# Mysql 主备份部署

## 功能实现：

Mysql 主从备份：主数据库进行增删改操作时候，从数据库自动进行同步操作

## 实现原理

主数据库开启日志写入功能，每次操作数据库都会同步到文件，对应的position（偏移量）字段会变化

从库监听到字段变化，然后读取主库的文件，同步到从库

## 部署详细：

### 主msyql 部署

**Mysq版本5.7.16**

#### Msql.cnf重点配置

注释：mysql-bin：日志文件名称 server-id唯一id

log-bin=mysql-bin

server-id=111111

#### 全部cnf配置

[client]

default-character-set=utf8mb4

[mysqld]

port=3306

datadir=/var/lib/mysql/data

log-bin=mysql-bin

server-id=111111

symbolic-links=0

explicit\_defaults\_for\_timestamp=true

character-set-server = utf8mb4

sql\_mode=STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION

collation-server = utf8mb4\_unicode\_ci

init\_connect='SET NAMES utf8mb4'

lower\_case\_table\_names=1

max\_connections=800

#SQL数据包发送的大小，如果有BLOB对象建议修改成1G

max\_allowed\_packet = 200M

[mysql]

default-character-set=utf8mb4

#### docker-compose 安装mysql

version: "3"

services:

mysql:

restart: always

image: mysql:5.7.16

container\_name: mysql

environment:

- "MYSQL\_ROOT\_PASSWORD=123456"

- "TZ=Asia/Shanghai"

ports:

- 33066:3306

volumes:

- ./volumes/mysql/data:/var/lib/mysql/data

- ./volumes/mysql/conf/my.cnf:/etc/mysql/my.cnf

#### 启动后执行sql

1 创建从库使用的登陆用户

Sql: create user '用户名'@'%' identified by ‘密码‘;

2 授权从库访问日志文件 ，FILE权限指的是对服务器主机上文件的访问

Sql: grant FILe on \*.\* to '用户名'@'从库ip' identified by '密码';

3 授权从库访问权限

Sql: grant replication slave on \*.\* to '用户名'@'从库ip' identified by '123456'

4 刷新日志

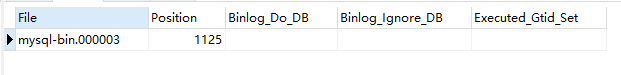
Sql: flush logs

5 刷新权限

Sql: flush privileges;

查询日志文件状态

Sql: show master status;



此时不要对数据库内容进行操作，否色 position会有变化

（记住File 字段，和position字段，后面从库配置用到）

### 从mysql部署

#### 安装mysql 方式同上

#### cnf配置文件配置同上，只要改 server-id 和上面不同即可

#### 启动后执行sqls

1 停止slave功能

Sql: stop slave;

2 配置与主库的连接

注释：master\_log\_file：主mysql 执行sql(show master status;)查询出来的file字段的值。

master\_log\_pos：主mysql 执行sql(show master status;)查询出来的position字段的值

change master to master\_host='192.168.1.66',master\_user='用户名',master\_port=端口,master\_password='密码',master\_log\_file='mysql-bin.000003',master\_log\_pos=154;

5 打开 slave功能

Sql: start slave;

3 查询配置状态

Sql: show slave status;

4 如果上一个sql查询出来的slave\_sql\_running=no (未执行备份动作) 则执行

Sql: SET GLOBAL SQL\_SLAVE\_SKIP\_COUNTER=1; START SLAVE;

完成

修改主库数据进行测试