### Cmake编译安装MySQL&多配置文件部署MySQL多实例方案

上一节已经简单介绍过MySQL多实例、应用场景、优缺点以及两种实施方案，本文主要介绍第一种即多配置文件部署多实例MySQL,单一配置文件部署方案会在下一节进行实战。

说明：

1. 本文参考老男孩MySQL教程,为本人实战操作记录
2. 以创建2个为例

MySQL二进制包和源码包的区分

* 二进制格式的包名字很长，都带有版本号、适应平台、适应的硬件类型等
  + mysql-5.0.45.tar.gz 是源码包 （编译安装）
* 源码格式仅仅就是一个版本号的tar包
  + mysql-5.0.45-linux-x86\_64-glibc23.tar.gz 是二进制包

所有的操作根据实际情况而定

* **Camek下载：**<https://cmake.org/download/>
* **MySQL下载：**<https://dev.mysql.com/downloads/mysql/>

## ****准备安装环境****

### ****1.首先检查是否已经安装过mysql：****

> rpm -qa | grep mysql

### 2.有的话就卸载掉以前安装的mysql：

> rpm -e --nodeps xxx（xxx是搜索结果）

### 3.并删除所有的相关文件：

> rm -f /etc/my.cnf

## 下载cmake安装包编译安装cmake

### 1.下载解压Cmake

> wget [https://cmake.org/files/v3.9/cmake-3.9.1.tar.gz](https://cmake.org/files/v3.9/cmake-3.9.1-Linux-x86_64.tar.gz)

> tar zxf cmake-3.9.1.tar.gz

### 2.编译安装cmake

> cd cmake-3.9.1

> ./configure

> gmake

> echo $? #编译但是未安装之前判断是否编译有错

> gmake install

## 下载MySQL编译安装MySQL

### 1.安装依赖包

> yum -y install libaio libaio-devel

> yum -y install ncurses-devel

### 2.创建用户组 用户

> groupadd mysql

> useradd mysql -s /sbin/nologin -M -g mysql

### 3.下载解压

> wget https://dev.mysql.com/get/Downloads/MySQL-5.5/mysql-5.5.57.tar.gz

> tar -zxf mysql-5.5.57.tar.gz

### 4.cmake编译安装(编译时不能指定配置文件端口号mysql.sock地址等)

> cd mysql-5.5.57

> cmake \-DCMAKE\_INSTALL\_PREFIX=/usr/local/mysql5.5.57 -DMYSQL\_UNIX\_ADDR=/usr/local/mysql/tmp/mysql.sock -DMYSQL\_DATADIR=/usr/local/mysql/data -DSYSCONFDIR=/etc/my.cnf -DWITH\_MYISAM\_STORAGE\_ENGINE=1 -DWITH\_INNOBASE\_STORAGE\_ENGINE=1 -DWITH\_MEMORY\_STORAGE\_ENGINE=1 -DWITH\_READLINE=1 -DMYSQL\_TCP\_PORT=3306 -DENABLED\_LOCAL\_INFILE=1 -DWITH\_PARTITION\_STORAGE\_ENGINE=1 -DEXTRA\_CHARSETS=all

> make && make install

### 5.检查是否安装成功

> echo $?

> 0

#（输出0表示成功）

### 6.设置软链接及配置文件

> ln -s /usr/local/mysql55 /usr/local/mysql

## 部署多实例的MySQL数据库

### 1.创建多实例目录

> mkdir -pv /data/{3306,3307}/data

> tree data

### 2.为不同的实例创建创建配置文件（文章结尾附配置文件）

> vim /data/3306/my.cnf

> vim /data/3307/my.cnf

### 3.创建MySQL多实例启动文件（文章结尾附启动文件）

> vim /data/3306/mysql

> vim /data/3307/mysql

### 4.授权mysql用户目录权限

> chown -R mysql.mysql /data

1. 将my.cnf文件权限设置成644

> chmod 644 /data/3306/my.cnf

> chmod 644 /data/3307/my.cnf

1. 由于mysql脚本中保存了登录数据库的密码，需要将文件权限设置成700，只允许root,mysql访问

> chmod 700 /data/3306/mysql

> chmod 700 /data/3307/mysql

### 5.配置MySQL命令全局使用路径

如果不为MySQL的命令配置全局路径，就无法直接在命令行输入mysql,这样只能用全路径/application/msyq/bin/mysql 这样带着路径输出会比较麻烦

1. 确认mysql命令所在路径

> ll /data/mysql/bin/mysql

2.修改PATH路径

#修改文件

> vim /etc/profile

#添加如下一行

> export PATH=/data/mysql/bin:$PATH

#让其生效

> source /etc/profile

3.检查PATH是否生效

> echo $PATH

4.确认上个命令的输出是否有/mysql/bin:(Mysql安装的bin目录)

### 6.初始化数据库

> cd /usr/local/mysql/scripts/

1.注意5.5和MySQL 5.1的路径不同，MySQL 5.1 不在mysql/bin下

> ./mysql\_install\_db --basedir=/usr/local/mysql --datadir=/data/3306/data/ --user=mysql

> ./mysql\_install\_db --basedir=/usr/local/mysql --datadir=/data/3307/data/ --user=mysql

2.每个初始化出现两个ok则说明成功

### 7.启动MySQL

> /data/3306/mysql start

> /data/3307/mysql start

### 8.检查是否启动成功

> ps -ef | grep mysqld

或者

> netstat -anp | grep 3306

### 9.为用户增加密码，修改密码

> mysqladmin -S /data/3306/mysql.sock -u root password '123.asd'

### 10.多实例数据库的登陆

本地登陆：

> mysql -u root -p -S /data/3306/mysql.sock

远程登录：mysql –h主机名 -u用户名 –p密码 –P3306

> system mysql -u root -p -S /data/3307/mysql.sock

## 添加实例（比如3308,ruguo ）

> mkdir -p /data/3308/data

#下面的两步操作注意文件中的端口号

> vim /data/3308/my.cnf

> vim /data/3308/mysql

> chown -R mysql.mysql /data

> chmod 644 /data/3308/my.cnf

> chmod 700 /data/3308/mysql

> cd /usr/local/mysql/scripts/

> ./mysql\_install\_db --basedir=/usr/local/mysql --datadir=/data/3308/data/ --user=mysql

> /data/3308/mysql start

## 附：

**1.MySQL多实例的配置文件(标注的地方需要注意,每个实例的端口要修改)**

[client]

port= 3306

socket = /data/3306/mysql.sock

[mysql]

no-auto-rehash

[mysqld]

user= mysql

port= 3306

socket= /data/3306/mysql.sock

basedir = /usr/local/mysql

datadir = /data/3306/data

open\_files\_limit = 1024

back\_log = 600

max\_connections = 800

max\_connect\_errors = 3000

table\_cache = 614

external-locking = FALSE

max\_allowed\_packet =8M

sort\_buffer\_size = 1M

join\_buffer\_size = 1M

thread\_cache\_size = 100

thread\_concurrency = 2

query\_cache\_size = 2M

query\_cache\_limit = 1M

query\_cache\_min\_res\_unit = 2k

#default\_table\_type = InnoDB

thread\_stack = 192K

#transaction\_isolation = READ-COMMITTED

tmp\_table\_size = 2M

max\_heap\_table\_size = 2M

long\_query\_time = 1

#log\_long\_format

#log-error = /data/3306/error.log

#log-slow-queries = /data/3306/slow.log

pid-file = /data/3306/mysql.pid

log-bin = /data/3306/mysql-bin

relay-log = /data/3306/relay-bin

relay-log-info-file = /data/3306/relay-log.info

binlog\_cache\_size = 1M

max\_binlog\_cache\_size = 1M

max\_binlog\_size = 2M

expire\_logs\_days = 7

key\_buffer\_size = 16M

read\_buffer\_size = 1M

read\_rnd\_buffer\_size = 1M

bulk\_insert\_buffer\_size = 1M

#myisam\_sort\_buffer\_size = 1M

#myisam\_max\_sort\_file\_size = 10G

#myisam\_max\_extra\_sort\_file\_size = 10G

#myisam\_repair\_threads = 1

#myisam\_recover

lower\_case\_table\_names = 1

skip-name-resolve

slave-skip-errors = 1032,1062

replicate-ignore-db=mysql

server-id = 1

innodb\_additional\_mem\_pool\_size = 4M

innodb\_buffer\_pool\_size = 32M

innodb\_data\_file\_path = ibdata1:128M:autoextend

innodb\_file\_io\_threads = 4

innodb\_thread\_concurrency = 8

innodb\_flush\_log\_at\_trx\_commit = 2

innodb\_log\_buffer\_size = 2M

innodb\_log\_file\_size = 4M

innodb\_log\_files\_in\_group = 3

innodb\_max\_dirty\_pages\_pct = 90

innodb\_lock\_wait\_timeout = 120

innodb\_file\_per\_table = 0

[mysqldump]

quick

max\_allowed\_packet = 2M

[mysqld\_safe]

log-error=/data/3306/mysql\_3306.err

pid-file=/data/3306/mysqld.pid

1. **MySQL多实例的启动文件(标注的地方需要注意,每个实例的端口要修改)**

#init

port=3306

mysql\_user="root"

mysql\_pwd="yourpwd"

CmdPath="/usr/local/mysql/bin"

mysql\_sock="/data/${port}/mysql.sock"

#startup function

function\_start\_mysql()

{

if [ ! -e "$mysql\_sock" ];then

printf "Starting MySQL...\n"

/bin/sh ${CmdPath}/mysqld\_safe --defaults-file=/data/${port}/my.cnf 2>&1 > /dev/null &

else

printf "MySQL is running...\n"

exit

fi

}

#stop function

function\_stop\_mysql()

{

if [ ! -e "$mysql\_sock" ];then

printf "MySQL is stopped...\n"

exit

else

printf "Stoping MySQL...\n"

${CmdPath}/mysqladmin -u ${mysql\_user} -p${mysql\_pwd} -S /data/${port}/mysql.sock shutdown

fi

}

#restart function

function\_restart\_mysql()

{

printf "Restarting MySQL...\n"

function\_stop\_mysql

sleep 2

function\_start\_mysql

}

case $1 in

start)

function\_start\_mysql

;;

stop)

function\_stop\_mysql

;;

restart)

function\_restart\_mysql

;;

\*)

printf "Usage: /data/${port}/mysql {start|stop|restart}\n"

esac