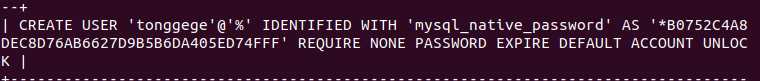
实际上该文件就是记录了我们执行的所有sql语句, 那么恢复的方法也就是重新执行上面的sql语句即可.

导入的方式: mysql>source ~/JavaExp.sql

授权及访问

Mysql访问权限有个范围,首先是数据库范围,这个是用来定义连接(connection)的,数据库需要授权数据库的连接.查看方法:SHOW GRANTS FOR ['bob'@'pc84.example.com'](mailto:'bob'@'pc84.example.com')

执行结果如图:

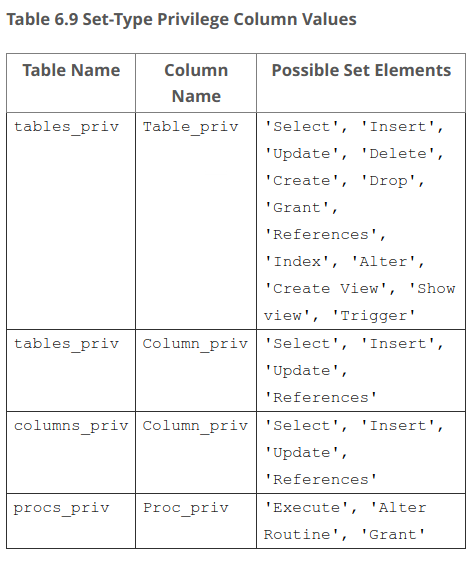


上图看到CREATE USER 'tonggege'@'%', 也就是说tonggege这个用户可以在任何主机上连接该数据库.

接下来是User表, 和db表类似,也是进行访问控制, 但是可以额外执行一些命令,比如show databases指令.

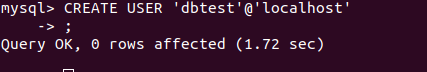
table权限表用来控制数据表的权限,用户是否对该表拥有权限

column权限表用来控制列的权限.其余权限暂时不在这里讨论.接下来是Mysql官方文档提供的权限设置表格,如图:



创建用户: CREATE USER 'dbtest'@'localhost'

结果如图:



这一步发生的事是,在 mysql表中插入一条记录,其中username列是dbtest, hostname列是localhost, 这样

dbtest用户可以在本机上登录该数据库.接下来设置角色:SET ROLE 'rolename'@'hostname'

设置权限:

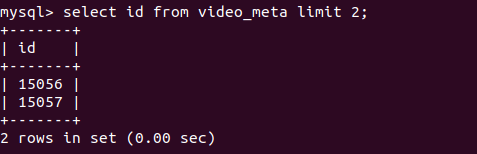
GRANT SELECT (id) ON TABLE BStation.video\_meta TO dbtest@localhost;

结果图:

2019-04-22 11-10-21 的屏幕截图

(为了不破坏原数据库结构,我就授权其他数据库了)

现在执行已授权数据:



可以正常执行

那么执行超过权限范围的命令,如图:

2019-04-22 11-16-52 的屏幕截图

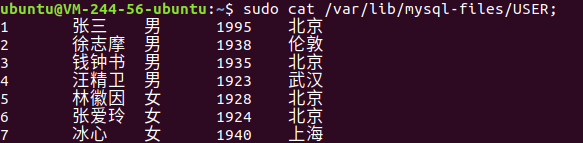
从上图可以看到, 超越权限的不可以使用.

MySQL数据导出:

首先是导出成为原始数据,使用的SQL语句是SELECT \* FROM tablename INTO OUTFILE '~/database/tablename.txt'

2019-04-25 19-44-55 的屏幕截图

则生成的文件保存在/var/lib/mysql-files/USER文件中,该导出的数据是表格文件,而不是SQL语句,我们用cat命令查看一下,查看结果如下:



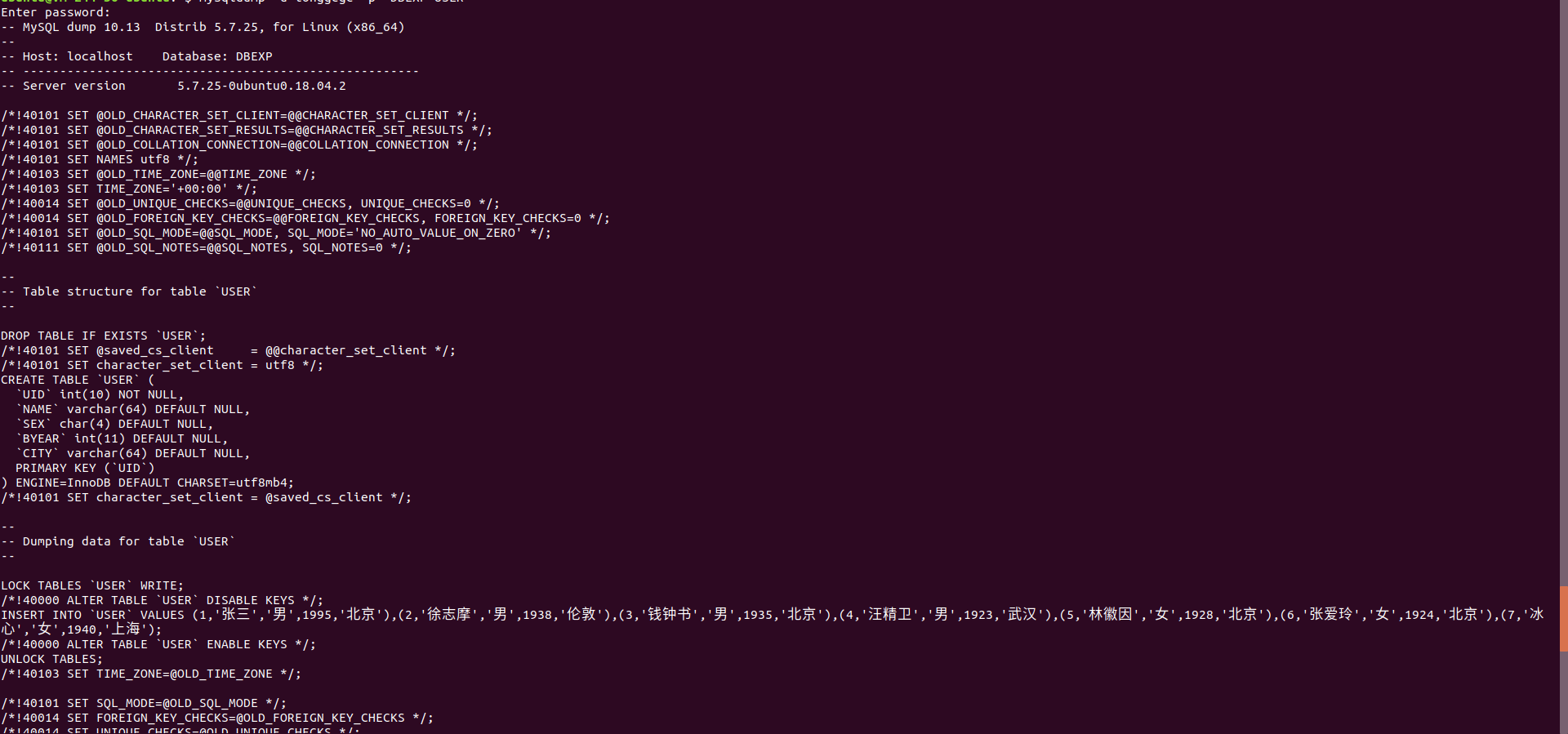
2.可以导出原SQL语句, 使用mysqldump工具

mysql -u root -p 数据库名 数据表名 > 导出文件名

在本次实验中,我们导出USER表,mysql -u root -p DBEXP USER > USER.sql

2019-04-25 20-03-10 的屏幕截图

为了验证结果是否正确,打开USER.sql观察如下:



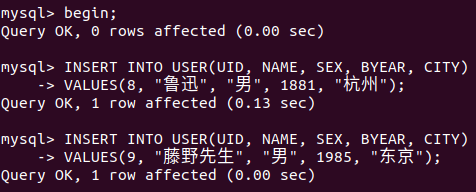
可以看到,将所有数据以SQL形式保存下来.

导入的方法同样使用.sql文件来进行,在终端界面使用source命令

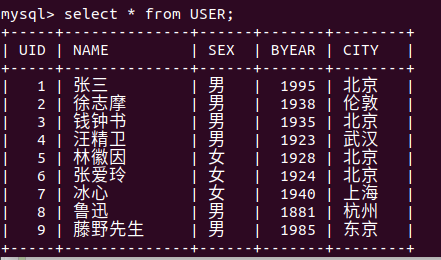


事务使用首先需要BEGIN, 然后输入相应的QUERY, 最后再COMMIT, 当操作表完毕后,如果有意外需要取消事务,则需要调用ROLLBACK回滚事务.

首先开始事务,如图所示:



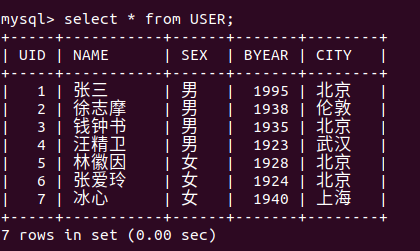
此时查看数据库插入结果,如图:



可以看到鲁迅和藤野先生记录已经在库中,但是此时如果有意外发生,不能完成这个原子操作,那么需要回滚,如图:

2019-04-29 09-57-26 的屏幕截图

此时再查询表,看一看是否回滚成功:



从上图可以看到,原子操作已经全部回滚.