Subject: Re: [RFD][PATCH] memcg: Move Usage at Task Move
Posted by KAMEZAWA Hiroyuki on Wed, 11 Jun 2008 04:14:37 GMT
View Forum Message <> Reply to Message

On Wed, 11 Jun 2008 12:44:46 +0900 (JST)
yamamoto@valinux.co.jp (YAMAMOTO Takashi) wrote:

> > I'm now considering following logic. How do you think ?
> >
> > Assume: move TASK from group:CURR to group:DEST.
> >
> > == move_task(TASK, CURR, DEST)
> >
> > if (DEST's limit is unlimited)
> > moving TASK
> > return success.
> >
> > usage = check_usage_of_task(TASK).
> >
> > /* try to reserve enough room in destionation */
> > if (try_to_reserve_enough_room(DEST, usage)) {
> > move TASK to DEST and move pages AMAP.
> > /* usage_of_task(TASK) can be changed while we do this.
> > Then, we move AMAP. */
> > return success;
> > }
> > return failure.
> > ==
>
> AMAP means that you might leave some random charges in CURR?
>
yes. but we can reduce bad case by some way
 - reserve more than necessary.
 or
 - read_lock mm->sem while move.

> i think that you can redirect new charges in TASK to DEST
> so that usage_of_task(TASK) will not grow.
>

Hmm, to do that, we have to handle complicated cgroup's attach ops.

at this moving, memcg is pointed by
 - TASK->cgroup->memcg(CURR)
after move
 - TASK->another_cgroup->memcg(DEST)

This move happens before cgroup is replaced by another_cgroup.

Thanks,
-Kame

_____

Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers