

---

Subject: [PATCH net-2.6.26 3/6][NETNS][SOCK]: Swap unhash and net change for icmp and tcp service sockets.

Posted by [Pavel Emelianov](#) on Thu, 27 Mar 2008 08:16:32 GMT

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

---

This is tricky.

The problem is that these sockets are created in `init_net` not to pin the new one and then are moved to it with `sk_change_net`. `_After_` this the socket is unhashed.

The problem is that `unhash` will decrement the `inuse` counter for the socket in a new net, but no in the one it was incremented (`init_net`). This will lead to RAW `inuse` counter increment each time a new net is created and will be negative for the new net.

Swap unhashing with net changing so that RAW `inuse` is decremented in the proper namespace.

Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

```
---
net/ipv4/icmp.c | 2 +-
net/ipv6/icmp.c | 2 +-
net/ipv6/mcast.c | 2 +-
net/ipv6/ndisc.c | 2 +-
4 files changed, 4 insertions(+), 4 deletions(-)
```

```
diff --git a/net/ipv4/icmp.c b/net/ipv4/icmp.c
index 3697e05..74eb856 100644
--- a/net/ipv4/icmp.c
+++ b/net/ipv4/icmp.c
@@ -1160,7 +1160,6 @@ int __net_init icmp_sk_init(struct net *net)
     goto fail;

     net->ipv4.icmp_sk[i] = sk = sock->sk;
-   sk_change_net(sk, net);

     sk->sk_allocation = GFP_ATOMIC;

@@ -1179,6 +1178,7 @@ int __net_init icmp_sk_init(struct net *net)
     * packets.
     */
     sk->sk_prot->unhash(sk);
+   sk_change_net(sk, net);
 }

 /* Control parameters for ECHO replies. */
```

```

diff --git a/net/ipv6/icmp.c b/net/ipv6/icmp.c
index 63309d1..fec5d55 100644
--- a/net/ipv6/icmp.c
+++ b/net/ipv6/icmp.c
@@ -820,7 +820,6 @@ static int __net_init icmpv6_sk_init(struct net *net)
 }

 net->ipv6.icmp_sk[i] = sk = sock->sk;
- sk_change_net(sk, net);

 sk->sk_allocation = GFP_ATOMIC;
 /*
@@ -839,6 +838,7 @@ static int __net_init icmpv6_sk_init(struct net *net)
 (2 * ((64 * 1024) + sizeof(struct sk_buff)));

 sk->sk_prot->unhash(sk);
+ sk_change_net(sk, net);
 }
 return 0;

```

```

diff --git a/net/ipv6/mcast.c b/net/ipv6/mcast.c
index d810cff..8fbfffc 100644
--- a/net/ipv6/mcast.c
+++ b/net/ipv6/mcast.c
@@ -2686,9 +2686,9 @@ static int igmp6_net_init(struct net *net)
 }

 net->ipv6.igmp_sk = sk = sock->sk;
- sk_change_net(sk, net);
 sk->sk_allocation = GFP_ATOMIC;
 sk->sk_prot->unhash(sk);
+ sk_change_net(sk, net);

```

```

 np = inet6_sk(sk);
 np->hop_limit = 1;
diff --git a/net/ipv6/ndisc.c b/net/ipv6/ndisc.c
index b4d8e33..a13ec57 100644
--- a/net/ipv6/ndisc.c
+++ b/net/ipv6/ndisc.c
@@ -1731,7 +1731,6 @@ static int ndisc_net_init(struct net *net)
 }

 net->ipv6.ndisc_sk = sk = sock->sk;
- sk_change_net(sk, net);

 np = inet6_sk(sk);
 sk->sk_allocation = GFP_ATOMIC;
@@ -1739,6 +1738,7 @@ static int ndisc_net_init(struct net *net)

```

```
/* Do not loopback ndisc messages */  
np->mc_loop = 0;  
sk->sk_prot->unhash(sk);  
+ sk_change_net(sk, net);
```

```
return 0;  
}
```

```
--
```

1.5.3.4

---