

搭建 MySQL 主从数据库的步骤

1 在 Windows 上搭建 MySQL 主从数据库.....	2
1.1 资源地址.....	2
1.2 安装.....	2
1.3 设置环境变量.....	2
1.4 配置主库.....	2
1.5 导入数据库.....	3
1.6 建立从库.....	4
2 在 Linux 上搭建 Mysql 主从数据库.....	6
2.1 资源地址.....	6
2.2 清理环境.....	6
2.3 安装.....	6
2.4 设置账户安全.....	7
2.5 设置字符集.....	7
2.6 验证安装.....	7
2.7 导入数据库.....	8
2.8 创建从库.....	8
2.9 修改配置文件.....	8
2.10 创建复制账号.....	9
2.11 配置从库.....	10

1 在 Windows 上搭建 MySQL 主从数据库

1.1 资源地址

官网地址:

<http://dev.mysql.com/downloads/>

1.2 安装

安装 mysql，选择“开发者”类型就行了，不要定制，一路缺省就好了。注意不要将 mysql 安装在包含汉字的路径下。

安装过程中需要设置 root 的密码。

安装完毕，mysql 服务将自动启动，MySQL Notifier 也在系统托盘（右下角）开始工作。

1.3 设置环境变量

将以下路径加入系统环境变量“PATH”中（替换为实际安装目录）

[C:\Program Files\MySQL\MySQL Server 5.7\bin](#)

1.4 配置主库

MySQL 正常安装完毕以后会自动创建和配置好一个缺省的数据库实例，并自动设置为系统服务。我们把这个缺省实例作为主库。

执行以下步骤以配置主库：

1) 在系统服务列表中右键 MySQL 服务名选择“停止”。或者以管理员账户运行以下命令（替换为实际服务名）：

```
net stop MySQL57
```

2) 找到主库的数据目录，例如“[C:\ProgramData\MySQL\MySQL Server 5.7](#)”。

3) 编辑主库数据目录下的配置文件“[my.ini](#)”，添加或者修改以下参数：

```
[client]
```

```
port=3306
```

```
[mysqld]
```

```
datadir=C:/ProgramData/MySQL/MySQL Server 5.7/Data
```

```
port=3306
```

```
server-id=1
```

```
character-set-server=utf8
```

```
collation-server = utf8_unicode_ci
```

```
sql-mode="STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION"
```

```
log-bin=mysql-bin
```

```
innodb_flush_log_at_trx_commit=1
sync_binlog=1
gtid_mode=on
enforce_gtid_consistency=on
binlog_format=ROW
binlog-do-db=beidou
expire_logs_days=10
binlog_row_image=minimal

max_sp_recursion_depth=100
event_scheduler=1
```

4) 在系统服务列表中右键 Mysql 服务名选择“启动”。或者以管理员账户运行以下命令（替换为实际服务名）：

```
net start MySQL57
```

5) 登入主库：

```
mysql -u root -p
```

6) 输入以下命令（xxxxxx 换成实际密码），以创建数据复制账户：

```
CREATE USER 'repl'@'localhost' IDENTIFIED BY 'xxxxxx';
GRANT REPLICATION SLAVE ON *.* TO 'repl'@'localhost';
SHOW MASTER STATUS;
```

以下为示意图：

```
C:\Users\mara>mysql -u root -p beidou
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.7.9-log MySQL Community Server (GPL)
Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE USER 'repl'@'localhost' IDENTIFIED BY 'xxxxxx';
Query OK, 0 rows affected (0.05 sec)

mysql> GRANT REPLICATION SLAVE ON *.* TO 'repl'@'localhost';
Query OK, 0 rows affected (0.05 sec)

mysql> SHOW MASTER STATUS;
+-----+-----+-----+-----+-----+
| File          | Position | Binlog_Do_DB | Binlog_Ignore_DB | Executed_Gtid_Set |
+-----+-----+-----+-----+-----+
| mysql-bin.000001 | 623 |      |      |      |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

1.5 导入数据库

导入表结构、存储过程、初始化数据、事件、触发器

```
mysql -u root -p < bd_schema_sp.sql
```

```
mysql -u root -p beidou < bd_adjust_schema.sql
mysql -u root -p beidou < bd_data.sql
mysql -u root -p beidou < bd_events.sql
mysql -u root -p beidou < bd_triggers.sql
```

1.6 建立从库

- 1) 停止主库的运行。
- 2) 将主库的数据目录整体复制成一个新的数据目录，如：“C:\ProgramData\MySQL\MySQL Server 5.7 – repl”。

- 3) 删除新数据目录下的“data\auto.cnf”文件。这个文件将由系统自动生成的，用以唯一标识数据库实例。

- 4) 编辑新目录下配置文件“my.ini”，删除或注释参数“log-bin”，然后添加或者修改以下参数：

```
[client]
port=3307

[mysqld]
datadir=C:/ProgramData/MySQL/MySQL Server 5.7 – repl\Data
port=3307
server-id=2

character-set-server=utf8
collation-server = utf8_unicode_ci
sql-mode="STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION"

gtid_mode=on
enforce_gtid_consistency=on
skip-slave-start=1

log-error="MARA-PC-repl.err"
max_sp_recursion_depth=100
event_scheduler=1
```

- 5) 以系统管理员账户执行以下命令，生成从库的系统服务“MySQL57-repl”：

```
mysqld --install MySQL57-repl --defaults-file="C:\ProgramData\MySQL\MySQL Server 5.7 – repl\my.ini"
```

- 6) 在系统服务列表中分别启动主库“MySQL57”和复制库“MySQL57-repl”。或者以管理员账户运行以下命令：

```
net start MySQL57
net start MySQL57-repl
```

- 7) 登入复制库：（由于是直接复制主库，因此root密码与主库相同）

```
mysql --port=3307 -u root -p
```

- 8) 输入以下命令，以建立与主库的数据复制关系：（xxxxx换成前面建立的复制账户的密码）

```
change master to master_host='localhost', master_user='repl', master_password='xxxxx', master_auto_position=1;
```

- 9) 输入命令“show slave status \G”，并检查以下这行的值是否为1：

```
Auto_Position: 1
```

- 10) 输入以下命令，以启动数据复制：

```
start slave;
```

- 11) 输入命令“show slave status \G”，并检查以下这些行的值（有的值可能随时间变化），以验证数据复制正在正确进行：

```
Slave_IO_State: Waiting for master to send event
Master_Host: localhost
Master_User: repl
Master_Port: 3306
Master_Log_File: mysql-bin.000004
Read_Master_Log_Pos: 486
Slave_IO_Running: Yes
```

Slave_SQL_Running: Yes

Seconds_Behind_Master: 0

Master_Server_Id: 1

Master_UUID: 6bd8673c-9bee-11e5-a939-1c872c722172

Slave_SQL_Running_State: Slave has read all relay log; waiting for more updates

- 12) 可以随时输入命令 “start slave;”或者 “stop slave;”，以启停数据同步。
- 13) 观察 “show slave status \G”的输出，当 “**Seconds_Behind_Master: 0**”意味着数据已全部同步到复制库中。
- 14) 如果主库中定义了事件，则需要把从库里的相应的事件删除或者让它们失效。输入命令 “show events\G”查看事件，并且对于每一个由主库复制而来的事件运行命令 “alter event XXXXX ON COMPLETION PRESERVE DISABLE;”。这一步保证从库不会重复运行主库的事件。
- 15) 如果需要从库每次启动后自动开始同步，则在它的配置文件中修改为 “skip-slave-start=0”。

2 在 Linux 上搭建 Mysql 主从数据库

2.1 资源地址

官网地址:

<http://dev.mysql.com/downloads/>

2.2 清理环境

运行以下命令来检查并卸载系统中已有的 mysql 版本:

```
yum remove mariadb, mariadb-devel, mariadb-libs, mariadb-server
yum-config-manager --disable mariadb
yum repolist all | grep mysql
rpm -qa | grep mysql
rpm -e <mysql 包名>
```

2.3 安装

运行以下命令以安装 mysql 资源库:

```
rpm -ihv mysql57-community-release-el7-7.noarch.rpm
```

然后运行以下命令以安装 mysql 服务器和客户端: (该步骤需要等待下载 mysql 各个包, 时间较长)

```
yum install mysql
```

最好确保 Mysql 版本在 5.7 以上。

接着运行以下命令以升级系统中已有的 mysql 数据:

```
mysql_upgrade -u root -p --force
```

最后运行以下命令以启动或者终止 mysql 服务器:

```
service mysqld start
service mysqld status
```

请参考以下官网资料安装配置 mysql:

<http://dev.mysql.com/doc/refman/5.7/en/linux-installation.html>

<http://dev.mysql.com/doc/refman/5.7/en/linux-installation-yum-repo.html>

若碰到如下错误, 则是因为系统中还残留旧版本 mysql, 安装之前应当彻底清除已存在版本:

```
mysql-community-release conflicts with mysql57-community-release-el7-7.noarch
```

2.4 设置账户安全

运行以下命令以查看 mysql 安装时自动生成的 root 密码：

```
cat /var/log/mysqld.log | grep 'temporary password'
```

（在 `/var/log/mysqld.log` 中找到 `temporary password` 这一行，双引号中的既是密码）

进入 mysql，`#mysql -uroot -p`，密码即上一步中的密码。

运行以下 mysql 命令来修改 root 密码：

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY 'new_password';
```

然后参考以下网页设置账号安全（如删除测试账户和测试数据库）：

<http://dev.mysql.com/doc/refman/5.7/en/postinstallation.html>

<http://dev.mysql.com/doc/refman/5.7/en/default-privileges.html>

2.5 设置字符集

编辑 “`/etc/my.cnf`”，添加或者修改以下行：

```
character-set-server=utf8
```

```
collation-server=utf8_unicode_ci
```

然后重启 mysql。

运行以下 mysql 命令以验证字符集已正确设置为 utf8：

```
mysql> show variables like 'character_set%';
```

```
+-----+-----+
| Variable_name | Value                |
+-----+-----+
| character_set_client | utf8                |
| character_set_connection | utf8                |
| character_set_database | utf8                |
| character_set_filesystem | binary              |
| character_set_results | utf8                |
| character_set_server | utf8                |
| character_set_system | utf8                |
| character_sets_dir | /usr/share/mysql/charsets/ |
+-----+-----+
8 rows in set (0.00 sec)
```

2.6 验证安装

运行以下命令：

```
mysql -u root -p
```

应该能够用修改后的 root 密码登录进去。

运行以下 mysql 命令：（用实际的密码）

```
SET PASSWORD = PASSWORD('XXXXXXX');
```

若不执行 `set password` 命令则——mysql 命令返回以下错误，（用户修改未同步）：

ERROR 1820 (HY000): You must reset your password using ALTER USER statement before executing this statement.

2.7 导入数据库

导入表结构、存储过程、初始化数据、事件、触发器

```
mysql -u root -p < bd_schema_sp.sql
mysql -u root -p beidou < bd_adjust_schema.sql
mysql -u root -p beidou < bd_data.sql
mysql -u root -p beidou < bd_events.sql
mysql -u root -p beidou < bd_triggers.sql
```

2.8 创建从库

Mysql 正常安装完毕以后会自动创建和配置好一个缺省的数据库实例，我们把这个缺省实例作为主库。在 CentOS 7 上，主库的数据目录是 “/var/lib/mysql”。

以 root 账号执行以下步骤：

- 1) 停止主库：

```
systemctl stop mysqld
```

- 2) 复制主库数据目录：

```
cd /var/lib
cp -R mysql mysql-repl
rm -f mysql-repl/auto.cnf
rm -rf mysql-repl/mysql-bin*
chmod -R 750 mysql-repl
chown -R mysql:mysql mysql-repl
```

- 3) 若 SELinux 正在启用（命令 “getenforce” 返回 “Enforcing”），则必须执行以下命令：

```
chcon -R --reference=/var/lib/mysql /var/lib/mysql-repl/
semanage fcontext -a -t mysqld_db_t "/var/lib/mysql-repl(/.*)?"
restorecon -R /var/lib/mysql-repl
semanage port -a -t mysqld_port_t -p tcp 3307
```

- 4) 检查数据目录的属性，应当如下：

```
[root@localhost lib]# cd /var/lib
[root@localhost lib]# ls -ladZ mysql*
drwxr-x--x. mysql mysql system_u:object_r:mysqld_db_t:s0 mysql
drwxr-x---. mysql mysql system_u:object_r:mysqld_db_t:s0 mysql-files
drwxr-x---. mysql mysql system_u:object_r:mysqld_db_t:s0 mysql-keyring
drwxr-x---. mysql mysql system_u:object_r:mysqld_db_t:s0 mysql-repl
```

2.9 修改配置文件

以 root 账号执行以下步骤：

- 1) 停止主库：

```
systemctl stop mysqld
```

- 2) 编辑配置文件 “/etc/my.cnf”，添加或者修改以下参数：

```
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
symbolic-links=0
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
```



```

port=3306
server-id=1

log-bin=mysql-bin
sync_binlog=1
gtid_mode=on
enforce_gtid_consistency=on
binlog_format=MIX
binlog-do-db=beidou
expire_logs_days=10
binlog_row_image=minimal

max_sp_recursion_depth=100

character-set-server=utf8
collation-server=utf8_general_ci
sql_mode=STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION

event_scheduler=1

[mysqld@repl]
datadir=/var/lib/mysql-repl
socket=/var/lib/mysql-repl/mysql.sock
symbolic-links=0
log-error=/var/log/mysqld-repl.log
pid-file=/var/run/mysqld/mysqld-repl.pid

port=3307
server-id=2

gtid_mode=on
enforce_gtid_consistency=on
skip-slave-start=1

max_sp_recursion_depth=100

character-set-server=utf8
collation-server=utf8_general_ci
sql_mode=STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION

event_scheduler=1

```

2.10 创建复制账号

以 root 账号执行以下步骤：

- 1) 启动主库：


```
systemctl start mysqld
```
- 2) 登入主库：


```
mysql -u root -p
```
- 3) 输入以下命令（xxxxxx 换成实际密码），以创建数据复制账户：


```
CREATE USER 'repl'@'localhost' IDENTIFIED BY 'xxxxxx';
GRANT REPLICATION SLAVE ON *.* TO 'repl'@'localhost';
```

SHOW MASTER STATUS;

2.11 配置从库

以 root 账号执行以下步骤:

1) 启动复制库:

`systemctl start mysqld@repl`

此命令正常应当无返回。

用命令 “`ps -ef mysql | grep mysql`”检查复制库的运行状态:

```
[root@localhost ~]# ps -ef | grep mysql
mysql      410      1  0 16:55 ?        00:00:01 /usr/sbin/mysqld --daemonize --pid-file=/var/run
/mysqld/mysqld.pid
mysql      991      1  0 17:02 ?        00:00:00 /usr/sbin/mysqld --defaults-group-suffix=@repl -
-daemonize --pid-file=/var/run/mysqld/mysqld-repl.pid
root       1040    2613  0 17:03 pts/1    00:00:00 mysql --port=3307 -u root -p
root       1539    1175  0 17:09 pts/4    00:00:00 grep --color=auto mysql
[root@localhost ~]#
```

2) 登入复制库: (由于是直接复制主库, 因此 root 密码与主库相同)

`mysql --host=127.0.0.1 --port=3307 -u root -p`

或

`mysql -S /var/lib/mysql-repl/mysql.sock -u root -p`

3) 输入以下命令, 以检查当前数据库实例的唯一标识:

`show variables like '%server_uuid%';`

`show variables like 'server_id';`

```
mysql> show variables like '%server_uuid%';
+-----+-----+
| Variable_name | Value                               |
+-----+-----+
| server_uuid   | def01a04-b96b-11e7-ac40-68f728cafa36 |
+-----+-----+
1 row in set (0.00 sec)

mysql> show variables like 'server_id';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| server_id     | 2     |
+-----+-----+
1 row in set (0.00 sec)

mysql>
```

数据库服务器实例的 UUID 也可以用命令 “`cat /var/lib/mysql-repl/auto.cnf`”来查看验证。

4) 输入以下命令, 以建立与主库的数据复制关系: (xxxxx 换成前面建立的复制账户的密码)

`change master to master_host='localhost', master_user='repl', master_password='xxxxx', master_auto_position=1;`

5) 输入命令 “`show slave status \G`”, 并检查以下这行的值是否为 1:

Auto_Position: 1

6) 输入以下命令, 以启动数据同步复制:

`start slave;`

7) 输入命令 “`show slave status \G`”, 并检查以下这些行的值 (有的值可能随时间变化), 以验证数据复制正在正确进行:

Slave_IO_State: Waiting for master to send event

Master_Host: localhost

Master_User: repl

Master_Port: 3306
Master_Log_File: mysql-bin.000004
Read_Master_Log_Pos: 486
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
Seconds_Behind_Master: 0
Master_Server_Id: 1
Master_UUID: 6bd8673c-9bee-11e5-a939-1c872c722172
Slave_SQL_Running_State: Slave has read all relay log; waiting for more updates

8) 可以随时输入命令 “start slave;”或者 “stop slave;”, 以启停数据同步。

9) 观察 “show slave status \G”的输出, 当 “Seconds_Behind_Master: 0”意味着数据已全部同步到复制库中。

10) 如果主库中定义了事件, 则需要把从库里的相应的事件删除或者让它们失效。输入命令 “show events\G”查看事件, 并且对于每一个由主库复制而来的事件运行命令 “alter event XXXXX ON COMPLETION PRESERVE DISABLE;”。这一步保证从库不会重复运行主库的事件。

11) 如果需要从库每次启动后自动开始同步, 则在配置文件中修改为 “skip-slave-start=0”。