Subject: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n
Posted by Benjamin Thery on Thu, 25 Oct 2007 12:59:03 GMT
View Forum Message <> Reply to Message

Hello Pavel,

I've found a problem with one of your patch related to netns:

* [NETNS] Move some code into __init section when CONFIG_NET_NS=n (v2)
  http://www.spinics.net/lists/netdev/msg43310.html

This patch introduces the __net_init/__net_exit/__net_initdata
defines to save some memory when CONFIG_NET_NS is not set.

Cedric Le Goater reported he had a *non-fatal* oops when booting
a 2.6.23-mm1-lxc1 kernel with CONFIG_NET_NS=n. (2.6.23-mm1-lxc1
contains the NETNS49 patchset). The oops occured when modules
related to iptables were loaded after the boot completes.

The problem is the following:

- Your patch adds the __net_initdata attribute to pernet_operations
  structures.

- pernet_operations are registered via register_pernet_subsys() and
  linked in the pernet_list during boot.

- At the end of boot, pernet_operations are freed (because of the
  __net_initdata attribute), and the pernet_list (or first_device list)
  points to freed memory.

- After boot, network modules which are netns-aware try to register
  themselves with register_pernet_subsys() and ...KABOOM... page
  fault when accessing pernet_list (or first_device list).
  (I reproduce Cedric's oops with the command: iptables --list)

This is not a problem right now in 2.6.23-mm1 or net-2.6 because
there are very few netns-aware network subsystems merged and they
are all initialized during boot. But it will be problematic when
we will merge netns code for subsystems which can be built as
modules (eg. iptables, ...). I'm not sure we can use
__net_init_data for pernet_operations then.
Maybe we can add some checks in register_pernet_operations
when CONFIG_NET_NS=n.

I haven't found a fix yet.

Regards,

Benjamin

--
Benjamin Thery - BULL/DT/Open Software R&D

http://www.bull.com

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

---

Subject: Re: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n
Posted by den on Thu, 25 Oct 2007 14:00:33 GMT

View Forum Message <> Reply to Message

The patch attached should help. The idea is simple. The "init" should be
called only once without NETNS. Period. No need for any lists.

I'll resend it to Dave after the ACK.

Regards,
 Den

Benjamin Thery wrote:
> Hello Pavel,
>
> I've found a problem with one of your patch related to netns:
>
> * [NETNS] Move some code into __init section when CONFIG_NET_NS=n (v2)
>   http://www.spinics.net/lists/netdev/msg43310.html
>
> This patch introduces the __net_init/__net_exit/__net_initdata
> defines to save some memory when CONFIG_NET_NS is not set.
>
> Cedric Le Goater reported he had a *non-fatal* oops when booting
> a 2.6.23-mm1-lxc1 kernel with CONFIG_NET_NS=n. (2.6.23-mm1-lxc1
> contains the NETNS49 patchset). The oops occured when modules
> related to iptables were loaded after the boot completes.
>
> The problem is the following:
>
> - Your patch adds the __net_initdata attribute to pernet_operations
>   structures.
>
> - pernet_operations are registered via register_pernet_subsys() and
>   linked in the pernet_list during boot.

>
> - At the end of boot, pernet_operations are freed (because of the
>   __net_initdata attribute), and the pernet_list (or first_device list)
>   points to freed memory.
>
> - After boot, network modules which are netns-aware try to register
>   themselves with register_pernet_subsys() and ...KABOOM... page
>   fault when accessing pernet_list (or first_device list).
>   (I reproduce Cedric's oops with the command: iptables --list)
>
> This is not a problem right now in 2.6.23-mm1 or net-2.6 because there are very few netns-aware network subsystems merged and they are all initialized during boot. But it will be problematic when we will merge netns code for subsystems which can be built as modules (eg. iptables, ...). I'm not sure we can use __net_init_data for pernet_operations then.
> Maybe we can add some checks in register_pernet_operations when CONFIG_NET_NS=n.
>
> I haven't found a fix yet.
>
> Regards,
> Benjamin
>

```
diff --git a/net/core/net_namespace.c b/net/core/net_namespace.c
index 6f71db8..9ba4809 100644
--- a/net/core/net_namespace.c
+++ b/net/core/net_namespace.c
@@ -186,7 +186,11 @@ static int register_pernet_operations(struct list_head *list,
  int error;

  error = 0;
+#ifdef CONFIG_NET_NS
  list_add_tail(&ops->list, list);
+#endif
+ INIT_LIST_HEAD(&ops->list);
+#endif
  for_each_net(net) {
   if (ops->init) {
    error = ops->init(net);
```

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

Subject: Re: Re: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n (resend, was wrong patch)
Posted by den on Thu, 25 Oct 2007 14:14:50 GMT
View Forum Message <> Reply to Message

Denis V. Lunev wrote:
> The patch attached should help. The idea is simple. The "init" should be
> called only once without NETNS. Period. No need for any lists.
>
> I'll resend it to Dave after the ACK.
>
> Regards,
> Den
>
> Benjamin Thery wrote:
>> Hello Pavel,
>>
>> I've found a problem with one of your patch related to netns:
>>
>> * [NETNS] Move some code into __init section when CONFIG_NET_NS=n (v2)
>>   http://www.spinics.net/lists/netdev/msg43310.html
>>
>> This patch introduces the __net_init/__net_exit/__net_initdata
>> defines to save some memory when CONFIG_NET_NS is not set.
>>
>> Cedric Le Goater reported he had a *non-fatal* oops when booting
>> a 2.6.23-mm1-lxc1 kernel with CONFIG_NET_NS=n. (2.6.23-mm1-lxc1
>> contains the NETNS49 patchset). The oops occured when modules
>> related to iptables were loaded after the boot completes.
>>
>> The problem is the following:
>>
>> - Your patch adds the __net_initdata attribute to pernet_operations
>>   structures.
>>
>> - pernet_operations are registered via register_pernet_subsys() and
>>   linked in the pernet_list during boot.
>>
>> - At the end of boot, pernet_operations are freed (because of the
>>   __net_initdata attribute), and the pernet_list (or first_device list)
>>   points to freed memory.
>>
>> - After boot, network modules which are netns-aware try to register
>>   themselves with register_pernet_subsys() and ...KABOOM... page
>>   fault when accessing pernet_list (or first_device list).
>>   (I reproduce Cedric's oops with the command: iptables --list)
>>
>> This is not a problem right now in 2.6.23-mm1 or net-2.6 because
>> there are very few netns-aware network subsystems merged and they

>> are all initialized during boot. But it will be problematic when
>> we will merge netns code for subsystems which can be built as
>> modules (eg. iptables, ...). I'm not sure we can use
>> __net_init_data for pernet_operations then.
>> Maybe we can add some checks in register_pernet_operations
>> when CONFIG_NET_NS=n.
>>
>> I haven't found a fix yet.
>>
>> Regards,
>> Benjamin

```
diff --git a/net/core/net_namespace.c b/net/core/net_namespace.c
index 6f71db8..cc306dc 100644
--- a/net/core/net_namespace.c
+++ b/net/core/net_namespace.c
@@ -186,7 +186,11 @@ static int register_pernet_operations(struct list_head *list,
  int error;

  error = 0;
+#ifdef CONFIG_NET_NS
  list_add_tail(&ops->list, list);
+#else
+ INIT_LIST_HEAD(&ops->list);
+#endif
  for_each_net(net) {
   if (ops->init) {
    error = ops->init(net);
```

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

Subject: Re: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n
Posted by Benjamin Thery on Thu, 25 Oct 2007 14:50:08 GMT

View Forum Message <> Reply to Message

Denis V. Lunev wrote:
> The patch attached should help. The idea is simple. The "init" should be
> called only once without NETNS. Period. No need for any lists.

This is the kind of idea I had but I didn't think it could be
that simple. :)
Thanks Denis.

> I'll resend it to Dave after the ACK.

Tested on x86_64 with CONFIG_NET_NS=n and y.
It fixes the issue we observed.

Acked-by: Benjamin Thery <benjamin.thery@bull.net>

> Regards,
> Den
>
> Benjamin Thery wrote:
>> Hello Pavel,
>>
>> I've found a problem with one of your patch related to netns:
>>
>> * [NETNS] Move some code into __init section when CONFIG_NET_NS=n (v2)
>>   http://www.spinics.net/lists/netdev/msg43310.html
>>
>> This patch introduces the __net_init/__net_exit/__net_initdata
>> defines to save some memory when CONFIG_NET_NS is not set.
>>
>> Cedric Le Goater reported he had a *non-fatal* oops when booting
>> a 2.6.23-mm1-lxc1 kernel with CONFIG_NET_NS=n. (2.6.23-mm1-lxc1
>> contains the NETNS49 patchset). The oops occured when modules
>> related to iptables were loaded after the boot completes.
>>
>> The problem is the following:
>>
>> - Your patch adds the __net_initdata attribute to pernet_operations
>>   structures.
>>
>> - pernet_operations are registered via register_pernet_subsys() and
>>   linked in the pernet_list during boot.
>>
>> - At the end of boot, pernet_operations are freed (because of the
>>   __net_initdata attribute), and the pernet_list (or first_device list)
>>   points to freed memory.
>>
>> - After boot, network modules which are netns-aware try to register
>>   themselves with register_pernet_subsys() and ...KABOOM... page
>>   fault when accessing pernet_list (or first_device list).
>>   (I reproduce Cedric's oops with the command: iptables --list)
>>
>> This is not a problem right now in 2.6.23-mm1 or net-2.6 because
>> there are very few netns-aware network subsystems merged and they
>> are all initialized during boot. But it will be problematic when

>> we will merge netns code for subsystems which can be built as
>> modules (eg. iptables, ...). I'm not sure we can use
>> __net_init_data for pernet_operations then.
>> Maybe we can add some checks in register_pernet_operations
>> when CONFIG_NET_NS=n.
>>
>> I haven't found a fix yet.
>>
>> Regards,
>> Benjamin
>>
>

--
Benjamin Thery - BULL/DT/Open Software R&D

http://www.bull.com

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

---

Subject: Re: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n
Posted by ebiederm on Thu, 25 Oct 2007 15:03:37 GMT

View Forum Message <> Reply to Message

"Denis V. Lunev" <den@sw.ru> writes:

> The patch attached should help. The idea is simple. The "init" should be
> called only once without NETNS. Period. No need for any lists.
>
> I'll resend it to Dave after the ACK.

First in the case of the code that is currently merged none of
the __net_init __net_exit or __net_initdata can be modular, so for
2.6.24 there is no fix needed. Yeah.

Second the whole concept of concept pernet_operations being __init
doesn't work when you have modular code that calls unregister_pernet_subsys().
Because unregister calls the exit method from the pernet_operations
structure. So the patch doesn't even begin to address the real
issue.

Third from my perspective CONFIG_NET_NS is a temporary measure
designed to last only until we have enough implementation experience

so that we can feel comfortable removing the experimental status of
the network namespace work. It was not my intention for it to be a
space saving measure. So I think it is silly to go marking up the
patches in development with __net_initdata etc.

At least so far I think __net_initdata is a totally bogus concept
and I'm not certain about the other two.

Eric

Subject: Re: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n
Posted by ebiederm on Thu, 25 Oct 2007 15:04:22 GMT

View Forum Message <> Reply to Message

Benjamin Thery <benjamin.thery@bull.net> writes:

> Denis V. Lunev wrote:
>> The patch attached should help. The idea is simple. The "init" should be
>> called only once without NETNS. Period. No need for any lists.
>
> This is the kind of idea I had but I didn't think it could be
> that simple. :)
> Thanks Denis.

It isn't.

>> I'll resend it to Dave after the ACK.
>
> Tested on x86_64 with CONFIG_NET_NS=n and y.
> It fixes the issue we observed.
>
> Acked-by: Benjamin Thery <benjamin.thery@bull.net>

Try rmmod.

Eric

Subject: Re: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n
Posted by Benjamin Thery on Thu, 25 Oct 2007 15:10:38 GMT
View Forum Message <> Reply to Message

Eric W. Biederman wrote:
> Benjamin Thery <benjamin.thery@bull.net> writes:
>
>> Denis V. Lunev wrote:
>>> The patch attached should help. The idea is simple. The "init" should be
>>> called only once without NETNS. Period. No need for any lists.
>> This is the kind of idea I had but I didn't think it could be
>> that simple. :)
>> Thanks Denis.
>
> It isn't.
>
>>> I'll resend it to Dave after the ACK.
>> Tested on x86_64 with CONFIG_NET_NS=n and y.
>> It fixes the issue we observed.
>>
>> Acked-by: Benjamin Thery <benjamin.thery@bull.net>
>
> Try rmmod.

rmmod was part of my tests and it does work.
I did:

```
$ iptables --list

  modules x_tables, ip_tables & iptable_filter are loaded
  each calling register_pernet_subsys.

$ rmmod iptable_filter ip_tables x_tables

  No problem here

$ iptables --list

  To be sure I can load the modules again.
```

>
> Eric
> -
> To unsubscribe from this list: send the line "unsubscribe netdev" in
> the body of a message to majordomo@vger.kernel.org
> More majordomo info at http://vger.kernel.org/majordomo-info.html
>

--
Benjamin Thery - BULL/DT/Open Software R&D

   http://www.bull.com

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

Subject: Re: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n
Posted by ebiederm on Thu, 25 Oct 2007 16:39:41 GMT
View Forum Message <> Reply to Message

Benjamin Thery <benjamin.thery@bull.net> writes:

```
> Eric W. Biederman wrote:
>> Benjamin Thery <benjamin.thery@bull.net> writes:
>>
>>> Denis V. Lunev wrote:
>>>> The patch attached should help. The idea is simple. The "init" should be
>>>> called only once without NETNS. Period. No need for any lists.
>>> This is the kind of idea I had but I didn't think it could be
>>> that simple. :)
>>> Thanks Denis.
>>
>> It isn't.
>>
>>>> I'll resend it to Dave after the ACK.
>>> Tested on x86_64 with CONFIG_NET_NS=n and y.
>>> It fixes the issue we observed.
>>>
>>> Acked-by: Benjamin Thery <benjamin.thery@bull.net>
>>
>> Try rmmod.
>
> rmmod was part of my tests and it does work.
> I did:
>
> $ iptables --list
>
>   modules x_tables, ip_tables & iptable_filter are loaded
>   each calling register_pernet_subsys.
>
> $ rmmod iptable_filter ip_tables x_tables
```

>
> No problem here
>
> $ iptables --list
>
> To be sure I can load the modules again.

You haven't changed those modules to be mark struct
pernet_operations as __net_initdata have you?

If that is the case the symptoms you are seeing make sense.

Not doing the list walks helps when if it is only compiled in
kernel data structures that are removed. However if it
is potentially modular data structures that are removed
the dereference of exit in unregister_pernet_subsys will also have
problems.

Eric

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

Subject: Re: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n

Posted by dlunev on Thu, 25 Oct 2007 16:51:28 GMT

View Forum Message <> Reply to Message

Eric W. Biederman wrote:
> Benjamin Thery <benjamin.thery@bull.net> writes:
>
>> Eric W. Biederman wrote:
>>> Benjamin Thery <benjamin.thery@bull.net> writes:
>>>
>>>> Denis V. Lunev wrote:
>>>>> The patch attached should help. The idea is simple. The "init" should be
>>>>> called only once without NETNS. Period. No need for any lists.
>>>> This is the kind of idea I had but I didn't think it could be
>>>> that simple. :)
>>>> Thanks Denis.
>>> It isn't.

this will work due to INIT_LIST_HEAD with circles list to itself and a
del operation will work.

By the way, I think that we can in the case of undefined CONFIG_NET_NS

reduce register to calling ->init method and unregister to calling
->exit method.

This is a correct thing at least for now and will be welcomed by the all
embedded/etc people.

Regards,
 Den

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

Subject: Re: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n

Posted by ebiederm on Thu, 25 Oct 2007 17:21:55 GMT
View Forum Message <> Reply to Message

"Denis V. Lunev" <dlunev@gmail.com> writes:

> Eric W. Biederman wrote:
>> Benjamin Thery <benjamin.thery@bull.net> writes:
>>
>>> Eric W. Biederman wrote:
>>>> Benjamin Thery <benjamin.thery@bull.net> writes:
>>>>
>>>>> Denis V. Lunev wrote:
>>>>>> The patch attached should help. The idea is simple. The "init" should be
>>>>>> called only once without NETNS. Period. No need for any lists.
>>>>> This is the kind of idea I had but I didn't think it could be
>>>>> that simple. :)
>>>>> Thanks Denis.
>>>> It isn't.
>
> this will work due to INIT_LIST_HEAD with circles list to itself and a
> del operation will work.

Suppose I have this fragment of code in a module:

```
> static int __net_init xt_net_init(struct net *net)
> {
>         ...
> }
>
> static void __net_exit xt_net_exit(struct net *net)
> {
>         ...
> }
>
> static struct pernet_operations __net_initdata  xt_net_ops = {
>  .init = xt_net_init,
>  .exit = xt_net_exit,
> };
>
> static int __init xt_init(void)
> {
>  return register_pernet_subsys(&xt_net_ops);
> }
>
> static void __exit xt_fini(void)
> {
>  unregister_pernet_subsys(&xt_net_ops);
> }
>
> module_init(xt_init);
> module_exit(xt_fini);
```

What happens during module removal when unregister_pernet_subys calls xt_net_ops.exit after xt_net_ops has been removed from the kernels memory?

> By the way, I think that we can in the case of undefined CONFIG_NET_NS
> reduce register to calling ->init method and unregister to calling
> ->exit method.
>
> This is a correct thing at least for now and will be welcomed by the all
> embedded/etc people.

I'm not fundamentally opposed. Earlier versions of my patchset did that and more. However I think the pain is greater then the gain right now. Especially since this concept seem to require having quality inspected into it.

Eric

Subject: Re: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n
Posted by davem on Fri, 26 Oct 2007 11:31:55 GMT

View Forum Message <> Reply to Message

From: ebiederm@xmission.com (Eric W. Biederman)
Date: Thu, 25 Oct 2007 11:21:55 -0600

> > By the way, I think that we can in the case of undefined CONFIG_NET_NS
> > reduce register to calling ->init method and unregister to calling
> > ->exit method.
> >
> > This is a correct thing at least for now and will be welcomed by the all
> > embedded/etc people.
>
> I'm not fundamentally opposed. Earlier versions of my patchset
> did that and more.   However I think the pain is greater then the
> gain right now. Especially since this concept seem to require
> having quality inspected into it.

I think the correct thing to do for now is to simply remove these
__net_* markers and their definitions. There are so many tricky cases
that it is easier to just get rid of them.

Could someone send me a patch which does that?
____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n
Posted by Benjamin Thery on Fri, 26 Oct 2007 11:41:55 GMT

View Forum Message <> Reply to Message

David Miller wrote:
> From: ebiederm@xmission.com (Eric W. Biederman)
> Date: Thu, 25 Oct 2007 11:21:55 -0600
>
>>> By the way, I think that we can in the case of undefined CONFIG_NET_NS
>>> reduce register to calling ->init method and unregister to calling
>>> ->exit method.
>>>
>>> This is a correct thing at least for now and will be welcomed by the all
>>> embedded/etc people.
>> I'm not fundamentally opposed. Earlier versions of my patchset
>> did that and more.   However I think the pain is greater then the
>> gain right now. Especially since this concept seem to require
>> having quality inspected into it.
>

> I think the correct thing to do for now is to simply remove these
> __net_* markers and their definitions. There are so many tricky cases
> that it is easier to just get rid of them.
>
> Could someone send me a patch which does that?

The attached patch revert Pavel's orginal patch from 2.6.23-mm1.
It should work fine with net-2.6 too.

Benjamin

--
Benjamin Thery - BULL/DT/Open Software R&D

   http://www.bull.com

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

Subject: Re: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n
Posted by davem on Fri, 26 Oct 2007 11:55:03 GMT

View Forum Message <> Reply to Message

From: Benjamin Thery <benjamin.thery@bull.net>
Date: Fri, 26 Oct 2007 13:41:55 +0200

> David Miller wrote:
> > From: ebiederm@xmission.com (Eric W. Biederman)
> > Date: Thu, 25 Oct 2007 11:21:55 -0600
> >
> >>> By the way, I think that we can in the case of undefined CONFIG_NET_NS
> >>> reduce register to calling ->init method and unregister to calling
> >>> ->exit method.
> >>>
> >>> This is a correct thing at least for now and will be welcomed by the all
> >>> embedded/etc people.
> >> I'm not fundamentally opposed. Earlier versions of my patchset
> >> did that and more.   However I think the pain is greater then the
> >> gain right now. Especially since this concept seem to require
> >> having quality inspected into it.
> >
> > I think the correct thing to do for now is to simply remove these
> > __net_* markers and their definitions. There are so many tricky cases
> > that it is easier to just get rid of them.

> >
> > Could someone send me a patch which does that?
>
> The attached patch revert Pavel's orginal patch from 2.6.23-mm1.
> It should work fine with net-2.6 too.

Thanks for doing this.

But this appears to be still discussed, so I'll give
Denis and others another day to work out the fix they
want to include.

Subject: Re: [NETNS] Oops in register_pernet_operations() with CONFIG_NET_NS=n
Posted by ebiederm on Fri, 26 Oct 2007 23:40:19 GMT

View Forum Message <> Reply to Message

David Miller <davem@davemloft.net> writes:

> Thanks for doing this.
>
> But this appears to be still discussed, so I'll give
> Denis and others another day to work out the fix they
> want to include.

At this point I think all that really needs to happen is to remove
__net_initdata. The function attributes seem sane. Not that
I have any problem with the complete revert either.

I will send a patch to that effect in just a moment.

Eric

Subject: [PATCH] net: Marking struct pernet_operations __net_initdata was inappropriate
Posted by ebiederm on Fri, 26 Oct 2007 23:45:33 GMT

It is not safe to to place struct pernet_operations in a special section.
We need struct pernet_operations to last until we call unregister_pernet_subsys.
Which doesn't happen until module unload.

So marking struct pernet_operations is a disaster for modules in two ways.
- We discard it before we call the exit method it points to.
- Because I keep struct pernet_operations on a linked list discarding
  it for compiled in code removes elements in the middle of a linked
  list and does horrible things for linked insert.

So this looks safe assuming __exit_refok is not discarded
for modules.

Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>

```
---
 drivers/net/loopback.c      |    2 +-
 fs/proc/proc_net.c          |    2 +-
 include/net/net_namespace.h |    2 --
 net/core/dev.c              |    6 +++---
 net/core/dev_mcast.c        |    2 +-
 net/netlink/af_netlink.c    |    2 +-
 6 files changed, 7 insertions(+), 9 deletions(-)

diff --git a/drivers/net/loopback.c b/drivers/net/loopback.c
index 662b8d1..45f30a2 100644
--- a/drivers/net/loopback.c
+++ b/drivers/net/loopback.c
@@ -284,7 +284,7 @@ static __net_exit void loopback_net_exit(struct net *net)
  unregister_netdev(dev);
 }
 
-static struct pernet_operations __net_initdata loopback_net_ops = {
+static struct pernet_operations loopback_net_ops = {
        .init = loopback_net_init,
        .exit = loopback_net_exit,
 };
diff --git a/fs/proc/proc_net.c b/fs/proc/proc_net.c
index 4edaad0..749def0 100644
--- a/fs/proc/proc_net.c
+++ b/fs/proc/proc_net.c
@@ -185,7 +185,7 @@ static __net_exit void proc_net_ns_exit(struct net *net)
  kfree(net->proc_net_root);
 }
 
-static struct pernet_operations __net_initdata proc_net_ns_ops = {
+static struct pernet_operations proc_net_ns_ops = {
  .init = proc_net_ns_init,
  .exit = proc_net_ns_exit,
 };
diff --git a/include/net/net_namespace.h b/include/net/net_namespace.h
index 93aa87d..5279466 100644
--- a/include/net/net_namespace.h
+++ b/include/net/net_namespace.h
@@ -102,11 +102,9 @@ static inline void release_net(struct net *net)
 #ifdef CONFIG_NET_NS
 #define __net_init
 #define __net_exit
-#define __net_initdata
 #else
 #define __net_init __init
 #define __net_exit __exit_refok
-#define __net_initdata __initdata
 #endif

 struct pernet_operations {
diff --git a/net/core/dev.c b/net/core/dev.c
index ddfef3b..853c8b5 100644
--- a/net/core/dev.c
+++ b/net/core/dev.c
@@ -2668,7 +2668,7 @@ static void __net_exit dev_proc_net_exit(struct net *net)
  proc_net_remove(net, "dev");
 }

-static struct pernet_operations __net_initdata dev_proc_ops = {
+static struct pernet_operations dev_proc_ops = {
  .init = dev_proc_net_init,
  .exit = dev_proc_net_exit,
 };
@@ -4328,7 +4328,7 @@ static void __net_exit netdev_exit(struct net *net)
  kfree(net->dev_index_head);
 }

-static struct pernet_operations __net_initdata netdev_net_ops = {
+static struct pernet_operations  netdev_net_ops = {
  .init = netdev_init,
  .exit = netdev_exit,
 };
@@ -4359,7 +4359,7 @@ static void __net_exit default_device_exit(struct net *net)
  rtnl_unlock();
 }

-static struct pernet_operations __net_initdata default_device_ops = {
+static struct pernet_operations  default_device_ops = {
  .exit = default_device_exit,
 };
```

```
diff --git a/net/core/dev_mcast.c b/net/core/dev_mcast.c
index 15241cf..ae35405 100644
--- a/net/core/dev_mcast.c
+++ b/net/core/dev_mcast.c
@@ -285,7 +285,7 @@ static void __net_exit dev_mc_net_exit(struct net *net)
  proc_net_remove(net, "dev_mcast");
 }

-static struct pernet_operations __net_initdata dev_mc_net_ops = {
+static struct pernet_operations dev_mc_net_ops = {
  .init = dev_mc_net_init,
  .exit = dev_mc_net_exit,
 };
diff --git a/net/netlink/af_netlink.c b/net/netlink/af_netlink.c
index 3252729..4f994c0 100644
--- a/net/netlink/af_netlink.c
+++ b/net/netlink/af_netlink.c
@@ -1888,7 +1888,7 @@ static void __net_exit netlink_net_exit(struct net *net)
 #endif
 }

-static struct pernet_operations __net_initdata netlink_net_ops = {
+static struct pernet_operations netlink_net_ops = {
  .init = netlink_net_init,
  .exit = netlink_net_exit,
 };
--
1.5.3.rc6.17.g1911
```

Subject: Re: [PATCH] net: Marking struct pernet_operations __net_initdata was inappropriate
Posted by davem on Sat, 27 Oct 2007 05:55:16 GMT

View Forum Message <> Reply to Message

From: ebiederm@xmission.com (Eric W. Biederman)
Date: Fri, 26 Oct 2007 17:45:33 -0600

>
> It is not safe to to place struct pernet_operations in a special section.
> We need struct pernet_operations to last until we call unregister_pernet_subsys.
> Which doesn't happen until module unload.

>
> So marking struct pernet_operations is a disaster for modules in two ways.
> - We discard it before we call the exit method it points to.
> - Because I keep struct pernet_operations on a linked list discarding
>   it for compiled in code removes elements in the middle of a linked
>   list and does horrible things for linked insert.
>
> So this looks safe assuming __exit_refok is not discarded
> for modules.
>
> Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>

Applied, thanks Eric.

Although juding by his comments I though that Denis had different
plans in mind to fix this.

Subject: Re: [PATCH] net: Marking struct pernet_operations __net_initdata was inappropriate
Posted by ebiederm on Sat, 27 Oct 2007 06:07:12 GMT

View Forum Message <> Reply to Message

David Miller <davem@davemloft.net> writes:
>
> Applied, thanks Eric.
>
> Although juding by his comments I though that Denis had different
> plans in mind to fix this.

He might. Somehow I wasn't on that thread so I missed it until after
I sent this patch. Reading through that thread again it looks like
he had a thought for another attribute. It all sounded very clever.

This patch is minimal stupid and should just work. Doubtless the
clever patch can be applied on top, once the details are figured
out.

Eric

Subject: Re: [PATCH] net: Marking struct pernet_operations __net_initdata was inappropriate
Posted by davem on Sat, 27 Oct 2007 07:29:43 GMT
View Forum Message <> Reply to Message

From: ebiederm@xmission.com (Eric W. Biederman)
Date: Sat, 27 Oct 2007 00:07:12 -0600

> This patch is minimal stupid and should just work. Doubtless the
> clever patch can be applied on top, once the details are figured
> out.

That is true and that's why I applied your patch.

Thanks!

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