
Subject: Re: [PATCH] Virtual ethernet device (v2.1)
Posted by [Pavel Emelianov](#) on Wed, 11 Jul 2007 13:00:35 GMT
[View Forum Message](#) <> [Reply to Message](#)

Patrick McHardy wrote:

> Pavel Emelianov wrote:

>> Patrick McHardy wrote:

>>

>>> Pavel Emelianov wrote:

>>>

```
>>>> +static const struct nla_policy veth_policy[VETH_INFO_MAX + 1] = {
>>>> + [VETH_INFO_MAC] = { .type = NLA_BINARY, .len = ETH_ALEN },
>>>> + [VETH_INFO_PEER] = { .type = NLA_STRING },
>>>> + [VETH_INFO_PEER_MAC] = { .type = NLA_BINARY, .len = ETH_ALEN },
>>>> +};
```

>>>

>>> Looks good, just one question. What happened to the IFLA_PARTNER
>>> attribute idea? I have a patch to allow specifying the initial
>>> MAC address for a device, IFLA_PARTNER would make the whole thing
>>> symmetrical.

>>

>> Well, the notion of a partner is not applicable to any generic link
>> unlike the device name, physical (MAC) address or MTU value. So i
>> think that it's better to keep this as a specific driver information
>> not to confuse the generic code. I think it's worth making this as
>> some more generic code than it is now, but since this driver is the
>> only user of "partner" maybe it's better not to make it right now.
>> Later, when we do know what do we want "partner" to be.

>

>

> Mhh doing it later means dealing with compatibility issues, which
> is why I'm asking now. We currently support IFLA_NAME, IFLA_MTU,

Oh, I see.

> IFLA_TXQLEN, IFLA_WEIGHT, IFLA_OPERSTATE and IFLA_LINKMODE, and
> with my patch additionally IFLA_ADDRESS and IFLA_BROADCAST.
> AFAICT they are all applicable for the partner link as well.

Agree. Maybe it is better to make some generic routine to create the
device with the parameters specified in the netlink packet. Then the
generic code creates one end of a tunnel and calls ->new_link callback.
This callback extracts the PARTNER packet part and calls this generic
routine to create the second pair.

Thanks,
Pavel
