**HAproxy代理服务器**

【HAproxy代理服务器】

HAProxy软件与Nginx是一致的，原理也一样，都是代理服务器软件.

LVS:4层调度,仅关心TCP|UDP和端口

HAproxy和Nginx既可以做4层代理也可以做7层代理

性能:LVS>HAproxy>Nginx

功能:Nginx>HAproxy>LVS

【准备环境：把前面的实验删除】

1)把两台LVS主机的keepalived服务关闭

[root@proxy ~]# systemctl stop keepalived

[root@proxy2 ~]# systemctl stop keepalived

2)把web1和web2的伪装IP删除

[root@web1 ~]# rm -rf /etc/sysconfig/network-scripts/ifcfg-lo:0

[root@web1 ~]# systemctl restart network

[root@web2 ~]# rm -rf /etc/sysconfig/network-scripts/ifcfg-lo:0

[root@web2 ~]# systemctl restart network

【在192.168.4.5主机配置haproxy服务】

[root@proxy ~]# yum -y install haproxy

[root@proxy ~]# vim +86 /etc/haproxy/haproxy.cfg

#在文件的末尾手动添加如下内容:

listen servers \*:80

#servers可以是任意字符,\*代表所有IP(或者0.0.0.0)

#算法是轮询

server web1 192.168.4.100:80 check

server web2 192.168.4.200:80 check

[root@proxy ~]# systemctl start haproxy

【注意事项：一定确保本机没有任何其他服务使用80端口】

【注意事项：一定确保SELinux是关闭的】

【备注】

确保web1和web2的网络是可以通信的

确保web1和web2的httpd服务是启动

balance roundrobin

【备注haproxy的配置文件说明】

balance roundrobin

#该参数代表算法,参考baidu

或/usr/share/doc/haproxy-1.5.18/configuration.txt帮助(1584行)

server web1 192.168.4.100:80 weight 2 check rise 5 fall 3

#server配置后端真实服务器,web1这个名称可以任意

#后端服务器IP和端口是192.168.4.100:80

#check是健康检查

#fall 3代表最大失败次数(几次测试失败算失败)

#rise 5代表最大成功次数(几次测试成功算成功)

#weight 2代表权重

【继续修改配置，开启haproxy的状态页面】

[root@proxy ~]# vim /etc/haproxy/haproxy.cfg

在文件的末尾手动添加如下内容:

listen stats 0.0.0.0:1080 #监听端口(任意端口)

stats refresh 30s #统计页面自动刷新时间,任意时间

stats uri /stats #统计页面url(任意字符)

stats realm Haproxy Manager #进入管理解面查看状态信息

stats auth admin:admin #统计页面用户名和密码设置,任意

[root@proxy ~]# systemctl restart haproxy

[root@proxy ~]# ss -ntulp |grep haproxy

客户端浏览器访问http://192.168.4.5:1080/stats

用户名:admin,密码:admin

Queue队列数据的信息（当前队列数量，最大值，队列限制数量）；

Session rate每秒会话率（当前值，最大值，限制数量）；

Sessions总会话量（当前值，最大值，总量，Lbtot: total number of times a server was selected选中一台服务器所用的总时间）；

Bytes（入站、出站流量）；

Denied（拒绝请求、拒绝回应）；

Errors（错误请求、错误连接、错误回应）；

Warnings（重新尝试警告retry、重新连接redispatches）；

Server(状态、最后检查的时间（多久前执行的最后一次检查）、权重、备份服务器数量、down机服务器数量、down机时长)。

扩展:

IP对应的是主机,端口对应服务.

不同的主机不能IP冲突,不同的服务不能端口冲突.

不允许两个或多个服务同时占用相同的一个端口.

一台主机可以配置多个IP(前提是不能冲突)

一个服务也可以占用多个端口(前提是不能冲突)