**keepalived配置高可用集群**

1、准备2台设备，1台master（21），1台backup（24）

2、安装keepalived //2台设备都要安装

yum install -y keepalived

3、在master上编辑配置文件

vim /etc/keepalived/keepalived.conf //去掉以前的配置

global\_defs { //全局配置

notification\_email {

googleqicq@sina.com

}

notification\_email\_from root@googleqicq@sina.com

smtp\_server 127.0.0.1

smtp\_connect\_timeout 30

router\_id LVS\_DEVEL

}

vrrp\_script chk\_nginx { //检查nginx

script "/usr/local/sbin/check\_ng.sh" //定义检查nginx的脚本路径

interval 3 //每3秒检查一次

}

vrrp\_instance VI\_1 {

state MASTER

interface eno16777736 //定义网卡名称

virtual\_router\_id 51 //id backup上要一致

priority 100 //权重

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 123456

}

virtual\_ipaddress {

192.168.1.100 //vip

}

track\_script {

chk\_nginx

}

}

4、定义检查nginx的脚本

cat /usr/local/sbin/check\_ng.sh

#!/bin/bash

#时间变量，用于记录日志

d=`date --date today +%Y%m%d\_%H:%M:%S`

#计算nginx进程数量

n=`ps -C nginx --no-heading|wc -l`

#如果进程为0，则启动nginx，并且再次检测nginx进程数量，

#如果还为0，说明nginx无法启动，此时需要关闭keepalived

if [ $n -eq "0" ]; then

/etc/init.d/nginx start

n2=`ps -C nginx --no-heading|wc -l`

if [ $n2 -eq "0" ]; then

echo "$d nginx down,keepalived will stop" >> /var/log/check\_ng.log

systemctl stop keepalived

fi

fi

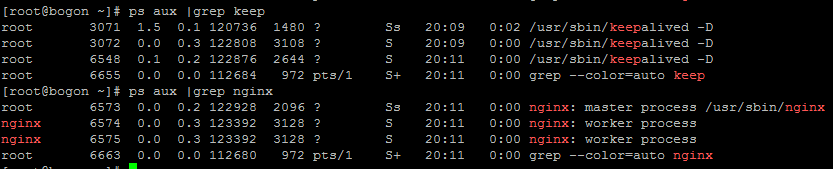
5、设置监控nginx脚本权限

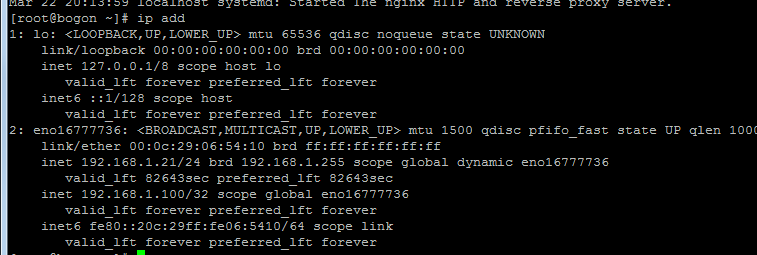
chmod 755 /usr/local/sbin/check\_ng.sh

6、启动master上的服务

systemctl start keepalived

7、查看启动是否正常，启动keepalived后，会自动启动nginx，无需手动





8、编辑配置文件，在backup上操作

vim /etc/keepalived/keepalived.conf

global\_defs {

notification\_email {

googlqicq@sina.com

}

notification\_email\_from root@googleqicq@sina.com

smtp\_server 127.0.0.1

smtp\_connect\_timeout 30

router\_id LVS\_DEVEL

}

vrrp\_script chk\_nginx {

script "/usr/local/sbin/check\_ng.sh"

interval 3

}

vrrp\_instance VI\_1 {

state BACKUP

interface eno16777736

virtual\_router\_id 51

priority 90

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 123456

}

virtual\_ipaddress {

192.168.1.100

}

track\_script {

chk\_nginx

}

}

9、在backup上定义检测nginx脚本

vim /usr/local/sbin/check\_ng.sh

#时间变量，用于记录日志

d=`date --date today +%Y%m%d\_%H:%M:%S`

#计算nginx进程数量

n=`ps -C nginx --no-heading|wc -l`

#如果进程为0，则启动nginx，并且再次检测nginx进程数量，

#如果还为0，说明nginx无法启动，此时需要关闭keepalived

if [ $n -eq "0" ]; then

systemctl start nginx //backup没有nginx启动脚本

n2=`ps -C nginx --no-heading|wc -l`

if [ $n2 -eq "0" ]; then

echo "$d nginx down,keepalived will stop" >> /var/log/check\_ng.log

systemctl stop keepalived

fi

fi

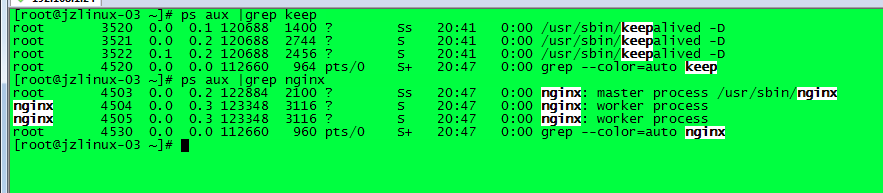
10、设置监控nginx脚本权限

chmod 755 /usr/local/sbin/check\_ng.sh

11、启动backup上的keepalived

systemctl start keepalived

12、backup上的keepalived启动后，ningx服务会自动启动，但是vip不会监听，只有master上的keepalived断开后，才会监听backup的vip





13、测试

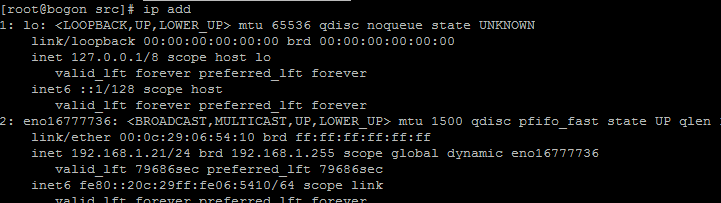
/usr/share/nginx/html/index.html //yum安装nginx的默认访问路径

在master上写入 “this is master”

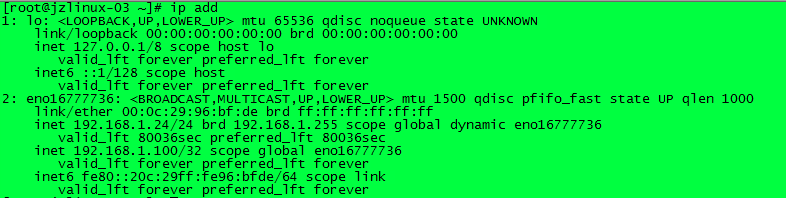
在backup上写入 “this is slave”

可以在浏览器上输入VIP，来确定区分是哪个角色在工作。

systemctl stop keepalived //停掉master上的服务，此时上面的vip会释放出来



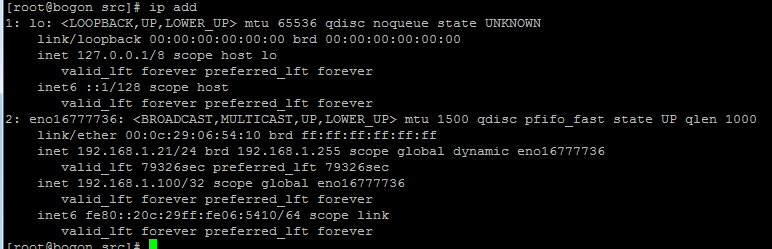
此时backup上的vip会被监听



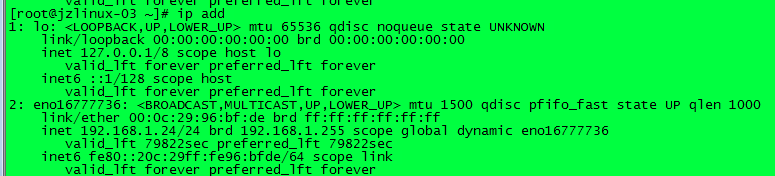


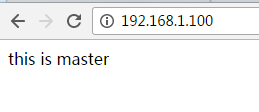
启动master的服务

systemctl start keepalived //vip被监听



backup上的vip被释放





**LVS NAT模式搭建**

1、准备3台虚拟机：

1台分发器，2个网卡，一个内网网卡：192.168.1.30，与服务器通信，

一个公网网卡：192.168.119.128（虚拟机上设置仅主机模式）用于和外部通信

2台服务器，均配置一个内网IP，rs1:192.168.1.21 rs2:192.168.1.24,这2台服务器的网关，要设置成服务器的内网IP，及192.168.1.30

2、关闭iptables和selinux //设备上都要执行

systemctl stop firewalld; systemctl disable firewalld

systemctl start iptables; iptables -F; service iptables save

setenforce 0

3、在分发器上安装软件包

yum install -y ipvsadm

4、vim /usr/local/sbin/lvs\_nat.sh //在分发器上

#! /bin/bash

# director 服务器上开启路由转发功能

echo 1 > /proc/sys/net/ipv4/ip\_forward

# 关闭icmp的重定向

echo 0 > /proc/sys/net/ipv4/conf/all/send\_redirects

echo 0 > /proc/sys/net/ipv4/conf/default/send\_redirects

# 注意区分网卡名字，阿铭的两个网卡分别为ens33和ens37

echo 0 > /proc/sys/net/ipv4/conf/eno16777736/send\_redirects

echo 0 > /proc/sys/net/ipv4/conf/ens37/send\_redirects

# director 设置nat防火墙

iptables -t nat -F

iptables -t nat -X

iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -j MASQUERADE

# director设置ipvsadm

IPVSADM='/usr/sbin/ipvsadm'

$IPVSADM -C

$IPVSADM -A -t 192.168.119.128:80 -s wlc -p 3 //-p 3秒，该时间内同一来源IP可分发到同一设备上去。

$IPVSADM -a -t 192.168.119.128:80 -r 192.168.1.21:80 -m -w 1

$IPVSADM -a -t 192.168.119.128:80 -r 192.168.1.24:80 -m -w 1

// -r 指定rs(服务器)的IP，-m：nat模式 -w：权重

5、执行脚本

4、可以在分发器上

curl 192.168.119.128 （公网IP）

也可以在浏览器上输入该IP

**LVS DR模式搭建**

1、在分发器上编辑脚本

vim /usr/local/sbin/lvs\_dr.sh

#! /bin/bash

echo 1 > /proc/sys/net/ipv4/ip\_forward

ipv=/usr/sbin/ipvsadm

vip=192.168.1.200 //公网IP

rs1=192.168.1.21

rs2=192.168.1.24

#注意这里的网卡名字

ifdown eno16777736

ifup eno16777736

ifconfig eno16777736:2 $vip broadcast $vip netmask 255.255.255.255 up

route add -host $vip dev eno16777736:2

$ipv -C

$ipv -A -t $vip:80 -s rr

$ipv -a -t $vip:80 -r $rs1:80 -g -w 1

$ipv -a -t $vip:80 -r $rs2:80 -g -w 1

//-g DR模式

2、在2台RS服务器上编辑脚本

#/bin/bash

vip=192.168.1.200

#把vip绑定在lo上，是为了实现rs直接把结果返回给客户端

ifdown lo

ifup lo

ifconfig lo:0 $vip broadcast $vip netmask 255.255.255.255 up

route add -host $vip lo:0

#以下操作为更改arp内核参数，目的是为了让rs顺利发送mac地址给客户端

#参考文档www.cnblogs.com/lgfeng/archive/2012/10/16/2726308.html

echo "1" >/proc/sys/net/ipv4/conf/lo/arp\_ignore

echo "2" >/proc/sys/net/ipv4/conf/lo/arp\_announce

echo "1" >/proc/sys/net/ipv4/conf/all/arp\_ignore

echo "2" >/proc/sys/net/ipv4/conf/all/arp\_announce

3、分别执行脚本

4、测试

在浏览器上输入vip（192.168.1.200）

或者重新开启一台虚拟机，curl 192.168.1.200

**keepalived lvs**

1、分发器上安装 keepalived //在2台分发器上安装，一台maser,一台backup

yum install -y keepalived

echo 1 > /proc/sys/net/ipv4/ip\_forward //在分发器上执行。

2、编辑配置文件

vrrp\_instance VI\_1 {

#备用服务器上为 BACKUP

state MASTER

#绑定vip的网卡为ens33，你的网卡和阿铭的可能不一样，这里需要你改一下

interface eno16777736

virtual\_router\_id 51

#备用服务器上为90

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass aminglinux

}

virtual\_ipaddress {

192.168.1.200

}

}

virtual\_server 192.168.1.200 80 {

#(每隔10秒查询realserver状态)

delay\_loop 10

#(lvs 算法)

lb\_algo wlc

#(DR模式)

lb\_kind DR

#(同一IP的连接60秒内被分配到同一台realserver)

persistence\_timeout 60

#(用TCP协议检查realserver状态)

protocol TCP

real\_server 192.168.1.21 80 {

#(权重)

weight 100

TCP\_CHECK {

#(10秒无响应超时)

connect\_timeout 10

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

real\_server 192.168.1.24 80 {

weight 100

TCP\_CHECK {

connect\_timeout 10

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

}

3、启动服务 systemctl start keepalived

需要2台keepalived，设置主要是：state （MASTER||BACKUP), priority(100||90)

4、

在2台RS服务器上编辑脚本

#/bin/bash

vip=192.168.1.200

#把vip绑定在lo上，是为了实现rs直接把结果返回给客户端

ifdown lo

ifup lo

ifconfig lo:0 $vip broadcast $vip netmask 255.255.255.255 up

route add -host $vip lo:0

#以下操作为更改arp内核参数，目的是为了让rs顺利发送mac地址给客户端

#参考文档www.cnblogs.com/lgfeng/archive/2012/10/16/2726308.html

echo "1" >/proc/sys/net/ipv4/conf/lo/arp\_ignore

echo "2" >/proc/sys/net/ipv4/conf/lo/arp\_announce

echo "1" >/proc/sys/net/ipv4/conf/all/arp\_ignore

echo "2" >/proc/sys/net/ipv4/conf/all/arp\_announce