Subject: Re: [PATCH] various dst_ifdown routines to catch refcounting bugs
Posted by davem on Thu, 27 Sep 2007 19:44:38 GMT
View Forum Message <> Reply to Message

From: ebiederm@xmission.com (Eric W. Biederman)
Date: Thu, 27 Sep 2007 10:27:43 -0600

> "Denis V. Lunev" <den@openvz.org> writes:
>
> > Moving dst entries into init_net.loopback_dev is not a good thing.
> > This hides obvious and non-obvious ref-counting bugs.
>
> Acked-by: "Eric W. Biederman" <ebiederm@xmission.com>

Patch applied.

> I do have a question I would like to bring up, because I like avoiding
> explicit references to loopback_dev when I can.
>
> /* Dirty hack. We did it in 2.2 (in __dst_free),
> * we have _very_ good reasons not to repeat
> * this mistake in 2.3, but we have no choice
> * now. _It_ _is_ _explicit_ _deliberate_
> * _race_ _condition_.
> *
> * Commented and originally written by Alexey.
> */
>
> What is the race that is talked about in that comment. Can we just
> assign NULL instead of the loopback device when we bring a route down.
> My gut feeling is that something like:
> dst->input = dst->output = dst_discard;
> may be enough. But I don't know where the deliberate race is.

The packet output path accesses the cached route device
asynchronously, and we are resetting the device to be loopback without
any synchronization whatsoever. None is in fact possible, and we
don't want to add it because that would be way too expensive.

So another thread on the system can either see the original device or
the loopback one.

It all works out because as the device goes down we'll purge any
packets queued into the transmit queue and packet scheduler for that
device.