### 安装环境要求

一台操作系统为ubuntu18.04的服务器。

### 配置环境

###### 安装docker

在线安装。

参考网址：<https://www.runoob.com/docker/ubuntu-docker-install.html>

使用一键安装脚本。

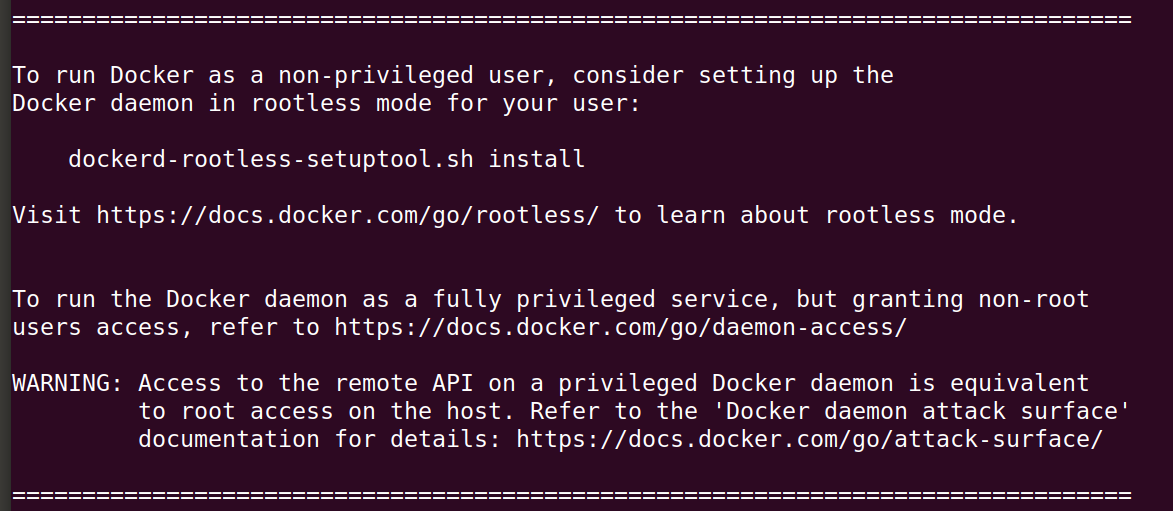
curl -fsSL https://get.docker.com | bash -s docker --mirror Aliyun



需要先安装curl。



装好后回来继续装docker。



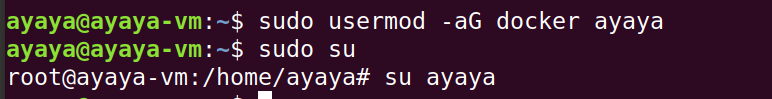
安装完成。

将当前用户添加到docker组（需要重新登录用户）：

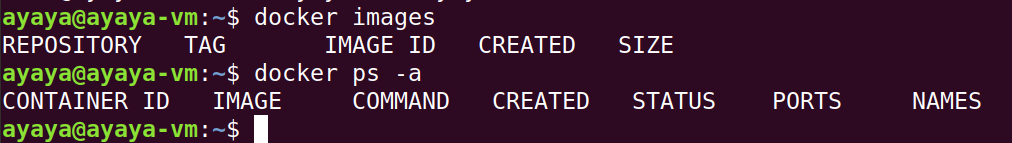
sudo usermod -aG docker your-user

sudo su

su your-user （需要将这里的your-user替换成实际的用户名）



测试一下是否安装成功：

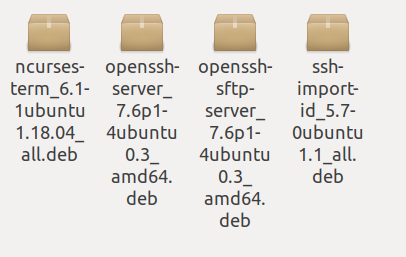


没有报错，成功。

###### 在宿主机上安装openssh服务端

参考网址：<https://blog.csdn.net/u010432354/article/details/102703767>

将 ssh离线安装材料 文件夹下的四个包复制到任意位置，这里以home目录为例。



有四个包需要安装

ncurses-term ncurses-term\_6.1-1ubuntu1.18.04\_all.deb

openssh-server openssh-server\_7.6p1-4ubuntu0.3\_amd64.deb

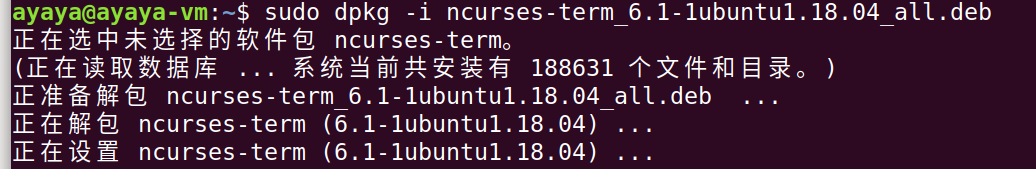
openssh-sftp-server openssh-sftp-server\_7.6p1-4ubuntu0.3\_amd64.deb

ssh-import-id ssh-import-id\_5.7-0ubuntu1.1\_all.deb

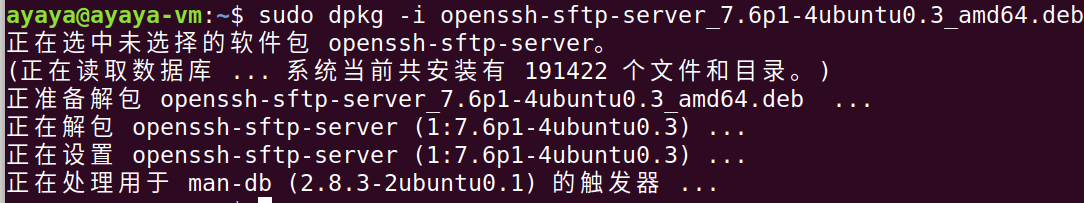
安装命令：sudo dpkg -i xxxx.deb （需要把xxxx替换成对应的包名字）

四个包的安装有依赖问题，因此要按一定顺序。

先安装ncurses-term。



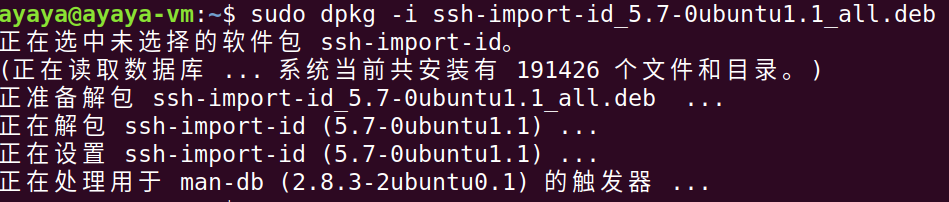
再安装openssh-sftp-server。



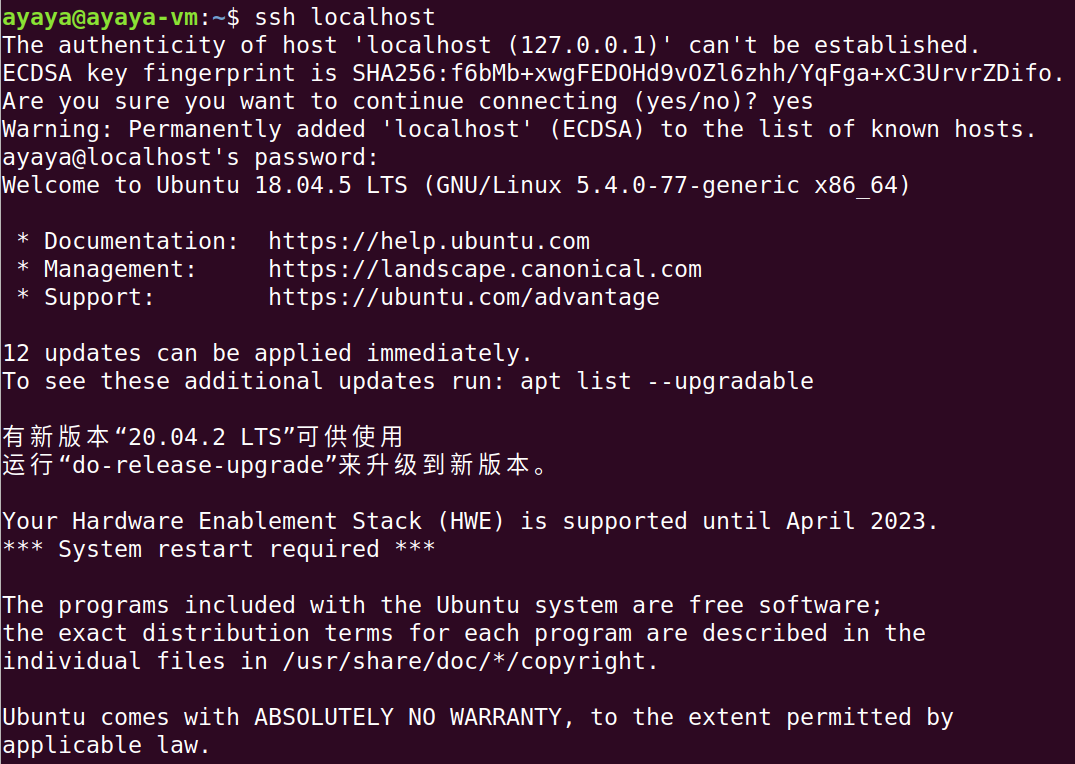
再安装openssh-server。



最后安装ssh-import-id。



测试一下是否成功。

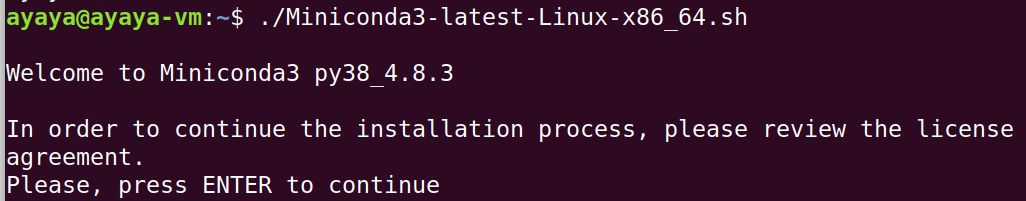


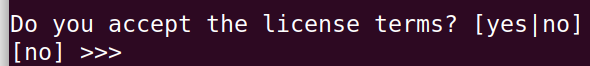
没有报错，安装成功。

###### 安装miniconda3

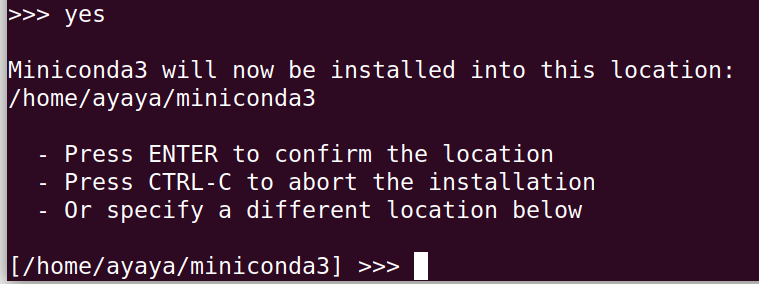
将Miniconda3-latest-Linux-x86\_64.sh文件复制到任意目录，这里以home目录为例。

运行该脚本。

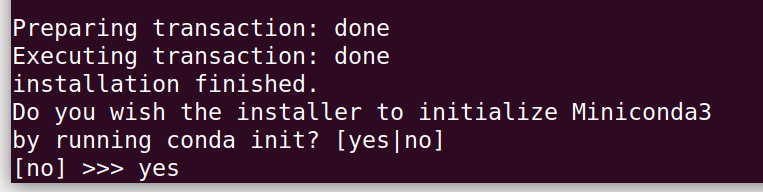




输入yes。



直接回车。



这里很重要，要输入yes而不是默认的no。

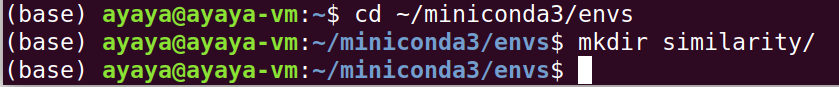
安装完成后退出终端重新进入，命令提示符前面出现（base）即安装成功。



###### Conda环境迁移安装

将similarity.tar.gz复制到任意目录，这里以home目录为例。

创建环境文件夹：先cd ~/miniconda3/envs/，然后mkdir similarity/



回到similarity.tar.gz所在目录下，tar -xzvf similarity.tar.gz -C ~/miniconda3/envs/similarity/

激活环境：source ~/miniconda3/bin/activate

conda activate similarity

前面的括号变成（similarity）



最后unpack：conda-unpack

检查是否安装成功：

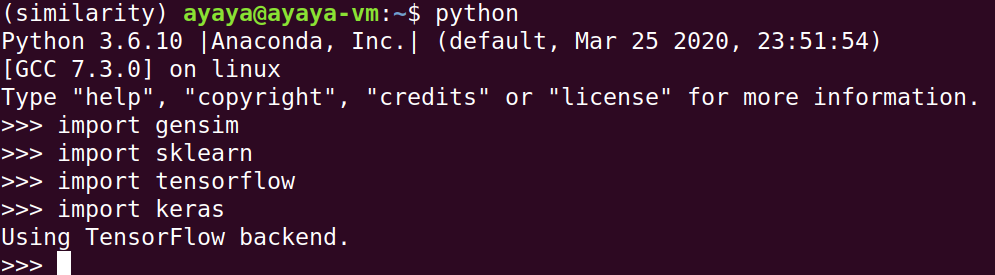
打开python，尝试import所有以下需要的库，如果没有报错则说明环境配置成功：

import gensim

import sklearn

import tensorflow

import keras



没有报错，安装成功。

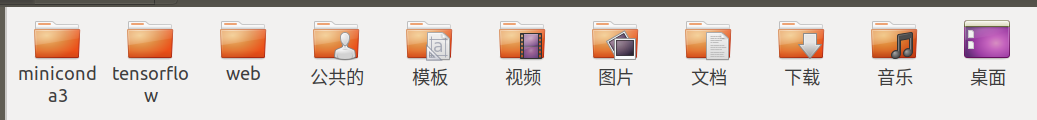
###### 部署web端代码和后端深度学习代码

在home目录下创建tensorflow文件夹。



将similarity-multi/文件夹放在该目录下。

将web/目录放在home目录下（解压就行了）。



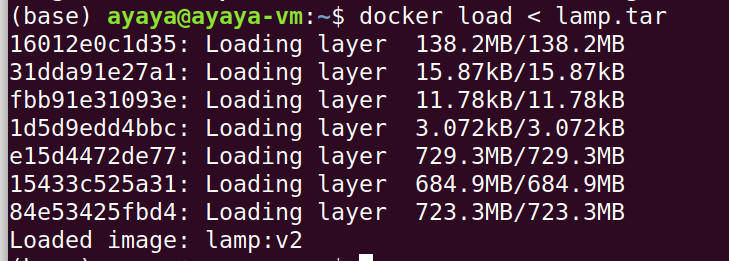
###### Docker镜像导入

把lamp\_v3.tar复制到任意位置，这里以home目录为例。

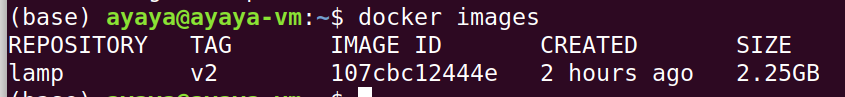
docker load < lamp\_v3.tar

（如果出现docker提示permission denied，就再次使用sudo su切换到root用户，再su your-user切换回来即可）

(这里图没有更新，用的是老版本的lamp.tar,要改成lamp\_v3.tar)

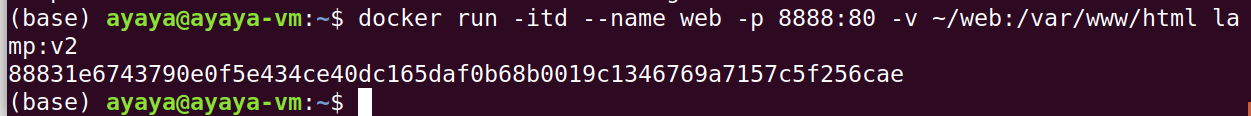


docker images 验证是否导入成功：



###### 从镜像启动容器并指定目录映射

docker run -itd --name web -p 8888:80 -v ~/web:/var/www/html lamp:v3



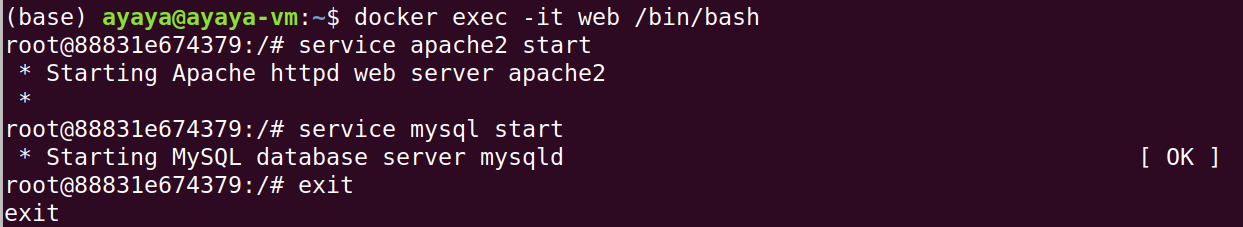
进入容器手动启动服务：

docker exec -it web /bin/bash

service apache2 start

service mysql start

exit



注意：如果service mysql start执行失败：

docker中mysql启动失败的解决方法：

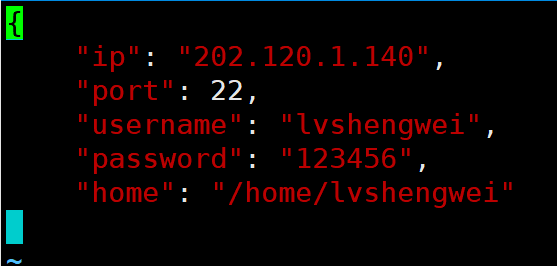
参考地址：<https://blog.csdn.net/think_ycx/article/details/84233981>

chown -R mysql:mysql /var/lib/mysql /var/run/mysqld && service mysql start

###### 设置web目录权限以及配置文件

web目录路径：~/web

在web目录下找到配置文件config.json，并直接修改其中的配置项。



将图中这些项修改成实际服务器的信息。

第一项ip修改成docker宿主的ip地址，一般情况下是172.17.0.1。

port是端口，因为该端口用于与后台进行ssh连接，所以写22。

username写服务器的实际用户名

password写服务器的实际密码

home写服务器当前用户的home目录路径，一般是/home/your-user

对web目录进行权限设置，否则会报错：

sudo chmod 777 -R ~/web

