



链滴

mysql 主从同步，同步阿里云 mysql(RDS) 数据

作者: [yexuejc](#)

原文链接: <https://ld246.com/article/1614610300341>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



1. 先建立从数据库, 这里用docker创建

```
docker run -d \  
--name mysql-slave \  
-e MYSQL_ROOT_PASSWORD=123456 \  
-v /docker/mysql-slave/conf:/etc/mysql \  
-v /docker/mysql-slave/data:/var/lib/mysql \  
-v /docker/mysql-slave/logs:/var/log/mysql \  
-p 3308:3306 \  
-d mysql:5.7.26
```

2. 同步前要先把RDS的数据库备份一份到从数据库, 因为RDS已经有数据了。使用 `mysqldump` 备份RDS的数据库

```
> docker exec -it mysql-slave /bin/bash  
> mysqldump -hxxxxxxxxx.mysql.rds.aliyuncs.com -uroot -pPassword --databases 需要的数据库(如`test test1 test2 test3`一般排除 information_schema、performance_schema 和 mysql) > db.sql  
> //也可以使用如下备份全部数据库  
> mysqldump -hxxxxxxxxx.mysql.rds.aliyuncs.com -uroot -pPassword --all-databases > db.sql
```

3. 在从库恢复备份

```
> mysql -uroot -p123456  
> mysql> source < db.sql
```

4. 恢复备份后, 使用 `show global variables like '%gtid%'` 在主库(RDS)上查询gtid_purged的, 如`8177958d-a3b3-11ea-8f76-00163e0fbc31:1-2372913`

Variable_name	Value
binlog_gtid_simple_recovery	ON
enforce_gtid_consistency	ON
gtid_executed	8177958d-a3b3-11ea-8f76-00163e0fbc31:1-2378653
gtid_executed_compression_period	1000
gtid_mode	ON
gtid_owned	
gtid_purged	8177958d-a3b3-11ea-8f76-00163e0fbc31:1-2372913
opt_rds_last_error_gtid	ON
rds_push_gtid_default_interval	10
session_track_gtids	OFF

5. 使用 `show global variables like '%gtid%\G` 在从库查询gtid_purged的值是否为空，不为空，用`reset master` 清空

```

mysql> show global variables like '%gtid%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| binlog_gtid_simple_recovery | ON |
| enforce_gtid_consistency | ON |
| gtid_executed | bc07a35e-7a91-11eb-b28e-0242ac11000d:1-5 |
| gtid_executed_compression_period | 1000 |
| gtid_mode | ON |
| gtid_owned | |
| gtid_purged | |
| session_track_gtids | OFF |
+-----+-----+
8 rows in set (0.10 sec)

mysql> stop slave;
Query OK, 0 rows affected (0.03 sec)

mysql> reset slave;
Query OK, 0 rows affected (0.08 sec)

mysql> reset master;
Query OK, 0 rows affected (0.12 sec)

mysql> show global variables like '%gtid%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| binlog_gtid_simple_recovery | ON |
| enforce_gtid_consistency | ON |
| gtid_executed | |
| gtid_executed_compression_period | 1000 |
| gtid_mode | ON |
| gtid_owned | |
| gtid_purged | |
| session_track_gtids | OFF |
+-----+-----+
8 rows in set (0.00 sec)

mysql> set @@global.gtid_purged='8177958d-a3b3-11ea-8f76-00163e0fbc31:1-2372913'
->
-> ;
Query OK, 0 rows affected (0.06 sec)

mysql> show global variables like '%gtid%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| binlog_gtid_simple_recovery | ON |
| enforce_gtid_consistency | ON |
| gtid_executed | 8177958d-a3b3-11ea-8f76-00163e0fbc31:1-2372913 |
| gtid_executed_compression_period | 1000 |
| gtid_mode | ON |
| gtid_owned | |
| gtid_purged | 8177958d-a3b3-11ea-8f76-00163e0fbc31:1-2372913 |
| session_track_gtids | OFF |
+-----+-----+
8 rows in set (0.00 sec)

```

6. 同步主从信息

```

docker exec -it mysql-slave mysql -P3306 -uroot -pPassw0rd -e "CHANGE MASTER TO MASTER_HOST='xxxxxxx.mysql.rds.aliyuncs.com', MASTER_PORT=3306, MASTER_USER='root', MASTER_PASSWORD='Password', MASTER_AUTO_POSITION=1;" \
-e "START SLAVE;"

```

```
docker exec -it mysql-slave mysql -P3306 -uroot -pPassw0rd -e "show slave status\G"
```

```
mysql> show slave status\G
***** 1. row *****
      Slave_IO_State: Waiting for master to send event
      Master_Host: [REDACTED].mysql.rds.aliyuncs.com
      Master_User: [REDACTED]
      Master_Port: [REDACTED]
      Connect_Retry: 60
      Master_Log_File: mysql-bin.001097
      Read_Master_Log_Pos: 325303
      Relay_Log_File: f35956ae5c2e-relay-bin.000002
      Relay_Log_Pos: 128480
      Relay_Master_Log_File: mysql-bin.001094
      Slave_IO_Running: Yes
      Slave_SQL_Running: Yes
      Replicate_Do_DB:
      Replicate_Ignore_DB: mysql,information_schema,performance_schema
      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: 128307
      Relay_Log_Space: 1637463
      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: 74550
      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: 2586290671
      Master_UUID: 8177958d-a3b3-11ea-8f76-00163e0fbc31
      Master_Info_File: /var/lib/mysql/master.info
      SQL_Delay: 0
      SQL_Remaining_Delay: NULL
      Slave_SQL_Running_State: Reading event from the relay log
      Master_Retry_Count: 86400
      Master_Bind:
      Last_IO_Error_Timestamp:
      Last_SQL_Error_Timestamp:
      Master_SSL_Crl:
      Master_SSL_Crlpath:
      Retrieved_Gtid_Set: 8177958d-a3b3-11ea-8f76-00163e0fbc31:2372914-2378577
      Executed_Gtid_Set: 8177958d-a3b3-11ea-8f76-00163e0fbc31:1-2373357
      Auto_Position: 1
      Replicate_Rewrite_DB:
      Channel_Name:
      Master_TLS_Version:
1 row in set (0.00 sec)

mysql> docker exec -it mysql-slave mysql [ ]
```

7. 按照常规操作，RDS的一般会报错，主要问题在 4、5步，重新走一下就ok

参考

[解决mysql开启GTID主从同步出现1236错误问题](#)

[使用 RDS for MySQL 配置到自建数据库的主从复制](#)

[阿里云RDS与ECS自建库搭建主从复制](#)