
Subject: Re: [PATCH 03/16] net: Basic network namespace infrastructure.
Posted by [ebiederm](#) on Mon, 10 Sep 2007 06:40:15 GMT
[View Forum Message](#) <> [Reply to Message](#)

Krishna Kumar2 <krkumar2@in.ibm.com> writes:

```
> Eric W. Biederman wrote on 09/09/2007 02:45:34 AM:
>
> Hi Eric,
>
>> +static int register_pernet_operations(struct list_head *list,
>> +      struct pernet_operations *ops)
>> +{
>> <snip>
>> +out:
>> +  return error;
>> +
>> +out_undo:
>> +  /* If I have an error cleanup all namespaces I initialized */
>> +  list_del(&ops->list);
>> +  for_each_net(undo_net) {
>> +    if (undo_net == net)
>> +      goto undone;
>> +    if (ops->exit)
>> +      ops->exit(undo_net);
>> +  }
>> +undone:
>> +  goto out;
>> +}
>
> You could remove "undone" label (and associated) goto with a "break".
```

I could but there is there is no guarantee that the `for_each_net` macro is implemented with standard C looping construct such that `break` will work, and in one of the earlier versions that wasn't the case. So my paranoia says an explicit label safer and just as clear.

```
>> +static void unregister_pernet_operations(struct pernet_operations *ops)
>> +{
>> +  struct net *net;
>> +
>> +  list_del(&ops->list);
>> +  for_each_net(net)
>> +    if (ops->exit)
>> +      ops->exit(net);
>> +}
>
> Don't you require something like for_each_net_backwards to 'exit' in
```

> reverse order? Same comment for unregister_subnet_subsys(). Should this
> be done for failure in register_pernet_operations too?

There are no ordering guarantees between the initialization and cleanup of different network namespaces. The only real guarantee is that the initial network namespace is always present. Which means any order will work so always doing it in a forwards order in the list should be fine.

Eric

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