Subject: Re: [RFD] L2 Network namespace infrastructure
Posted by Patrick McHardy on Sat, 23 Jun 2007 18:00:45 GMT
View Forum Message <> Reply to Message

Eric W. Biederman wrote:
> Patrick McHardy <kaber@trash.net> writes:
>
>>I believe OpenVZ stores the current namespace somewhere global,
>>which avoids passing the namespace around. Couldn't you do this
>>as well?
>
>
> It sucks. Especially in the corner cases.   Think macvlan
> with the real network device in one namespace and the ``vlan''
> device in another device.
>
> The implementation of a global is also pretty a little questionable.
> Last I looked it didn't work on the transmit path at all and
> interesting on the receive path.
>
> Further and fundamentally all a global achieves is removing the need
> for the noise patches where you pass the pointer into the various
> functions. For long term maintenance it doesn't help anything.
>
> All of the other changes such as messing with the
> initialization/cleanup and changing access to access the per network
> namespace data structure, and modifying the code partly along the way
> to reject working in other non-default network namespaces that are
> truly intrusive we both still have to make.
>
> So except as an implementation detail how we pass the per network
> namespace pointer is uninteresting.
>
> Currently I am trying for the least clever most straight forward
> implementation I can find, that doesn't give us a regression
> in network stack performance.
>
> So yes if we want to do passing through a magic per cpu global on
> the packet receive path now is the time to decide to do that.
> Currently I don't see the advantage in doing that so I'm not
> suggesting it.

I think your approach is fine and is probably a lot easier
to review than using something global.

>>>Depending upon the data structure it will either be modified to hold
>>>a per entry network namespace pointer or it there will be a separate

>>>copy per network namespace. For large global data structures like
>>>the ipv4 routing cache hash table adding an additional pointer to the
>>>entries appears the more reasonable solution.
>>
>>
>>So the routing cache is shared between all namespaces?
>
>
> Yes. Each namespaces has it's own view so semantically it's not
> shared. But the initial fan out of the hash table 2M or something
> isn't something we want to replicate on a per namespace basis even
> assuming the huge page allocations could happen.
>
> So we just tag the entries and add the network namespace as one more
> part of the key when doing hash table look ups.

I can wait for the patches, but I would be interested in how
GC is performed and whether limits can be configured per
namespace.