

---

Subject: Re: [PATCH 2/12] L2 network namespace: network devices virtualization  
Posted by [ebiederm](#) on Fri, 08 Dec 2006 20:14:38 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Dmitry Mishin <dim@openvz.org> writes:

```
> Added ability to have per-namespace network devices.
>
> Signed-off-by: Dmitry Mishin <dim@openvz.org>
>
> ---
> include/linux/net_namespace.h | 6 +-
> include/linux/netdevice.h    | 10 +++++
> net/core/dev.c | 78 ++++++++++++++++++++++++++++++++++++++-----
> net/core/net-sysfs.c        | 23 ++++++++
> net/core/net_namespace.c    | 11 +++++
> 5 files changed, 114 insertions(+), 14 deletions(-)
>
> --- linux-2.6.19-rc6-mm2.orig/include/linux/net_namespace.h
> +++ linux-2.6.19-rc6-mm2/include/linux/net_namespace.h
> @@ -6,8 +6,10 @@
> #include <linux/errno.h>
>
> struct net_namespace {
> - struct kref kref;
> - struct nsproxy *ns;
> + struct kref kref;
> + struct nsproxy *ns;
> + struct net_device *dev_base_p, **dev_tail_p;
> + unsigned int hash;
> };
>
> extern struct net_namespace init_net_ns;
> --- linux-2.6.19-rc6-mm2.orig/include/linux/netdevice.h
> +++ linux-2.6.19-rc6-mm2/include/linux/netdevice.h
> @@ -379,6 +379,9 @@ struct net_device
> int promiscuity;
> int allmulti;
>
> +#ifdef CONFIG_NET_NS
> + struct net_namespace *net_ns;
> +#endif
>
> /* Protocol specific pointers */
>
> @@ -557,9 +560,16 @@ struct packet_type {
>
> #include <linux/interrupt.h>
```

```

> #include <linux/notifier.h>
> +#include <linux/net_namespace.h>
>
> extern struct net_device loopback_dev; /* The loopback */
> +#ifndef CONFIG_NET_NS
> extern struct net_device *dev_base; /* All devices */
> +#define dev_base_ns(dev) dev_base
> +#else
> +#define dev_base (current_net_ns->dev_base_p)
> +#define dev_base_ns(dev) (dev->net_ns->dev_base_p)
> +#endif
> extern rwlock_t dev_base_lock; /* Device list lock */
>
> extern int netdev_boot_setup_check(struct net_device *dev);
> --- linux-2.6.19-rc6-mm2.orig/net/core/dev.c
> +++ linux-2.6.19-rc6-mm2/net/core/dev.c
> @@ -90,6 +90,7 @@
> #include <linux/if_ether.h>
> #include <linux/netdevice.h>
> #include <linux/etherdevice.h>
> +#include <linux/net_namespace.h>
> #include <linux/notifier.h>
> #include <linux/skbuff.h>
> #include <net/sock.h>
> @@ -174,20 +175,28 @@ static spinlock_t net_dma_event_lock;
> * unregister_netdevice(), which must be called with the rtnl
> * semaphore held.
> */
> +#ifndef CONFIG_NET_NS
> struct net_device *dev_base;
> static struct net_device **dev_tail = &dev_base;
> -#DEFINE_RWLOCK(dev_base_lock);
> -
> +#define dev_tail_ns(dev) dev_tail
> EXPORT_SYMBOL(dev_base);
> +#else
> +#define dev_tail (current_net_ns->dev_tail_p)
> +#define dev_tail_ns(dev) (dev->net_ns->dev_tail_p)
> +#endif

```

No. Please do not hide a macro expansion (i.e. dev\_tail).  
That has the possibility of hiding costs, that shouldn't be hidden.

```

> --- linux-2.6.19-rc6-mm2.orig/net/core/net-sysfs.c
> +++ linux-2.6.19-rc6-mm2/net/core/net-sysfs.c
> @@ -453,6 +453,12 @@ static struct class net_class = {
>
> void netdev_unregister_sysfs(struct net_device * net)

```

```
> {  
> + #ifdef CONFIG_NET_NS  
> + if (net->net_ns != &init_net_ns)  
> + /* not supported yet: sysfs virtualization is required */  
> + return;  
> + #endif  
> +  
> device_del(&(net->dev));  
> }
```

This is something we need to seriously look at how to handle sysfs.

I believe separate directories per namespace and a magic symlink is going to be the sane approach.

Eric

---

Containers mailing list

Containers@lists.osdl.org

<https://lists.osdl.org/mailman/listinfo/containers>

---