
Subject: [PATCH 2.6.25] [IPV6] Always pass a valid nl_info to inet6_rt_notify.
Posted by [den](#) on Thu, 13 Dec 2007 13:57:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

[IPV6] Always pass a valid nl_info to inet6_rt_notify.

This makes the code in the inet6_rt_notify more straightforward and provides ground for namespace passing.

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

```
net/ipv6/ip6_fib.c | 3 +-
net/ipv6/route.c | 27 ++++++++-----
2 files changed, 15 insertions(+), 15 deletions(-)
```

```
diff --git a/net/ipv6/ip6_fib.c b/net/ipv6/ip6_fib.c
```

```
index 5fae045..df05c6f 100644
```

```
--- a/net/ipv6/ip6_fib.c
```

```
+++ b/net/ipv6/ip6_fib.c
```

```
@@ -1315,6 +1315,7 @@ static int fib6_walk(struct fib6_walker_t *w)
```

```
static int fib6_clean_node(struct fib6_walker_t *w)
```

```
{
+ struct nl_info info = {};
  int res;
  struct rt6_info *rt;
  struct fib6_cleaner_t *c = container_of(w, struct fib6_cleaner_t, w);
@@ -1323,7 +1324,7 @@ static int fib6_clean_node(struct fib6_walker_t *w)
  res = c->func(rt, c->arg);
  if (res < 0) {
    w->leaf = rt;
- res = fib6_del(rt, NULL);
+ res = fib6_del(rt, &info);
    if (res) {
      #if RT6_DEBUG >= 2
        printk(KERN_DEBUG "fib6_clean_node: del failed: rt=%p@%p err=%d\n", rt, rt->rt6i_node,
res);
```

```
diff --git a/net/ipv6/route.c b/net/ipv6/route.c
```

```
index 1530934..02354a7 100644
```

```
--- a/net/ipv6/route.c
```

```
+++ b/net/ipv6/route.c
```

```
@@ -604,7 +604,8 @@ static int __ip6_ins_rt(struct rt6_info *rt, struct nl_info *info)
```

```
int ip6_ins_rt(struct rt6_info *rt)
```

```
{
- return __ip6_ins_rt(rt, NULL);
+ struct nl_info info = {};
+ return __ip6_ins_rt(rt, &info);
```

```

}

static struct rt6_info *rt6_alloc_cow(struct rt6_info *ort, struct in6_addr *daddr,
@@ -1261,7 +1262,8 @@ static int __ip6_del_rt(struct rt6_info *rt, struct nl_info *info)

int ip6_del_rt(struct rt6_info *rt)
{
- return __ip6_del_rt(rt, NULL);
+ struct nl_info info = {};
+ return __ip6_del_rt(rt, &info);
}

static int ip6_route_del(struct fib6_config *cfg)
@@ -2238,29 +2240,26 @@ errout:
void inet6_rt_notify(int event, struct rt6_info *rt, struct nl_info *info)
{
    struct sk_buff *skb;
- u32 pid = 0, seq = 0;
- struct nlmsg_hdr *nlh = NULL;
- int err = -ENOBUFS;
-
- if (info) {
-     pid = info->pid;
-     nlh = info->nlh;
-     if (nlh)
-         seq = nlh->nlmsg_seq;
- }
+ u32 seq;
+ int err;
+
+ err = -ENOBUFS;
+ seq = info->nlh != NULL ? info->nlh->nlmsg_seq : 0;

    skb = nlmsg_new(rt6_nlmsg_size(), gfp_any());
    if (skb == NULL)
        goto errout;

- err = rt6_fill_node(skb, rt, NULL, NULL, 0, event, pid, seq, 0, 0);
+ err = rt6_fill_node(skb, rt, NULL, NULL, 0,
+     event, info->pid, seq, 0, 0);
    if (err < 0) {
        /* -EMSGSIZE implies BUG in rt6_nlmsg_size() */
        WARN_ON(err == -EMSGSIZE);
        kfree_skb(skb);
        goto errout;
    }
- err = rtnl_notify(skb, &init_net, pid, RTNLGRP_IPV6_ROUTE, nlh, gfp_any());
+ err = rtnl_notify(skb, &init_net, info->pid,

```

```
+ RTNLGRP_IPV6_ROUTE, info->nlh, gfp_any());
errout:
if (err < 0)
    rtnl_set_sk_err(&init_net, RTNLGRP_IPV6_ROUTE, err);
--
```

1.5.3.rc5
