Haproxy动静分离

环境：centos7.4虚拟机7台，关闭防火墙、SELinux，搭建好yum，清空iptables规则

规划：

Web1 httpd 192.168.1.1 静态html网页

Web2 httpd 192.168.1.2 静态html网页

Web3 nginx+php-fpm 192.168.1.3 动态php网页

Web4 nginx+php-fpm 192.168.1.4 动态php网页

Web5 Tomcat 192.168.1.5 动态jsp网页

Web6 Tomcat 192.168.1.6 动态jsp网页

Proxy Haproxy 192.168.1.7

##################################################################

静态后台部署

[root@web1 ~]# yum -y install httpd

[root@web1 ~]# echo html1 > /var/www/html/test.html

[root@web1 ~]# systemctl start httpd

[root@web1 ~]# netstat -anptu | grep httpd

tcp6 0 0 :::80 :::\* LISTEN 1211/httpd

[root@web1 ~]#

[root@web2 ~]# yum -y install httpd

[root@web2 ~]# echo html2 > /var/www/html/test.html

[root@web2 ~]# systemctl start httpd

[root@web2 ~]# netstat -anptu | grep httpd

tcp6 0 0 :::80 :::\* LISTEN 1213/httpd

[root@web2 ~]#

测试静态页面

[root@hostos ~]# curl 192.168.1.1/test.html

html1

[root@hostos ~]# curl 192.168.1.2/test.html

html2

[root@hostos ~]#

##################################################################

Php后台部署

[root@php1 ~]# ls

in\_nginx.sh nginx-1.12.2.tar.gz php-fpm-5.4.16-42.el7.x86\_64.rpm

[root@php1 ~]# cat in\_nginx.sh

#!/bin/bash

#Auther: Mark Li

#Date: 2019-12-04

#Desc: 安装NGINX

#Version: V1

######exit mask######

#2-->没指定安装包

#3-->Yum源不可用

#4-->安装失败

#5-->编译失败

#6-->环境检测失败

#检查安装包是否存在

if [ $# -ne 1 ];then

echo "请指定一个NGINX安装包"

exit 2

fi

ngx\_ver=`echo $1 | sed -n 's/.tar.gz//p'`

#检查yum源是否可用

yum clean all &> /dev/null

pac\_nums=`yum repolist | tail -1 | awk -F: '{print $2}' | sed -n 's/,//p' | awk '{print $1}'`

if [ ${pac\_nums} -eq 0 ];then

echo "Yum源不可用，请先配置Yum源"

exit 3

fi

#安装依赖包

yum -y install gcc make zlib-devel pcre-devel openssl-devel &> /dev/null

echo "依赖包安装完成"

id nginx &> /dev/null

if [ $? -ne 0 ];then

useradd -s /sbin/nologin nginx

fi

#编译安装NGINX

tar -xf $1

cd ${ngx\_ver}

cp -r contrib/vim/\* /usr/share/vim/vimfiles/ &> /dev/null

./configure --prefix=/usr/local/nginx --user=nginx --group=nginx --with-http\_ssl\_module --with-stream --with-http\_stub\_status\_module --without-http\_autoindex\_module --without-http\_ssi\_module &> /dev/null

if [ $? -eq 0 ];then

echo "环境检测通过，开始编译"

make &> /dev/null

if [ $? -eq 0 ];then

echo "二进制编译成功，开始安装"

make install &> /dev/null

if [ $? -eq 0 ];then

echo "安装成功"

else

echo "安装失败"

exit 4

fi

else

echo "二进制编译失败"

exit 5

fi

else

echo "环境监测未通过"

exit 6

fi

#清理安装包

cd ..

rm -rf ${ngx\_ver}

#创建service脚本

cat >> /lib/systemd/system/nginx.service << EOF

[Unit]

Description=nginx

After=network.target

[Service]

Type=forking

PIDFile=/usr/local/nginx/logs/nginx.pid

ExecStartPre=/usr/local/nginx/sbin/nginx -tc /usr/local/nginx/conf/nginx.conf

ExecStart=/usr/local/nginx/sbin/nginx -c /usr/local/nginx/conf/nginx.conf

ExecReload=/usr/local/nginx/sbin/nginx -s reload

ExecStop=/usr/local/nginx/sbin/nginx/-s quit

PrivateTmp=true

[Install]

WantedBy=multi-user.target

EOF

[root@php1 ~]# bash in\_nginx.sh nginx-1.12.2.tar.gz

依赖包安装完成

环境检测通过，开始编译

二进制编译成功，开始安装

安装成功

[root@php1 ~]#

[root@php1 ~]# yum -y install php php-devel

[root@php1 ~]# rpm -ivh --force --nodeps php-fpm-5.4.16-42.el7.x86\_64.rpm

[root@php1 ~]# cd /usr/local/nginx/

[root@php1 nginx]# vim conf/nginx.conf

[root@php1 nginx]# sed -rn '65,70p' conf/nginx.conf

location ~ \.php$ {

root html;

fastcgi\_pass 127.0.0.1:9000;

fastcgi\_index index.php;

include fastcgi.conf;

}

[root@php1 nginx]# ./sbin/nginx

[root@php1 nginx]# netstat -anptu | grep nginx

tcp 0 0 0.0.0.0:80 0.0.0.0:\* LISTEN 4006/nginx: master

[root@php1 nginx]# systemctl start php-fpm.service

[root@php1 nginx]# vim html/test.php

[root@php1 nginx]# cat html/test.php

<?php

$i="php1";

echo $i;

?>

[root@php2 ~]# ls

in\_nginx.sh nginx-1.12.2.tar.gz php-fpm-5.4.16-42.el7.x86\_64.rpm

[root@php2 ~]# bash in\_nginx.sh nginx-1.12.2.tar.gz

依赖包安装完成

环境检测通过，开始编译

二进制编译成功，开始安装

安装成功

[root@php2 ~]# yum -y install php php-devel &> /dev/null

[root@php2 ~]# rpm -ivh --force --nodeps php-fpm-5.4.16-42.el7.x86\_64.rpm

[root@php2 ~]# cd /usr/local/nginx/

[root@php2 nginx]# vim conf/nginx.conf

[root@php2 nginx]# sed -rn '65,70p' conf/nginx.conf

location ~ \.php$ {

root html;

fastcgi\_pass 127.0.0.1:9000;

fastcgi\_index index.php;

include fastcgi.conf;

}

[root@php2 nginx]# ./sbin/nginx

[root@php2 nginx]# netstat -antpu | grep nginx

tcp 0 0 0.0.0.0:80 0.0.0.0:\* LISTEN 3999/nginx: master

[root@php2 nginx]# systemctl start php-fpm.service

[root@php2 nginx]# netstat -antpu | grep php

tcp 0 0 127.0.0.1:9000 0.0.0.0:\* LISTEN 4026/php-fpm: maste

[root@php2 nginx]# vim html/test.php

[root@php2 nginx]# cat html/test.php

<?php

$i="php2";

echo $i;

?>

[root@php2 nginx]#

测试php页面

[root@hostos lnmp\_soft]# curl 192.168.1.3/test.php

php1[root@hostos lnmp\_soft]#

[root@hostos lnmp\_soft]# curl 192.168.1.4/test.php

php2[root@hostos lnmp\_soft]#

[root@hostos lnmp\_soft]#

##################################################################

Java后台部署

[root@tomcat1 ~]# ls

apache-tomcat-8.0.30.tar.gz

[root@tomcat1 ~]# yum -y install java-1.8.0-openjdk-devel

[root@tomcat1 ~]# tar -xf apache-tomcat-8.0.30.tar.gz

[root@tomcat1 ~]# mv apache-tomcat-8.0.30 /usr/local/tomcat

[root@tomcat1 ~]# rm -rf /dev/random

[root@tomcat1 ~]# ln -s /dev/urandom /dev/random

[root@tomcat1 ~]# vim /usr/local/tomcat/webapps/ROOT/test.jsp

[root@tomcat1 ~]# cat /usr/local/tomcat/webapps/ROOT/test.jsp

Tomcat1 now time is: <%=new java.util.Date()%>

[root@tomcat1 ~]# /usr/local/tomcat/bin/startup.sh

Using CATALINA\_BASE: /usr/local/tomcat

Using CATALINA\_HOME: /usr/local/tomcat

Using CATALINA\_TMPDIR: /usr/local/tomcat/temp

Using JRE\_HOME: /usr

Using CLASSPATH: /usr/local/tomcat/bin/bootstrap.jar:/usr/local/tomcat/bin/tomcat-juli.jar

Tomcat started.

[root@tomcat1 ~]# jps

1394 Jps

1373 Bootstrap

[root@tomcat1 ~]# netstat -anptu | grep java

tcp6 0 0 :::8080 :::\* LISTEN 1373/java

tcp6 0 0 127.0.0.1:8005 :::\* LISTEN 1373/java

tcp6 0 0 :::8009 :::\* LISTEN 1373/java

[root@tomcat1 ~]#

[root@tomcat2 ~]# ls

apache-tomcat-8.0.30.tar.gz

[root@tomcat2 ~]# yum -y install java-1.8.0-openjdk-devel

[root@tomcat2 ~]# tar -xf apache-tomcat-8.0.30.tar.gz

[root@tomcat2 ~]# mv apache-tomcat-8.0.30 /usr/local/tomcat

[root@tomcat2 ~]# rm -rf /dev/random

[root@tomcat2 ~]# ln -s /dev/urandom /dev/random

[root@tomcat2 ~]# vim /usr/local/tomcat/webapps/ROOT/test.jsp

[root@tomcat2 ~]# cat /usr/local/tomcat/webapps/ROOT/test.jsp

Tomcat2 now time is: <%=new java.util.Date()%>

[root@tomcat2 ~]# /usr/local/tomcat/bin/startup.sh

Using CATALINA\_BASE: /usr/local/tomcat

Using CATALINA\_HOME: /usr/local/tomcat

Using CATALINA\_TMPDIR: /usr/local/tomcat/temp

Using JRE\_HOME: /usr

Using CLASSPATH: /usr/local/tomcat/bin/bootstrap.jar:/usr/local/tomcat/bin/tomcat-juli.jar

Tomcat started.

[root@tomcat2 ~]# jps

1378 Jps

1353 Bootstrap

[root@tomcat2 ~]# netstat -anptu | grep java

tcp6 0 0 :::8080 :::\* LISTEN 1353/java

tcp6 0 0 127.0.0.1:8005 :::\* LISTEN 1353/java

tcp6 0 0 :::8009 :::\* LISTEN 1353/java

[root@tomcat2 ~]#

测试jsp页面

[root@hostos ~]# curl 192.168.1.5:8080/test.jsp

Tomcat1 now time is: Sun Jan 19 16:36:52 CST 2020

[root@hostos ~]# curl 192.168.1.6:8080/test.jsp

Tomcat2 now time is: Sun Jan 19 16:36:55 CST 2020

[root@hostos ~]#

##################################################################

部署调度器

[root@proxy ~]# vim /etc/hosts

[root@proxy ~]# tail -7 /etc/hosts

192.168.1.1 web1

192.168.1.2 web2

192.168.1.3 php1

192.168.1.4 php2

192.168.1.5 tomcat1

192.168.1.6 tomcat2

192.168.1.7 proxy

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

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

[root@proxy ~]# sed -rn '12p;26,33p;36p;42,58p;87,$p' /etc/haproxy/haproxy.cfg

global

log 127.0.0.1 local2

chroot /var/lib/haproxy

pidfile /var/run/haproxy.pid

maxconn 4000

user haproxy

group haproxy

daemon

stats socket /var/lib/haproxy/stats

defaults

mode http

log global

option httplog

option dontlognull

option http-server-close

option forwardfor except 127.0.0.0/8

option redispatch

retries 3

timeout http-request 10s

timeout queue 1m

timeout connect 10s

timeout client 1m

timeout server 1m

timeout http-keep-alive 10s

timeout check 10s

maxconn 3000

#---------------------------------------------------------------------

# haproxy health check configure

#---------------------------------------------------------------------

listen admin\_stats

bind 0.0.0.0:1080

stats enable

mode http

log global

stats uri /stats

stats realm Haproxy\ Statistics

stats auth admin:admin

stats hide-version

stats admin if TRUE

stats refresh 30s

#---------------------------------------------------------------------

# static and dynamic split

#---------------------------------------------------------------------

listen static\_dynamic

bind 0.0.0.0:80

maxconn 5000

mode http

log global

option httplog

option httpclose

option forwardfor

log global

#定义默认后台

default\_backend default

##定义acl列表

#静态资源（路径、页面）

acl url\_static path\_beg -i /static /image /img /javascript /stylesheets

acl url\_static path\_end -i .jpg .gif .png .css .js .html

#php页面

acl url\_php path\_end -i .php

#jsp页面

acl url\_jsp path\_end -i .jsp .do

#acl列表应用规则

use\_backend static\_pool if url\_static #or host\_static

use\_backend php\_pool if url\_php

use\_backend jsp\_pool if url\_jsp

backend static\_pool

option httpchk GET /test.html

server web1 192.168.1.1:80 cookie id1 check inter 2000 rise 2 fall 3

server web2 192.168.1.2:80 cookie id1 check inter 2000 rise 2 fall 3

backend php\_pool

option httpchk GET /test.php

server php1 192.168.1.3:80 cookie id2 check inter 2000 rise 2 fall 3

server php2 192.168.1.4:80 cookie id2 check inter 2000 rise 2 fall 3

backend jsp\_pool

option httpchk GET /test.jsp

server tomcat1 192.168.1.5:8080 cookie id3 check inter 2000 rise 2 fall 3

server tomcat2 192.168.1.6:8080 cookie id3 check inter 2000 rise 2 fall 3

backend default

mode http

option httpchk GET /test.html

server default 192.168.1.1:80 cookie id1 check inter 2000 rise 2 fall 3 maxconn 5000

server default 192.168.1.2:80 cookie id1 check inter 2000 rise 2 fall 3 maxconn 5000

[root@proxy ~]# systemctl restart haproxy.service

[root@proxy ~]# netstat -anptu | grep haproxy

tcp 0 0 0.0.0.0:80 0.0.0.0:\* LISTEN 1445/haproxy

tcp 0 0 0.0.0.0:1080 0.0.0.0:\* LISTEN 1445/haproxy

udp 0 0 0.0.0.0:52465 0.0.0.0:\* 1444/haproxy

[root@proxy ~]#

##################################################################

测试动静分离

[root@hostos ~]# curl 192.168.1.7/test.html

html1

[root@hostos ~]# curl 192.168.1.7/test.html

html2

[root@hostos ~]# curl 192.168.1.7/test.php

php2[root@hostos ~]#

[root@hostos ~]# curl 192.168.1.7/test.php

php1[root@hostos ~]#

[root@hostos ~]# curl 192.168.1.7/test.jsp

Tomcat1 now time is: Sun Jan 19 16:54:31 CST 2020

[root@hostos ~]# curl 192.168.1.7/test.jsp

Tomcat2 now time is: Sun Jan 19 16:54:33 CST 2020

[root@hostos ~]#

