

学生云服务器使用记录

想法与步骤设计

基本材料列举

- 1. ubuntu系统（腾讯云）
- 2. Anaconda 进行Python版本控制
- 3. docker 使用docker来创建管理维护web服务器

| vue-nginx 前端管理服务器 | | Apache-Django 后台管理服务器 | | mysql-redis 数据库管理服务器 | |
|-------------------|--------|-----------------------|--------|----------------------|-------|
| 前端框架 | 静态页面处理 | 动态部署 | 页面生成 | 快速缓存 | 主存 |
| vue | Nginx | Apache | Django | redis | mysql |

- daniel-web 项目管理服务器
- mysql-web 前期数据库服务器
- redis-web 生产环境数据库服务器

项目搭建基本框架步骤(Docker)

- 1. 搭建项目基本环境: 获取基本镜像（httpd）（mysql）（redis）
 - 配置基本工具
 - daniel-web(httpd) : python3.6.8 vim gcc
 - 制作v1镜像文件
- 2. 搭建并运行HelloWorld
- 3.

基本操属性（个人使用 mac iTerm2）

服务器属性

| 云平台 | 内核 | 内核版本 | 操作系统版本 | 版本号 |
|-----|-----------|--------------------------|--------|---------|
| 腾讯云 | GNU/Linux | 4.15.0-54-generic x86_64 | Ubuntu | 18.04.1 |

登录服务器

```
ssh <username>@<public_IP>
<初始密码>
```

基本个人环境配置

服务器用户配置

| 配置内容 | 执行命令 | 执行操作内容 | 备注事项 |
|------|-------------------------------|---|---------------|
| 用户密码 | passwd [username] | <用户原密码> <新密码> <重复输入> | 密码复杂度要求 |
| 主机名 | sudo vim /etc/cloud/cloud.cfg | preservehostname: false => preservehostname: true | Ubuntu18.04 |
| | sudo vim /etc/hostname | <oldhostname> => <newhostname> | |
| | sudo vim /etc/hosts | <oldhostname> => <newhostname> | |
| | sudo reboot | | 可以执行exit,重新连接 |

服务器编辑环境配置

| 配置内容 | 执行命令 | 执行操作内容 | 备注事项 |
|------|--|--------------------------|-----------------------|
| vim | apt-get install vim | <确认> | 如无法下载, apt-get update |
| | wget http://files.cnblogs.com/ma6174/vimrc.zip | | ma6174的主题 |
| | unzip vimrc.zip | | 使用ls -a进行查看.vimrc文件 |
| | vim ~/.vimrc | 打开syntax on 修改新标题文件补全 | 具体配置根据个人情况进行配置 |

开发环境配置

本机配置

| 配置内容 | 执行命令 | 执行操作内容 | 备注事项 |
|----------|--|--------------------------|-------------------|
| Anaconda | wget https://repo.anaconda.com/archive/Anaconda3-2018.12-Linux-x86_64.sh | | Anaconda3 Linux版本 |
| | sudo bash Anaconda3-2018.12-Linux-x86_64.sh | <确认> <安装路径选择> <配件> | |
| | source ~/.bashrc | | 将conda加载入系统路径 |
| Django | pip install django | | 安装django |
| ssl | sudo apt-get install openssl | | |
| | sudo apt-get install libssl-dev | | |
| Docker | curl -fsSL https://get.docker.com sh | | CentOS与Ubuntu通用 |
| sysstat | sudo apt-get install sysstat | | 系统状态工具包 |

docneaten 项目内部环境配置

Anaconda 虚拟环境下配置

| 配置内容 | 执行命令 | 路径 | 备注事项 |
|---------|--|-------|------|
| python3 | conda create -n docneaten python=3.6.8 | | |
| pip | pip install django | | |
| django | django-admin startproject docneaten | /home | |

Docker配置

| 配置内容 | 执行命令 | 执行操作内容 | 备注事项 |
|--------|---|---|--|
| Apache | sudo docker run -d --name daniel-web -p 8080:80 -v /home/docneaten:/home/docneaten/ httpd | 创建apache容器 | -d 在后台运行 |
| Mysql | sudo docker run --name mysql-web -p 3307:3306 -v /home/mysqldata:/home/mysqldata -e MYSQLROOTPASSWORD=6940588666035 -d mysql/mysql-server:5.7 | 创建mysql容器 | |
| redis | docker run -d --name redis-web -p 6379:6379 redis --requirepass "6940588666035" | 创建redis容器 | |
| Django | sudo docker pull library/django:1.10.4-python3 | | |
| ssl | docker cp <ssl_files_path> <container_name>:<apache2_path>/cert/ | | 将ssl的cert文件放到apache目录下 |
| | sudo docker exec -it daniel-web /bin/bash | | 进入container |
| | vim /user/local/apache2/conf/httpd.conf | /ssl 去掉注释(0 x) | 添加ssl模块(确认LoadModule */mod_ssl.so) |
| | vim /user/local/apache2/conf/extra/httpd-ssl.conf | <VirtualHost 0.0.0.0:443> DocumentRoot "/var/www/html" ServerName www.domain.com #填写证书名称 ServerAdmin root@domain.com SSLEngine on #启用 SSL 功能 SSLCertificateFile /etc/httpd/ssl/2_www.domain.com.crt #证书文件的路径 SSLCertificateKeyFile /etc/httpd/ssl/3_www.domain.com.key #私钥文件的路径 SSLCertificateChainFile /etc/httpd/ssl/1_root_bundle.crt #证书链文件的路径 </VirtualHost> | |
| | vim /user/local/apache2/conf/httpd.conf | /rew 去掉注释(0 x) | http跳转https模块(确认LoadModule */mod_rewrite.so) |
| | | <Directory "/var/www/html"> # 新增 RewriteEngine on | |

| | | | |
|--------------------|--|---|-----------|
| | | <div>RewriteCond %{SERVERPORT} ! 443\$ RewriteRule .*\$ https://% {SERVERNAME}%{REQUEST_URI} [L,R] </Directory></div> | |
| | apachectl restart | | 重启服务? ? ? |
| docker- compose | sudo curl -L "https://github.com/docker/compose/releases/download/1.24.1/docker- compose-\$(uname -s)-\$(uname -m)" -o /usr/local/bin/docker-compose | | |
| | chmod +x /usr/local/bin/docker-compose | | |

daniel-web虚拟机配置<httpd:latest>

| 配置内容 | 执行命令 | 执行操作内容 | 备注事项 |
|-------------|---|--|----------------------------------|
| apt-get | apt-get update | apt-get -y upgrade | |
| | apt-get install wget | | 安装wget |
| | apt-get install vim | mv /home/.../.vimrc ~/ | 从服务器拷贝.vimrc文件 |
| | apt-get install gcc make | | 安装python的前导包 |
| | apt-get install build-essential libncursesw5-dev libssl-dev libgdbm-dev libc6-dev libsqlite3-dev tk-dev | | 安装python的前导包 |
| | apt-get install libreadline-dev | | 解决在python3命令行交互模式下上下左右键，屏幕上会打印字符 |
| | apt-get install libapache2-mod-wsgi-py3 | | 实现apache与django的连接 |
| python3 | tar -zxvf 4.5.15.tar.gz | | |
| | ./configure && make && make install | | |
| | ln -s /usr/local/bin/python3 /usr/local/bin/python | | 制作软连接 |
| | ln -s /usr/local/bin/pip3 /usr/local/bin/pip | | 制作软连接 |
| pip | pip install --upgrade pip | | 更新pip |
| | pip install django | | |
| | pip install pymysql | | |
| mod_wsgi.so | 确保存在该文件 | ls /usr/local/apache2/modules/ grep wsgi.so | 没有需要下载 |
| | vim /usr/local/apache2/conf/httpd.conf | (200G o) LoadModule wsgimodule modules/modwsgi.so | |
| | vim /home/docneaten/docneaten/wsgi.py | import sys sys.path.append('/usr/local/lib/python3.6/site-packages') curPath = os.path.abspath(os.path.dirname(file)) rootPath = os.path.split(curPath)[0] sys.path.append(rootPath) | 导入django与该项目的路径 |
| | vim /home/docneaten/docneaten/setting.py | ALLOWED_HOSTS = ['*'] | 修改并允许所有主机进行访问 |

```

llvim /etc/apache2/apache2.conf(G o) ServerName localhost:80
WSGIPythonPath /var/www/botmail
WSGIPythonHome <pythonpath>ll lcd mods-enabled/ll vim wsgimodule.loadlLoadModule wsgimodule /usr/lib/apache2/modules/modwsgi.sol

```

django-web虚拟机配置<django-web_image:v2>

| 配置内容 | 执行命令 | 执行操作内容 | 备注事项 |
|--------|--|--|----------------|
| curl | apt-get install curl | | |
| git | apt-get install git | | |
| xz | 官网下载 | | |
| | tar -jxvf xz.tar.bz2 | | |
| | ./config --prefix=/opt/gnu/xz | 生成Makefile | |
| | make | | |
| | make install | | |
| | vim ~/.bashrc | export PATH=/opt/gnu/xz/bin/:\$PATH\$ | 导入路径 |
| pyenv | curl -L https://github.com/pyenv/pyenv-installer/raw/master/bin/pyenv-installer bash | | |
| | curl -O https://www.python.org/ftp/python/3.6.4/Python-3.6.4.tar.xz | | |
| | mkdir ~/.pyenv/cache | | |
| | mv Python-3.6.4.tar.xz ~/.pyenv/cache/ | | |
| | vim ~/.pyenv/plugins/python-build/share/python-build/3.6.4 | if has_tar_xz_support; then install_package "Python-3.6.4" "/root/.pyenv/cache/Python-3.6.4.tar.xz" ldflagsdirs standard verifypy36 copypythongdb ensurepip else ... | 注意安装依赖 |
| | pyenv install 3.6.4 | | |
| pip | pip install --upgrade pip | 更新pip版本 | |
| django | ./pip install django==2.1.7 | | /usr/local/bin |
| mysql | pip install pymysql | | |
| | vim __init__.py | import pymysql pymysql.install_as_MySQLdb() | |
| | vim ~/.pyenv/versions/3.6.4/lib/python3.6/site-packages/django/db/backends/mysql/base.py | #if version < (1, 3, 13): # raise ImproperlyConfigured('mysqlclient 1.3.13 or newer is required; you have %s.' % Database.__version__) | |
| | vim ~/.pyenv/versions/3.6.4/lib/python3.6/site-packages/django/db/backends/mysql/operations.py | decode -> encode | |
| django | python manage.py makemigrations | | |
| | python manage.py migrate | | |
| | python manage.py runserver | | |

系统命令

系统控制

| 命令 | 参数 | 参数意义 | 命令意义 |
|-----------|--------------------|------|------------------|
| reboot | | | 重启服务器 |
| htop | | | 查看系统情况 |
| systemctl | start docker | | 开启docker服务 |
| | status docker | | 查看状态 |
| apachectl | start restart stop | | 开启 重启 停止Apache服务 |

网络命令

| 命令 | 参数 | 参数意义 | 命令意义 |
|------|---------------|--------|--------|
| curl | icanhazip.com | 访问特定网址 | 获取公网IP |

Docker命令

| 命令 | 参数 | 参数意义 | 命令意义 |
|--|------------------------------------|---------|--------------------|
| docker | --version | | 查看docker版本 |
| run <image_name> | -dit -d | 在后台运行 | 从镜像启动容器 |
| | --name <container_name> | 命名容器 | |
| | -p <server_port>: <container_port> | 定义端口映射 | |
| | -v <server_path>: <container_path> | 定义挂载映射 | |
| search <image_name> | | | 查找镜像文件 |
| pull <image_name> | | | 拉取镜像文件 |
| rename <container_name> <new_container_name> | | | 容器重命名 |
| ps | -a | 查看所有状态 | 查看容器基本情况(默认为运行的容器) |
| start restart stop rm <container> | | | 启动 重启 停止 删除 容器 |
| rmi <image_name> | | 删除镜像 | |
| exec <container_name> <order> | -it | 以终端方式运行 | 在container中运行命令 |
| commit <container> <new_image> | -m "<commit_msg>" | 提交标签 | 从容器创建镜像 |
| | -a "<author>" | 标记创建者 | |

mysql命令

| 级别 | 类别 | 命令 | 辅助命令 | 目的 |
|-----|------|---|-------------------|----------------------------------|
| 系统 | 登录 | mysql -h<IP> --port=<server_port> -u<user_name> -p | | 登录数据库 |
| 权限 | 用户管理 | grant all privileges on *.* to '<user_name>'@'<IP>' identified by '<password>' with grant option; | flush privileges; | 添加<user_name>允许<IP>以<password>登录 |
| | | drop user zhangsan@'[% <IP>]'; | | 删除用户 |
| | 数据库 | show databases; | | 打印数据库 |
| 数据库 | 表 | show tables; | | 打印表格信息 |
| 表 | 查询 | show columns from <table_name>; | | 打印表头信息 |
| | | select <key_name> <alias>, <key_name> from <table_name> where <condition> | | 打印表内信息 |

pyenv命令

| 命令 | 意义 |
|--------------------|----------|
| pyenv global 3.6.4 | 切换当前系统环境 |

conda命令

| 命令 | 意义 |
|---|------------|
| conda create -n <env_name> python=<version_num> | python版本控制 |
| conda activate autosortsys | 激活环境 |
| conda remove -n <env_name> --all | 删除环境 |

django命令

| 命令 | 意义 | 路径要求 |
|--|--------------|-------|
| django-admin startproject <pro_name> | 创建一个django工程 | |
| python manage.py runserver <host_ip>:<host_port> | 开启简易web程序 | 工程目录下 |

其他问题

| 问题 | 现象 | 解决方法 |
|--------------|----------|--|
| 系统Python版本问题 | 不支持高版本操作 | 安装高版本并配置系统软连接 |
| | 步骤: | cd Python-3.6.4/ ./configure --prefix=<syspath=\usr\local\Python-3.6.4> make && make install ln -s <newversionfile> <oldversion_file=/usr/local/bin/> |

