

如何为MySQL搭建一套主从复制架构？（1）

今天我们来讲解一下如何为MySQL搭建一套主从复制架构，其实这个MySQL主从复制的原理之前也都讲过了，大致来说，就是主库接受增删改操作，把增删改操作binlog写入本地文件，然后从库发送请求来拉取binlog，接着在从库上重新执行一遍binlog的操作，就可以还原出一样的数据了。

那么搭建的时候肯定是需要两台机器的，一台机器放主库，一台机器放从库，至于主库和从库如何安装和启动？这个就不在我们讲的范围了，随便网上一搜就大把的MySQL安装步骤，我们这里就讲解搭建主从复制架构要做的一些配置。

首先呢，要确保主库和从库的server-id是不同的，这个是必然的，其次就是主库必须打开binlog功能，你必须打开binlog功能主库才会写binlog到本地磁盘，接着就可以按如下步骤在主库上执行一通操作了。

首先在主库上要创建一个用于主从复制的账号：

```
create user 'backup_user'@'192.168.31.%' identified by 'backup_123';  
grant replication slave on . to 'backup_user'@'192.168.31.%';  
flush privileges;
```

接着你要考虑一个问题，假设你主库都跑了一段时间了，现在要挂一个从库，那从库总不能把你主库从0开始的所有binlog都拉一遍吧！这是不对的，此时你就应该在凌晨的时候，在公司里直接让系统对外不可用，说是维护状态，然后对主库和从库做一个数据备份和导入。

可以使用如下的mysqldump工具把主库在这个时刻的数据做一个全量备份，但是此时一定是不能允许系统操作主库了，主库的数据此时是不能有变动的。

```
/usr/local/mysql/bin/mysqldump --single-transaction -uroot -proot --master-data=2 -A >  
backup.sql
```

注意，mysqldump工具就在你的MySQL安装目录的bin目录下，然后用上述命令就可以对你主库所有的数据都做一个备份，备份会以SQL语句的方式进入指定的backup.sql文件，只要执行这个backup.sql文件，就可以恢复出来跟主库一样的数据。

至于上面命令里的--master-data=2，意思就是说备份SQL文件里，要记录一下此时主库的binlog文件和position号，这是为主从复制做准备的。

接着你可以通过scp之类的命令把这个backup.sql文件拷贝到你的从库服务器上去就行了，这个scp命令怎么用就不用我们来说了，大家随便网上查一下就知道这个命令是怎么用的了，这个是很简单的。

接着操作步骤转移到从库上去执行，在从库上执行如下命令，把backup.sql文件里的语句都执行一遍，这就相当于把主库所有的数据都还原到从库上去了，主库上的所有database、table以及数据，在从库里全部都有了。

接着在从库上执行下面的命令去指定从主库进行复制。

```
CHANGE MASTER TO MASTER_HOST='192.168.31.229',  
MASTER_USER='backup_user',MASTER_PASSWORD='backup_123',MASTER_LOG_FILE='mysql-  
bin.000015',MASTER_LOG_POS=1689;
```

可能有人会疑惑，上面的master机器的ip地址我们是知道的，master上用于执行复制的用户名和密码是我们自己创建的，也没问题，但是master的binlog文件和position是怎么知道的？这不就是之前我们mysqldump导出的backup.sql里就有，大家在执行上述命令前，打开那个backup.sql就可以看到如下内容：

```
MASTER_LOG_FILE='mysql-bin.000015',MASTER_LOG_POS=1689
```

然后你就把上述内容写入到主从复制的命令里去了。

接着执行一个开始进行主从复制的命令：start slave，再用show slave status查看一下主从复制的状态，主要看到Slave_IO_Running和Slave_SQL_Running都是Yes就说明一切正常了，主从开始复制了。

接着就可以在主库插入一条数据，然后在从库查询这条数据，只要能够在从库查到这条数据，就说明主从复制已经成功了。

这仅仅是最简单的一种主从复制，就是异步复制，就是之前讲过的那种原理，从库是异步拉取binlog来同步的，所以肯定会出现短暂的主从不一致的问题的，比如你在主库刚插入数据，结果在从库立马查询，可能是查不到的。

后续我们会再继续讲解MySQL主从同步的其他几种方式。

End