Subject: Re: The issues for agreeing on a virtualization/namespaces implementation.
Posted by Hubertus Franke on Wed, 08 Feb 2006 15:57:22 GMT

View Forum Message <> Reply to Message

Kirill Korotaev wrote:
>>> Eric W. Biederman wrote:
>>> So it seems the clone(flags) is a reasonable approach to create new
>>> namespaces. Question is what is the initial state of each namespace?
>>> In pidspace we know we should be creating an empty pidmap !
>>> In network, someone suggested creating a loopback device
>>> In uts, create "localhost"
>>> Are there examples where we rather inherit ? Filesystem ?
>>
>> Of course filesystem is already implemented, and does inheret a full
>> copy.
>
>
> why do we want to use clone()? Just because of its name and flags?
> I think it is really strange to fork() to create network context. What
> has process creation has to do with it?
>
> After all these clone()'s are called, some management actions from host
> system are still required, to add these IPs/routings/etc.
> So? Why mess it up? Why not create a separate clean interface for
> container management?
>
> Kirill
>

We need a "init" per container, which represents the context of the
system represented by the container.
If that is the case, then why not create the container such that
we specify what namespaces need to be new for a container at
the container creation time and initialize them to a well understood
state that makes sense (e.g. copy namespace (FS, uts) , new fresh state (pid)).

Then use the standard syscall to modify state (now "virtualized" through
the task->xxx_namespace access).

Do you see a need to change the namespace of a container after it
has been created. I am not referring to the state of the namespace
but truely moving to a completely different namespace after the
container has been created.

Obviously you seem to have some other usage in mind, beyond what my
limited vision can see. Can you share some of those examples, because
that would help this discussion along ...

Thanks a 10^6.

-- Hubertus