mysql\_day04

数据库服务器:192.168.4.50 192.168.4.51

1. **用户授权**

1.1什么事用户授权:在数据库服务器上添加客户端连接时使用的用户,并且设置访问权限.

1.2 为什么要学习用户授权:默认只允许数据库管理源root用户本机登录

1.3 怎么配置用户授权:用户授权配置格式:

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];

权限列表:

个别权限:select,insert,update,delete

所有权限:all

无权限:usage

数据库名:\*.\*(所有库中所有表) db3.\*(db3库下的所有表) db3.user(db3中user表设置权限)

用户名:客户端连接时私用的名字,授权时自定义即可

客户端地址:本机:localhost 某1台主机:192.168.4.51 某个网端主机:192.168.4.% 所有主机:%

密码:连接使用密码,要与数据库服务器的密码策略匹配

with grant option: 可选项,授权用户有执行grant命令的权限

1.4例子

mysql>grant all on db4.\* to root@”%" identified by "123qqq...A"; #用户授权

]#mysql -h 192.168.4.50 -u yaya -p123qqq...A #客户端连接

相关命令

|  |  |
| --- | --- |
| select user(); | 显示登录用户名及客户端地址 |
| show grants; | 用户显示自身访问权限 |
| show grants for 用户名@”客户端地址” | 管理员查看已有授权用户的权限 |
| set passsword=password(“密码”) | 授权用户连接后修改连接密码 |
| set password for 用户名@”客户端地址”=password(“密码”); | 管理员重置授权用户连接密码 |
| drop user 用户名@”客户端地址” | 删除授权用户(必须有管理员权限) |

mysql> show grants for yaya@"%"; #查看指定授权用户信息

mysql> set password for [yaya@"%"=password("123aaa...B");](mailto:yaya@\"%\"=password(\"123aaa...B\");) #修改授权用户的密码

mysql> drop yaya@"%"; #删除授权

验证权限:建库 建表 对记录做增删改查

1.5 授权库

mysql授权库 记录授权信息主要表如下:

-user表 记录已有收取那用户及权限

-db表 记录已有授权用户对数据库的访问权限

-tables\_priv表 记录已有授权用户对表的范围权限

-columns\_priv表 记录已有授权用户对字段的访问权限

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

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

mysql> select host,user from mysql.user;

mysql> select \* from mysql.user where user="root" and host="localhost"\G; #G:以行形式显示

]#mysql -h 192.168.4.50 -uroot -p123qqq...A;

mysql> show grants;

mysql> show grants for [yaya99@”192.168.4.51”](mailto:yaya99@\”192.168.4.51\”)

mysql> select \* from mysql.tables\_priv where user=”yaya99” and host=”192.168.4.51”\G;

mysql> update mysql.tables\_priv set Table\_priv=”select,insert,delete” where user=”yaya99” and

host=”192.168.4.51”;

mysql> flush privileges;

mysql> show grants for yaya99@”192.168.4.51”;

1.6撤销权限

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

mysql> select user,host from mysql.user;

mysql> show grants for root@”%”; #查看指定授权用户的权限

mysql> revoke grant option on \*.\* from root@”%”; #撤销指定用户的权限

mysql> revoke delete,update on \*.\* from root@"%"; #撤销指定用户的指定权限

mysql> show grants for root@”%”

mysql> revoke all on \*.\*from root@”%”;

mysql> show grants fro root@”%”;

mysql> drop user root@”%”; #删除是定授权用户

1.7 root密码

1.7.1 修改密码

[root@mysql]#mysqladmin -hlocalhost -uroot -p password ”新密码” #修改用户密码

password 旧密码

1.7.2 回复密码

]#vim /etc/my.cof

[mysqld]

skip-grant-tables #设置无密码登录

:wq

]#systemctl restart mysqld

]#mysql

mysql> update mysql.user set authentication\_string=password(123qqq...Q) where user="root" and

host="loacalhost"; #修改用户密码

mysql> flush privileges; #刷新设置

]# vim /eyc/my.cnf

[mysqld]

#skip-grant-tables #取消无密码登录

:wq

]#systemctl restart mysqld

]#mysql -u rpoot -p123qqq...Q

**2.完全备份**

2.1 物理备份及回复

2.1.2 第一台数据库服务器操作步骤如下:

mkdir /dbback

ls /dbback

cp -r /var/lib/mysql /dbbak

ls /dbbak/mysql

tar -zxvf /dbbak/mysql.tar.gz /var/lib/mysql/\*

scp -r /dbbak/mysql [root@192.168.4.51:/root/](mailto:root@192.168.4.51:/root/)

2.1.2 第二台数据库服务器操作步骤

rm -rf /var/lib/mysql

cp -r /root/mysql /var/lib/

chown -R mysql:mysql /var/lib/mysql

2.2逻辑备份

2.2.1 数据备份策略

\*完全备份:备份所有数据

\*增量备份:备份上次备份后,所有新产生的数据

\*差异备份:备份完全备份后,所有新产生的数据

2.2.2 备份时库名表示方式

--all-databases 或 -A //所有库

数据库名 //单个库

数据库名 表明 //单张表

-B 数据库1 数据库2 //多个库

注意事项

无论备份还是回复,都要验证用户权限!!!

2.2.3 完全备份及回复

完全备份

]#mysqldump -uroot -p密码 库名 > 目录/xxx.sql

mysql> mysqldump -uroot -p123456 -A > /dbbak/all.sql #所有数据库备份

mysql> mysqldump -uroot -p123456 db3 > /dbbak/db3.sql #备份指定数据库

mysql> mysqldump -uroot -p123456 db3 user > /dbbak/db3\_user.sql #备份指定数据库表

mysql> mysqldump -uroot -p123456 -B mysql db3 > /dbbak/twodb.sql #备份多个数据库

完全回复

]#mysqldump -uroot -p密码 [库名] < 目录/xxx.sql

scp /dbbak/db3\_user [root@192.168.4.51:/opt/](mailto:root@192.168.4.51:/opt/)

mysql -u root -p123456 userdb3 < db3\_user

mysql -u root -p123456 userdb3

1. 增量备份

启用mysql服务的binlog日志文件实现岁数据的实时增量备份

3.1.1什么是binlog日志

1. 也成为二进制日志
2. Mysql服务日志文件的一种
3. 记录除查询之外的所有SQL命令
4. 可用于数据备份回复
5. 配置MySQL主从同步的必要条件

3.1.2 启用日式

|  |  |
| --- | --- |
| 配置项 | 用途 |
| server\_id=数字 | 指定id值(1-255) |
| log\_bin[=目录名/文件名] | 启用binlog日志 |
| max\_binlog\_size=数值m | 指定日志文件容量,默认1G |

[root@mysql]#vim /etc/my.cnf

[mysqld]

......

log\_bin #开启binlog日志

server\_id=50 #指定值

:wq

[root@mysql]#systemctl restart mysqld

[root@mysql]#mysql -u root -p123456

mysql> show master status; #查看日志文件和偏移量

mysql> create table db3.usera(id int);

mysql> insert into db3.usera values(101);

mysql> show master status;

mysql> exit

3.1.3使用binlog日志回复数据

[root@host50 mysql]# scp /var/lib/mysql/host50-bin.000001 root@192.168.4.51:/opt/

命令格式]# mysqlbinlog 目录/文件 | mysql -uroot -pabc123...Q

例子[root@host51 ~]# mysqlbinlog /opt/host50-bin.000001 | mysql -uroot -pabc123...Q

]# mysql -uroot -pabc123...Q

> select \* from db3.usera;

3.2.1自定义日志文件名和存储目录

]#vim /etc/my.cnf

[mysql]

server\_id=50

log\_bin=/mylog/jw

:wq

]#mkdir /mylog

]#chown mysql /mylog

]#systemctl restart mysqld

]#ls /mylog/jw.\*

]#mysql -uroot -p123456

mysql> show master status

3.2.2查看正在使用的日志文件

show master status

3.2.3手动创建新的日志文件

(1)重启服务生成新的日志文件 systemctl restart mysqld

(2)flush logs;刷新日志会生成新的日志文件或(mysql -uroot -p123456 -e ‘flush log’)

(3)mysqldump db3 --flush-logs > /dbbak/db3.sql执行完全备份时加--flush-logs生成新的日志文件

3.2.4清理日志

删除指定编号之前的binlog日志文件

mysql> purge master logs to “binlog文件名”;

删除所有binlog日志,重建新日志

mysql> reset master;

3.2.4修改日志记录格式

三种记录方式:statement<报表模式> row<行模式> mixed<混合模式>

修改日式记录格式

]#vim /etc/my.cnf

[mysqld]

... ...

binlog\_format=”名称”

]#systemctl restart mysqld

]#mysql -uroot -p123456

mysql> insert into db3.user(name) values("kaisha");

mysql> insert into db3.user(name) values("kaika");

mysql> update db3.user set password=”F” where name like “kai%”

mysql> exit

]mysqlbinlog /mylog/jw000003 grep -i insert

3.2.5日志文件如何区分记录多条sql命令(日志记录sql命令的方式)

偏移量 --start-position=数字 --stop-postion=数字

时间点 --start-datatime=”yyyy-mm-dd hh:mm:ss” --stop-datatime=”yyyy-mm-dd hh:mm:ss”

读取日志指定范围的sql命令恢复数据

]#scp /mylog/jw.000002 [root@192.168.4.51:/opt/](mailto:root@192.168.4.51:/opt/)

]#mysqlbinlog --start-position= --stop-position= /opt/jw.000002 | mysql -u root -p123456

]#mysql -uroot -p123456 -e “select name from db3.user where name like ‘kai%’”