**haproxy扩展**

LB负载均衡的配置

1.做一群LAMP

主机node52、node53、node54、node55

Yum -y install httpd mariadb-server mariadb php php-mysql

Systemctl restart httpd mariadb

Systemctl enable httpd mariadb

测试LAMP,任意一台主机，每台主机都要测试

vim /var/ww/html/1.php

<?php

$x=mysql\_connect("localhost","root","");

if($x){echo "ok" ; }else{echo "error"; };

?>

2.配置LB调度器（分发器）第四层，基于端口的调度

Node51:

yum -y install haproxy

cp /et/haproxy/haproxy.cfg /root/ //备份

**vim /etc/haproxy/haproxy.cfg**

Default

stats uri /admin #以友好的页面展示LB集群实际状况

listen weblb 192.168.4.51:80 //webla为机群的名字，使用51的80做集群监听

cookie SERVERID rewrite //记录下面服务主机的cookie名

balance roundrobin //使用轮询的调度算法

server weba 192.168.4.55:80 cookie app1inst1 check inter 2000 rise 2 fall 5

server webb 192.168.4.54:80 cookie app2inst2 check inter 2000 rise 2 fall 5

server webc 192.168.4.53:80 cookie app3inst3 check inter 2000 rise 2 fall 5

server webd 192.168.4.52:80 cookie app4inst4 check inter 2000 rise 2 fall 5

//一个server表示一个服务器，weba为该服务器自有取的名字，cookie的名字是 app1inst1，每隔2000毫秒(2秒)检查一次，如果连不上rise表示隔2毫秒再连一次，如果连续5次都连不上就判断这台主机down掉了

如果有其他的应用占用了80，需要将其停掉

启动，并设置开机自启

Systemctl restart haproxy

Systemctl enable haproxy

Systemctl status haproxy

测试访问，如果能狗以轮询的方式显示node52、node53、node54、node55上web的内容，表示配置无误

Fontend 前端服务

Backend 后端服务器

**Haproxy基于业务流量划分**

frontend weblb 192.168.4.51:80 前端服务

#acl url\_static path\_beg -i .jpg .gif .png .css .js

#acl acl\_name beg->开头 end->结尾 -i->忽略大小写 匹配以什么扩展名结尾的文件

acl url\_html path\_end -i .html #匹配到以.php页面的请求采用url\_php这条规则

acl url\_php path\_end -i .php

use\_backend htmlgrp if url\_html #符合url\_html规则的请求,将请求发到到后端

htmlgrp服务器组中

use\_backend htmlgrp if url\_php

default\_backend htmlgrp #不指定acl规则，默认访问的服务器中为htmlgrp

backend htmlgrp #定义服务器组,和组名htmlgrp

balance roundrobin #使用负载均衡的算法

server weba 192.168.4.55:80 check #real server 使用default中的方法对该主机进行健康性检查

server webb 192.168.4.54:80 check

backend phpgrp #定义服务器组,和组名htmlgrp

balance roundrobin #使用负载均衡的算法

server webc 192.168.4.52:80 check

server webd 192.168.4.53:80 check

**使用Haproxy软件配置LB集群**

工作模式 1.mode http

2.mode tcp

3.mode htalth

Mode http

1.http close

每次响应请求后立即断开连接

2.keep-alive

每一连接可以发起多个请求，但是多个请求需要服务器按顺序一个一个响应不能一次性作出响应

3.pipelining

一次连接可以发起多次请求，这些请求服务器并行进行响应，同时处理请求，减少了延迟，但是对服务器的要求比较高，

配置：

1.装包

yum -y install haproxy

1. 配置文件及介绍

只保留全局配置和默认配置

Vim /etc/haproxy/haproxy.cfg

default：为后续的其他部分设置默认参数，缺省参数可以被后续部分覆盖

frontend：描述接收客户端侦听套接紫集

backend：描述转发连接的服务器集

listen：把frontend和backend结合到一起的完整声明

//这里为/etc/haproxy/haproxy.cfg的配置

vim /et/harpoxy/haproxy.cfg

default

stats uri /admin #以友好的页面展示LB集群实际状况

listen webla 192.168.4.56:80 //webla为机群的名字，使用56的80做集群监听

cookie SERVERID rewrite //记录下面服务主机的cookie名

balance roundrobin //使用轮询的调度算法

server weba 192.168.4.55:80 cookie app1inst1 check inter 2000 rise 2 fall 5

server webb 192.168.4.57:80 cookie app2inst2 check inter 2000 rise 2 fall 5

//一个server表示一个服务器，weba为该服务器自有取的名字，cookie的名字是 app1inst1，每隔2000毫秒检查一次，如果连不上rise表示隔2秒再连一次，如

果连续5次都连不上就判断这台主机down掉了

1. 启动服务及查看状态

Systemctl restart haproxy

Systemctl status haproxy

1. 有好页面的访问

http://192.168.4.56/admin