
Subject: avahi-daemon and service discovery

Posted by [heruan](#) on Wed, 27 Aug 2008 09:33:08 GMT

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

[HN: Debian Lenny 2.6.18-12-fza-686, vzctl 3.0.22]

I have created a Debian VE on which there's a Firefly media server (mt-daapd). Firefly uses avahi-daemon to publish a DAAP mDNS service via multicast. To use multicast, I added a veth to the VE:

```
[host-node]# vzctl set 1028 --netif_add eth0 --save
[host-node]# ifconfig veth1028.0 0
[host-node]# echo 1 > /proc/sys/net/ipv4/conf/veth1028.0/forwarding
[host-node]# echo 1 > /proc/sys/net/ipv4/conf/veth1028.0/proxy_arp
[host-node]# ip route add 192.168.10.1028 dev veth1028.0
[host-node]# vzctl enter 1028
[ve-1028]# /sbin/ifconfig eth0 0
[ve-1028]# /sbin/ip addr add 192.168.10.28 dev eth0
[ve-1028]# /sbin/ip route add default dev eth0
```

I can ping the VE from my notebook, so it seems the interface works. When I start the mt-daapd service, on the VE I can see the service published on both IPv4 and IPv6:

```
[ve-1028]# avahi-browse -r _daap._tcp
+ eth0 IPv6 iTunes Audio Access local
+ eth0 IPv4 iTunes Audio Access local
= eth0 IPv6 iTunes Audio Access local
  hostname = [ve-1028.local]
  address = [fe80::218:51ff:xxxx:xxxx]
  port = [3689]
= eth0 IPv4 iTunes Audio Access local
  hostname = [ve-1028.local]
  address = [192.168.10.28]
  port = [3689]
```

But on the HN (and my notebook) I see only the IPv6 service:

```
[host-node]# avahi-browse -r _daap._tcp
+ veth1028.0 IPv6 iTunes Audio Access local
= veth1028.0 IPv6 iTunes Audio Access local
  hostname = [ve-1028.local]
  address = [fe80::218:51ff:xxxx:xxxx]
  port = [3689]
```

EDIT: firewall and routing informations:

```
[host-node]# iptables -L
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
[host-node]# ip route
192.168.10.28 dev veth1028.0 scope link
192.168.10.0/24 dev eth0 proto kernel scope link src 192.168.10.10
default via 192.168.10.254 dev eth0
[host-node]# ip addr
2: 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
4: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:13:46:xx:xx:xx brd ff:ff:ff:ff:ff:ff
    inet 192.168.10.10/24 brd 192.168.10.255 scope global eth0
    inet6 fe80::213:46ff:xxxx:xxxx/64 scope link
        valid_lft forever preferred_lft forever
8: sit0: <NOARP> mtu 1480 qdisc noop state DOWN
    link/sit 0.0.0.0 brd 0.0.0.0
1: venet0: <BROADCAST,POINTOPOINT,NOARP,UP,LOWER_UP> mtu 1500 qdisc noqueue
state UNKNOWN
    link/void
9: veth1028.0: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1500 qdisc
noqueue state UNKNOWN
    link/ether 00:18:51:xx:xx:xx brd ff:ff:ff:ff:ff:ff
    inet6 fe80::218:51ff:xxxx:xxxx/64 scope link
        valid_lft forever preferred_lft forever

[ve-1028]# iptables -L
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
[ve-1028]# ip route
192.168.10.0/24 dev eth0 proto kernel scope link src 192.168.10.28
default dev eth0 scope link
[ve-1028]# ip addr
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
3: venet0: <BROADCAST,POINTOPOINT,NOARP,UP,LOWER_UP> mtu 1500 qdisc noqueue
state UNKNOWN
    link/void
    inet 127.0.0.1/32 scope host venet0
5: eth0: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1500 qdisc noqueue state
UNKNOWN
    link/ether 00:18:51:xx:xx:xx brd ff:ff:ff:ff:ff:ff
    inet 192.168.10.28/24 brd 192.168.10.255 scope global eth0
    inet6 fe80::218:51ff:xxxx:xxxx/64 scope link
        valid_lft forever preferred_lft forever
```

I can't explain that... Why IPv4 multicast are ignored?

Subject: Re: avahi-daemon and service discovery
Posted by [maratrus](#) on Thu, 28 Aug 2008 15:39:37 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

eth0 inside VE has both ipv4 and ipv6 ip addresses, but veth has only ipv6.

veth doesn't need ip address because it doesn't receive packets. it serves for transferring packets inside VE.

Subject: Re: avahi-daemon and service discovery
Posted by [heruan](#) on Thu, 28 Aug 2008 16:15:12 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thank for the reply! I assigned a random ipv4 address to veth on HN and now avahi works. Is this the correct way to solve the issue?

Subject: Re: avahi-daemon and service discovery

Posted by [maratrus](#) on Thu, 28 Aug 2008 16:26:31 GMT

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

Hi,

I don't know what is mt-daapd, but you shouldn't assign any IP address to veth, it doesn't make any sense.

could you please clarify in two word why do you want to do it?

Subject: Re: avahi-daemon and service discovery

Posted by [heruan](#) on Thu, 28 Aug 2008 16:45:18 GMT

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

Forget mt-daapd and consider a simple Zeroconf service: I wish to inform the network that a certain VE has, for example, an SSH server. Inside to VE, I'd do:

```
# avahi-publish -s _ssh._tcp
```

This will send multicast advertises on the network to inform an available SSH server. If I do that without assigning an IPv4 address to veth, only the IPv6 address is received on the network. Assigning an IPv4 address, solves this issue. I used 127.0.10.28, hoping it will cause no networking problems.

If there's a more correct way to do this, I'll be willing to test it!

Subject: Re: avahi-daemon and service discovery

Posted by [maratrus](#) on Fri, 29 Aug 2008 07:30:42 GMT

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

Hi,

Quote:

If I do that without assigning an IPv4 address to veth, only the IPv6 address is received on the network

why do you think that? IP address on veth interface doesn't make any sense.

You have appropriate ip address on eth interface inside your VE and this should be sufficient. Packets from VE follow to eth0 interface inside VE.

Packets to VE follow to veth interface inside HN. (see "192.168.10.28 dev veth1028.0 scope link").
