Subject: Re: [PATCH 2/5] net: Make rtnetlink infrastructure network namespace aware
Posted by ebiederm on Sat, 29 Sep 2007 21:00:48 GMT
View Forum Message <> Reply to Message

Patrick McHardy <kaber@trash.net> writes:

> Maybe I can save you some time: we used to do down_trylock()
> for the rtnl mutex, so senders would simply return if someone
> else was already processing the queue *or* the rtnl was locked
> for some other reason. In the first case the process already
> processing the queue would also process the new messages, but
> if it the rtnl was locked for some other reason (for example
> during module registration) the message would sit in the
> queue until the next rtnetlink sendmsg call, which is why
> rtnl_unlock does queue processing. Commit 6756ae4b changed
> the down_trylock to mutex_lock, so senders will now simply wait
> until the mutex is released and then call netlink_run_queue
> themselves. This means its not needed anymore.

Sounds reasonable.

I started looking through the code paths and I currently cannot
see anything that would leave a message on a kernel rtnl socket.

However I did a quick test adding a WARN_ON if there were any messages
found in the queue during rtnl_unlock and I found this code path
getting invoked from linkwatch_event. So there is clearly something I
don't understand, and it sounds at odds just a bit from your
description.

If we can remove the extra queue processing that would be great,
as it looks like a nice way to simplify the locking and the special
cases in the code.

Eric

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