
Subject: Re: [RFD] L2 Network namespace infrastructure
Posted by [Patrick McHardy](#) on Sat, 23 Jun 2007 10:40:44 GMT
[View Forum Message](#) <> [Reply to Message](#)

Eric W. Biederman wrote:

- > -- The basic design
- >
- > There will be a network namespace structure that holds the global
- > variables for a network namespace, making those global variables
- > per network namespace.
- >
- > One of those per network namespace global variables will be the
- > loopback device. Which means the network namespace a packet resides
- > in can be found simply by examining the network device or the socket
- > the packet is traversing.
- >
- > Either a pointer to this global structure will be passed into
- > the functions that need to reference per network namespace variables
- > or a structure that is already passed in (such as the network device)
- > will be modified to contain a pointer to the network namespace
- > structure.

I believe OpenVZ stores the current namespace somewhere global, which avoids passing the namespace around. Couldn't you do this as well?

- > Depending upon the data structure it will either be modified to hold
- > a per entry network namespace pointer or it there will be a separate
- > copy per network namespace. For large global data structures like
- > the ipv4 routing cache hash table adding an additional pointer to the
- > entries appears the more reasonable solution.

So the routing cache is shared between all namespaces?

- > --- Performance
- >
- > In initial measurements the only performance overhead we have been
- > able to measure is getting the packet to the network namespace.
- > Going through ethernet bridging or routing seems to trigger copies
- > of the packet that slow things down. When packets go directly to
- > the network namespace no performance penalty has yet been measured.

It would be interesting to find out whats triggering these copies.
Do you have NAT enabled?

Containers mailing list
Containers@lists.linux-foundation.org
<https://lists.linux-foundation.org/mailman/listinfo/containers>
