# [centos7 下通过nginx+uwsgi部署django应用](http://projectsedu.com/2017/08/15/centos7-%E4%B8%8B%E9%80%9A%E8%BF%87nginx-uwsgi%E9%83%A8%E7%BD%B2django%E5%BA%94%E7%94%A8/)

##### **1. 安装python3.6**

1. 获取

wget https://www.python.org/ftp/python/3.6.2/Python-3.6.2.tgz

tar -xzvf Python-3.6.2.tgz -C /tmp

cd /tmp/Python-3.6.2/

2. 把Python3.6安装到 /usr/local 目录

./configure --prefix=/usr/local

make

make altinstall

3. 更改/usr/bin/python链接

ln -s /usr/local/bin/python3.6 /usr/bin/python3

##### **2. maridb**

1. 安装

sudo yum install mariadb-server

2. 启动， 重启

sudo systemctl start mariadb

sudo systemctl restart mariadb

3. 设置bind-ip

vim /etc/my.cnf

在 [mysqld]:

下面加一行

bind-address = 0.0.0.0

4. 设置外部ip可以访问

先进入mysql才能运行下面命令:

mysql 直接进入就行

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

FLUSH PRIVILEGES

5. 设置阿里云的对外端口

视频中有讲解这部分

6. 安装mysqlclient出问题

centos 7：

yum install python-devel mariadb-devel -y

ubuntu：

sudo apt-get install libmysqlclient-dev

然后：

pip install mysqlclient

##### **3. 安装nginx**

https://www.digitalocean.com/community/tutorials/how-to-install-nginx-on-centos-7

##### **4. 安装virtualenvwrapper**

yum install python-setuptools python-devel

pip install virtualenvwrapper

编辑.bashrc文件

export WORKON\_HOME=$HOME/.virtualenvs

source /usr/local/bin/virtualenvwrapper.sh

重新加载.bashrc文件

source ~/.bashrc

新建虚拟环境

mkvirtualenv mxonline

进入虚拟环境

workon mxonline

安装pip包

我们可以通过 pip freeze > requirements.txt 将本地的虚拟环境安装包相信信息导出来

然后将requirements.txt文件上传到服务器之后运行：

pip install -r requirements.txt

安装依赖包

##### **5. 安装uwsgi**

pip install uwsgi

##### **6. 测试uwsgi**

uwsgi --http :8000 --module MxOnline.wsgi

##### **7. 配置nginx**

新建uc\_nginx.conf

# the upstream component nginx needs to connect to

upstream django {

# server unix:///path/to/your/mysite/mysite.sock; # for a file socket

server 127.0.0.1:8000; # for a web port socket (we'll use this first)

}

# configuration of the server

server {

# the port your site will be served on

listen 80;

# the domain name it will serve for

server\_name 你的ip地址 ; # substitute your machine's IP address or FQDN

charset utf-8;

# max upload size

client\_max\_body\_size 75M; # adjust to taste

# Django media

location /media {

alias 你的目录/Mxonline/media; # 指向django的media目录

}

location /static {

alias 你的目录/Mxonline/static; # 指向django的static目录

}

# Finally, send all non-media requests to the Django server.

location / {

uwsgi\_pass django;

include uwsgi\_params; # the uwsgi\_params file you installed

}

}

##### **8. 将该配置文件加入到nginx的启动配置文件中**

sudo ln -s 你的目录/Mxonline/conf/nginx/uc\_nginx.conf /etc/nginx/conf.d/

##### **8. 拉取所有需要的static file 到同一个目录**

在django的setting文件中，添加下面一行内容：

STATIC\_ROOT = os.path.join(BASE\_DIR, "static/")

运行命令

python manage.py collectstatic

##### **9. 运行nginx**

sudo /usr/sbin/nginx

这里需要注意 一定是直接用nginx命令启动， 不要用systemctl启动nginx不然会有权限问题

##### **10. 通过配置文件启动uwsgi**

新建uwsgi.ini 配置文件， 内容如下：

# mysite\_uwsgi.ini file

[uwsgi]

# Django-related settings

# the base directory (full path)

chdir = /home/bobby/Projects/MxOnline

# Django's wsgi file

module = MxOnline.wsgi

# the virtualenv (full path)

# process-related settings

# master

master = true

# maximum number of worker processes

processes = 10

# the socket (use the full path to be safe

socket = 127.0.0.1:8000

# ... with appropriate permissions - may be needed

# chmod-socket = 664

# clear environment on exit

vacuum = true

virtualenv = /home/bobby/.virtualenvs/mxonline

logto = /tmp/mylog.log

注：

chdir： 表示需要操作的目录，也就是项目的目录

module： wsgi文件的路径

processes： 进程数

virtualenv：虚拟环境的目录

workon mxonline

uwsgi -i 你的目录/Mxonline/conf/uwsgi.ini &

##### **访问**

http://你的ip地址/

# **ImportError: No module named zlib**

**bash部分**

yum install zlib zlib-devel openssl

**python部分**

cd Python2.7.11

./configure --prefix=/usr/local/python27

make

make install

**验证**

>>> import zlib>>> >>>

# **代码：**

mkvirtualenv --python=/usr/local/bin/python3.6 mxshop

Deactivate

pip install -r requirements.txt