
Subject: [PATCH 1/5] rtnl_unlock cleanups
Posted by [den](#) on Fri, 05 Oct 2007 14:43:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

There is no need to process outstanding netlink user->kernel packets during rtnl_unlock now. There is no rtnl_trylock in the rtnetlink_rcv anymore.

Normal code path is the following:

```
netlink_sendmsg
  netlink_unicast
    netlink_sendskb
      skb_queue_tail
        netlink_data_ready
          rtnetlink_rcv
            mutex_lock(&rtnl_mutex);
            netlink_run_queue(sk, qlen, &rtnetlink_rcv_msg);
            mutex_unlock(&rtnl_mutex);
```

So, it is possible, that packets can be present in the rtnl->sk_receive_queue during rtnl_unlock, but there is no need to process them at that moment as rtnetlink_rcv for that packet is pending.

Signed-off-by: Denis V. Lunev <den@openvz.org>

Acked-by: Alexey Kuznetsov <kuznet@ms2.inr.ac.ru>

--- ./net/core/rtnetlink.c.rtnl2 2007-08-26 19:30:38.000000000 +0400

+++ ./net/core/rtnetlink.c 2007-10-01 13:09:03.000000000 +0400

@@ -75,8 +75,6 @@ void __rtnl_unlock(void)

void rtnl_unlock(void)

```
{
  mutex_unlock(&rtnl_mutex);
- if (rtnl && rtnl->sk_receive_queue.qlen)
-   rtnl->sk_data_ready(rtnl, 0);
  netdev_run_todo();
}
```

@@ -1319,11 +1317,9 @@ static void rtnetlink_rcv(struct sock *s
 unsigned int qlen = 0;

```
do {
-   mutex_lock(&rtnl_mutex);
+   rtnl_lock();
    qlen = netlink_run_queue(sk, qlen, &rtnetlink_rcv_msg);
-   mutex_unlock(&rtnl_mutex);
-
-   netdev_run_todo();
+   rtnl_unlock();
```

```
} while (qlen);  
}
```
