Subject: Re:  Re: namespace and nsproxy syscalls
Posted by Herbert Poetzl on Sun, 08 Oct 2006 12:17:41 GMT
View Forum Message <> Reply to Message

On Sat, Oct 07, 2006 at 11:40:10PM +0200, Cedric Le Goater wrote:
> Herbert Poetzl wrote:
>
> >> * but we also said that a pid namespace can not survive the death
> >>   of its pid 1.
> >
> > which makes it unusable for our lightweight guest
> > purpose if it requires a separate init process
>
> the pid 1 process in a namespace can be the same for multiple
> namespaces, which makes it a SPOF one would say, but we need

SPOF as in Single Point of Failure?

I don't think that a 'fake' init process is a SPoF
because it actually does nothing in the setup, except
for being 'shown' in the procfs to make certain
(slightly misguided) apps happy

> a child reaper different from the "real" init process to avoid
> pid value collisions.

I agree (well IIRC I already stated that reaper and
init do not have to be identical and the reaper could
be a kernel thread as well), the question here just
is: do we still need a reaper _for_each_ guest?

> >> yes, i'm testing such a patch as discussed on the list. I have good
> >> results for a full nsproxy but i'm having trouble with the mnt
> >> namespace (used to be called namespace) which is stored in nsproxy
> >> and the fs_struct which is stored in the task_struct.
> >
> > what's the problem with handing out *space handles to userspace, which
> > can be later used to reach a specific namespace and/or manipulate
> > specific settings?
>
> no problem. that's fine.

okay, then we should consider using whatever seems
appropriate as a namespace handle and make the proxy
completely transparent/invisible to userspace
(as was discussed and suggested several times at the
beginning)

> I'm being cautious with the mnt namespace.

they are 'somewhat' special ATM, as they allow some
kind of 'inheritance', but I think pid spaces would
be a good candidate for similar behaviour ...

best,
Herbert

> cheers,
>
> C.

---