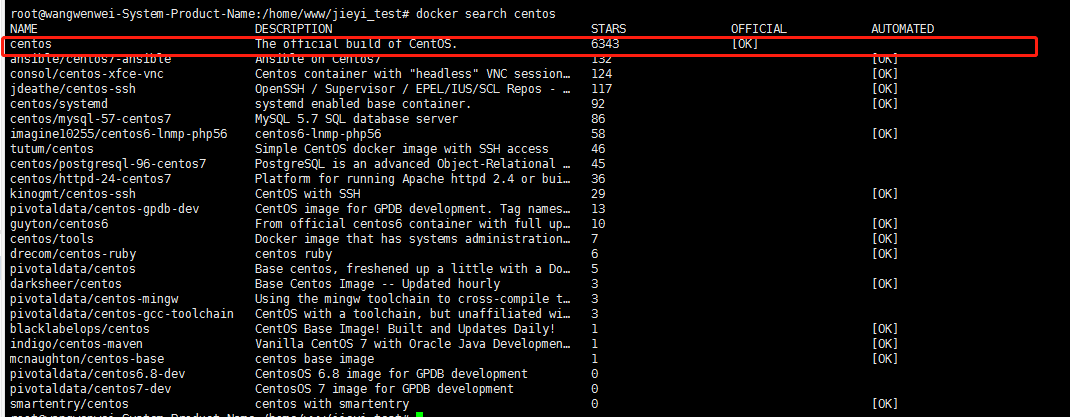
# 安装并启动docker服务（略）

# 从远程仓库拉取 centos 系统镜像，并创建容器

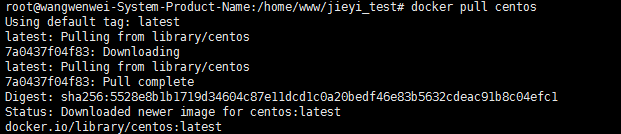
查询 centos 相关的镜像

docker search centos



下载镜像到本地,如我这里是：

docker pull centos



查看本地下载的镜像

docker images



执行以下命令，创建容器

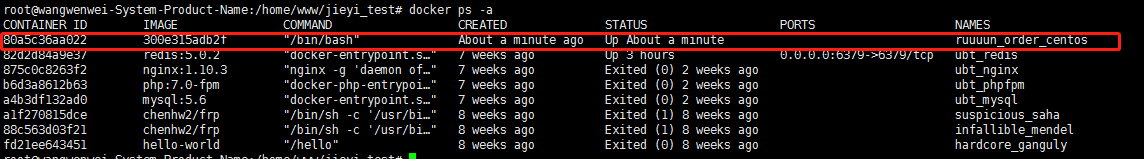
格式：docker  run  -dit  --name=容器名  镜像 id  /bin/bash

如我这里为了奔跑科技创建商城环境的容器：

docker run -dit --name=ruuuun\_order\_centos 300e315adb2f /bin/bash

查看当前所有的容器，确认刚刚的容器已经创建完毕

docker ps –a



进入容器

格式：docker  exec  -it  容器名  /bin/bash

我这里的命令如下：

docker exec -it ruuuun\_order\_centos /bin/bash

想要退出容器，执行以下命令：

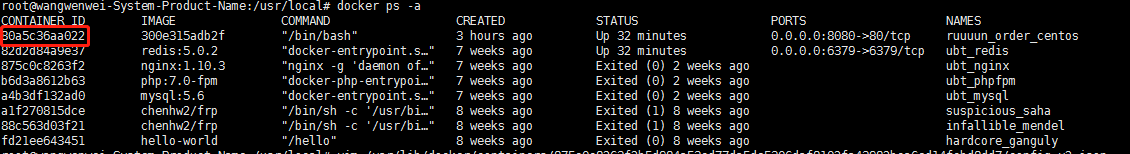
exit

# Centos容器基本配置

## 3.1 统一挂载目录

所有要修改容器配置的操作，先要记下需要操作的容器id,然后关闭docker服务，配置完了重启docker服务即可。

执行docker ps –a 记下容器id,记住前缀80a即可



执行以下命令，关闭docker服务

systemctl stop docker

进入指定容器配置目录，

格式：cd /var/lib/docker/containers/{container\_id}

如我刚刚要配置的是80a

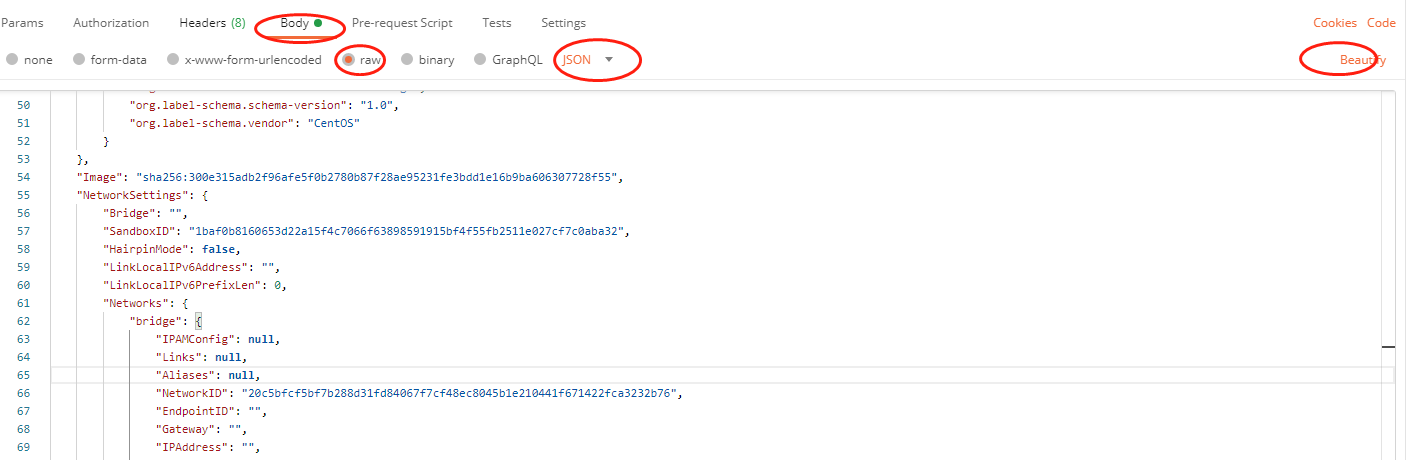
cd /var/lib/docker/containers/80a5c36aa02216b820c0b84bdf24c38353d0f1b4e9538b7d51006340a50cb810/

vim config.v2.json 编辑文件

这里说明以下json文件修改，先复制文件内容



打开json文件格式化工具，这里用postman演示





拷贝以下模板，进行挂载修改（我将宿机/home目录直接挂到容器/home,实现同步）：

    "MountPoints": {

        "/home": {

            "Source": "/home",

            "Destination": "/home",

            "RW": **true**,

            "Name": "",

            "Driver": "",

            "Type": "bind",

            "Propagation": "rprivate",

            "Spec": {

                "Type": "bind",

                "Source": "/home",

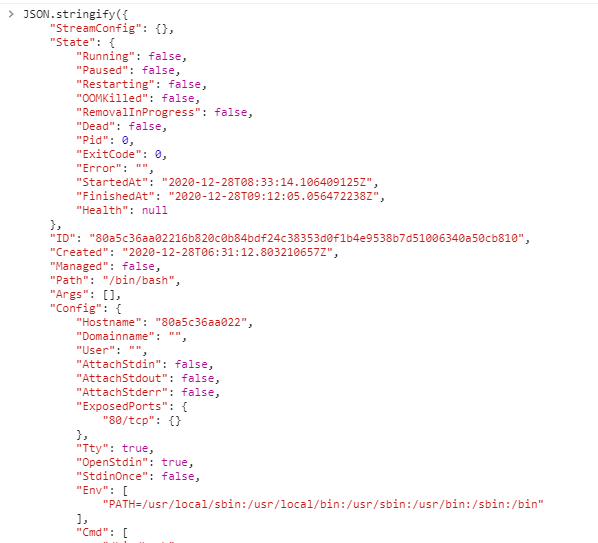
                "Target": "/home"

            }

        }

    },

修改完了，复制整个JSON内容，打开浏览器，按F12，在console选项卡中，执行JS语句





这样就得到了压缩的JSON文本数据，拷贝到服务器config.v2.json中保存即可。

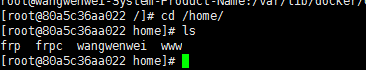
重启docker服务：

systemctl restart docker

执行docker start ruuuun\_order\_centos 启动容器

执行 docker exec –it ruuuun\_order\_centos /bin/bash 进入容器

cd /home 检测目录是否同步挂载了



可以看到，容器中的目录成功挂载了。

## 3.2 容器中修改为阿里yum源

进入容器（进入方法前面提到过），查看当前Centos版本：

cat /etc/redhat-release



我这个是Centos8

执行以下命令，备份系统自带yum源配置文件/etc/yum.repos.d/CentOS-Base.repo

cp /etc/yum.repos.d/ CentOS-Base.repo

/etc/yum.repos.d/CentOS-Linux-BaseOS.repo.backup

注意，有些人的系统配置文件命名是CentOS-Base.repo 没有-Linux

如果你的容器没有安装wget 请先执行yum install wget.

保证有wget之后，执行以下命令，删除当前CentOS-Base.repo

rm /etc/yum.repos.d/CentOS-Linux-BaseOS.repo

下载ailiyun的yum源配置文件到/etc/yum.repos.d/，根据你的版本选择命令：

CentOS8

wget -O /etc/yum.repos.d/CentOS-Linux-BaseOS.repo http://mirrors.aliyun.com/repo/Centos-8.repo

CentOS7

wget -O /etc/yum.repos.d/CentOS-Linux-BaseOS.repo http://mirrors.aliyun.com/repo/Centos-7.repo

CentOS6

wget -O /etc/yum.repos.d/CentOS-Linux-BaseOS.repo http://mirrors.aliyun.com/repo/Centos-6.repo

CentOS5

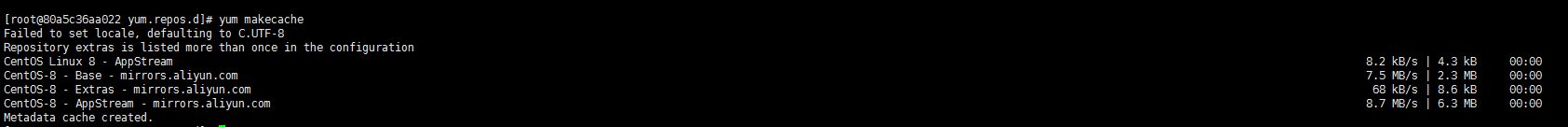
wget –O /etc/yum.repos.d/CentOS-Linux-BaseOS.repo http://mirrors.aliyun.com/repo/Centos-5.repo

如果你的容器没有安装wget 请先执行yum install wget



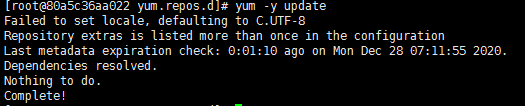
执行以下命令，生成yum缓存

yum makecache



执行以下命令，更新yum

yum -y update



# 在容器中安装对应的生产环境

我这里，演示安装

nginx:1.10.3

php:5.6+7.2

mysql:5.6

redis:5.0.2

## 4.1 安装niginx

安装依赖包，使用yum工具安装

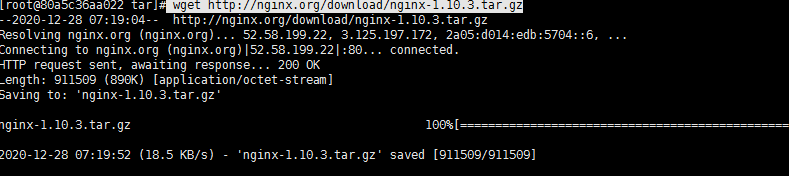
yum -y install gcc make zlib-devel pcre pcre-devel openssl-devel

在/home目录，创建文件夹tar,用来存放依赖包

mkdir /home/tar

执行以下命令，下载nginx

wget <http://nginx.org/download/nginx-1.10.3.tar.gz>



解压这个包

tar -xvf nginx-1.10.3.tar.gz

进入解压后的目录

cd nginx-1.10.3

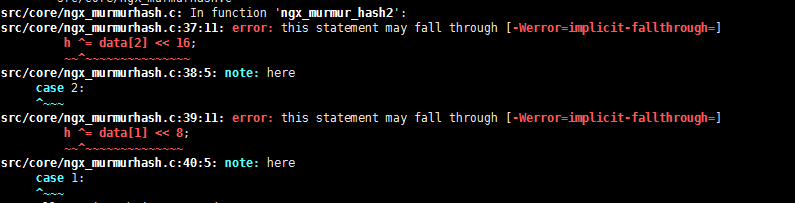
执行配置命令

./configure

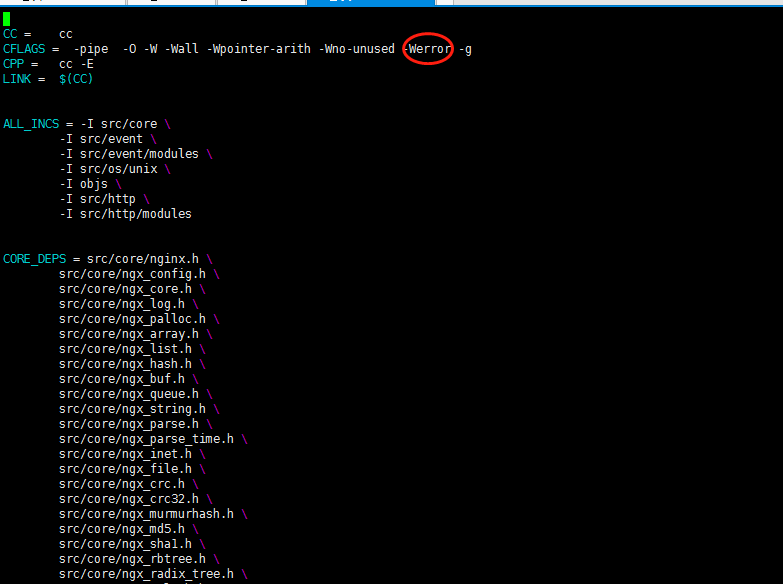
接下来编译文件

make

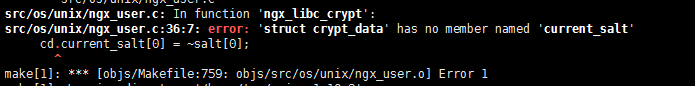
一般在这里会出现问题，下面给出问题截图和解决方法



vim objs/Makefile

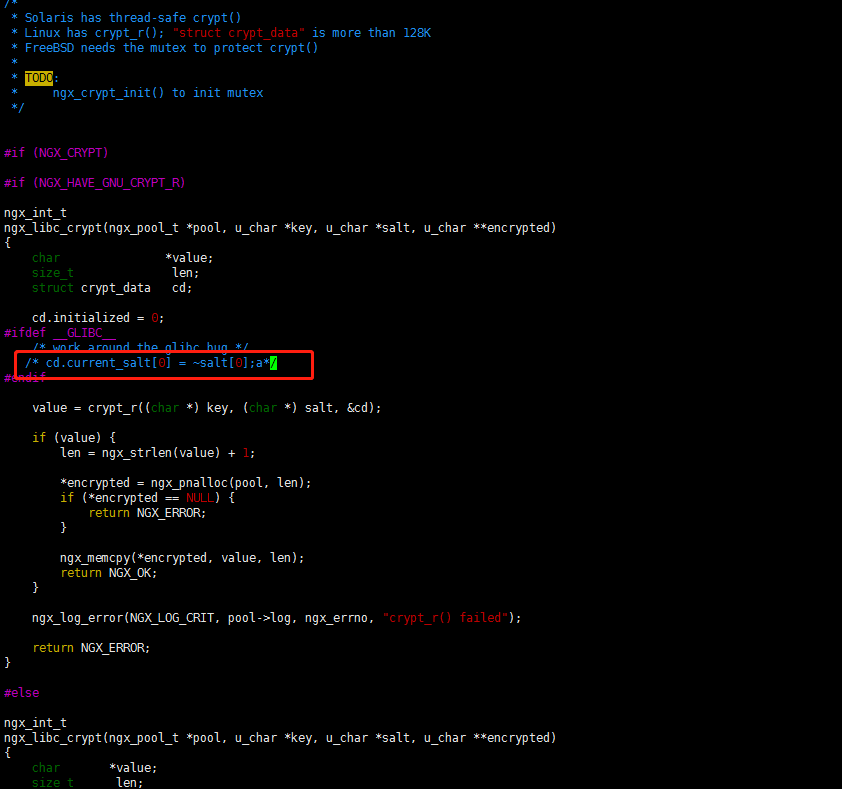


把这行内容中的 “-Werror”去掉,保存退出，再次make,又有错误

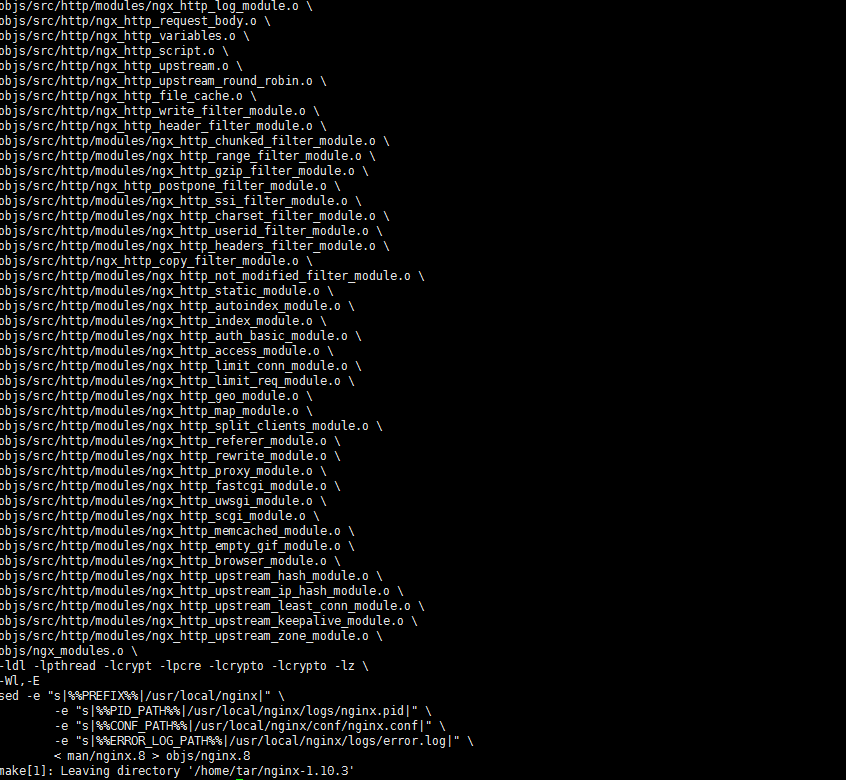


进入解压后的nginx-1.10.3，编辑以下文件

vim /src/os/unix/ngx\_user.c，将以下行注释掉



再次make



make install执行安装

启动nginx服务器

sudo /usr/local/nginx/sbin/nginx

安装net-tools(如果有，请忽略)

yum install net-tools

查看端口80

netstat -tunlp | grep 80

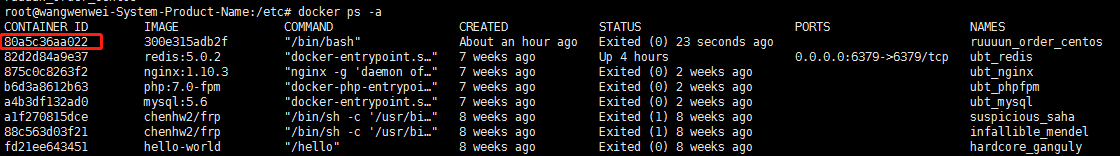


看到nginx已经启动成功

将容器的80端口映射到服务器的8080端口

执行exit退出容器，

执行docker ps –a 查看该容器id,记住是80a开头



执行以下命令，先将docker关闭

systemctl stop docker

进入该容器配置目录

cd /var/lib/docker/containers/80a5c36aa02216b820c0b84bdf24c38353d0f1b4e9538b7d51006340a50cb810/

修改hostconfig.json

vim hostconfig.json

修改端口映射配置

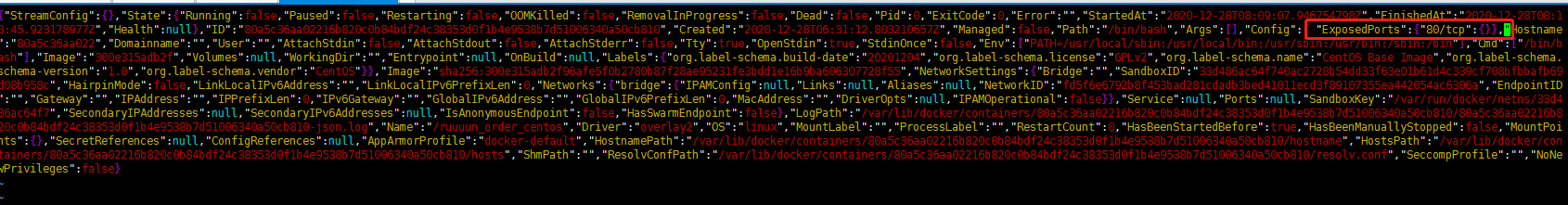
"PortBindings":{"80/tcp":[{"HostIp":"","HostPort":"8080"}]}

保存退出，

由于端口是新添加的，需要修改config.v2.json，添加要暴露的端口

vim /var/lib/docker/containers/80a5c36aa02216b820c0b84bdf24c38353d0f1b4e9538b7d51006340a50cb810/config.v2.json

将 "ExposedPorts":{"80/tcp":{}} 加入Config中



然后执行以下命令，

重启docker

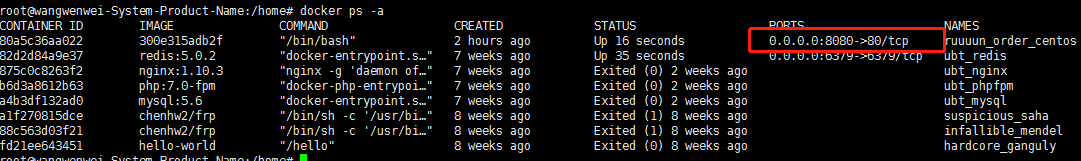
systemctl restart docker

重启容器

docker restart 容器ID

如我的是：docker restart 80a

执行docker ps –a 看到端口映射情况



进入容器，启动niginx

docker exec -it ruuuun\_order\_centos /bin/bash

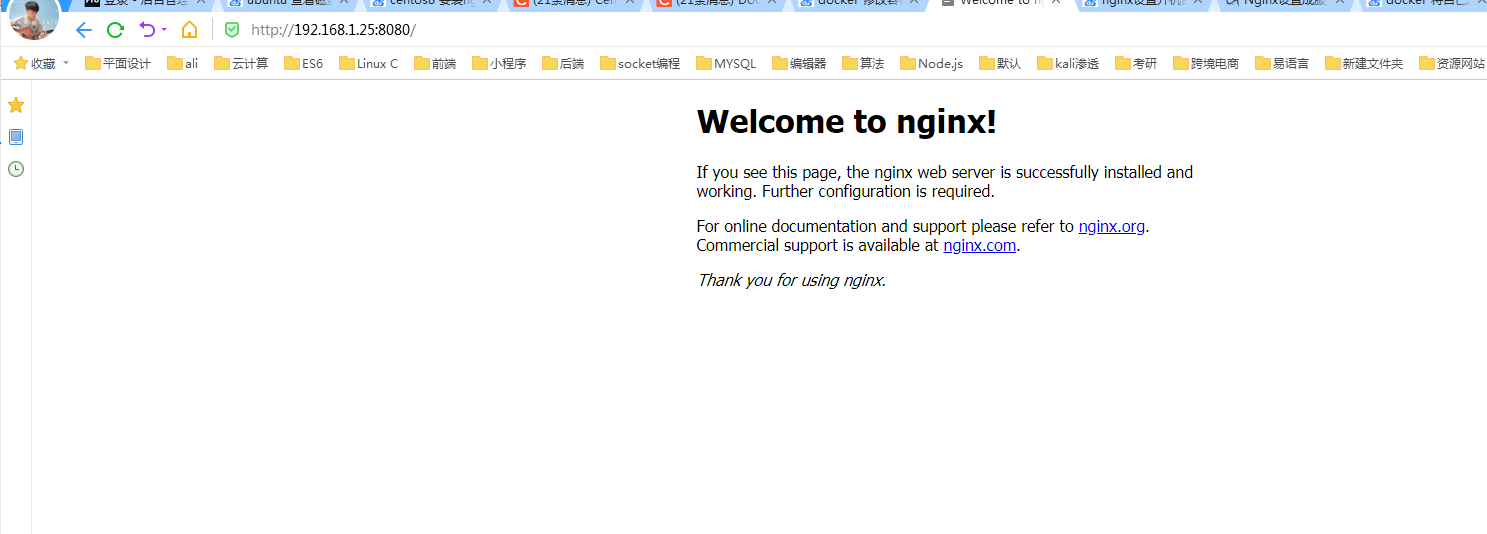
/usr/local/nginx/sbin/nginx

用浏览器打开

http://虚拟机IP:8080/

如我的是：

<http://192.168.1.25:8080/>



成功搭建。

## 4.2 安装php