查看并卸载老版本

首先查看是否已经安装了mysql5.1

```
rpm -qa | grep mysql
```

卸载

```
sudo rpm -e mysql-libs --nodeps
sudo rpm -e mysql-server-5.1.73-5.el6_7.1.x86_64  --nodeps
sudo rpm -e  qt-mysql-4.6.2-28.el6_5.x86_64  --nodeps
sudo rpm -e  mysql-5.1.73-5.el6_7.1.x86_64  --nodeps

sudo rm -rf /var/lib/mysql

sudo rm -rf /usr/lib64/mysql
sudo rm -rf  /usr/share/mysql
sudo rm -rf  /usr/lib64/perl5/DBD/mysql
sudo rm -rf  /usr/lib64/perl5/auto/DBD/mysql

chkconfig --list | grep -i mysql
chkconfig --del mysqld
```

cd ~ 到 home 目录，vi mysql_install
安装脚本如下

```
#!/bin/sh
# define mysql5.5 rpm pkg url
client_pkg_url="http://resource.cmwebgame.com/resource/soft/MySQL-client-5.5.21-1.linux2.6.x86_64.rpm"
devel_pkg_url="http://resource.cmwebgame.com/resource/soft/MySQL-devel-5.5.21-1.linux2.6.x86_64.rpm"
embedded_pkg_url="http://resource.cmwebgame.com/resource/soft/MySQL-embedded-5.5.21-1.linux2.6.x86_64.rpm"
```

```
server_pkg_url="http://resource.cmwebgame.com/resource/soft/MySQL-server-5.5.21-1.linux2.6.x86_64.rpm"
shared_pkg_url="http://resource.cmwebgame.com/resource/soft/MySQL-shared-5.5.21-1.linux2.6.x86_64.rpm"
test_pkg_url="http://resource.cmwebgame.com/resource/soft/MySQL-test-5.5.21-1.linux2.6.x86_64.rpm"
libaio_pkg_url="http://mirror.centos.org/centos/6/os/x86_64/Packages/libaio-0.3.107-10.el6.x86_64.rpm"
# define rpm names
client_rpm=MySQL-client-5.5.21-1.linux2.6.x86_64.rpm
devel_rpm=MySQL-devel-5.5.21-1.linux2.6.x86_64.rpm
embedded_rpm=MySQL-embedded-5.5.21-1.linux2.6.x86_64.rpm
server_rpm=MySQL-server-5.5.21-1.linux2.6.x86_64.rpm
shared_rpm=MySQL-shared-5.5.21-1.linux2.6.x86_64.rpm
test_rpm=MySQL-test-5.5.21-1.linux2.6.x86_64.rpm
libaio_rpm=libaio-0.3.107-10.el6.x86_64.rpm
if [ -f $client_rpm ] ;then
   echo "$client_rpm has download --OK"
else
  echo "begin download $client_rpm"
  wget $client_pkg_url
fi
if [ -f $devel_rpm ] ;then
   echo "$devel_rpm has download --OK"
else
  echo "begin download $devel_rpm"
  wget $devel_pkg_url
fi
if [ -f $embedded_rpm ] ;then
   echo "$embedded_rpm has download --OK"
else
  echo "begin download $embedded_rpm"
  wget $embedded_pkg_url
fi
if [ -f $server_rpm ] ;then
   echo "$server_rpm has download --OK"
else
  echo "begin download $server_rpm from $server_pkg_url"
  wget $server_pkg_url
fi
if [ -f $shared_rpm ] ;then
   echo "$shared_rpm has download --OK"
```

```
else
  echo "begin download $shared_rpm"
  wget $shared_pkg_url
fi
if [ -f $test_rpm ] ;then
   echo "$test_rpm has download --OK"
else
  echo "begin download $test_rpm"
  wget $test_pkg_url
fi
if [ -f $libaio_rpm ] ;then
   echo "$libaio_rpm has download --OK"
else
  echo "begin download $libaio_rpm"
  wget $libaio_pkg_url
fi
echo "mysql5.5 rpm pkgs have been downloaded OK"
echo "begin install rpm packages ..."
sudo rpm -ivh libaio-0.3.107-10.el6.x86_64.rpm
sudo rpm -ivh MySQL-server-5.5.21-1.linux2.6.x86_64.rpm
sudo rpm -ivh MySQL-devel-5.5.21-1.linux2.6.x86_64.rpm
sudo rpm -ivh MySQL-embedded-5.5.21-1.linux2.6.x86_64.rpm
sudo rpm -ivh MySQL-client-5.5.21-1.linux2.6.x86_64.rpm
sudo rpm -ivh MySQL-shared-5.5.21-1.linux2.6.x86_64.rpm
sudo rpm -ivh MySQL-test-5.5.21-1.linux2.6.x86_64.rpm
echo "------All Mysql RPM packages have been installed:OK------"
```

修改执行权限后，执行脚本，进行安装mysql

```
chmod 700 mysql_install
./mysql_install
```

测试是否能启动

```
sudo /etc/init.d/mysql start && ps -ef | grep -i mysql
```

测试启动成功成功后，停止，进行其他配置

```
sudo /etc/init.d/mysql stop
```

copy模板配置文件并修改配置文件

```
cd /usr/share/mysql
sudo cp my-medium.cnf /etc/my.cnf
sudo vi /etc/my.cnf
```

完整配置可以直接复制最终文件配置即可

[client]新加

```
default-character-set = utf8
```

[mysqld]下增加设定字符集

```
lower_case_table_names=1
init_connect='SET collation_connection=utf8_general_ci'
init_connect='SET NAMES utf8'
character-set-server=utf8
collation-server=utf8_general_ci
skip-character-set-client-handshake
tmpdir=/var/lib/mysql/temp
```

开启Uncomment the following if you are using InnoDB tables下面开启

```
innodb_data_home_dir = /var/lib/mysql
innodb_data_file_path = ibdata1:10M:autoextend
```

```
innodb_log_group_home_dir = /var/lib/mysql
# You can set ..._buffer_pool_size up to 50 - 80 %
# of RAM but beware of setting memory usage too high
innodb_buffer_pool_size = 16M
innodb_additional_mem_pool_size = 2M
# Set ..._log_file_size to 25 % of buffer pool size
innodb_log_file_size = 5M
innodb_log_buffer_size = 8M
innodb_flush_log_at_trx_commit = 1
innodb_lock_wait_timeout = 50
```

[mysql]新加

```
default-character-set = utf8
```

最终文件参考：

```
# Example MySQL config file for medium systems.
#
# This is for a system with little memory (32M - 64M) where MySQL plays
# an important part, or systems up to 128M where MySQL is used together with
# other programs (such as a web server)
#
# MySQL programs look for option files in a set of
# locations which depend on the deployment platform.
# You can copy this option file to one of those
# locations. For information about these locations, see:
# http://dev.mysql.com/doc/mysql/en/option-files.html
#
# In this file, you can use all long options that a program supports.
# If you want to know which options a program supports, run the program
# with the "--help" option.

```

```
# The following options will be passed to all MySQL clients
[client]
#password   = your_password
port    = 3306
socket  = /var/lib/mysql/mysql.sock
default-character-set = utf8

# Here follows entries for some specific programs

# The MySQL server
[mysqld]
lower_case_table_names=1
init_connect='SET collation_connection=utf8_general_ci'
init_connect='SET NAMES utf8'
character-set-server=utf8
collation-server=utf8_general_ci
skip-character-set-client-handshake
tmpdir=/var/lib/mysql/temp

port    = 3306
socket  = /var/lib/mysql/mysql.sock
skip-external-locking
key_buffer_size = 16M
max_allowed_packet = 1M
table_open_cache = 64
sort_buffer_size = 512K
net_buffer_length = 8K
read_buffer_size = 256K
read_rnd_buffer_size = 512K
myisam_sort_buffer_size = 8M

# Don't listen on a TCP/IP port at all. This can be a security enhancement,
# if all processes that need to connect to mysqld run on the same host.
# All interaction with mysqld must be made via Unix sockets or named pipes.
# Note that using this option without enabling named pipes on Windows
# (via the "enable-named-pipe" option) will render mysqld useless!
#
```

```
#skip-networking

# Replication Master Server (default)
# binary logging is required for replication
log-bin=mysql-bin

# binary logging format - mixed recommended
binlog_format=mixed

# required unique id between 1 and 2^32 - 1
# defaults to 1 if master-host is not set
# but will not function as a master if omitted
server-id   = 1

# Replication Slave (comment out master section to use this)
#
# To configure this host as a replication slave, you can choose between
# two methods :
#
# 1) Use the CHANGE MASTER TO command (fully described in our manual) -
# the syntax is:
#
# CHANGE MASTER TO MASTER_HOST=<host>, MASTER_PORT=<port>,
# MASTER_USER=<user>, MASTER_PASSWORD=<password> ;
#
# where you replace <host>, <user>, <password> by quoted strings and
# <port> by the master's port number (3306 by default).
#
# Example:
#
# CHANGE MASTER TO MASTER_HOST='125.564.12.1', MASTER_PORT=3306,
# MASTER_USER='joe', MASTER_PASSWORD='secret';
#
# OR
#
# 2) Set the variables below. However, in case you choose this method, then
# start replication for the first time (even unsuccessfully, for example
```

```
# if you mistyped the password in master-password and the slave fails to
# connect), the slave will create a master.info file, and any later
# change in this file to the variables' values below will be ignored and
# overridden by the content of the master.info file, unless you shutdown
# the slave server, delete master.info and restart the slaver server.
# For that reason, you may want to leave the lines below untouched
# (commented) and instead use CHANGE MASTER TO (see above)
#
# required unique id between 2 and 2^32 - 1
# (and different from the master)
# defaults to 2 if master-host is set
# but will not function as a slave if omitted
#server-id = 2
#
# The replication master for this slave - required
#master-host = <hostname>
#
# The username the slave will use for authentication when connecting
# to the master - required
#master-user = <username>
#
# The password the slave will authenticate with when connecting to
# the master - required
#master-password = <password>
#
# The port the master is listening on.
# optional - defaults to 3306
#master-port = <port>
#
# binary logging - not required for slaves, but recommended
#log-bin=mysql-bin

# Uncomment the following if you are using InnoDB tables
innodb_data_home_dir = /var/lib/mysql
```

```
innodb_data_file_path = ibdata1:10M:autoextend
innodb_log_group_home_dir = /var/lib/mysql
# You can set .._buffer_pool_size up to 50 - 80 %
# of RAM but beware of setting memory usage too high
innodb_buffer_pool_size = 16M
innodb_additional_mem_pool_size = 2M
# Set .._log_file_size to 25 % of buffer pool size
innodb_log_file_size = 5M
innodb_log_buffer_size = 8M
innodb_flush_log_at_trx_commit = 1
innodb_lock_wait_timeout = 50

[mysqldump]
quick
max_allowed_packet = 16M

[mysql]
default-character-set = utf8
no-auto-rehash
# Remove the next comment character if you are not familiar with
SQL
#safe-updates

[myisamchk]
key_buffer_size = 20M
sort_buffer_size = 20M
read_buffer = 2M
write_buffer = 2M

[mysqlhotcopy]
interactive-timeout

```

创建临时文件目录

```
sudo  mkdir /var/lib/mysql/temp
sudo chown -R mysql. /var/lib/mysql/temp
```

启动mysql

```
sudo /etc/init.d/mysql start && ps -ef | grep -i mysql
```

执行以下脚本进行mysql的设置

```
cd /usr/bin
sudo ./mysql_secure_installation
```

创建用户
注意：此处的"localhost"，是指该用户只能在本地登录，不能在另外一台机器上远程登录。如果想远程登录的话，将"localhost"改为"%"，表示在任何一台电脑上都可以登录。也可以指定某台机器可以远程登录。

```
#用设置好的root用户登陆
mysql -uroot -pcoxxxxxxxxx -hlocalhost

##授权，userxxxxxx为用户名；dbpasswdxxxxxx为密码
grant all privileges on *.* to userxxxxxx@'%' Identified by 'dbpasswdxxxxxx';
flush privileges;
```