vipMySQL主从同步

环境：dbsvr1（主服务器） 192.168.4.10

dbsvr2（从服务器） 192.168.4.20

#传软件到两台虚拟机

[root@localhost ~]# cd software/

[root@localhost software]# scp mysql-5.7.17-1.el7.x86\_64.rpm-bundle.tar 192.168.4.10:/root

root@192.168.4.10's password:

mysql-5.7.17-1.el7.x 100% 557MB 79.6MB/s 00:07

[root@localhost software]# scp mysql-5.7.17-1.el7.x86\_64.rpm-bundle.tar 192.168.4.20:/root

root@192.168.4.20's password:

mysql-5.7.17-1.el7.x 100% 557MB 69.6MB/s 00:08

[root@localhost software]#

#在dbsvr1上安装MySQL5.7

[root@dbsvr1 ~]# ls

mysql-5.7.17-1.el7.x86\_64.rpm-bundle.tar

[root@dbsvr1 ~]# tar -xf mysql-5.7.17-1.el7.x86\_64.rpm-bundle.tar

[root@dbsvr1 ~]# ls

mysql-5.7.17-1.el7.x86\_64.rpm-bundle.tar

mysql-community-client-5.7.17-1.el7.x86\_64.rpm

mysql-community-common-5.7.17-1.el7.x86\_64.rpm

mysql-community-devel-5.7.17-1.el7.x86\_64.rpm

mysql-community-embedded-5.7.17-1.el7.x86\_64.rpm

mysql-community-embedded-compat-5.7.17-1.el7.x86\_64.rpm

mysql-community-embedded-devel-5.7.17-1.el7.x86\_64.rpm

mysql-community-libs-5.7.17-1.el7.x86\_64.rpm

mysql-community-libs-compat-5.7.17-1.el7.x86\_64.rpm

mysql-community-minimal-debuginfo-5.7.17-1.el7.x86\_64.rpm

mysql-community-server-5.7.17-1.el7.x86\_64.rpm

mysql-community-server-minimal-5.7.17-1.el7.x86\_64.rpm

mysql-community-test-5.7.17-1.el7.x86\_64.rpm

[root@dbsvr1 ~]# tar -xf mysql-5.7.17-1.el7.x86\_64.rpm-bundle.tar

[root@dbsvr1 ~]# rm -rf mysql-community-server-minimal-5.7.17-1.el7.x86\_64.rpm #删除不不要的软件包

[root@dbsvr1 ~]# yum -y localinstall mysql-community-\*.rpm

[root@dbsvr1 ~]# systemctl start mysqld.service #启动服务

[root@dbsvr1 ~]# grep 'password' /var/log/mysqld.log

2017-11-25T08:03:47.613909Z 1 [Note] A temporary password is generated for root@localhost: i35Xc%HCF:lD

[root@dbsvr1 ~]# mysql -u root -pi35Xc%HCF:lD

#登录mysql数据库

mysql> set global validate\_password\_policy=0;

Query OK, 0 rows affected (0.00 sec)

mysql> set global validate\_password\_length=4;

Query OK, 0 rows affected (0.00 sec)

mysql> SET PASSWORD FOR 'root'@'localhost'=PASSWORD('123456');

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

#修改数据库密码

##dbsvr2上重复同样的操作，完成实验基础环境

[root@dbsvr1 ~]# mysql -uroot -p123456

mysql> reset master;

Query OK, 0 rows affected (0.01 sec)

mysql> exit

Bye

重置主服务器状态

[root@dbsvr1 ~]# mysqldump -uroot -p123456 --all-databases > /root/mytest.sql

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

[root@dbsvr1 ~]# ls /root/mytest.sql

/root/mytest.sql

#导出主服务器数据库

scp /root/mytest.sql 192.168.4.20:/root #上传到从服务器

[root@dbsvr2 ~]# mysql -u root -p123456 < /root/mytest.sql

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

#从服务器导入主数据库服务器数据

[root@dbsvr1 ~]# cat /etc/my.cnf | grep -v "^#"

[mysqld]

log\_bin=dbsvr1-bin

server\_id=10

datadir=/var/lib/mysql

socket=/var/lib/mysql/mysql.sock

symbolic-links=0

log-error=/var/log/mysqld.log

pid-file=/var/run/mysqld/mysqld.pid

[root@dbsvr1 ~]# systemctl restart mysqld.service

#修改MySQL主配置文件，重启服务

[root@dbsvr1 ~]# mysql -uroot -p123456

mysql> set global validate\_password\_policy=0;

Query OK, 0 rows affected (0.00 sec)

mysql> set global validate\_password\_length=4;

Query OK, 0 rows affected (0.00 sec)

mysql> GRANT REPLICATION SLAVE ON \*.\* TO 'replicater'@'192.168.4.%' IDENTIFIED BY "123456";

Query OK, 0 rows affected, 1 warning (0.01 sec)

#授权从用户

mysql> show master status\G

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

File: dbsvr1-bin.000003

Position: 453

Binlog\_Do\_DB:

Binlog\_Ignore\_DB:

Executed\_Gtid\_Set:

1 row in set (0.00 sec)

#查看主服务器状态

[root@dbsvr2 ~]# cat /etc/my.cnf | grep -v ^#

[mysqld]

log\_bin=dbsvr2-bin

server\_id=20

datadir=/var/lib/mysql

socket=/var/lib/mysql/mysql.sock

symbolic-links=0

log-error=/var/log/mysqld.log

pid-file=/var/run/mysqld/mysqld.pid

[root@dbsvr2 ~]# systemctl restart mysqld.service

#配置从服务器

[root@dbsvr2 ~]# mysql -uroot -p123456

mysql> change master to

-> master\_host='192.168.4.10',

-> master\_user='replicater',

-> master\_password='123456',

-> master\_log\_file='dbsvr1-bin.000003',

-> master\_log\_pos=453;

Query OK, 0 rows affected, 2 warnings (0.09 sec)

#通过change master语句指定主服务器

mysql> start slave;

Query OK, 0 rows affected (0.01 sec)

#启动同步

mysql> show slave status\G

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Slave\_IO\_State: Waiting for master to send event

Master\_Host: 192.168.4.10

Master\_User: replicater

Master\_Port: 3306

Connect\_Retry: 60

Master\_Log\_File: dbsvr1-bin.000003

Read\_Master\_Log\_Pos: 453

Relay\_Log\_File: dbsvr2-relay-bin.000002

Relay\_Log\_Pos: 321

Relay\_Master\_Log\_File: dbsvr1-bin.000003

Slave\_IO\_Running: Yes

Slave\_SQL\_Running: Yes

Replicate\_Do\_DB:

Replicate\_Ignore\_DB:

Replicate\_Do\_Table:

Replicate\_Ignore\_Table:

Replicate\_Wild\_Do\_Table:

Replicate\_Wild\_Ignore\_Table:

Last\_Errno: 0

Last\_Error:

Skip\_Counter: 0

Exec\_Master\_Log\_Pos: 453

Relay\_Log\_Space: 529

Until\_Condition: None

Until\_Log\_File:

Until\_Log\_Pos: 0

Master\_SSL\_Allowed: No

Master\_SSL\_CA\_File:

Master\_SSL\_CA\_Path:

Master\_SSL\_Cert:

Master\_SSL\_Cipher:

Master\_SSL\_Key:

Seconds\_Behind\_Master: 0

Master\_SSL\_Verify\_Server\_Cert: No

Last\_IO\_Errno: 0

Last\_IO\_Error:

Last\_SQL\_Errno: 0

Last\_SQL\_Error:

Replicate\_Ignore\_Server\_Ids:

Master\_Server\_Id: 10

Master\_UUID: 2a3cf8f9-d1b7-11e7-831f-525400beaa01

Master\_Info\_File: /var/lib/mysql/master.info

SQL\_Delay: 0

SQL\_Remaining\_Delay: NULL

Slave\_SQL\_Running\_State: Slave has read all relay log; waiting for more updates

Master\_Retry\_Count: 86400

Master\_Bind:

Last\_IO\_Error\_Timestamp:

Last\_SQL\_Error\_Timestamp:

Master\_SSL\_Crl:

Master\_SSL\_Crlpath:

Retrieved\_Gtid\_Set:

Executed\_Gtid\_Set:

Auto\_Position: 0

Replicate\_Rewrite\_DB:

Channel\_Name:

Master\_TLS\_Version:

1 row in set (0.00 sec)

#确认Slave\_IO\_Running: Yes和Slave\_SQL\_Running: Yes两个线程为yes则成功

mysql> create database newdb;

Query OK, 1 row affected (0.02 sec)

mysql> use newdb;

Database changed

mysql> create table newtable(id int(4));

Query OK, 0 rows affected (0.10 sec)

mysql> insert into newtable values(1234),(2234);

Query OK, 2 rows affected (0.04 sec)

Records: 2 Duplicates: 0 Warnings: 0

mysql> select \* from newdb.newtable;

+------+

| id |

+------+

| 1234 |

| 2234 |

+------+

2 rows in set (0.00 sec)

#主服务器创建新数据库和表，并测试

mysql> select \* from newdb.newtable;

+------+

| id |

+------+

| 1234 |

| 2234 |

+------+

2 rows in set (0.00 sec)

#从服务器测试同步结果

[root@localhost software]# scp maxscale-2.1.2-1.rhel.7.x86\_64.rpm 192.168.4.100:/root

root@192.168.4.100's password:

maxscale-2.1.2-1.rhel.7.x86\_64.rpm 100% 17MB 16.5MB/s 00:01

#上传代理软件到代理服务器

[root@proxy ~]# yum -y localinstall maxscale-2.1.2-1.rhel.7.x86\_64.rpm

#安装代理软件

[root@proxy ~]# cat /etc/maxscale.cnf | grep -v ^#

[maxscale]

threads=1

[server1]

type=server

address=192.168.4.10

port=3306

protocol=MySQLBackend

[server2]

type=server

address=192.168.4.20

port=3306

protocol=MySQLBackend

[MySQL Monitor]

type=monitor

module=mysqlmon

servers=server1, server2

user=scalemon

passwd=123456

monitor\_interval=10000

[Read-Write Service]

type=service

router=readwritesplit

servers=server1, server2

user=maxscale

passwd=123456

max\_slave\_connections=100%

[MaxAdmin Service]

type=service

router=cli

[Read-Write Listener]

type=listener

service=Read-Write Service

protocol=MySQLClient

port=4006

[MaxAdmin Listener]

type=listener

service=MaxAdmin Service

protocol=maxscaled

socket=default

#修改代理软件配置文件

mysql> grant replication slave, replication client on \*.\* to scalemon@'%' identified by "123456";

Query OK, 0 rows affected, 1 warning (0.04 sec)

mysql> grant select on \*.\* to maxscale@"%" identified by '123456';

Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql> grant all on \*.\* to student@"%" identified by '123456';

Query OK, 0 rows affected, 1 warning (0.02 sec)

#主服务器设置用户授权

[root@proxy ~]# maxscale --config=/etc/maxscale.cnf

[root@proxy ~]# netstat -anptu | grep maxscale

tcp 0 0 192.168.4.100:42654 192.168.4.20:3306 ESTABLISHED 1113/maxscale

tcp 0 0 192.168.4.100:33882 192.168.4.10:3306 ESTABLISHED 1113/maxscale

tcp6 0 0 :::4006 :::\* LISTEN 1113/maxscale

#启动代理服务并确认监听端口

[root@localhost software]# mysql -h192.168.4.100 -P4006 -ustudent -p123456

MySQL [(none)]> select @@hostname;

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

| @@hostname |

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

| dbsvr2.tedu.cn |

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

1 row in set (0.00 sec)

MySQL [(none)]> insert into newdb.newtable values(3234);

Query OK, 1 row affected (0.04 sec)

#客户端进行测试