
Subject: Re: [patch 1/1][NETNS][IPV6] protect addrconf from loopback registration
Posted by [ebiederm](#) on Mon, 12 Nov 2007 16:51:58 GMT
[View Forum Message](#) <> [Reply to Message](#)

"Denis V. Lunev" <den@sw.ru> writes:

```
>> Index: linux-2.6-netns/net/ipv6/addrconf.c
>> =====
>> --- linux-2.6-netns.orig/net/ipv6/addrconf.c
>> +++ linux-2.6-netns/net/ipv6/addrconf.c
>> @@ -2272,7 +2272,8 @@ static int addrconf_notify(struct notifi
>>
>> switch(event) {
>> case NETDEV_REGISTER:
>> - if (!idev && dev->mtu >= IPV6_MIN_MTU) {
>> + if (!(dev->flags & IFF_LOOPBACK) &&
>> +     !idev && dev->mtu >= IPV6_MIN_MTU) {
```

It is idev being true here for the loopback device that would prevent things not missing the REGISTER event.

Hmm. But we do call `ipv6_add_dev` on loopback and now the loopback device is practically guaranteed to be the first device so we can probably just remove the special case in `addrconf_init`.

Anyway Daniels patch makes increasingly less sense the more I look at it.

> Namespaces are good to catch leakage using standard codepaths, so they
> should be preserved as much as possible. So, `_all_` normal down code
> should be called for a loopback device in other than `init_net` context.

In any context. After the code path is aware of multiple network namespaces `init_net` should not be special in any way.

I completely agree about the ability to catch weird leakage scenarios.

Eric

Containers mailing list
Containers@lists.linux-foundation.org
<https://lists.linux-foundation.org/mailman/listinfo/containers>
