

---

Subject: Re: [PATCH 11/15] Signal semantics  
Posted by [serue](#) on Thu, 02 Aug 2007 20:09:39 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Quoting Kirill Korotaev (dev@sw.ru):

> Serge E. Hallyn wrote:  
> > Quoting Pavel Emelyanov (xemul@openvz.org):  
> >  
> >>[snip]  
> >>  
> >>  
> >>>>| Maybe it's worth disabling cross-namespaces ptracing...  
> >>>>  
> >>>>I think so too. Its probably not a serious limitation ?  
> >>>  
> >>>Several people think we will implement 'namespace entering' through a  
> >>>ptrace hack, where maybe the admin ptraces the init in a child pidns,  
> >>  
> >>Why not implement namespace entering w/o any hacks? :)  
> >  
> >  
> > I did, as a patch on top of the nsproxy container subsystem. The  
> > response was that that is a hack, and ptrace is cleaner :)  
> >  
> > So the current options for namespace entering would be:  
> >  
> > \* using Cedric's bind\_ns() functionality, which assigns an  
> > integer global id to a namespace, and allows a process to  
> > enter a namespace by that global id  
> >  
> looks more or less good and what OVZ actually does.  
> So I would prefer this one.

I think this was Cedric's last post of it:

<https://lists.linux-foundation.org/pipermail/containers/2007-June/005665.html>

However I'm pretty sure Eric would be soundly against this.

> > \* using my nsproxy container subsystem patch, which lets  
> > a process enter another namespace using  
> > echo pid > /container/some/cont/directory/tasks  
> > and eventually might allow construction of custom  
> > namespaces, i.e.  
> > mkdir /container/c1/c2  
> > ln -s /container/c1/c1/network /container/c1/c2/network  
> > echo \$\$ > /container/c1/c2/tasks  
>

> Sound ok and logical as well.

I last posted this here:

<http://www.mail-archive.com/devel@openvz.org/msg00295.html>

In the ensuing thread, the ptrace-based solution is also discussed.

> > \* using ptrace to coerce a process in the target namespace

> > into forking and executing the desired program.

>

> you'll need to change ptrace interface in this case imho...

> doesn't sound ok at all... at least for me. So I agree with Pavel.

Well maybe the problem is that while I think of it as ptrace-based, it's really only like ptrace in that it hijacks another process for a moment to clone it, but it likely will end up a completely different system call.

Eric, just curious, have you worked on this at all since february?

-serge

---

Containers mailing list

[Containers@lists.linux-foundation.org](mailto:Containers@lists.linux-foundation.org)

<https://lists.linux-foundation.org/mailman/listinfo/containers>

---