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

Posted by [Daniel Lezcano](#) on Tue, 04 Dec 2007 15:19:38 GMT

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

Daniel Lezcano wrote:

> Ben Greear wrote:

>> Eric W. Biederman wrote:

>>> Patrick McHardy <kaber@trash.net> writes:

>>>

>>>

>>>> Ben Greear wrote:

>>>>

>>>>> I have a binary module that uses dev_get_by_name...it's sort of a

>>>>> bridge-like

>>>>> thing and

>>>>> needs user-space to tell it which device to listen for packets on...

>>>>>

>>>>> This code doesn't need or care about name-spaces, so I don't see

>>>>> how it could

>>>>> really

>>>>> be infringing on the author's code (any worse than loading a binary

>>>>> driver

>>>>> into the kernel

>>>>> ever does).

>>>>>

>>>

>>> Regardless of infringement it is incompatible with a complete network

>>> namespace implementation. Further it sounds like the module you are

>>> describing defines a kernel ABI without being merged and hopes that

>>> ABI will still be supportable in the future. Honestly I think doing so

>>> is horrible code maintenance policy.

>>>

>> I don't mind if the ABI changes, so long as I can still use something

>> similar.

>>

>> The namespace logic is interesting to me in general, but at this point

>> I can't think of a way that

>> it actually helps this particular module. All I really need is a way

>> to grab every frame

>> from eth0 and then transmit it to eth1. I'm currently doing this by

>> finding the netdevice

>> and registering a raw-packet protocol (ie, like tcpdump would do). At

>> least up to 2.6.23,

>> this does not require any hacks to the kernel and uses only non GPL

>> exported symbols.

>>

>> Based on my understanding of the namespace logic, if I never add any

>> namespaces,

>> the general network layout should look similar to how it does today,
>> so I should have
>> no logical problem with my module.
>>
>>> Once things are largely complete it makes sense to argue with out of
>>> tree module authors that because they don't have network namespace
>>> support in their modules, their modules are broken.
>> Does this imply that every module that accesses the network code
>> *must* become
>> GPL simply because it must interact with namespace logic that is
>> exported as GPL only symbols?
>
> That's right, with `init_net`'s `EXPORT_SYMBOL_GPL` and `dev_get_xx`, we
> enforce people to be GPL whatever they didn't asked to have the
> namespaces in their code.
>
> Eric, why can we simply change `EXPORT_SYMBOL_GPL` to `EXPORT_SYMBOL` for
> `init_net` ?

Another suggestion/question, is it acceptable to say non-gpl driver
should use `init_task.nsproxy->net_ns` instead of `&init_net` ?

Or does it make sense to have `init_net` gpl-exported, since we can access
it through `init_task` which is exported without gpl mention ?

Containers mailing list
Containers@lists.linux-foundation.org
<https://lists.linux-foundation.org/mailman/listinfo/containers>
