1 用户授权

1.1 什么是用户授权：数据库管理连接服务后添加新用户 设置访问权限 和密码

1.2 命令格式

mysql> grant 权限列表 on 数据库名 to 用户名@"客户端地址" identified by "密码" [ with grant option ] ;

1.3 授权配置项的表示方式：

权限列表 all usage insert,select update(字段名列表)

数据库名 \*.\* 库名.\* 库名.表名

客户端地址 % 192.168.4.51 192.168.4.% localhost

1.4 客户端使用授权用户连接数据库服务器

]# mysql -h数据库服务器ip -u授权用户 -p密码

]# mysql -h192.168.4.50 -uyaya -p123456

例子

50 添加用户

mysql> grant all on \*.\* to mydba@"%" identified by "123qqq...A" with grant option;

mysql> grant select on db3.user to admin@"192.168.4.%" identified by "123qqq...A";

mysql> grant select , insert , update , delete on db3.\* to admin2@"localhost" identified by "123qqq...A" ;

mysql> grant all on \*.\* to root@"192.168.4.51" identified by "123qqq...A" with grant option;

51测试用户

[root@host51 ~]# mysql -h192.168.4.50 -umydba -p123qqq...A

mysql> 对数据做管理的命令都可以执行

[root@host51 ~]# mysql -h192.168.4.50 -uadmin -p123qqq...A

mysql> 对数据做管理

[root@host51 ~]# mysql -h192.168.4.50 -uadmin2 -p123qqq...A

mysql: [Warning] Using a password on the command line interface can be insecure.

ERROR 1045 (28000): Access denied for user 'admin2'@'192.168.4.51' (using password: YES)

[root@host51 ~]#

[root@host50 ~]# mysql -hlocalhost -uadmin2 -p123qqq...A

mysql> 对数据做管理时执行 select insert update delete 之外的sql命令会报错

[root@host51 ~]# mysql -h192.168.4.50 -uroot -p123qqq...A

mysql> grant all on db1.\* to admin3@"%" identified by "123qqq...A";

mysql> grant select on \*.\* to yaya88@"localhost" identified by "123qqq...A";

相关命令

mysql> select @@hostname; 查看主机名

mysql> select host , user from mysql.user;怎么看已有授权用户

show grants for admin3@"%" ; 数据库管理员 怎么看已有授权用户的权限？

连接数据库服务器的用户怎么查看自己的访问权限 show grants;

数据管理员已有授权用户的连接密码

mysql> select password("654321")；

mysql> set password for admin3@"%"=password("654321");

[root@host51 ~]# mysql -h192.168.4.50 -uadmin3 -p654321

mysql> select user();

mysql> show grants;

mysql> set password=password("123qqq...A"); 客户端可以修改连接密码？

MySQL> exit;

[root@host51 ~]# mysql -h192.168.4.50 -uadmin3 -p123qqq...A

[root@host50 ~]# mysql -uroot -ptarena

mysql> drop user admin3@"localhost"; 删除用户

授权库mysql （存储的时grant命令的执行结果）

user表 存储用户名 、客户端地址 全局权限 ( all \*.\* )

desc mysql.user;

select host , user from mysql.user;

select \* from mysql.user \G

show grants for admin3@"%";

db表 存储授权用户对服务器上已有 库的访问

desc mysql.db;

select host , user , db from mysql.db;

mysql> select \* from mysql.db where db="db1" \G

mysql> show grants for admin3@"%";

mysql> update mysql.db set Delete\_priv="N" where user="admin3";

mysql> flush privileges;

tables\_priv表 存储授权用户对服务器上已有 表的访问

MySQL> desc mysql.tables\_priv;

mysql> select host , user , db , table\_name from mysql.tables\_priv;

mysql> select \* from mysql.tables\_priv where Table\_name="user" \G

mysql> update mysql.tables\_priv set Table\_priv="insert,delete,select" where Table\_name="user";

mysql> flush privileges;

show grants for admin@"192.168.4.%" ;

columns\_priv表 存储授权用户对表里 字段的访问权限

desc mysql.columns\_priv;

mysql> select \* from mysql.columns\_priv;

Empty set (0.00 sec)

mysql> grant select , update(name,password) on db3.user to admin4@"%" identified by "123qqq...A";

Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> select \* from mysql.columns\_priv;

+------+-----+--------+------------+-------------+---------------------+-------------+

| Host | Db | User | Table\_name | Column\_name | Timestamp | Column\_priv |

+------+-----+--------+------------+-------------+---------------------+-------------+

| % | db3 | admin4 | user | name | 0000-00-00 00:00:00 | Update |

| % | db3 | admin4 | user | password | 0000-00-00 00:00:00 | Update |

+------+-----+--------+------------+-------------+---------------------+-------------+

2 rows in set (0.01 sec)

mysql>

目的：查看表记录可以获取用户权限；也可以通过更新记录，修改用户权限

撤销权限(删除已有用户的访问权限）

MySQL> revoke 权限列表 on 库名.表 from 用户名@"客户端地址";

查看授权用户权限

mysql> select host , user from mysql.user;

mysql> show grants for mydba@"%";

mysql> revoke grant option on \*.\* from mydba@"%";

mysql> revoke delete , drop on \*.\* from mydba@"%";

mysql> revoke all on \*.\* from mydba@"%";

mysql> show grants for mydba@"%";

+-----------------------------------+

| Grants for mydba@% |

+-----------------------------------+

| GRANT USAGE ON \*.\* TO 'mydba'@'%' |

+-----------------------------------+

1 row in set (0.00 sec)

mysql>

[root@host51 ~]# mysql -h192.168.4.50 -uyaya99 -p123456

mysql> show grants;

root密码管理（操作系统的管理员root用户）

修改数据库管理员root用户本机登录密码

命令格式[root@host50 ~]# mysqladmin -uroot -p旧密码 password "新密码"

例子

[root@host50 ~]# mysqladmin -uroot -ptarena password "666888"

[root@host50 ~]# mysql -uroot -p666888 #使用新密码登录

恢复数据库管理员root用户本机登录密码（忘记密码）

[root@host50 ~]# systemctl stop mysqld

[root@host50 ~]# vim /etc/my.cnf

[mysqld]

skip-grant-tables

#validate\_password\_policy=0

#validate\_password\_length=6

:wq

[root@host50 ~]# systemctl start mysqld

[root@host50 ~]# mysql

mysql> select user , host , authentication\_string from mysql.user;

mysql> update mysql.user set authentication\_string=password("123qqq...A")

where user="root" and host="localhost" ;

mysql> flush privileges;

mysql> exit

[root@host50 ~]# vim /etc/my.cnf

[mysqld]

#skip-grant-tables

validate\_password\_policy=0

validate\_password\_length=6

:wq

[root@host50 ~]# systemctl restart mysqld

[root@host50 ~]# mysql -uroot -p123qqq...A

数据备份的相关概念？

备份？ 备份的是数据 备份的目的：恢复数据

备份方式：

物理备份 把数据库目录 拷贝 一份 存放到其他的 存储设备上

物理恢复 把备份的文件拷贝回数据库目录，具体步骤如下：

停止数据库服务

清空数据库目录下数据

拷贝备份目录下的文件 到 数据库目录下

递归修改数据库目录的所有者和用户组 为mysql用户

启动数据库服务

管理员登录查看数据

演示

[root@host50 ~]# cp -r /var/lib/mysql /mysql.bak

[root@host50 ~]# scp -r /mysql.bak root@192.168.4.51:/opt/

[root@host51 ~]# systemctl stop mysqld

[root@host51 ~]# rm -rf /var/lib/mysql/\*

[root@host51 ~]#

[root@host51 ~]# ls /var/lib/mysql

[root@host51 ~]#

[root@host51 ~]# ls /opt/mysql.bak/

[root@host51 ~]# cp -r /opt/mysql.bak/\* /var/lib/mysql/

[root@host51 ~]# chown -R mysql:mysql /var/lib/mysql

[root@host51 ~]# systemctl start mysqld

[root@host51 ~]# mysql -uroot -p密码

逻辑备份：使用mysql软件提供备份命令做备份，在执行备份命令时，根据已有的库表记录生成对应的sql命令 保存定义的文件里。

数据备份策略：

完全备份：备份所有数据

只 备份 1张表的所有数据

只 备份 1个库的所有数据

备份1台数据库服务器的所有数据

只备份新产生的数据：

差异备份：备份完全备份后，所有新产生的数据

增量备份：备份上次备份后，所有新产生的数据

工作中的备份策略

完全备份+差异备份 （用当前的所有数据减去第1天的数据 剩下的就是 要备份的新数据）

完全备份+增量备份 （判断新数据，只和前一天比较）

休息+订餐 到 11：05

2 完全备份 （数据备份与恢复）

数据完全备份命令 mysqldump

]# mysqldump -uroot -p密码 数据库名 > 文件名.sql

数据库名的表示方式？

--all-databases 或 -A //所有库

数据库名 //单个库

数据库名 表名 //单张表

-B 数据库1 数据库2 //多个库

例子：

[root@host50 ~]# mkdir /bakdir

[root@host50 ~]# mysqldump -uroot -p123qqq...A --all-databases > /bakdir/allbak.sql

[root@host50 ~]# mysqldump -uroot -p123qqq...A db1 > /bakdir/db1.sql

[root@host50 ~]# mysqldump -uroot -p123qqq...A db3 user > /bakdir/db3\_user.sql

[root@host50 ~]# mysqldump -uroot -p123qqq...A -B db1 db3 > /bakdir/twodb.sql

[root@host50 ~]# ls /bakdir/

allbak.sql db1.sql db3\_user.sql twodb.sql

[root@host50 ~]#

数据完全恢复命令 mysql

]# mysql -uroot -p密码 库名 < 文件名.sql

例子

mysql> create database db3;

mysql>exit

[root@host51 ~]# mysql -uroot -p123qqq...A db3 < /root/db3\_user.sql

[root@host51 ~]# mysql -uroot -p123qqq...A -e ' select \* from db3.user '

[root@host51 ~]# mysql -uroot -p123qqq...A -e ' select count(\*) from db3.user '

删除已有库 只有不重启数据库服务器 就可以使用备份文件恢复数据

[root@host51 ~]# mysql -uroot -p123qqq...A < /root/allbak.sql

3 数据的增量备份与恢复

3.1 启用binlog日志实现数据的时时备份

3.1.1 binlog日志介绍 ？

3.1.2 启用binlog日志

]# vim /etc/my.cnf

[mysqld]

server\_id=50

log\_bin

:wq

]# systemctl restart mysqld

]# mysql -uroot -p密码

mysql> show master status;

mysql> system ls /var/lib/mysql/

mysql> insert into db3.user(name,uid) values("bob",3);

mysql> show master status;

mysql> select \* from db3.user;

mysql> show master status;

修改日志文件存放的目录和文件名

[root@host50 ~]# mkdir /mylog

[root@host50 ~]# chown mysql /mylog

[root@host50 ~]# ls -ld /mylog

drwxr-xr-x. 2 mysql root 6 8月 18 14:44 /mylog

[root@host50 ~]#

[root@host50 ~]# setenforce 0

]# vim /etc/my.cnf

[mysqld]

server\_id=50

#log\_bin

log\_bin=/mylog/plj

:wq

[root@host50 ~]# systemctl restart mysqld

[root@host50 ~]# ls /mylog

plj.000001 plj.index

[root@host50 ~]#

mysql: [Warning] Using a password on the command line interface can be insecure.

+------------+----------+--------------+------------------+-------------------+

| File | Position | Binlog\_Do\_DB | Binlog\_Ignore\_DB | Executed\_Gtid\_Set |

+------------+----------+--------------+------------------+-------------------+

| plj.000001 | 154 | | | |

+------------+----------+--------------+------------------+-------------------+

[root@host50 ~]#

[root@host50 ~]# mysql -uroot -p123qqq...A -e 'insert into db3.user(name) values("aaa")'

mysql: [Warning] Using a password on the command line interface can be insecure.

[root@host50 ~]#

[root@host50 ~]# mysql -uroot -p123qqq...A -e 'insert into db3.user(name) values("bbb")'

mysql: [Warning] Using a password on the command line interface can be insecure.

[root@host50 ~]#

[root@host50 ~]# mysql -uroot -p123qqq...A -e 'show master status'

mysql: [Warning] Using a password on the command line interface can be insecure.

+------------+----------+--------------+------------------+-------------------+

| File | Position | Binlog\_Do\_DB | Binlog\_Ignore\_DB | Executed\_Gtid\_Set |

+------------+----------+--------------+------------------+-------------------+

| plj.000001 | 696 | | | |

+------------+----------+--------------+------------------+-------------------+

[root@host50 ~]#

3.1.3 查看日志文件内容

]# mysqlbinlog 目录名/日志文件名

[root@host50 ~]# mysqlbinlog /mylog/plj.000001

3.1.4 手动创建新的日志文件

]# systemctl restart mysqld

[root@host50 ~]# mysql -uroot -p123qqq...A -e 'flush logs'

mysql> flush logs;

[root@host50 ~]# mysqldump -uroot -p123qqq...A --flush-logs db3 user > /opt/user2.sql

3.1.5 删除已有的日志文件

mysql> purge master logs to "plj.000004"; 删除指定编号之前的binlog日志文件

mysql> reset master ; 重建新日志

]# rm -rf \*.0000\* \*.index 使用系统命令2种全删除

3.2 使用日志恢复数据

使用日志恢复所有数据，命令格式

]#mysqlbinlog /目录名/日志名 | mysql -uroot -p密码

[root@host50 ~]#mysql> select count(\*) from db3.user;

[root@host50 ~]# mysqldump -uroot -p123qqq...A --flush-logs db3 user > /root/user.sql

mysqldump: [Warning] Using a password on the command line interface can be insecure.

[root@host50 ~]# scp /root/user.sql root@192.168.4.51:/root/

root@192.168.4.51's password:

user.sql 100% 3543 184.2KB/s 00:00

[root@host50 ~]#

[root@host50 ~]# mysql -uroot -p123qqq...A -e ' show master status'

mysql: [Warning] Using a password on the command line interface can be insecure.

+------------+----------+--------------+------------------+-------------------+

| File | Position | Binlog\_Do\_DB | Binlog\_Ignore\_DB | Executed\_Gtid\_Set |

+------------+----------+--------------+------------------+-------------------+

| plj.000002 | 154 | | | |

+------------+----------+--------------+------------------+-------------------+

[root@host50 ~]#

[root@host50 ~]# mysql -uroot -p123qqq...A -e 'insert into db3.user(name,uid) values("a1",33)'

mysql: [Warning] Using a password on the command line interface can be insecure.

[root@host50 ~]# mysql -uroot -p123qqq...A -e 'insert into db3.user(name,uid) values("a2",33)'

mysql: [Warning] Using a password on the command line interface can be insecure.

[root@host50 ~]# mysql -uroot -p123qqq...A -e 'insert into db3.user(name,uid) values("a3",33)'

mysql: [Warning] Using a password on the command line interface can be insecure.

[root@host50 ~]# mysql -uroot -p123qqq...A -e 'insert into db3.user(name,uid) values("a4",33)'

mysql: [Warning] Using a password on the command line interface can be insecure.

[root@host50 ~]#

[root@host50 ~]# mysql -uroot -p123qqq...A -e ' show master status'

mysql: [Warning] Using a password on the command line interface can be insecure.

+------------+----------+--------------+------------------+-------------------+

| File | Position | Binlog\_Do\_DB | Binlog\_Ignore\_DB | Executed\_Gtid\_Set |

+------------+----------+--------------+------------------+-------------------+

| plj.000002 | 1250 | | | |

+------------+----------+--------------+------------------+-------------------+

[root@host50 ~]# scp /mylog/plj.000002 root@192.168.4.51:/opt/

[root@host51 ~]# mysql -uroot -p123qqq...A db3 < /root/user.sql

mysql: [Warning] Using a password on the command line interface can be insecure.

[root@host51 ~]#

[root@host51 ~]# mysql -uroot -p123qqq...A -e 'select count(\*) from db3.user'

mysql: [Warning] Using a password on the command line interface can be insecure.

+----------+

| count(\*) |

+----------+

| 29 |

+----------+

[root@host51 ~]# ls /opt/plj.000002

/opt/plj.000002

[root@host51 ~]# mysqlbinlog /opt/plj.000002 | mysql -uroot -p123qqq...A

mysql: [Warning] Using a password on the command line interface can be insecure.

[root@host51 ~]#

[root@host51 ~]# mysql -uroot -p123qqq...A -e 'select count(\*) from db3.user'

mysql: [Warning] Using a password on the command line interface can be insecure.

+----------+

| count(\*) |

+----------+

| 33 |

+----------+

[root@host51 ~]#

使用日志恢复部分数据 , 命令格式

日志记录sql命令的方式?

记录命令执行的时间

记录命令执行的偏移量

]# mysqlbinlog 选项 /目录名/文件名 | mysql -uroot -p密码

读偏移量范围内记录的sql命令

--start-position=数字 --stop-position=数字

都时间范围内记录的sql命令

--start-datetime="yyyy/mm/dd hh:mm:ss" --stop-datetime="yyyy/mm/dd hh:mm:ss"

修改日志文件的记录格式查看

[root@host50 mylog]# mysql -uroot -p123qqq...A -e 'show variables like "binlog\_format"'

mysql: [Warning] Using a password on the command line interface can be insecure.

+---------------+-------+

| Variable\_name | Value |

+---------------+-------+

| binlog\_format | ROW |

+---------------+-------+

[root@host50 mylog]#

修改日志格式

]# vim /etc/my.cnf

[mysqld]

binlog\_format="mixed"

:wq

[root@host50 mylog]# systemctl restart mysqld

[root@host50 mylog]# mysql -uroot -p123qqq...A -e 'show variables like "binlog\_format"'

mysql: [Warning] Using a password on the command line interface can be insecure.

+---------------+-------+

| Variable\_name | Value |

+---------------+-------+

| binlog\_format | MIXED |

+---------------+-------+

[root@host50 mylog]# mysql -uroot -p123qqq...A -e 'show master status'

mysql: [Warning] Using a password on the command line interface can be insecure.

+------------+----------+--------------+------------------+-------------------+

| File | Position | Binlog\_Do\_DB | Binlog\_Ignore\_DB | Executed\_Gtid\_Set |

+------------+----------+--------------+------------------+-------------------+

| plj.000003 | 154 | | | |

+------------+----------+--------------+------------------+-------------------+

[root@host50 mylog]#

[root@host50 mylog]# mysql -uroot -p123qqq...A -e 'insert into db3.user(name,uid) values("c1",5)'

[root@host50 mylog]# mysql -uroot -p123qqq...A -e 'insert into db3.user(name,uid) values("c2",5)'

[root@host50 mylog]# mysql -uroot -p123qqq...A -e 'insert into db3.user(name,uid) values("c3",5)'

[root@host50 mylog]# mysql -uroot -p123qqq...A -e "delete from db3.user where name like 'a%'"

[root@host50 mylog]# mysql -uroot -p123qqq...A -e 'show master status'

mysql: [Warning] Using a password on the command line interface can be insecure.

+------------+----------+--------------+------------------+-------------------+

| File | Position | Binlog\_Do\_DB | Binlog\_Ignore\_DB | Executed\_Gtid\_Set |

+------------+----------+--------------+------------------+-------------------+

| plj.000003 | 1382 | | | |

+------------+----------+--------------+------------------+-------------------+

[root@host50 mylog]#

[root@host50 mylog]# mysqlbinlog /mylog/plj.000003 | grep insert

insert into db3.user(name,uid) values("c1",5)

insert into db3.user(name,uid) values("c2",5)

insert into db3.user(name,uid) values("c3",5)

[root@host50 mylog]#

[root@host50 mylog]# mysqlbinlog /mylog/plj.000003 | grep delete

delete from db3.user where name like 'a%'

[root@host50 mylog]#

[root@host50 mylog]#scp /mylog/plj.000003 root@192.168.4.51:/root/

[root@host51 ~]# mysqlbinlog --start-position=325 --stop-position=470 /root/plj.000003 | mysql -uroot -p123qqq...A

[root@host51 ~]# mysqlbinlog --start-datetime="2020/08/18 16:56:36" --stop-datetime="2020/08/18 16:59:29" /root/plj.000003 | mysql -uroot -p123qqq...A

[root@host51 ~]# mysql -uroot -p123qqq...A -e "select \* from db3.user WHERE name like 'c%'"