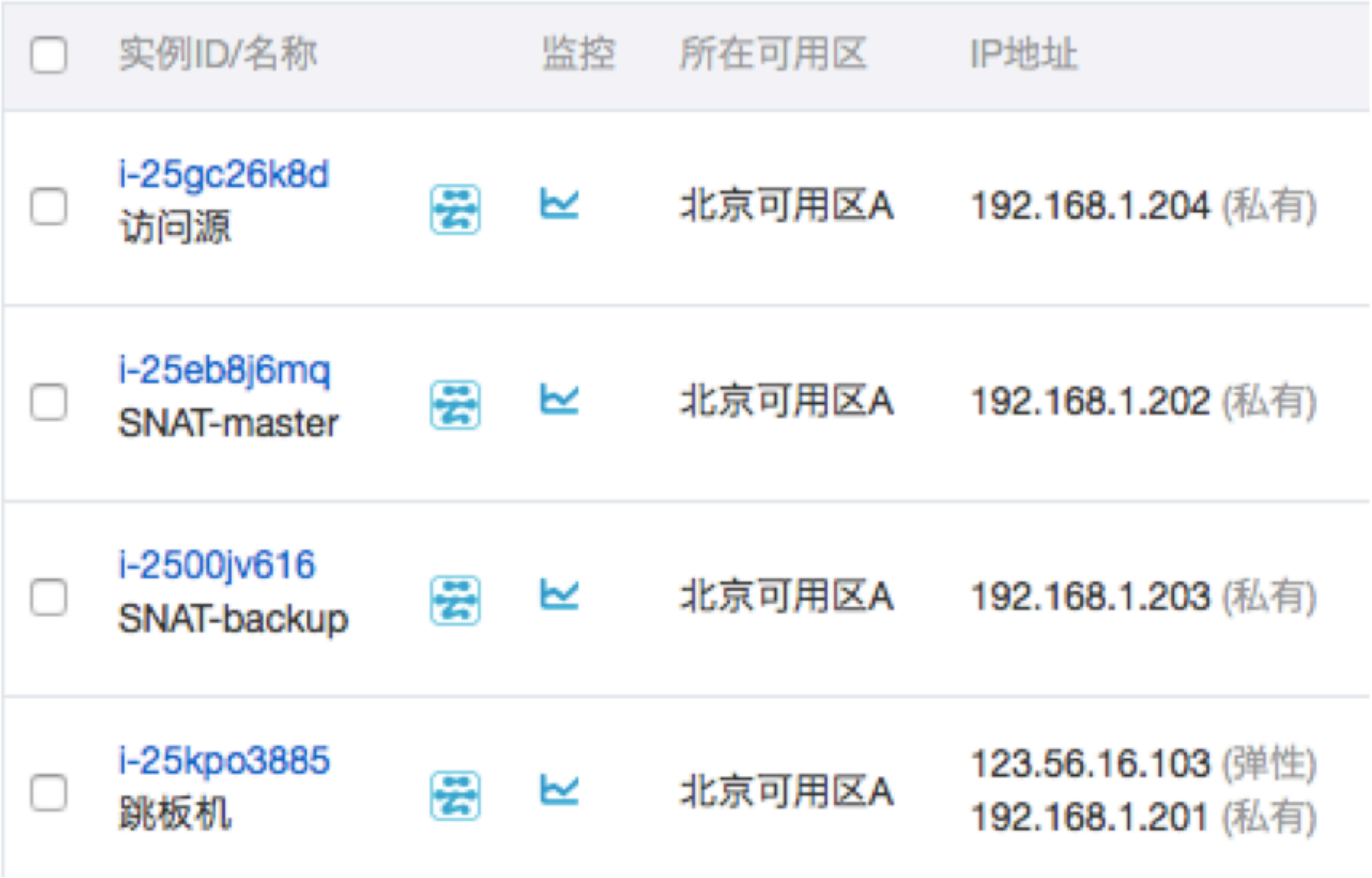
使用keepalived和HaVip搭建具备高可用能力的SNAT网关

在VPC中，为一台ECS绑定EIP，并在该ECS上搭建代理软件，可以将该ECS实例建设为一个SNAT网关，让同VPC内其他实例将该实例作为公网网关进行公网访问。

然而，这种方式搭建的SNAT网关是个单点，可用性较差。我们可以使用keepalived和HaVip来搭建一个具备主备切换能力的高可用SNAT网关。

环境

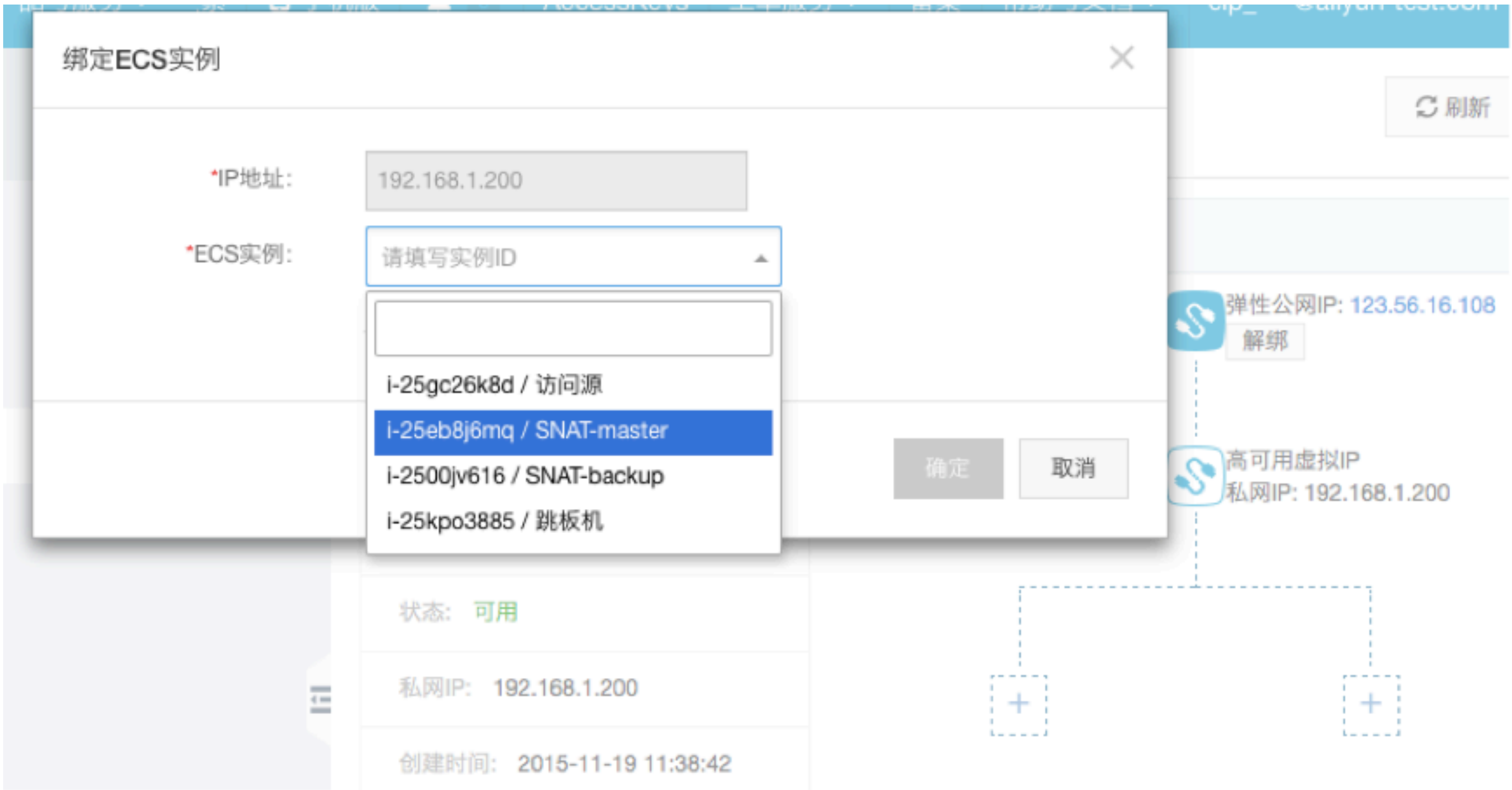
1. 准备两个EIP。
2. 准备四台ECS实例，在一个VPC的同一个VSwitch下。
    - 192.168.1.201 （绑定了EIP 123.56.16.103）：当做跳板机使用，SSH到这台机器上以后，再SSH私网IP跳转到其他机器。
    - 192.168.1.202：SNAT双机中的主实例，一会儿会绑在HaVip上。
    - 192.168.1.203：SNAT双机中的备实例，一会儿会绑在HaVip上。
    - 192.168.1.204：当作需要上网的实例，用它来测试SNAT的效果。



□	实例ID/名称		监控	所在可用区	IP地址
□	i-25gc26k8d 访问源			北京可用区A	192.168.1.204 (私有)
□	i-25eb8j6mq SNAT-master			北京可用区A	192.168.1.202 (私有)
□	i-2500jv616 SNAT-backup			北京可用区A	192.168.1.203 (私有)
□	i-25kpo3885 跳板机			北京可用区A	123.56.16.103 (弹性) 192.168.1.201 (私有)

1. 准备一个HaVip：

   私网IP：192.168.1.200 绑定了EIP：123.56.16.108 绑定了两个实例：192.168.1.202、192.168.1.203；



绑定ECS实例

*IP地址: 192.168.1.200

*ECS实例: 请填写实例ID

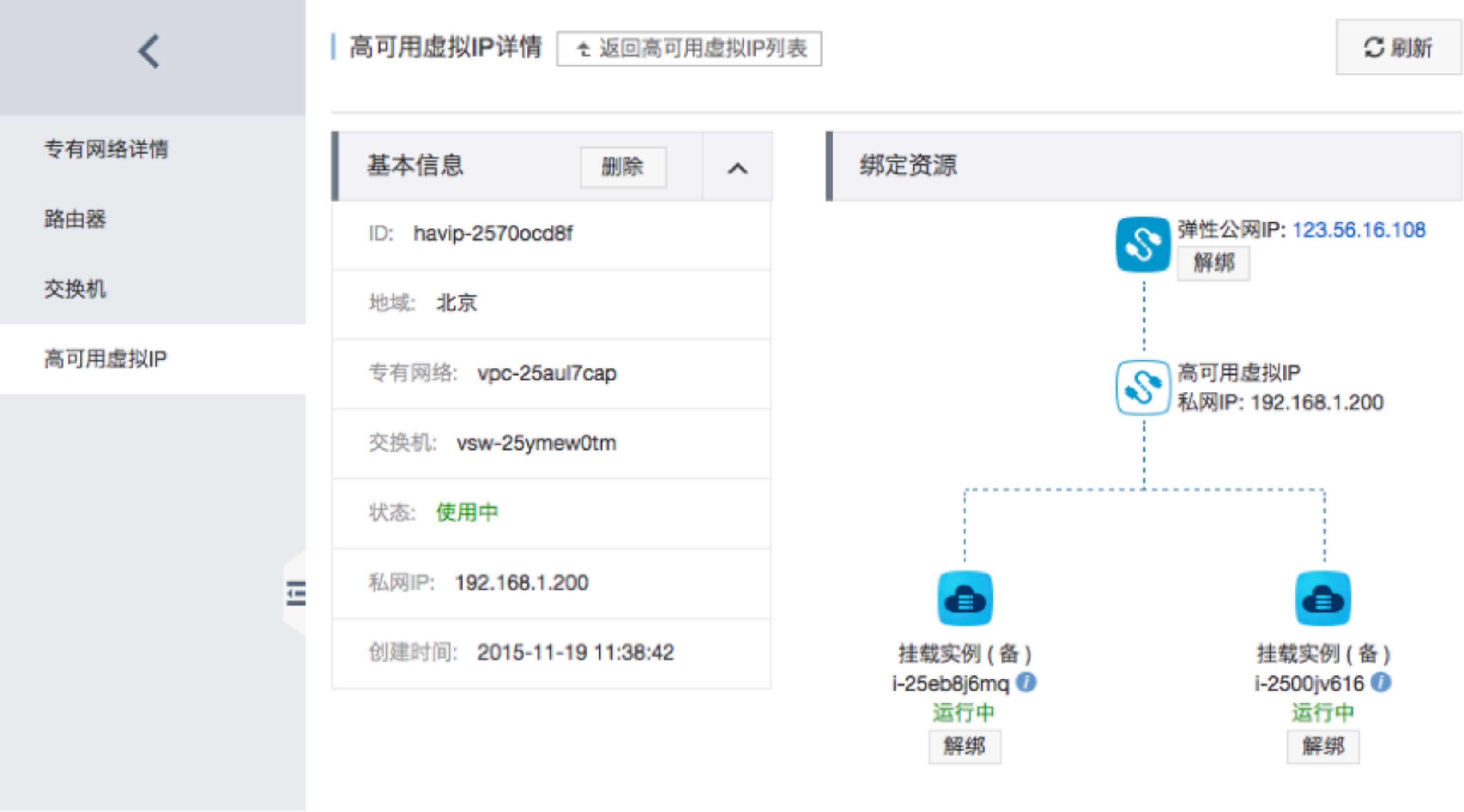
弹性公网IP: 123.56.16.108

高可用虚拟IP
私网IP: 192.168.1.200

状态: 可用

私网IP: 192.168.1.200

创建时间: 2015-11-19 11:38:42



高可用虚拟IP详情 返回高可用虚拟IP列表 刷新

基本信息 删除

ID: havip-2570ocd8f

地域: 北京

专有网络: vpc-25aul7cap

交换机: vsw-25ymew0tm

状态: 使用中

私网IP: 192.168.1.200

创建时间: 2015-11-19 11:38:42

绑定资源

弹性公网IP: 123.56.16.108 解绑

高可用虚拟IP
私网IP: 192.168.1.200

挂载实例（备）
i-25eb8j6mq
运行中
解绑

挂载实例（备）
i-2500jv616
运行中
解绑

搭建与配置

Keepalived的安装：

在要当做SNAT服务器的两台ECS实例上，执行以下keepalived安装流程：

下载：

```
[root@iZ250sept0mZ ~]# wget   http://www.keepalived.org/software/keepaliv

由于目前这台机器目前不能直接连上公网，所以可以在跳板机上进行wget，然后scp到这两台机器_
```

安装：

```
[root@iZ250sept0mZ ~]# tar -zxf keepalived-1.2.19.tar.gz
[root@iZ250sept0mZ ~]# cd keepalived-1.2.19
[root@iZ250sept0mZ keepalived-1.2.19]# ./configure
[root@iZ250sept0mZ keepalived-1.2.19]# make && make install
```

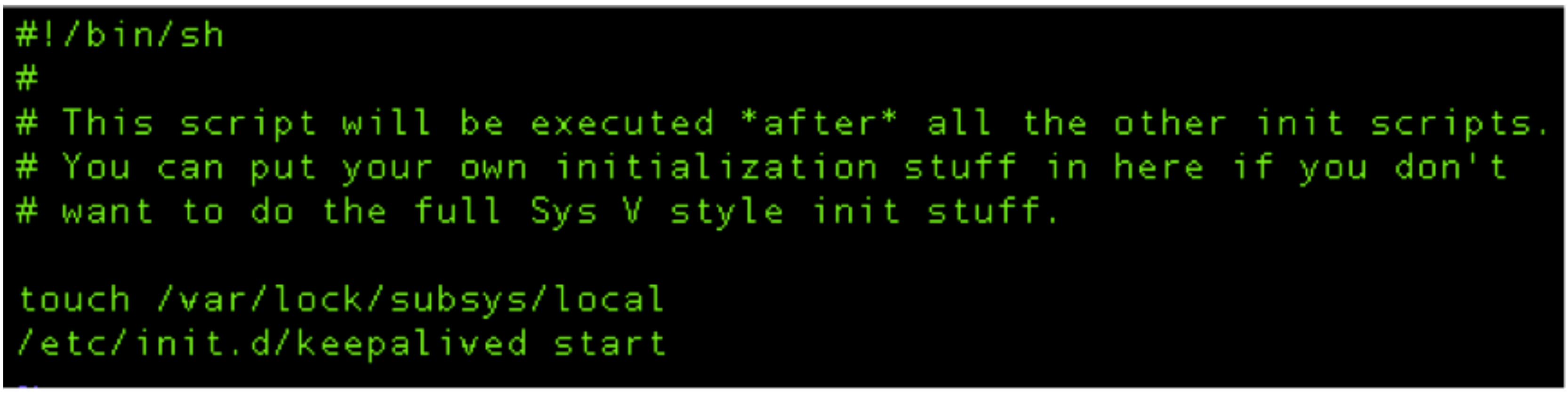
修改配置文件路径：

```
[root@iZ250sept0mZ keepalived-1.2.19]# cp /usr/local/etc/rc.d/init.d/keep
[root@iZ250sept0mZ keepalived-1.2.19]# cp /usr/local/etc/sysconfig/keepal
[root@iZ250sept0mZ keepalived-1.2.19]# mkdir /etc/keepalived
```

```
[root@iZ250sept0mZ keepalived-1.2.19]# cp /usr/local/etc/keepalived/keepa
[root@iZ250sept0mZ keepalived-1.2.19]# cp /usr/local/sbin/keepalived /usr
```

将keepalived设置为服务，开机启动：

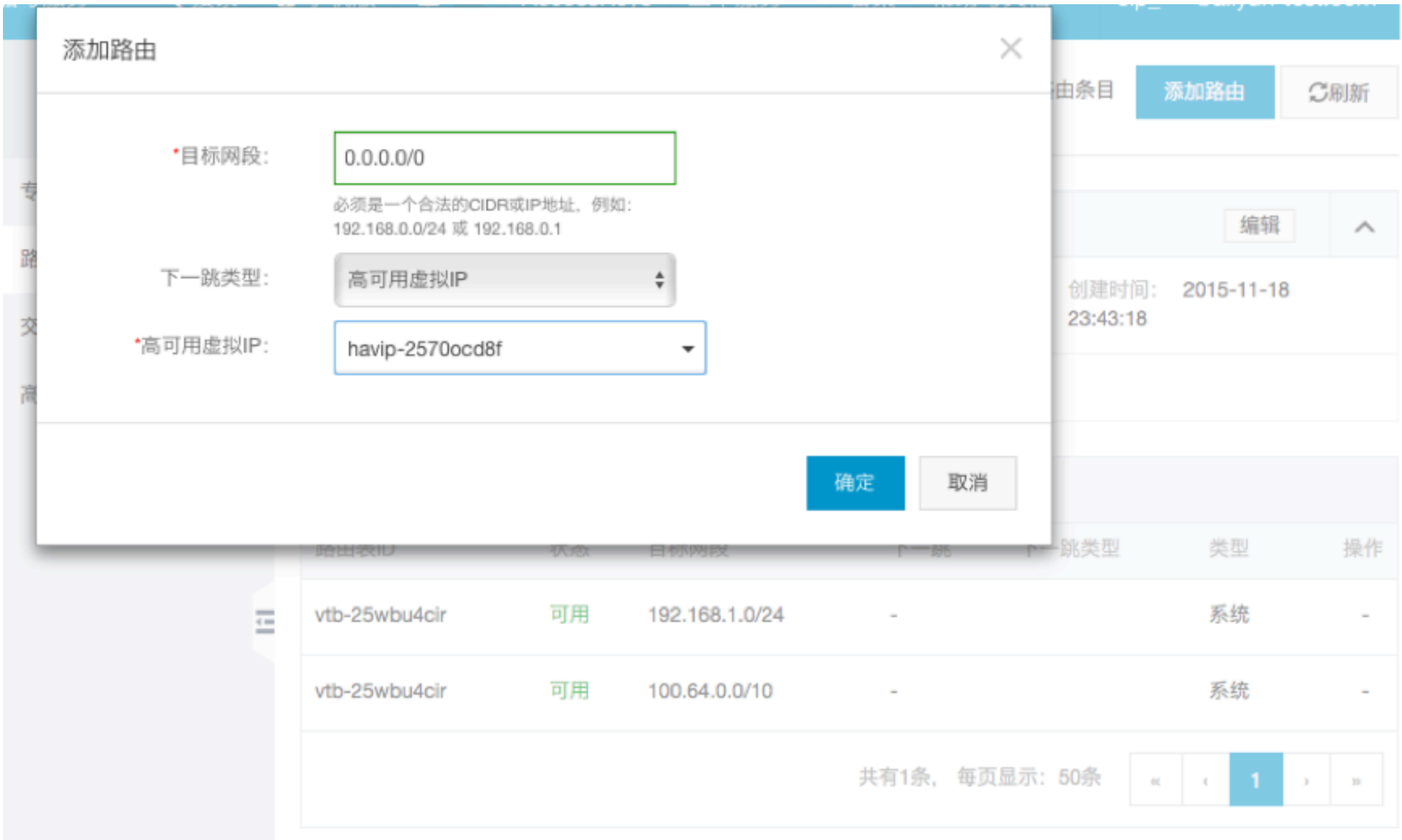
```
[root@iZ250sept0mZ keepalived-1.2.19]# vi /etc/rc.local
```



```
#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.

touch /var/lock/subsys/local
/etc/init.d/keepalived start
```

自定义路由配置：

```
添加一条自定义路由，目的CIDR为0.0.0.0/0 ， 下一跳指向HaVip对象
```



添加路由

*目标网段：0.0.0.0/0

必须是一个合法的CIDR或IP地址，例如：192.168.0.0/24 或 192.168.0.1

下一跳类型：高可用虚拟IP

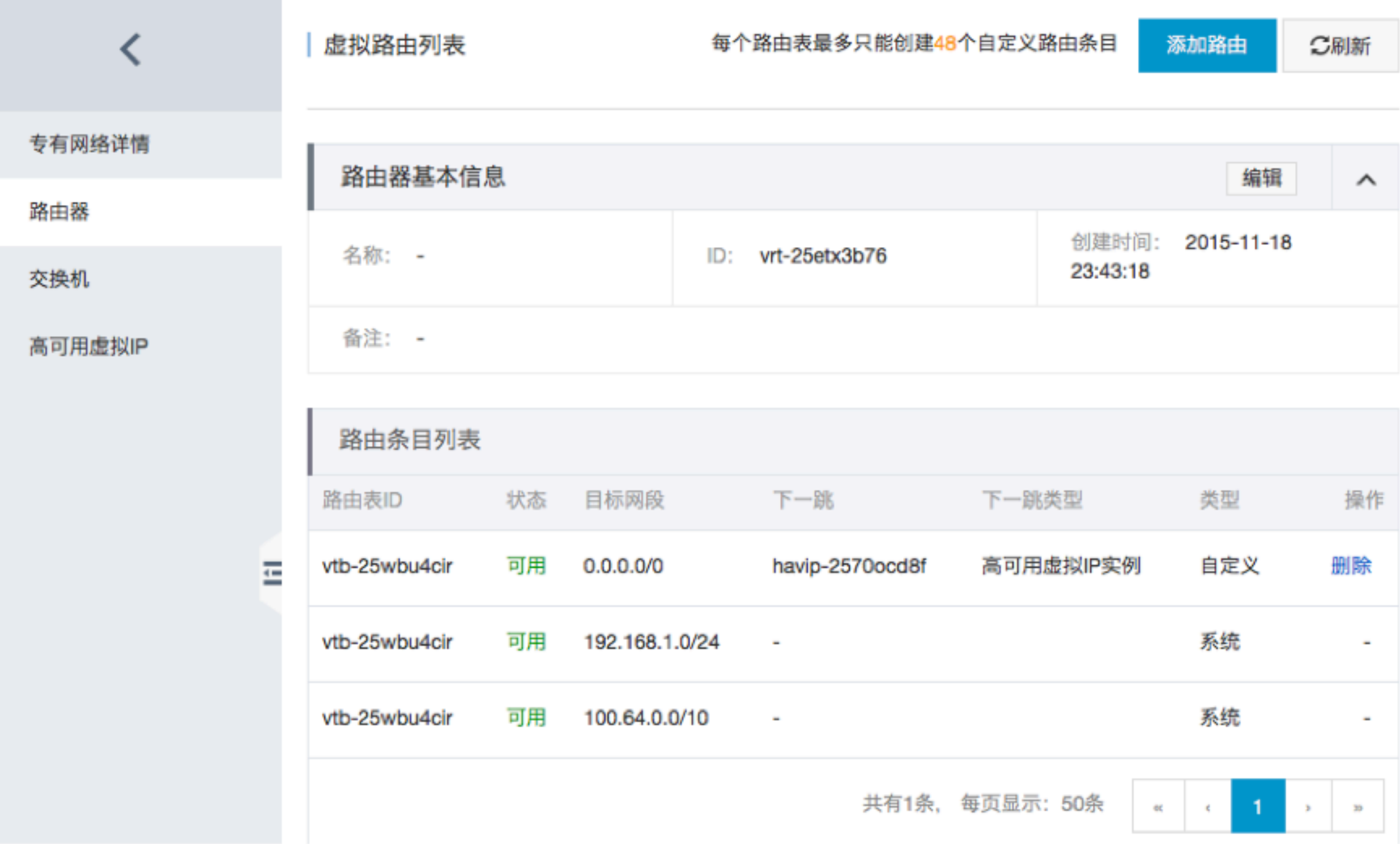
*高可用虚拟IP：havip-2570ocd8f

创建时间：2015-11-18 23:43:18

vtb-25wbu4cir	可用	192.168.1.0/24	-	系统	-
vtb-25wbu4cir	可用	100.64.0.0/10	-	系统	-

共有1条， 每页显示：50条

添加完成后的效果如下：



每个路由表最多只能创建48个自定义路由条目

路由器基本信息

名称： -

ID: vrt-25etx3b76

创建时间： 2015-11-18 23:43:18

备注： -

路由条目列表

路由表ID	状态	目标网段	下一跳	下一跳类型	类型	操作
vtb-25wbu4cir	可用	0.0.0.0/0	havip-2570ocd8f	高可用虚拟IP实例	自定义	删除
vtb-25wbu4cir	可用	192.168.1.0/24	-		系统	-
vtb-25wbu4cir	可用	100.64.0.0/10	-		系统	-

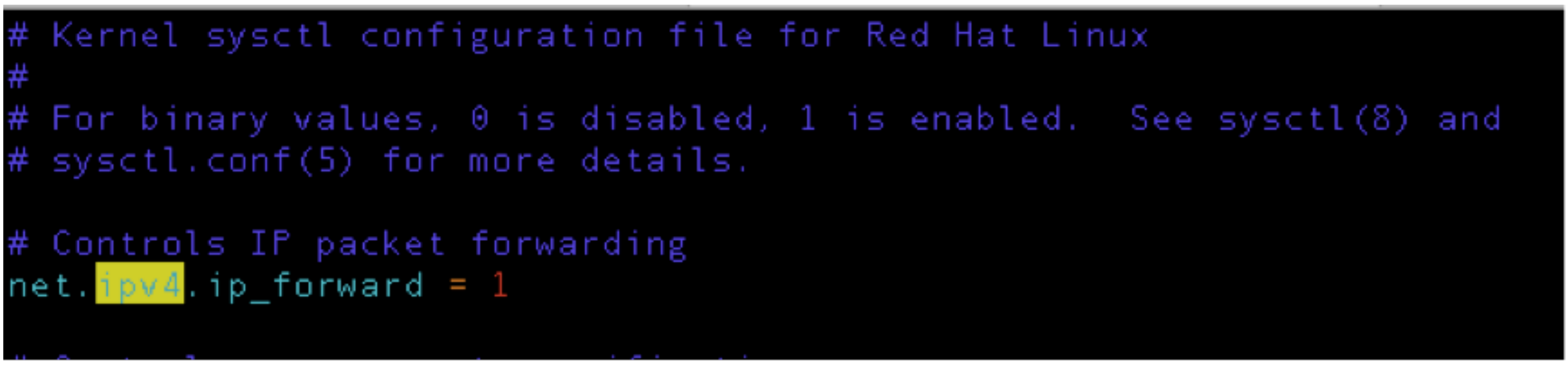
共有1条， 每页显示：50条 « ‹ 1 › »

开启内核的IP转发选项：

在两台SNAT服务器ECS实例上都进行如下修改：

运行： `sysctl -w net.ipv4.ip_forward=1`

为了保证实例重启后依然是开启这个选项的，需要对/etc/sysctl.conf 这个配置文件进行修改，将net.ipv4.ip_forward的值改为1，见下图：



```
# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled.  See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 1
```

SNAT配置

在两台当做SNAT服务器的ECS实例上，准备两个shell脚本:

- 脚本一： 用于主备切换时让新的master机自动开启IP转发、加载SNAT规则，实现SNAT转发：
- 脚本二： 用于主机切换成备机时或者主机keepalived出错时去除SNAT转发规则（不

去除的话，上网会有问题）； P.S. 如果备机带着这几条SNAT规则工作，会导致主机无法直接上网；

两个脚本的内容如下：

脚本一：/etc/keepalived/scripts/ha_vip_start.sh

```
#!/bin/bash

echo "start; `date`" >> /tmp/log
sysctl -w net.ipv4.ip_forward=1
iptables -t nat -A POSTROUTING -d 100.64.0.0/10 -j RETURN
iptables -t nat -A POSTROUTING -d 10.0.0.0/8 -j RETURN
iptables -t nat -A POSTROUTING -s 192.168.0.0/16 ! -p vrrp -j SNAT --to-s
```

需要注意的地方：

1. 红字地方尤其需要注意，应该是HaVip的私网IP。
2. 需要注意几条规则的顺序，要按照上文给出的顺序才行。
3. 如果您的VPC的CIDR是10.0.0.0/8网段，那么需要略过第二条规则不加。

脚本二：/etc/keepalived/scripts/ha_vip_stop.sh

```
#!/bin/bash

echo "stop; `date`" >> /tmp/log
iptables -t nat -F
```

Keepalived配置

配置文件位置：/etc/keepalived/keepalived.conf

Master实例 (例子中的192.168.1.202)的完整配置文件：

```
! Configuration File for keepalived

global_defs {
   notification_email {
     acassen@firewall.loc
     failover@firewall.loc
     sysadmin@firewall.loc
   }
   notification_email_from zhao.wang_havip@firewall.loc
   smtp_server 192.168.200.1
   smtp_connect_timeout 30
   router_id LVS_DEVEL
}

vrrp_instance VI_1 {
    state MASTER
    interface eth0
    virtual_router_id 51
    priority 100
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        192.168.1.200 dev eth0 label eth0:havip
}
    notify_master /etc/keepalived/scripts/ha_vip_start.sh
    notify_backup /etc/keepalived/scripts/ha_vip_stop.sh
    notify_fault  /etc/keepalived/scripts/ha_vip_stop.sh
    notify_stop   /etc/keepalived/scripts/ha_vip_stop.sh
    unicast_src_ip 192.168.1.202
    unicast_peer {
            192.168.1.203
                 }
}
```

注意：配置文件中，

- `192.168.1.202`和`192.168.1.203`应该换成你的两台实例的私网IP； 注意两个IP分别的位置，不要写反了。
- `192.168.1.200`应该是你的HaVip的私网IP地址。

backup 实例（例子中的192.168.1.203)的完整配置文件：

```
! Configuration File for keepalived

global_defs {
   notification_email {
     acassen@firewall.loc
     failover@firewall.loc
     sysadmin@firewall.loc
   }
   notification_email_from zhao.wang_havip@firewall.loc
   smtp_server 192.168.200.1
   smtp_connect_timeout 30
   router_id LVS_DEVEL
}

vrrp_instance VI_1 {
    state BACKUP
    interface eth0
    virtual_router_id 51
    priority 99
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        192.168.1.200 dev eth0 label eth0:havip
}
    notify_master /etc/keepalived/scripts/ha_vip_start.sh
    notify_backup /etc/keepalived/scripts/ha_vip_stop.sh
    notify_fault  /etc/keepalived/scripts/ha_vip_stop.sh
    notify_stop   /etc/keepalived/scripts/ha_vip_stop.sh
    unicast_src_ip 192.168.1.203
    unicast_peer {
            192.168.1.202
                 }
}
```

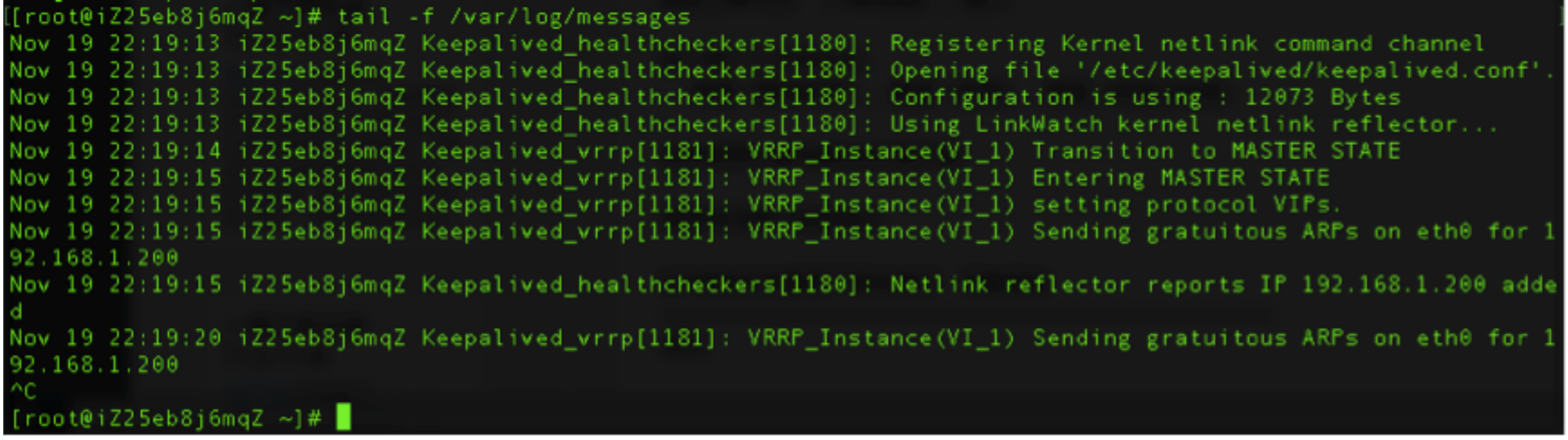
注意：同样需要注意其中的私网IP，换成你的两台实例的私网IP；注意对应位置，不要写反了。

启动服务、验证SNAT效果

在202上启动keepalived

```
[root@iZ25eb8j6mqZ ~]# service keepalived start
```

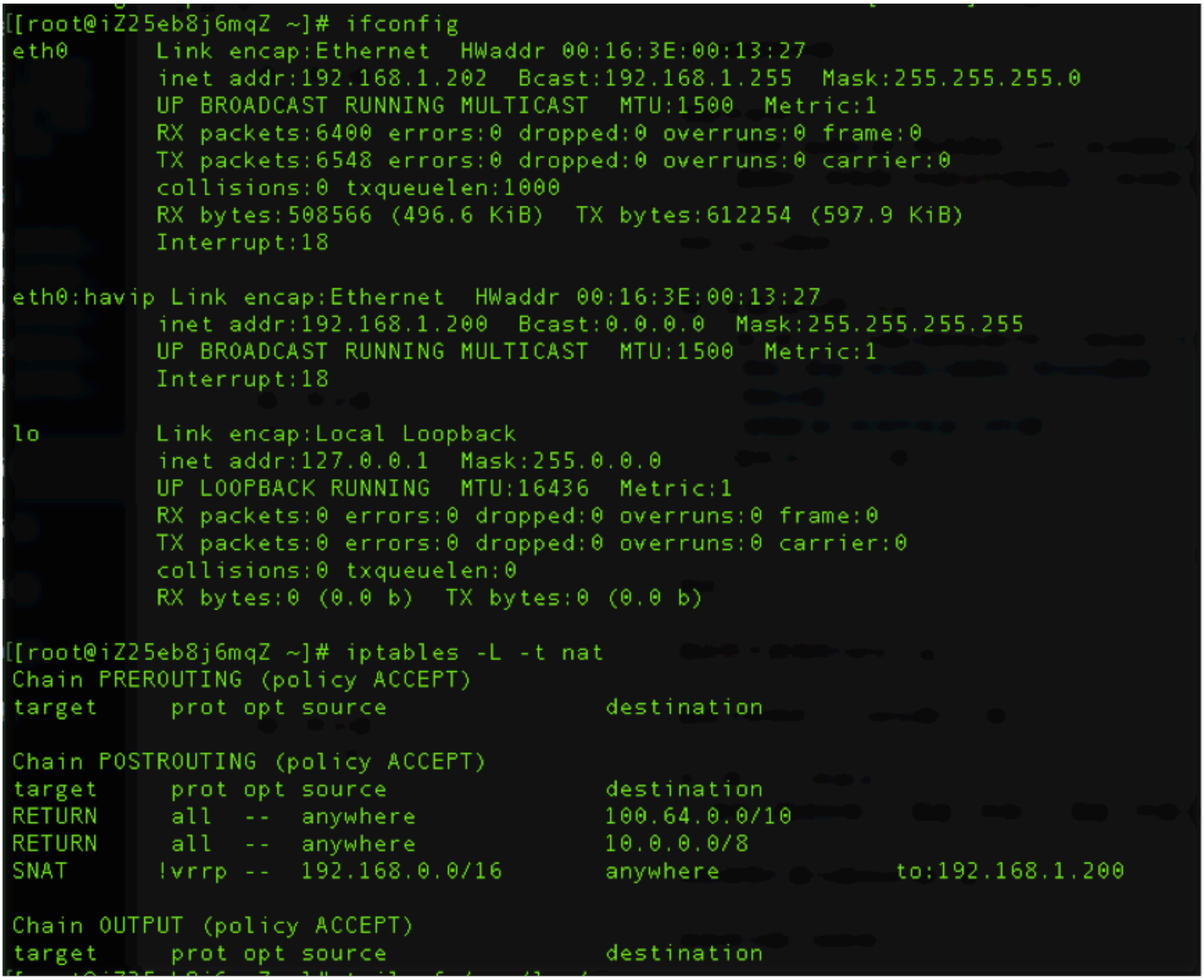
观察log，进入master状态：



```
[root@iZ25eb8j6mqZ ~]# tail -f /var/log/messages
Nov 19 22:19:13 iZ25eb8j6mqZ Keepalived_healthcheckers[1180]: Registering Kernel netlink command channel
Nov 19 22:19:13 iZ25eb8j6mqZ Keepalived_healthcheckers[1180]: Opening file '/etc/keepalived/keepalived.conf'.
Nov 19 22:19:13 iZ25eb8j6mqZ Keepalived_healthcheckers[1180]: Configuration is using : 12073 Bytes
Nov 19 22:19:13 iZ25eb8j6mqZ Keepalived_healthcheckers[1180]: Using LinkWatch kernel netlink reflector...
Nov 19 22:19:14 iZ25eb8j6mqZ Keepalived_vrrp[1181]: VRRP_Instance(VI_1) Transition to MASTER STATE
Nov 19 22:19:15 iZ25eb8j6mqZ Keepalived_vrrp[1181]: VRRP_Instance(VI_1) Entering MASTER STATE
Nov 19 22:19:15 iZ25eb8j6mqZ Keepalived_vrrp[1181]: VRRP_Instance(VI_1) setting protocol VIPs.
Nov 19 22:19:15 iZ25eb8j6mqZ Keepalived_vrrp[1181]: VRRP_Instance(VI_1) Sending gratuitous ARPs on eth0 for 1
92.168.1.200
Nov 19 22:19:15 iZ25eb8j6mqZ Keepalived_healthcheckers[1180]: Netlink reflector reports IP 192.168.1.200 adde
d
Nov 19 22:19:20 iZ25eb8j6mqZ Keepalived_vrrp[1181]: VRRP_Instance(VI_1) Sending gratuitous ARPs on eth0 for 1
92.168.1.200
^C
[root@iZ25eb8j6mqZ ~]#
```

查看网卡配置，出现了192.168.1.200的ip;

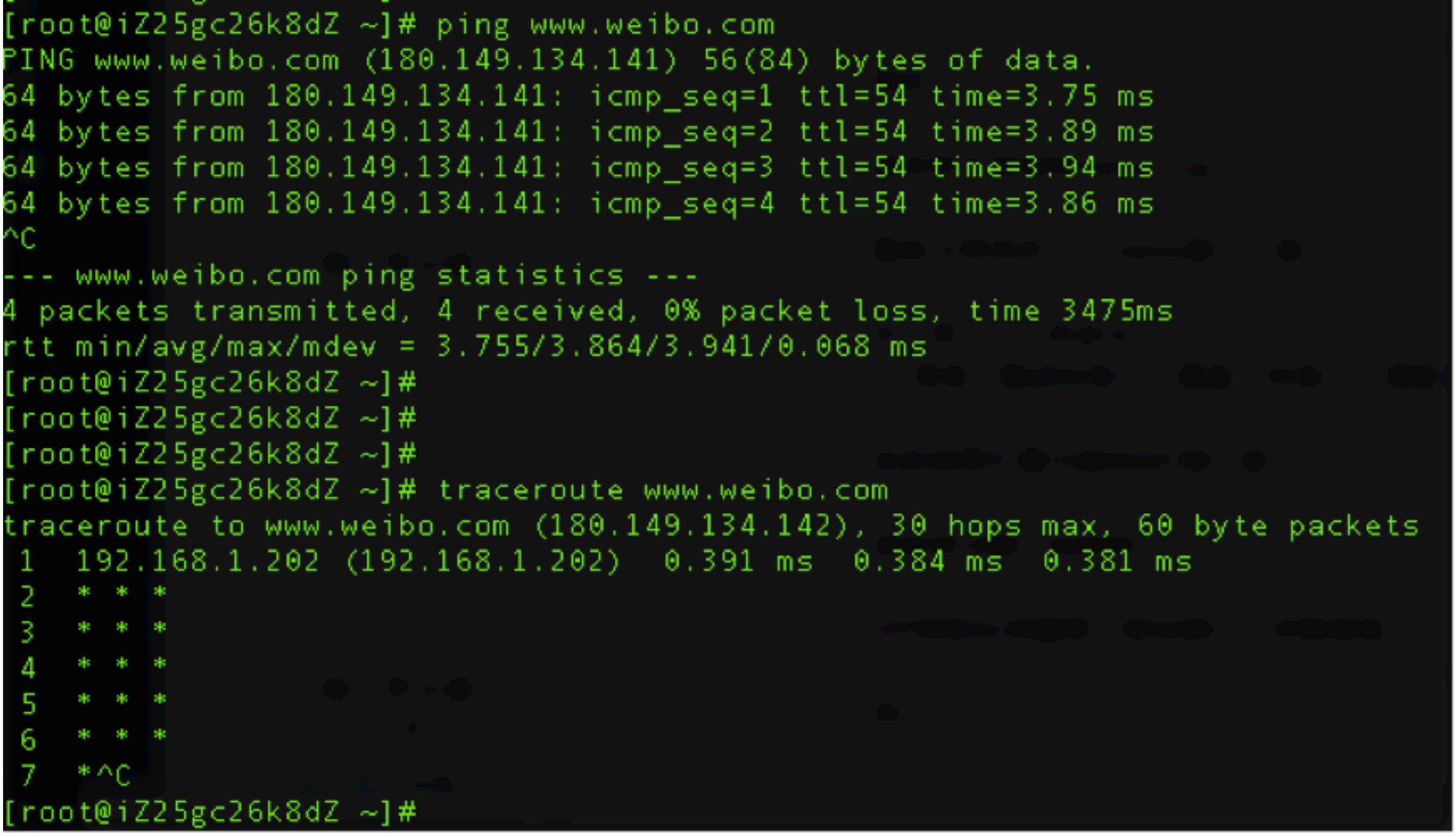
查看iptables规则，出现了SNAT相关规则；



```
[root@iZ25eb8j6mqZ ~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:16:3E:00:13:27
          inet addr:192.168.1.202  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6400 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6548 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:508566 (496.6 KiB)  TX bytes:612254 (597.9 KiB)
          Interrupt:18

eth0:havip Link encap:Ethernet  HWaddr 00:16:3E:00:13:27
          inet addr:192.168.1.200  Bcast:0.0.0.0  Mask:255.255.255.255
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          Interrupt:18

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)

[root@iZ25eb8j6mqZ ~]# iptables -L -t nat
Chain PREROUTING (policy ACCEPT)
target     prot opt source               destination

Chain POSTROUTING (policy ACCEPT)
target     prot opt source               destination
RETURN     all  --  anywhere             100.64.0.0/10
RETURN     all  --  anywhere             10.0.0.0/8
SNAT      !vrrp --  192.168.0.0/16       anywhere            to:192.168.1.200

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

在204上验证上网效果

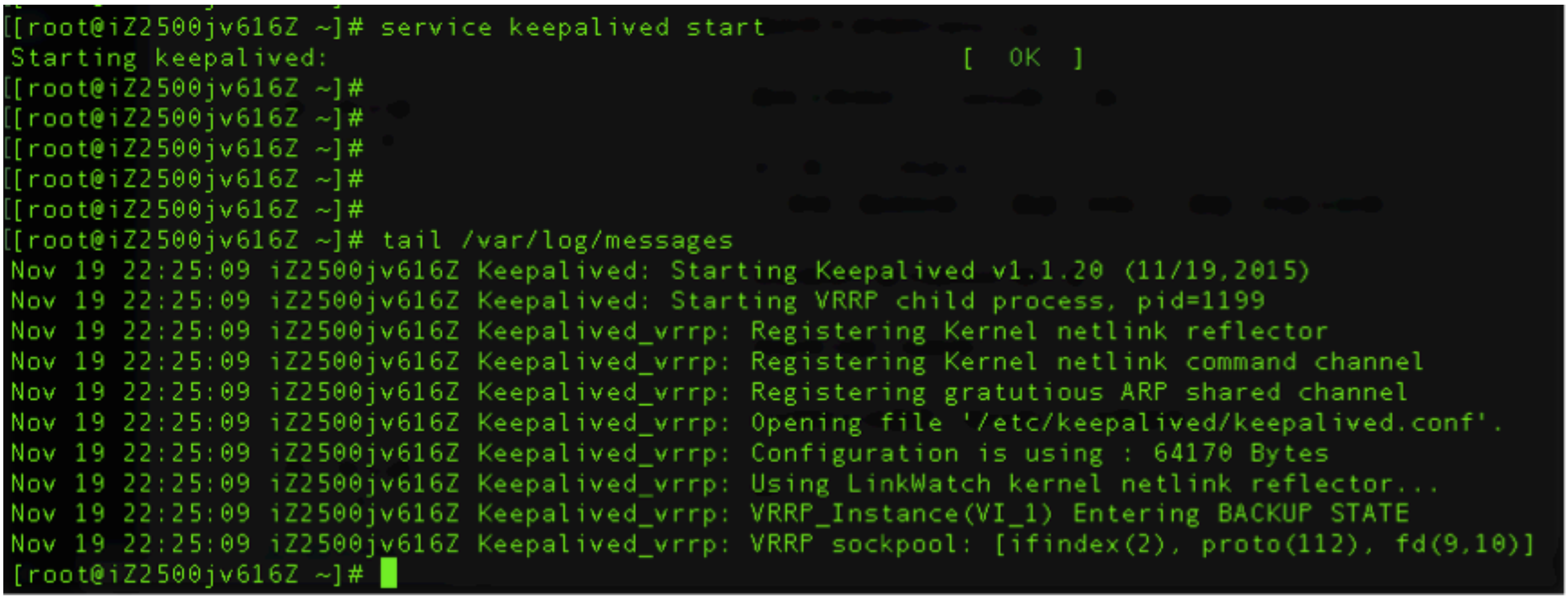
ping公网网址可以通；traceroute可以看到第一跳为192.168.1.202



```
[root@iZ25gc26k8dZ ~]# ping www.weibo.com
PING www.weibo.com (180.149.134.141) 56(84) bytes of data.
64 bytes from 180.149.134.141: icmp_seq=1 ttl=54 time=3.75 ms
64 bytes from 180.149.134.141: icmp_seq=2 ttl=54 time=3.89 ms
64 bytes from 180.149.134.141: icmp_seq=3 ttl=54 time=3.94 ms
64 bytes from 180.149.134.141: icmp_seq=4 ttl=54 time=3.86 ms
^C
--- www.weibo.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3475ms
rtt min/avg/max/mdev = 3.755/3.864/3.941/0.068 ms
[root@iZ25gc26k8dZ ~]#
[root@iZ25gc26k8dZ ~]#
[root@iZ25gc26k8dZ ~]#
[root@iZ25gc26k8dZ ~]# traceroute www.weibo.com
traceroute to www.weibo.com (180.149.134.142), 30 hops max, 60 byte packets
 1  192.168.1.202 (192.168.1.202)  0.391 ms  0.384 ms  0.381 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  *^C
[root@iZ25gc26k8dZ ~]#
```

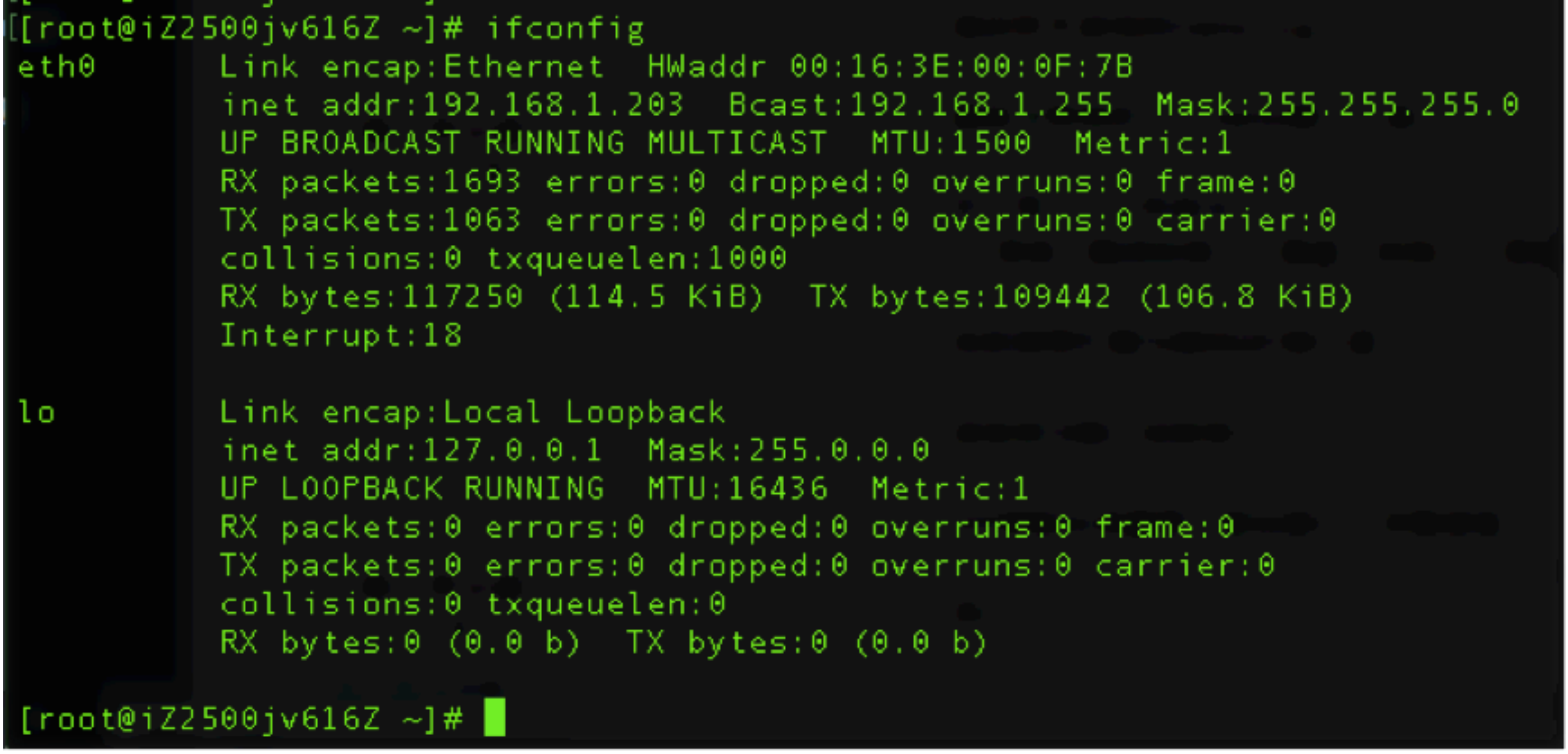
在203上启动keepalived

观察log，进入backup状态：



```
[root@iZ2500jv616Z ~]# service keepalived start
Starting keepalived:                                       [  OK  ]
[root@iZ2500jv616Z ~]#
[root@iZ2500jv616Z ~]#
[root@iZ2500jv616Z ~]#
[root@iZ2500jv616Z ~]#
[root@iZ2500jv616Z ~]#
[root@iZ2500jv616Z ~]# tail /var/log/messages
Nov 19 22:25:09 iZ2500jv616Z Keepalived: Starting Keepalived v1.1.20 (11/19,2015)
Nov 19 22:25:09 iZ2500jv616Z Keepalived: Starting VRRP child process, pid=1199
Nov 19 22:25:09 iZ2500jv616Z Keepalived_vrrp: Registering Kernel netlink reflector
Nov 19 22:25:09 iZ2500jv616Z Keepalived_vrrp: Registering Kernel netlink command channel
Nov 19 22:25:09 iZ2500jv616Z Keepalived_vrrp: Registering gratutious ARP shared channel
Nov 19 22:25:09 iZ2500jv616Z Keepalived_vrrp: Opening file '/etc/keepalived/keepalived.conf'.
Nov 19 22:25:09 iZ2500jv616Z Keepalived_vrrp: Configuration is using : 64170 Bytes
Nov 19 22:25:09 iZ2500jv616Z Keepalived_vrrp: Using LinkWatch kernel netlink reflector...
Nov 19 22:25:09 iZ2500jv616Z Keepalived_vrrp: VRRP_Instance(VI_1) Entering BACKUP STATE
Nov 19 22:25:09 iZ2500jv616Z Keepalived_vrrp: VRRP sockpool: [ifindex(2), proto(112), fd(9,10)]
[root@iZ2500jv616Z ~]#
```

查看网卡信息，发现并没有出现192.168.1.200，因为此时202是master，203还只是个备胎：



```
[root@iZ2500jv616Z ~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:16:3E:00:0F:7B
          inet addr:192.168.1.203  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1693 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1063 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:117250 (114.5 KiB)  TX bytes:109442 (106.8 KiB)
          Interrupt:18

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)

[root@iZ2500jv616Z ~]#
```

在204上验证上网效果

可以看到，和刚才验证的效果一样

将202的keepalived停掉

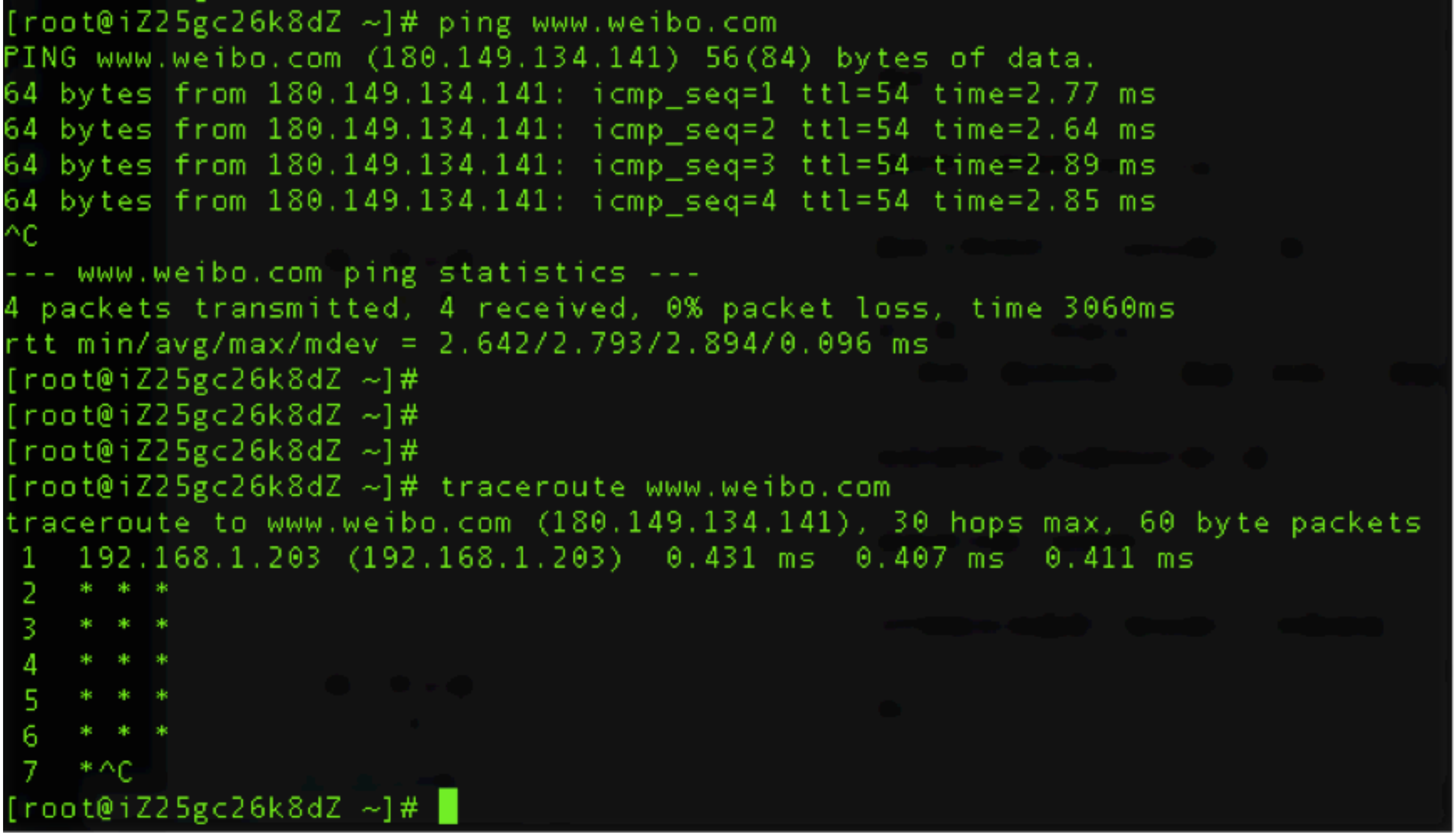
```
[root@iZ25eb8j6mqZ ~]# service keepalived stop
```

可以观察到：

1. 202的网卡上不再有192.168.1.200
2. 203的keepalived log显示，进入master状态
3. 203 的网卡上出现192.168.1.200

在204上验证上网效果

可以看到，依然可以ping通，traceroute变成了第一跳为203。说明，此时203成为了master，接管了vip。



```
[root@iZ25gc26k8dZ ~]# ping www.weibo.com
PING www.weibo.com (180.149.134.141) 56(84) bytes of data.
64 bytes from 180.149.134.141: icmp_seq=1 ttl=54 time=2.77 ms
64 bytes from 180.149.134.141: icmp_seq=2 ttl=54 time=2.64 ms
64 bytes from 180.149.134.141: icmp_seq=3 ttl=54 time=2.89 ms
64 bytes from 180.149.134.141: icmp_seq=4 ttl=54 time=2.85 ms
^C
--- www.weibo.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3060ms
rtt min/avg/max/mdev = 2.642/2.793/2.894/0.096 ms
[root@iZ25gc26k8dZ ~]#
[root@iZ25gc26k8dZ ~]#
[root@iZ25gc26k8dZ ~]#
[root@iZ25gc26k8dZ ~]# traceroute www.weibo.com
traceroute to www.weibo.com (180.149.134.141), 30 hops max, 60 byte packets
 1  192.168.1.203 (192.168.1.203)  0.431 ms  0.407 ms  0.411 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  *^C
[root@iZ25gc26k8dZ ~]# 
```

将202的keepalived重新启动

观察到：

1. 203回到backup状态，并移除192.158.1.200的ip
2. 202进入master状态，并接管vip

在204上验证上网效果

可以看到，依然可以ping通，traceroute变回第一跳为202

上面的主备迁移过程，您也可以停机/系统重启的方式模拟宕机，来观察vip的切换。