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

Posted by [ebiederm](#) on Sun, 02 Dec 2007 01:08:09 GMT

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

Mark Lord <lkml@rtr.ca> writes:

> Eric W. Biederman wrote:

>> Mark Lord <lkml@rtr.ca> writes:

>>

>>> Arjan van de Ven wrote:

>>>> On Sat, 01 Dec 2007 15:21:12 -0500

>>>> Mark Lord <lkml@rtr.ca> wrote:

>>>>

>>>>> Eric W. Biederman wrote:

>>>>>> Stephen Hemminger <shemminger@linux-foundation.org> writes:

>>>>>> Sure. We keep the updated dev_get_by_XXXX that takes a network

>>>>>> namespace parameter.

>>>>>> ..

>>>>>>

>>>>>> And what should code be passing in when "# CONFIG_NET_NS is not set" ?

>>>> network drivers probably really really don't want to call

>>>> dev_get_by_XXX...

>>>> ..

>>>>

>>>> Fine. But all of them want to call sk_alloc(),

>>>>

>>> network drivers should be calling sk_alloc less then they should

>>> call dev_get_by_XXXX. Only protocols call sk_alloc.

>>> ..

>>>

>> I think I saw some bridge code that calls it, too.

>> Regardless, it's EXPORT_SYMBOL(), but now unusable

>> due to the GPL-only symbol "init_net".

>>

>> Meanwhile, you are avoiding answering the question:

What EXPORT_SYMBOL_GPL means is I don't have a clue how you can use this without being a derivative work of the linux kernel. I think that is true for init_net and I stand by that. It is a very internal symbol.

Further the only case that I think might pass the derivative work test would be a network driver, and I am not affecting those.

I also believe that since it isn't mandatory you get your struct net pointer by taking the address of a GPL only symbol that those interfaces can continue to be used.

However I do suspect you are right that it may be more correct to realized that only a derivative work could possibly use those interfaces and tell 3rd parties that loud and clear.

We also have the policy of not changing exports from EXPORT_SYMBOL to EXPORT_SYMBOL_GPL. So making that such a change is difficult.

```
>>>> And what should code be passing in when "# CONFIG_NET_NS is not set" ?
> ..
>
>>> and many want to do register_netdev().
>>
>> I haven't even touched register_netdev.
> ..
>
> Bull-pucky. Somebody did:
>
>> @@ -3361,6 +3595,8 @@ int register_netdevice(struct net_device *dev)
>>
>>     /* When net_device's are persistent, this will be fatal. */
>>     BUG_ON(dev->reg_state != NETREG_UNINITIALIZED);
>> +     BUG_ON(!dev->nd_net);
>> +     net = dev->nd_net;
>>
>>     spin_lock_init(&dev->queue_lock);
>>     spin_lock_init(&dev->_xmit_lock);
> ..
>
> That new BUG_ON() line complains if nd_net has not been initialized,
> and the only thing I see drivers putting there is GPL-only "&init_net".
```

Yes and alloc_netdev initializes it. All network drivers are required to call alloc_netdev.

So register_netdev from a device driver perspective has not changed.

Anything the above change breaks happens to be a broken network device driver.

```
>>> So what should they be using there ?
>>
>> What are you having problems with?
> ..
>
> Avoiding the question again there, too.
```

No. I am saying that I don't see a problem. All of the real world problems that I actually know of are code bugs. I asked for

a real world problem to see if there was something I was missing in my analysis. That isn't avoiding the problem that is trying to see if I was wrong.

- > I personally am not having problems with anything.
- > But I noticed the API change, and would like somebody to fix it.
- > Any of the three ways (or a fourth, if you've got one) might do.

Honestly I think I have done the best I can with the knowledge and information that I have. I do not think dropping the GPL from the export of `init_net` makes sense. I think turning a NULL into `init_net` is a technical joke. That parameter is there because that is someplace where we need to pay attention to our network namespace.

Which only leaves your first option of making more symbols `EXPORT_SYMBOL_GPL` as even interesting.

That would seem to make some sense, given the attitudes and perceptions of the current network developers.

However I'm not volunteering for that one as finishing up the network and pid namespaces already has my plate full.

Eric

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