LVS-DR：直接路由

各集群节点，必须要跟directory在同一物理网络中

RIP可以使用公网地址，实现远程管理（也可以使用私有地址）

directory仅负责处理入展请求，响应报文则有realserver 直接发往客户端

集群节点不能将网关指向DIP

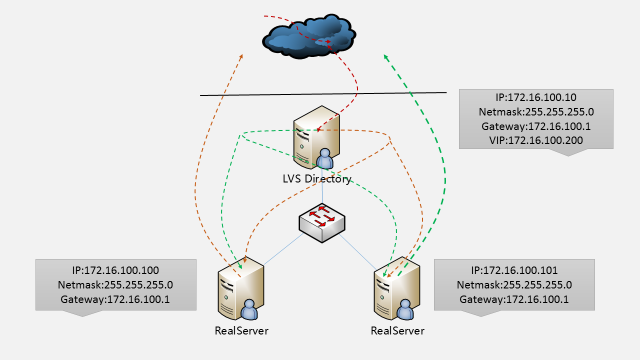
directory不支持端口映射

实施规划：

1. 同一物理网段的三个共有地址
2. 2.使用私有ip，映射VIP地址到外网，

实验信息和拓扑：  
备注：Centos 6.5 selinux –disabled iptables off

|  |  |
| --- | --- |
| Server Name | Ip address information |
| LVS Directory | 172.16.100.10/24 |
| VIP | 172.16100.200/24 |
| RealServer1 | 172.16.100.100/24 |
| RealServer2 | 172.16.100.101/24 |



检查内核是否已经支持LVS：

# grep -i 'CONFIG\_IP\_VS' /boot/config-2.6.32-431.el6.x86\_64

CONFIG\_IP\_VS=m

CONFIG\_IP\_VS\_IPV6=y

# CONFIG\_IP\_VS\_DEBUG is not set

CONFIG\_IP\_VS\_TAB\_BITS=12

CONFIG\_IP\_VS\_PROTO\_TCP=y

CONFIG\_IP\_VS\_PROTO\_UDP=y

CONFIG\_IP\_VS\_PROTO\_AH\_ESP=y

CONFIG\_IP\_VS\_PROTO\_ESP=y

CONFIG\_IP\_VS\_PROTO\_AH=y

CONFIG\_IP\_VS\_PROTO\_SCTP=y

CONFIG\_IP\_VS\_RR=m

CONFIG\_IP\_VS\_WRR=m

CONFIG\_IP\_VS\_LC=m

CONFIG\_IP\_VS\_WLC=m

CONFIG\_IP\_VS\_LBLC=m

CONFIG\_IP\_VS\_LBLCR=m

CONFIG\_IP\_VS\_DH=m

CONFIG\_IP\_VS\_SH=m

CONFIG\_IP\_VS\_SED=m

CONFIG\_IP\_VS\_NQ=m

CONFIG\_IP\_VS\_FTP=m

CONFIG\_IP\_VS\_PE\_SIP=m

[root@localhost ~]#

出现以上说明内核已经支持LVS模块

Directory的配置如下：  
安装ipvsadm管理工具：

# yum install ipvsadm –y  
使用ipvsadm配置lvs  
vim dirctory.sh

#!/bin/bash

#

# LVS script for VS/DR

#

. /etc/rc.d/init.d/functions

#

VIP=172.16.100.200

RIP1=172.16.100.100

RIP2=172.16.100.101

PORT=80

#

case "$1" in

start)

/sbin/ifconfig eth0:1 $VIP broadcast $VIP netmask 255.255.255.255 up

/sbin/route add -host $VIP dev eth0:1

# Since this is the Director we must beable to forward packets

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

# Clear all iptables rules.

/sbin/iptables -F

# Reset iptables counters.

/sbin/iptables -Z

# Clear all ipvsadm rules/services.

/sbin/ipvsadm -C

# Add an IP virtual service for VIP192.168.0.219 port 80

# In this recipe, we will use theround-robin scheduling method.

# In production, however, you should use aweighted, dynamic scheduling method.

/sbin/ipvsadm -A -t $VIP:80 -s wlc

# Now direct packets for this VIP to

# the real server IP (RIP) inside thecluster

/sbin/ipvsadm -a -t $VIP:80 -r $RIP1 -g -w 1

/sbin/ipvsadm -a -t $VIP:80 -r $RIP2 -g -w 2

/bin/touch /var/lock/subsys/ipvsadm &> /dev/null

;;

stop)

# Stop forwarding packets

echo 0 > /proc/sys/net/ipv4/ip\_forward

# Reset ipvsadm

/sbin/ipvsadm -C

# Bring down the VIP interface

/sbin/ifconfig eth0:1 down

/sbin/route del $VIP

/bin/rm -f /var/lock/subsys/ipvsadm

echo "ipvs is stopped..."

;;

status)

if [ ! -e /var/lock/subsys/ipvsadm ]; then

echo "ipvsadm is stopped ..."

else

echo"ipvs is running ..."

ipvsadm -L -n

fi

;;

\*)

echo "Usage: $0 {start|stop|status}"

;;

Esac

# sh dirctory.sh start  
两台R而阿里Server配置如下：  
保证正常的WEB访问即可  
vim realserver.sh

#!/bin/bash

#

# Script to start LVS DR real server.

# description: LVS DR real server

#

. /etc/rc.d/init.d/functions

VIP=172.16.100.200

host=`/bin/hostname`

case "$1" in

start)

# Start LVS-DR real server on this machine.

/sbin/ifconfig lo down

/sbin/ifconfig lo up

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

/sbin/ifconfig lo:0 $VIP broadcast $VIP netmask 255.255.255.255 up

/sbin/route add -host $VIP dev lo:0

;;

stop)

# Stop LVS-DR real server loopback device(s).

/sbin/ifconfig lo:0 down

echo 0 > /proc/sys/net/ipv4/conf/lo/arp\_ignore

echo 0 > /proc/sys/net/ipv4/conf/lo/arp\_announce

echo 0 > /proc/sys/net/ipv4/conf/all/arp\_ignore

echo 0 >/proc/sys/net/ipv4/conf/all/arp\_announce

;;

status)

# Status of LVS-DR real server.

islothere=`/sbin/ifconfig lo:0 | grep $VIP`

isrothere=`netstat -rn | grep "lo:0" | grep $VIP`

if [ ! "$islothere" -o ! "isrothere" ];then

# Either the route or the lo:0 device

# not found.

echo "LVS-DR real server Stopped."

else

echo "LVS-DR real server Running."

fi

;;

\*)

# Invalid entry.

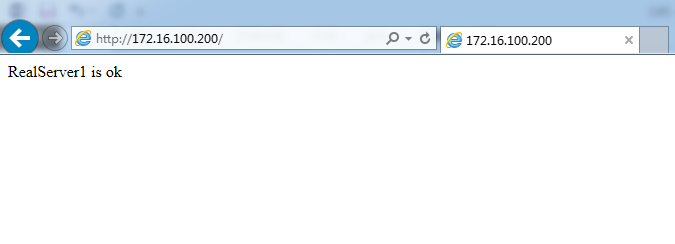
echo "$0: Usage: $0 {start|status|stop}"

exit 1

;;

esac

# sh realserver.sh start

访问WEB服务器：  


这里尝试刷新，发现在两台web服务器之间来回切换，实现负载。

# ip addr

1: lo: <LOOPBACK,UP,LOWER\_UP> mtu 16436 qdisc noqueue state UNKNOWN

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00

inet 127.0.0.1/8 scope host lo

inet6 ::1/128 scope host

valid\_lft forever preferred\_lft forever

2: eth0: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP qlen 1000

link/ether 00:0c:29:63:2b:5c brd ff:ff:ff:ff:ff:ff

inet 172.16.100.10/24 brd 172.16.100.255 scope global eth0

inet 172.16.100.200/32 brd 172.16.100.200 scope global eth0:1

inet6 fe80::20c:29ff:fe63:2b5c/64 scope link

valid\_lft forever preferred\_lft forever

[root@localhost ~]#

备注：

目前才在一个问题，就是如果后端的realserver宕机，如果将其从从lvs集群中剔除此服务器，并且当realserver恢复又是如何将其加入LVS集群中呢，这就是健康检查的功能，这里使用如下的脚本实现。

# vim check\_health.sh

#!/bin/bash

#

VIP=172.16.100.200

CPORT=80

FAIL\_BACK=127.0.0.1

RS=("172.16.100.100" "172.16.100.101")

declare -a RSSTATUS

RW=("2" "1")

RPORT=80

TYPE=g

CHKLOOP=3

LOG=/var/log/ipvsmonitor.log

addrs(){

ipvsadm -a -t $VIP:$CPORT -r $1:$RPORT -$TYPE-w $2

[ $? -eq 0 ] && return 0 || return 1

}

delrs(){

ipvsadm -d -t $VIP:$CPORT -r $1:$RPORT

[ $? -eq 0 ] && return 0 || return 1

}

checkrs(){

local I=1

while [ $I -le $CHKLOOP ]; do

if curl --connect-timeout 1 http://$1&> /dev/null; then

return 0

fi

let I++

done

return 1

}

initstatus(){

local I

local COUNT=0;

for I in ${RS[\*]}; do

if ipvsadm -L -n | grep "$I:$RPORT" && > /dev/null ; then

RSSTATUS[$COUNT]=1

else

RSSTATUS[$COUNT]=0

fi

let COUNT++

done

}

initstatus

while :; do

let COUNT=0

for I in ${RS[\*]}; do

if checkrs $I; then

if [ ${RSSTATUS[$COUNT]} -eq 0 ]; then

addrs $I ${RW[$COUNT]}

[ $? -eq 0 ] &&RSSTATUS[$COUNT]=1 && echo "`date +'%F %H:%M:%S'`, $I isback." >> $LOG

fi

else

if [ ${RSSTATUS[$COUNT]} -eq 1 ]; then

delrs $I

[ $? -eq 0 ] &&RSSTATUS[$COUNT]=0 && echo "`date +'%F %H:%M:%S'`, $I isgone." >> $LOG

fi

fi

let COUNT++

done

sleep 5

done

执行脚本：

# touch /var/log/ipvsmonitor.log

# chmod +x check\_health.sh

# nohup ./check\_health.sh &

# tail -f /var/log/ipvsmonitor.log

2014-10-25 18:11:59, 172.16.100.100 isgone.

2014-10-25 18:11:59, 172.16.100.101 isgone.

2014-10-25 18:18:08, 172.16.100.100 isback.

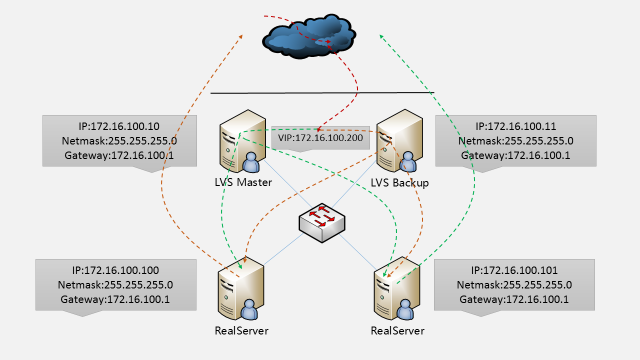
2014-10-25 18:23:26, 172.16.100.101 isback.

2014-10-25 18:23:56, 172.16.100.101 isgone.

2014-10-25 18:24:12, 172.16.100.101 isback.

启动和停止realserver上的服务，可以看到日志记录，LVS集群会对其进行剔除和添加功能。

|  |  |
| --- | --- |
| Server Name | Ip address information |
| LVS Master | 172.16.100.10/24 |
| LVS Backup | 172.16.100.11/24 |
| VIP | 172.16100.200/24 |
| RealServer1 | 172.16.100.100/24 |
| RealServer2 | 172.16.100.101/24 |



1. 配置lvs Master & lvs Backup

# yum install popt popt-devel popt-static libnl-devel libnl –y

# mkdir -p mkdir /usr/local/src/lvs

# cd /usr/local/src/lvs/

安装ipvsadm

# wget <http://www.linuxvirtualserver.org/software/kernel-2.6/ipvsadm-1.26.tar.gz>

# tar zxvf ipvsadm-1.26.tar.gz

# make && make install && echo "install LVS ok"

安装keepalived

# wget <http://www.keepalived.org/software/keepalived-1.2.13.tar.gz>

# tar zxvf keepalived-1.2.13.tar.gz

# cd keepalived-1.2.13

# ./configure && make && make install && echo "install keepalived ok" || echo "install keepalived is failed"

配置keepalived为启动脚本：

# cp /usr/local/etc/rc.d/init.d/keepalived /etc/init.d/

# cp /usr/local/etc/sysconfig/keepalived /etc/sysconfig/

# mkdir /etc/keepalived

# cp /usr/local/etc/keepalived/keepalived.conf /etc/keepalived/

# cp /usr/local/sbin/keepalived /usr/sbin/

LVS Backup 配置同上

Master keepalived.conf配置文件：

! Configuration File for keepalived

global\_defs {

notification\_email {

mengtao10@163.com

}

notification\_email\_from mengtao10@163.com

smtp\_server 127.0.0.1

router\_id LVS\_DEVEL

}

vrrp\_instance VI\_1 {

state MASTER

interface eth0

virtual\_router\_id 51

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1q2w3e4r5t6y

}

virtual\_ipaddress {

172.16.100.200

}

}

virtual\_server 172.16.100.200 80 {

delay\_loop 6

lb\_algo wrr

lb\_kind DR

persistence\_timeout 60

protocol TCP

real\_server 172.16.100.100 80 {

weight 3

TCP\_CHECK {

connect\_timeout 10

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

real\_server 172.16.100.101 80 {

weight 3

TCP\_CHECK {

connect\_timeout 10

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

}

Lvs Backup配置文件：

! Configuration File for keepalived

global\_defs {

notification\_email {

mengtao10@163.com

}

notification\_email\_from mengtao10@163.com

smtp\_server 127.0.0.1

router\_id LVS\_DEVEL

}

vrrp\_instance VI\_1 {

state BACKUP

interface eth0

virtual\_router\_id 51

priority 99

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1q2w3e4r5t6y

}

virtual\_ipaddress {

172.16.100.200

}

}

virtual\_server 172.16.100.200 80 {

delay\_loop 6

lb\_algo wrr

lb\_kind DR

persistence\_timeout 60

protocol TCP

real\_server 172.16.100.100 80 {

weight 3

TCP\_CHECK {

connect\_timeout 10

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

real\_server 172.16.100.101 80 {

weight 3

TCP\_CHECK {

connect\_timeout 10

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

}

1. 访问验证：

1.任意关闭一台web服务器，服务都可以正常访问  
2.任意关闭一台lvs，服务都可以正常访问