Subject: Re: [PATCH 1/4] Virtualization/containers: introduction
Posted by ebiederm on Wed, 08 Feb 2006 16:39:51 GMT
View Forum Message <> Reply to Message

Kirill Korotaev <dev@sw.ru> writes:

>> My point was to mainly identify the performance culprits and provide
>> an direct access to those "namespaces" for performance reasons.
>> So we all agreed on that we need to do that..
> After having looked at Eric's patch, I can tell that he does all these
> dereferences in quite the same amount.
>
> For example, lot's of skb->sk->host->...
> while in OpenVZ it would be econtainer()->... which is essentially
> current->container->...

Except at that point in time I cannot use current, because it
does not have necessarily have the appropriate context. So doubt
you could correctly use econtainer there.

> So until someone did measurements it looks doubtfull that one solution is better
> than the another.

I agree, exactly where we look is a minor matter, unless we can find
an instance where there are good reasons for preferring one representation
over another. Performance currently does not look a sufficient reason.

My basis for preferring a flat layout is essentially because
that is how other similar interfaces are currently implemented
in the kernel.

In linux we have a plan9 inspired system call called clone.
One of the ideas with clone is that when you create a new
task you either share or you don't share resources. Namespaces
are one of those resources.

Since we are dealing with concepts that appear to fit the
existing model beautifully my feeling is to use the existing
model of how things are currently implemented in the kernel.

The only other reason I can think of is namespace implementation
independence. If we have an additional structure that we collect
up the pointers it feels like it causes unnecessary tying.

The best justification I can think of for putting it all in
one structure seems like the current->container current->econtainer
thing that comes out of OpenVZ.  Currently I have followed the
threads enough to know it exists but I yet understand it.

If there is a good reason for the container/econtainer thing I can
certainly seem some benefit.

Eric

---