Subject: Re: [ckrm-tech] [PATCH 00/10] Containers(V10): Generic Process Containers
Posted by serue on Thu, 07 Jun 2007 20:17:23 GMT
View Forum Message <> Reply to Message

Quoting Paul Jackson (pj@sgi.com):
> > For /cpusets/set0/set1 to have cpu 1 exclusively, does /cpusets/set0
> > also have to have it exclusively?
>
> Yes.
>
> > If so, then clearly this approach won't work, since if any container has
> > exclusive cpus, then every container will have siblings with exclusive
> > cpus, and unshare still isn't possible on the system.
>
> Well, if I'm following you, not exactly.
>
> If we have some exclusive flags set, then every top level container
> will have exclusive siblings, but further down the hierarchy, some
> subtree might be entirely free of any exclusive settings. Then nodes
> below the top of that subtree would not have exclusive set, and would
> not have any exclusive siblings.
>
> But, overall, yeah, exclusive is no friend of container cloning.
>
> I just wish I had been thinking harder about how container cloning
> will impact my life, and the lives of the customers in my cpuset
> intensive corner of the world.
>
> There are certainly a whole bunch of people who will never have any
> need for exclusive cpusets.
>
> Perhaps (speculating wildly from great ignorance) there are a whole
> bunch of people who will never have need for container cloning.
>
> And perhaps, hoping to get lucky here, the set of people who need both
> at the same time on the same system is sufficiently close to empty
> that we can just tell them tough toenails - you cannot do both at once.
>
> How wide spread will be the use of container cloning, if it proceeds
> as envisioned?

It's not just container cloning, but all namespace unsharing. So uses include (1) providing 'polyinstantiated directory' functionality, i.e. private per-user /tmp's or per-security-level /tmp and /home's. (2) any virtual server usage (3) hpc checkpoint/restart users.

> The set of people using exclusive cpusets is roughly some subset of

> those running multiple, cpuset isolated, non-cooperating jobs on big
> iron, usually with the aid of a batch scheduler.

Unfortunately I would imagine these users to be very intereseted in
providing checkpoint/restart/migrate functionality.

> Well, that's what
> I am aware of anyway. If there are any other friends of exclusive
> cpusets lurking here, you might want to speak up, before I sell your
> interests down the river.
>
> --
> I won't rest till it's the best ...
> Programmer, Linux Scalability
> Paul Jackson <pj@sgi.com> 1.925.600.0401

Can you explain to me, though, why it should be that if /cpusets/set0
has access to cpus 0-8, and /cpusets/set0/set1 has exclusive access to
cpus 0-2, and /cpusets/set0/set2 has exclusive access to cpus 3-4,
why i a process in /cpusets/set0 creates /cpusets/set0/set3 through
container_clone, it would be unsafe to have it automatically get cpus 5-8?

Surely if the admin wants to give cpus 5-6 exclusively to /cpusets/set0/set4
later, those cpus can just be taken away from set3?

thanks,
-serge