Subject: Re: pspace child_reaper
Posted by ebiederm on Tue, 29 Aug 2006 16:19:01 GMT
View Forum Message <> Reply to Message

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

> Eric W. Biederman wrote:
>> Cedric Le Goater <clg@fr.ibm.com> writes:
>>
>>> Hello All,
>>>
>>> Eric, in your initial proof of concept on the pid namespace, you were
>>> defining a child_reaper per pid namespace.
>>>
>>> IMO, we can't use init_task as a child_reaper in a pid namespace because we
>>> will have pid collision which might result in a breakage of the init_task.
>>
>> The kernel doesn't use init_task (The idle thread) once it starts
>> init. Reaping children is the job of pid == 1.
>
> agree.
>
>>> Here are some questions on the model you intended to follow :
>>>
>>> Do you think we should have a child_reaper task per container ?
>> We have an init per container so yes.
>
> hmm, have we always ? what if i don't start an /sbin/init process in my
> newly created pid namespace or container. where do I collect all the SIGCHLD ?

And this is the core question.

>>> Any completely different idea on the topic ?
>> Init reaps the children, and I believe there are parts of user space
>> that depend on this. We shouldn't change that semantic.
>
> IMHO, the only semantic i see is in the kernel, which needs someone to take
> care of sigchld. /sbin/init is a very good candidate bc it collects sigchld
> anyway and discards the ones it doesn't know about.

Roughly. The other is a complete process tree. Not having an init process will
break the process tree.

I think there could be a compelling case made for not having an init
process, but the semantic changes are subtle and hairy. I don't think
it is what we want to as a first pass.

I also believe that most of the advantages of not having an init

process can be had with a trivial (probably static) init program that only calls waitpid. Taking essentially no resources.

Eric

_____

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