Subject: Re: [patch 2/6] [Network namespace] Network device sharing by view
Posted by ebiederm on Tue, 27 Jun 2006 11:52:52 GMT
View Forum Message <> Reply to Message

Daniel Lezcano <dlezcano@fr.ibm.com> writes:

>>>>My point is that if you make namespace tagging at routing time, and
>>>>your packets are being routed only once, you lose the ability
>>>>to have separate routing tables in each namespace.
>>>
>>>Right. What is the advantage of having separate the routing tables ?
>> Routing is everything.
>> For example, I want namespaces to have their private tunnel devices.
>> It means that namespaces should be allowed have private routes of local type,
>> private default routes, and so on...
>>
>
> Ok, we are talking about the same things. We do it only in a different way:
>
> * separate routing table :
>   namespace
> |
>   \--- route_tables
> |
>   \---routes
>
> * tagged routing table :
>   route_tables
> |
>   \---routes
> |
>   \---namespace

There is a third possibility, that falls in between these two if local
communication is really the bottle neck.

We have the dst cache for caching routes and cache multiple transformations
that happen on a packet.

With a little extra knowledge it is possible to have the separate
routing tables but have special logic that recognizes the local tunnel
device that connects namespaces and have it look into the next
namespaces routes, and build up a complete stack of dst entries of
where the packet needs to go.

I keep forgetting about that possibility. But as long as everything
is done at the routing layer that should work.

> I use the second method, because I think it is more effecient and reduce the
> overhead. But the isolation is minimalist and only aims to avoid the application
> using ressources outside of the container (aka namespace) without taking care of
> the system. For example, I didn't take care of network devices, because as far
> as see I can't imagine an administrator wanting to change the network device
> name while there are hundred of containers running. Concerning tunnel devices
> for example, they should be created inside the container.

Inside the containers I want all network devices named eth0!

> I think, private network ressources method is more elegant and involves more
> network ressources, but there is probably a significant overhead and some
> difficulties to have __lightweight__ container (aka application container), make
> nfs working well, etc... I did some tests with tbench and the loopback with the
> private namespace and there is roughly an overhead of 4 % without the isolation
> since with the tagging method there is 1 % with the isolation.

The overhead went down?

> The network namespace aims the isolation for now, but the container based on the
> namespaces will probably need checkpoint/restart and migration ability. The
> migration is needed not only for servers but for HPC jobs too.

Yes.

> So I don't know what level of isolation/virtualization is really needed by
> users, what should be acceptable (strong isolation and overhead / weak isolation
> and efficiency). I don't know if people wanting strong isolation will not prefer
> Xen (cleary with much more overhead than your patches ;))

We need a clean abstraction that optimizes well.

However local communication between containers is not what we should
benchmark. That can always be improved later. So long as the
performance is reasonable. What needs to be benchmarked is the
overhead of namespaces when connected to physical networking devices
and on their own local loopback, and comparing that to a kernel
without namespace support.

If we don't hurt that core case we have an implementation we can
merge. There are a lot of optimization opportunities for local
communications and we can do that after we have a correct and accepted
implementation. Anything else is optimizing too soon, and will
just be muddying the waters.

Eric