Subject: Re: [ckrm-tech] [PATCH] BC: resource beancounters (v4) (added user memory)
Posted by Rohit Seth on Wed, 13 Sep 2006 01:25:51 GMT
View Forum Message <> Reply to Message

On Tue, 2006-09-12 at 18:10 -0700, Chandra Seetharaman wrote:
> On Tue, 2006-09-12 at 17:39 -0700, Rohit Seth wrote:
> <snip>
> > > yes, it would be there, but is not heavy, IMO.
> > >
> > I think anything greater than 1% could be a concern for people who are
> > not very interested in containers but would be forced to live with them.
> >
> If they are not interested in resource management and/or containers, i
> do not think they need to pay.
> >

Think of a single kernel from a vendor that has container support built in.

> > > >
> > > > > >
> > > > > > And anything running outside a container should be limited by default
> > > > > > Linux settings.
> > > > > >
> > > > > > note that the resource available to the default RG will be (total system
> > > > > > resource - allocated to RGs).
> > > > >
> > > > > I think it will be preferable to not change the existing behavior for
> > > > > applications that are running outside any container (in your case
> > > > > default resource group).
> > > >
> > > > hmm, when you provide QoS for a set of apps, you will affect (the
> > > > resource availability of) other apps. I don't see any way around it. Any
> > > > ideas ?
> > >
> > When I say, existing behavior, I mean not getting impacted by some
> > artificial limits that are imposed by container subsystem. IOW, if a
> >
> That is what I understood and replied above.
> > sysadmin is okay to have certain apps running outside of container then
> > he is basically forgoing any QoS for any container on that system.
> >
> Not at all. If the container they are interested in is guaranteed, I do
> not see how apps running outside a container would affect them.
>

Because the kernel (outside the container subsystem) doesn't know of

these guarantees...unless you modify the page allocator to have another
variant of overcommit memory.

> <snip>
> > > > > Not really.
> > > > > - Each RG will have a guarantee and limit of each resource.
> > > > > - default RG will have (system resource - sum of guarantees)
> > > > > - Every RG will be guaranteed some amount of resource to provide QoS
> > > > > - Every RG will be limited at "limit" to prevent DoS attacks.
> > > > > - Whoever doesn't care either of those set them to don't care values.
> > > > > >
> > > > >
> > > > > For the cases that put this don't care, do you depend on existing
> > > > > reclaim algorithm (for memory) in kernel?
> > > >
> > > > Yes.
> > >
> > > So one container with these don't care condition(s) can turn the whole
> > > guarantee thing bad. Because existing kernel reclaimer does not know
> > > about memory commitments to other containers. Right?
> >
> > No, the reclaimer would free up pages associated with the don't care RGs
> > (as the user don't care about the resource made available to them).
> >

And how will the kernel reclaimer know which RGs are don't care?

-rohit