Subject: [PATCH 15/17] [NETNS]: Enable all routing manipulation via netlink inside namespace.
Posted by den on Wed, 06 Feb 2008 10:53:35 GMT
View Forum Message <> Reply to Message

Signed-off-by: Denis V. Lunev <den@openvz.org>
---
 net/ipv4/route.c | 16 ++++++++--------
 1 files changed, 8 insertions(+), 8 deletions(-)

diff --git a/net/ipv4/route.c b/net/ipv4/route.c
index 8a31e33..92ff622 100644
--- a/net/ipv4/route.c
+++ b/net/ipv4/route.c
@@ -2672,9 +2672,6 @@ static int inet_rtm_getroute(struct sk_buff *in_skb, struct nlmsghdr* nlh, void
  int err;
  struct sk_buff *skb;

- if (net != &init_net)
- return -EINVAL;
-
  err = nlmsg_parse(nlh, sizeof(*rtm), tb, RTA_MAX, rtm_ipv4_policy);
  if (err < 0)
   goto errout;
@@ -2704,7 +2701,7 @@ static int inet_rtm_getroute(struct sk_buff *in_skb, struct nlmsghdr* nlh, void
  if (iif) {
   struct net_device *dev;

- dev = __dev_get_by_index(&init_net, iif);
+ dev = __dev_get_by_index(net, iif);
   if (dev == NULL) {
    err = -ENODEV;
    goto errout_free;
@@ -2730,7 +2727,7 @@ static int inet_rtm_getroute(struct sk_buff *in_skb, struct nlmsghdr* nlh, void
    },
    .oif = tb[RTA_OIF] ? nla_get_u32(tb[RTA_OIF]) : 0,
   };
- err = ip_route_output_key(&init_net, &rt, &fl);
+ err = ip_route_output_key(net, &rt, &fl);
  }

  if (err)
@@ -2741,11 +2738,11 @@ static int inet_rtm_getroute(struct sk_buff *in_skb, struct nlmsghdr* nlh, void
   rt->rt_flags |= RTCF_NOTIFY;

```
  err = rt_fill_info(skb, NETLINK_CB(in_skb).pid, nlh->nlmsg_seq,
-    RTM_NEWROUTE, 0, 0);
+      RTM_NEWROUTE, 0, 0);
  if (err <= 0)
   goto errout_free;

- err = rtnl_unicast(skb, &init_net, NETLINK_CB(in_skb).pid);
+ err = rtnl_unicast(skb, net, NETLINK_CB(in_skb).pid);
 errout:
  return err;

@@ -2759,6 +2756,9 @@ int ip_rt_dump(struct sk_buff *skb,  struct netlink_callback *cb)
  struct rtable *rt;
  int h, s_h;
  int idx, s_idx;
+ struct net *net;
+
+ net = skb->sk->sk_net;

  s_h = cb->args[0];
  if (s_h < 0)
@@ -2768,7 +2768,7 @@ int ip_rt_dump(struct sk_buff *skb,  struct netlink_callback *cb)
   rcu_read_lock_bh();
   for (rt = rcu_dereference(rt_hash_table[h].chain), idx = 0; rt;
        rt = rcu_dereference(rt->u.dst.rt_next), idx++) {
-   if (idx < s_idx)
+   if (rt->u.dst.dev->nd_net != net || idx < s_idx)
     continue;
    if (rt->rt_genid != atomic_read(&rt_genid))
     continue;
--
1.5.3.rc5
```