# LVS操作手册

BY 王平安\_阿骨打 吴佳明\_普空

## 1 集群模式-配置（单台）

1. **安装系统和工具**

# 参见lvs-fullnat-synproxy.tar中的README, 包括 LVS kernel和keepalived等的编译方法；

1. **内核启动参数**

在kernel一行中，添加“**nohz=off** ”

注：如果不关闭nohz，大压力下CPU0可能会消耗过高，压力不均匀；

1. **Sysctl配置**

路径：/etc/sysctl.conf

# configure for lvs

net.ipv4.conf.all.arp\_ignore = 1

net.ipv4.conf.all.arp\_announce = 2

net.core.netdev\_max\_backlog = 500000

1. **配置网卡参数**

路径：/etc/rc.local

**关闭网卡LRO和GRO**

# ethtool -K eth0 gro off

# ethtool -K eth0 lro off

**绑定网卡中断**

# set\_irq\_affinity eth0 #脚本参见附录,该脚本是ixgbe/igb driver网卡

1. **关闭系统参数**

路径：/etc/rc.local

关闭irqbalance

# service irqbalance stop

# chkconfig --level 2345 irqbalance off

1. **LocalAddress配置**

路径：/etc/rc.local

Local address绑定到内网(下联)网卡上

ip addr add 192.168.104.1/32 dev eth1

ip addr add 192.168.104.2/32 dev eth1

ip addr add 192.168.104.3/32 dev eth1

ip addr add 192.168.104.4/32 dev eth1

ip addr add 192.168.104.5/32 dev eth1

1. **Zebra.conf**

路径：/etc/quagga/zebra.conf

启动方式：/usr/sbin/zebra -d -f /etc/quagga/zebra.conf

hostname lvs-route-4

password 8 123456

enable password 8 123456

log file /var/log/zebra.log

service password-encryption

1. **Ospf.conf**

路径：/etc/quagga/ospf.conf

启动方式：/usr/sbin/ospfd -d -f /etc/quagga/ospf.conf

hostname lvs-4-ospfd

password 8 123456

enable password 8 123456

log file /var/log/ospf.log

log stdout

log syslog

service password-encryption

interface p3p1 //上连网卡号

ip ospf message-digest-key 8 md5 123456

 ip ospf hello-interval 3
 ip ospf dead-interval 12

router ospf

ospf router-id 192.168.0.14 //route id配置为 上连网卡接口ip

log-adjacency-changes

auto-cost reference-bandwidth 1000

network 123.125.65.0/24 area 0.0.0.11 // VIP网段

network 192.168.0.12/30 area 0.0.0.11 // 上连IP网段

area 0.0.0.11 authentication message-digest

area 0.0.0.11 stub no-summary

1. **Keepalived.conf**

启动：service keepalived start

更新：service keepalived reload

停止：service keepalived stop

Keepalived的配置包含2个文件，以image-p业务为例：

说明：一个集群内的所有LVS配置文件基本相同，区别的地方见红色区域

* + 1. **主配置文件keepalived.conf**

路径：/etc/keepalived/keepalived.conf

! Configuration File for keepalived

global\_defs {

# notification\_email {

# hegen@taobao.com

# }

# notification\_email\_from hegen@taobao.com

# smtp\_server 192.168.200.1

# smtp\_connect\_timeout 40

}

local\_address\_group laddr\_g1 {

 192.168.104.1

 192.168.104.2

 192.168.104.3

 192.168.104.4

 192.168.104.5

}

! include virtual server configure file

include image-p.conf

* + 1. **业务配置文件 “业务名.conf”**

路径：/etc/keepalived/image-p.conf

virtual\_server\_group image-p {

 125.76.224.240 80 //vip1

 125.76.224.250 80 //vip2

}

!for image-p.taobao.com

virtual\_server group image-p {

 delay\_loop 7

lb\_algo rr

lb\_kind FNAT

protocol TCP

syn\_proxy

laddr\_group\_name laddr\_g1

 alpha //启动alpha模式，以便自动绑定vip

 omega // 启动omega模式，以便自动解除vip

 quorum 1

 hysteresis 0

 quorum\_up " ip addr add 125.76.224.240/32 dev lo; ip addr add 125.76.224.250/32 dev lo;"

 quorum\_down " ip addr del 125.76.224.240/32 dev lo; ip addr del 125.76.224.250/32 dev lo;"

 /\* healthcheck for L4 \*/

real\_server 125.76.224.12 80 {

 weight 100

inhibit\_on\_failure

 TCP\_CHECK {

 connect\_timeout 5

 }

}

/\* healthcheck for L7 \*/

 virtualhost cdn.hc.org

 real\_server 125.76.224.161 80 {

 weight 100

 inhibit\_on\_failure

 HTTP\_GET {

 url {

 path /status?SERVICE=haproxy&&VIP=125.76.224.240

 status\_code 200

 }

 connect\_timeout 3

 nb\_get\_retry 2

 delay\_before\_retry 5

 }

 }

}

1. **环境检查**
	* 1. **重要性高**

在LVS刚部署完毕，或者运维操作完毕时，都必须检查以下配置；

* 命令ip addr list，查看后端VIP是否绑定正确, 查看local address是否绑定正确
* 命令ipvsadm –ln，查看流量是否过来，各RS上流量是否均匀，流量大小是否符合预期；
* 命令ps aux | grep keepalived，查看keepalived进程个数是否正确
* 命令tcpdump –i any –nnn | grep OSPF， 查看ospf心跳是否正常
* 命令route –n, 查看ospf生成的路由是否正常
* 命令tail –n 1000 /var/log/message, 查看keepalived启动日志是否异常
	+ 1. **重要性低**

除了检查“3.1 重要性高”的点，还需要检查以下信息：

* 执行cat /proc/interrupts | grep ethx，其中ethx为万兆网卡，查看网卡中断是否被正确地绑定在N个核上；
* 在client上curl vip，在lvs上curl rs\_ip，查看能否curl通；

## 2 主备模式-配置（单台）

1. **安装系统和工具**

# 参见lvs-fullnat-synproxy.tar中的README，包括 LVS kernel和keepalived等的编译方法；

1. **内核启动参数**

在kernel一行中，添加“**nohz=off** ”

注：如果不关闭nohz，大压力下CPU0可能会消耗过高，压力不均匀；

1. **Sysctl配置**

路径：/etc/sysctl.conf

# configure for lvs

net.core.netdev\_max\_backlog = 500000

1. **配置网卡参数**

路径：/etc/rc.local

**关闭网卡LRO和GRO**

# ethtool -K p3p1 gro off

# ethtool -K p3p1 lro off

**绑定网卡中断**

# set\_irq\_affinity eth0 #脚本参见附录,该脚本是ixgbe/igb driver网卡

1. **关闭系统参数**

路径：/etc/rc.local

关闭irqbalance

# service irqbalance stop

# chkconfig --level 2345 irqbalance off

1. **LocalAddress配置**

路径：/etc/rc.local

Local address绑定到内网(下联)网卡上

ip addr add 192.168.104.1/32 dev eth1

ip addr add 192.168.104.2/32 dev eth1

ip addr add 192.168.104.3/32 dev eth1

ip addr add 192.168.104.4/32 dev eth1

ip addr add 192.168.104.5/32 dev eth1

1. **Keepalived.conf**

启动：service keepalived start

更新：service keepalived reload

停止：service keepalived stop

Keepalived的配置包含2个文件，以image-p业务为例：

说明：一个集群内的所有LVS配置文件基本相同，区别的地方见红色区域

* + 1. **主配置文件keepalived.conf**

路径：/etc/keepalived/keepalived.conf

! Configuration File for keepalived

global\_defs {

# notification\_email {

# hegen@taobao.com

# }

# notification\_email\_from hegen@taobao.com

# smtp\_server 192.168.200.1

# smtp\_connect\_timeout 40

}

local\_address\_group laddr\_g1 {

 192.168.104.1

 192.168.104.2

 192.168.104.3

 192.168.104.4

 192.168.104.5

}

! include virtual server configure file

include image-p.conf

* + 1. **业务配置文件 “业务名.conf”**

路径：/etc/keepalived/image-p.conf

virtual\_server\_group image-p {

 125.76.224.240 80 //vip1

 125.76.224.250 80 //vip2

}

vrrp\_instance VI\_1 {

 state MASTER/BACKUP

 interface eth0

 virtual\_router\_id 200

 priority 150/90

 advert\_int 1

 authentication {

 auth\_type PASS

 auth\_pass 1111

 }

 virtual\_ipaddress {

 125.76.224.240

 125.76.224.250

 }

}

!for image-p.taobao.com

virtual\_server group image-p {

 delay\_loop 7

lb\_algo rr

lb\_kind FNAT

protocol TCP

syn\_proxy

laddr\_group\_name laddr\_g1

 /\* healthcheck for L4 \*/

real\_server 125.76.224.12 80 {

 weight 100

inhibit\_on\_failure

 TCP\_CHECK {

 connect\_timeout 5

 }

}

/\* healthcheck for L7 \*/

 virtualhost cdn.hc.org

 real\_server 125.76.224.161 80 {

 weight 100

 inhibit\_on\_failure

 HTTP\_GET {

 url {

 path /status?SERVICE=haproxy&&VIP=125.76.224.240

 status\_code 200

 }

 connect\_timeout 3

 nb\_get\_retry 2

 delay\_before\_retry 5

 }

 }

}

## 3 RS配置

1. **安装系统**

# 参见lvs-fullnat-synproxy.tar中的README,其中有TOA(RS内核)的patch；

1. **加载TOA模块，命令：# modprobe toa**

# vim /etc/rc.local

添加 modproble toa

## 4日常操作（以集群模式为例）

### 4.1 添加/删除realserver

如果添加，请确保realserver的监听的port是打开的（可以telnet连接该端口）。

1. 第1步，配置realserver，具体参见附录5.1；

更新内核版本，加载相应的TOA模块：

# modprobe ttm.ko

# vim /etc/rc.local

添加 modproble toa

1. 第2步，修改keepalived的配置，注意**所有LVS**上都得修改；

例如realserver的IP为10.251.X.X，业务名image-p

# vim /etc/keepalived/image-p.conf

virtual\_server 220.181.5.193 80 {

 ……

 real\_server 10.251.X.X 80 {

 weight 1

 TCP\_CHECK {

 connect\_timeout 5

 }

 }

}

1. 第3步，发送HUP信号给keepalived，使配置修改生效；

# service keepalived reload

1. 第4步，检查realserver是否操作成功

在LVS上，分别运行ipvsadm -ln观察该realserver的健康检查是否成功，并在LVS查看session分配是否均匀。

## 4.2添加/删除vip

假设新添vip为220.181.5.194，业务名称image-p；

1. 修改keepalived配置文件，添加如下内容；

第一步，创建业务配置文件；

#vim /etc/keepalived/image-p.conf

virtual\_server 220.181.5.194 80 {

 delay\_loop 6

lb\_algo rr

lb\_kind FNAT

protocol TCP

syn\_proxy

laddr\_group\_name laddr\_g1

alpha //启动alpha模式，以便自动绑定vip

quorum 1

hysteresis 0

quorum\_up " ip addr add 220.181.5.194/32 dev lo;"

quorum\_down " ip addr del 220.181.5.194/32 dev lo;"

 ……

}

第二步，修改keepalived配置文件；

#vim /etc/keepalived/keepalived.conf

……

! include virtual server configure file

include www.conf

include image-p.conf

1. 发送HUP信号给keepalived，使配置修改生效；

# service keepalived reload

1. 检查vip配置是否生效；

# ipvsadm –ln //查看vip是否已经配置到lvs中

# ip addr list //查看lo上vip是否绑定成功

# 模拟用户访问vip，结果是否正确

## 4.3 添加/删除local address

注：local address和内网接口ip绝对不能重合；

以添加/删除192.168.104.4为例，其内网网卡为p3p1，需要配置2个地方：

1. 修改/etc/rc.local

添加ip addr add 192.168.104.4/32 dev p3p1

1. 修改/etc/keepalived/keepalived.conf

#vim /etc/keepalived/keepalived.conf

local\_address\_group laddr\_g1 {

 ……

 192.168.104.4

}

1. 发送HUP信号给keepalived，使配置修改生效；

# service keepalived reload

1. 检查local address配置是否生效；

# ip addr list //查看网卡上是否已经绑定ip

# ipvsadm –G //查看vip上是否已经绑定ip

## 5 set\_irq\_affinity脚本(源自intel 82599 driver)

# setting up irq affinity according to /proc/interrupts

# 2008-11-25 Robert Olsson

# 2009-02-19 updated by Jesse Brandeburg

#

# > Dave Miller:

# (To get consistent naming in /proc/interrups)

# I would suggest that people use something like:

# char buf[IFNAMSIZ+6];

#

# sprintf(buf, "%s-%s-%d",

# netdev->name,

# (RX\_INTERRUPT ? "rx" : "tx"),

# queue->index);

#

# Assuming a device with two RX and TX queues.

# This script will assign:

#

# eth0-rx-0 CPU0

# eth0-rx-1 CPU1

# eth0-tx-0 CPU0

# eth0-tx-1 CPU1

#

set\_affinity()

{

 if [ $VEC -ge 32 ]

 then

 MASK\_FILL=""

 MASK\_ZERO="00000000"

 let "IDX = $VEC / 32"

 for ((i=1; i<=$IDX;i++))

 do

 MASK\_FILL="${MASK\_FILL},${MASK\_ZERO}"

 done

 let "VEC -= 32 \* $IDX"

 MASK\_TMP=$((1<<$VEC))

 MASK=`printf "%X%s" $MASK\_TMP $MASK\_FILL`

 else

 MASK\_TMP=$((1<<$VEC))

 MASK=`printf "%X" $MASK\_TMP`

 fi

 printf "%s mask=%s for /proc/irq/%d/smp\_affinity\n" $DEV $MASK $IRQ

 printf "%s" $MASK > /proc/irq/$IRQ/smp\_affinity

}

if [ "$1" = "" ] ; then

 echo "Description:"

 echo " This script attempts to bind each queue of a multi-queue NIC"

 echo " to the same numbered core, ie tx0|rx0 --> cpu0, tx1|rx1 --> cpu1"

 echo "usage:"

 echo " $0 eth0 [eth1 eth2 eth3]"

fi

# check for irqbalance running

IRQBALANCE\_ON=`ps ax | grep -v grep | grep -q irqbalance; echo $?`

if [ "$IRQBALANCE\_ON" == "0" ] ; then

 echo " WARNING: irqbalance is running and will"

 echo " likely override this script's affinitization."

 echo " Please stop the irqbalance service and/or execute"

 echo " 'killall irqbalance'"

fi

#

# Set up the desired devices.

#

for DEV in $\*

do

 for DIR in rx tx TxRx

 do

 MAX=`grep $DEV-$DIR /proc/interrupts | wc -l`

 if [ "$MAX" == "0" ] ; then

 MAX=`egrep -i "$DEV:.\*$DIR" /proc/interrupts | wc -l`

 fi

 if [ "$MAX" == "0" ] ; then

 echo no $DIR vectors found on $DEV

 continue

 fi

 for VEC in `seq 0 1 $MAX`

 do

 IRQ=`cat /proc/interrupts | grep -i $DEV-$DIR-$VEC"$" | cut -d: -f1 | sed "s/ //g"`

 if [ -n "$IRQ" ]; then

 set\_affinity

 else

 IRQ=`cat /proc/interrupts | egrep -i $DEV:v$VEC-$DIR"$" | cut -d: -f1 | sed "s/ //g"`

 if [ -n "$IRQ" ]; then

 set\_affinity

 fi

 fi

 done

 done

done