



链滴

# Centos 选择版本安装 mysql

作者: [luomuren](#)

原文链接: <https://ld246.com/article/1647833581612>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



## 1. 下载Mysql源

mysql官网地址: [mysql官网](#)

可以先查看下centos版本, 已选择对应的Mysql版本:

```
<pre><div class="hljs"><code class="lang-shell">cat /etc/redhat-release </code></div>
</pre>
```

```
[root@iZ8vb6tda6e8mx7yz8e7s7Z ~]# cat /etc/redhat-release
CentOS Linux release 8.1.1911 (Core)
```

下载mysql源:

当前官网最新源, 最好去前往官网按照centos版本获取最新源下载链接, 不然很容易报错

```
<pre><div class="hljs"><code class="lang-shell">//centos7:
wget https://dev.mysql.com/get/mysql80-community-release-el7-3.noarch.rpm
//centos6:
wget https://dev.mysql.com/get/mysql80-community-release-el6-3.noarch.rpm </code></div>
</pre>
```

注: 如果出现以下错误:

无法解析主机地址wget: 无法解析主机地址

执行命令:

```
<pre><div class="hljs"><code class="lang-shell">echo 'nameserver 8.8.8.8'>>/etc/resolv.co
f </code></div>
</pre>
```

## 安装mysql源

注: mysql80-community-release-el8-1.noarch.rpm请更改为您下载的文件

```
<pre> <div class="hljs"> <code class="lang-shell">sudo rpm -Uvh mysql80-community-relea
e-el8-1.noarch.rpm </code> </div>
</pre>
```

安装成功后,可以在/etc/yum.repos.d/文件夹中看见两个repo文件

```
-bash: cd: /usr/yum/repos.d: No such file or directory
[root@iZ8vb6tda6e8mx7yz8e7s7Z ~]# cd /etc/yum.repos.d
[root@iZ8vb6tda6e8mx7yz8e7s7Z yum.repos.d]# ls
CentOS-AppStream.repo  CentOS-Debuginfo.repo  CentOS-Media.repo  mysql-community.repo
CentOS-Base.repo      CentOS-epel.repo       CentOS-PowerTools.repo  mysql-community-source.repo
CentOS-centosplus.repo  CentOS-Extras.repo     CentOS-Sources.repo
CentOS-CR.repo         CentOS-fasttrack.repo  CentOS-Vault.repo
```

## 2.安装Mysql

注: 默认安装最新版mysql, 如果需要更换版本, 请按以下代码执行:

```
<pre> <div class="hljs"> <code class="lang-shell">sudo yum-config-manager --disable mysql
80-community //禁用8.x版本 </code> </div>
</pre>
```

```
[root@olr-diaodu soft]# sudo yum-config-manager --disable mysql80-community
Loaded plugins: fastestmirror, refresh-packagekit
----- repo: mysql80-community -----
[mysql80-community]
bandwidth = 0
base_persistdir = /var/lib/yum/repos/x86_64/6
baseurl = http://repo.mysql.com/yum/mysql-8.0-community/el/6/x86_64/
cache = 0
cachedir = /var/cache/yum/x86_64/6/mysql80-community
cost = 1000
enabled = False
enablegroups = True
exclude =
failovermethod = priority
gpgcadir = /var/lib/yum/repos/x86_64/6/mysql80-community/gpgcadir
gpgkey =
gpgcheck = True
gpgdir = /var/lib/yum/repos/x86_64/6/mysql80-community/gpgdir
gpgkey = file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql
hdrdir = /var/cache/yum/x86_64/6/mysql80-community/headers
http_caching = all
includepkgs =
keepalive = True
mdpolicy = group:primary
mediaid =
metadata_expire = 21600
metalink =
mirrorlist =
mirrorlist_expire = 86400
name = MySQL 8.0 Community Server
old_base_cache_dir =
password =
persistdir = /var/lib/yum/repos/x86_64/6/mysql80-community
pkgdir = /var/cache/yum/x86_64/6/mysql80-community/packages
proxy =
proxy_dict =
proxy_password =
proxy_username =
repo_gpgcheck = False
retries = 10
skip_if_unavailable = False
```

```
<pre> <div class="hljs"> <code class="lang-shell">sudo yum-config-manager --enable mysql
7-community //开启5.7版本 </code> </div>
</pre>
```

```
[root@solr-diaodul soft]# sudo yum-config-manager --enable mysql57-community
Loaded plugins: fastestmirror, refresh-packagekit
===== repo: mysql57-community =====
[mysql57-community]
bandwidth = 0
base_persistdir = /var/lib/yum/repos/x86_64/6
baseurl = http://repo.mysql.com/yum/mysql-5.7-community/el/6/x86_64/
cache = 0
cachedir = /var/cache/yum/x86_64/6/mysql57-community
cost = 1000
enabled = True
enablegroups = True
exclude =
failovermethod = priority
gpgcadir = /var/lib/yum/repos/x86_64/6/mysql57-community/gpgcadir
gpgcakey =
gpgcheck = True
gpgdir = /var/lib/yum/repos/x86_64/6/mysql57-community/gpgdir
gpgkey = file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql
hdrdir = /var/cache/yum/x86_64/6/mysql57-community/headers
http_caching = all
includepkgs =
keepalive = True
mdpolicy = group:primary
mediaid =
metadata_expire = 21600
metalink =
mirrorlist =
mirrorlist_expire = 86400
name = MySQL 5.7 Community Server
```

安装mysql, 执行如下语句即可

```
<pre> <div class="hljs"> <code class="lang-Shell">sudo yum install mysql-server </code> </div> </pre>
```

注意: 如果这里安装使用的命令时sudo yum install mysql-community-server会报错:

```
<pre> <div class="hljs"> <code class="lang-Shell">No match for argument: mysql-communit
-server
Error: Unable to find a match: mysql-community-server> </code> </div> </pre>
```

```
mysql57-community source      MySQL 5.7 Community Server    disabled
[root@iZ8vb6tda6e8mx7yz8e7s7Z ~]# sudo yum install mysql-community-server
Last metadata expiration check: 0:01:20 ago on Wed 01 Apr 2020 08:22:28 PM CST.
No match for argument: mysql-community-server
Error: Unable to find a match: mysql-community-server
```

### 3.启动mysql服务

```
<pre> <div class="hljs"> <code class="lang-Shell">sudo systemctl start mysqld.service
//centos6请使用以下命令:
service mysqld start </code> </div> </pre>
```

注: 如果发生以下错误, 请参照本系列文章: [centos6.5安装mysql5.7时,启动服务失败](#)

```
[root@solr-diaodul soft]# service mysqld start
初始化 MySQL 数据库: 2020-12-23T06:44:13.524313Z 0 [Warning] TIMESTAMP with implicit DEFAULT value is deprecated. Please use --explicit_defaults_for_timestamp server option (see document
ation for more details).
2020-12-23T06:44:13.526951Z 0 [ERROR] --initialize specified but the data directory has files in it. Aborting.
2020-12-23T06:44:13.526974Z 0 [ERROR] Aborting

[...]
```

### 4.进入Mysql

使用root账号登陆，第一次登陆不需要输入密码，直接回车即可：

```
<pre><div class="hljs"><code class="lang-shell">mysql -uroot -p </code></div>
</pre>
```

注：如果需要密码，通过如下命令可以在日志文件中找出密码：

```
<pre><div class="hljs"><code class="lang-shell">grep "password" /var/log/mysql.log </c
de></div>
</pre>
```

修改密码：

```
<pre><div class="hljs"><code class="lang-sql"> <span class="hljs-keyword">alter</span>
span class="hljs-keyword">user</span> <span class="hljs-string">'root'</span>@<span cla
s="hljs-string">'localhost'</span> <span class="hljs-keyword">IDENTIFIED</span> <span c
ass="hljs-keyword">BY</span> <span class="hljs-string">'XXX'</span>;
<span class="hljs-keyword">FLUSH</span> <span class="hljs-keyword">PRIVILEGES</span>
; //立即刷新 </code></div>
</pre>
```

如果遇到策略问题导致无法修改密码，可以执行下面语句：

设置密码强度为弱：set global validate\_password\_policy=LOW;  
可以修改validate\_password\_policy=0,这样就是只检查长度

我们觉着8位太长了，我们可以改为4：set global validate\_password\_length=4;

然后下次登陆的时候就可以使用新设置的密码了。

## 5.设置远程登录

进入mysql，使用新的密码进入mysql -uroot -p，一般来说，远程登录不应该使用root账户，所以们新建一个账号

```
<pre><div class="hljs">
  <p><span>//设置root远程登录：<span>grant all privileges on *.* to root@"%" identified by
密码";</span></span></p>
  <p><code class="lang-sql">//创建新用户 <span class="hljs-keyword">create</span> <spa
class="hljs-keyword">user</span> mysql_user;</code></p>
  <code class="lang-sql">//授权，%表示是所有的外部机器，如果指定某一台机，就将%改为相应
机器名 <span class="hljs-keyword">GRANT</span> ALL <span class="hljs-keyword">ON</span> </s
an> *.* <span class="hljs-keyword">TO</span> <span class="hljs-string">'mysql_user'</sp
n>@<span class="hljs-string">'%'</span>;
//更新密码 <span class="hljs-keyword">ALTER</span> <span class="hljs-keyword">USER</
pan> <span class="hljs-string">'mysql_user'</span>@<span class="hljs-string">'%'</span>
<span class="hljs-keyword">IDENTIFIED</span> <span class="hljs-keyword">WITH</span>
mysql_native_password <span class="hljs-keyword">BY</span> <span class="hljs-string">'x
x'</span>;
//强制刷新 <span class="hljs-keyword">flush</span> <span class="hljs-keyword">privileges
/span>; </code></div>
</pre>
```

此时，查询用户权限情况

进入mysql数据库，

```
<pre><div class="hljs"><code class="lang-sql"><span class="hljs-keyword">select</span>
host,<span class="hljs-keyword">user</span> <span class="hljs-keyword">from</span> <
span class="hljs-keyword">user</span>; </code></div>
</pre>
```

可以看到mysql\_user已经允许远程访问

```
mysql> use mysql;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed

mysql> select host,user from user;
+-----+-----+
| host      | user          |
+-----+-----+
| %         | mysql.user   |
| localhost | mysql.infoschema |
| localhost | mysql.session |
| localhost | mysql.sys    |
| localhost | root         |
+-----+-----+
5 rows in set (0.00 sec)

https://blog.csdn.net/lzh6hao0
```

然后退出mysql，使用exit;

```
mysql> exit;
Bye
```

查看3306端口是否开启

```
[root@iZ8vb6tda6e8mx7yz8e7s7Z ~]# netstat -an |grep 3306
tcp6      0      0 :::3306          :::*              LISTEN
tcp6      0      0 :::33060         :::*              LISTEN
```

这表示3306端口已对所有远程开放

但是到这一步，使用navicat连接mysql还是不行。我们还得在某云上配置下安全策略  
进到你云服务器的安全组中，添加一组新的规则

网卡类型: 内网

规则方向: 入方向

授权策略: 允许

协议类型: MySQL (3306)

\* 端口范围: 3306/3306

优先级: 1

授权类型: IPv4地址段访问

\* 授权对象: 0.0.0.0/0

描述: Mysql

长度为2-256个字符, 不能以http://或https://开头。

取值范围从1到65535; 设置格式例如 "1/200"、"80/80", 其中"-1/-1"不能独设置, 代表不限制端口。

教我设置

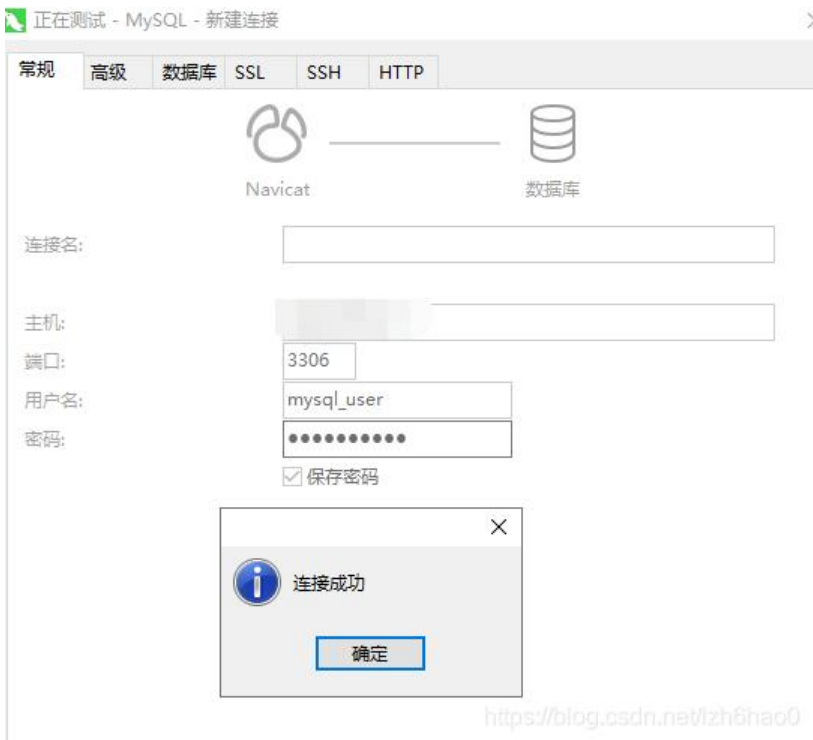
确定 取消

<https://blog.csdn.net/vzh6hao0>

配置好之后, 可以重启一下系统。

如果mysql没有设置开机启动的话, 你需要重新启动一下服务

然后使用navicat连接mysql,连接成功。



到这就搞定收工。