Subject: Re: [PATCH v2 04/11] kmem accounting basic infrastructure
Posted by Michal Hocko on Wed, 15 Aug 2012 13:26:21 GMT
View Forum Message <> Reply to Message

On Wed 15-08-12 17:04:31, Glauber Costa wrote:
> On 08/15/2012 05:02 PM, Michal Hocko wrote:
> > On Wed 15-08-12 16:53:40, Glauber Costa wrote:
> > [...]
> >>>>> This doesn't check for the hierachy so kmem_accounted might not be in
> >>>>> sync with it's parents. mem_cgroup_create (below) needs to copy
> >>>>> kmem_accounted down from the parent and the above needs to check if this
> >>>>> is a similar dance like mem_cgroup_oom_control_write.
> >>>>>
> >>>>
> >>>> I don't see why we have to.
> >>>>
> >>>> I believe in a A/B/C hierarchy, C should be perfectly able to set a
> >>>> different limit than its parents. Note that this is not a boolean.
> >>>
> >>> Ohh, I wasn't clear enough. I am not against setting the _limit_ I just
> >>> meant that the kmem_accounted should be consistent within the hierarchy.
> >>>
> >>
> >> If a parent of yours is accounted, you get accounted as well. This is
> >> not the state in this patch, but gets added later. Isn't this enough ?
> >
> > But if the parent is not accounted, you can set the children to be
> > accounted, right? Or maybe this is changed later in the series? I didn't
> > get to the end yet.
> >
>
> Yes, you can. Do you see any problem with that?

Well, if a child contributes with the kmem charges upwards the hierachy
then a parent can have kmem.usage > 0 with disabled accounting.
I am not saying this is a no-go but it definitely is confusing and I do
not see any good reason for it. I've considered it as an overlook rather
than a deliberate design decision.
--
Michal Hocko
SUSE Labs