
Subject: [PATCH 3/9] network namespaces: playing and debugging

Posted by [Andrey Savochkin](#) on Tue, 15 Aug 2006 14:48:43 GMT

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

Temporary code to play with network namespaces in the simplest way.

Do

```
exec 7< /proc/net/net_ns
```

in your bash shell and you'll get a brand new network namespace.

There you can, for example, do

```
ip link set lo up
```

```
ip addr list
```

```
ip addr add 1.2.3.4 dev lo
```

```
ping -n 1.2.3.4
```

Signed-off-by: Andrey Savochkin <saw@swsoft.com>

```
dev.c | 20 ++++++
```

```
1 files changed, 20 insertions(+)
```

--- ./net/core/dev.c.vensxdbg Tue Aug 15 13:46:44 2006

+++ ./net/core/dev.c Tue Aug 15 13:46:44 2006

```
@ @ -3597,6 +3597,8 @ @ int net_ns_start(void)
```

```
if (err)
```

```
goto out_register;
```

```
put_net_ns(orig_ns);
```

```
+ printk(KERN_DEBUG "NET_NS: created new netcontext %p for %s (pid=%d)\n",
```

```
+ ns, task->comm, task->tgid);
```

```
return 0;
```

```
out_register:
```

```
@ @ -3629,14 +3631,29 @ @ static void net_ns_destroy(void *data)
```

```
ip_fib_struct_cleanup();
```

```
pop_net_ns(orig_ns);
```

```
kfree(ns);
```

```
+ printk(KERN_DEBUG "NET_NS: netcontext %p freed\n", ns);
```

```
}
```

```
void net_ns_stop(struct net_namespace *ns)
```

```
{
```

```
+ printk(KERN_DEBUG "NET_NS: netcontext %p scheduled for stop\n", ns);
```

```
INIT_WORK(&ns->destroy_work, net_ns_destroy, ns);
```

```
schedule_work(&ns->destroy_work);
```

```
}
```

```
EXPORT_SYMBOL(net_ns_stop);
```

```
+
```

```
+static int net_ns_open(struct inode *i, struct file *f)
```

```
+{
```

```
+ return net_ns_start();
```

```

+}
+static struct file_operations net_ns_fops = {
+ .open = net_ns_open,
+};
+static int net_ns_init(void)
+{
+ return proc_net_fops_create("net_ns", S_IRWXU, &net_ns_fops)
+ ? 0 : -ENOMEM;
+}
#endif

/*
@@ -3701,6 +3718,9 @@ static int __init net_dev_init(void)
    hotcpu_notifier(dev_cpu_callback, 0);
    dst_init();
    dev_mcast_init();
+#ifdef CONFIG_NET_NS
+ net_ns_init();
+#endif
    rc = 0;
out:
    return rc;

```

Subject: Re: [PATCH 3/9] network namespaces: playing and debugging
 Posted by [Stephen Hemminger](#) on Wed, 16 Aug 2006 16:46:41 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Tue, 15 Aug 2006 18:48:43 +0400
 Andrey Savochkin <saw@sw.ru> wrote:

```

> Temporary code to play with network namespaces in the simplest way.
> Do
>     exec 7< /proc/net/net_ns
> in your bash shell and you'll get a brand new network namespace.
> There you can, for example, do
>     ip link set lo up
>     ip addr list
>     ip addr add 1.2.3.4 dev lo
>     ping -n 1.2.3.4
>
> Signed-off-by: Andrey Savochkin <saw@swsoft.com>

```

NACK, new /proc interfaces are not acceptable.

Subject: Re: [PATCH 3/9] network namespaces: playing and debugging

Posted by [ebiederm](#) on Wed, 16 Aug 2006 17:22:28 GMT

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

Stephen Hemminger <shemminger@osdl.org> writes:

```
> On Tue, 15 Aug 2006 18:48:43 +0400
> Andrey Savochkin <saw@sw.ru> wrote:
>
>> Temporary code to play with network namespaces in the simplest way.
>> Do
>>     exec 7< /proc/net/net_ns
>> in your bash shell and you'll get a brand new network namespace.
>> There you can, for example, do
>>     ip link set lo up
>>     ip addr list
>>     ip addr add 1.2.3.4 dev lo
>>     ping -n 1.2.3.4
>>
>> Signed-off-by: Andrey Savochkin <saw@swsoft.com>
>
> NACK, new /proc interfaces are not acceptable.
```

The rule is that new /proc interfaces that are not process related are not acceptable. If structured right a network namespace can arguably be process related.

I do agree that this interface is pretty ugly there.

Eric

Subject: Re: [PATCH 3/9] network namespaces: playing and debugging

Posted by [Andrey Savochkin](#) on Thu, 17 Aug 2006 06:28:14 GMT

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

On Wed, Aug 16, 2006 at 11:22:28AM -0600, Eric W. Biederman wrote:

```
> Stephen Hemminger <shemminger@osdl.org> writes:
>
>> On Tue, 15 Aug 2006 18:48:43 +0400
>> Andrey Savochkin <saw@sw.ru> wrote:
>>
>>> Temporary code to play with network namespaces in the simplest way.
>>> Do
>>>     exec 7< /proc/net/net_ns
>>> in your bash shell and you'll get a brand new network namespace.
>>> There you can, for example, do
>>>     ip link set lo up
>>>     ip addr list
```

```
> >> ip addr add 1.2.3.4 dev lo
> >> ping -n 1.2.3.4
> >>
> >> Signed-off-by: Andrey Savochkin <saw@swsoft.com>
> >
> > NACK, new /proc interfaces are not acceptable.
>
> The rule is that new /proc interfaces that are not process related
> are not acceptable. If structured right a network namespace can
> arguably be process related.
>
> I do agree that this interface is pretty ugly there.
```

This proc interface was a backdoor to play with namespaces without compiling any user-space programs.

As you wish.
Do you want to have a new clone flag right away?

Andrey

Subject: Re: [PATCH 3/9] network namespaces: playing and debugging
Posted by [dev](#) on Thu, 17 Aug 2006 08:29:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

```
>>Temporary code to play with network namespaces in the simplest way.
>>Do
>>  exec 7< /proc/net/net_ns
>>in your bash shell and you'll get a brand new network namespace.
>>There you can, for example, do
>>  ip link set lo up
>>  ip addr list
>>  ip addr add 1.2.3.4 dev lo
>>  ping -n 1.2.3.4
>>
>>Signed-off-by: Andrey Savochkin <saw@swsoft.com>
>
>
> NACK, new /proc interfaces are not acceptable.
```

As you can find from the comment this patch is just for playing with network namespace.

Kirill
