**MYSQL配置同步文档**

V1.0.0

杜标 @ Enlink 数据中心

**目录**

[1. 概述 3](#_Toc30340)

[1.1. 说明 3](#_Toc17001)

[2. 安装MYSQL 3](#_Toc9053)

[2.1. 安装步骤 3](#_Toc26377)

[2.2. 配置ROOT用户 3](#_Toc2843)

[3. 配置主从同步 4](#_Toc10646)

[3.1. 用途 4](#_Toc26178)

[3.2. 基本配置 4](#_Toc32594)

[3.3. 其它 5](#_Toc16537)

[3.3.1. 其它同步命令 5](#_Toc10829)

[3.3.2. 其它配置项 6](#_Toc11210)

[4. 同步失败 6](#_Toc22011)

[4.1. 问题 6](#_Toc19484)

[4.2. 方法 7](#_Toc18821)

# 概述

本文档描述mysql在linux系统下的安装以及两个server间互为主从的配置。

## 说明

本文档使用的mysql版本是5.7.23，应确保主从机的版本一致。不同的linux发行版本可以使用mysql的二进制通用安装包进行安装，本文档不详细讲述安装步骤，本文使用的是yum源（centos）和apt源（ubuntu）进行的安装。本文针对centos7下的mysql说明互为主从的详细配置，其他linux发行版本的配置会有些许的差异。

# 安装MYSQL

本章节描述mysql的安装步骤及root用户的配置。

## 安装步骤

* 获取源

|  |
| --- |
| wget mysql57-community-release-el7-10.noarch.rpm  rpm -ivh mysql57-community-release-el7-10.noarch.rpm |

* 安装

|  |
| --- |
| yum -y install mysql-server |

* 启动

|  |
| --- |
| service mysqld start |

centos下默认配置文件路径：

配置文件：/etc/my.cnf

日志文件：/var/log/mysqld.log

服务启动脚本：/usr/lib/systemd/system/mysqld.service

进程号文件：/var/run/mysqld/mysqld.pid

## 配置ROOT用户

* 第一次启动后，会有一个临时的root密码，查看密码（ubuntu的日志路径不一样）：

|  |
| --- |
| grep "password" /var/log/mysqld.log |

* 使用root用户登录mysql，修改root密码，进入sql命令行：

|  |
| --- |
| mysql -u root -p |
| alter user 'root'@'localhost' identified by '123456'; |

* 重启mysql

|  |
| --- |
| service mysqld restart; |

* 给root用户所有的权限，包括远程登录的权限。以root登录mysql，进入命令行：

|  |
| --- |
| use mysql;  GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'%' IDENTIFIED BY '123456' |

注意：当使用的是ubuntu的时候，需要注意将 /etc/mysql/my.cnf配置文件下的bind-address = 127.0.0.1这一行需要修改为0.0.0.0，不然从机在请求时是连接不到的。

# 配置主从同步

本章节描述主从同步的用途、基本配置及其他同步命令、配置项。

## 用途

mysql的主从复制的主要优点是同步备份，在从机上的数据库就相当于一个(基本实时)备份库。在主从复制基础上，通过mysqlproxy可以做到读写分离, 由从机分担一些查询压力。 做一个双向的主从复制，两台机器互相为主机从机，这样在任何一个机器的库中写入，都会“实时”同步到另一台机器，双向的优点在于当一台主机发生故障时，另一台主机可以快速的切换过来继续服务。

## 基本配置

* 在两台机器上添加一个用于从机访问的帐号，赋予REPLICATION SLAVE权限：

|  |
| --- |
| GRANT REPLICATION SLAVE ON \*.\* TO 'slave'@'%' IDENTIFIED BY 'slave'; |

为slave用户赋予任何数据库中任何表上的REPLICATION SLAVE权限，此用户可以在网络任意位置访问，访问时以密码slave标记。

* 编辑mysql配置文件/etc/my.cnf（ubuntu是/etc/mysql/my.cnf）

找到[mysqld]这个标签，在它的下面有两行：

|  |
| --- |
| server-id               = 1  (ubuntu)log\_bin   = /var/log/mysql/mysql-bin.log  (centos)log\_bin |

解开这两行的注释，centos的log\_bin不需指定路径。注意这里的server-id是服务器编号，所以两台服务器上的值要设置的不一样，比如1和2。

* 重启mysql

|  |
| --- |
| service mysqld restart |

* 在mysql命令模式下执行:

|  |
| --- |
| FLUSH TABLES WITH READ LOCK;#全局读锁定，执行了命令之后所有库所有表都被锁定只读。这个时候数据库的写操作将被阻塞，读操作顺利进行。  SHOW MASTER STATUS; |

此时请保证执行这两条命令的mysql控制台不要退出，使锁继续生效。

* 分别重新打开一个mysql控台台, 分别配置master主机：

|  |
| --- |
| CHANGE MASTER TO  MASTER\_HOST = '192.168.101.102', #另一台机器的地址  MASTER\_PORT = 3306, #另一台机器的端口  MASTER\_USER = 'slave',#另一台机器上第一步分配的用户名  MASTER\_PASSWORD = 'slave', #另一台机器上第一步分配的密码  MASTER\_LOG\_FILE = 'mysql-bin.000001',#另一台机器上执行SHOW MASTER STATUS得到的文件名  MASTER\_LOG\_POS = 192; #另一台机器上执行SHOW MASTER STATUS得到的偏移量 |

* 开启同步

|  |
| --- |
| START SLAVE; |

* 验证正确性

|  |
| --- |
| SHOW SLAVE STATUS; |

* 释放读锁

|  |
| --- |
| unlock tables; |

现在同步开始正常运行了。

## 其它

本小节描述其它同步命令和其它同步配置项。

### 其它同步命令

|  |
| --- |
| Stop SLAVE; |

停止从机同步。

|  |
| --- |
| reset SLAVE; |

将使slave忘记主从复制关系的位置信息。该语句将被用于干净的启动，它删除master.info文件和relay-log.info文件以及所有的relay log文件并重新启用一个新的relaylog文件。

|  |
| --- |
| reset master; |

将删除日志索引文件中记录的所有binlog文件，创建一个新的日志文件，起始值从000001开始。

### 其它配置项

* master端

binlog-do-db     只同步的数据库， 二进制日志记录的数据库（多数据库用逗号，隔开）  
binlog-ignore-db 不同步的数据库，二进制日志中忽略数据库（多数据库用逗号，隔开）

* slave端

replicate-do-db    设定需要复制的数据库（多数据库使用逗号，隔开）

replicate-ignore-db 设定需要忽略的复制数据库 （多数据库使用逗号，隔开）

replicate-do-table  设定需要复制的表

replicate-ignore-table 设定需要忽略的复制表

replicate-wild-do-table 同replication-do-table功能一样，但是可以通配符

replicate-wild-ignore-table 同replication-ignore-table功能一样，但是可以加通配符

如replicate-wild-do-table=db\_name.%   只复制哪个库的哪个表  
如replicate-wild-ignore-table=mysql.%   忽略哪个库的哪个表

# 同步失败

本章节描述同步失败遇到的问题和解决方法。

## 问题

当表中存在唯一性约束的字段时，双向的互为主从的同步中会遇到如下的问题：'Duplicate entry '15' for key 'PRIMARY'' on query. Default database: ''. Query: 'insert into  persons……’，查看同步状态发现[Slave\_SQL\_Running: No](https://www.percona.com/forums/questions-discussions/percona-xtrabackup/11842-backup-stopped-working-slave-sql-running-no" \t "https://www.cnblogs.com/qdpurple/p/_blank)。

注：persons是新建的一张表。

## 方法

|  |
| --- |
| mysql> show binlog events [IN 'log\_name'] [FROM pos] [LIMIT [offset,] row\_count];  选项解析：  IN 'log\_name' 指定要查询的binlog文件名(不指定就是第一个binlog文件)  FROM pos指定从哪个pos起始点开始查起(不指定就是从整个文件首个pos点开始算)  LIMIT [offset,] 偏移量(不指定就是0)  row\_count 查询总条数(不指定就是所有行) |

找到slave SQL线程执行到哪个position，判断出slave库中执行binlog的position，重新设定slave库的binlog position，启动slave。