
Subject: [PATCH net-2.6.25 3/6][NETNS]: Make the ctl-tables per-namespace

Posted by [Pavel Emelianov](#) on Thu, 10 Jan 2008 14:03:10 GMT

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

This includes passing the net to __addrconf_sysctl_register and saving this on the ctl_table->extra2 to be used in handlers (those, needing it).

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

net/ipv6/addrconf.c | 24 ++++++++-----
1 files changed, 14 insertions(+), 10 deletions(-)

diff --git a/net/ipv6/addrconf.c b/net/ipv6/addrconf.c

index 18d4334..bde50c6 100644

--- a/net/ipv6/addrconf.c

+++ b/net/ipv6/addrconf.c

@@ -456,13 +456,13 @@ static void dev_forward_change(struct inet6_dev *idev)
{

-static void addrconf_forward_change(void)

+static void addrconf_forward_change(struct net *net)

{
 struct net_device *dev;
 struct inet6_dev *idev;

read_lock(&dev_base_lock);

- for_each_netdev(&init_net, dev) {

+ for_each_netdev(net, dev) {

rcu_read_lock();

idev = __in6_dev_get(dev);

if (idev) {

@@ -478,12 +478,15 @@ static void addrconf_forward_change(void)

static void addrconf_fixup_forwarding(struct ctl_table *table, int *p, int old)

{

+ struct net *net;

+

+ net = (struct net *)table->extra2;

if (p == &ipv6_devconf_dflt.forwarding)

return;

if (p == &ipv6_devconf.forwarding) {

ipv6_devconf_dflt.forwarding = ipv6_devconf.forwarding;

- addrconf_forward_change();

+ addrconf_forward_change(net);

```

} else if ((!*p) ^ (!old))
    dev_forward_change((struct inet6_dev *)table->extra1);

@@ -4044,8 +4047,8 @@ static struct addrconf_sysctl_table
},
};

-static int __addrconf_sysctl_register(char *dev_name, int ctl_name,
- struct inet6_dev *idev, struct ipv6_devconf *p)
+static int __addrconf_sysctl_register(struct net *net, char *dev_name,
+ int ctl_name, struct inet6_dev *idev, struct ipv6_devconf *p)
{
    int i;
    struct addrconf_sysctl_table *t;
@@ -4068,6 +4071,7 @@ static int __addrconf_sysctl_register(char *dev_name, int ctl_name,
    for (i=0; t->addrconf_vars[i].data; i++) {
        t->addrconf_vars[i].data += (char*)p - (char*)&ipv6_devconf;
        t->addrconf_vars[i].extra1 = idev; /* embedded; no ref */
+    t->addrconf_vars[i].extra2 = net;
    }

    /*
@@ -4082,7 +4086,7 @@ static int __addrconf_sysctl_register(char *dev_name, int ctl_name,
    addrconf_ctl_path[ADDRCONF_CTL_PATH_DEV].procname = t->dev_name;
    addrconf_ctl_path[ADDRCONF_CTL_PATH_DEV].ctl_name = ctl_name;

- t->sysctl_header = register_sysctl_paths(addrconf_ctl_path,
+ t->sysctl_header = register_net_sysctl_table(net, addrconf_ctl_path,
    t->addrconf_vars);
    if (t->sysctl_header == NULL)
        goto free_procname;
@@ -4118,8 +4122,8 @@ static void addrconf_sysctl_register(struct inet6_dev *idev)
    NET_IPV6_NEIGH, "ipv6",
    &ndisc_ifinfo_sysctl_change,
    NULL);
- __addrconf_sysctl_register(idev->dev->name, idev->dev->ifindex,
- idev, &idev->cnf);
+ __addrconf_sysctl_register(idev->dev->nd_net, idev->dev->name,
+ idev->dev->ifindex, idev, &idev->cnf);
}

static void addrconf_sysctl_unregister(struct inet6_dev *idev)
@@ -4215,9 +4219,9 @@ int __init addrconf_init(void)
    ipv6_addr_label_rtnl_register();

#ifdef CONFIG_SYSCTL
- __addrconf_sysctl_register("all", NET_PROTO_CONF_ALL,
+ __addrconf_sysctl_register(&init_net, "all", NET_PROTO_CONF_ALL,

```

```
    NULL, &ipv6_devconf);  
- __addrconf_sysctl_register("default", NET_PROTO_CONF_DEFAULT,  
+ __addrconf_sysctl_register(&init_net, "default", NET_PROTO_CONF_DEFAULT,  
    NULL, &ipv6_devconf_dflt);  
#endif
```

```
--
```

1.5.3.4
