


原创

lvs_dr 负载均衡模式分析

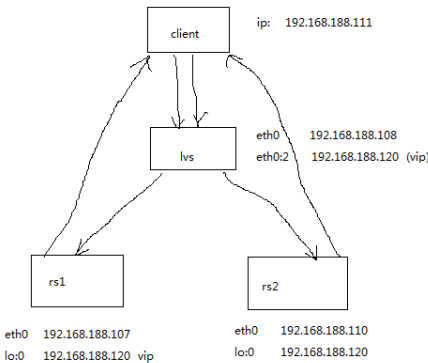
 hello_cjq 关注

2018-03-28 11:54:50 5842人阅读 0人评论

1.前言

上一篇文章《lvs_nat 负载均衡模式及抓包分析》，已经对开源负载均衡软件的 nat 模式进行了实验和 tcpdump 数据包分析。经过分析，我们知道 lvs 的 nat 负载均衡模式，它的性能瓶颈在 lvs 调度器。因为网络上的客户端请求连接和后端服务的响应数据都要经过 lvs 调度器，所以 lvs 调度器在大请求量的情况，就容易出现瓶颈。所以，在这篇文章，对 lvs 负载均衡的另一种架构 dr 模式进行分析，dr 模式它最大的特点就是，负载均衡集群的后端服务响应，直接从后端服务发回客户端，也就是说返回数据不需经过 lvs 调度器。大大减轻了集群的调度器的负载。

2. lvs_dr 模式的架构图



lvs_dr 模式架构图

说明：图中展示的是最简单的 lvs_dr 模式架构，也是这篇博客的实验环境。

3. arp 基础知识扫盲

为什么要先了解arp 的知识呢？因为，在lvs_dr 负载均衡模式中，realserver 和 lvs 调度器都配置了vip。客户端需要和 lvs 的 vip 通信，所以，就要抑制realserver 对vip 的mac 地址的请求。具体会涉及到 arp_ignore 和 arp_announce 两个参数的设置。注意在文章的后半段，我会再详细讲一下。

什么是arp广播？通俗讲，就是在网络中，根据ip地址找mac地址的协议，即ARP 协议。

3.1 举一个例子：

当主机A要与主机B通信时：

第1步：根据主机A上的路由表内容，IP确定用于访问主机B的转发IP地址是192.168.1.2。然后A主机在自己的本地ARP缓存中检查主机B的匹配MAC地址。

第2步：如果主机A在ARP缓存中没有找到映射，它将询问192.168.1.2的硬件地址，从而将ARP请求帧广播到本地网络上的所有主机。源主机A的IP地址和MAC地址都包括在ARP请求中。本地网络上的每台主机都接收到ARP请求并且检查是否与自己的IP地址匹配。如果主机发现请求的IP地址与自己的IP地址不匹配，它将丢弃ARP请求。

第3步：主机B确定ARP请求中的IP地址与自己的IP地址匹配，则将主机A的IP地址和MAC地址映射添加到本地ARP缓存中。

第4步：主机B将包含其MAC地址的ARP回复消息直接发送回主机A。

第5步：当主机A收到从主机B发来的ARP回复消息时，会用主机B的IP和MAC地址映射更新ARP缓存。本机缓存是有生存期的，生存期结束后，将再次重复上面的过程。主机B的MAC地址一旦确定，主机A就能向主机B发送IP通信了。

4. 准备工作——服务器的ip分配

4.1 lvs

```
eth0:      192.168.188.108
eth0:2     192.168.188.120    (vip)
mac:       00:0c:29:50:d5:63
```

4.2 realeserver

```
第一台 (cenvm71) :
eno16777736 :      192.168.188.107
mac :              00:0c:29:e4:4d:1f
lo:0              192.168.188.120    (vip)

第二台 (cenvm72) :
eth0:      192.168.188.110
mac:       00:0c:29:2c:a5:a0
lo:0       192.168.188.120    (vip)
```



4.3 在两台 rs 主机安装nginx

```
yum install -y nginx
```

安装完nginx 服务后，最好将nginx 的默认html 文件修改一下输出内容，让它们能够区分就可以了。

5. 搭建过程

5.1 DR 安装ipvsadm 和配置 lvs_dr.sh 文件

```
安装ipvsadm 软件：
yum install -y ipvsadm

配置文件：
[root 18:40:39 @CentOS3 sbin] cat /usr/local/sbin/lvs_dr.sh
#!/bin/bash
echo 1 > /proc/sys/net/ipv4/ip_forward
ipvs=/sbin/ipvsadm
vip=192.168.188.120
rs1=192.168.188.107
rs2=192.168.188.110
#注意这里的网卡名字
ifdown eth0
ifup eth0
ifconfig eth0:2 $vip broadcast $vip netmask 255.255.255.255 up
route add -host $vip dev eth0:2
```

在线
客服

```
$ipvs -a -t $vip:80 -r $rs1:80 -g -w 1
$ipvs -a -t $vip:80 -r $rs2:80 -g -w 1
```

5.2 realeserver 配置 lvs_rs.sh 文件

```
[root@cenvm72 network-scripts]# cat /usr/local/sbin/lvs_rs.sh
#!/bin/bash
vip=192.168.188.120
#把vip绑定在lo上，是为了实现rs直接把结果返回给客户端
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letrs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
```

两台realeserver 主机的 lvs_rs.sh 配置文件都一样

6. 测试过程

6.1 启动 lvs

现在 rs1 和 rs2 上运行 lvs_rs.sh 脚本，启动nginx

```
/bin/bash /usr/local/sbin/lvs_rs.sh

systemctl start nginx
```

然后，在 lvs 上运行lvs_dr.sh 脚本，启动nginx

```
/bin/bash /usr/local/sbin/lvs_dr.sh

systemctl start nginx
```



在 lvs 查看：

```
[root 18:58:10 @CentOS3 sbin] ipvsadm -ln
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
-> RemoteAddress:Port Forward Weight ActiveConn InActConn
TCP 192.168.188.120:80 wrr
-> 192.168.188.107:80 Route 1 0 0
-> 192.168.188.110:80 Route 1 0 0
```

说明lvs 功能已经启动

6.2 lvs dr 模式请求过程

整个请求过程如下：

client在发起请求之前，会发一个arp广播的包，在网络中找“谁是vip”，由于所有的服务器，lvs和rs都有vip，为了让client的请求送到lvs上，所以必须让rs不能响应client发出的arp请求，（这也是为什么要禁止rs上arp的请求和响应）下面就是lvs转发的事情了：

1. client向目标vip发送请求，lvs接收；此时ip包和数据信息如下：

src mac	dst mac	src ip	dst ip
00:0c:29:f3:1f:de	00:0c:29:50:d5:63	192.168.188.111	192.168.188.120

2. lvs根据负载均衡的算法，选择一台realserver，然后把realserver1的mac地址作为目的mac地址，发送到局域网中

在线客服

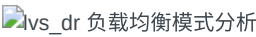
src mac	dst mac	src ip	dst ip
00:0c:29:50:d5:63	00:0c:29:e4:4d:1f	192.168.188.111	192.168.188.120

3. realserver1在局域网中收到这个请求以后，发现目的ip和本地匹配，于是进行处理，处理完成以后，直接把源ip和目的ip直接对调，然后经过网关直接返回给用户；

src mac	dst mac	src ip	dst ip
00:0c:29:e4:4d:1f	00:0c:29:f3:1f:de	192.168.188.120	192.168.188.111

6.3 抓包分析验证

1. lvs 转发到 realeserver



从抓包中可以看到，客户端发送请求给 lvs，lvs 马上就会将请求转发给后端的 rs2 (mac 地址：00:0c:29:2c:a5:a0)；

2. realserver 直接会应客户端数据

```
[root@chenw72 network-scripts]# tcpdump -i eth0 -e -nn 'dst host 192.168.188.111 and !port 22 and !arp'
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 25544 bytes
11:12:25.484846 00:0c:29:2c:a5:a0 > 00:0c:29:f3:1f:de, ethertype IPv4 (0x0800), length 74: 192.168.188.120.80 > 192.168.188.111.42960: Flags [S.], seq 3134128613, ack 4124450911, win 28960, options [mss 1460,sackOK,TS val 82857875
ecr 82582043,nop,wscale 9], length 0
11:12:25.688374 00:0c:29:2c:a5:a0 > 00:0c:29:f3:1f:de, ethertype IPv4 (0x0800), length 66: 192.168.188.120.80 > 192.168.188.111.42960: Flags [T.], ack 209, win 939, options [nop,nop,TS val 82857875 ecr 82582043], length 0
11:12:25.688604 00:0c:29:2c:a5:a0 > 00:0c:29:f3:1f:de, ethertype IPv4 (0x0800), length 368: 192.168.188.120.80 > 192.168.188.111.42960: Flags [P.], seq 1:243, ack 209, win 939, options [nop,nop,TS val 82857877 ecr 82582043], lengt
h 243: HTTP/1.1 200 OK
11:12:25.699012 00:0c:29:2c:a5:a0 > 00:0c:29:f3:1f:de, ethertype IPv4 (0x0800), length 66: 192.168.188.120.80 > 192.168.188.111.42960: Flags [F.], seq 243, ack 210, win 939, options [nop,nop,TS val 82857889 ecr 82582045], length 0
```

从抓到的数据包分析，rs2 处理请求后，直接就把数据发回给客户端了。源ip 是 vip，目标ip 是客户端的 ip。

7. 重要的补充，关于arp 抑制

lvs dr 模式中，lvs 和 rs 处于同一个网络中，而且他们都配置相同的 vip。所以，当客户端要向vip 发送网络请求的时候，它会先在整个网络广播一条 arp 请求，询问 vip 对应的mac 地址是什么。arp 广播有可能被lvs 响应，也有可能被realeserver 响应。如果这条请求 vip 的广播被realeserver 服务器响应了，那么客户端的 arp 缓存表就记录了vip 的mac 地址是realserver 的了。这样会导致客户端直接可以将数据请求发送到realserser ，lvs 的负载均衡作用就完全失效，其他的realserver 服务器也不会被请求。



由于存在以上的问题，lvs dr 模式的架构中，需要将realserver 的arp 协议响应和宣告功能进行限制。具体来说就是在 rs 里的 lvs_rs.sh 配置脚本的以下几条操作：

```
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
```

说明：

7.1 arp_ignore =1

作用就是，限制rs 对arp 广播的响应。当 arp 请求的目的 ip 是本机的网络入口设备的ip 时才响应。所以，当arp_ignore设为1后，rs 对网络中询问vip 的 arp 广播包都不再响应。因为rs 的vip 设置在lo:0 虚拟网卡上，不是rs这台机器的网络流入设备。

7.2 arp_announce = 2

作用就是，限制rs 在对外宣告arp 广播时所使用的源ip 地址。因为，rs 要直接返回客户的请求数据。我们从本文的第6节抓包分析里，就知道rs 需要知道客户机的mac地址，数据包才能发送到网络中，且能准确找到客户机的网卡。

rs 在广播arp 请求时，默认如果arp_announce=0时，发出请求的ip是什么，arp 请求里的源ip就应该是什。即如果arp_announce=0，那么rs 的arp 广播的源ip 就会是vip。当rs 发出这个源ip是vip ，源mac地址是 eth0 的请求包后，客户机就会更新自己的arp 缓存表里vip 的mac 地址，这样之前记录的lvs 的 mac 地址就失效了

应自己mac 地址。这样，客户机也不会更新自己arp 缓存表里的vip 的mac地址。
arp 协议有一个特点，它不会记住自己询问过的ip地址和mac主机。所以，当有接收到新的arp广播请求，如果发现新的mac 地址，他就会更新。而不会验证发送方是不是自己曾经询问过的ip。这个缺陷也引起了arp ***，就是我们常说的arp 欺骗。有些中间代理机器，不断地发arp 请求，让你的机器arp 缓存表里的mac 地址混乱。

8. 总结

在lvs dr 模式中，通过抓包理解请求发送的流转方向，清晰地理解为什么lvs 调度器不会成为网络性能的瓶颈。在理解arp 抑制时，理解arp_ignore 和 arp_announce 两个参数的作用，最为重要。

©著作权归作者所有：来自51CTO博客作者hello_cjq的原创作品，如需转载，请注明出处，否则将追究法律责任

lvsdr模式

4收藏分享

上一篇：虚拟机如何添加一块新的网卡并开启... 下一篇：lvs+keepalived 高...



hello_cjq


55篇文章，64W+人气，3粉丝

关注



提问和评论都可以，用心的回复会被更多人看到和认可

Ctrl+Enter 发布取消发布



推荐专栏更多



基于Python的DevOps实战
自动化运维开发新概念
共20章 | 抚琴煮酒
¥51.00 377人订阅



全局视角看大型园区网
路由交换+安全+无线+优化+运维
共40章 | 51CTO夏杰
¥51.00 1174人订阅




网工2.0晋级攻略——零基础入门Python/A...
网络工程师2.0进阶指南
共20章 | 姜斗鹏讲

订阅

在线

客服

42分享

hello_cjq

关注



负载均衡高手炼成记

高并发架构之路

共15章 | sery

¥ 51.00 465人订阅

订 阅



带你玩转高可用

前百度高级工程师的架构高可用实战

共15章 | 曹林华

¥ 51.00 440人订阅

订 阅

猜你喜欢

Mysql 在线新建或重做主从

下载豆，了解它，收获它【51CTO下载中心帮助】

UML建模之时序图（Sequence Diagram）

Nginx 反向代理、负载均衡、页面缓存、URL重写及读...

随手分享资料链接，坐等下载豆惊喜！【51CTO下载中...

响应式Spring的道法术器（Spring WebFlux 快速上手 + ...

Linux自定义快捷工具

Python脚本修改阿里云的访问控制列表

kubernetes之kubeadm最佳实践

亲测LNMP 的总体基本框架服务器的安装搭建

shell 练习(13) —— 监控 httpd 进程数是否异常

思科路由交换部分命令大全。

Hystrix 分布式系统限流、降级、熔断框架

分布式消息队列RocketMQ部署与监控

FTP主动模式和被动模式的比较

Ubuntu环境下挂载新硬盘

记一次线上DPDK-LVS的故障排查

记一次混合云API暴露的反思

rsync基本操作与安装

一次线上zabbix server 挂掉的思考



在线
客服