

1、主机 mysql-master 配置

创建主机镜像

```
docker run -d -p 3301:3306 --privileged=true -v /mydata/mysql-master/log:/var/log/mysql -v /mydata/mysql-master/data:/var/lib/mysql -v /mydata/mysql-master/conf:/etc/mysql/conf.d -e MYSQL_ROOT_PASSWORD=95cf663915092f47 --name mysql-master mysql:8.0.30
```

宿主机：

vim /mydata/mysql-master/conf/mysql.cnf

```
[mysqld]
#服务id,在一个主从复制集群中要唯一,值范围1-255
server-id = 1
#开启log-bin日志,非常重要,复制原理也是基于这一个,后面的值随便写我这里就写mysql-master-bin了
log-bin = mysql-master-bin
#从机复制时,忽略的数据库,也就是说这里配置的数据库不会被从机同步
binlog-ignore-db = mysql,information_schema,performance_schema,sys
```

登录数据库 mysql-master

```
# 修改root用户密码插件
ALTER USER 'root'@'%' IDENTIFIED WITH mysql_native_password BY '95cf663915092f47';
flush privileges;
```

重启

```
docker restart mysql-master
docker exec -it mysql-master /bin/bash
mysql -uroot -p
enter password:
```

```
mysql> show master status;
```

```
+-----+-----+-----+-----+
+-----+-----+
| File                | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
|                     | Executed_Gtid_Set |
+-----+-----+-----+-----+
| mysql-master-bin.000001 | 157 | | |
mysql,information_schema,performance_schema,sys | |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

?

```
CREATE USER 'slave'@'%' IDENTIFIED BY '123456';
GRANT REPLICATION SLAVE, REPLICATION CLIENT ON *.* TO 'slave'@'%';
```

2、从机 mysql-slave配置

- 从机配置

在宿主机中输入命令得到 mysql-master 容器的ip

```
[root@aliyun ~]# docker inspect mysql-master | grep IPAddress
    "SecondaryIPAddresses": null,
    "IPAddress": "172.17.0.4",
    "IPAddress": "172.17.0.4",
```

创建从机镜像

```
docker run -d -p 3302:3306 --privileged=true -v /mydata/mysql-slave/log:/var/log/mysql -v /mydata/mysql-slave/data:/var/lib/mysql -v /mydata/mysql-slave/conf:/etc/mysql/conf.d -e MYSQL_ROOT_PASSWORD=0c592c3f8cc7611b --name mysql-slave mysql:8.0.30
```

宿主机:

vim /mydata/mysql-slave/conf/mysql.cnf

```
[mysqld]
server-id = 2
```

重启从机mysql-slave

```
docker restart mysql-slave
docker exec -it mysql-slave /bin/bash
mysql -uroot -p
Enter password

mysql> change master to master_host='172.17.0.4',
master_user='root',master_password='95cf663915092f47',master_log_file='mysql-
master-bin.000001',master_log_pos=157;

mysql> # 启动slave命令
mysql> start slave;

mysql> # 查看slave状态
mysql> show slave status\G;
```

成功

```
mysql> start slave
-> ;
Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql> show slave status\G
***** 1. row *****
      Slave_IO_State: Waiting for source to send event
        Master_Host: 172.17.0.4
        Master_User: root
        Master_Port: 3306
        Connect_Retry: 60
        Master_Log_File: mysql-master-bin.000001
        Read_Master_Log_Pos: 157
        Relay_Log_File: 8dcdd96cac7b-relay-bin.000002
        Relay_Log_Pos: 333
        Relay_Master_Log_File: mysql-master-bin.000001
        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: 157
        Relay_Log_Space: 550
        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:
```

3、参考博客

[docker中MySQL8主从复制配置搭建（超清楚）](#)