



链滴

Can't Connect to MySQL Server on IP Address (10061) 错误的解决方案

作者: [HiJiangChuan](#)

原文链接: <https://ld246.com/article/1637320532411>

来源网站: 链滴

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



如果你打算从[远程连接 MySQL 服务器](#)的话，有可能会碰到 10061 错误，这个错误特别常见，通常错误提示是「Driver Error, Can't connect to MySQL server on 'YOUR_IP_ADDRESS' (10061)」

导致 10061 这个错误的情况有两种

- 登录账号远程访问权限问题
- MySQL 配置文件设置问题

本教程将详细讲解，如何针对这两种情况进行配置，以修正 10061 错误。

特别提示：关于如何打开 MySQL 远程访问功能，可看这篇《[如何远程连接 MySQL 数据库，阿里云外网连接教程](#)》，如果想开启服务器可查看这份教程，本篇教程只讲开启后，为什么会出现 10061 错误。另外推荐一下卡拉云，只要你能写 SQL，不会任何前端也可以用[卡拉云](#)快速搭建属于自己的台管理系统，详见本文文末

一. 授权登录 MySQL 服务器的账号远程访问权限

如果账号没有远程访问权限或 host 配置错误，会导致 10061 错误。我们可以新建一个账号用于远程登录，也可以修改已有账号的 host 配置，使它可以远程访问。

1. 新建用于远程登录的 MySQL 账号

MySQL 用户账号是否可以远程登录，取决于账号中的 **host** 配置。host 指定该账号在哪些主机上可登录，如果是本地用户可用 **localhost**，如果是远程用户，需要指定远程计算机的 IP，如果想任意主机均可登录，那么可以使用通配符 %

本教程使用通配符 % 来作为账号 host 的设置，你可以根据自己的情况将 % 改为指定主机 IP，这样

以是 MySQL 远程登录更加安全。

首先登录 MySQL Server

```
mysql -u root -p
```

然后新建一个用于远程登录的 MySQL 账号，这里的「password」换成你的密码，如果MySQL 设置严格密码的话，需要「数字+英文大小写+符号」

```
CREATE USER 'kalacloud.com-remote'@'%' IDENTIFIED WITH mysql_native_password BY 'password';
```

接着，根据自己的需要，给你用于远程访问的账号赋予权限。下面的例子是给账号全局权限，包括创（CREATE）、修改（ALTER）、删除（DROP）数据库、表、用户，任意表的插入（INSERT）、更（UPDATE）、删除（DELETE）操作权限。可以使用 SELECT 查询数据，使用 REFERENCES 建立外关系权限，以及使用 RELOAD 权限执行 FLUSH 操作的权限。

当然，你也可以根据自己需求，对账号权限进行调整。

```
GRANT CREATE, ALTER, DROP, INSERT, UPDATE, DELETE, SELECT, REFERENCES, RELOAD on *.* TO 'kalacloud.com-remote'@'%' WITH GRANT OPTION;
```

最后，运行 FLUSH PRIVILEGES 命令，刷新 MySQL 的系统权限相关表，更新缓存。

```
FLUSH PRIVILEGES;
```

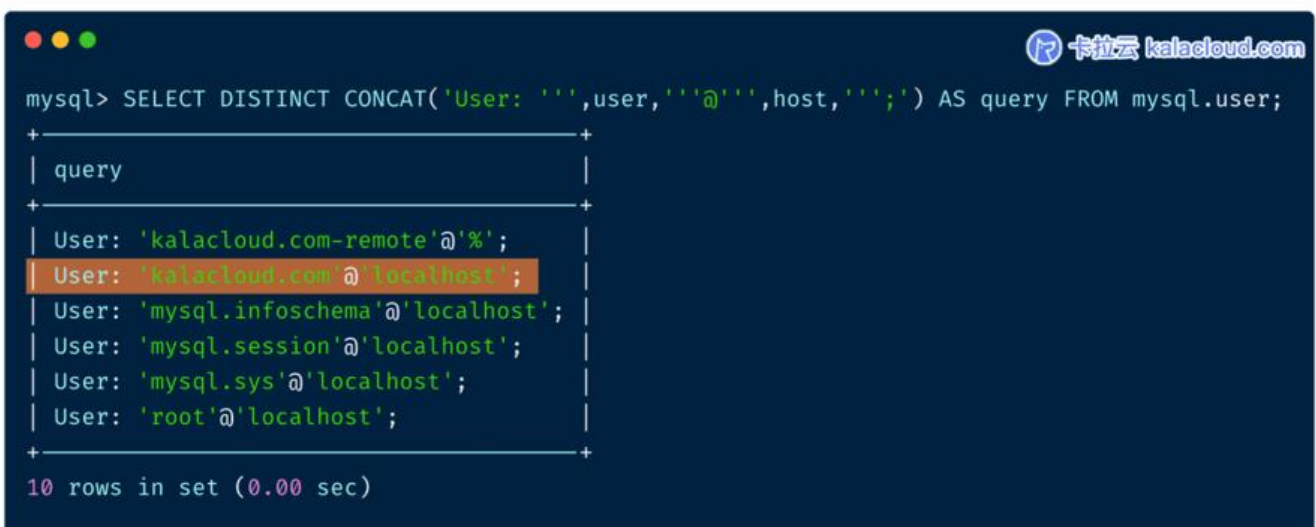
扩展阅读：[最好用的 10 款 MySQL / MariaDB 管理工具横向测评 - 免费和付费到底怎么选?](#)

2. 将已有账号更改为可远程登录的账号

如果你不想新建账号，只是想沿用已有的MySQL 登录账号，那么我们也可以直接把它改为可远程登录的账号。

首先，我们查看 MySQL 所有账号

```
SELECT DISTINCT CONCAT('User: ',user,'@',host,';') AS query FROM mysql.user;
```



```
mysql> SELECT DISTINCT CONCAT('User: ',user,'@',host,';') AS query FROM mysql.user;
+-----+
| query                                     |
+-----+
| User: 'kalacloud.com-remote'@'%';         |
| User: 'kalacloud.com'@'localhost';        |
| User: 'mysql.infoschema'@'localhost';     |
| User: 'mysql.session'@'localhost';       |
| User: 'mysql.sys'@'localhost';           |
| User: 'root'@'localhost';                |
+-----+
10 rows in set (0.00 sec)
```

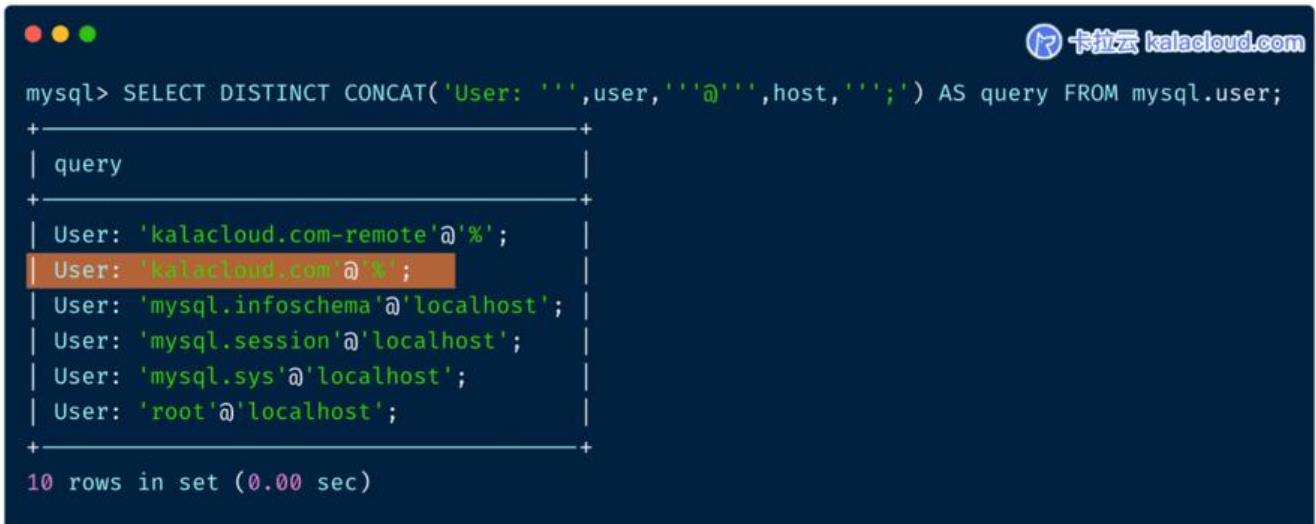
接下来，我们把 'kalacloud.com'@'localhost' 这个账号改为可远程登录的账号。我们把这个账号 hos

改为任意主机 (%) 或者是固定主机 ip 。可以使用 RENAME USER 命令来实现:

```
RENAME USER 'kalacloud.com'@'localhost' TO 'kalacloud.com'@'%';
```

执行后, 再查看 MySQL 所有账号列表

```
SELECT DISTINCT CONCAT('User: ',user,'@',host,';') AS query FROM mysql.user;
```

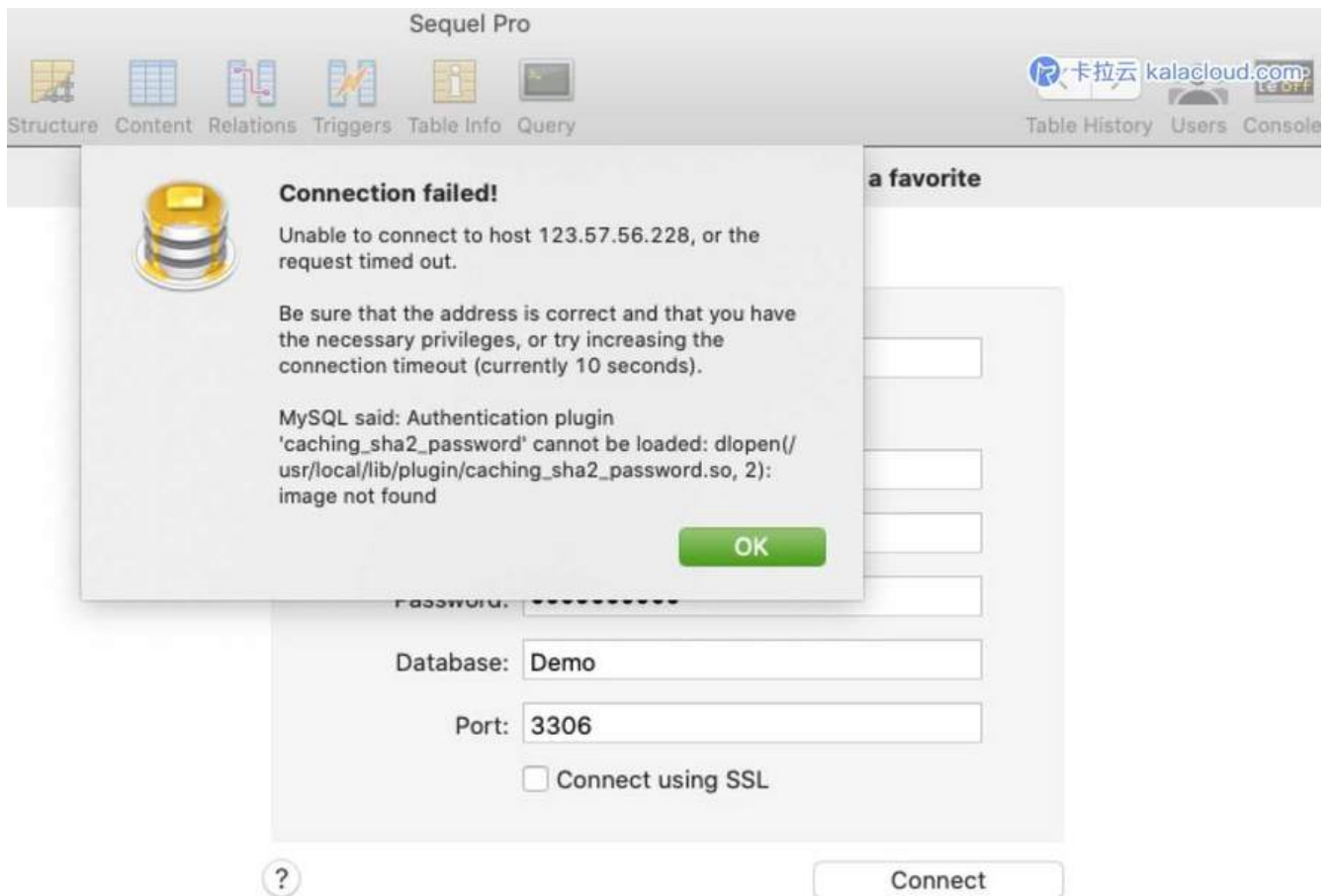


The screenshot shows a terminal window with a dark blue background. At the top right, there is a logo for '卡拉云 kalacloud.com'. The terminal prompt is 'mysql>'. The command entered is 'SELECT DISTINCT CONCAT('User: ',user,'@',host,';') AS query FROM mysql.user;'. The output is a table with one column named 'query'. The first row is 'User: 'kalacloud.com-remote'@'%';'. The second row is 'User: 'kalacloud.com'@'%';', which is highlighted in orange. The other rows are 'User: 'mysql.infoschema'@'localhost';', 'User: 'mysql.session'@'localhost';', 'User: 'mysql.sys'@'localhost';', and 'User: 'root'@'localhost';'. At the bottom, it says '10 rows in set (0.00 sec)'.

我们可以看到 'kalacloud.com'@'localhost' 已经变为 'kalacloud.com'@'%' 了。

特别提示: 如果你使用的是 MySQL 8.0, 使用这个账号远程登录可能会出现 `caching_sha2_password` 报错, 这是因为 MySQL 8.0 默认认证方式改为 `SHA2` 了, 如果不支持 `SHA2` 插件认证方式, 那么就报错, 比如使用 Workbench 和 Sqel Pro 登录时, 如下图。

```
MySQL said: Authentication plugin 'caching_sha2_password' cannot be loaded: dlopen(/usr/local/lib/plugin/caching_sha2_password.so, 2): image not found
```



我们可以把使用 `caching_sha2_plugin` 认证的 `'kalacloud'@'%'` 账号改为使用密码认证。

```
ALTER USER 'kalacloud.com'@'%' IDENTIFIED WITH mysql_native_password BY 'password';
```

最后运行 `FLUSH PRIVILEGES` 命令 刷新 MySQL 系统权限相关表，更新缓存

```
FLUSH PRIVILEGES;
```

至此，MySQL 远程登录账号部分就配置完成了，如果还是报 10061 错误，那么你还应该检查 MySQL 配置文件是否有问题。

扩展阅读：[MySQL Workbench 中文使用指南](#)

二. 改 MySQL 配置文件

在默认情况下，MySQL 数据库仅监听本地连接。如果想让外网远程连接到数据库，我们需要修改配置文件，让 MySQL 可以监听远程固定 ip 或者监听所有远程 ip。

首先打开 `mysqld.cnf` 配置文件。

```
sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf
```

找到 `bind - address` 这一行，如下图所示。


```
root@kalacloud: ~
GNU nano 4.8 /etc/mysql/mysql.conf.d/mysql.d.cnf 卡拉云 Modified
[mysqld]
#
# * Basic Settings
#
user                = mysql
# pid-file          = /var/run/mysqld/mysqld.pid
# socket            = /var/run/mysqld/mysqld.sock
# port              = 3306
# datadir           = /var/lib/mysql

# If MySQL is running as a replication slave, this should be
# changed. Ref https://dev.mysql.com/doc/refman/8.0/en/server-system-variables.html#sysvar-tmpdir
# tmpdir            = /tmp
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address         = 127.0.0.1
|

^G Get Help      ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify
^X Exit          ^R Read File   ^\ Replace     ^U Paste Text  ^T To Spell
```

默认情况下，bind-address 的值为 127.0.0.1，所以只能监听本地连接。我们需要将这个值改为远连接 ip 可访问，可使用通配符 ip 地址 *，::，0.0.0.0，当然也可以是单独的固定 ip，这样就仅允许指定 ip 连接，更加安全。

```
root@kalacloud: ~
GNU nano 4.8 /etc/mysql/mysql.conf.d/mysqld.cnf 卡拉云 kalacloud.com
[mysqld]
#
# * Basic Settings
#
user                = mysql
# pid-file          = /var/run/mysqld/mysqld.pid
# socket            = /var/run/mysqld/mysqld.sock
# port              = 3306
# datadir           = /var/lib/mysql

# If MySQL is running as a replication slave, this should be
# changed. Ref https://dev.mysql.com/doc/refman/8.0/en/server-system-variables.html#sysvar-tmpdir
# tmpdir            = /tmp
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address        = 0.0.0.0

^G Get Help      ^O Write Out    ^W Where Is    ^K Cut Text     ^J Justify
^X Exit          ^R Read File   ^\ Replace     ^U Paste Text  ^T To Spell
```

提示：在某些 MySQL 版本的配置文件中，没有 bind - address 这一行，这种情况下，在合适的位置加上就可以了。

提示：在某些 MySQL 版本的配置文件中，没有 bind - address 这一行，这种情况下，在合适的位置加上就可以了。

更改后，保存并退出编辑器（使用 CTRL+X 保存并退出 nano 编辑器。）

然后重启 MySQL 服务，使刚刚编辑的 `mysqld.cnf` 文件生效：

```
sudo systemctl restart mysql
```

扩展阅读：[如何在 MySQL / MariaDB 中查找和删除重复记录？](#)

三. 总结

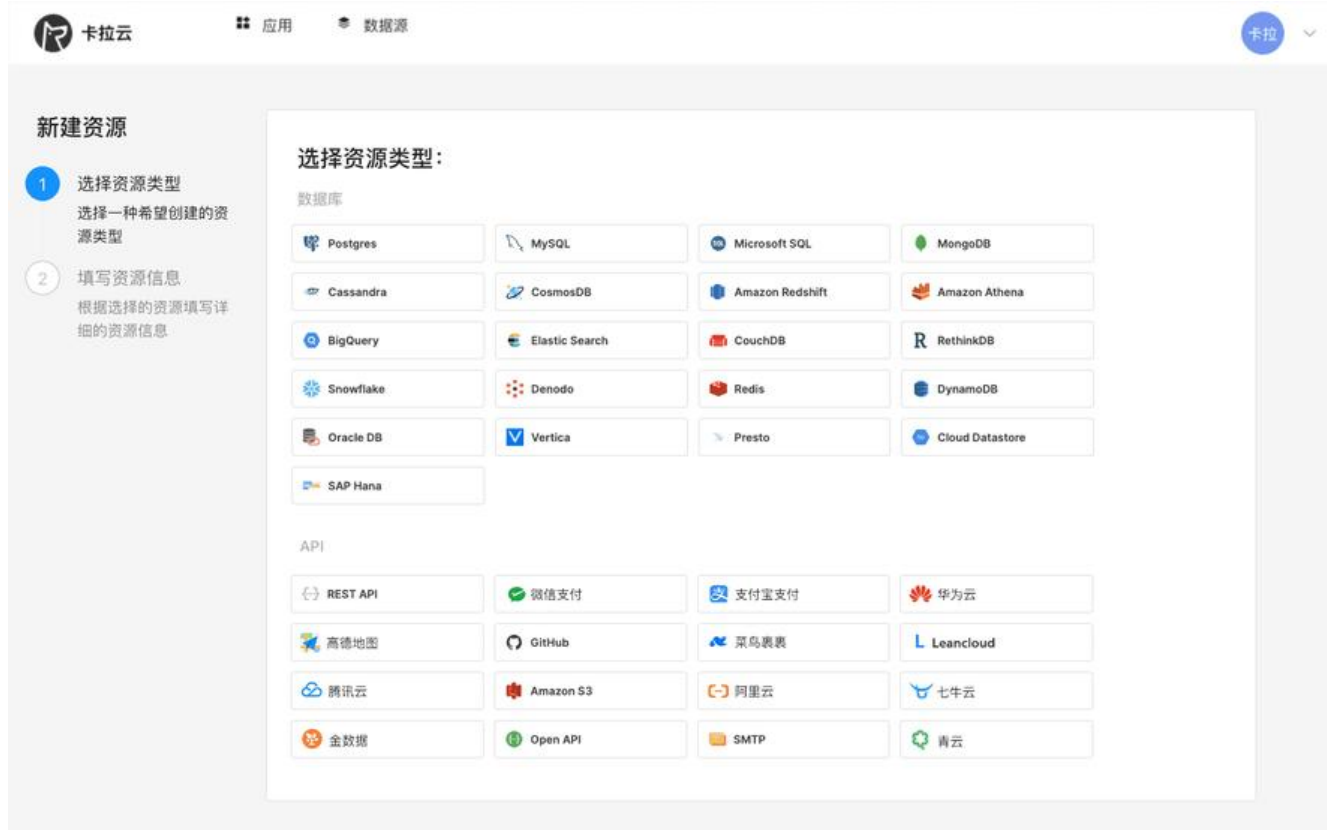
到这里，只要你跟着教程走，一定已经解决了 MySQL 10061 错误。

MySQL 除了远程连接这类适合使用终端命令操作外，大多数对 MySQL / MariaDB 数据导入导出操还是为了数据展示、分析、协同共享等产品和运营层面的应用场景。

比如后端工程师接到产品需求，协助导出某类数据等场景，如果这类需求频繁出现，推荐使用我开发卡拉云。卡拉云是新一代低代码开发工具，免安装部署，可一键接入包括 MySQL 在内的常见数据库及 API。

卡拉云不仅可以像命令行一样灵活，还可根据自己的 workflow，定制开发。无需繁琐的前端开发，只需

简单拖拽，即可快速搭建企业内部工具。数月的开发工作量，使用卡拉云后可缩减至数天，[欢迎免费用卡拉云](#)。



卡拉云可快速接入的常见数据库及 API

卡拉云可根据公司 workflow 需求，轻松搭建数据看板，并且可分享给组内的小伙伴共享数据



原文链接: [Can't Connect to MySQL Server on IP Address \(10061\) 错误的解决方案](#)

仅需拖拽一键生成前端代码，简单一行代码即可映射数据到指定组件中。

优惠券发放后台

coupon_id	code	优惠券名称	优惠券类型	核销状态	金额	满	减	用户类型	发券人	发券时间	过期时间	发券
892	U9xJ8B3H	新春满减	2	已使用		300	100	2	销售铁蛋	2023-01-01	2031-01-01	新增
896	Yxt6fso6	现金直减	1	已使用	200			3	销售小课	2023-01-02	2024-01-01	知平
891	XFcu8pfn	老用户回馈券	1	未使用	100			1	运营铁柱	2021-11-01	2021-12-31	合作
895	nKlUq8mLK	快乐父亲节	2	已过期		200	10	1	运营铁柱	2020-01-23	2020-12-31	朋友
894	3AMz7Dol	阳光普照	1	未使用				2	运营小棒	2023-01-01	2024-12-01	拉新
893	T24EQ8WU	老用户激活券	1	已使用	100			3	主管张铁人	2021-07-09	2022-12-31	旧群

优惠券信息

优惠券名称: 老用户回馈券

优惠券类型: 现金券

现金券金额: 100

满减券, 满: [] 减: []

优惠券基本规则

固定期, 有效期至: 2021-12-31

领券后, 生效天数: []

用户类型: 全用户

发券人操作信息

发券人: 运营铁柱

发券原因: 合作厂商渠道销售

更新「优惠券」 生成「优惠券」

SQL语句

```
INSERT INTO users (coupon_name,type,amount,over,by,date,day,user_type,operator,reason) VALUES ({{coupon_name.value}},{{type.value}},{{amount.value}},{{over.value}},{{by.value}},{{date.value}},{{day.value}},{{user_type.value}},{{operator.value}},{{reason.value}});
```

成功时触发 = INSERT INTO users (coupon_name,type,amount,over,by,date,day,user_type,operator,reason) VALUES ("老用户回馈券","1","100","","","2021-12-31","","1","运营铁柱","合作厂商渠道销售");

卡拉云可直接添加导出按钮，导出适用于各类分析软件的数据格式，方便快捷。立即开通卡拉云，搭属于自己的后台管理工具。

如果觉得我的文章对你有帮助，还请点个赞再走。欢迎评论区一起交流。