

# MySQL 简介

2014/8/9

renyl

## 1 下载安装文件

安装 MySQL 需要两个文件：

MySQL-server-4.0.23-0.i386.rpm

MySQL-client-4.0.23-0.i386.rpm

注：下载地址为：<http://www.mysql.com/downloads/mysql-4.0.html>。

## 2 安装 MySQL

### 2.1 安装服务器端

```
# rpm -ivh MySQL-server-4.0.23-0.i386.rpm
```

显示如下信息：

```
warning: MySQL-client-4.0.23-0.i386.rpm
signature: NOKEY, key ID 5072elf5
Preparing... ##### [100%]
1:MySQL-server ##### [100%]
..... (省略显示)
/usr/bin/mysqladmin -u root password 'new-password'
/usr/bin/mysqladmin -u root -h test1 password 'new-password'
..... (省略显示)
Starting mysqld daemon with databases from /var/lib/mysql
```

如出现如上信息，表示服务端安装完毕。使用 netstat 命令查看 MySQL 服务是否已经启动：

```
# netstat -nat
```

| Active Internet connections (servers and established) |        |        |               |                 |        |  |
|---|--------|--------|---------------|-----------------|--------|--|
| Proto   | Recv-Q | Send-Q | Local Address | Foreign Address | State  |  |
| tcp   | 0      | 0      | 0.0.0.0:3306  | 0.0.0.0:*       | LISTEN |  |

注：MySQL 默认的端口是 3306（可以在/etc/services 文件下查看）。

## 2.2 安装客户端

运行如下命令：

```
# rpm -ivh MySQL-client-4.0.23-0.i386.rpm
```

```
warning: MySQL-client-4.0.23-0.i386.rpm: V3 DSA signature: NOKEY, key ID 5072e1f5
Preparing...      ##### [100%]
1:MySQL-client    ##### [100%]
```

显示安装完毕。

注：如果是用 iso 文件内的 rpm 包进行安装，使用如下三个命令即可。

```
yum install mysql
```

```
yum install mysql-server
```

```
yum install mysql-devel
```

## 3 登录 MySQL

### 3.1 启动 mysql 服务

`/etc/init.d/mysqld start` 或 `service mysqld start`

### 3.2 登录 MySQL

mysql 命令的使用语法如下：

`mysql [-u username] [-h host] [-p[password]] [dbname]`

注：username 与 password 分别是 MySQL 的用户名与密码，mysql 的初始管理帐号是 root，没有密码，这个 root 用户不是 Linux 的系统用户。MySQL 默认用户是 root，由于 初始没有密码，第一次进时只需键入 mysql 即可。

### 3.3 修改登录密码

使用如下命令进行修改：

`/usr/bin/mysqladmin -u root -p old_password password new_password`

EX:给 root 用户加个密码 rootroot

`# /usr/bin/mysqladmin -u root password rootroot`

注：因为开始时 root 用户没有密码，所以 -p 旧密码一项就可以省略了。

如果上述方法不行，用下面来试试：

- 1: 先关闭 mysql 服务器 `/etc/init.d/mysqld stop`
- 2: 然后以不检查权限的方式启 `/usr/bin/mysqld_safe --skip-grant-tables &`
- 3: 然后直接用 root 登录 `mysql -u root`
- 4: 然后修改 root 用户的密码 `mysql> update mysql.user set Password=PASSWORD('新密码') where User='root';`
- 5: `mysql> flush privileges;`
- 6: `mysql> exit`
- 7: 然后关闭 mysql 服务器 `pidof mysqld | xargs kill`
- 8: 然后再以平常的启动方式启动 `/etc/init.d/mysqld start`

经过试验，直接使用步骤 4, 5, 6 也可以的。

### 3.4 增加密码后的登录格式

```
mysql -u username -p  
Enter password: (输入密码)
```

注: 其中-u 后跟的是用户名, -p 要求输入密码, 回车后在输入密码处输入密码。这个 mysql 文件在/usr/bin 目录下, 与后面讲的启动文件/etc/init.d/mysql 不是一个文件。

## 4 MySQL 目录

MySQL 安装完成后不像 SQL Server 默认安装在一个目录, 它的数据库文件、配置文件和命令文件分别在不同的目录。下面就介绍一下这几个目录。

### 4.1 数据库目录

/var/lib/mysql/

### 4.2 配置文件目录

/usr/share/mysql (mysql.server 命令及配置文件)

### 4.3 相关命令目录

/usr/bin(mysqladmin mysqldump 等命令)

### 4.4 启动脚本目录

/etc/rc.d/init.d/ (启动脚本文件 mysql 的目录)

## 5 更改数据库目录

MySQL 默认的数据库目录为 `/var/lib/mysql`，假如要把目录移到 `/home/data` 下。需要进行下面几步：

1) home 目录下建立 data 目录

```
cd /home; mkdir data
```

2) 把 MySQL 服务进程停掉：

```
/usr/bin/mysqladmin -u root -p shutdown 或者 service mysqld stop
```

3) 把 `/var/lib/mysql` 整个目录移到 `/home/data`

```
mv /var/lib/mysql /home/data/
```

或者

```
cp -a /var/lib/mysql /home/data/
```

注：

1) 使用 `cp -a` 这样就不会因为目录权限问题导致启动 `mysql` 失败。

2) 还有个可能的原因是 `selinux`，解决方案有两个。

A: 关闭 `selinux` 服务。

B: `chcon -reference=target source` (`ll -Z /dir` 可以查看)

4) 修改 `my.cnf` 配置

```
#vim my.cnf
```

```
# The MySQL server
[mysqld]
#socket = /var/lib/mysql/mysql.sock (注释此行)
socket = /home/data/mysql/mysql.sock (加上此行)

#datadir=/var/lib/mysql (注释此行)
datadir=/home/data/mysql (加上此行)
```

注：如果 `/etc/` 目录下没有 `my.cnf` 配置文件，请到 `/usr/share/mysql/` 下找到 `*.cnf` 文件，拷贝其中一个到 `/etc/` 并改名为 `my.cnf`。

5) 重新启动 MySQL 服务

```
/etc/rc.d/init.d/mysql star
```

注：如果工作正常移动就成功了，否则进行如下操作：

a) 关闭 `selinux` 服务后重启 MySQL 服务。

b) 如果出现 “`ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/var/lib/mysql/mysql.sock' (2)`” 这种错误，使用如下命令登陆：

```
# mysql -u root -p -S /home/data/mysql/mysql.sock
```

## 6 MySQL 的常用操作

MySQL 中每个命令后都要以分号：结尾。

### 6.1 显示数据库

```
mysql> show databases;
```

MySQL 刚安装完有两个数据库：mysql 和 test。mysql 库非常重要，它里面有 MySQL 的系统信息，我们改密码和新增用户，实际上就是用这个库中的相关表进行操作。

### 6.2 显示数据库中的表

```
mysql> use mysql; （打开库，对每个库进行操作前要打开此库）  
Database changed  
mysql> show tables;
```

### 6.3 显示数据表的结构

```
mysql> describe 表名;
```

### 6.4 显示表中的记录

```
mysql> select * from 表名;
```

### 6.5 建库

```
mysql> create database dfg;
```

### 6.6 建表

```
mysql> use dfg;  
mysql> create table name (id int(3) auto_increment not null primary key, xm  
char(8), xb char(2), csny date);
```

## 6.7 增加记录

```
mysql> insert into table_name values('','张三','男','1971-10-01');  
mysql> insert into table_name values('','白云','女','1972-05-20');
```

## 6.8 修改纪录

将张三的出生年月改为 1971-01-10:

```
mysql> update name set csny='1971-01-10' where xm='张三';
```

## 6.9 删除纪录

删除张三的纪录:

```
mysql> delete from name where xm='张三';
```

## 6.10 删库和删表

```
mysql> drop database 库名;  
mysql> drop table 表名
```

## 6.11 更改表属性

ALTER TABLE 的用法 增加列、删除列、改列名、改列约束、改表名:

```
ALTER TABLE table [ * ] ADD [ COLUMN ] column type  
ALTER TABLE table [ * ] ALTER [ COLUMN ] column { SET DEFAULT value | DROP DEFAULT }  
ALTER TABLE table [ * ] RENAME [ COLUMN ] column TO newcolumn  
ALTER TABLE table RENAME TO newtable  
ALTER TABLE table ADD table constraint definition
```

## 7 增加 MySQL 用户

格式: `grant * on 数据库.* to 用户名@登录主机 identified by "密码"`

### 7.1 在任何主机上可以登陆

增加一个用户 `user_1` 密码为 123, 让他可以在任何主机上登录, 并对所有数据库有查询、插入、修改、删除的权限:

```
mysql> grant select,insert,update,delete on *.* to user_1@"%" Identified by "123";
```

### 7.2 只在 localhost 上登陆

增加一个用户 `user_2` 密码为 123, 让此用户只可以在 `localhost` 上登录, 并可以对数据库 `dfg` 进行查询、插入、修改、删除的操作 (`localhost` 指本地主机, 即 MySQL 数据库所在的那台主机):

```
mysql> grant select,insert,update,delete on dfg.* to user_2@localhost identified by "123";
```

或者:

```
mysql> use mysql;
```

```
mysql> update user set Host = 'localhost' where User = 'root';
```

### 7.3 从指定 ip 登陆

如果你想允许用户 `myuser` 从 `ip` 为 192.168.1.3 的主机连接到 `mysql` 服务器, 并使用 `mypassword` 作为密码, 命令如下:

```
mysql> GRANT ALL PRIVILEGES ON *.* TO myuser@'192.168.1.3' IDENTIFIED BY 'mypassword' WITH GRANT OPTION;
```

```
mysql> flush privileges //使修改生效
```

注: 远程访问数据库的方式:

```
mysql -u username -p -h 192.168.113.50 (-h 后跟的是要登录主机的 ip 地址)
```

### 7.4 新增用户无法登陆

使用 `grant` 命令创建新用户, 如下:

```
grant select,insert,update,delete on tpcw.* to tpcw@"%" identified by "tpcw";
```

注: 其中 "%" 表示可以在任何地址登录。



### 7.4.1 问题

创建后到“mysql.user”表下查看,有该用户,但是使用命令#mysql -u appadmin -p password 登录,提示无法登录,并提示错误如下: ERROR 1045 (28000): Access denied for user 'appadmin'@'localhost' (using password: YES) 。

### 7.4.2 原因

因为“mysql.user”表中有另外一些记录产生了作用,最有可能的就是已经有一条 localhost 记录,就是用户名是空,主机字段是 localhost 的记录,查看该表果然有,如下所示:

```
mysql> select host,user,password from mysql.user;
```

| host                  | user      | password         |
|-----------------------|-----------|------------------|
| localhost             | root      | *6C362347EBEAA7  |
| localhost.localdomain | root      |                  |
| 127.0.0.1             | root      |                  |
| localhost             |           |                  |
| localhost.localdomain |           |                  |
| %                     | tpcw      | *B6B90B872522C1B |
| localhost             | mynewuser | *890406EBC2A6D76 |

MySQL 手册中增加用户部分有这么一段话:

- 1) 一个账户 ('monty'@'localhost')只用于从本机连接时。
- 2) 另一个账户('monty'@'%')可用于从其它主机连接。
- 3) 请注意 monty 的两个账户必须能从任何主机以 monty 连接。
- 4) 没有 localhost 账户,当 monty 从本机连接时,mysql\_install\_db 创建的 localhost 的匿名用户账户将占先。
- 5) 结果是 monty 将被视为匿名用户。
- 6) 原因是匿名用户账户的 host 列值比'monty'@'%' 账户更具体,这样在 user 表排序顺序中排在前面。
- 7) 因为在登录 mysql -u root -p 里面默认的-h 是 localhost,而用用户名"DanielW"创建时是通配符%,这个是 mysql 不能识别的。

### 7.4.3 解决方案

```
grant select,insert,update,delete on tcpw.* to tcpw@"localhost" identified by "tcpw";
```

注: 使用 localhost 权限。

## 8 备份与恢复

### 8.1 备份

例如：将上例创建的 dfg 库备份到文件 back\_dfg 中：

```
#cd /home/data/mysql
(进入到库目录，本例库已由 val/lib/mysql 转到/home/data/mysql)
# mysqldump -u root -p --opt dbname> dbname.bak.sql
```

### 8.2 恢复

```
mysql> create database new_dbname; (登陆 mysql，创建数据库 new_dbname)
# mysql -u root -p new_dbname < dbname.bak.sql
```

## 9 安装 MySQL Administrator 图形界面

mysql-admin 包或者 mysql-query-browser 包或者 phpMyAdmin 包。

## 10 修改数据库默认字符集

安装后的数据库编码默认是 latin1，这个在编码下，存储中文时是会乱码的，所以在使用时还得把数据库的字符集改成支持中文的字符集。

下面以 utf-8 为例，简单记录下修改字符集需要的操作：

1) 登录数据库后，使用命令 `show variables like 'character%'` 可以查看数据库使用字符集的情况，下面是在没有修改前 mysql 的字符集设置：

| Variable_name            | Value                      |
|--------------------------|----------------------------|
| character_set_client     | latin1                     |
| character_set_connection | latin1                     |
| character_set_database   | latin1                     |
| character_set_filesystem | binary                     |
| character_set_results    | latin1                     |
| character_set_server     | latin1                     |
| character_set_system     | utf8                       |
| character_sets_dir       | /usr/share/mysql/charsets/ |

2) 要把字符集换成 utf-8 的, 我们只要修改 mysql 的配置文件 my.cnf, 在其中加入下面的配置段即可:

```
在[client]下加入:
default-character-set = utf8

在[mysqld]下加入:
init_connect = 'SET NAMES utf8'
default-character-set = utf8
default-collation = utf8_general_ci
```

3) 完成上面的修改, 保存后重启 mysql。

重新登录 mysql 后, 再执行: show variables like 'character%'; 如果看到规则发生了变化了, 那么就成功了。

## 11 多事务连接

1) 原始配置文件如下所示:

```
#cat /etc/my.cnf
```

```
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0

[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
```

2) 修改之后的 my.cnf 文件如下所示:

```
#cat /etc/my.cnf
```

```
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0

[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
```

```
[mysql_multi]
mysqld = /usr/bin/mysqld_safe
mysqladmin = /usr/bin/mysqladmin
user = root

[mysqld0]
datadir=/home/PT/nativ_disk_0
socket=/var/lib/mysql/mysql.sock0
user=mysql
port=3307

[mysqld1]
datadir=/home/PT/nativ_disk_1
socket=/var/lib/mysql/mysql.sock1
user=mysql
port=3308
```

注：配置文件中的 port 很重要，由 port 确定访问的数据库实例。

### 3) 相关操作：

```
mkdir /home/PT/nativ_disk_0; mkdir /home/PT/nativ_disk_0
#建立两个目录存放新的数据库
```

```
cp -r /var/lib/mysql/mysql/ /home/PT/nativ_disk_0
#默认的 mysql 数据库复制过来
chown -R mysql:mysql /home/PT/nativ_disk_0
#该目录改为 mysql 管理权
```

```
service mysqld start      #该命令是否执行与下面的命令无关
mysqld_multi start 1      #启动 0 号实例
mysqld_multi start 2      #启动 1 号实例
```

```
mysql -h 127.0.0.1 -P 3307 #IP 需要指定的
mysql -h 127.0.0.1 -P 3308
```

Note：使用命令 mysqld\_safe 可以重建数据库 sock 等文件。

## 12 mysql 调优

```
innodb_buffer_pool_size=4GB
innodb_flush_log_at_trx_commit=2
back_log=5000
max_connections=2000
connect_timeout=50
table_cache=2048
key_buffer_size=100M
sort_buffer_size=10M
join_buffer_size=10M
read_buffer_size=100M
query_cache_size=100M
tmp_table_size=100M
myisam_sort_buffer_size=64M
thread_cache_size=200
thread_concurrency=24
```

内存缓冲区的大小  
不写硬盘，写缓冲区

## 13 启动|关闭|重建数据库

启动:

```
/usr/bin/mysqld_safe &
/usr/bin/mysqld_multi num
```

关闭:

```
mysqladmin -u root shutdown
```

重建数据库:

```
mysql_install_db
```