

---

Subject: Re: namespace support requires network modules to say "GPL"

Posted by [Ben Greear](#) on Tue, 04 Dec 2007 19:35:47 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Eric W. Biederman wrote:

> Ben Greear <greearb@candelatech.com> writes:

>

>> Eric W. Biederman wrote:

>>> However there also seem to be simpler cases like Ben's bridge module,

>>> that don't appear to have any global state.

>>>

>> Well, my module has some global state, but I don't think it needs to care about

>> namespaces. My first impression is that my module should be able to bridge

>> namespaces...not be contained within one. I can have user-space make sure that

>> I don't bridge between

>> devices in different name-spaces, or perhaps bridging between namespaces

>> wouldn't be a problem anyway.

>

> Bridging between namespaces should not be a problem, but it could be

> a bit of a challenge to setup (in finding the network devices).

> Probably the easy way is to setup the bridging and then move one of the

> network devices to the other network namespace.

>

> Essentially bridging between two network devices in two network

> namespaces looks like bridging between two network devices on two

> separate network stacks. Although internally things look a little

> better.

Ok, that sounds fine.

>> Currently I use procfs and ioctls bound to a procfs file descriptor.

>

> Which is where it gets tricky You are defining new userspace ABIs.

> I can see where they occasionally make sense during development

> and prototyping but long term out of tree userspace interfaces appear

> to me to be a real maintenance problem.

They are completely contained within my module, and no one is going to change my module w/out me knowing, so actually I have very little problem here :)

>> For namespaces in general, will there be a way to just do a dev\_get\_by\_\* and

>> find the

>> device in \*any\* namespace and query the device to see what namespace it is in?

>> Then my module or some other more clever piece of code can determine the

>> namespaces

>> (by comparing pointers if nothing else) and make proper decision. For instance,

>> maybe

>> we want to bridge two namespaces, or maybe we want to forbid that ever  
>> happening...  
>  
> The issue is that fundamentally all userspace device identifiers can  
> be duped between namespaces. So since there is no unique identifier  
> we can not implement a function to do that.

Ok, but can a netdev at least know what namespace it is in? I don't  
need this for my module, but it seems very useful knowledge...

Thanks,  
Ben

--

Ben Greear <greearb@candelatech.com>  
Candela Technologies Inc <http://www.candelatech.com>

---

Containers mailing list  
[Containers@lists.linux-foundation.org](mailto:Containers@lists.linux-foundation.org)  
<https://lists.linux-foundation.org/mailman/listinfo/containers>

---