

Nginx 反向代理设置

一、设置反向代理例子一

需求：电脑访问www.123.com,显示对应服务器Tomcat首页

1、在Windows系统的host文件进行域名配置

host文件加 192.168.198.128 www.123.com

```
# localhost name resolution is handled within DNS itself.
#       127.0.0.1      localhost
#       ::1            localhost
192.168.198.128 www.123.com
```

2、修改nginx 配置文件

备注：由于我这里80端口占用，所以用的90

```
server {
    listen      90;
    server_name 192.168.198.128

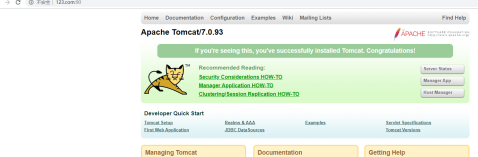
    #charset koi8-r;

    #access_log logs/host.access.log main;

    location / {
        root    html;
        index   index.html index.htm;
        proxy_pass http://127.0.0.1:8090;
    }
}
```

3、客户端访问www.123.com,页面跳转到Tomcat首页

备注：由于我这里使用的90端口，所以www.123.com后面要加90端口，如果是默认的80端口是不需带端口号的



二、反向代理例子二

需求：Nginx根据不同的路径跳转到不同的服务页面中去

1、准备工作

准备两个Tomcat，一个是8090端口，一个是8070端口

修改端口文件server.xml，如下图：

修改Tomcat的端口号：

在默认情况下，tomcat的端口是8080，如果出现8080端口号冲突，用如下方法可以修改Tomcat的端口号：

首先：在Tomcat的解（安装）目录下，有一个conf文件夹，双击进入conf文件夹，在里面找到Server.xml文件，打开该文件。

其次：在文件中找到如下文本：

```
<Connector port="8080" protocol="HTTP/1.1"
    maxThreads="150" connectionTimeout="20000"
    redirectPort="8443" />
```

也有可能像这样的：

```
<Connector port="8080" maxThreads="150" minSpareThreads="25" maxSpareThreads="75" enableLookups="false" redirectPort="8443"
    acceptCount="100" debug="0" connectionTimeout="20000"
    disableShutdown="true" />
```

最后：将port="8080"改为其它的就可以了，如port="8090"等。

保存server.xml文件，重新启动Tomcat服务器，Tomcat就可以使用8090端口了。

注意：有的时候需要两个tomcat，那么就需要修改其中的一个的端口号才能使得两个同时工作。

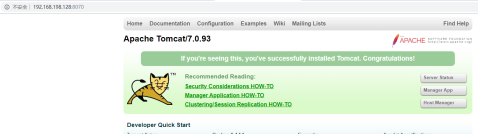
修改了上面的以后，还要修改两处：

(1) 将 <Connector port="8090" enableLookups="false" redirectPort="8443" debug="0" protocol="AJP/1.3" />的8090改为其它的端口。

(2) 继续将<Server port="8005" shutdown="SHUTDOWN" debug="0">的8005改为其它的端口。

经过以上三个修改，应该就可以了。

修改后通过IP：8090或IP：8070，做一下配置是否修改成功的验证，显示如下图，表明配置成功

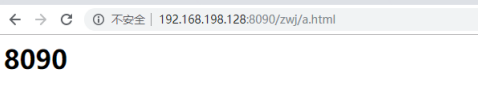


2、创建文件夹和测试页面

在Tomcat的webapps目录下建目录且放入测试页面，如下图：

```
[root@besttest webapps]# mkdir zwj
[root@besttest webapps]# ls
dangdang docs examples host-manager manager ROOT zwj
[root@besttest webapps]# cd zwj
[root@besttest zwj]# rz -E
rz waiting to receive.
[root@besttest zwj]# ls
a.html
```

验证是否可正常访问测试页面，如下图表示配置正确：



3、配置反向代理

配置nginx.conf文件，如下图：

```
#
server {
    listen      90;
    server_name 192.168.198.128;

    location ~ /edu/ {
        proxy_pass http://192.168.198.128:8070;
    }

    location ~ /zwj/ {
        proxy_pass http://192.168.198.128:8090;
    }
}
```

4、开发对外访问的端口号

firewall-cmd --add-port=90/tcp --permanent

5、重启（./nginx -s reload）Nginx，进行验证

访问IP：90/edu/a.html和访问IP： 90/zwj/a.html，如下图

