27.MySQL主从复制–传统主从架构多种备份方式构建√

初始环境

各种方式搭建基于初始环境

Bash | Copy

```
# 三台虚拟机
10.0.0.51  db01
10.0.0.52  db02
10.0.0.53  db03
防火墙关闭

#清理环境:
pkill mysqld
rm -rf /data/3306/*
mkdir -p /data/3306/data /data/3306/binlog
chown -R mysql.mysql /data/*

# 准备配置文件
主库db01:
mv /etc/my.cnf /tmp
cat > /etc/my.cnf <<EOF
[mysqld]
basedir=/usr/local/mysql
datadir=/data/3306/data
socket=/tmp/mysql.sock
server_id=51
port=3306
secure-file-priv=/tmp
log_bin=/data/3306/binlog/mysql-bin
binlog_format=row
gtid-mode=on
enforce-gtid-consistency=true
log-slave-updates=1
[mysql]
prompt=db01 [\\d]>
EOF

slave1(db02):
mv /etc/my.cnf /tmp
cat > /etc/my.cnf <<EOF
[mysqld]
basedir=/usr/local/mysql
datadir=/data/3306/data
socket=/tmp/mysql.sock
server_id=52
port=3306
secure-file-priv=/tmp
log_bin=/data/3306/binlog/mysql-bin
binlog_format=row
gtid-mode=on
enforce-gtid-consistency=true
log-slave-updates=1
[mysql]
prompt=db02 [\\d]>
EOF
```

```
slave2(db03):
mv /etc/my.cnf /tmp
cat > /etc/my.cnf <<EOF
[mysqld]
basedir=/usr/local/mysql
datadir=/data/3306/data
socket=/tmp/mysql.sock
server_id=53
port=3306
secure-file-priv=/tmp
log_bin=/data/3306/binlog/mysql-bin
binlog_format=row
gtid-mode=on
enforce-gtid-consistency=true

log-slave-updates=1
[mysql]
prompt=db03 [\\d]>
EOF

初始化数据
mysqld --initialize-insecure --user=mysql --basedir=/usr/local/mysql --datadir=/data/3306/data

启动数据库
/etc/init.d/mysqld start
```

第一种方式-通过MDP备份构建传统主从

1.首先检查各个节点的server_id,server_uuid,binlog状态

```
 mysql -e "select @@server_id; select @@server_uuid;show variables like 'log_bin%';"
 +-------------+
| @@server_id |
+-------------+
|          51 |
+-------------+
+--------------------------------------+
| @@server_uuid                        |
+--------------------------------------+
| 66135e76-b15e-11eb-8dbe-000c29edc386 |
+--------------------------------------+
+---------------------------------+---------------------------------+
| Variable_name                   | Value                           |
+---------------------------------+---------------------------------+
| log_bin                         | ON                              |
| log_bin_basename                | /data/3306/binlog/mysql-bin     |
| log_bin_index                   | /data/3306/binlog/mysql-bin.index |
| log_bin_trust_function_creators | OFF                             |
| log_bin_use_v1_row_events       | OFF                             |
+---------------------------------+---------------------------------+
```

2.主库（51）创建复制用户和远程管理用户

```
复制用户
mysql -e "create user repl@'10.0.0.%' identified with mysql_native_password by '123';
grant replication slave on *.* to repl@'10.0.0.%';"
远程管理用户
mysql -e "create user root@'10.0.0.%' identified with mysql_native_password by '123';
grant all on *.* to root@'10.0.0.%';

检查创建的用户
[root@db01 data]#  mysql -e "select user,host,plugin from mysql.user;"
+------------------+-----------+-----------------------+
| user             | host      | plugin                |
+------------------+-----------+-----------------------+
| repl             | 10.0.0.%  | mysql_native_password |
| root             | 10.0.0.%  | mysql_native_password |
| mysql.infoschema | localhost | caching_sha2_password |
| mysql.session    | localhost | caching_sha2_password |
| mysql.sys        | localhost | caching_sha2_password |
| root             | localhost | caching_sha2_password |
+------------------+-----------+-----------------------+
```

3.从库（52）备份主库数据并恢复

```
[root@db02 ~]#  mysqldump -uroot -p123 -h 10.0.0.51 -P 3306 -A --master-data=2 --single-transaction -R -E --tri
登陆从库恢复数据
db02 [mysql]>source /tmp/full.sql
```

4. 告诉从库（52）复制起点信息

```
获取复制起点
[root@db02 ~]# grep  "\--\ CHANGE MASTER" /tmp/full.sql
-- CHANGE MASTER TO MASTER_LOG_FILE='mysql-bin.000002', MASTER_LOG_POS=1187;
告知复制起点
[root@db02 data]# mysql -e \
"CHANGE MASTER TO \
  MASTER_HOST='10.0.0.51',\
  MASTER_USER='repl', \
  MASTER_PASSWORD='123', \
  MASTER_PORT=3306, \
  MASTER_LOG_FILE='mysql-bin.000002', \
  MASTER_LOG_POS=1187, \
  MASTER_CONNECT_RETRY=10;"
```

5.从库（52）启动专用复制线程，并查看

```
[root@db02 ~]# mysql -e "start slave;"
[root@db02 ~]# mysql -e "show slave status \G"|grep "Running:"
               Slave_IO_Running: Yes
               Slave_SQL_Running: Yes
```

第二种方式–通过XPK备份构建传统主从

1.上传xpk软件包，并进行安装

```
[root@db03 opt]# yum -y install percona-xtrabackup-80-8.0.13-1.el7.x86_64.rpm
```

2.主库（51）root用户设置密码

```
[root@db01 ~]# mysql
db01 [(none)]>alter user root@'localhost' identified by '123';
```

3.主库（51）创建备份目录并授权

```bash
[root@db01 ~]#  mkdir -p /data/backup
[root@db01 ~]#  chown -R mysql. /data/backup/
```

4.主库（51）修改配置文件

```bash
[root@db01 ~]# vim /etc/my.cnf
[client]
socket=/tmp/mysql.sock
```

5. 主库（51）进行全备

```bash
[root@db01 ~]# xtrabackup --defaults-file=/etc/my.cnf  --socket=/tmp/mysql.sock --user=root --password=123   --
```

6.从库（53）拷贝主库（51）配置文件和备份文件

```bash
[root@db03 opt]# scp -r /etc/my.cnf 10.0.0.53:/etc/my.cnf
root@10.0.0.53's password:
my.cnf                                                      100%  271   380.8KB/s   00:00
[root@db03 opt]# scp -r /data/backup/full/ 10.0.0.53:~ （拷贝到家目录下）
```

7.从库（53）修改复制主库的配置文件

```
[mysqld]
basedir=/usr/local/mysql
datadir=/data/3306/data
socket=/tmp/mysql.sock
server_id=53                ---修改这里
port=3306
secure-file-priv=/tmp
log_bin=/data/3306/binlog/mysql-bin
binlog_format=row
gtid-mode=on
enforce-gtid-consistency=true
log-slave-updates=1
[mysql]
prompt=db01 [\d]>
[client]
socket=/tmp/mysql.sock
```

8.从库（53）恢复主库（51）数据

```
因为我们环境从库53是克隆主库51的虚拟机，所以本身就有数据
所以我们这里删除从库53的数据，再恢复成主库数据
0.清空数据
[root@db03 ~]# pkill mysqld
[root@db03 ~]# rm -rf /data/3306/data/*
[root@db03 ~]# rm -rf /data/3306/logs/*
[root@db03 ~]# rm -rf /data/3306/binlog/*
1.CR保证数据一致
[root@db03 ~]# xtrabackup  --defaults-file=/etc/my.cnf  --socket=/tmp/mysql.sock --user=root --password=123  -
2.恢复数据
[root@db03 ~]# xtrabackup --defaults-file=/etc/my.cnf  --socket=/tmp/mysql.sock --user=root --password=123 --
3.将恢复数据的目录进行授权
[root@db03 full]# chown -R mysql.mysql /data/*
4.重启服务
[root@db03 full]# /etc/init.d/mysqld start
```

9.告诉从库（53）复制起点信息

```
获取本次拷贝到的binlog位置点
[root@db03 full]# cat /root/full/xtrabackup_binlog_info
binlog.000015 156
[root@db03 data]# mysql -uroot -p123 -e \
"CHANGE MASTER TO \
  MASTER_HOST='10.0.0.51',\
  MASTER_USER='repl', \
  MASTER_PASSWORD='123', \
  MASTER_PORT=3306, \
  MASTER_LOG_FILE='mysql-bin.000015', \
  MASTER_LOG_POS=156, \
  MASTER_CONNECT_RETRY=10;"
```

10.从库（52）启动专用复制线程，并查看

```
db03 [(none)]>start slave;
[root@db02 ~]# mysql -e "show slave status \G"|grep "Running:"
             Slave_IO_Running: Yes
             Slave_SQL_Running: Yes
```

第三种方式-通过Clone-plugin搭建传统和GTID主从

1.主库操作

```
加载插件，创建克隆捐赠者用户，授权
mysql -e "INSTALL PLUGIN clone SONAME 'mysql_clone.so';create user test1@'%' identified by '123';grant backup_
```

2.从库操作

```
0.加载插件，创建克隆接受者用户，授权，设置白名单
[root@db03 ~]#  mysql -e "INSTALL PLUGIN clone SONAME 'mysql_clone.so';create user test2@'%' identified by '123';grant clone_admin on *.* to test2@'%';SET GLOBAL clone_valid_donor_list='10.0.0.51:3306';"
1.克隆操作
[root@db03 ~]#  mysql -utest2 -p123 -h10.0.0.53  -P3306 -e "CLONE INSTANCE FROM test1@'10.0.0.51':3306 IDENTIFIED BY '123';"
2.查看从库复制起点
[root@db03 ~]#  mysql -uroot -p123 -e  "SELECT BINLOG_FILE, BINLOG_POSITION FROM performance_schema.clone_status;"
mysql: [Warning] Using a password on the command line interface can be insecure.
+------------------+------------------+
| BINLOG_FILE      | BINLOG_POSITION |
+------------------+------------------+
| mysql-bin.000003 |              923 |
+------------------+------------------+
```

3.从库启动主从复制

```
1.从库设置复制起点
[root@db03 ~]# mysql -uroot -p123  -e \
"CHANGE MASTER TO \
  MASTER_HOST='10.0.0.51',\
  MASTER_USER='repl', \
  MASTER_PASSWORD='123', \
  MASTER_PORT=3306, \
  MASTER_LOG_FILE='mysql-bin.000003', \
  MASTER_LOG_POS=923, \
  MASTER_CONNECT_RETRY=10;"
2.启动线程
[root@db03 ~]# mysql -uroot -p123 -e "start slave;"
```

4.查看

```
[root@db03 ~]#  mysql -uroot -p123 -e "show slave status \G"|grep "Running:"
mysql: [Warning] Using a password on the command line interface can be insecure.
               Slave_IO_Running: Yes
              Slave_SQL_Running: Yes
```

第四种方式 GTID搭建新的主从环境

Bash | Copy

```
# 三台虚拟机
10.0.0.51  db01
10.0.0.52  db02
10.0.0.53  db03
防火墙关闭

#清理环境:
pkill mysqld
rm -rf /data/3306/*
mkdir -p /data/3306/data /data/3306/binlog
chown -R mysql.mysql /data/*


# 准备配置文件
主库db01:
mv /etc/my.cnf /tmp
cat > /etc/my.cnf <<EOF
[mysqld]
basedir=/usr/local/mysql
datadir=/data/3306/data
socket=/tmp/mysql.sock
server_id=51
port=3306
secure-file-priv=/tmp
log_bin=/data/3306/binlog/mysql-bin
binlog_format=row
gtid-mode=on
enforce-gtid-consistency=true
log-slave-updates=1
[mysql]
prompt=db01 [\\d]>
EOF

slave1(db02):
mv /etc/my.cnf /tmp
cat > /etc/my.cnf <<EOF
```

五 ♥ 从库快速重新构建主从环境（mdp备份方式）

Bash | Copy

```
mysql> stop slave;
mysql> reset slave all;
mysql> show slave status;
Empty set (0.00 sec)

从库备份主库数据到本地，保证与主库数据一致
]# mysqldump -uroot -p123 -h 10.0.0.51 -P 3306 -A --master-data=2 --single-transaction -R -E --triggers >/tmp/
从库获取复制起点
]# grep  "\--\ CHANGE MASTER" /tmp/full.sql
-- CHANGE MASTER TO MASTER_LOG_FILE='mysql-bin.000006', MASTER_LOG_POS=1021;
从库设置复制起点
mysql -e \
"CHANGE MASTER TO \
  MASTER_HOST='10.0.0.51',\
  MASTER_USER='repl', \
  MASTER_PASSWORD='123', \
  MASTER_PORT=3306, \
  MASTER_LOG_FILE='mysql-bin.000006', \
  MASTER_LOG_POS=1021, \
  MASTER_CONNECT_RETRY=10;"
从库恢复备份的数据
mysql -e "source /tmp/full.sql"

```

%E4%BC%A0%E7%BB%9F%E4%B8%BB%E4%BB%8E%E6%9E%B6%E6%9E%84%E5%A4%9A%E7%A7%8D%E5%A4%87%E4%BB%BD%E6%96%B9%E5%BC%8F%E6%9E%84%E5%BB%BA%E2%88%9A%2