Subject: Re: [RFC] network namespaces
Posted by kir on Wed, 06 Sep 2006 18:56:43 GMT
View Forum Message <> Reply to Message

Eric W. Biederman wrote:
> Kir Kolyshkin <kir@openvz.org> writes:
>
>
>> Herbert Poetzl wrote:
>>
>>> my point (until we have an implementation which clearly
>>> shows that performance is equal/better to isolation)
>>> is simply this:
>>>
>>> of course, you can 'simulate' or 'construct' all the
>>> isolation scenarios with kernel bridging and routing
>>> and tricky injection/marking of packets, but, this
>>> usually comes with an overhead ...
>>>
>>>
>> Well, TANSTAAFL*, and pretty much everything comes with an overhead.
>> Multitasking comes with the (scheduler, context switch, CPU cache, etc.)
>> overhead -- is that the reason to abandon it? OpenVZ and Linux-VServer
>> resource management also adds some overhead -- do we want to throw it away?
>>
>> The question is not just "equal or better performance", the question is
>> "what do we get and how much we pay for it".
>>
>
> Equal or better performance is certainly required when we have the code
> compiled in but aren't using it. We must not penalize the current code.
>
That's a valid argument. Although it's not applicable here (at least for
both network virtualization types which OpenVZ offers). Kirill/Andrey,
please correct me if I'm wrong here.
>> 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?
>>
> o
> If there are not compelling arguments for using both ways of doing
> it is silly to merge both, as it is more maintenance overhead.
>
Definitely a valid argument as well.

I am not sure about "network isolation" (used by Linux-VServer), but as
it comes for level2 vs. level3 virtualization, I see a need for both.

Here is the easy-to-understand comparison which can shed some light:
http://wiki.openvz.org/Differences_between_venet_and_veth

Here are a couple of examples
* Do we want to let container's owner (i.e. root) to add/remove IP addresses? Most probably not, but in some cases we want that.
* Do we want to be able to run DHCP server and/or DHCP client inside a container? Sometimes...but not always.
* Do we want to let container's owner to create/manage his own set of iptables? In half of the cases we do.

The problem here is single solution will not cover all those scenarios.
> That said I think there is a real chance if we can look at the bind
> filtering and find a way to express that in the networking stack
> through iptables. Using the security hooks conflicts with things
> like selinux.   Although it would be interesting to see if selinux
> can already implement general purpose layer 3 filtering.
>
> The more I look the gut feel I have is that the way to proceed would
> be to add a new table that filters binds, and connects. Plus a new
> module that would look at a process creating a socket and tell us if
> it is the appropriate group of processes. With a little care that
> would be a general solution to the layer 3 filtering problem.
>
> Eric
>