
Subject: [PATCH 2/6 net-2.6.25] [NETNS] Add netns parameter to inetdev_by_index.

Posted by [den](#) on Mon, 21 Jan 2008 14:50:42 GMT

[View Forum Message](#) <> [Reply to Message](#)

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

```
---
include/linux/inetdevice.h | 2 +-
net/ipv4/devinet.c         | 6 +++---
net/ipv4/fib_semantics.c   | 2 +-
net/ipv4/igmp.c           | 4 +++-
net/ipv4/ip_gre.c         | 3 +-
5 files changed, 9 insertions(+), 8 deletions(-)

diff --git a/include/linux/inetdevice.h b/include/linux/inetdevice.h
index 45f3731..e74a2ee 100644
--- a/include/linux/inetdevice.h
+++ b/include/linux/inetdevice.h
@@ -133,7 +133,7 @@ extern struct net_device *ip_dev_find(__be32 addr);
extern int inet_addr_onlink(struct in_device *in_dev, __be32 a, __be32 b);
extern int devinet_ioctl(unsigned int cmd, void __user *);
extern void devinet_init(void);
-extern struct in_device *inetdev_by_index(int);
+extern struct in_device *inetdev_by_index(struct net *, int);
extern __be32 inet_select_addr(const struct net_device *dev, __be32 dst, int scope);
extern __be32 inet_confirm_addr(struct in_device *in_dev, __be32 dst, __be32 local, int scope);
extern struct in_ifaddr *inet_ifa_byprefix(struct in_device *in_dev, __be32 prefix, __be32 mask);
diff --git a/net/ipv4/devinet.c b/net/ipv4/devinet.c
index e381edb..21f71bf 100644
--- a/net/ipv4/devinet.c
+++ b/net/ipv4/devinet.c
@@ -409,12 +409,12 @@ static int inet_set_ifa(struct net_device *dev, struct in_ifaddr *ifa)
    return inet_insert_ifa(ifa);
}

-struct in_device *inetdev_by_index(int ifindex)
+struct in_device *inetdev_by_index(struct net *net, int ifindex)
{
    struct net_device *dev;
    struct in_device *in_dev = NULL;
    read_lock(&dev_base_lock);
- dev = __dev_get_by_index(&init_net, ifindex);
+ dev = __dev_get_by_index(net, ifindex);
    if (dev)
        in_dev = in_dev_get(dev);
    read_unlock(&dev_base_lock);
@@ -454,7 +454,7 @@ static int inet_rtm_deladdr(struct sk_buff *skb, struct nlmsg_hdr *nlh, void
*arg
```

```

goto errout;

ifm = nlmsg_data(nlh);
- in_dev = inetdev_by_index(ifm->ifa_index);
+ in_dev = inetdev_by_index(net, ifm->ifa_index);
  if (in_dev == NULL) {
    err = -ENODEV;
    goto errout;
diff --git a/net/ipv4/fib_semantics.c b/net/ipv4/fib_semantics.c
index ecd91c6..8b47e11 100644
--- a/net/ipv4/fib_semantics.c
+++ b/net/ipv4/fib_semantics.c
@@ -583,7 +583,7 @@ out:
  if (nh->nh_flags&(RTNH_F_PERVASIVE|RTNH_F_ONLINK))
    return -EINVAL;

- in_dev = inetdev_by_index(nh->nh_oif);
+ in_dev = inetdev_by_index(&init_net, nh->nh_oif);
  if (in_dev == NULL)
    return -ENODEV;
  if (!(in_dev->dev->flags&IFF_UP)) {
diff --git a/net/ipv4/igmp.c b/net/ipv4/igmp.c
index 285d262..b4df39a 100644
--- a/net/ipv4/igmp.c
+++ b/net/ipv4/igmp.c
@@ -1389,7 +1389,7 @@ static struct in_device * ip_mc_find_dev(struct ip_mreqn *imr)
  struct in_device *idev = NULL;

  if (imr->imr_ifindex) {
- idev = inetdev_by_index(imr->imr_ifindex);
+ idev = inetdev_by_index(&init_net, imr->imr_ifindex);
  if (idev)
    __in_dev_put(idev);
  return idev;
@@ -2222,7 +2222,7 @@ void ip_mc_drop_socket(struct sock *sk)
  struct in_device *in_dev;
  inet->mc_list = iml->next;

- in_dev = inetdev_by_index(iml->multi.imr_ifindex);
+ in_dev = inetdev_by_index(&init_net, iml->multi.imr_ifindex);
  (void) ip_mc_leave_src(sk, iml, in_dev);
  if (in_dev != NULL) {
    ip_mc_dec_group(in_dev, iml->multi.imr_multiaddr.s_addr);
diff --git a/net/ipv4/ip_gre.c b/net/ipv4/ip_gre.c
index 8b81deb..a74983d 100644
--- a/net/ipv4/ip_gre.c
+++ b/net/ipv4/ip_gre.c
@@ -1193,7 +1193,8 @@ static int ipgre_close(struct net_device *dev)

```

```
{
  struct ip_tunnel *t = netdev_priv(dev);
  if (ipv4_is_multicast(t->parms.iph.daddr) && t->mlink) {
- struct in_device *in_dev = inetdev_by_index(t->mlink);
+ struct in_device *in_dev;
+ in_dev = inetdev_by_index(dev->nd_net, t->mlink);
  if (in_dev) {
    ip_mc_dec_group(in_dev, t->parms.iph.daddr);
    in_dev_put(in_dev);
  }
}
```

--

1.5.3.rc5
