Subject: Re:  Re: [ckrm-tech] [RFC] Resource Management - Infrastructure choices
Posted by Matt Helsley on Thu, 02 Nov 2006 00:31:37 GMT
View Forum Message <> Reply to Message

On Thu, 2006-11-02 at 02:01 +0300, Kir Kolyshkin wrote:
> Chris Friesen wrote:
> > Srivatsa Vaddagiri wrote:
> >
> >>>>      - Support limit (soft and/or hard depending on the resource
> >>>>       type) in controllers. Guarantee feature could be indirectly
> >>>>       met thr limits.
> >
> > I just thought I'd weigh in on this.  As far as our usage pattern is
> > concerned, guarantees cannot be met via limits.
> >
> > I want to give "x" cpu to container X, "y" cpu to container Y, and "z"
> > cpu to container Z.
> >
> > If these are percentages, x+y+z must be less than 100.
> >
> > However, if Y does not use its share of the cpu, I would like the
> > leftover cpu time to be made available to X and Z, in a ratio based on
> > their allocated weights.
> >
> > With limits, I don't see how I can get the ability for containers to
> > make opportunistic use of cpu that becomes available.
> This is basically how "cpuunits" in OpenVZ works. It is not limiting a
> container in any way, just assigns some relative "units" to it, with sum
> of all units across all containers equal to 100% CPU. Thus, if we have

 So the user doesn't really specify percentage but values that feed into
ratios used by the underlying controller? If so then it's not terribly
different from the "shares" of single level of Resource Groups.

 Resource groups goes one step further and defines a denominator for
child groups to use. This allows the shares to be connected vertically
so that changes don't need to propagate beyond the parent and child
groups.

> cpuunits 10, 20, and 30 assigned to containers X, Y, and Z, and run some
> CPU-intensive tasks in all the containers, X will be given
> 10/(10+20+30), or 20% of CPU time, Y -- 20/50, i.e. 40%, while Z gets

nit: I don't think this math is correct.

Shouldn't they all have the same denominator (60), or am I
misunderstanding something?

If so then it should be:
X = 10/60     16.666...%
Y = 20/60     33.333...%
Z = 30/60     50.0%
Total:        100.0%

> 60%. Now, if Z is not using CPU, X will be given 33% and Y -- 66%. The
> scheduler used is based on a per-VE runqueues, is quite fair, and works
> fine and fair for, say, uneven case of 3 containers on a 4 CPU box.

<snip>

Cheers,
 -Matt Helsley