

负载均衡

定义：

通俗：将负载变的均衡。

负载（请求、工作任务）、均衡（算法，中间件）。

实验

目的：

1. 通过浏览器多次访问一个地址（<http://www.cpf.com/load-balance>）。
2. nginx接受上面的请求，并进行转发。
3. 那么每个请求的响应，是来自于不同的tomcat提供的。（2台tomcat，端口：8081，8082）。

两台tomcat，不同的响应内容：“8081”和“8082”。

步骤：

1. 准备2个tomcat，并做好响应的页面，启动，测试。

```
[root@localhost tomcat8082]# cd apache-tomcat-8.5.78/webapps/
[root@localhost webapps]# ll
total 4
drwxr-x---. 15 root root 4096 Apr  7 08:18 docs
drwxr-x---.  7 root root  99 Apr  7 08:18 examples
drwxr-x---.  6 root root  79 Apr  7 08:18 host-manager
drwxr-x---.  6 root root 114 Apr  7 08:18 manager
drwxr-x---.  3 root root 241 Apr  7 08:25 ROOT
drwxr-xr-x.  2 root root  24 Apr  7 08:57 shanghai
[root@localhost webapps]# mkdir load-balance
[root@localhost webapps]# cd load-balance/
[root@localhost load-balance]# ll
total 0
[root@localhost load-balance]# vi index.html
[root@localhost load-balance]# ll
total 4
-rw-r--r--. 1 root root 5 Apr  8 07:16 index.html
[root@localhost load-balance]# cd .
[root@localhost load-balance]# cd ..
[root@localhost webapps]# ll
total 4
```

```
drwxr-x---. 15 root root 4096 Apr  7 08:18 docs
drwxr-x---.  7 root root  99 Apr  7 08:18 examples
drwxr-x---.  6 root root  79 Apr  7 08:18 host-manager
drwxr-xr-x.  2 root root  24 Apr  8 07:16 load-balance
drwxr-x---.  6 root root 114 Apr  7 08:18 manager
drwxr-x---.  3 root root 241 Apr  7 08:25 ROOT
drwxr-xr-x.  2 root root  24 Apr  7 08:57 shanghai
[root@localhost webapps]# cat load-balance/index.html
8082
[root@localhost webapps]#
```

剩下的tomcat 同理。

启动2个tomcat。

```
[root@localhost bin]# ./startup.sh
Using CATALINA_BASE:  /tomcat/tomcat8081/apache-tomcat-8.5.78
Using CATALINA_HOME:  /tomcat/tomcat8081/apache-tomcat-8.5.78
Using CATALINA_TMPDIR: /tomcat/tomcat8081/apache-tomcat-8.5.78/temp
Using JRE_HOME:       /usr
Using CLASSPATH:      /tomcat/tomcat8081/apache-tomcat-8.5.78/bin/bootstrap.jar:/tomcat/tomcat8081/a
Using CATALINA_OPTS:
Tomcat started.
[root@localhost bin]#
```

测试：

```
[root@localhost bin]# curl localhost:8081/load-balance/index.html
8081
[root@localhost bin]# curl localhost:8082/load-balance/index.html
8082
[root@localhost bin]#
```

至此：2个tomcat启动，并成功返回对应的页面。

2. 配置nginx.conf。

```
[root@localhost nginx]# pwd
/etc/nginx
[root@localhost nginx]# vi nginx.conf
```

```
# server list
upstream myServers {
    server localhost:8081;
    server localhost:8082;
```

```

}

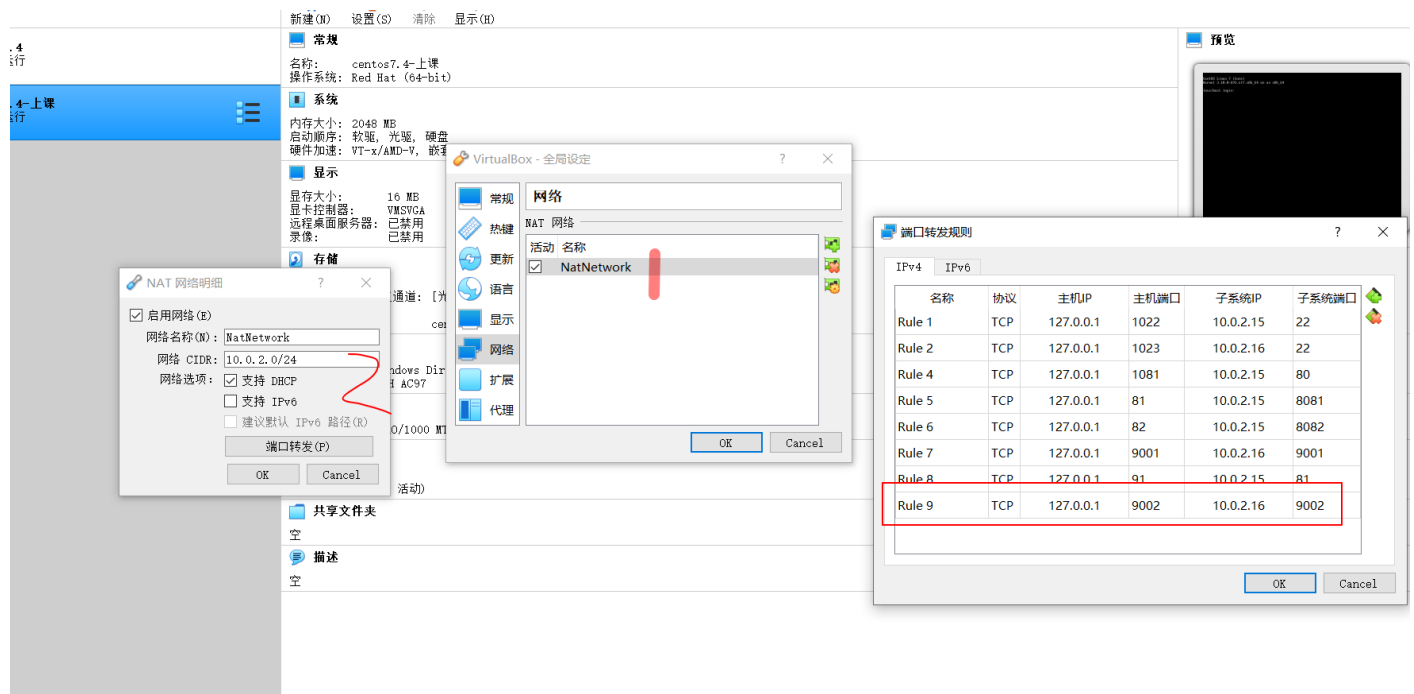
server {
    listen 9002;
    server_name www.cpf.com;
    location / {
        proxy_pass http://myServers;
    }
}

```

重新加载 nginx.conf

```
[root@localhost sbin]# ./nginx -s reload
```

3. 配置NAT模式的端口映射（如果大家用桥接，不用理会）。



测试：

浏览器请求：<http://www.cpf.com/load-balance/index.html>。

← → ↻ ⚠ 不安全 | cpf.com:9002/load-balance/index.html

8081

8082

负载均衡的方法（算法）

1. Round Robin。轮询。

```
# server list
upstream myServers {
    server localhost:8081;
    server localhost:8082;
}
```

2. [Least Connections](#)

```
# server list
upstream myServers {
    least_conn;
    server localhost:8081;
    server localhost:8082;
}
```

3. [IP Hash](#)

```
# server list
upstream myServers {
    ip_hash;
    server localhost:8081;
    server localhost:8082;
}
```

将 8082 标记为down

```
# server list
upstream myServers {
    ip_hash;
    server localhost:8081;
    server localhost:8082 down;
```

```
}
```

4. Generic [Hash](#)。

```
# server list
upstream myServers {
    hash $request_uri consistent;
    server localhost:8081;
    server localhost:8082;
}
```

5. 暂时不管。

6. [Random](#)

```
# server list
upstream myServers {
    random two least_conn;
    server localhost:8081;
    server localhost:8082;
}
```

服务权重

```
# server list
upstream myServers {
    server localhost:8081 weight=2;
    server localhost:8082 weight=8;
}
```