

---

Subject: Re: [PATCH 1/1] Revert "[PATCH] identifier to nsproxy"  
Posted by [serue](#) on Mon, 11 Dec 2006 20:24:37 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Quoting Eric W. Biederman (ebiederm@xmission.com):

> Cedric Le Goater <clg@fr.ibm.com> writes:

>

> > Eric W. Biederman wrote:

> >> This reverts commit 373beb35cd6b625e0ba4ad98baace12310a26aa8.

> >>

> >> No one is using this identifier yet. The purpose of this identifier

> >> is to export nsproxy to user space which is wrong. nsproxy is

> >> an internal implementation optimization, which should keep our

> >> fork times from getting slower as we increase the number of global

> >> namespaces you don't have to share.

> >>

> >> Adding a global identifier like this is inappropriate because it makes

> >> namespaces inherently non-recursive, greatly limiting what we can

> >> do with them in the future.

> >

> > Future will tell us, until then, let's see how useless and buggy this non

> > feature is.

> >

> > So for the moment, I would keep it and let people experiment.

>

> Even if the id is a sane idea nsproxy is very much the wrong place to

> put it. nsproxy is an optimization so we don't bloat task struct with

> several additional pointers, and it keeps fork times under control because

> in the normal case we only have a single increment instead of several.

> I'm not fully convinced it isn't a pessimization because it adds an

> extra indirection. It is fully inappropriate to export that to user

> space.

>

> Now I don't mind a little experimentation but not in the stable kernel

> when several people disagree.

>

> To a very large degree adding an id to struct nsproxy violates the compromise

> we came to when we agreed to add nsproxy.

>

> I am willing to discuss this but not while it is silently being added

Now now, it's not being silently added, it was a very clearly commented part of a proposed patchset sent to all interested parties for review, and now being argued over. Sounds kosher to me.

I think the problem is that some people want to see an answer to the namespace entering problem right now, but the alternate solution used on using pids as implicit identifiers can't be used until the pidspaces are

fully implemented.

- > to the user interface and being exported to userspace in a way we have
- > to support for the foreseeable future. To that I strongly object.
- >
- > The fact that it is simply dead code for 2.6.20 is probably sufficient
- > justification to revert it until we can agree.

Cedric, do you mind moving these patches to the end of the set, so that we can continue to develop patches against the -lxc tree without being depending on these patches? I'd hate to have this become a reason for people not to develop against -lxc.

-serge

---

Containers mailing list  
Containers@lists.osdl.org  
<https://lists.osdl.org/mailman/listinfo/containers>

---