准备

master:192.168.52.200

slave:192.168.52.201

web静态服务器(front-web或者manager-web)，这里只部署front-web静态服务器

nginx1:192.168.52.202

nginx2:192.168.52.203

#### nginx1,nginx2 安装

共同部分

yum install -y pcre pcre-devel 使用pcre库中的正则表达式

yum install -y gcc-c++ 需要gcc对源码进行编译

yum install -y zlib zlib-devel 主要用于对http包的压缩

yum install -y openssl openssl-devel 密码算法库，主要用于https

yum install -y net-tools 常用的网络工具

cd /opt 下载到opt目录

wget [http://nginx.org/download/nginx-1.14.2.tar.gz](https://nginx.org/download/nginx-1.14.2.tar.gz) 下载nginx包

tar -zxvf nginx-1.14.2.tar.gz 解压

cd nginx-1.14.2

mkdir /usr/local/software/nginx 安装位置

./configure --prefix=/usr/local/software/nginxt

make & make install

安装完毕后 分别启动nginx

/usr/local/software/nginx/sbin/nginx

cd /etc/init.d

touch realserver.sh 建立脚本

chmod u+x realserver.sh 赋予权限

vim realserver.sh

#!/bin/bash

# description: Config realserver lo and apply noarp

SNS\_VIP=192.168.52.100

case "$1" in

start)

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

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

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

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

echo "RealServer Start OK"

;;

stop)

ifconfig lo:0 down

route del $SNS\_VIP >/dev/null 2>&1

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

echo "RealServer Stoped"

;;

\*)

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

exit 1

esac

exit 0

两台nginx 分别执行脚本

./realserver.sh start

这里的LVS 采用的DR模式，调度算法采用的是RR，Master为主要机器，slave为热备份，master宕机，slave替换

#### master,slave 主机共同部分

安装必要的依赖包

yum install -y zlib zlib-devel gcc gcc-c++ openssl openssl-devel openssh

yum -y install libnl libnl-devel wget

yum -y install ipvsadm lvs工具

cd /opt

mkdir /usr/local/software/keepalived/ keepalived 安装位置

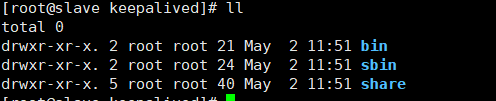
wget https://www.keepalived.org/software/keepalived-2.0.10.tar.gz

./configure --prefix=/usr/local/software/keepalived/ --sysconf=/etc

将默认的配置文件放到etc 目录下

make & make install 编译安装

cd /usr/local/software/keepalived/



Keepalived 可执行程序在sbin中

默认加载/etc/keepalived/keepalived.conf 文件

#### Master主机:

vim /etc/keepalived/keepalived.conf

! Configuration File for keepalived

global\_defs {　　　　　#全局配置

router\_id LVS\_DEVEL #lvs Id整个网络唯一

vrrp\_skip\_check\_adv\_addr

vrrp\_strict

vrrp\_garp\_interval 0

vrrp\_gna\_interval 0

}

vrrp\_instance VI\_1 { #vrrp 实例部分

state MASTER #指明是master主机

interface eth33 #所使用的网卡名称

virtual\_router\_id 60 #设置的路由标识同一个vrrp 数字唯一

priority 100 #设置优先级 master优先级必须要大于slave

advert\_int 1 #设定master与backup负载均衡器之间同步检查的时间间隔，单位是秒

authentication { #验证类型和密码

auth\_type PASS # 主要分为PASS和AH

auth\_pass 1111 #验证码同一个vrrp\_instance mater和slave 必须相同

}

virtual\_ipaddress { #虚拟机主机可设置多个每行一个

192.168.52.100

}

}

virtual\_server 192.168.52.100 80 { #设置虚拟服务器，需要指定虚拟ip和服务端口

delay\_loop 3 #健康检查时间间隔

lb\_algo rr #负载均衡调度算法

lb\_kind DR 　#负载均衡转发规则

persistence\_timeout 50 #设置会话保持时间，对动态网页非常有用

protocol TCP　　　　#指定转发协议类型，有TCP和UDP两种

real\_server 192.168.52.202 80 { #配置服务器节点1，需要指定真实主机IP地址和端口

weight 1 #设置权重，数字越大权重越高

TCP\_CHECK {

connect\_timeout 3 #超时时间

nb\_get\_retry 3 #重试次数

delay\_before\_retry 3 #重试间隔

connect\_port 80 #监听端口

}

}

real\_server 192.168.52.203 80 {

weight 1

TCP\_CHECK {

connect\_timeout 3

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

}

#### Slave 主机:

vim /etc/keepalived/keepalived.conf

! Configuration File for keepalived

global\_defs {

smtp\_connect\_timeout 30

router\_id LVS\_DEVEL

vrrp\_skip\_check\_adv\_addr

vrrp\_strict

vrrp\_garp\_interval 0

vrrp\_gna\_interval 0

}

vrrp\_instance VI\_1 {

state BACKUP

interface eth1

virtual\_router\_id 60

priority 80

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1111

}

virtual\_ipaddress {

192.168.52.100

}

}

virtual\_server 192.168.52.100 80 {

delay\_loop 3

lb\_algo rr

lb\_kind DR

persistence\_timeout 3

protocol TCP

real\_server 192.168.52.202 80 {

weight 1

TCP\_CHECK {

connect\_timeout 3

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

real\_server 192.168.52.203 80 {

weight 1

TCP\_CHECK {

connect\_timeout 3

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

}

Master,slave 共同部分

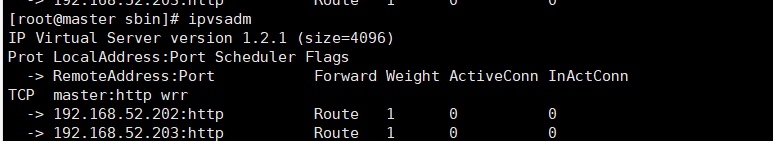
/usr/local/software/keepalived/sbin/keepalived

默认启动了三个进程

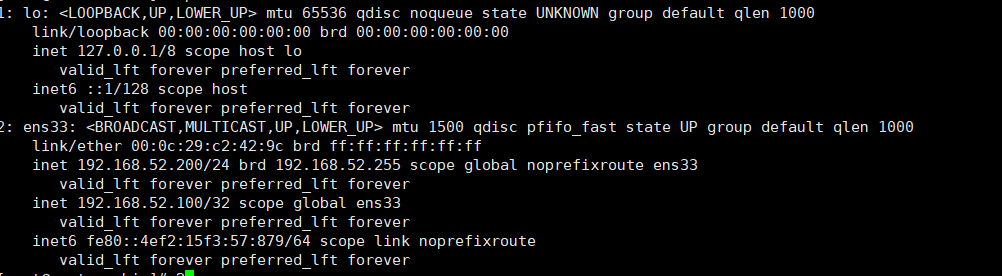


tail -f /var/log/messages 查看启动日志

可以看到负载两台主机

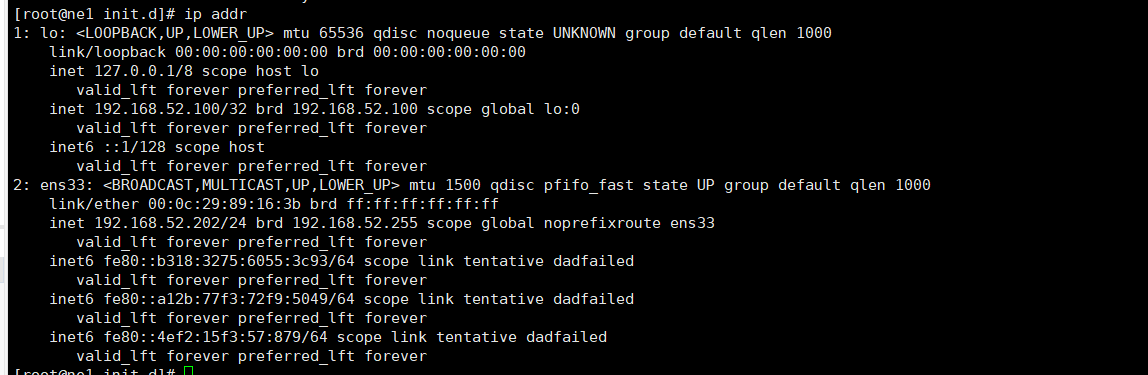
master 主机

ip addr 可以看到VIP 已经生成



Nginx

ip addr 可以看到本地返回地址已经设置成VIP



网络结构如图所示

