# Source based routing / policy routing

22.02.25

Pegasi Knowledge
https://ghost.pegasi.fi/wiki/

# Table of Contents

# Source based routing / policy routing

**Update:** Added Network Manager support due to contradicting advice out there. Below is tested and working solution I am using myself.

Idea is to enable multiple network device server to talk to right gateway depending on what address the server has been called with. For example you may have an internal NATted IP and a public service IP and you wish to conduct all your administrational stuff with the NATted interface leaving the public network device to serve HTTP to the public internet.

## Option A: Network Manager config

Network Manager requires routing and dispatcher services. In this example your box would have IP config like this:

- public IP address 123.123.123.11, dev eth0
- private IP address 172.16.3.3, dev eth1, default gateway 172.16.3.1

With this config when someone from outside world connects to your public IP 123.123.123.11 the server tries to answer using your default gateway 172.16.3.1 which fails as the response is sent with IP address 172.16.3.3 since the address is in the same network. What we want is to answer the call using public IP 123.123.123.11 and gateway 123.123.123.1.

Below is the whole shebang in one dump.

```
dnf install NetworkManager-config-routing-rules
systemctl enable NetworkManager-dispatcher.service
systemctl start NetworkManager-dispatcher.service
echo '1       internal'                                          >>
/etc/iproute2/rt_tables
echo '2       public'                                            >>
/etc/iproute2/rt_tables
echo "from 123.123.123.11 table public"                          >
/etc/sysconfig/network-scripts/rule-eth0
echo "to 123.123.123.11 table public"                            >>
/etc/sysconfig/network-scripts/rule-eth0
echo "123.123.123.0/27 dev eth0 src 123.123.123.11 table public" >
/etc/sysconfig/network-scripts/route-eth0
echo "default via 123.123.123.1 dev eth0 table public"           >>
/etc/sysconfig/network-scripts/route-eth0
restorecon -r /etc/sysconfig/network-scripts
nmcli con reload
nmcli con down eth0
nmcli con up eth0
ip route flush table public
```

## Option B: Initscripts without Network Manager

This is the vintage model without Network Manager, using sysv initscripts.

### Remove NetworkManager

NetworkManager is always full of surprises. Some day when you update your box remotely you may find yourself cut out from your server. And policy routing does not work when your interfaces are NetworkManager controlled.

Firstly make your interfaces free from NetworkManager by adding a line

```
NM_CONTROLLED=no
```

to your /etc/sysconfig/network-scripts/ifcfg-* files.

Then erase NetworkManager with command

```
yum erase NetworkManager
```

### Create route tables in rt_tables

Edit /etc/iproute2/rt_tables and add the following (note example table names)

```
1       internal
2       public
```

### Create routes

We use networks 1.2.3.0/24 and 172.16.10.0/24 with devices eth0 and eth1.

Set /etc/sysconfig/network-scripts/route-eth0 to

```
1.2.3.0/24 dev eth0 src 1.2.3.123 table public
default via 1.2.3.1 dev eth0 table public
```

Set /etc/sysconfig/network-scripts/route-eth1 to

```
172.16.10.0/24 dev eth1 src 172.16.10.123 table internal
default via 172.16.10.1 dev eth1 table internal
```

## Create rules

Set /etc/sysconfig/network-scripts/rule-eth0 to

```
from 1.2.3.123 table public
```

Set /etc/sysconfig/network-scripts/rule-eth1 to

```
from 172.16.10.123 table internal
```

## Test

You can reboot or try ifdown + ifup ethN but better be sure you have console access locally or via virtual console.

```
ip rule show
ip route show table internal
ip route show table public
```

Also don't forget to update your iptables and other stuff.