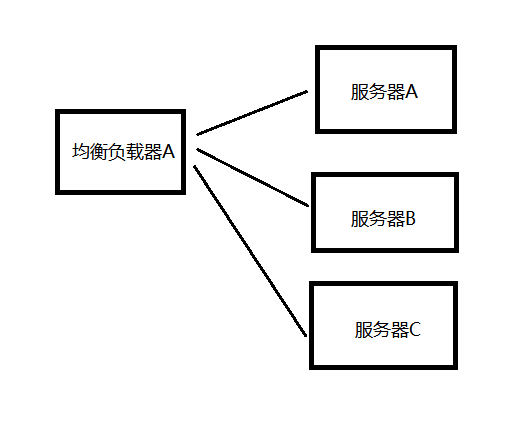
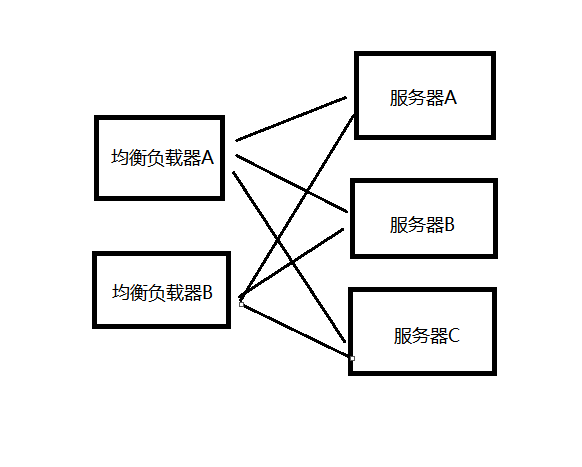
**LVS介绍**

我们熟知的均衡负载（如nginx、kong等）都是1对多，如



如果均衡负载器A出现问题，那么将导致整个系统都不可用，LVS中加入了均衡负载器B，其作用为在均衡负载器A出现问题时均衡负载器B将接收A的数据，均衡负载到后端服务器，在负载器A正常运行时，负载器B作为备选服务器可以不运行，也可以加入到均衡负载中（前提是要给B一个独立的IP）



**LVS负载模式**

先了解下如下术语

DS：LVS均衡负载服务器

RS：后端服务器

VIP：DS暴露的虚拟IP

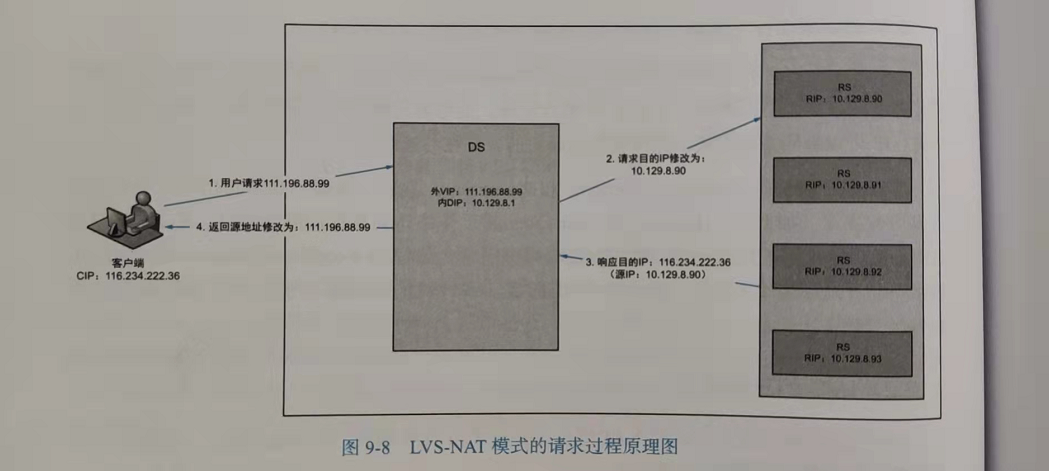
DIP：DS和后端服务器通信的IP

RIP：后端服务器的IP

CIP：客户端IP

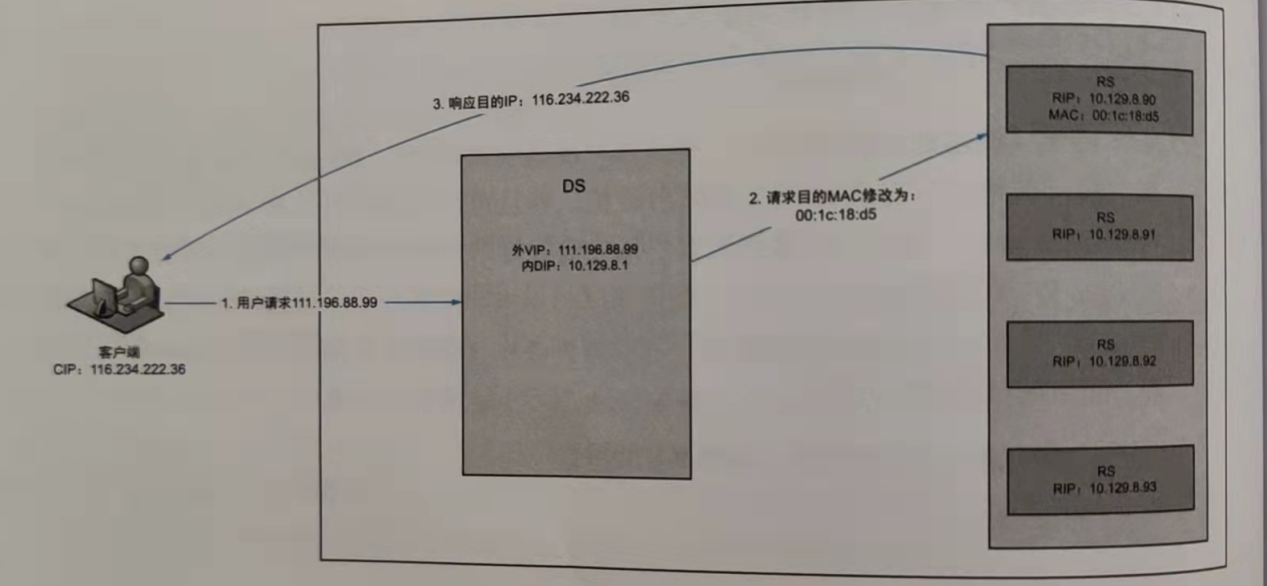
1. NAT模式

NAT模式在接收和返回请求时都需要经过DS进行处理，即DS可能成为性能颈瓶



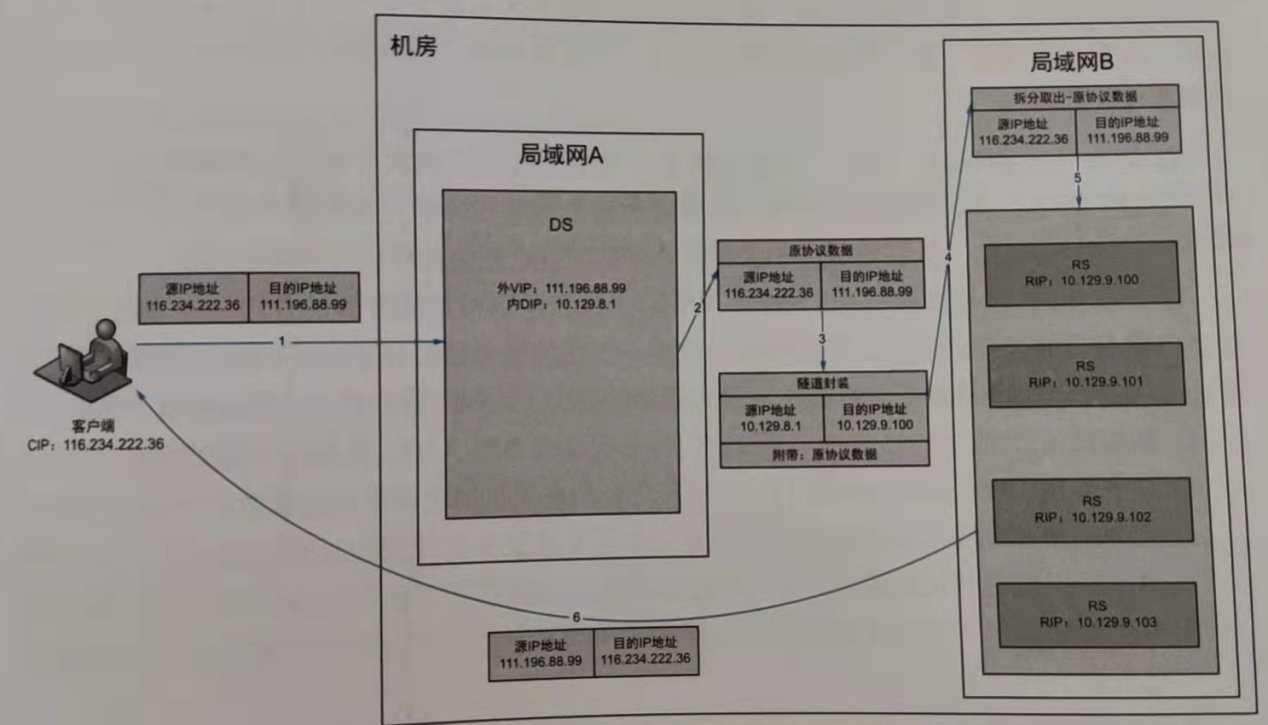
1. DR模式

DR模式返回时不需要经过DS，可以减轻DS的压力



1. TUN模式

TUN模式主要是应用在2个局域网之间，详情去百度

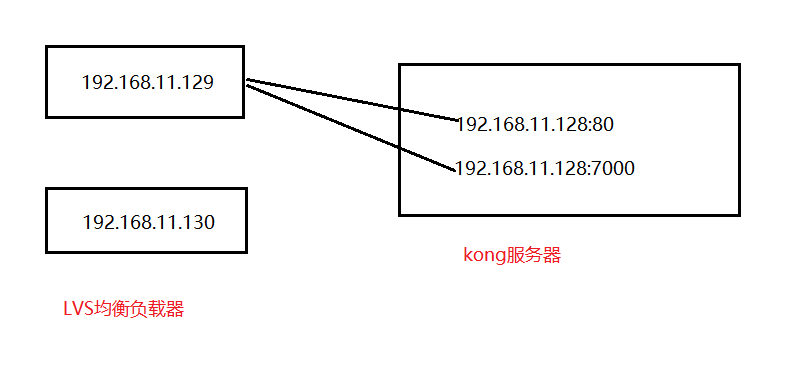


**LVS虚拟IP（VIP）**

VIP就是和DS的IP在同一网段内的IP，如果不在同一网段内，路由器不会转发收到的消息，DS自然无法接收到该消息，VIP可以是DS的IP或者是其他没有被占用的IP

**准备工作**

我们需要新增2台虚拟机，如下，129（主服务器MASTER）和130（备用服务器BACKUP）作为LVS均衡负载器，128上面运行着2个kong的服务器（就是我们前面教程示例的站点）



**安装keepalived**

我们在129和130上面安装keepalived

# sudo apt install keepalived

1. 主均衡负载器
2. 编辑/etc/keepalived/keepalived.conf配置文件

global\_defs {

notification\_email { //定义邮件服务的

root@localhost //定义收件人，这里改为本机，只是测试使用

}

notification\_email\_from kaadmin@localhost //定义发件人，

smtp\_server 127.0.0.1 //定义邮件服务器，一定不能使用外部地址

smtp\_connect\_timeout 30 //超时时间

router\_id LVS1 //LVS的ID

}

vrrp\_instance VI\_1 { //定义虚拟路由，VI\_1 为虚拟路由的标示符，自己定义名称

state MASTER //标识本keepalived为主服务器，备用服务器请使用BACKUP

interface ens33 //使用的网卡名，不同机器网卡名可能不一致，请注意

virtual\_router\_id 7 //虚拟路由的ID，主备需要一致

priority 100 //初始优先级，主服务器比备用服务器高

advert\_int 1 //主备之间同步检查的时间间隔，默认为1秒

authentication { //认证机制，主备必须一致

auth\_type PASS //认证类型

auth\_pass 1234 //密码，应该为随机的字符串

}

virtual\_ipaddress { //虚拟地址，即VIP

192.168.11.200 //VIP的地址

}

}

# 均衡负载地址

virtual\_server 192.168.11.200 80 { //指定VIP和端口

delay\_loop 6 //健康检查时间，秒

lb\_algo rr //负载均衡调度算法，RR即轮询

lb\_kind DR //均衡负载模式，这里是DR

nat\_mask 255.255.255.0 //掩码

persistence\_timeout 0 //持久连接时间

protocol TCP //健康检测协议

real\_server 192.168.11.128 80 { //定义后端RS地址

weight 1 //配置节点权重值

TCP\_CHECK { //RS TCP健康监测

connect\_timeout 10 //连接超时时间，秒

nb\_get\_retry 3 //尝试次数

delay\_before\_retry 3 //每次尝试连接的等待时间

connect\_port 80 //连接端口

}

}

real\_server 192.168.11.128 7000 { //定义后端RS地址

weight 1 //配置节点权重值

TCP\_CHECK { //RS TCP健康监测

connect\_timeout 10 //连接超时时间，秒

nb\_get\_retry 3 //尝试次数

delay\_before\_retry 3 //每次尝试连接的等待时间

connect\_port 7000 //连接端口

}

}

}

1. 开启ip4转发功能

编辑vim /etc/sysctl.conf文件，将如下配置设置为1（没有就添加）

net.ipv4.ip\_forward=1

执行如下配置使修改生效

# sysctl -p

1. 启动keepalived

# service keepalived start

1. 验证

我们ping 200查看是否能ping通

# ping 192.168.11.200

1. 备用服务器

备用服务器的流程和主服务器一样，只需配置以下几个配置

router\_id LVS2

state BACKUP

priority 99

当主服务器挂掉时，备用服务器将接手

**配置后端服务器健康监测**

keepalived虽然部署了，但器需要监测后端服务器的健康状态才会转发给后端服务器，如下配置后端服务器

1. 新建脚本文件vim /bin/realserve

#!/bin/bash

#虚拟IP

SNS\_VIP=192.168.11.200

case "$1" in

start)

`ifconfig lo:0 $SNS\_VIP broadcast $SNS\_VIP netmask 255.255.255.255`

`route add -host $SNS\_VIP dev lo:0`

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

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

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

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

sysctl -p > /dev/null 2>&1

echo "RealServer Start OK"

;;

stop)

`ifconfig lo:0 down`

`route del $SNS\_VIP`

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

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

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

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

echo "RealServer Stoped"

;;

\*)

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

exit 1

esac

exit 0

1. 添加可执行权限

# chmod 777 /bin/realserver

1. 运行脚本，配置后端服务器

# realserver start

1. 验证

如下，我们访问http://192.168.11.200/netkongtest/health/getNode，就可以看到后端服务返回的结构了

