Subject: Re: [PATCH 0/13] Pid namespaces (OpenVZ view)
Posted by serue on Thu, 24 May 2007 15:09:09 GMT
View Forum Message <> Reply to Message

Quoting Pavel Emelianov (xemul@openvz.org):
> That's how OpenVZ sees the pid namespaces.
>
> The main idea is that kernel keeps operating with tasks pid
> as it did before, but each task obtains one more pid for each
> pid type - the virtual pid. When putting the pid to user or
> getting the pid from it kernel operates with the virtual ones.
>
> E.g. virtual pid is returned from getpid(), virtual pgid -
> from getpgid() and so on. Getting virtual pid from user is
> performed in setpgid(), setsid() and kill() mainly and in some
> other places.
>
> As far as the namespace are concerned I propose the following
> scheme. The namespace can be created from unshare syscall only.
> This makes fork() code look easier. Of course task must be

So is your main reason for posting this as a counter to Suka's patchset
the concern of overhead at clone?

thanks,
-serge

> prepared to have its pids changed. When task creates a new
> namespace it becomes its init and sees the tasks from it only.
> Tasks from init namespace see all the tasks.
>
> One relevant thing left behind is shrinking both proc's entries
> on task death. The reason I didn't do that is the following: this
> does not guarantee that the pid will be put (and thus still may
> hold the namespace), but makes the patch more complicated. So if
> this set will turns out to be interesting I will implement this
> thing as well.
>
> The patches are for 2.6.22-rc1-mm1 tree.
>
> Thanks,
> Pavel