Subject: Re: [RFC] network namespaces
Posted by ebiederm on Wed, 06 Sep 2006 17:58:39 GMT
View Forum Message <> Reply to Message

Herbert Poetzl <herbert@13thfloor.at> writes:

> On Wed, Sep 06, 2006 at 11:10:23AM +0200, Daniel Lezcano wrote:
>>
>> As far as I see, vserver use a layer 3 solution but, when needed, the
>> veth "component", made by Nestor Pena, is used to provide a layer 2
>> virtualization. Right ?
>
> well, no, we do not explicitely use the VETH daemon
> for networking, although some folks probably make use
> of it, mainly because if you realize that this kind
> of isolation is something different and partially
> complementary to network virtualization, you can do
> live without the layer 2 virtualization in almost
> all cases, nevertheless, for certain purposes layer
> 2/3 virtualization is required and/or makes perfect
> sense
>
>> Having the two solutions, you have certainly a lot if information
>> about use cases.
>
>> From the point of view of vserver, can you give some
>> examples of when a layer 3 solution is better/worst than
>> a layer 2 solution ?
>
> 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 ...
>
>> Who wants a layer 2/3 virtualization and why ?
>
> there are some reasons for virtualization instead of
> pure isolation (as Linux-VServer does it for now)
>
> - context migration/snapshot (probably reason #1)
> - creating network devices inside a guest
>   (can help with vpn and similar)
> - allowing non IP protocols (like DHCP, ICMP, etc)
>

> the problem which arises with this kind of network
> virtualization is that you need some additional policy
> for example to avoid sending 'evil' packets and/or
> (D)DoSing one guest from another, which again adds
> further overhead, so basically if you 'just' want
> to have network isolation, you have to do this:
>
> - create a 'copy' of your hosts networking inside
>   the guest (with virtual interfaces)
> - assign all the same (subset) ips and this to
>   the virtual guest interfaces
> - activate some smart bridging code which 'knows'
>   what ports can be used and/or mapped
> - add policy to block unwanted connections and/or
>   packets to/from the guest
>
> all this sounds very intrusive and for sure (please
> proove me wrong here :) adds a lot of overhead to the
> networking itself, while a 'simple' isolation approach
> for IP (tcp/udp) is (almost) without any cost, certainly
> without overhead once a connection is established.

Thanks, for the good summary of the situation.

I think we can prove you wrong but it is going to take
some doing to build a good implementation and take
the necessary measurements.

Hmm. I wonder if the filtering layer 3 style of isolation can be built with
netfilter rules. Just skimming it looks we may be able to do it with something
like the netfilter owner module, possibly in conjunction with the connmark or
conntrack modules. If not if the infrastructure is close enough we can write
our own module.

Has anyone looked at network isolation from the netfilter perspective?

Eric