Subject: Re: [ckrm-tech] [patch00/05]: Containers(V2)- Introduction
Posted by Chandra Seetharaman on Thu, 21 Sep 2006 00:45:59 GMT
View Forum Message <> Reply to Message

On Wed, 2006-09-20 at 13:49 -0700, Paul Jackson wrote:

I concur with most of the comments (except as noted below)
> Paul M wrote:
> > Even if the resource control portions aren't totally compatible,
> > having two separate process container abstractions in the kernel is
> > sub-optimal
>
> At heart, CKRM (ne Resource Groups) are (well, have been until now)
> different than cpusets.
>
> Cpusets answers the question 'where', and Resource Groups 'how much'.
>
> The fundamental motivation behind cpusets was to be able to enforce
> job isolation. A job can get dedicated use of specified resources,
> -even- if it means those resources are severely underutilized by that
> job.
>
> The fundamental motivation (Chandra or others correct me if I'm wrong)
> of Resource Groups is to improve capacity utilization while limiting
> starvation due to greedy, competing users for the same resources.
>
> Cpusets seeks maximum isolation. Resource Groups seeks maximum
> capacity utilization while preserving guaranteed levels of quality
> of service.
>
> Cpusets are that wall between you and the neighbor you might not
> trust. Resource groups are a large family of modest wealth sitting
> down to share a meal.

I am thinking hard about how to bring guarantee into this picture :).

>
> It seems that cpusets can mimic memory resource groups. I don't

I am little confused w.r.t how cpuset can mimic memory resource groups.
How can cpuset provide support for over commit.

> see how cpusets could mimic other resource groups. But maybe I'm
> just being a dimm bulb.
>
--

Chandra Seetharaman | Be careful what you choose....
- sekharan@us.ibm.com | .....you may get it.
