
Subject: Re: [RFC][PATCH 5/7] VPIDs: vpid/pid conversion in VPID enabled case
Posted by [serue](#) on Mon, 06 Feb 2006 16:24:54 GMT

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

Quoting Alexey Kuznetsov (kuznet@ms2.inr.ac.ru):

> Hello!

>

> > into the pidhash, or make the first exec()d process the container's

> > init? Does that seem reasonable?

>

> It is exactly what openvz does, the first task becomes child reaper.

>

> Cedric asked why do we have an init in each container, I answered that

> we want to. But openvz approach is enough flexible not to do this, nothing

> will break if a container process is reparented to normal init.

>

> The question was not about openvz, it was about (container,pid) approach.

> How are you going to reap children without a child reaper inside container?

> If you reparent all the children to a single init in root container,

> what does wait() return? In openvz it returns global pid and child is buried

> in peace. If you do not have global pid, you cannot just return private pid.

Yes, /sbin/init would need to be modified to use whatever enhanced api is produced to support cross-container wait and pidlookup. I'm not sure that in this set of threads we've discussed enhanced wait requirements, though for pidlookup it's been mentioned that we can just require some kind of

```
sys_execute_container(pid_lookup(pid));
```

to avoid having to expand userspace pid lookup routines - ie to avoid having any userspace code (except init) need to know about containers.

But perhaps a per-container init is in fact cleaner. Other opinions?

-serge
