## 主从复制学习资料

### 1 简介

MySQL数据库支持单向、双向、链式级联、环状等不同业务场景的复制。在复制过程中，一台服务器充当主服务器（Master），接收来自用户的内容更新，而一个或多个其他的服务器充当从服务器（Slave），接收来自主服务器binlog文件的日志内容，解析出SQL，重新更新到从数据库，使得主从服务器的数据达到一致。

如果设置了链式级联复制，那么从服务器本身除了充当从服务器外，也会同时充当其下面从服务器的主服务器，链式级联复制类似A—>B—>C的复制形式。

MySQL的主从复制的主要场景有以下几个：

1）从服务器作为主服务器的实时数据备份；

2）主从服务器实现读写分离（主写从读），从服务器实现负载均衡；

3）把多个从服务器根据业务重要性进行拆分访问；

### 2 MySQL主从复制原理介绍

MySQL的主从复制是一个异步的复制过程（一般情况下感觉是实时的），数据将从一个MySQL数据库（Master）复制到另外一个MySQL数据库（Slave），在Master与Slave之间实现整个主从复制的过程是由三个线程参与完成的。其中有两个线程（SQL线程和I/O线程）在Slave端，另外一个线程（I/O线程）在Master端。

要实现MySQL的主从复制，首先必须打开Master端的binlog记录功能，否则就无法实现。因为整个复制过程实际上就是Slave端从Master端获取binlog日志，然后再在Slave上以相同的顺序执行获取的binlog日志中所记录的各种SQL操作。

要打开MySQL的binlog记录功能，可以通过在MySQL的配置文件my.cnf中的mysqld模块（[mysqld]标识后的参数部分）增加"log-bin"参数来实现。

### 3 一主一从参数配置

前提条件：

1．数据库版本尽可能一致，本文采用mysql-5.5.6

2. 数据库安装方式采用源码安装方式，其他安装方式，可能导致一些源文件找不到

172.16.119.6

172.16.119.4

1. master配置mysql的配置文件，/etc/my.cnf, 执行vi my.cnf， 对应位置添加如下配置：

[mysqld]

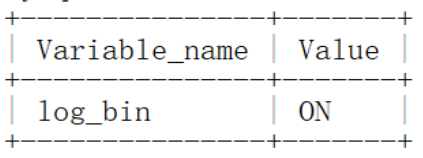
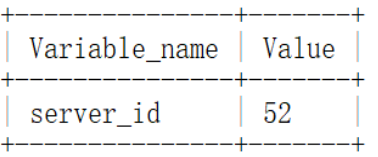
log-bin=/mnt/ns/MySQL/mysql-bin

binlog\_ignore\_db=mysql

replicate-do-db = test #要同步的数据库，默认所有库

server\_id = 1

1. 重启msater数据， 执行 service mysql restrt
2. 进入mysql数据库, mysql -uyourUserName -ppassword,并查看之前的参数并是否正确，分别使用命令：show variables like server\_id”, show variables like “log\_bin”, 出现如下格式者表示正确：



1. 在master上建立用于主从复制的账号，检查账号建立成功

grant replication slave on \*.\* to 'yourUserName'@'172.16.119.%' identified by 'yourPassword';

('yourUserName, 'yourPassword' 表示你将要将要新建的账号和密码，与172.16.119.% **从机的数据库无关**)

flush privileges;

1. 配置从数据库

/etc/my.cnf, 执行vi my.cnf， 对应位置添加如下配置：

# log-bin （关闭log-bin）

server\_id = 2 （不能重复）

1. 重启从数据库 service mysql restart
2. 对slave 数据库操作步骤3），查看配置参数的正确性
3. 从库上操作，配置复制参数

Mysql中执行如下命令：

CHANGE MASTER TO

MASTER\_HOST='172.16.119.3',（master 地址）

MASTER\_PORT=3306, （master 数据库端口）

MASTER\_USER='master', （master 用户名，即步骤4）之前在master上新建账号）

MASTER\_PASSWORD= 'master', （master 即步骤4）之前在master上新建账号密码）

MASTER\_LOG\_FILE='mysql-bin.000001', （master 地址上的日志文件）

MASTER\_LOG\_POS= 698962728; （master 日志位置）

MASTER\_LOG\_FILE， MASTER\_LOG\_POS 参数是依据master 得到，获取方式：

在master mysql 中执行： show master status;

这个操作实际上就是把用户、密码以及主库的其他信息写入到从库的master.info文件去

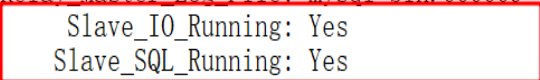
9) 从库上开启同步开关

数据库中执行：start slave;

验证配置正确性：

执行命令 show slave status\G;

显示如下则配置正确：



10) 测试

在主数据库上新建数据库，表，表插入数据，检查slave是否同步。

11）错误排除

出现问题： Mysql主从同步时Slave\_IO\_Running：Connecting ; Slave\_SQL\_Running：Yes的情况故障排除

可能原因：

1. 网络不通

排除方法，ping master 地址，检测是否能ping 通；如果不能ping 通，检查防火墙，数据库是否开启远程可连接配置等，

1. 账户密码错误

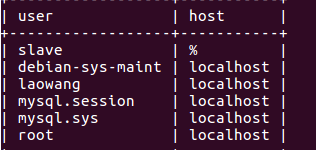
排除方法：主服务器创建了账号slave，密码slave的权限账号，在主服务器可以登录slave帐号，排除帐号密码问题；

1. Mysql配置文件问题

排除方法, 确认log\_bin和server-id设置无错误;

1. Mysql用户权限

排除方法，查看创建的账号的host是否为%或则对应的地址



在slave服务器上登录master服务器创建的帐号

mysql -uxxx -pxxx -hmaster\_ip

### 4 一主一主参数配置

1. 配置服务器A参数

log-bin = mysql-bin

server-id = 1

expire-logs-days = 100

replicate-do-db = test

binlog-ignore-db = mysql

binlog-ignore-db = information\_schema

auto-increment-increment = 2

auto-increment-offset = 1

1. 配置服务器B的参数

log-bin = mysql-bin

server-id = 2

expire-logs-days = 100

replicate-do-db = test

binlog-ignore-db = mysql

binlog-ignore-db = information\_schema

auto-increment-increment = 2

auto-increment-offset = 2

注：二都只有server-id不同和 auto-increment- offset不同

auto-increment-offset是用来设定数据库中自动增长的起点的，回为这两能服务器都设定了一次自动增长值2，所以它们的起点必须得不同，这样才能避免两台服务器数据同步时出现主键冲突

replicate-do-db 指定同步的数据库，我们只在两台服务器间同步test数据库

另：auto-increment-increment的值应设为整个结构中服务器的总数，本案例用到两台服务器，所以值设为2

1. 相互授权用户(在A服务器授权一个允许B访问的用户，反之亦然)

在服务器A执行：

GRANT REPLICATION SLAVE ON \*.\* TO 'mysync'@'172.16.119.6' IDENTIFIED BY PASSWORD '123456';

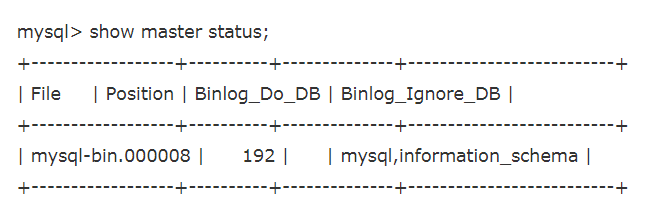
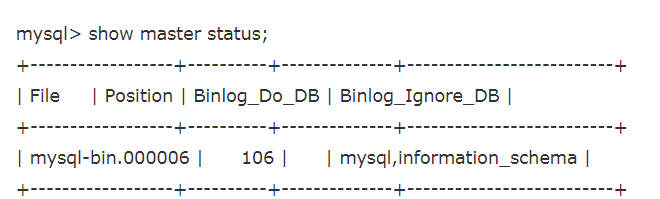
flush privileges;

在服务器B执行：

GRANT REPLICATION SLAVE ON \*.\* TO 'mysync'@'172.16.119.4' IDENTIFIED BY PASSWORD '123456';

flush privileges;

4) 互告bin-log信息



A

B

以A为主：在B上执行：

change master to master\_host='172.16.119.6',master\_user='mysync',master\_password='123456',master\_log\_file='mysql-bin.000006',master\_log\_pos=106;

以B为主：在A上执行：

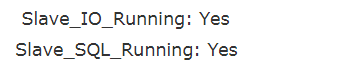
change master to master\_host='172.16.119.4',master\_user='mysync',master\_password='123456',master\_log\_file='mysql-bin.000008',master\_log\_pos=192;

5) 开启进程并查看状态

start slave

show slave status\G

出现如下所示状态：



CHANGE MASTER TO MASTER\_HOST='172.16.119.3',MASTER\_PORT=3306,MASTER\_USER=master, MASTER\_PASSWORD= master,MASTER\_LOG\_FILE='mysql-bin.000001', MASTER\_LOG\_POS= 698962728;