Subject: Re: namespace support requires network modules to say "GPL"
Posted by Patrick McHardy on Sun, 02 Dec 2007 21:59:46 GMT
View Forum Message <> Reply to Message

Adrian Bunk wrote:
> On Sun, Dec 02, 2007 at 09:03:56PM +0100, Patrick McHardy wrote:
>
>> For all I care binary modules can break, but frankly I don't see
>> how encapsulating a couple of structures and pointers in a new
>> structure and adding a new argument to existing functions shifts
>> the decision about how a function should be usable to the namespace
>> guys. IMO all functions should continue to be usable as before,
>> as decided by whoever actually wrote them.
>> ...
>
> Even ignoring the fact that it's unclear whether distributing modules
> with not GPLv2 compatible licences is legal at all or might bring you in
> jail,

Agreed, lets ignore that :)

> your statement has an interesting implication:
>
> Stuff like e.g. the EXPORT_SYMBOL(sk_alloc) predates the
> EXPORT_SYMBOL_GPL stuff.
>
> Who is considered the author of this code?
>
> And when should he state whether he prefers to use EXPORT_SYMBOL_GPL
> but wasn't able to use it at that when he wrote it since his code
> predates it and is glad to be able to decide this now?

He can state it when he feels like it, I don't see the point.
Authors generally get to decide whether they use EXPORT_SYMBOL
or EXPORT_SYMBOL_GPL unless in cases where its really clear-cut
that EXPORT_SYMBOL is inappropriate. But thats a different matter.

If a symbol was OK to be used previously and something using it
would not automatically be considered a derived work, how does
passing &init_net to the function just to make the compiler
happy, avoid BUG_ONs and generally keep things working as before
make it more of a derived work?

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