Subject: [PATCH net-2.6.25] [IPV4] fib_rules_unregister is essentially void.
Posted by den on Mon, 14 Jan 2008 13:48:21 GMT
View Forum Message <> Reply to Message

fib_rules_unregister is called only after successful register and the return
code is never checked.

Signed-off-by: Denis V. Lunev <den@openvz.org>
---
 include/net/fib_rules.h | 2 +-
 net/core/fib_rules.c | 21 ++++----------------
 2 files changed, 5 insertions(+), 18 deletions(-)

diff --git a/include/net/fib_rules.h b/include/net/fib_rules.h
index 88f870f..34349f9 100644
--- a/include/net/fib_rules.h
+++ b/include/net/fib_rules.h
@@ -104,7 +104,7 @@ static inline u32 frh_get_table(struct fib_rule_hdr *frh, struct nlattr **nla)
 }

 extern int fib_rules_register(struct net *, struct fib_rules_ops *);
-extern int fib_rules_unregister(struct net *, struct fib_rules_ops *);
+extern void fib_rules_unregister(struct net *, struct fib_rules_ops *);
 extern void   fib_rules_cleanup_ops(struct fib_rules_ops *);

 extern int   fib_rules_lookup(struct fib_rules_ops *,
diff --git a/net/core/fib_rules.c b/net/core/fib_rules.c
index 0eecf4c..42ccaf5 100644
--- a/net/core/fib_rules.c
+++ b/net/core/fib_rules.c
@@ -115,29 +115,16 @@ void fib_rules_cleanup_ops(struct fib_rules_ops *ops)
 }
 EXPORT_SYMBOL_GPL(fib_rules_cleanup_ops);

-int fib_rules_unregister(struct net *net, struct fib_rules_ops *ops)
+void fib_rules_unregister(struct net *net, struct fib_rules_ops *ops)
 {
- int err = 0;
- struct fib_rules_ops *o;

   spin_lock(&net->rules_mod_lock);
- list_for_each_entry(o, &net->rules_ops, list) {
- if (o == ops) {
-   list_del_rcu(&o->list);
-   fib_rules_cleanup_ops(ops);
-   goto out;
- }
- }

```
-
- err = -ENOENT;
-out:
+ list_del_rcu(&ops->list);
+ fib_rules_cleanup_ops(ops);
  spin_unlock(&net->rules_mod_lock);

  synchronize_rcu();
- if (!err)
-  release_net(net);
-
- return err;
+ release_net(net);
 }

 EXPORT_SYMBOL_GPL(fib_rules_unregister);
--
1.5.3.rc5
```

Subject: [PATCH net-2.6.25] [ARP] Remove overkill checks from neigh_param_alloc.

Posted by den on Mon, 14 Jan 2008 13:48:22 GMT

View Forum Message <> Reply to Message

Valid network device is always passed into neigh_param_alloc, so remove
extra checking for dev == NULL. Additionally, cleanup bogus netns assignment.

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

```
---
 net/core/neighbour.c |   18 +++++++-----------
 1 files changed, 7 insertions(+), 11 deletions(-)

diff --git a/net/core/neighbour.c b/net/core/neighbour.c
index af49137..32f1a23 100644
--- a/net/core/neighbour.c
+++ b/net/core/neighbour.c
@@ -1301,10 +1301,7 @@ struct neigh_parms *neigh_parms_alloc(struct net_device *dev,
  struct neigh_parms *p, *ref;
  struct net *net;

- net = &init_net;
- if (dev)
-  net = dev->nd_net;
-
+ net = dev->nd_net;
  ref = lookup_neigh_params(tbl, net, 0);
```

```
  if (!ref)
   return NULL;
@@ -1316,15 +1313,14 @@ struct neigh_parms *neigh_parms_alloc(struct net_device *dev,
   INIT_RCU_HEAD(&p->rcu_head);
   p->reachable_time =
     neigh_rand_reach_time(p->base_reachable_time);
-  if (dev) {
-   if (dev->neigh_setup && dev->neigh_setup(dev, p)) {
-    kfree(p);
-    return NULL;
-   }

-   dev_hold(dev);
-   p->dev = dev;
+  if (dev->neigh_setup && dev->neigh_setup(dev, p)) {
+   kfree(p);
+   return NULL;
   }
+
+  dev_hold(dev);
+  p->dev = dev;
   p->net = hold_net(net);
   p->sysctl_table = NULL;
   write_lock_bh(&tbl->lock);
--
1.5.3.rc5
```