
Subject: Re: [PATCH 2/2] move unneeded data to initdata section
Posted by [ebiederm](#) on Thu, 15 Nov 2007 14:32:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

"Denis V. Lunev" <den@openvz.org> writes:

> This patch reverts Eric's commit 2b008b0a8e96b726c603c5e1a5a7a509b5f61e35
>
> It diets .text & .data section of the kernel if CONFIG_NET_NS is not set.
> This is safe after list operations cleanup.

Ok. This patch is technically safe because none of the touched code can live in a module and so we never touch the exit code path.

However in the general case and as a code idiom this `__net_initdata` on struct `pernet_operations` is fundamentally horribly broken.

Look at what happens if we use this idiom in module. There is only one definition of `__initdata ".init.data"`. The module loader places all sections that begin with `.init` in a region of memory that will be discarded after module initialization.

So in `register_pernet_operations` we pass in the a pointer to struct `pernet_operations` and call the `init` method. Later when we remove the module we again pass in the pointer to struct `pernet_operations` which lived in an `init` section so it has been discarded. We dereference that pointer to find the `exit` method and KABOOM!!!!

So I'm still opposed to `__net_initdata` on the grounds that at best it is like putting our head under a guillotine and reaching up and sawing at the row that holds the blade up with a pocket knife. It is a think rope and a puny knife so you are safe for a while....

Eric

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