Subject: [PATCH] Ensure that pneigh_lookup is protected with RTNL
Posted by Pavel Emelianov on Mon, 15 Oct 2007 13:38:57 GMT
View Forum Message <> Reply to Message

The pnigh_lookup is used to lookup proxy entries and to
create them in case lookup failed.

However, the "creation" code does not perform the re-lookup
after GFP_KERNEL allocation. This is done because the code
is expected to be protected with the RTNL lock, so add the
assertion (mainly to address future questions from new network
developers like me :)).

Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

---
diff --git a/net/core/neighbour.c b/net/core/neighbour.c
index c52df85..cd3af59 100644
--- a/net/core/neighbour.c
+++ b/net/core/neighbour.c
@@ -481,6 +481,8 @@ struct pneigh_entry * pneigh_lookup(struct neigh_table *tbl, const void
*pkey,
  if (!creat)
   goto out;

+ ASSERT_RTNL();
+
  n = kmalloc(sizeof(*n) + key_len, GFP_KERNEL);
  if (!n)
   goto out;

Subject: Re: [PATCH] Ensure that pneigh_lookup is protected with RTNL
Posted by davem on Mon, 15 Oct 2007 19:54:35 GMT
View Forum Message <> Reply to Message

From: Pavel Emelyanov <xemul@openvz.org>
Date: Mon, 15 Oct 2007 17:38:57 +0400

> The pnigh_lookup is used to lookup proxy entries and to
> create them in case lookup failed.
>
> However, the "creation" code does not perform the re-lookup
> after GFP_KERNEL allocation. This is done because the code
> is expected to be protected with the RTNL lock, so add the
> assertion (mainly to address future questions from new network
> developers like me :)).
>

> Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

Thanks for this patch, applied.