# **MySQL-5.7离线部署**

目录

[MySQL-5.7离线部署 2](#_Toc8487)

[一.彻底卸载mysql 1](#_Toc18804)

[二.下载rpm安装包 2](#_Toc28143)

[三.安装mysql 3](#_Toc19079)

[四.启动mysql服务 4](#_Toc12310)

[五.修改mysql中root本地登录密码 4](#_Toc16654)

[六.修改密码策略 5](#_Toc28200)

[七.允许远程登录 5](#_Toc6147)

[八.设置MySQL默认编码为utf8 6](#_Toc21355)

[九.忘记root用户密码解决方案 7](#_Toc20733)

[十.创建新用户 8](#_Toc30935)

[十一.主从复制配置 8](#_Toc20429)

## 一.彻底卸载mysql

**1.使用以下命令查看当前安装mysql情况，查找以前是否装有mysql**

rpm -qa|grep -i mysql

**2.如果查询到已安装mysql需停止mysql服务并删除之前安装的mysql**

#停止mysql服务

systemctl stop mysqld

#删除软件命令

rpm -e –nodeps $(rpm -qa|grep -i mysql)

如果提示依赖包错误，则使用以下命令尝试：

rpm -ev $(rpm -qa|grep -i mysql) --nodeps

如果提示错误：error: %preun(xxxxxx) scriptlet failed, exit status 1

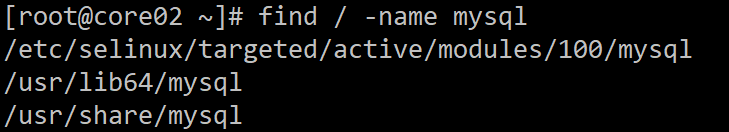
则用以下命令尝试：

rpm -e --noscripts $(rpm -qa|grep -i mysql)

**3.查找之前老版本mysql的目录、并且删除老版本mysql的文件和库**

find / -name mysql

查找结果如下



删除对应的mysql目录

rm -rf $(find / -name mysql)

**4.删除老版本mysql的配置文件**

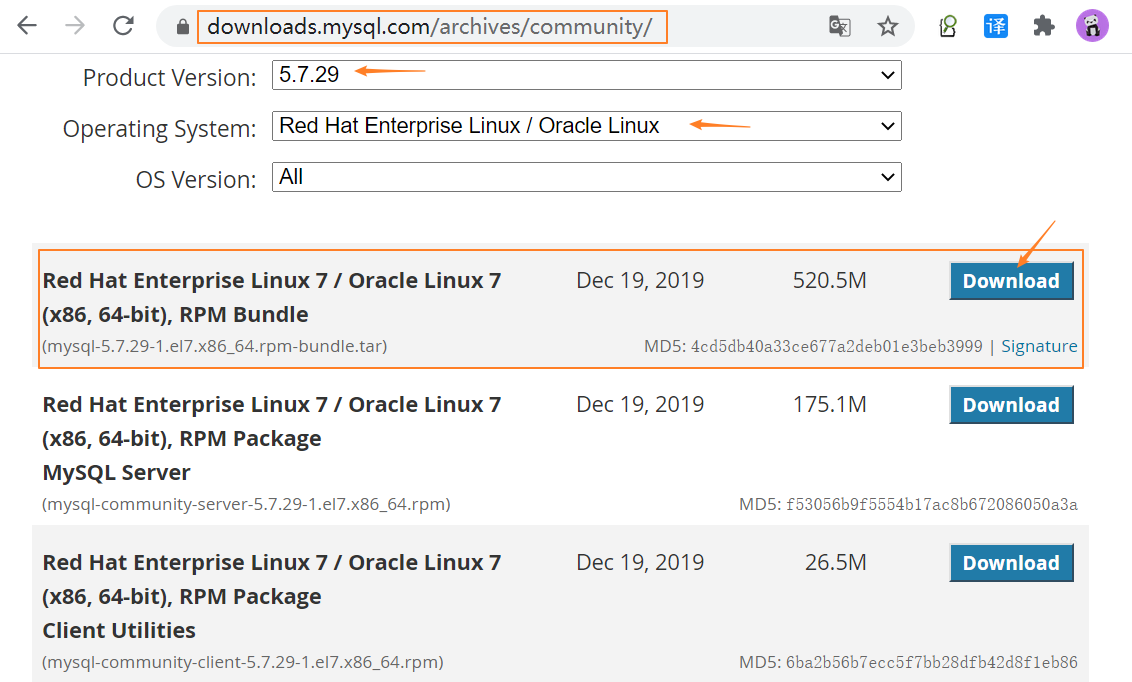
rm -rf /etc/my.cnf

rm -rf /etc/my.cnf.d

## 二.下载rpm安装包

在MySQL官网中找到YUM源rpm安装包：

下载页面：https://downloads.mysql.com/archives/community/



具体下载地址：<https://cdn.mysql.com/archives/mysql-5.7/mysql-5.7.29-1.el7.x86_64.rpm-bundle.tar>

或者使用百度网盘下载：

链接：https://pan.baidu.com/s/1Z8NX1IQNni4zykrWvnesig

提取码：275a

**上传并解压rpm的压缩包**

mkdir -p /baicdt/softwares/mysql\_rpm

cd /baicdt/softwares/

# 使用rz命令将下载的tar包上传到当前目录

# 解压至/baicdt/softwares/mysql\_rpm目录

tar -xvf /baicdt/softwares/mysql-5.7.29-1.el7.x86\_64.rpm-bundle.tar -C /baicdt/softwares/mysql\_rpm/

## 三.安装mysql

# 进入解压后的目录删除不需要安装的包

cd /baicdt/softwares/mysql\_rpm

扩展：各rpm包的作用，仅安装我们使用到的包即可

| **NO** | **包名** | **作用** | **是否安装** |
| --- | --- | --- | --- |
| 1 | client | MySQL客户端 | 是 |
| 2 | common | 服务端和客户端常用库文件 | 是 |
| 3 | devel | 客户端开发用的头文件及库 | 否 |
| 4 | embedded | 嵌入式版本 | 否 |
| 5 | embedded-compat | 兼容老版本嵌入式的共享库 | 否 |
| 6 | embedded-devel | 嵌入式版本的头文件与库 | 否 |
| 7 | libs | 数据库客户端共享库 | 是 |
| 8 | libs-compat | 兼容老版本的共享库 | 是 |
| 9 | server | 数据库服务端与相关工具 | 是 |
| 10 | test | MySQL测试套件 | 否 |

cd /baicdt/softwares/mysql\_rpm

rpm -ivh mysql-community-common-5.7.29-1.el7.x86\_64.rpm

rpm -ivh mysql-community-libs-5.7.29-1.el7.x86\_64.rpm

rpm -ivh mysql-community-libs-compat-5.7.29-1.el7.x86\_64.rpm

rpm -ivh mysql-community-client-5.7.29-1.el7.x86\_64.rpm

rpm -ivh mysql-community-server-5.7.29-1.el7.x86\_64.rpm

# 删除安装包后，查看是否安装成功

rm -rf /baicdt/softwares/mysql\_rpm/\* && rpm -qa|grep -i mysql

## 四.启动mysql服务

临时启动mysql服务

systemctl start mysqld

开机启动mysql服务

systemctl enable mysqld

查看MySQL的启动状态

systemctl status mysqld

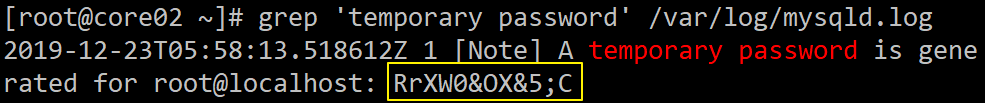
ps -le | grep mysqld

netstat -ntlp | grep mysqld

## 五.修改mysql中root本地登录密码

mysql安装完成之后，在/var/log/mysqld.log文件中给root生成了一个默认密码。通过以下命令找到密码：

grep 'temporary password' /var/log/mysqld.log



登录MySQL，使用刚才找到的密码

mysql -u root -p

修改root用户密码（二选一）

ALTER USER 'root'@'localhost' IDENTIFIED BY 'Wsygzgr\_2019';

set password for 'root'@'localhost'=password('Wsygzgr\_2019');

**注意：**mysql5.7默认安装了密码安全检查插件（validate\_password），默认密码检查策略要求密码必须包含：大小写字母、数字和特殊符号，并且长度不能少于8位。否则会提示ERROR 1819 (HY000): Your password does not satisfy the current policy requirements错误。通过msyql环境变量可以查看密码策略的相关信息

show variables like '%password%';

## 六.修改密码策略

生产环境为了安全考虑，不要改密码策略！！！

虚拟机环境为了方便记mysql密码，可选择下面的方案①：

vim /etc/my.cnf

①如果不需要密码策略，添加my.cnf文件中添加如下配置禁用即可：

validate\_password=off

②选择0(LOW)，1(MEDIUM)，2(STRONG)其中一种，选择2需提供密码字典文件

validate\_password\_policy=0

**重新启动mysql服务使配置生效：**

systemctl restart mysqld

重新设置mysql密码为123456

mysql -u root -pWsygzgr\_2019

修改root用户密码（二选一）

ALTER USER 'root'@'localhost' IDENTIFIED BY '123456';

set password for 'root'@'localhost'=password('123456');

## 七.允许远程登录

mysql默认只允许root帐户在本地登录，想要远程连接mysql，必须开启root用户允许远程连接，或者添加一个允许远程连接的帐户。

①开启root用户远程连接（任意IP都可以访问）

GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'%' IDENTIFIED BY '123456' WITH GRANT OPTION;

②指定特定的IP，开启root用户远程连接

GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'指定的IP' IDENTIFIED BY 'root用户的密码' WITH GRANT OPTION;

③一般为了安全起见，会创建专用的远程访问用户

GRANT ALL PRIVILEGES ON \*.\* TO 'dadeity'@'%' IDENTIFIED BY 'daDeity@163.com' WITH GRANT OPTION;

刷新MySQL的系统权限相关表，否则会出现拒绝访问

flush privileges;

## 八.设置MySQL默认编码为utf8

修改 /etc/my.cnf配置文件，如下所示：

vim /etc/my.cnf

添加

character\_set\_server=utf8

init\_connect='SET NAMES utf8'

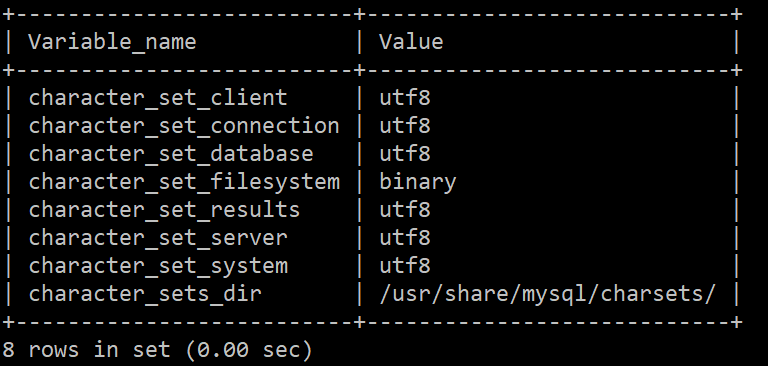
重启MySQL服务

systemctl restart mysqld

查看MySQL默认编码

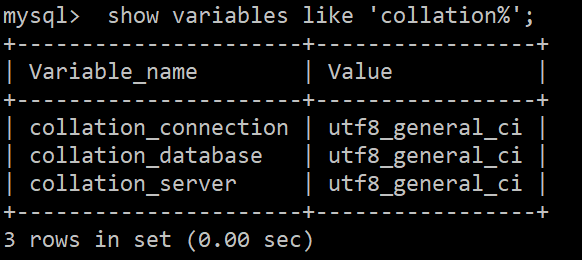
mysql -uroot -p123456

show variables like 'char%';



查看排序规则

show variables like 'collation%';



默认配置文件路径

* 配置文件： /etc/my.cnf
* 日志文件： /var/log//var/log/mysqld.log
* 服务启动脚本： /usr/lib/systemd/system/mysqld.service
* socket文件： /var/run/mysqld/mysqld.pid

## 九.忘记root用户密码解决方案

在/etc/my.cnf文件[mysqld]下面加如下命令

vim /etc/my.cnf

[mysqld]

skip-grant-tables

登陆mysql并改改密码

mysql -u root

update mysql.user set authentication\_string=password('123456') where user='root' and Host = 'localhost';

flush privileges

设置好密码后再删掉/etc/my.cnf中的skip-grant-tables

## 十.创建新用户

CREATE USER '新用户名'@'%'IDENTIFIED BY '设置新用户的密码';

#把数据库的访问权限给新用户

GRANT ALL PRIVILEGES ON 数据库名.\* TO '用户名'@'%' IDENTIFIED BY '密码';

#给新用户开启所有远程访问权限

GRANT ALL PRIVILEGES ON \*.\* TO '用户名'@'%' IDENTIFIED BY '密码' WITH GRANT OPTION;

#刷新mysql的权限列表

FLUSH PRIVILEGES;

## 十一.主从复制配置

**安装环境**

主(Master) centos7：192.168.52.100

从(Slave) centos7：192.168.52.110

### 1.Master的配置

修改/etc/my.cnf

vim /etc/my.cnf

#主节点开启二进制日志功能

log-bin=mysql-bin

#标识唯一id（必须），一般使用ip最后位

server-id=100

#不需要同步的数据库，可设置多个

binlog-ignore-db=information\_schema

binlog-ignore-db=mysql

#指定需要同步的数据库，可以设置多个

binlog-do-db=test

binlog-do-db=demo

修改完配置，重启mysql

systemctl restart mysqld

进入Mysql

mysql -u root -p123456

赋予从库权限帐号，允许用户在主库上读取日志，赋予192.168.52.110也就是Slave机器有File权限，只赋予Slave机器有File权限还不行，还要给它REPLICATION SLAVE的权限才可以。

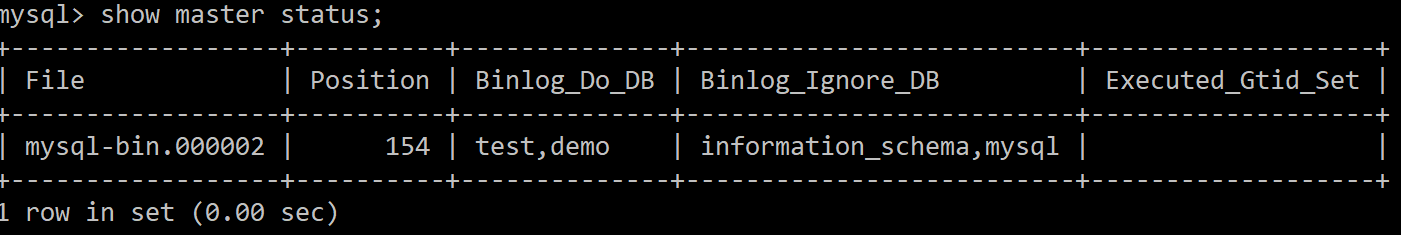
grant FILE on \*.\* to 'root'@'192.168.52.110' identified by '123456';

grant replication slave on \*.\* to 'root'@'192.168.52.110' identified by '123456';

flush privileges;

重启Mysql后查看主库信息

show master status;



这里的 File 、Position 是在配置Salve的时候要使用到的，Binlog\_Do\_DB表示要同步的数据库，Binlog\_Ignore\_DB 表示Ignore的数据库，这些都是在配置的时候进行指定的。

另外：如果执行这个步骤始终为Empty set(0.00 sec)，那说明前面的my.cnf没配置对。

### 2.Slave的配置

**修改/etc/my.cnf**

vim /etc/my.cnf

#主节点开启二进制日志功能

log-bin=mysql-bin

#标识唯一id（必须），一般使用ip最后位

server-id=110

#指定不需要复制的数据库(与主节点binlog-ignore-db要一一对应)

replicate-ignore-db=information\_schema

replicate-ignore-db=mysql

#指定需要同步的数据库(与主节点binlog-ignore-db要一一对应)

replicate-do-db=test

replicate-do-db=demo

#此选项告诉从属服务器将其SQL线程执行的更新记录到其自己的二进制日志中。

log-slave-updates

#跳过所有错误

slave-skip-errors=all

slave-net-timeout=60

**进入Mysql**

mysql -u root -p123456

stop slave;

change master to master\_host='192.168.52.100',master\_user='root',master\_password='123456',master\_log\_file='mysql-bin.000002', master\_log\_pos=154;

start slave; #开启Slave

在这里指定Master的信息，master\_log\_file是在配置Master的时候的File选项， master\_log\_pos是在配置Master的Position 选项，这里要进行对应。

**查看配置的信息：**

show slave status \G;

当下面两个参数为Yes，则表示配置成功了

Slave\_IO\_Running: Yes

Slave\_SQL\_Running: Yes

查看的所有结果如下

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Slave\_IO\_State: Waiting for master to send event

Master\_Host: 192.168.52.100

Master\_User: root

Master\_Port: 3306

Connect\_Retry: 60

Master\_Log\_File: mysql-bin.000002

Read\_Master\_Log\_Pos: 154

Relay\_Log\_File: core02-relay-bin.000002

Relay\_Log\_Pos: 320

Relay\_Master\_Log\_File: mysql-bin.000002

Slave\_IO\_Running: Yes

Slave\_SQL\_Running: Yes

Replicate\_Do\_DB: test,demo

Replicate\_Ignore\_DB: information\_schema,mysql

Replicate\_Do\_Table:

Replicate\_Ignore\_Table:

Replicate\_Wild\_Do\_Table:

Replicate\_Wild\_Ignore\_Table:

Last\_Errno: 0

Last\_Error:

Skip\_Counter: 0

Exec\_Master\_Log\_Pos: 154

Relay\_Log\_Space: 528

Until\_Condition: None

Until\_Log\_File:

Until\_Log\_Pos: 0

Master\_SSL\_Allowed: No

Master\_SSL\_CA\_File:

Master\_SSL\_CA\_Path:

Master\_SSL\_Cert:

Master\_SSL\_Cipher:

Master\_SSL\_Key:

Seconds\_Behind\_Master: 0

Master\_SSL\_Verify\_Server\_Cert: No

Last\_IO\_Errno: 0

Last\_IO\_Error:

Last\_SQL\_Errno: 0

Last\_SQL\_Error:

Replicate\_Ignore\_Server\_Ids:

Master\_Server\_Id: 100

Master\_UUID: 32913ced-252e-11ea-9486-0050563beaf3

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; waiting for more updates

Master\_Retry\_Count: 86400

Master\_Bind:

Last\_IO\_Error\_Timestamp:

Last\_SQL\_Error\_Timestamp:

Master\_SSL\_Crl:

Master\_SSL\_Crlpath:

Retrieved\_Gtid\_Set:

Executed\_Gtid\_Set:

Auto\_Position: 0

Replicate\_Rewrite\_DB:

Channel\_Name:

Master\_TLS\_Version:

1 row in set (0.00 sec)

ERROR:

No query specified

对主库做增删改操作，验证从库数据同步情况。