

MySQL 主从复制

mysql 的主从复制实现的原理就是 binlog 日志，那么我们主节点负责数据库写操作，而从节点负责读操作，这样在从节点上不需要使用事务，能够大大提高数据库的性能。那么这个时候面临的问题就是从节点如何来同步主节点数据的问题，就用到了我们之前说过的 binlog 日志。从节点通过 binlog 日志来与主节点进行数据同步。

三台节点，一个 master，两个 slave，规划如下

master	192.168.153.201
slave1	192.168.153.152
slave2	192.168.153.153

三台节点都是使用 CentOS6.9 操作系统，数据库使用的是 mysql5.7。

配置 master

1. 创建用户并授权，比如我们的用户叫做 repl，创建用户语法如下：

```
create user 'repl'
```

```
mysql> create user 'repl' identified by 'repl';  
Query OK, 0 rows affected (0.08 sec)
```

2. 授权。用户必须具备 **replication slave** 权限，除此之外不需要其他的权限

```
grant replication slave on *.* to 'repl'@'192.168.153.%' identified by 'repl';
```

```
mysql> grant replication slave on *.* to 'repl'@'192.168.153.9';  
Query OK, 0 rows affected, 1 warning (0.00 sec)
```

3. 开启 binlog 日志

这里可以参考开启 binlog 日志的文档。

至此，主服务器就配置好了。下面来进行配置从服务器

slave1

1. 修改 my.cnf 配置文件

```
[mysqld]server-id=2relay-log-index=slave-relay-bin.indexrelay-log=slave-relay-bin
```

2. 连接 Master

```
change master to master_host='192.168.0.104', //Master 服务器 Ip  
master_port=3306,  
master_user='repl',  
master_password='mysql',
```

```
master_log_file='master-bin.000001', //Master 服务器产生的日志
master_log_pos=0;
```

```
change master to
master_host='192.168.153.201',master_port=3306,master_user='repl',master_passwo
rd='repl',master_log_file='/var/lib/mysql/mysql-bin.000001',master_log_pos=0;
```

3. 查看状态

```
show slaves status\G
```

```
mysql> show slave status\G
```

```
***** 1. row *****
Slave_IO_State:
  Master_Host: 192.168.153.201
  Master_User: repl
  Master_Port: 3306
  Connect_Retry: 60
  Master_Log_File: /var/lib/mysql/mysql-bin.000000
  Read_Master_Log_Pos: 4
  Relay_Log_File: slave-relay-bin.000002
  Relay_Log_Pos: 4
  Relay_Master_Log_File: /var/lib/mysql/mysql-bin.000000
  Slave_IO_Running: No
  Slave_SQL_Running: Yes
  Replicate_Do_DB:
  Replicate_Ignore_DB:
  Replicate_Do_Table:
  Replicate_Ignore_Table:
  Replicate_Wild_Do_Table:
```

这里关注两个地方，第一个是 Slave_IO_Running，第二个是 Slave_SQL_Running。这两个必须都为 yes 才可以启动从节点。那么问题出现在哪里呢，后面会有日志提示

```

Master_SSL_Verify_Server_Cert: No
      Last_IO_Errno: 1593
      Last_IO_Error: Fatal error: The slave I/O thread
JIDs; these UUIDs must be different for replication to work.
      Last_SQL_Errno: 0
      Last_SQL_Error:
Replicate_Ignore_Server_Ids:
      Master_Server_Id: 12345
      Master_UUID:
      Master_Info_File: /var/lib/mysql/master.info
      SQL_Delay: 0
      SQL_Remaining_Delay: NULL
Slave_SQL_Running_State: Slave has read all relay log; v
      Master_Retry_Count: 86400
      Master_Bind:
      Last_IO_Error_Timestamp: 170708 15:38:54
      Last_SQL_Error_Timestamp:
      Master_SSL_Crl:
      Master_SSL_Crlpath:

```

这里是说两台节点的 uuid 一致，要让这两台机器的 uuid 不一致才行，那么 uuid 在哪呢，在我们数据目录下 auto.cnf 文件中，只要修改这个文件就可以了。

```

[root@c1 mysql]# cat auto.cnf
[auto]
server-uuid=8114ab68-635d-11e7-ab7c-000c29e9d9fb

```

再来查看状态，然后就为 yes 了。

4. 启动

```
start slave
```

5. 关闭

```
stop slave
```

6. 重置

```
reset slave
```