

Subject: [PATCH 6/11 net-2.6.26] [NETNS]: Process IP layer in the context of the correct namespace.

Posted by [den](#) on Mon, 24 Mar 2008 14:36:07 GMT

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

Replace all the rest of the init_net with a proper net on the IP layer.

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

```
net/ipv4/ip_fragment.c | 5 ++++-
net/ipv4/ip_input.c    | 6 ++++--
net/ipv4/ip_options.c  | 2 +-
net/ipv4/ip_output.c   | 2 +-
net/ipv4/ip_sockglue.c | 7 ++++---
5 files changed, 14 insertions(+), 8 deletions(-)
```

```
diff --git a/net/ipv4/ip_fragment.c b/net/ipv4/ip_fragment.c
index 3b2e5ad..8b448c4 100644
```

```
--- a/net/ipv4/ip_fragment.c
+++ b/net/ipv4/ip_fragment.c
@@ -204,8 +204,11 @@ static void ip_expire(unsigned long arg)
```

```
    if ((qp->q.last_in&FIRST_IN) && qp->q.fragments != NULL) {
        struct sk_buff *head = qp->q.fragments;
+   struct net *net;
+
+   net = container_of(qp->q.net, struct net, ipv4.frags);
        /* Send an ICMP "Fragment Reassembly Timeout" message. */
-   if ((head->dev = dev_get_by_index(&init_net, qp->iif)) != NULL) {
+   if ((head->dev = dev_get_by_index(net, qp->iif)) != NULL) {
        icmp_send(head, ICMP_TIME_EXCEEDED, ICMP_EXC_FRAGTIME, 0);
        dev_put(head->dev);
    }
```

```
diff --git a/net/ipv4/ip_input.c b/net/ipv4/ip_input.c
index f3a7a08..eb1fa27 100644
```

```
--- a/net/ipv4/ip_input.c
+++ b/net/ipv4/ip_input.c
@@ -160,6 +160,7 @@ int ip_call_ra_chain(struct sk_buff *skb)
    struct ip_ra_chain *ra;
    u8 protocol = ip_hdr(skb)->protocol;
    struct sock *last = NULL;
+   struct net_device *dev = skb->dev;

    read_lock(&ip_ra_lock);
    for (ra = ip_ra_chain; ra; ra = ra->next) {
@@ -170,7 +171,8 @@ int ip_call_ra_chain(struct sk_buff *skb)
    */
    if (sk && inet_sk(sk)->num == protocol &&
```

```

    (!sk->sk_bound_dev_if ||
-   sk->sk_bound_dev_if == skb->dev->ifindex)) {
+   sk->sk_bound_dev_if == dev->ifindex) &&
+   sk->sk_net == dev->nd_net) {
    if (ip_hdr(skb)->frag_off & htons(IP_MF | IP_OFFSET)) {
        if (ip_defrag(skb, IP_DEFRAG_CALL_RA_CHAIN)) {
            read_unlock(&ip_ra_lock);
@@ -286,7 +288,7 @@ static inline int ip_rcv_options(struct sk_buff *skb)
    opt = &(IPCB(skb)->opt);
    opt->optlen = iph->ihl*4 - sizeof(struct iphdr);

- if (ip_options_compile(&init_net, opt, skb)) {
+ if (ip_options_compile(dev->nd_net, opt, skb)) {
    IP_INC_STATS_BH(IPSTATS_MIB_INHDRERRORS);
    goto drop;
}
diff --git a/net/ipv4/ip_options.c b/net/ipv4/ip_options.c
index 59f7ddf..87cc122 100644
--- a/net/ipv4/ip_options.c
+++ b/net/ipv4/ip_options.c
@@ -145,7 +145,7 @@ int ip_options_echo(struct ip_options * dopt, struct sk_buff * skb)
    __be32 addr;

    memcpy(&addr, sptr+soffset-1, 4);
-   if (inet_addr_type(&init_net, addr) != RTN_LOCAL) {
+   if (inet_addr_type(skb->dst->dev->nd_net, addr) != RTN_LOCAL) {
        dopt->ts_needtime = 1;
        soffset += 8;
    }
diff --git a/net/ipv4/ip_output.c b/net/ipv4/ip_output.c
index dc494ea..349fae5 100644
--- a/net/ipv4/ip_output.c
+++ b/net/ipv4/ip_output.c
@@ -351,7 +351,7 @@ int ip_queue_xmit(struct sk_buff *skb, int ipfragok)
    * itself out.
    */
    security_sk_classify_flow(sk, &fl);
-   if (ip_route_output_flow(&init_net, &rt, &fl, sk, 0))
+   if (ip_route_output_flow(sk->sk_net, &rt, &fl, sk, 0))
        goto no_route;
}
sk_setup_caps(sk, &rt->u.dst);
diff --git a/net/ipv4/ip_sockglue.c b/net/ipv4/ip_sockglue.c
index 0857f2d..b854431 100644
--- a/net/ipv4/ip_sockglue.c
+++ b/net/ipv4/ip_sockglue.c
@@ -449,7 +449,8 @@ static int do_ip_setsockopt(struct sock *sk, int level,
    struct ip_options * opt = NULL;

```

```

    if (optlen > 40 || optlen < 0)
        goto e_inval;
-   err = ip_options_get_from_user(&init_net, &opt, optval, optlen);
+   err = ip_options_get_from_user(sk->sk_net, &opt,
+       optval, optlen);
    if (err)
        break;
    if (inet->is_icsk) {
@@ -589,13 +590,13 @@ static int do_ip_setsockopt(struct sock *sk, int level,
        err = 0;
        break;
    }
-   dev = ip_dev_find(&init_net, mreq.imr_address.s_addr);
+   dev = ip_dev_find(sk->sk_net, mreq.imr_address.s_addr);
    if (dev) {
        mreq.imr_ifindex = dev->ifindex;
        dev_put(dev);
    }
} else
-   dev = __dev_get_by_index(&init_net, mreq.imr_ifindex);
+   dev = __dev_get_by_index(sk->sk_net, mreq.imr_ifindex);

```

```
err = -EADDRNOTAVAIL;
```

```
--
```

1.5.3.rc5
