Subject: [PATCH 2/5] make netlink processing routines semi-synchronious (inspired by rtnl) v2
Posted by den on Fri, 05 Oct 2007 14:46:17 GMT
View Forum Message <> Reply to Message

The code in netfilter/nfnetlink.c and in ./net/netlink/genetlink.c looks
like outdated copy/paste from rtnetlink.c. Push them into sync with the
original.

Changes from v1:
- deleted comment in nfnetlink_rcv_msg by request of Patrick McHardy

Signed-off-by: Denis V. Lunev <den@openvz.org>
Acked-by: Patrick McHardy <kaber@trash.net>

```
--- ./net/netfilter/nfnetlink.c.nlk3 2007-10-01 09:47:53.000000000 +0400
+++ ./net/netfilter/nfnetlink.c 2007-10-01 17:13:09.000000000 +0400
@@ -44,26 +44,14 @@ static struct sock *nfnl = NULL;
 static const struct nfnetlink_subsystem *subsys_table[NFNL_SUBSYS_COUNT];
 static DEFINE_MUTEX(nfnl_mutex);

-static void nfnl_lock(void)
+static inline void nfnl_lock(void)
 {
  mutex_lock(&nfnl_mutex);
 }

-static int nfnl_trylock(void)
-{
- return !mutex_trylock(&nfnl_mutex);
-}
-
-static void __nfnl_unlock(void)
-{
- mutex_unlock(&nfnl_mutex);
-}
-
-static void nfnl_unlock(void)
+static inline void nfnl_unlock(void)
 {
  mutex_unlock(&nfnl_mutex);
- if (nfnl->sk_receive_queue.qlen)
-  nfnl->sk_data_ready(nfnl, 0);
 }

 int nfnetlink_subsys_register(const struct nfnetlink_subsystem *n)
@@ -147,9 +135,7 @@ static int nfnetlink_rcv_msg(struct sk_b
  ss = nfnetlink_get_subsys(type);
```

```
  if (!ss) {
 #ifdef CONFIG_KMOD
-  /* don't call nfnl_unlock, since it would reenter
-   * with further packet processing */
-  __nfnl_unlock();
+  nfnl_unlock();
   request_module("nfnetlink-subsys-%d", NFNL_SUBSYS_ID(type));
   nfnl_lock();
   ss = nfnetlink_get_subsys(type);
@@ -188,10 +174,9 @@ static void nfnetlink_rcv(struct sock *s
  unsigned int qlen = 0;

  do {
-  if (nfnl_trylock())
-   return;
+  nfnl_lock();
   qlen = netlink_run_queue(sk, qlen, nfnetlink_rcv_msg);
-  __nfnl_unlock();
+  nfnl_unlock();
  } while (qlen);
 }

--- ./net/netlink/genetlink.c.nlk3 2007-08-26 19:30:38.000000000 +0400
+++ ./net/netlink/genetlink.c 2007-10-01 16:05:29.000000000 +0400
@@ -22,22 +22,14 @@ struct sock *genl_sock = NULL;

 static DEFINE_MUTEX(genl_mutex); /* serialization of message processing */

-static void genl_lock(void)
+static inline void genl_lock(void)
 {
  mutex_lock(&genl_mutex);
 }

-static int genl_trylock(void)
-{
- return !mutex_trylock(&genl_mutex);
-}
-
-static void genl_unlock(void)
+static inline void genl_unlock(void)
 {
  mutex_unlock(&genl_mutex);
-
- if (genl_sock && genl_sock->sk_receive_queue.qlen)
-  genl_sock->sk_data_ready(genl_sock, 0);
 }
```

```
 #define GENL_FAM_TAB_SIZE 16
@@ -483,8 +475,7 @@ static void genl_rcv(struct sock *sk, in
  unsigned int qlen = 0;

  do {
-  if (genl_trylock())
-   return;
+  genl_lock();
   qlen = netlink_run_queue(sk, qlen, genl_rcv_msg);
   genl_unlock();
  } while (qlen && genl_sock && genl_sock->sk_receive_queue.qlen);
```