Ubuntu Server 16

背景

最近正在学nodejs,想到曾经有台云服务器，但是很久不用了，由于怕麻烦，一股脑的把云主机重装了个Ubuntu系统，于是配置MySQL成了配置服务中的一个环节（node用不用MySQL不管，主要是闲的重新配置一个-.-），但是配置的过程中，遇到不少问题，所以在解决一系列问题后留篇博备以后使用。

步骤

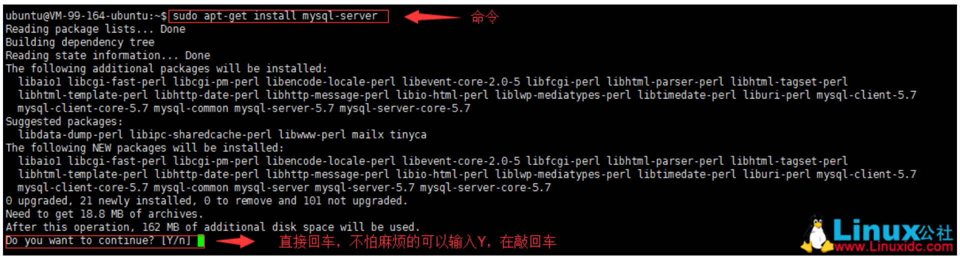
1.安装MySQL

由于本人用的是Ubuntu Server 16.04，用的XShell工具，没用桌面版，所以没有高大上的图形界面，一股脑用软件源提供的mysql即可。

命令如下：

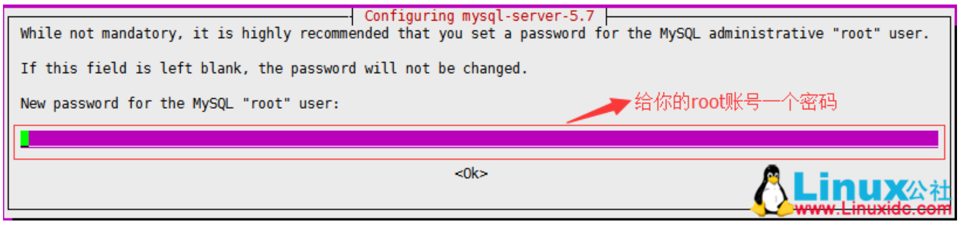
```
sudoapt-get installmysql-server
```

运行完这句命令后，不是root的话会要你输入root密码，密码输入正确后，系统就自动给你下载MySQL了，如下图：



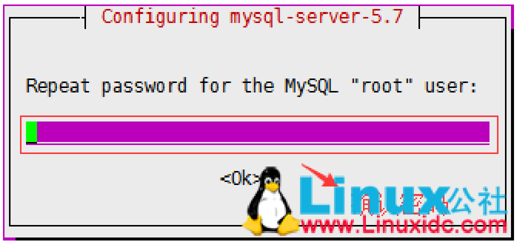
```
ubuntu@VM-99-164-ubuntu:~$ sudo apt-get install mysql-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libaio1 libcgi-fast-perl libcgi-pm-perl libencode-locale-perl libevent-core-2.0-5 libfcgi-perl libhtml-parser-perl libhtml-tagset-perl
  libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl liblwp-mediatypes-perl libtimedate-perl liburi-perl mysql-client-5.7
  mysql-client-core-5.7 mysql-common mysql-server-5.7 mysql-server-core-5.7
Suggested packages:
  libdata-dump-perl libipc-sharedcache-perl libwww-perl mailx tinyca
The following NEW packages will be installed:
  libaio1 libcgi-fast-perl libcgi-pm-perl libencode-locale-perl libevent-core-2.0-5 libfcgi-perl libhtml-parser-perl libhtml-tagset-perl
  libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl liblwp-mediatypes-perl libtimedate-perl liburi-perl mysql-client-5.7
  mysql-client-core-5.7 mysql-common mysql-server mysql-server-5.7 mysql-server-core-5.7
0 upgraded, 21 newly installed, 0 to remove and 101 not upgraded.
Need to get 18.8 MB of archives.
After this operation, 162 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

执行完上面步骤后，会进入一个“图形界面:)”，用于创建MySQL的root密码，如图：



```
Configuring mysql-server-5.7
While not mandatory, it is highly recommended that you set a password for the MySQL administrative "root" user.

If this field is left blank, the password will not be changed.

New password for the MySQL "root" user:

<Ok>
```

输入完后回车，会让你确认密码，如图：



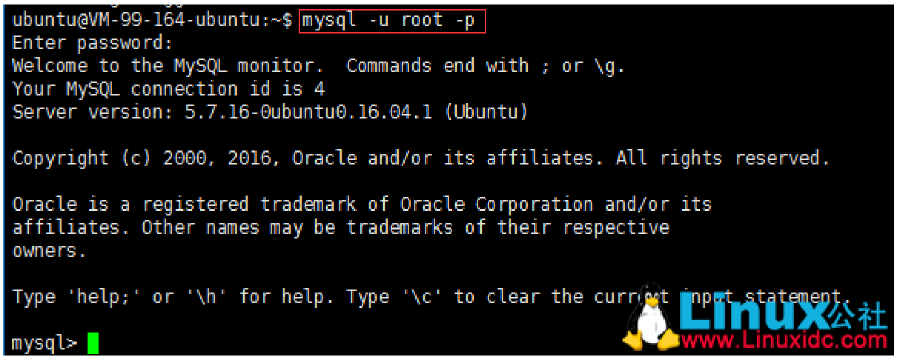
```
Configuring mysql-server-5.7

Repeat password for the MySQL "root" user:

<Ok>
```

2.授权用户，并允许远程登录

两次密码输入无误的话，系统就帮你下载完MySQL了，可是默认的MySQL只有一个root账号，所以不妨先建一个和root一样权利的账号，并授权远程登陆的许可，那么我们先登录MySQL：

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

系统会要你输入密码，密码输入无误后，进入MySQL：



```
ubuntu@VM-99-164-ubuntu:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.7.16-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> 
```

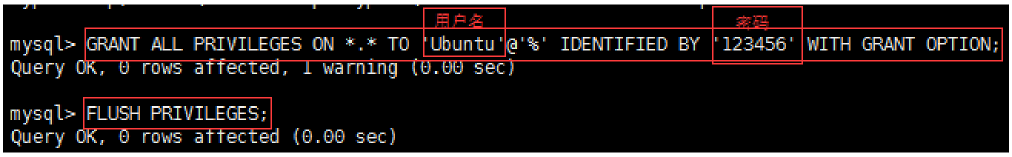
首先我们授权一个叫Ubuntu（叫什么由你定）的账户，并授予它远程连接的权力，命令如下：

```
GRANT ALL PRIVILEGES ON *.* TO 'Ubuntu'@'%' IDENTIFIED BY '123456'WITH GRANT OPTION;
```

运行完后紧接着输入，以更新数据库：

```
FLUSH PRIVILEGES;
```

效果如下图：



```
mysql> GRANT ALL PRIVILEGES ON *.* TO 'Ubuntu'@'%' IDENTIFIED BY '123456' WITH GRANT OPTION;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)
```

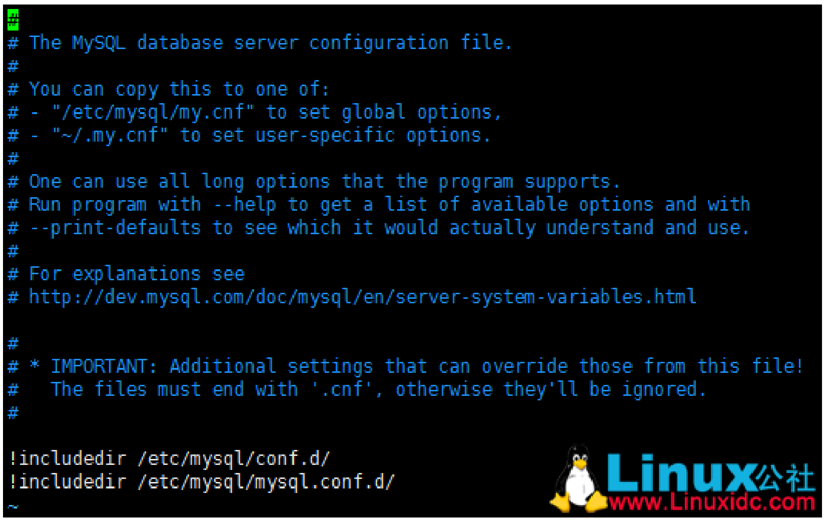
执行quit退出mysql。

由于MySQL默认支队本地使用，没有开放远程连接，于是需要到配置文件中去修改，当然新版的MySQL不知道为什么配置文件和以前不太一样了，以前都放在:/etc/mysql/my.cnf里，但是现在我们去看看变成什么样了：

运行：

```
sudovi /etc/mysql/my.cnf
```

结果发现my.cnf里的内容是这样的，本人个人猜测是MySQL优化结构了，效果如图：



```
#
# The MySQL database server configuration file.
#
# You can copy this to one of:
# - "/etc/mysql/my.cnf" to set global options,
# - "~/.my.cnf" to set user-specific options.
#
# One can use all long options that the program supports.
# Run program with --help to get a list of available options and with
# --print-defaults to see which it would actually understand and use.
#
# For explanations see
# http://dev.mysql.com/doc/mysql/en/server-system-variables.html

#
# * IMPORTANT: Additional settings that can override those from this file!
#   The files must end with '.cnf', otherwise they'll be ignored.
#

!includedir /etc/mysql/conf.d/
!includedir /etc/mysql/mysql.conf.d/
~
```

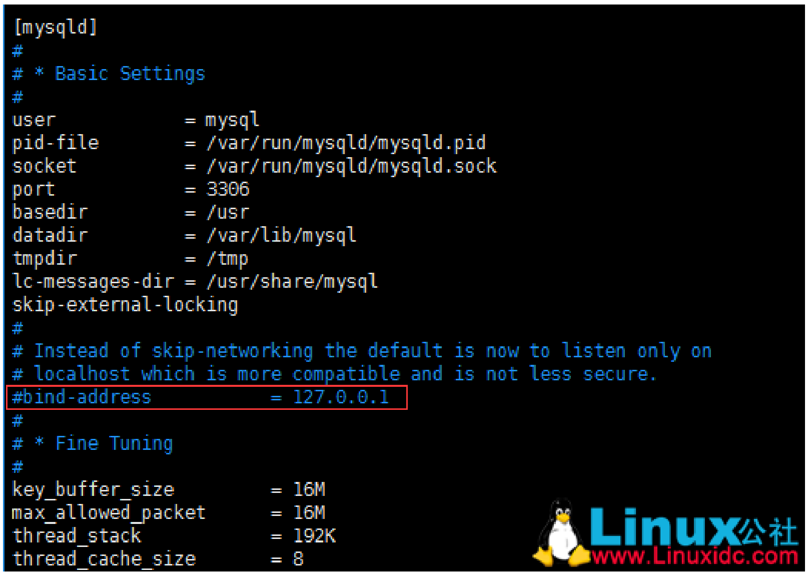
原来的配置文件变成了包含目录结构，于是在上面提到的两个目录里找找，很快就能找到配置文件原来

是：/etc/mysql/mysql.conf.d/mysqld.cnf

用管理员权限编辑之：

```
sudovi /etc/mysql/mysql.conf.d/mysqld.cnf
```

添加'#'注释掉其中的"bind-address = 127.0.0.1"，如下图：

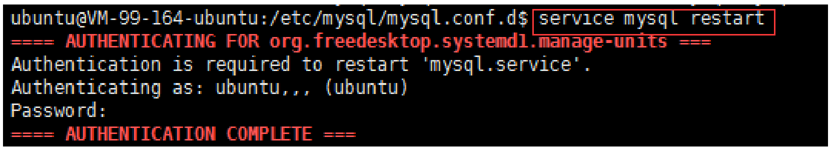


```
[mysqld]
#
# * Basic Settings
#
user            = mysql
pid-file        = /var/run/mysqld/mysqld.pid
socket          = /var/run/mysqld/mysqld.sock
port            = 3306
basedir         = /usr
datadir         = /var/lib/mysql
tmpdir          = /tmp
lc-messages-dir = /usr/share/mysql
skip-external-locking
#
# 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
#
# * Fine Tuning
#
key_buffer_size         = 16M
max_allowed_packet      = 16M
thread_stack            = 192K
thread_cache_size       = 8
```

注释后:wq保存，重启MySQL服务：

```
service mysql restart
```

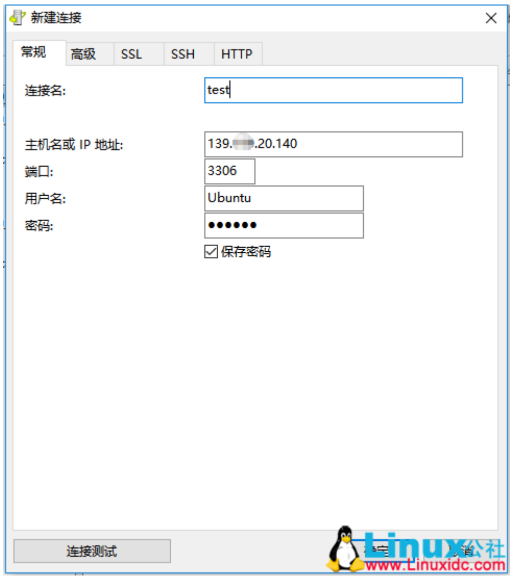
验证完你的Ubuntu密码后，重启服务成功！



```
ubuntu@VM-99-164-ubuntu:/etc/mysql/mysql.conf.d$ service mysql restart
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ===
Authentication is required to restart 'mysql.service'.
Authenticating as: ubuntu,,, (ubuntu)
Password:
==== AUTHENTICATION COMPLETE ===
```

3. 测试验证

我用Windows下的Navicat for MySQL试试，配置信息如下（打码防被黑:D）：



连接名: test

主机名或 IP 地址: 139.[redacted].20.140

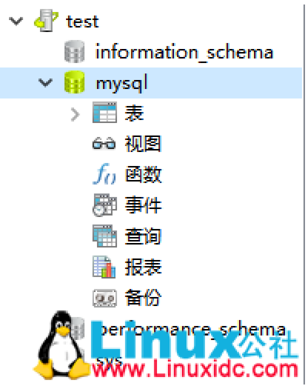
端口: 3306

用户名: Ubuntu

密码: ●●●●●●

☑ 保存密码

看看效果吧：



总结

本人是深受网上各种描述，但是由于部分不详细，导致出错，所以特开此文，希望能帮到用到的人！

更多Ubuntu相关信息见Ubuntu 专题页面 http://www.linuxidc.com/topicnews.aspx?tid=2

本文永久更新链接地址：http://www.linuxidc.com/Linux/2017-01/139502.htm