Subject: Re: [ckrm-tech] [PATCH 0/7] containers (V7): Generic Process Containers
Posted by Paul Menage on Tue, 13 Feb 2007 01:47:20 GMT
View Forum Message <> Reply to Message

On 2/12/07, Sam Vilain <sam@vilain.net> wrote:
>
> Not every module, you just make them on sensible, planned groupings.
> The danger is that the "container" group becomes a fallback grouping for
> things when people can't be bothered thinking about it properly, and
> everything including the kitchen sink gets thrown in. Then later you
> find a real use case where you don't want them together, but it's too
> late because it's already a part of the official API.

This paragraph seems to have a contradiction - first you say that
modules should use "planned groupings", then you complain that modules
using generic containers will find that they can't separate from other
modules "because they're part of an official API".

On the contrary - a module that's written over generic containers can
be combined with other such modules in the same groupings, or used in
different groupings, depending on what the user desires. This is
discussed in some detail in Documentation/containers.txt in patch 3 of
my patchset.

My patchset doesn't try to make all modules use the same *grouping*,
it tries to make them use the same *abstraction* (and hence share
code), but allows them to easily use different groupings just by
binding modules to different hierarchies.

>
> > As an example, the CPU accounting patch that in included in my patch
> > set as an illustration of a simple resource monitoring module is just
> > 250 lines, almost entirely in one file; if it also had to handle
> > associating tasks together into groups and presenting a filesystem
> > interface to the user it would be far larger and would have a much
> > bigger footprint on the kernel.
> >
>
> It's also less flexible. What if I want to do CPU accounting on some
> other boundaries than the "virtual server" a process is a part of?

No problem - that's why there are multiple hierarchies available. You
can mount the CPU accounting on one hierarchy, the virtual servers on
a different hierarchy, and they have completely independent mappings
from processes to containers - essentially parallel trees of
containers.

Paul

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