
Subject: Re: [PATCH 1/4] Virtualization/containers: introduction
Posted by [ebiederm](#) on Tue, 07 Feb 2006 14:31:33 GMT
[View Forum Message](#) <> [Reply to Message](#)

Kirill Korotaev <dev@sw.ru> writes:

>>>> We are never going to form a consensus if all of the people doing
>>>> implementations don't talk.
>>>
>>> Speaking of which - it would be interesting to get Kirill's
>>> comments on Eric's patchset ;)
> I'll do comment.

Thank you I will look forward to your comments.

>>> Once we know what's good and bad about both patchsets, we'll
>>> be a lot closer to knowing what exactly should go upstream.
> I'm starting to think that nothing in upstream can be better for all of us :)

In a thread voicing the concerns for maintaining out of tree patches
that is a natural concern.

>> Let's compare approaches of patchsets before the patchsets themselves.
>> It seems to be, should we:
>> A) make a general form of virtualising PIDs, and hope this assists
>> later virtualisation efforts (Eric's patch)
>> B) make a general form of containers/jails/vservers/vpses, and layer
>> PID virtualisation on top of it somewhere (as in openvz, vserver)
> >
>> I can't think of any real use cases where you would specifically want A)
>> without B).
> Exactly! All these patches for A) look weird for me without containers itself. A
> try to make half-solution which is bad.

I am willing to contend that my approach also leads to a complete solution.
In fact I believe my network virtualization has actually gone much farther
than yours. Although I admit there is still some work to do before
the code is in shape to be merged.

You notice in the kernel there is also not a struct process?

To me having a container structure while an obvious approach to the problem
seems to add unnecessary policy to the kernel. Lumping together the
implementation of multiple instances of different namespaces in a way
that the implementation does not require.

Eric
