

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

```
1 # 三台虚拟机
2 10.0.0.51 db01
3 10.0.0.52 db02
4 10.0.0.53 db03
5 防火墙关闭
6
7 #清理环境:
8 pkill mysqld
9 rm -rf /data/3306/*
10 mkdir -p /data/3306/data /data/3306/binlog
11 chown -R mysql:mysql /data/*
12
13 # 准备配置文件
14 主库db01:
15 mv /etc/my.cnf /tmp
16 cat > /etc/my.cnf <<EOF
17 [mysqld]
18 basedir=/usr/local/mysql
19 datadir=/data/3306/data
20 socket=/tmp/mysql.sock
21 server_id=51
22 port=3306
23 secure-file-priv=/tmp
24 log_bin=/data/3306/binlog/mysql-bin
25 binlog_format=row
26 gtid-mode=on
27 enforce-gtid-consistency=true
28 log-slave-updates=1
29 [mysql]
30 prompt=db01 [\\d]>
31 EOF
32
33 slave1(db02):
34 mv /etc/my.cnf /tmp
35 cat > /etc/my.cnf <<EOF
36 [mysqld]
37 basedir=/usr/local/mysql
38 datadir=/data/3306/data
39 socket=/tmp/mysql.sock
40 server_id=52
41 port=3306
42 secure-file-priv=/tmp
43 log_bin=/data/3306/binlog/mysql-bin
44 binlog_format=row
45 gtid-mode=on
46 enforce-gtid-consistency=true
47 log-slave-updates=1
48 [mysql]
49 prompt=db02 [\\d]>
50 EOF
```

```
51
52 slave2(db03):
53 mv /etc/my.cnf /tmp
54 cat > /etc/my.cnf <<EOF
55 [mysqld]
56 basedir=/usr/local/mysql
57 datadir=/data/3306/data
58 socket=/tmp/mysql.sock
59 server_id=53
60 port=3306
61 secure-file-priv=/tmp
62 log_bin=/data/3306/binlog/mysql-bin
63 binlog_format=row
64 gtid-mode=on
65 enforce-gtid-consistency=true
66
67 log-slave-updates=1
68 [mysql]
69 prompt=db03 [\\d]>
70 EOF
71
72 初始化数据
73 mysqld --initialize-insecure --user=mysql --basedir=/usr/local/mysql --datadir=/data/3306/data
74
75 启动数据库
76 /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%';"
```

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

```

1 复制用户
2  mysql -e "create user repl@'10.0.0.%' identified with mysql_native_password by '123';
   grant replication slave on *.* to repl@'10.0.0.%';"
3 远程管理用户
4  mysql -e "create user root@'10.0.0.%' identified with mysql_native_password by '123';
   grant all on *.* to root@'10.0.0.%';"
5
6 检查创建的用户
7  [root@db01 data]# mysql -e "select user,host,plugin from mysql.user;"
8
9  +-----+-----+-----+
10 | user      | host      | plugin      |
11 +-----+-----+-----+
12 | repl      | 10.0.0.%  | mysql_native_password |
13 | root      | 10.0.0.%  | mysql_native_password |
14 | mysql.infoschema | localhost | caching_sha2_password |
15 | mysql.session | localhost | caching_sha2_password |
16 | mysql.sys   | localhost | caching_sha2_password |
17 | root      | localhost | caching_sha2_password |
18 +-----+-----+-----+
19

```

3.从库（52） 备份主库数据并恢复

```

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

```

4. 告诉从库（52） 复制起点信息

```
▼ Bash Copy
1 获取复制起点
2 [root@db02 ~]# grep "\--\ CHANGE MASTER" /tmp/full.sql
3 -- CHANGE MASTER TO MASTER_LOG_FILE='mysql-bin.000002', MASTER_LOG_POS=1187;
4 告知复制起点
5 [root@db02 data]# mysql -e \
6 "CHANGE MASTER TO \
7   MASTER_HOST='10.0.0.51', \
8   MASTER_USER='repl', \
9   MASTER_PASSWORD='123', \
10  MASTER_PORT=3306, \
11  MASTER_LOG_FILE='mysql-bin.000002', \
12  MASTER_LOG_POS=1187, \
13  MASTER_CONNECT_RETRY=10;"
```

5.从库（52）启动专用复制线程，并查看

```
▼ Bash Copy
1 [root@db02 ~]# mysql -e "start slave;"
2 [root@db02 ~]# mysql -e "show slave status \G"|grep "Running:"
3      Slave_IO_Running: Yes
4      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

```
▼ Bash Copy
1 [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

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

3.主库（51）创建备份目录并授权

```
▼ Bash | Copy
1 [root@db01 ~]# mkdir -p /data/backup
2 [root@db01 ~]# chown -R mysql. /data/backup/
```

4.主库（51）修改配置文件

```
▼ Bash | Copy
1 [root@db01 ~]# vim /etc/my.cnf
2 [client]
3 socket=/tmp/mysql.sock
```

5. 主库（51）进行全备

```
▼ Bash | Copy
1 [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 | Copy
1 [root@db03 opt]# scp -r /etc/my.cnf 10.0.0.53:/etc/my.cnf
2 root@10.0.0.53's password:
3 my.cnf 100% 271 380.8KB/s 00:00
4 [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

```
1 [mysqld]
2 basedir=/usr/local/mysql
3 datadir=/data/3306/data
4 socket=/tmp/mysql.sock
5 server_id=53      ---修改这里
6 port=3306
7 secure-file-priv=/tmp
8 log_bin=/data/3306/binlog/mysql-bin
9 binlog_format=row
10 gtid-mode=on
11 enforce-gtid-consistency=true
12 log-slave-updates=1
13 [mysql]
14 prompt=db01 [\d]>
15 [client]
16 socket=/tmp/mysql.sock
```

8.从库（53） 恢复主库（51） 数据

```
1 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
2 所以我们这里删除从库53的数据，再恢复成主库数据
3 0. 清空数据
4 [root@db03 ~]# pkill mysqld
5 [root@db03 ~]# rm -rf /data/3306/data/*
6 [root@db03 ~]# rm -rf /data/3306/logs/*
7 [root@db03 ~]# rm -rf /data/3306/binlog/*
8 1. CR保证数据一致
9 [root@db03 ~]# xtrabackup --defaults-file=/etc/my.cnf --socket=/tmp/mysql.sock --user=root --password=123 --c
10 2. 恢复数据
11 [root@db03 ~]# xtrabackup --defaults-file=/etc/my.cnf --socket=/tmp/mysql.sock --user=root --password=123 --c
12 3. 将恢复数据的目录进行授权
13 [root@db03 full]# chown -R mysql:mysql /data/*
14 4. 重启服务
15 [root@db03 full]# /etc/init.d/mysqld start
```

9.告诉从库（53） 复制起点信息


```
1 获取本次拷贝到的binlog位置点
2 [root@db03 full]# cat /root/full/xtbackup_binlog_info
3 binlog.000015 156
4 [root@db03 data]# mysql -uroot -p123 -e \
5 "CHANGE MASTER TO \
6   MASTER_HOST='10.0.0.51', \
7   MASTER_USER='repl', \
8   MASTER_PASSWORD='123', \
9   MASTER_PORT=3306, \
10  MASTER_LOG_FILE='mysql-bin.000015', \
11  MASTER_LOG_POS=156, \
12  MASTER_CONNECT_RETRY=10;"
```

10.从库（52）启动专用复制线程，并查看

```
1 db03 [(none)]>start slave;
2 [root@db02 ~]# mysql -e "show slave status \G"|grep "Running:"
3         Slave_IO_Running: Yes
4         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

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

2.从库操作

Bash | Copy

```
1 0.加载插件, 创建克隆接受者用户, 授权, 设置白名单
2 [root@db03 ~]# mysql -e "INSTALL PLUGIN clone SONAME 'mysql_clone.so';create user test2@'%' identified by '1
3 23';grant clone_admin on *.* to test2@'%';SET GLOBAL clone_valid_donor_list='10.0.0.51:3306';"
4 1.克隆操作
5 [root@db03 ~]# mysql -utest2 -p123 -h10.0.0.53 -P3306 -e "CLONE INSTANCE FROM test1@'10.0.0.51':3306 IDENTI
6 FIED BY '123';"
7 2.查看从库复制起点
8 [root@db03 ~]# mysql -uroot -p123 -e "SELECT BINLOG_FILE, BINLOG_POSITION FROM performance_schema.clone_sta
9 tus;"
10 mysql: [Warning] Using a password on the command line interface can be insecure.
11 +-----+-----+
12 | BINLOG_FILE      | BINLOG_POSITION |
13 +-----+-----+
14 | mysql-bin.000003 |          923    |
15 +-----+-----+
```

3.从库启动主从复制

Bash | Copy

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

4.查看

Bash | Copy

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

第四种方式 GTID搭建新的主从环境

Bash | Copy

```
1 # 三台虚拟机
2
3 10.0.0.51 db01
4 10.0.0.52 db02
5 10.0.0.53 db03
6
7 防火墙关闭
8
9
10 #清理环境:
11 pkill mysqld
12
13 rm -rf /data/3306/*
14
15 mkdir -p /data/3306/data /data/3306/binlog
16
17 chown -R mysql:mysql /data/*
18
19
20 # 准备配置文件
21 主库db01:
22
23 mv /etc/my.cnf /tmp
24
25 cat > /etc/my.cnf <<EOF
26 [mysqld]
27 basedir=/usr/local/mysql
28 datadir=/data/3306/data
29 socket=/tmp/mysql.sock
30 server_id=51
31
32 port=3306
33
34 secure-file-priv=/tmp
35
36 log_bin=/data/3306/binlog/mysql-bin
37
38 binlog_format=row
39
40 gtid-mode=on
41 enforce-gtid-consistency=true
42 log-slave-updates=1
43
44 [mysql]
45 prompt=db01 [\\d]>
46 EOF
47
48
49 slave1(db02):
50
51 mv /etc/my.cnf /tmp
52
53 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
1  mysql> stop slave;
2  mysql> reset slave all;
3  mysql> show slave status;
4  Empty set (0.00 sec)
5
6  从库备份主库数据到本地，保证与主库数据一致
7  ]# mysqldump -uroot -p123 -h 10.0.0.51 -P 3306 -A --master-data=2 --single-transaction -R -E --triggers >/tmp/
8  从库获取复制起点
9  ]# grep "\-\ CHANGE MASTER" /tmp/full.sql
10 -- CHANGE MASTER TO MASTER_LOG_FILE='mysql-bin.000006', MASTER_LOG_POS=1021;
11 从库设置复制起点
12 mysql -e \
13 "CHANGE MASTER TO \
14     MASTER_HOST='10.0.0.51', \
15     MASTER_USER='repl', \
16     MASTER_PASSWORD='123', \
17     MASTER_PORT=3306, \
18     MASTER_LOG_FILE='mysql-bin.000006', \
19     MASTER_LOG_POS=1021, \
20     MASTER_CONNECT_RETRY=10;"
21 从库恢复备份的数据
22 mysql -e "source /tmp/full.sql"
23
```

%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