**DAY 2**

案例一：部署LNMP环境

1，nginx我们已经安装好了，如果没有，则参照前一天的笔记，安装nginx。

2，安装MariaDB、php、php-fpm

[root@proxy ~]# yum -y install mariadb mariadb-server mariadb-devel

[root@proxy ~]# yum -y install php php-mysql

[root@proxy ~]# cd lnmp\_soft/

[root@proxy lnmp\_soft]# yum -y install php-fpm-5.4.16-42.el7.x86\_64.rpm

3，启动服务，nginx、mariadb、php-fpm

[root@proxy ~]# systemctl start mariadb

[root@proxy ~]# systemctl enable mariadb

[root@proxy ~]# systemctl start php-fpm

[root@proxy ~]# systemctl enable php-fpm

案例二：构建LNMP平台

目标：

1，配置Fast-CGI支持PHP网页

2，创建PHP测试页面，测试使用PHP连接数据库的效果

步骤一：查看php-fpm配置文件（实验中不需要修改该文件）

[root@proxy ~]# vim /etc/php-fpm.d/www.conf

[www]

listen = 127.0.0.1:9000 //PHP端口号

pm.max\_children = 50 //最大进程数量

pm.start\_servers = 5 //最小进程数量

pm.min\_spare\_servers = 5 //最少需要几个空闲着的进程

pm.max\_spare\_servers = 35 //最多允许几个进程处于空闲状态

步骤二：修改nginx配置文件并重启服务

注意：如果是在前一天的实验基础上做的，需要把一些功能代码给注释掉，防止出错时，不知道如何去排错。

[root@proxy ~]# vim /usr/local/nginx/conf/nginx.conf

server {

listen 80;

server\_name www.a.com;

# auth\_basic "shuru:";

# auth\_basic\_user\_file "/usr/local/nginx/pass";

location / {

root html;

index index.php index.html index.htm;

}

error\_page 500 502 503 504 /50x.html;

location = /50x.html {

root html;

}

location ~ \.php$ {

root html;

fastcgi\_pass 127.0.0.1:9000;

fastcgi\_index index.php;

include fastcgi.conf;

}

}

[root@proxy ~]# nginx -s reload

步骤三：创建PHP页面1

[root@proxy ~]# cat lnmp\_soft/php\_scripts/test.php >> /usr/local/nginx/html/test1.php

创建PHP页面2，连接并查询MariaDB数据库

[root@proxy ~]# cat lnmp\_soft/php\_scripts/mysql.php >> /usr/local/nginx/html/test2.php

步骤四：客户端测试，2个都可正常访问

[root@client ~]# firefox http://192.168.4.5/test1.php

[root@client ~]# firefox http://192.168.4.5/test2.php

案例3：地址重写

沿用案例二，通过调整Nginx服务端配置，实现以下目标：

1，所有访问a.html的请求，重定向到b.html;

2，所有访问192.168.4.5的请求重定向至www.tmooc.cn；

3，所有访问192.168.4.5/下面子页面，重定向至www.tmooc.cn/下相同的页面；

4，实现firefox与curl访问相同页面文件，返回不同的内容。

目标一，访问a.html重定向到b.html

修改nginx配置文件，添加一行rewrite

[root@proxy ~]# vim /usr/local/nginx/conf/nginx.conf

server {

listen 80;

server\_name localhost;

location / {

root html;

index index.html index.htm;

rewirte /a.html /b.html;

}

设置首页文件，并重启服务

[root@proxy ~]# echo "BBBB" > /usr/local/nginx/html/b.html

[root@proxy ~]# echo "AAAA" > /usr/local/nginx/html/a.html

[root@proxy ~]# nginx -s reload

客户端测试（地址栏里不会变，但内容会显示b.html的）

[root@client ~]# firefox http://192.168.4.5/a.html

**目标1**，访问a.html重定向到b.html（跳转地址栏）

[root@proxy ~]# vim /usr/local/nginx/conf/nginx.conf

server {

listen 80;

server\_name localhost;

location / {

root html;

index index.html index.htm;

rewrite /a.html /b.html redirect;

}

[root@proxy ~]# nginx -s reload

客户端测试（地址栏也会有跳转变化）

[root@client ~]# firefox http://192.168.4.5/a.html

**目标2**，访问192.168.4.5的请求重定向至www.tmooc.cn

[root@proxy ~]# vim /usr/local/nginx/conf/nginx.conf

server {

listen 80;

server\_name localhost;

rewrite ^/ [http://uc.tmooc.cn/;](http://www.tmooc.cn/;) #新增此行

location / {

root html;

index index.html index.htm;

# rewrite /a.html /b.html redirect; #注释掉此行

}

[root@proxy ~]# nginx -s reload

真机上测试（因为只有真机可以访问tmooc）

[root@room5pc80 ~]# firefox http://192.168.4.5

思考题：为什么这里的rewrite要写在location外面？？

**目标3**，访问192.168.4.5/下面子页面，重定向至www.tmooc.cn/下相同的页面

[root@proxy ~]# vim /usr/local/nginx/conf/nginx.conf

server {

listen 80;

server\_name localhost;

rewrite ^/(.\*)$ http://uc.tmooc.cn/$1;

location / {

root html;

index index.html index.htm;

# rewrite /a.html /b.html redirect;

}

真机上测试

[root@room5pc80 ~]# firefox http://192.168.4.5/login/jumpTologin

**目标4**：实现firefox与curl访问相同页面文件，返回不同的内容

创建网页目录和页面文件

[root@proxy ~]# echo "who are you" > /usr/local/nginx/html/test.html

[root@proxy ~]# mkdir /usr/local/nginx/html/firefox

[root@proxy ~]# echo "i am tom" > /usr/local/nginx/html/firefox/test.html

修改配置文件，并重启服务

[root@proxy ~]# vim /usr/local/nginx/conf/nginx.conf

server {

listen 80;

server\_name localhost;

#rewrite ^/(.\*)$ http://uc.tmooc.cn/$1;

location / {

root html;

index index.html index.htm;

# rewrite /a.html /b.html redirect;

}

#这里，~符号代表正则匹配，\*符号代表不区分大小写

if ($http\_user\_agent ~\* firefox) { #识别客户端firefox浏览器

rewrite ^(.\*)$ /firefox/$1;

}

[root@proxy ~]# nginx -s reload

客户端测试（两个页面的内容不一样，则为正确）

[root@client ~]# firefox http://192.168.4.5/test.html

[root@client ~]# curl http://192.168.4.5/test.html