
Subject: Re: Network virtualization/isolation

Posted by [Daniel Lezcano](#) on Tue, 28 Nov 2006 20:26:52 GMT

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

Eric W. Biederman wrote:

- > I do not want to get into a big debate on the merits of various
- > techniques at this time. We seem to be in basic agreement
- > about what we are talking about.
- >
- > There is one thing I think we can all agree upon.
- > - Everything except isolation at the network device/L2 layer, does not
- > allow guests to have the full power of the linux networking stack.

Agree.

- >
- > - There has been a demonstrated use for the full power of the linux
- > networking stack in containers..

Agree.

- >
- > - There are a set of techniques which look as though they will give
- > us full speed when we do isolation of the network stack at the
- > network device/L2 layer.

Agree.

- > Is there any reason why we don't want to implement network namespaces
- > without the full power of the linux network stack?

Don't make me wrong, I never said layer 2 should not be used. I am only arguing a layer 3 should use the mechanism provided by the layer 2 and use a subset of it like the sockets virtualization/isolation.

Just IP isolation for lightweight containers, applications containers in order to have mobility.

- > If there is a case where we clearly don't want the full power of the
- > linux network stack in a guest but we still need a namespace we can
- > start looking at the merits of the alternatives.

Dmitry and I, we are looking for a I3 based on a subset of the I2 and according with Herbert needs.

If we can provide a I3 isolation based on the I2 which:

- does not collide with I2
- fit the needs of Herbert
- allows the migration
- use common code between I2 and I3

Should it not be sufficient to justify to have a I3 with the I2 isolation ?

>> What is this new paradigm you are talking about ?

>

- > The basic point is this. The less like stock linux the inside of a
- > container looks, and the more of a special case it is the more

- > confusing it is. The classic example is that for a system container
- > routing packets between containers over the loopback interface is
- > completely unexpected.

Right for system container, but not necessary for application containers.

- >
- >> There is not extra networking data structure instantiation in the
- >> Daniel's L3.
- > Nope just an extra field which serves the same purpose.
- >
- >>> - Bind/Connect/Accept filtering. There are so few places in
- >>> the code this is easy to maintain without sharing code with
- >>> everyone else.
- >> For isolation too ? Can we build network migration on top of that ?

- > As long as you can take your globally visible network address with you
- > when you migrate you can build network migration on top of it. So yes
- > bind/accept filtering is sufficient to implement migration, if you are
- > only using IP based protocols.

When you migrate an application, you must cleanup related sockets on the source machine. The cleanup can not rely on the IP addresses because you will be not able to discriminate all the sockets related to the container. Another stuff is the network objects life-cycle, the container will die when the application will finish, the timewait sockets will stay until all data are flushed to peer. You can not restart a new container with the same IP address, so you need to monitor the socket before relaunching a new container or unmounting the aliased interface associated with the container. You need a ref counting for the container and this refcount is exactly what has the network namespace. Another example, you can not have several application binding to INADDR_ANY:port without conflict. The multiport instantiation is exactly what brings the sockets isolation/virtualization with the I2/I3.

Containers mailing list
Containers@lists.osdl.org
<https://lists.osdl.org/mailman/listinfo/containers>
