Subject: Re: [RFD][PATCH] memcg: Move Usage at Task Move
Posted by KAMEZAWA Hiroyuki on Wed, 11 Jun 2008 08:26:37 GMT
View Forum Message <> Reply to Message

On Wed, 11 Jun 2008 01:04:14 -0700
"Paul Menage" <menage@google.com> wrote:

> An alternative way to support that would be to do nothing at move
> time, but provide a "pull_usage" control file that would slurp any
> pages in any mm in the cgroup into the cgroup.
> >> >
> >> > One reasone is that I think a typical usage of memory controller is
> >> > fork()->move->exec(). (by libcg ?) and exec() will flush the all usage.
> >>
> >> Exactly - this is a good reason *not* to implement move - because then
> >> you drag all the usage of the middleware daemon into the new cgroup.
> >>
> > Yes but this is one of the usage of cgroup. In general, system admin can
> > use this for limiting memory on his own decision.
> >
>
> Sorry, your last sentence doesn't make sense to me in this context.
>
Sorry. try another sentense..

I think cgroup itself is designed to be able to be used without middleware.
IOW, whether using middleware or not is the matter of users not of developpers.
There will be a system that system admin controlles all and move tasks by hand.
ex)...personal notebooks etc..

> If the common mode for middleware starting a new cgroup is fork() /
> move / exec() then after the fork(), the child will be sharing pages
> with the main daemon process. So the move will pull all the daemon's
> memory into the new cgroup
>
My patch (this patch) just moves Private Anon page to new cgroup. (of mapcount=1)

> > yes. but, at first, I'll try no-rollback approach.
> > And can I move memory resource controller's subsys_id to the last for now ?
> >
>
> That's probably fine for experimentation, but it wouldn't be something
> we'd want to commit to -mm or mainline.
>

Hmm, I'd like to post a patch to add "rollback" to cgroup if I find it necessary.

My first purpose of this post is showing the problem and starting discussion.
Anyway, I will remove "RFC" only when I got enough number of Acks.

Thanks,
-Kame