Subject: Source based routing
Posted by sspt on Tue, 18 Mar 2008 18:56:31 GMT
View Forum Message <> Reply to Message

I've two nics, two networks, two gateways:

Quote:
ip route add default dev eth1 via 10.2.193.129 table 7
ip route add default dev eth0 via 10.3.220.129 table 8

ip rule add from 10.2.193.128/28 table 7
ip rule add from 10.3.220.128/28 table 8

ip route add default via 10.3.220.129 dev eth0

Incoming bandwidth is fine, but outgoing goes thru the default gateway. What am i missing?

Thanks in advance,

Subject: Re: Source based routing
Posted by maratrus on Mon, 24 Mar 2008 09:07:41 GMT
View Forum Message <> Reply to Message

Hi,

May be you should flush route cache? For example, like that:
ip route flush cache

Subject: Re: Source based routing
Posted by sspt on Mon, 24 Mar 2008 18:01:59 GMT
View Forum Message <> Reply to Message

It works for about ten mins then i've to annouce the IPs on eth0 again every 10 mins otherwise the VE loses connectivity

arpsend -U -i VE_IP -c 1 eth0

Subject: Re: Source based routing
Posted by maratrus on Wed, 26 Mar 2008 06:56:51 GMT

Hi,

So, your first problem was solved, wasn't it?
And the second one was appeared. Your HN doesn't respond to arp requests.
So, please
1. Look at all your routing tables. Do they contain the route to reach your VE.
2. Look at "arp -n" output when everything works perfectly. Does output contain the record with VE ip address on both eth0 and eth1 interfaces?
3. Please conduct the previous experiment also when the problem appears.
4. Does the following commands helps when HN doesn't respond to arp requests:
ip neigh add proxy VE_IP dev eth0
ip neigh add proxy VE_IP dev eth1

P.S. and if it is possible please show us the outputs of all previous commands.

Subject: Re: Source based routing
Posted by sspt on Thu, 27 Mar 2008 00:40:34 GMT

All the outputs are the same before and after it stops working

```
[root@node ~]# ip route
69.xxx.xx2.2 dev venet0  scope link
67.xxx.x93.141 dev venet0  scope link
69.xxx.xx0.179 dev venet0  scope link
67.xxx.x93.131 dev venet0  scope link
67.xxx.x93.128/28 dev eth1  proto kernel  scope link  src 67.xxx.x93.130
69.xxx.xx0.128/26 dev eth0  proto kernel  scope link  src 69.xxx.xx0.181
default via 67.xxx.x93.129 dev eth1

[root@node ~]# ip rule
0:      from all lookup 255
32763:  from 69.xxx.xxx.0/27 lookup 8
32764:  from 69.xxx.xx2.0/27 lookup 8
32765:  from 69.xxx.xx0.128/26 lookup 8
32766:  from all lookup main
32767:  from all lookup default

[root@node ~]# arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
69.xxx.xx0.189           ether   00:30:48:76:EC:8C   C                     eth0
69.xxx.xx0.129           ether   00:01:30:BA:94:50   C                     eth0
67.xxx.x93.129           ether   00:01:30:B5:7A:D0   C                     eth1
69.xxx.xx0.186           ether   00:30:48:76:EC:8C   C                     eth0
69.xxx.xx0.182           ether   00:30:48:76:EC:8C   C                     eth0
```

69.xxx.xx0.188 ether 00:30:48:76:EC:8C C eth0
69.xxx.xx0.179 * * MP eth1
69.xxx.xx0.179 * * MP eth0
67.xxx.x93.131 * * MP eth1
67.xxx.x93.131 * * MP eth0
69.xxx.xx2.2 * * MP eth1
69.xxx.xx2.2 * * MP eth0
67.xxx.x93.141 * * MP eth1
67.xxx.x93.141 * * MP eth0

if i run

arpsend -U -i VE_IP -c 1 eth0

It'll work again

Subject: Re: Source based routing
Posted by maratrus on Fri, 28 Mar 2008 12:07:12 GMT
View Forum Message <> Reply to Message

Hi,

Sorry for delay.
Would you be so kind as to give us a little bit more information.
1. What is network configuration on the external node from which we cannot ping VE.
("ip a l"; "ip rule l"; "ip route l" from each table on the external node not only from main table)
2. "ip route l" from each table on the HN not only from main table.
3. Please conduct such experiment:
when it stops working:
a) ping your VE from the external node
b) use tcpdump on each network interface on that node and show it output
c) use tcpdump on eth0 and eth1 interfaces on HN and also show the output
3. "sysctl -a | grep rp_filter" from HN.

Thank You very much!

Subject: Re: Source based routing
Posted by sspt on Sat, 29 Mar 2008 03:31:48 GMT
View Forum Message <> Reply to Message

It happens from any external box which tries to reach the VE
External BOX
ping to VE
21:50:09.342794 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >

67.xxx.xxx.140: icmp 64: echo request seq 0
21:50:10.351723 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 1
21:50:11.360495 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 2
21:50:12.360282 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 3
21:50:13.360058 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 4
21:50:14.359834 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 5
21:50:15.359627 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 6
21:50:16.368357 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 7
21:50:17.368154 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 8
21:50:18.367927 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 9
21:50:19.367700 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 10
21:50:20.367476 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 11
21:50:21.376229 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 12
21:50:22.376023 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 13
21:50:23.379326 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.140: icmp 64: echo request seq 14

ping to node
21:59:08.040177 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.130: icmp 64: echo request seq 0
21:59:08.113576 IP (tos 0x0, ttl 40, id 56698, offset 0, flags [none], proto 1, length: 84)
67.xxx.xxx.130 > xxx.110.83.42: icmp 64: echo reply seq 0
21:59:09.040162 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.130: icmp 64: echo request seq 1
21:59:09.113478 IP (tos 0x0, ttl 40, id 56699, offset 0, flags [none], proto 1, length: 84)
67.xxx.xxx.130 > xxx.110.83.42: icmp 64: echo reply seq 1
21:59:10.039948 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.130: icmp 64: echo request seq 2
21:59:10.113954 IP (tos 0x0, ttl 40, id 56700, offset 0, flags [none], proto 1, length: 84)
67.xxx.xxx.130 > xxx.110.83.42: icmp 64: echo reply seq 2
21:59:11.039702 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.130: icmp 64: echo request seq 3
21:59:11.113510 IP (tos 0x0, ttl 40, id 56701, offset 0, flags [none], proto 1, length: 84)
67.xxx.xxx.130 > xxx.110.83.42: icmp 64: echo reply seq 3
21:59:12.048458 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >

67.xxx.xxx.130: icmp 64: echo request seq 4
21:59:12.121922 IP (tos 0x0, ttl 40, id 56702, offset 0, flags [none], proto 1, length: 84)
67.xxx.xxx.130 > xxx.110.83.42: icmp 64: echo reply seq 4
21:59:13.057232 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.130: icmp 64: echo request seq 5
21:59:13.130931 IP (tos 0x0, ttl 40, id 56703, offset 0, flags [none], proto 1, length: 84)
67.xxx.xxx.130 > xxx.110.83.42: icmp 64: echo reply seq 5
21:59:14.057046 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.130: icmp 64: echo request seq 6
21:59:14.129665 IP (tos 0x0, ttl 40, id 56704, offset 0, flags [none], proto 1, length: 84)
67.xxx.xxx.130 > xxx.110.83.42: icmp 64: echo reply seq 6
21:59:15.056793 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.130: icmp 64: echo request seq 7
21:59:15.129523 IP (tos 0x0, ttl 40, id 56705, offset 0, flags [none], proto 1, length: 84)
67.xxx.xxx.130 > xxx.110.83.42: icmp 64: echo reply seq 7
21:59:16.056568 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.130: icmp 64: echo request seq 8
21:59:16.141422 IP (tos 0x0, ttl 40, id 56706, offset 0, flags [none], proto 1, length: 84)
67.xxx.xxx.130 > xxx.110.83.42: icmp 64: echo reply seq 8
21:59:17.056378 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.130: icmp 64: echo request seq 9
21:59:17.129063 IP (tos 0x0, ttl 40, id 56707, offset 0, flags [none], proto 1, length: 84)
67.xxx.xxx.130 > xxx.110.83.42: icmp 64: echo reply seq 9
21:59:18.056139 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto 1, length: 84) xxx.110.83.42 >
67.xxx.xxx.130: icmp 64: echo request seq 10
21:59:18.128998 IP (tos 0x0, ttl 40, id 56708, offset 0, flags [none], proto 1, length: 84)
67.xxx.xxx.130 > xxx.110.83.42: icmp 64: echo reply seq 10

Node

While we ping VE IP
Nothing, a traceroute won't reach the VE IP, it dies at the switch
So i've ran '/usr/sbin/arpsend -U -i 67.xxx.xxx.140 -c 1 eth1' and then:
20:20:07.121016 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.140: ICMP echo request, id 60987, seq 81, length 64
20:20:07.121108 IP (tos 0x0, ttl 64, id 56946, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.140 > xxx.110.83.42: ICMP echo reply, id 60987, seq 81, length 64
20:20:07.121116 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.140: ICMP echo request, id 60987, seq 82, length 64
20:20:07.121142 IP (tos 0x0, ttl 64, id 56947, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.140 > xxx.110.83.42: ICMP echo reply, id 60987, seq 82, length 64
20:20:07.121643 IP (tos 0xc0, ttl 64, id 6287, offset 0, flags [none], proto: ICMP (1), length: 576)
67.xxx.xxx.140 > 198.32.64.12: ICMP 67.xxx.xxx.140 udp port 32915 unreachable, length 556
20:20:07.121763 IP (tos 0xc0, ttl 64, id 2440, offset 0, flags [none], proto: ICMP (1), length: 163)
67.xxx.xxx.140 > 192.228.79.201: ICMP 67.xxx.xxx.140 udp port 32915 unreachable, length 143
20:20:07.122048 IP (tos 0xc0, ttl 64, id 641, offset 0, flags [none], proto: ICMP (1), length: 341)
67.xxx.xxx.140 > 192.203.230.10: ICMP 67.xxx.xxx.140 udp port 32915 unreachable, length 321
20:20:08.052997 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)

xxx.110.83.42 > 67.xxx.xxx.140: ICMP echo request, id 63035, seq 8, length 64
20:20:08.053030 IP (tos 0x0, ttl 64, id 56948, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.140 > xxx.110.83.42: ICMP echo reply, id 63035, seq 8, length 64
20:20:09.048410 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.140: ICMP echo request, id 63035, seq 9, length 64
20:20:09.048439 IP (tos 0x0, ttl 64, id 56949, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.140 > xxx.110.83.42: ICMP echo reply, id 63035, seq 9, length 64
20:20:10.047784 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.140: ICMP echo request, id 63035, seq 10, length 64
20:20:10.047815 IP (tos 0x0, ttl 64, id 56950, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.140 > xxx.110.83.42: ICMP echo reply, id 63035, seq 10, length 64
20:20:11.047685 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.140: ICMP echo request, id 63035, seq 11, length 64
20:20:11.047713 IP (tos 0x0, ttl 64, id 56951, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.140 > xxx.110.83.42: ICMP echo reply, id 63035, seq 11, length 64
20:20:12.047350 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.140: ICMP echo request, id 63035, seq 12, length 64
20:20:12.047379 IP (tos 0x0, ttl 64, id 56952, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.140 > xxx.110.83.42: ICMP echo reply, id 63035, seq 12, length 64
20:20:13.047414 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.140: ICMP echo request, id 63035, seq 13, length 64
20:20:13.047442 IP (tos 0x0, ttl 64, id 56953, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.140 > xxx.110.83.42: ICMP echo reply, id 63035, seq 13, length 64
20:20:14.047377 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.140: ICMP echo request, id 63035, seq 14, length 64
20:20:14.047406 IP (tos 0x0, ttl 64, id 56954, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.140 > xxx.110.83.42: ICMP echo reply, id 63035, seq 14, length 64
20:20:15.048324 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.140: ICMP echo request, id 63035, seq 15, length 64
20:20:15.048352 IP (tos 0x0, ttl 64, id 56955, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.140 > xxx.110.83.42: ICMP echo reply, id 63035, seq 15, length 64
20:20:16.046954 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.140: ICMP echo request, id 63035, seq 16, length 64
A few minutes later, nothing again

While we ping node IP
20:17:43.646730 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.130: ICMP echo request, id 20283, seq 0, length 64
20:17:43.646757 IP (tos 0x0, ttl 64, id 56897, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.130 > xxx.110.83.42: ICMP echo reply, id 20283, seq 0, length 64
20:17:44.647210 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.130: ICMP echo request, id 20283, seq 1, length 64
20:17:44.647225 IP (tos 0x0, ttl 64, id 56898, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.130 > xxx.110.83.42: ICMP echo reply, id 20283, seq 1, length 64
20:17:45.647314 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.130: ICMP echo request, id 20283, seq 2, length 64
20:17:45.647327 IP (tos 0x0, ttl 64, id 56899, offset 0, flags [none], proto: ICMP (1), length: 84)

67.xxx.xxx.130 > xxx.110.83.42: ICMP echo reply, id 20283, seq 2, length 64
20:17:46.773103 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.130: ICMP echo request, id 20283, seq 3, length 64
20:17:46.773115 IP (tos 0x0, ttl 64, id 56900, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.130 > xxx.110.83.42: ICMP echo reply, id 20283, seq 3, length 64
20:17:47.647223 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.130: ICMP echo request, id 20283, seq 4, length 64
20:17:47.647237 IP (tos 0x0, ttl 64, id 56901, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.130 > xxx.110.83.42: ICMP echo reply, id 20283, seq 4, length 64
20:17:48.647086 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.130: ICMP echo request, id 20283, seq 5, length 64
20:17:48.647099 IP (tos 0x0, ttl 64, id 56902, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.130 > xxx.110.83.42: ICMP echo reply, id 20283, seq 5, length 64
20:17:49.647207 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.130: ICMP echo request, id 20283, seq 6, length 64
20:17:49.647221 IP (tos 0x0, ttl 64, id 56903, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.130 > xxx.110.83.42: ICMP echo reply, id 20283, seq 6, length 64
20:17:50.647366 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.130: ICMP echo request, id 20283, seq 7, length 64
20:17:50.647380 IP (tos 0x0, ttl 64, id 56904, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.130 > xxx.110.83.42: ICMP echo reply, id 20283, seq 7, length 64
20:17:51.647339 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.130: ICMP echo request, id 20283, seq 8, length 64
20:17:51.647355 IP (tos 0x0, ttl 64, id 56905, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.130 > xxx.110.83.42: ICMP echo reply, id 20283, seq 8, length 64
20:17:52.648187 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.130: ICMP echo request, id 20283, seq 9, length 64
20:17:52.648199 IP (tos 0x0, ttl 64, id 56906, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.130 > xxx.110.83.42: ICMP echo reply, id 20283, seq 9, length 64
20:17:53.647165 IP (tos 0x0, ttl 40, id 0, offset 0, flags [DF], proto: ICMP (1), length: 84)
xxx.110.83.42 > 67.xxx.xxx.130: ICMP echo request, id 20283, seq 10, length 64
20:17:53.647177 IP (tos 0x0, ttl 64, id 56907, offset 0, flags [none], proto: ICMP (1), length: 84)
67.xxx.xxx.130 > xxx.110.83.42: ICMP echo reply, id 20283, seq 10, length 64

[root@node ~]# ip route l table 7
default via 67.xxx.xxx.129 dev eth1

[root@node ~]# ip route l table 8
default via 69.xxx.xx0.129 dev eth0

[root@node ~]# ip route l table main
67.xxx.xxx.140 dev venet0 scope link
69.xxx.xx0.179 dev venet0 scope link
69.xxx.xx0.187 dev venet0 scope link
67.xxx.xxx.128/28 dev eth1 proto kernel scope link src 67.xxx.xxx.130

69.xxx.xx0.128/26 dev eth0 proto kernel scope link src 69.xxx.xx0.181
169.254.0.0/16 dev eth1 scope link
default via 69.xxx.xx0.129 dev eth0

[root@node ~]# ip route l table 255
broadcast 127.255.255.255 dev lo proto kernel scope link src 127.0.0.1
local 69.xxx.xx0.181 dev eth0 proto kernel scope host src 69.xxx.xx0.181
broadcast 67.xxx.xxx.142 dev eth1 proto kernel scope link src 67.xxx.xxx.130
broadcast 67.xxx.xxx.143 dev eth1 proto kernel scope link src 67.xxx.xxx.130
broadcast 69.xxx.xx0.128 dev eth0 proto kernel scope link src 69.xxx.xx0.181
broadcast 67.xxx.xxx.128 dev eth1 proto kernel scope link src 67.xxx.xxx.130
broadcast 69.xxx.xx0.191 dev eth0 proto kernel scope link src 69.xxx.xx0.181
local 67.xxx.xxx.130 dev eth1 proto kernel scope host src 67.xxx.xxx.130
broadcast 127.0.0.0 dev lo proto kernel scope link src 127.0.0.1
local 127.0.0.1 dev lo proto kernel scope host src 127.0.0.1
local 127.0.0.0/8 dev lo proto kernel scope host src 127.0.0.1

[root@node ~]# sysctl -a | grep rp_filter
net.ipv4.conf.venet0.arp_filter = 0
net.ipv4.conf.venet0.rp_filter = 1
net.ipv4.conf.eth1.arp_filter = 0
net.ipv4.conf.eth1.rp_filter = 1
net.ipv4.conf.eth0.arp_filter = 0
net.ipv4.conf.eth0.rp_filter = 1
net.ipv4.conf.lo.arp_filter = 0
net.ipv4.conf.lo.rp_filter = 0
net.ipv4.conf.default.arp_filter = 0
net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.all.arp_filter = 0
net.ipv4.conf.all.rp_filter = 0

Thank you for all the support

Subject: Re: Source based routing
Posted by maratrus on Fri, 04 Apr 2008 10:57:58 GMT
View Forum Message <> Reply to Message

Hi,

I'm terribly sorry for delay.
Did you solve this problem?

1. You wrote
Quote:

if i run

arpsend -U -i VE_IP -c 1 eth0

So i've ran '/usr/sbin/arpsend -U -i 67.xxx.xxx.140 -c 1 eth1'

So does it matter if you use eth0 or eth1 interfaces?

2. As a suggestion:
When it stops working please look at arp-table on the external node ("arp -n") and try to find the record with the VE_IP address (67.xxx.xxx.140) then please try to "ip r get 67.xxx.xxx.140". So you can find out where this package goes, I suppose that it goes through the default gateway (default gateway of the external node). You can observe where the package goes with tcpdump utility with the "-e" option which allows us to look at MAC addresses. That is why I think we should look at the default gateway (of the external node) behavior.

3.
Quote:

Nothing, a traceroute won't reach the VE IP, it dies at the switch

What interface did you listen with tcpdump. Try to specify it with -i parameter. Please listen both eth0 and eth1 interfaces. Does nothing come to both of them?

4. And another question:
Do your eth0 and eth1 interfaces connect to different switches?
Are 69.xxx.xx0.129 and 67.xxx.xxx.129 different nodes?

Subject: Re: Source based routing
Posted by sspt on Fri, 04 Apr 2008 12:00:04 GMT
View Forum Message <> Reply to Message

External node

```
[root@node02 ~]# arp -n | grep 67.xxx.xxx.140
67.xxx.xxx.140                (incomplete)                              eth1
```

```
[root@node02 ~]# ip r get 67.xxx.xxx.140
67.xxx.xxx.140 dev eth1  src 67.xxx.xxx.131
    cache  mtu 1500 advmss 1460 hoplimit 64
```

```
[root@node02 ~]# tcpdump -i eth1 -v -e
```

tcpdump: listening on eth1, link-type EN10MB (Ethernet), capture size 96 bytes
04:52:37.660602 00:30:48:76:ec:8d (oui Unknown) > Broadcast, ethertype ARP (0x0806), length 42: arp who-has 67.xxx.xxx.140 tell 67.xxx.xxx.131
04:52:38.660574 00:30:48:76:ec:8d (oui Unknown) > Broadcast, ethertype ARP (0x0806), length 42: arp who-has 67.xxx.xxx.140 tell 67.xxx.xxx.131
04:52:39.659542 00:30:48:76:ec:8d (oui Unknown) > Broadcast, ethertype ARP (0x0806), length 42: arp who-has 67.xxx.xxx.140 tell 67.xxx.xxx.131
04:52:47.000300 00:30:48:76:ec:8d (oui Unknown) > Broadcast, ethertype ARP (0x0806), length 42: arp who-has 67.xxx.xxx.140 tell 67.xxx.xxx.131
04:52:47.999273 00:30:48:76:ec:8d (oui Unknown) > Broadcast, ethertype ARP (0x0806), length 42: arp who-has 67.xxx.xxx.140 tell 67.xxx.xxx.131
04:52:48.999239 00:30:48:76:ec:8d (oui Unknown) > Broadcast, ethertype ARP (0x0806), length 42: arp who-has 67.xxx.xxx.140 tell 67.xxx.xxx.131
04:52:50.999177 00:30:48:76:ec:8d (oui Unknown) > Broadcast, ethertype ARP (0x0806), length 42: arp who-has 67.xxx.xxx.140 tell 67.xxx.xxx.131
04:52:51.999142 00:30:48:76:ec:8d (oui Unknown) > Broadcast, ethertype ARP (0x0806), length 42: arp who-has 67.xxx.xxx.140 tell 67.xxx.xxx.131
04:52:52.999113 00:30:48:76:ec:8d (oui Unknown) > Broadcast, ethertype ARP (0x0806), length 42: arp who-has 67.xxx.xxx.140 tell 67.xxx.xxx.131
04:52:54.557053 00:30:48:76:ec:8d (oui Unknown) > Broadcast, ethertype ARP (0x0806), length 42: arp who-has 67.xxx.xxx.140 tell 67.xxx.xxx.131
04:52:55.557026 00:30:48:76:ec:8d (oui Unknown) > Broadcast, ethertype ARP (0x0806), length 42: arp who-has 67.xxx.xxx.140 tell 67.xxx.xxx.131

Eth0 is connected to a switch on network 1 and eth1 on network2 (different bandwidth providers)

Subject: Re: Source based routing
Posted by maratrus on Fri, 04 Apr 2008 12:35:42 GMT
View Forum Message <> Reply to Message

Thanks.

"arp -n" on HN at that moment! It should contain the record like this

67.xxx.xxx.140	*	*	MP	eth1
67.xxx.xxx.140	*	*	MP	eth0

Subject: Re: Source based routing
Posted by maratrus on Fri, 04 Apr 2008 13:15:45 GMT
View Forum Message <> Reply to Message

Quote:
[root@node ~]# ip route l table 7
default via 67.xxx.xxx.129 dev eth1

but
Quote:
[root@node ~]# ip rule
0:	from all lookup 255
32763:	from 69.xxx.xxx.0/27 lookup 8
32764:	from 69.xxx.xx2.0/27 lookup 8
32765:	from 69.xxx.xx0.128/26 lookup 8
32766:	from all lookup main
32767:	from all lookup default

doesn't contain table 7.
Have you changed configuration?
Could you please show "ip rule" again.

Subject: Re: Source based routing
Posted by sspt on Fri, 04 Apr 2008 14:45:51 GMT
View Forum Message <> Reply to Message

Sorry, i've changed the default gw since the first post, here are the current stuff:

[root@node ~]# ip route
67.xxx.xxx.140 dev venet0 scope link
69.xx.xxx.179 dev venet0 scope link
69.xx.xxx.187 dev venet0 scope link
67.xxx.xxx.128/28 dev eth1 proto kernel scope link src 67.xxx.xxx.130
69.xx.xxx.128/26 dev eth0 proto kernel scope link src 69.xx.xxx.181
169.254.0.0/16 dev eth1 scope link
default via 69.xx.xxx.129 dev eth0

[root@node ~]# ip rule
0:	from all lookup 255
32765:	from 67.xxx.xxx.128/28 lookup 7
32766:	from all lookup main
32767:	from all lookup default

[root@node ~]# ip route l table 7
default via 67.xxx.xxx.129 dev eth1

arp -n gives the output

67.220.193.140	*	*	MP	eth1
67.220.193.140	*	*	MP	eth0

[root@node ~]# ip route l table 7
default via 67.xxx.xxx.129 dev eth1

Subject: Re: Source based routing
Posted by maratrus on Fri, 04 Apr 2008 15:30:37 GMT
View Forum Message <> Reply to Message

Hi,

I think I know the reason.
try to do the following:
ip r add 67.xxx.xxx.140 dev venet0 table 7

Subject: Re: Source based routing
Posted by sspt on Sat, 05 Apr 2008 12:05:54 GMT
View Forum Message <> Reply to Message

Hi,
It worked

I really appreciate the time you spent helping me with this.

Btw, should the wiki be updated?

/sbin/ip rule add from $IP table $TBL
/sbin/ip route add default dev eth0 via $GW table $TBL

To

/sbin/ip rule add from $IP table $TBL
/sbin/ip route add default dev eth0 via $GW table $TBL
/sbin/ip r add $IP dev venet0 table $TBL

Thank you
