Subject: Re: [RFC] network namespaces
Posted by ebiederm on Wed, 06 Sep 2006 23:25:50 GMT
View Forum Message <> Reply to Message

"Caitlin Bestler" <caitlinb@broadcom.com> writes:

> ebiederm@xmission.com wrote:
>
>>
>>> Finally, as I understand both network isolation and network
>>> virtualization (both level2 and level3) can happily co-exist. We do
>>> have several filesystems in kernel. Let's have several network
>>> virtualization approaches, and let a user choose. Is that makes
>>> sense?
>>
>> If there are not compelling arguments for using both ways of
>> doing it is silly to merge both, as it is more maintenance overhead.
>>
>
> My reading is that full virtualization (Xen, etc.) calls for
> implementing
> L2 switching between the partitions and the physical NIC(s).
>
> The tradeoffs between L2 and L3 switching are indeed complex, but
> there are two implications of doing L2 switching between partitions:
>
> 1) Do we really want to ask device drivers to support L2 switching for
>   partitions and something *different* for containers?

No.

> 2) Do we really want any single packet to traverse an L2 switch (for
>   the partition-style virtualization layer) and then an L3 switch
>   (for the container-style layer)?

In general what has been done with layer 3 is to simply filter which
processes can use which IP addresses and it all happens at socket
creation time. So it is very cheap, and it can be done purely
in the network layer without any driver intervention.

Basically think of what is happening at layer 3 as an extremely light-weight
version of traffic filtering.

> The full virtualization solution calls for virtual NICs with distinct
> MAC addresses. Is there any reason why this same solution cannot work
> for containers (just creating more than one VNIC for the partition,
> and then assigning each VNIC to a container?)

The VNIC approach is the fundamental idea with the layer two networking and if we can push the work down into the device driver it so different destination macs show up a in different packet queues it should be as fast as a normal networking stack.

Implementing VNICs so far is the only piece of containers that has come close to device drivers, and we can likely do it without device driver support (but with more cost). Basically this optimization is a subset of the Grand Unified Lookup idea.

I think we can do a mergeable implementation without noticeable cost without when not using containers without having to resort to a grand unified lookup but I may be wrong.

Eric
