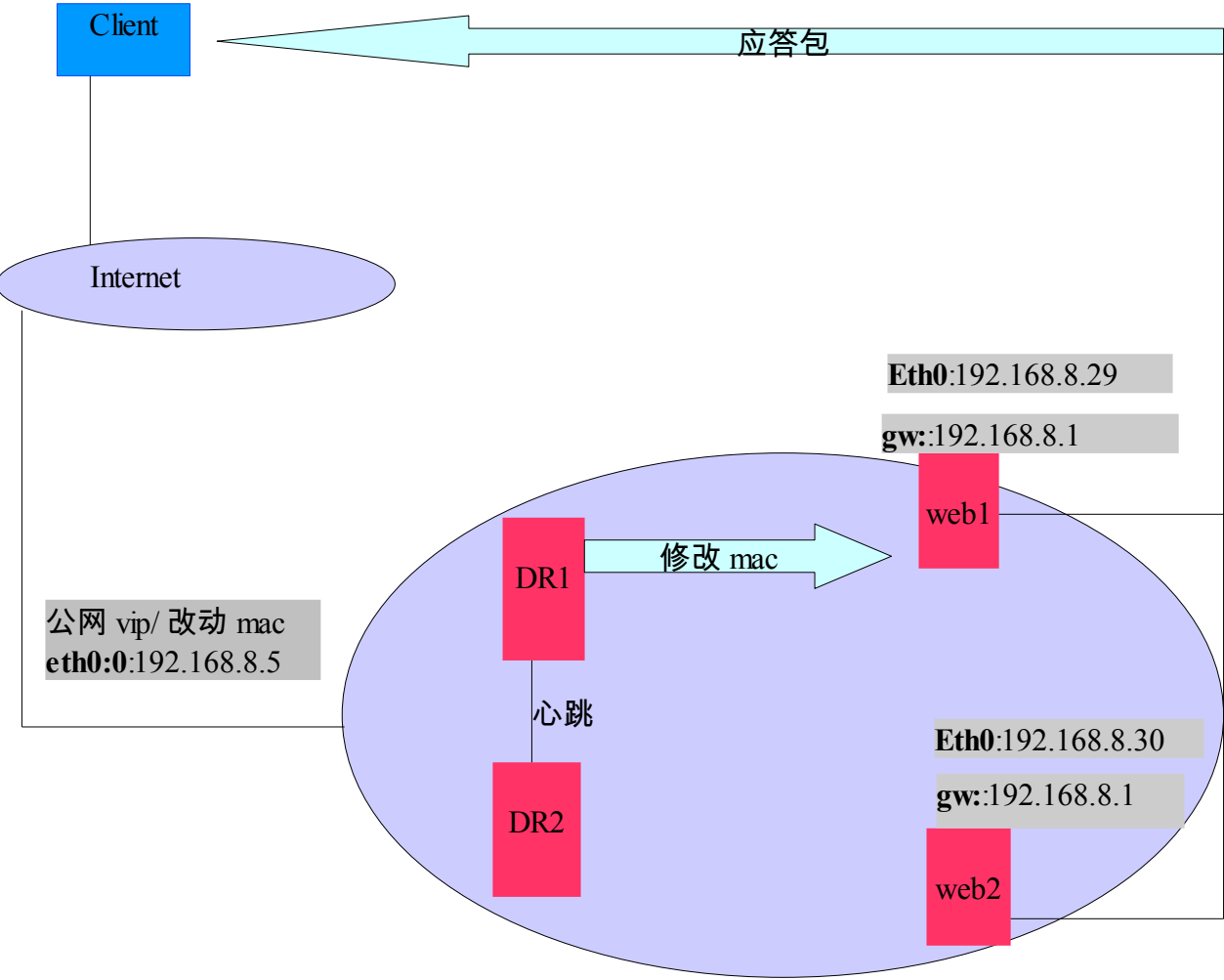


一. 结构图原理

注: 用 192. 168. 8. 模拟公网

图一



vip
eth0:0:192. 168. 8. 5

DR1 物理 nic 接口与 ip 分配如下:

Eth0:192.168.8.3/24 gw:192.168.8.1
Eth1:192.168.200.13/24 gw:为空, 用于心跳检测

DR2 物理 nic 接口与 ip 分配如下:

Eth0:192.168.8.4/24 gw:192.168.8.1
Eth1:192.168.200.14/24 gw:为空, 用于心跳检测

二. 目的:

1. 实现 Server 集群服务的高可用性, 做到 Super node 零故障转移, 同时对 realserver 的实时监控.

三. 测试环境 与 架构要求

1. 在 VMware Server 1.0.4 上模拟 lvs-dr+heartbeat+ldirectord 架构
2. vmwareServer 上安装四台 guest
3. 各 guest 机器名分别为:

DR1(主 director)两块网卡 eth0 接公网 eth1 用于连接心跳

DR2(备用 director)两块网卡 eth0 接公网 eth1 用于连接心跳

web1 单网卡 eth0 连接公网

web2 单网卡 eth0 连接公网

四. DR1(主 Director)配置

4.1 DR1 系统环境预设置, 关闭 ip 转发 (1 on, 0 off), 同时开启 icmp 响应, (并设置相关 *. send_redirects 为 1)

```
echo "0">/proc/sys/net/ipv4/ip_forward
echo "1">/proc/sys/net/ipv4/conf/all/send_redirects
echo "1">/proc/sys/net/ipv4/conf/default/send_redirects
echo "1">/proc/sys/net/ipv4/conf/eth0/send_redirects
```

4.2 配置 公网 ip 及心跳接口 ip

注: 浮动 Ip 即: vip 192.168.8.5 是不写入系统网口配置文件的, 它一般由 HA 软件来控制, 所以下面这两行不用写入系统

```
#ifconfig eth0:0 192.168.8.5 netmask 255.255.255.255 broadcast 192.168.8.5 up
```

```
#route add -host 192.169.8.5 eth0:0
```

eth0 配置

```
[[root@DR1 ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth0
```

```
DEVICE=eth0
```

```
BOOTPROTO=static
```

```
HWADDR=00:0C:29:EE:EB:10
```

```
ONBOOT=yes
```

```
TYPE=Ethernet
```

```
IPADDR=192.168.8.3
```

```
NETMASK=255.255.255.0
```

```
GATEWAY=192.168.8.1
```

```
[root@DR1 ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth1
```

```
# Advanced Micro Devices [AMD] 79c970 [PCnet32 LANCE]
```

```
DEVICE=eth1
```

```
ONBOOT=yes
```

```
BOOTPROTO=static
```

```
IPADDR=192.168.200.13
```

```
NETMASK=255.255.255.0
```

```
HWADDR=00:0C:29:EE:EB:1A
```

4.3 将 director 各自的主机名与 ip 对应都写入/etc/hosts 文件

```
[root@DR1 ha.d]# cat /etc/hosts
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1          DR1 localhost.localdomain localhost
192.168.8.3  DR1
192.168.8.4  DR2
```

4.4 所需软件列表用 yum 安装:

```
1. yum -y install ipvsadm      //用于 lvs-dr 架构
2. yum -y install libnet
3. yum -y install heartbeat-*  //用于 director HA, 可以得 heartbeat 所需的工具包:
```

```
ipvsadm-1.24-6
heartbeat-stonith-2.1.2-3.el4.centos
heartbeat-gui-2.1.2-3.el4.centos
heartbeat-pils-2.1.2-3.el4.centos
heartbeat-lldirectord-2.1.2-3.el4.centos  //用于监控 realServer 的运行状况
heartbeat-2.1.2-3.el4.centos
```

4.5 配置 heartbeat, 用到/etc/ha.d/目录下的 4 个配置文件

```
(1)authkeys      //认证文件, 里面有不同的高效验算法让你选, 这个文件的权限必须是 600
(2)ha.cf          //heartbeat 的主要配置文件, 可以对 heartbeat 的多数性能和状态进行
                  配置。大部分选项的取值可以采用默
(3)haresources    //此文件是节点调用的资源文件如可 heartbeat 去监管/etc/init.d/
                  或 /etc/ha.d/resource.d 目录下的脚本
(4)lldirectord.cf //监管 RealServer 的运行状况
```

4.6 将相关配置文件复制到/etc/ha.d 目录下

```
cp /usr/share/doc/heartbeat-2.1.2/ha.cf          /etc/ha.d/ha.cf
cp /usr/share/doc/heartbeat-2.1.2/authkeys        /etc/ha.d/authkeys
cp /usr/share/doc/heartbeat-2.1.2/haresources      /etc/ha.d/haresources
cp /usr/share/doc/heartbeat-lldirectord-2.1.2/lldirectord.cf  /etc/ha.d/lldirectord.cf
```

4.7 开始配置 authkeys 文件

```
chmod 600 /etc/ha.d/authkeys
vi /etc/ha.d/authkeys 内容如下:
auth 1                //打开 auth 并选择一个用于主从节点的高效验算法
1 crc                 //本例中使用 crc 算法
#2 sha1 HI!
#3 md5 Hello!
```

4.8 配置 ha.cf 文件

```
[root@DR1 ha.d]# cat ha.cf
debugfile /var/log/ha-debug
#crm yes
logfile /var/log/ha-log
```

```

logfacility      local0
keepalive 2
deadtime 20
warntime 10
initdead 40
udpport 694
bcast eth1      # Linux
auto_failback on
#watchdog /dev/watchdog
node    DR1
node    DR2
ping_group group1 192.168.8.3 192.168.8.4
apiauth ipfail gid=root uid=root
#use_logd yes/no
respawn root /usr/lib/heartbeat/ipfail

```

4.9 配置 haresources 文件

```
[root@DR1 ha.d]# cat haresources
```

增加两行

```
DR1 192.168.8.5/32/eth0:0 ipvsadm ldirectord::ldirectord.cf
```

注:

ldirectord 监管 realserver, 在 haresources 文件中是以 **ldirectord::配置文件**, 的形式出现, 所以::符号后跟的配置文件名是可以任意的, 只要它在/etc/ha.d/目录下存在!

4.10 配置 ldirectord.cf 文件

```

[root@DR1 ha.d]# cat ldirectord.cf
checktimeout=3
checkinterval=2
autoreload=yes
quiescent=yes
logfile="/var/log/ldirectord.log"
virtual=192.168.8.5:80 eth0:0
    real=192.168.8.29:80 gate 100
    real=192.168.8.30:80 gate 100
    fallback=127.0.0.1:80
    request="index.html"
    receive="TestPage"
    protocol=tcp
    checktype=negotiate
    checkport=80
    scheduler=wrr

```

关于 heartbeat 配置完成!

4.11 配置 ipvsadm

```

[root@DR1 ha.d]# cat /etc/sysconfig/ipvsadm
# tcp tcpfin udp timeout values(900 120 300)
ipvsadm -A -t 192.168.8.5:80 -s wrr
ipvsadm -a -t 192.168.8.5:80 -r 192.168.8.29:80 -g -w 100
ipvsadm -a -t 192.168.8.5:80 -r 192.168.8.30:80 -g -w 100

```

4.12#为了能在系统启动时, 相关服务也启动起来, 我们将命令写入/etc/rc.local 文件

```

[root@DR1 ha.d]# vi /etc/rc.local
echo "0">/proc/sys/net/ipv4/ip_forward
echo "1">/proc/sys/net/ipv4/conf/all/send_redirects
echo "1">/proc/sys/net/ipv4/conf/default/send_redirects
echo "1">/proc/sys/net/ipv4/conf/eth0/send_redirects
/etc/init.d/heartbeat start &

```

到此主 lvs-dr + heartbeat+ldirectord 配置完成

五 DR2(备用 director)配置:

注: DR2 除了如下 ip 与 DR1 配置不同之外, 其它所有的配置和 DR1 要一样

5. DR2 公网 ip 私网 ip 及心跳接口 ip

```
[root@DR2 network-scripts]# cat ifcfg-eth0
# Advanced Micro Devices [AMD] 79c970 [PCnet32 LANCE]
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=static
IPADDR=192.168.8.4
NETMASK=255.255.255.0
GATEWAY=192.168.8.1
```

```
[root@DR2 network-scripts]# cat ifcfg-eth1
# Advanced Micro Devices [AMD] 79c970 [PCnet32 LANCE]
DEVICE=eth1
ONBOOT=yes
BOOTPROTO=static
IPADDR=192.168.200.14
NETMASK=255.255.255.0
HWADDR=00:0C:29:89:7A:26
```

六. 两个 RealServer Web1, Web2 的配置

web1 和 web2 都采用 rpm 包安装版为 httpd-2.0.52 在两台 webserver 上都运行
yum -y install httpd 命令来安装 apache

6.1 配置 web1 在真实服务器上关闭 arp 响应

```
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
```

#vi /etc/sysctl.conf 内容如下

```
net.ipv4.ip_forward = 0
net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.default.accept_source_route = 0
kernel.sysrq = 0
kernel.core_uses_pid = 1
net.ipv4.conf.all.send_redirects=1
net.ipv4.conf.default.send_redirects=1
net.ipv4.conf.eth0.send_redirects=1
net.ipv4.conf.lo.send_redirects=1
```

再运行命令 sysctl -p

6.2 Web1 vip 配置

```
ifconfig lo:0 192.168.8.5 netmask 255.255.255.255 broadcast 192.168.8.5 up
route add -host 192.168.8.5 lo:0
```

6.3 web1 eth0 接口配置

```
[root@web1 network-scripts]# cat ifcfg-eth0
# Advanced Micro Devices [AMD] 79c970 [PCnet32 LANCE]
DEVICE=eth0
```

```
ONBOOT=yes
BOOTPROTO=static
IPADDR=192.168.8.29
NETMASK=255.255.255.0
GATEWAY=192.168.8.1
```

6.4 在 Real Server1 即 Web1 中添加监控页:

```
echo "TestPage1111111" >> /var/www/html/index.html
```

6.5 配置 web2, web2 的设定与 web1 配置方法相同!

6.6 配置 **web2** 在真实服务器上关闭 arp 响应

```
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
```

#vi /etc/sysctl.conf 内容如下

```
net.ipv4.ip_forward = 0
net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.default.accept_source_route = 0
kernel.sysrq = 0
kernel.core_uses_pid = 1
net.ipv4.conf.all.send_redirects=1
net.ipv4.conf.default.send_redirects=1
net.ipv4.conf.eth0.send_redirects=1
net.ipv4.conf.lo.send_redirects=1
```

再运行命令 sysctl -p

6.7 web2 vip 配置

```
ifconfig lo:0 192.168.8.5 netmask 255.255.255.255 broadcast 192.168.8.5 up
route add -host 192.168.8.5 lo:0
```

6.8 web2 eth0 接口配置

```
[root@web2 network-scripts]# cat ifcfg-eth0
# Advanced Micro Devices [AMD] 79c970 [PCnet32 LANCE]
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=static
IPADDR=192.168.8.30
NETMASK=255.255.255.0
GATEWAY=192.168.8.1
```

6.9 在 Real Server2 即 Web2 中添加监控页:

```
echo "TestPage2222" >> /var/www/html/index.html
```

七. 测试 heartbeat HA 效果的步骤:

1. 在主 DR1 和 DR2 都正常工作的情况下, 将 DR1 机器关闭或将 heartbeat 服务关闭, 关闭用如下命令/etc/init.d/heartbeat stop 或 service heartbeat stop
2. 去 DR2 上用 ifconfig 看一下应该看到 eth0:0 接口自动启用起来了, 而 DR1 上的 eth0:0 接口被自动关闭了, 说明 DR2 成功将 DR1 的工作接管过来了
3. 再将 DR1 开启 heartbeat 服务, 这时 DR1 上的 eth0:0 接口自动起来了, 而 DR2 上 eth0:0 接口也自动关闭了, 说明 DR1 故障恢复之后成功将之前 DR2 接管的工作又重新接任下来了。
4. 以上测试通过, 从而达到了 HA 真正的效果

八. 测试 ldirectord 监管 RealServer 效果的步骤:

1. 在 DR1 DR2 web1 web2 都正常运作的环境中, 先将 web1 或 web2 的 http 服务关闭, 然到调度 lvs 的 director 机器上观察 lvs 的调度情况, 可用命令:

`watch -n 1 ipvsadm -lnc` 动态查看

当然也可以在 client 端去访问 vip, 看此时访问的结果, 可用以下方法:

`http://192.168.8.5`