

---

Subject: Re: Pid namespaces problems

Posted by [ebiederm](#) on Tue, 06 Nov 2007 16:28:19 GMT

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

---

Pavel Emelyanov <[xemul@openvz.org](mailto:xemul@openvz.org)> writes:

> Yup. I agree with this too. I'm about to deprecate the pid and tgid  
> fields in the task\_struct (but not remove to make printk-s faster  
> and smaller).  
>  
> The work is in progress here.

Last time I looked at the printks, it looked to me like we wanted  
to print both the global pid and tsk->comm. When we printed which  
process did what, and helpers to do that would be nice.

printk of any of this information is rare enough that size is  
the important thing not speed.

I'm hoping we can come up with a nice wrapper to use with  
printk that we can bury all of the logic in. Possibly something  
like NIP\_QUAD.

>>  
>> - We need to store a struct pid reference on the sysvipc semaphores (and  
>> probably the other sysvipc objects) so that if they are used across  
>> namespace boundaries we can convert and give processes the pid for  
>> their local namespace.  
>  
> Hm.. What if they are used across two not-connected namespaces? E.g.  
> two different children of init namespace?

Well we get a 0 pid value. But at least we get a correct pid value when  
they are mixed in a way that it possible to have one.

There are also the credentials for AF\_UNIX sockets that need this same  
treatment.

>> - There are several architectures with their own signal functions for  
>> other OS compatibility that have are using \_pid and not \_vpid  
>> variants of functions. (irix and solaris)  
>> arch/mips/kernel/irixsig.c:irix\_waitsys  
>> arch/mips/kernel/sysirix.c:irix\_setpgrp  
>> arch/sparc64/solaris/misc.c:solaris\_procids  
>  
> Ok. Looks like your list is the same as mine. That's good to hear  
> that I haven't missed anything important.

I guess there are also all of the things we have been discussing with respect to signals, and clone semantics.

In particular after having looked at the issue it looks like we can send signals between namespaces, and it doesn't look like it is going to be hard to support setting `si_pid` properly.

So because we can do that I don't see any reason for the extra special case in clone for a new pid namespace.

I expect the only places we really differ is how to handle pids in the cross namespace cases.

> So, I see that you're about to take a closer look at the pid  
> namespaces. If so, then what time can we expect the net namespace  
> activity to go on? Or (if you don't mind) can we start merging  
> the patches to David as soon as he opens his 2.6.25 merge window?

My current plan is roughly:

- Take a trip and be offline for about a week (sorry folks my timing sucks)
- Poke on pid namespaces until I have a good feeling about where things are going (especially the cross namespace work).
- Get back to merging the network namespaces.  
sysctl, sysfs, af\_packet, af\_unix, and eventually ipv4.

I try and work where I can do the most good.

Eric

---

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

---