CentOS7环境下安装MySQL的

操作手册

目录

[1 准备工作 2](#_Toc470796003)

[1.1 添加用户组与用户 2](#_Toc470796004)

[1.2 介质 2](#_Toc470796005)

[1.3 目录规划 2](#_Toc470796006)

[2 MySQL的安装 3](#_Toc470796007)

[2.1 解压介质 3](#_Toc470796008)

[2.2 创建相关目录 3](#_Toc470796009)

[2.3 创建并配置MySQL实例的配置文件 4](#_Toc470796010)

[2.4 数据库的初始化 4](#_Toc470796011)

[2.5 清理初始化时生成的数据目录文件 6](#_Toc470796012)

[2.6 启动数据库实例 6](#_Toc470796013)

[2.7 登陆数据库 7](#_Toc470796014)

[2.8 查看必要的参数信息 7](#_Toc470796015)

[2.8.1 字符集 7](#_Toc470796016)

[2.8.2 字符比较规则 8](#_Toc470796017)

[2.9 创建应用database 8](#_Toc470796018)

[2.10 关闭数据库 8](#_Toc470796019)

[2.11 将数据库文件及binlog链接到其他目录里 8](#_Toc470796020)

[2.11.1 备份现有database文件目录 8](#_Toc470796021)

[2.11.2 创建软连接 9](#_Toc470796022)

[2.11.3 备份binlog 9](#_Toc470796023)

[2.11.4 创建软连接 9](#_Toc470796024)

[2.11.5 验证可行性 9](#_Toc470796025)

[3 用户组与权限 9](#_Toc470796026)

[3.1 root用户的处理 9](#_Toc470796027)

文档版本信息

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 版本号 | 修订人 | 提交时间 | 修改内容 | 备 注 |
|  |  |  |  |  |  |

1. 准备工作
   1. 添加用户组与用户

创建mysql用户组，命令为：

groupadd -g 1005 mysql



创建mysql用户，隶属于mysql组

useradd -u 1005 -g mysql mysql



设定mysql用户密码

passwd mysql



* 1. 介质

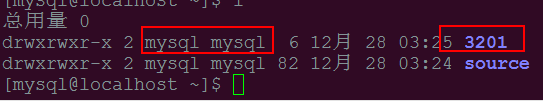
介质使用mysql-5.6.22-linux-glibc2.5-x86\_64.tar.gz，该版本已经经过编译，在CentOS7的环境下，可以直接解压使用。



* 1. 目录规划

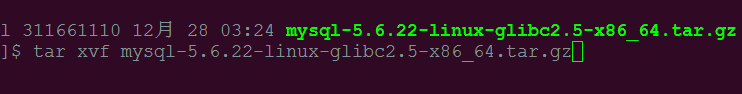
mysql用户的home目录为/home/mysql，考虑到数据文件与二进制日志文件均放在NAS上，因此每个dbms实例内容均放在/home/mysql目录下。如果home空间不足，也可以将实例内容放在其他目录下，每个实例按照服务端口号创建响应目录，该实例使用的所有二进制文件，配置文件均置于该目录下。

例如本次要创建服务在3201端口上的服务，则需要在/home/mysql下创建目录3201(source目录保存了安装介质),确保其用户组为mysql:mysql

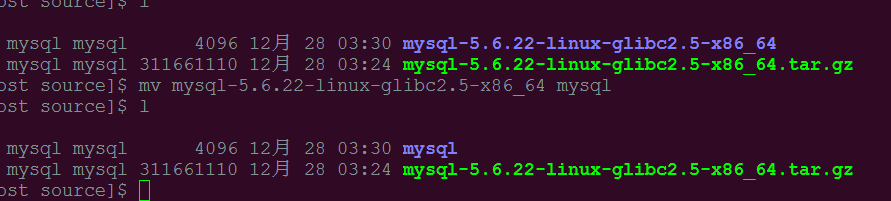


1. MySQL的安装
   1. 解压介质

tar xvf mysql-5.6.22-linux-glibc2.5-x86\_64.tar.gz

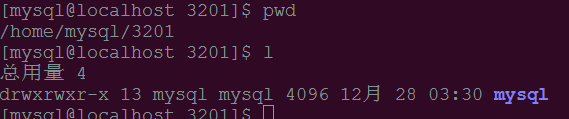


重命名解压出的目录为mysql, mv mysql-5.6.22-linux-glibc2.5-x86\_64 mysql



将重命名后的mysql目录挪至对应实例文件夹里

mv /home/mysql/source/mysql /home/mysql/3201



该目录即为该实例使用的mysql相关的二进制文件。

* 1. 创建相关目录

在实例目录下，创建cnf、conn、data、log、tmp目录

cnf为配置文件目录

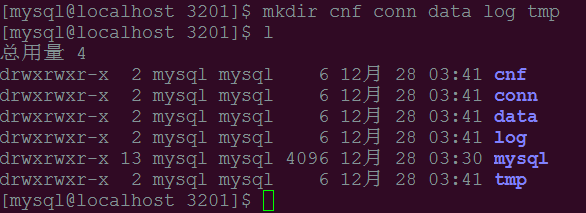
conn为创建本地连接.sock与进程.pid文件目录

data目录为数据文件目录

log目录为日志目录

tmp为临时表空间目录

mkdir cnf conn data log tmp



在log目录下,在分别创建binlog、error、general\_query、slow\_query的文件夹。

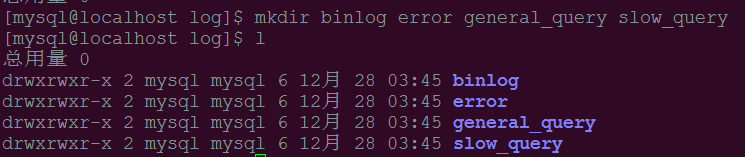
binlog用于存储二进制日志

error用于存储错误日志

general\_query可用于存储所有使用的sql，但是对性能影响较大，暂不使用

slow\_query用于存储慢查询，优化时可以有针对性。

mkdir binlog error general\_query slow\_query



* 1. 创建并配置MySQL实例的配置文件

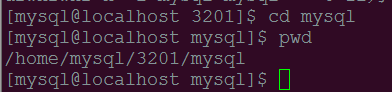
详见配置文件内容，根据实际情况调整相应参数，该配置文件主要面向使用MyISAM引擎，读性能要求大于写性能要求的情况，配置文件按my.服务端口号.cnf的方式进行命名。



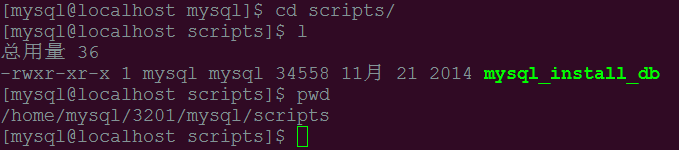
* 1. 数据库的初始化

在经过初始化之后，mysql实例启动所需要的数据表才能被建立起来，因此需要先进行初始化。

进入对应实例所使用的二进制目录



进入其scripts目录



使用以下命令初始化数据库：

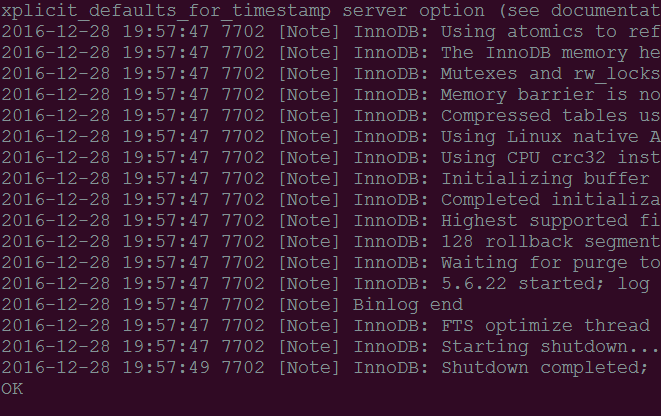
./mysql\_install\_db --datadir=/home/mysql/3201/data --basedir=/home/mysql/3201/mysql --user=mysql

其中datadir参数为数据文件保存的目录，basedir为mysql应用实例所使用的二进制文件的目录，user为实例用户

如果有报错，则按报错内容去处理，本例在安装时，缺少Perl相关组件，因此使用yum install perl-Module-Install.noarch



处理完错误后，再次运行之前的脚本

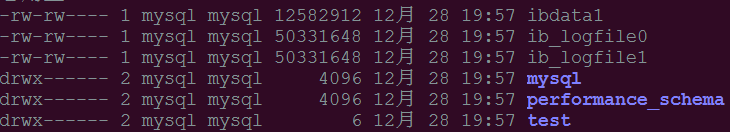


出现OK表示正常，如果其中的过程中，出现ERROR的内容，则按提示处理错误。

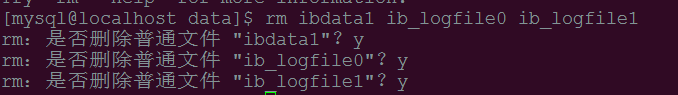
* 1. 清理初始化时生成的数据目录文件

(按文档要求进行清理，但是并没有验证不清理会如何)

进入数据文件目录(实例的data目录下)



用rm ibdata1 ib\_logfile0 ib\_logfile1命令删除掉以上三个文件，第一个为innodb引擎默认创建的第一个文件，后两个为innodb引擎使用的日志



* 1. 启动数据库实例

切换到该实例的mysql/bin目录下

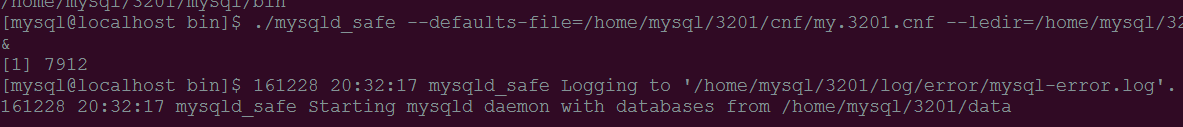


执行安全启动命令

./mysqld\_safe --defaults-file=/home/mysql/3201/cnf/my.3201.cnf --ledir=/home/mysql/3201/mysql/bin &

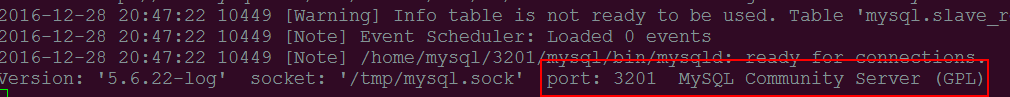
defaults-file参数表示使用的配置文件

ledir参数表示执行的mysqld等二进制文件所在位置



查看/home/mysql/3201/log/error/mysql-error.log日志文件，确定最后的运行状态。

出现红框内的内容则表明服务正常启动



* 1. 登陆数据库

由于使用的是解压安装，因此默认环境下，使用mysql命令无法打开实例。正确的方法是先进入该实例的bin目录(/home/mysql/3201/mysql/bin/)

先修改管理员密码

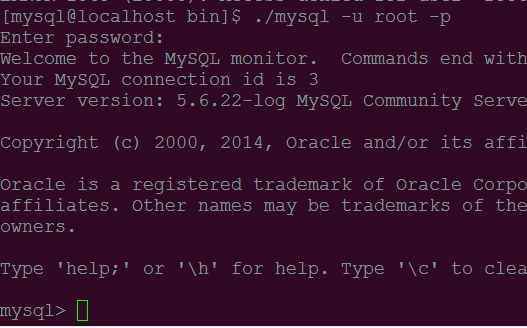
./mysqladmin -u root password



登陆数据库

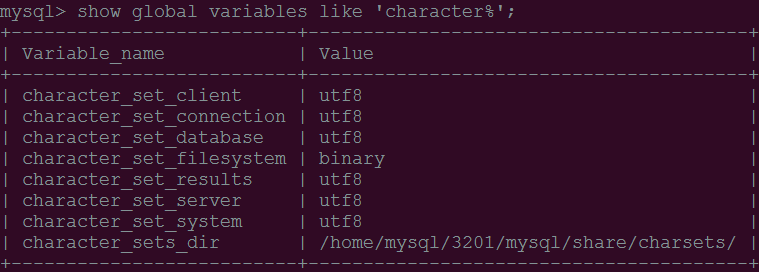
在bin目录下，执行./mysql -u root -p

并输入密码可完成登陆。



* 1. 查看必要的参数信息
     1. 字符集

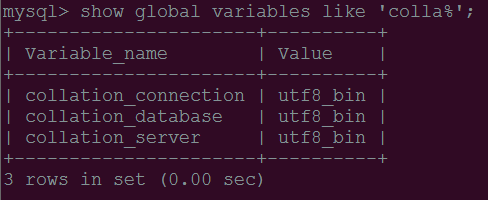
show global variables like 'character%';



请确保字符集信息和上面的结果一致，否则会产生大量乱码问题。

* + 1. 字符比较规则

show global variables like 'colla%';

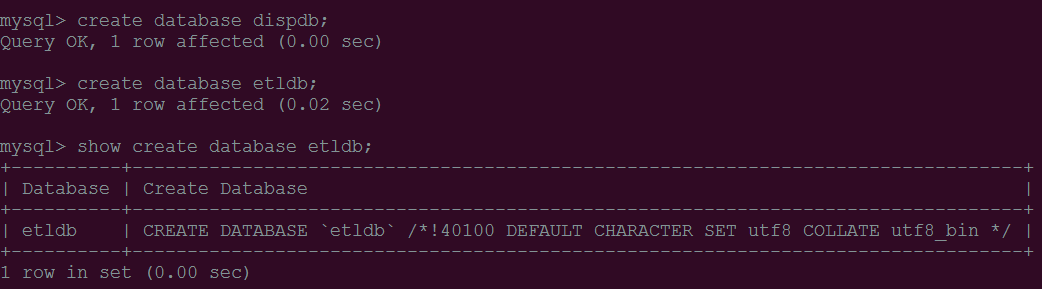


请确保字符比较规则和上面的结果一致，否则会产生order by顺序不一致的问题。

* 1. 创建应用database

create database etldb;

create database dispdb;



至此可以认为MySQL数据库实例正常工作。

* 1. 关闭数据库

在实例的bin目录下，执行

./mysqladmin -u root -p shutdown

输入密码即可关闭数据库。

* 1. 将数据库文件及binlog链接到其他目录里

(如果有)先确保NAS已经被正常挂载在了文件系统上。

* + 1. 备份现有database文件目录

本文档中，以etldb

mv etldb etldb\_bak

* + 1. 创建软连接

本文档中，意图将数据文件放在/mnt/nfs173db/3201/data下，在该目录下，按不同database分别存放。

先在/mnt/nfs173db/3201/data目录下，创建etldb文件夹

mkdir -p /mnt/nfs173db/3201/data

进行软连接

ln -s /mnt/nfs173db/3201/data/etldb /home/mysql/3201/data/etldb

再将etldb\_bak中的内容复制到etldb下

cp /home/mysql/3201/data/etldb\_bak/\* /home/mysql/3201/data/etldb

* + 1. 备份binlog

在日志目录下，先备份已经形成二进制日志

/home/mysql/3201/log/mv binlog binlog\_bak

* + 1. 创建软连接

目录结构等内容与数据文件链接方式，迁移相同，不再赘述

ln -s /mnt/nfs173db/3201/log/binlog/ /home/mysql/3201/log/binlog

* + 1. 验证可行性

重新启动数据库实例，如果可以正常启动，并且在链接目录下更新了文件修改时间，则表明链接创建成功。

1. 用户组与权限
   1. root用户的处理

只允许root从指定某些终端登录，’localhost’代表任何位置,identified by后边跟密码

grant all privileges on \*.\* to 'root'@'localhost' identified by 'root' with grant option

* 1. dba用户

允许dba用户从任何地方登录，拥有全权限

grant all privileges on \*.\* to 'dba'@'%' identified by 'dba' with grant option;

* 1. dbmnger用户

管理员用户，对某几个库拥有全部权限

#允许dbmnger用户对etldb,test,dispdb有全部权限

grant all on etldb.\* to 'dbmnger' @'%';

grant all on test.\* to 'dbmnger' @'%';

grant all on dispdb.\* to 'dbmnger' @'%';

* 1. dbuser用户

指定库的增删改查用户

#给dbuser在test库上增删改查的权限

grant select,insert,update,delete,create,drop on test.\* to 'dbuser' @'%';

#给dbuser用户在etldb库上读的权限

grant select on etldb.\* to 'dbuser' @'%';

* 1. dbreader用户

查询用户

#授予dbreader所有库上的select权限

grant select on \*.\* to 'dbreader' @'%';

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_