
Subject: HowTo route/use a public IP to a VE ?
Posted by [BeNe](#) on Mon, 08 Jun 2009 14:16:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello together,

i need your help
I want to setup a OpenVZ Server on my Root-Server.
The Kernel is running:

```
srv01:/# uname -r  
2.6.26-2-openvz-amd64
```

and i can create my VM (i use webVZ)
My big problem is the understanding, how i get my public IP's into the VE.

I read:
<http://wiki.openvz.org/Venet>
<http://wiki.openvz.org/Veth>
http://wiki.openvz.org/Using_NAT_for_container_with_private_IPs#Prerequisites

But don't know if must use venet or veth ?!
I have 20 public IP and i want to use them in the VE.
So there is no need to use private IP, i want use my public IPs.

The Problem is now that i can't ping from or into the VE.
Here is Hosts IPTable:

```
srv01:/# iptables -t nat -L && iptables -t filter -L && iptables -t mangle -L  
Chain PREROUTING (policy ACCEPT)  
target  prot opt source          destination  
  
Chain POSTROUTING (policy ACCEPT)  
target  prot opt source          destination  
  
Chain OUTPUT (policy ACCEPT)  
target  prot opt source          destination  
Chain INPUT (policy ACCEPT)  
target  prot opt source          destination  
  
Chain FORWARD (policy ACCEPT)  
target  prot opt source          destination  
  
Chain OUTPUT (policy ACCEPT)  
target  prot opt source          destination  
Chain PREROUTING (policy ACCEPT)  
target  prot opt source          destination
```

Chain INPUT (policy ACCEPT)

target prot opt source destination

Chain FORWARD (policy ACCEPT)

target prot opt source destination

Chain OUTPUT (policy ACCEPT)

target prot opt source destination

Chain POSTROUTING (policy ACCEPT)

target prot opt source destination

Here is the Hosts ifconfig:

srv01:/# ifconfig

eth0 Link encap:Ethernet HWaddr 00:19:db:d5:83:5e
inet addr:85.31.186.103 Bcast:85.31.187.255 Mask:255.255.254.0
inet6 addr: fe80::219:dbff:fed5:835e/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:426861 errors:0 dropped:0 overruns:0 frame:0
TX packets:12544 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:51190688 (48.8 MiB) TX bytes:3044511 (2.9 MiB)
Interrupt:23 Base address:0xd800

eth0:0 Link encap:Ethernet HWaddr 00:19:db:d5:83:5e
inet addr:91.143.83.160 Bcast:91.143.83.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
Interrupt:23 Base address:0xd800

eth0:1 Link encap:Ethernet HWaddr 00:19:db:d5:83:5e
inet addr:91.143.83.168 Bcast:91.143.83.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
Interrupt:23 Base address:0xd800

lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:17 errors:0 dropped:0 overruns:0 frame:0
TX packets:17 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:1372 (1.3 KiB) TX bytes:1372 (1.3 KiB)

venet0 Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
UP BROADCAST POINTOPOINT RUNNING NOARP MTU:1500 Metric:1
RX packets:21 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:1764 (1.7 KiB) TX bytes:0 (0.0 B)
```

```
veth101.0 Link encap:Ethernet HWaddr 00:18:51:bd:9f:65
inet6 addr: fe80::218:51ff:febd:9f65/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:6 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:384 (384.0 B) TX bytes:0 (0.0 B)
```

```
ip r l && ip a l
```

```
srv01:/# ip r l && ip a l
91.143.83.160 dev venet0 scope link
192.168.0.101 dev veth101.0 scope link
91.143.83.0/24 dev eth0 proto kernel scope link src 91.143.83.160
85.31.186.0/23 dev eth0 proto kernel scope link src 85.31.186.103
default via 85.31.186.1 dev eth0 src 91.143.83.168
default via 85.31.186.1 dev eth0 src 91.143.83.160
default via 85.31.186.1 dev eth0
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state
UNKNOWN qlen 1000
    link/ether 00:19:db:d5:83:5e brd ff:ff:ff:ff:ff:ff
    inet 85.31.186.103/23 brd 85.31.187.255 scope global eth0
    inet 91.143.83.160/24 brd 91.143.83.255 scope global eth0:0
    inet 91.143.83.168/24 brd 91.143.83.255 scope global secondary eth0:1
    inet6 fe80::219:dbff:fed5:835e/64 scope link
        valid_lft forever preferred_lft forever
3: venet0: <BROADCAST,POINTOPOINT,NOARP,UP,LOWER_UP> mtu 1500 qdisc noqueue
state UNKNOWN
    link/void
8: veth101.0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state
UNKNOWN
    link/ether 00:18:51:bd:9f:65 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::218:51ff:febd:9f65/64 scope link
        valid_lft forever preferred_lft forever
```

```
My sysctl.conf
```

```
srv01:/# cat /etc/sysctl.conf
#
# /etc/sysctl.conf - Configuration file for setting system variables
```

```
# See /etc/sysctl.d/ for additional system variables
# See sysctl.conf (5) for information.
#

#kernel.domainname = example.com

# Uncomment the following to stop low-level messages on console
#kernel.printk = 4 4 1 7

#####3
# Functions previously found in netbase
#

# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# This disables TCP Window Scaling (http://lkml.org/lkml/2008/2/5/167),
# and is not recommended.
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
#net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
#net.ipv6.conf.all.forwarding=1

#####
# Additional settings - these settings can improve the network
# security of the host and prevent against some network attacks
# including spoofing attacks and man in the middle attacks through
# redirection. Some network environments, however, require that these
# settings are disabled so review and enable them as needed.
#
# Ignore ICMP broadcasts
#net.ipv4.icmp_echo_ignore_broadcasts = 1
#
# Ignore bogus ICMP errors
#net.ipv4.icmp_ignore_bogus_error_responses = 1
#
# Do not accept ICMP redirects (prevent MITM attacks)
#net.ipv4.conf.all.accept_redirects = 0
#net.ipv6.conf.all.accept_redirects = 0
# _or_
```

```
# Accept ICMP redirects only for gateways listed in our default
# gateway list (enabled by default)
# net.ipv4.conf.all.secure_redirects = 1
#
# Do not send ICMP redirects (we are not a router)
#net.ipv4.conf.all.send_redirects = 0
#
# Do not accept IP source route packets (we are not a router)
#net.ipv4.conf.all.accept_source_route = 0
#net.ipv6.conf.all.accept_source_route = 0
#
# Log Martian Packets
#net.ipv4.conf.all.log_martians = 1
#
# The contents of /proc/<pid>/maps and smaps files are only visible to
# readers that are allowed to ptrace() the process
# kernel.maps_protect = 1

#-- OpenVZ begin --#

net.ipv4.icmp_echo_ignore_broadcasts = 1

# On Hardware Node we generally need
# packet forwarding enabled and proxy arp disabled
net.ipv4.conf.default.forwarding = 1
net.ipv4.conf.default.proxy_arp = 0

#net.ipv4.ip_forward = 1

# Enables source route verification
net.ipv4.conf.all.rp_filter = 1

# Enables the magic-sysrq key
kernel.sysrq = 1

# TCP Explicit Congestion Notification
#net.ipv4.tcp_ecn = 0

# we do not want all our interfaces to send redirects
net.ipv4.conf.default.send_redirects = 1
net.ipv4.conf.all.send_redirects = 0

net.ipv4.conf.eth0.proxy_arp = 1

#-- OpenVZ end --#
```

My routes on the Host:

```

srv01:/# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
91-143-83-160.b * 255.255.255.255 UH 0 0 0 venet0
192.168.0.101 * 255.255.255.255 UH 0 0 0 veth101.0
91.143.83.0 * 255.255.255.0 U 0 0 0 eth0
85.31.186.0 * 255.255.254.0 U 0 0 0 eth0
default gw-85-31-186.je 0.0.0.0 UG 0 0 0 eth0
default gw-85-31-186.je 0.0.0.0 UG 0 0 0 eth0
default gw-85-31-186.je 0.0.0.0 UG 0 0 0 eth0

```

The IP 91.143.83.160 is a public IP i want to use in my VE.
 So must i use bridging or must i use NAT ?

Here is the ifconfig of my VE 101

```

ox:/# ifconfig
eth0 Link encap:Ethernet HWaddr 00:18:51:88:01:2a
      inet addr:192.168.0.101 Bcast:0.0.0.0 Mask:255.255.255.255
      inet6 addr: fe80::218:51ff:fe88:12a/64 Scope:Link
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
      RX packets:0 errors:0 dropped:0 overruns:0 frame:0
      TX packets:6 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:0
      RX bytes:0 (0.0 B) TX bytes:384 (384.0 B)

lo Link encap:Local Loopback
   inet addr:127.0.0.1 Mask:255.0.0.0
   inet6 addr: ::1/128 Scope:Host
   UP LOOPBACK RUNNING MTU:16436 Metric:1
   RX packets:10 errors:0 dropped:0 overruns:0 frame:0
   TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
   collisions:0 txqueuelen:0
   RX bytes:500 (500.0 B) TX bytes:500 (500.0 B)

venet0 Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
      UP BROADCAST POINTOPOINT RUNNING NOARP MTU:1500 Metric:1
      RX packets:0 errors:0 dropped:0 overruns:0 frame:0
      TX packets:16 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:0
      RX bytes:0 (0.0 B) TX bytes:1344 (1.3 KiB)

venet0:0 Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
      inet addr:91.143.83.160 P-t-P:91.143.83.160 Bcast:0.0.0.0 Mask:255.255.255.255
      UP BROADCAST POINTOPOINT RUNNING NOARP MTU:1500 Metric:1

```

Is this internal IP 192.168.0.101 need if i use public IP's ?

IPTable for the VE:

```
ox:/# iptables -t nat -L && iptables -t filter -L && iptables -t mangle -L
```

Chain PREROUTING (policy ACCEPT)

```
target prot opt source destination
```

Chain POSTROUTING (policy ACCEPT)

```
target prot opt source destination
```

Chain OUTPUT (policy ACCEPT)

```
target prot opt source destination
```

Chain INPUT (policy ACCEPT)

```
target prot opt source destination
```

Chain FORWARD (policy ACCEPT)

```
target prot opt source destination
```

Chain OUTPUT (policy ACCEPT)

```
target prot opt source destination
```

Chain PREROUTING (policy ACCEPT)

```
target prot opt source destination
```

Chain INPUT (policy ACCEPT)

```
target prot opt source destination
```

Chain FORWARD (policy ACCEPT)

```
target prot opt source destination
```

Chain OUTPUT (policy ACCEPT)

```
target prot opt source destination
```

Chain POSTROUTING (policy ACCEPT)

```
target prot opt source destination
```

Is some more info needed ?

Big thanks for any help!

Subject: Re: HowTo route/use a public IP to a VE ?

Posted by [maratrus](#) on Tue, 09 Jun 2009 05:51:23 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi,

you've read useful articles.

Did you follow instructions described there?

There are some things which are not described there but which you did by yourself.

1.

Quote:

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state
UNKNOWN qlen 1000
    link/ether 00:19:db:d5:83:5e brd ff:ff:ff:ff:ff:ff
    inet 85.31.186.103/23 brd 85.31.187.255 scope global eth0
    inet 91.143.83.160/24 brd 91.143.83.255 scope global eth0:0
```

2.

Quote:

```
# ip r l && ip a l
91.143.83.160 dev venet0 scope link
192.168.0.101 dev veth101.0 scope link
91.143.83.0/24 dev eth0 proto kernel scope link src 91.143.83.160
85.31.186.0/23 dev eth0 proto kernel scope link src 85.31.186.103
default via 85.31.186.1 dev eth0 src 91.143.83.168
default via 85.31.186.1 dev eth0 src 91.143.83.160
default via 85.31.186.1 dev eth0
```

3. Assign NEIGHBOUR_DEVS variable to "all" in /etc/vz/vz.conf file.

There are a lot of redundant settings that are not needed in case of venet0 configuration. I'd recommend you to do it from scratch only following the instructions described in articles. Thank you.

Subject: Re: HowTo route/use a public IP to a VE ?

Posted by [BeNe](#) on Tue, 09 Jun 2009 05:56:53 GMT

[View Forum Message](#) <> [Reply to Message](#)

Thanks for your reply and help!

I'd recommend you to do it from scratch only following the instructions described in articles. I will do so and try it again.

I've added the IP "91.143.83.160" for the VE on my Host as Alias as my Provider said:

```
srv01:~# cat /etc/network/interfaces
# auto-generated system-config
# part of KDCTRL-NOC and FADC
```

```
auto lo eth0
auto eth0:0
```



```
auto eth0:1
iface lo inet loopback

iface eth0 inet static
address 85.31.186.103
netmask 255.255.254.0
broadcast 85.31.187.255
gateway 85.31.186.1
```

```
iface eth0:0 inet static
address 91.143.83.160
netmask 255.255.255.0
gateway 85.31.186.1
```

```
iface eth0:1 inet static
address 91.143.83.168
netmask 255.255.255.0
gateway 85.31.186.1
```

So you mean this is the Error ?
That the IP is set as Alias ?

Quote:3. Assign NEIGHBOUR_DEVS variable to "all" in /etc/vz/vz.conf file.
Done!

Subject: Re: HowTo route/use a public IP to a VE ?
Posted by [maratrus](#) on Tue, 09 Jun 2009 06:03:41 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

Quote:
I've added the IP "91.143.83.160" for the VE on my Host as Alias

You shouldn't do that.

Subject: Re: HowTo route/use a public IP to a VE ?
Posted by [BeNe](#) on Tue, 09 Jun 2009 14:40:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thanks for help/info!
That was exactly my (_understanding_) problem.

Subject: Re: HowTo route/use a public IP to a VE ?
Posted by [Balbuzard](#) on Fri, 12 Jun 2009 11:24:17 GMT
[View Forum Message](#) <> [Reply to Message](#)

Yes, why is there a problem with the fact of setting the VPS IP-address as aliases in the host? In this way, the host could send the packets to the appropriate VPS via a reverse proxy (Apache provides something like this).

But with this way, the VPS has in fact a private address and a translation has to be made by the HN;

Maybe the best way should be to set up a bridge, but I have not read enough to know how to do it (I only do NAT or port forwarding) I am still interested by this discussion!

Thanks!
