Subject: Re: [patch 2/6] [Network namespace] Network device sharing by view Posted by Andrey Savochkin on Mon, 26 Jun 2006 09:47:11 GMT

View Forum Message <> Reply to Message

Hi Daniel,

It's good that you kicked off network namespace discussion.

Although I wish you'd Cc'ed someone at OpenVZ so I could notice it earlier:).

Indeed, the first point to agree in this discussion is device list. In your patch, you essentially introduce a data structure parallel to the main device list, creating a "view" of this list. I see a fundamental problem with this approach.

When a device presents an skb to the protocol layer, it needs to know to which namespace this skb belongs.

Otherwise you would never get rid of problems with bind: what to do if device eth1 is visible in namespace1, namespace2, and root namespace, and each namespace has a socket bound to 0.0.0.0:80?

We have to conclude that each device should be visible only in one namespace. In this case, instead of introducing net_ns_dev and net_ns_dev_list structures, we can simply have a separate dev_base list head in each namespace. Moreover, separate device list in each namespace will be in line with making namespace isolation complete. Complete isolation will allow each namespace to set up own tun/tap devices, have own routes, netfilter tables, and so on.

My follow-up messages will contain the first set of patches with network namespaces implemented in the same way as network isolation in OpenVZ. This patchset introduces namespaces for device list and IPv4 FIB/routing. Two technical issues are omitted to make the patch idea clearer: device moving between namespaces, and selective routing cache flush + garbage collection.

If this patchset is agreeable, the next patchset will finalize integration with nsproxy, add namespaces to socket lookup code and neighbour cache, and introduce a simple device to pass traffic between namespaces. Then we will turn to less obvious matters including netlink messages, network statistics, representation of network information in proc and sysfs, tuning of parameters through sysctl, IPv6 and other protocols, and per-namespace netfilters.

Best regards Andrey