

## 实验要求：

搭建 Apache Web 服务器实现三个页面跳转，搭建 ftp 服务器，搭建 dns 服务器

## 实验环境：

Docker 基础镜像 nickistre/centos-lamp，配置如下

1. CentOS release 6.6
2. Apache/2.2.15 (CentOS)
3. PHP 5.3.3

Docker 网卡设置成与宿主机桥接，配置命令如下

```
docker network create --subnet=172.18.0.0/16 mynetwork
```

```
root@ubuntu:~/Desktop/networkbuild# docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
a52d60c0b7aa        bridge              bridge              local
f7c3dc056a68        host                host                local
794e996893b8        mynetwork           bridge              local
c3770166dca7        none                null                local
root@ubuntu:~/Desktop/networkbuild#
```

为了启动虚拟机容器配置的 ip 为 172.18.0.101/16，启动命令如下

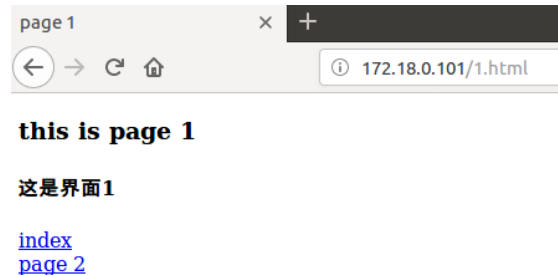
```
docker run -itd --name build2 --network mynetwork --ip 172.18.0.101 sijidou:networkbuild
```

### 1.配置 web 服务器

因为继承的基础镜像就是 lamp，已经配置好了 apache，所以我直接把 web 的 3 个页面 copy 进容器，目录情况如下

```
[root@8263306112e5 /]# cd /var/www/html/
[root@8263306112e5 html]# ls
1.html 2.html index.html phpinfo.php
[root@8263306112e5 html]#
```

在宿主机访问页面，能够直接跳转



### 2.配置 bind 的 dns 服务器

首先安装 bind 使用命令

```
yum install -y bind
```

接着对/etc/named.conf 进行配置，结果如下



```
@8263306112e5:/etc
$TTL 86400
@ IN SOA test.edu.cn. root.localhost. (
2011091200 ; serial
3H ; refresh
1H ; retry
1W ; expire
0) ; minimum
@ IN NS dns
www IN A 172.18.0.101
dns IN A 172.18.0.101
ftp IN A 172.18.0.101
```

在测试阶段，开启服务后，修改下本地 dns 解析的服务器地址，在/etc/resolv.conf 文件中，改为

```
@8263306112e5:/etc
search test.edu.cn
nameserver 172.18.0.101
#nameserver 127.0.0.11
options ndots:0
~
~
```

能够正常解析结果

```
[root@8263306112e5 etc]# service named start
Starting named: [ OK ]
[root@8263306112e5 etc]# dig www.test.edu.cn

; <<>> DiG 9.8.2rc1-RedHat-9.8.2-0.68.rc1.el6_10.1 <<>> www.test.edu.cn
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 38512
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;www.test.edu.cn.                IN      A

; ANSWER SECTION:
www.test.edu.cn.                86400   IN      A      172.18.0.101

; AUTHORITY SECTION:
test.edu.cn.                    86400   IN      NS      dns.test.edu.cn.

; ADDITIONAL SECTION:
dns.test.edu.cn.                86400   IN      A      172.18.0.101

;; Query time: 0 msec
;; SERVER: 172.18.0.101#53(172.18.0.101)
;; WHEN: Fri May 10 04:01:26 2019
;; MSG SIZE rcvd: 83

[root@8263306112e5 etc]#
```

### 3.ftp 服务器的配置

ftp 服务器首先下载依赖

```
yum install -y vsftpd
```

下载好了后其实就可以正常进行连接了，这里创建个账号和对应文件夹进行 ftp 连接使用

```
useradd -d /var/ftp/ftp ftptest
```

给该账号设置密码

```
[root@8263306112e5 etc]# passwd ftptest
Changing password for user ftptest.
New password:
BAD PASSWORD: it is based on a dictionary word
BAD PASSWORD: is too simple
Retype new password:
passwd: all authentication tokens updated successfully.
[root@8263306112e5 etc]#
```

重启服务器后，用宿主机去连接

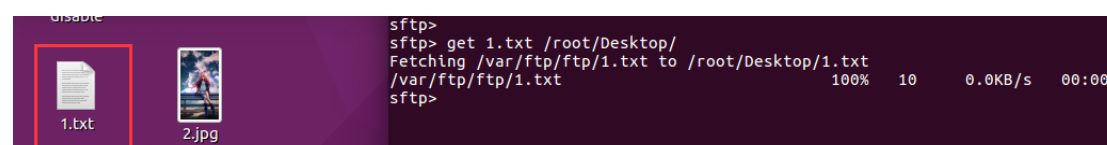
在客户端连接请求后的结果

```
root@ubuntu:~/Desktop/networkbuild# sftp ftptest@172.18.0.101
ftptest@172.18.0.101's password:
Connected to 172.18.0.101.
sftp> ls
1.txt
sftp>
```

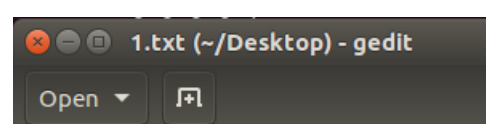
在服务器上对应文件位置

```
[root@8263306112e5 etc]# cd /var/ftp/
[root@8263306112e5 ftp]# ls
ftp pub
[root@8263306112e5 ftp]# cd ftp/
[root@8263306112e5 ftp]# ls
1.txt
[root@8263306112e5 ftp]# pwd
/var/ftp/ftp
[root@8263306112e5 ftp]#
```

把 1.txt 下载下来



```
sftp> get 1.txt /root/Desktop/
Fetching /var/ftp/ftp/1.txt to /root/Desktop/1.txt
/var/ftp/ftp/1.txt 100% 10 0.0KB/s 00:00
sftp>
```



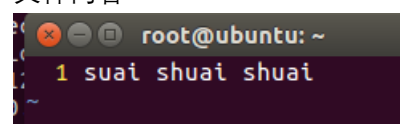
hello ftp

上传个 2.txt

在宿主机的位置

```
root@ubuntu:~# ls
2.txt Documents Music Public Videos
Desktop Downloads Pictures Templates
root@ubuntu:~# vim 2.txt
root@ubuntu:~# vim 2.txt
root@ubuntu:~#
```

文件内容



```
root@ubuntu:~#
1 suai shuai shuai
~
~
```

上传命令如下

```
sftp> put /root/2.txt
Uploading /root/2.txt to /var/ftp/ftp/2.txt
/root/2.txt 100% 17 0.0KB/s 00:00
sftp>
```

在服务器上打开目标，成功的上传了

```
[root@8263306112e5 ftp]# ls
1.txt 2.txt
[root@8263306112e5 ftp]# cat 2.txt
suai shuai shuai
[root@8263306112e5 ftp]#
```

最后 Docker 一键部署文件如下

```
root@ubuntu: ~/Desktop/networkbuild
1 FROM nickistre/centos-lamp
2
3 COPY yum.repos.d/ /etc/yum.repos.d/
4
5 COPY src/ /var/www/html/
6
7 RUN rpm --rebuild \
8     && yum install -y bind \
9     && sed -i 's/{ 127.0.0.1; }/{ any; };/g' /etc/named.conf \
10    && sed -i 's/{ ::1; }/{ any; };/g' /etc/named.conf \
11    && sed -i 's/{ localhost; }/{ any; };/g' /etc/named.conf \
12    && echo -e '\nzone "test.edu.cn" IN {\n\ttype master;\n\tfile "ip4v4.edu.zone";\n\tallow-update
{ none; };;\n};' >> /etc/named.rfc1912.zones
13
14 COPY named/ /var/named/
15
16 RUN service named start \
17    && rpm --rebuild \
18    && yum install -y vsftpd \
19    && useradd -d /var/ftp/ftp ftptest \
20    && echo "hello ftp" > /var/ftp/ftp/1.txt \
21    && chmod 777 /var/ftp/ftp/1.txt \
22    && service vsftpd start
```

项目文件夹下有 named 和 yum.repos.d 和 src 文件夹

分别对应的作用是写入 dns 解析规则，修改 yum 的镜像源为阿里镜像，上传 web 服务器的三个跳转页面文件

```
root@ubuntu:~/Desktop/networkbuild# ls
Dockerfile named README.md src start.sh yum.repos.d
root@ubuntu:~/Desktop/networkbuild# vim Dockerfile
root@ubuntu:~/Desktop/networkbuild#
```

部署方式,进入 networkbuild 这个文件夹下

docker pull nickistre/centos-lamp

docker build -t sijidou:networkbuild. (后面有个点!, 表示 Dockerfile 文件路径是当前目录)

启动容器

./start.sh

进入容器

docker exec -it build2 bash

进去后 shell 就会不一样，因为 ftp 账号设置的时候有设置密码，在 dockerfile 中没有写密码的设置所以要进去手动设置

passwd ftptest

然后会提示输入密码

有时候服务器没有启动，可以进入容器手动启动下

service named start

至于配置过程中没有写防火墙的关闭和开启，因为继承的基础镜像中貌似默认端口全部对外开放

最后的最后，我把源码丢到 github 上了: [https://github.com/SiJiDo/simple\\_network\\_build](https://github.com/SiJiDo/simple_network_build)