Subject: Re: [RFD][PATCH] memcg: Move Usage at Task Move
Posted by yamamoto on Tue, 10 Jun 2008 05:50:32 GMT
View Forum Message <> Reply to Message

> For avoiding complicated rollbacks,
> I think of following ways of policy for task moving (you can add here.)
>
> 1. Before moving usage, reserve usage in the new cgroup and old cgroup.
> Pros.
> - rollback will be very easy.
> Cons.
> - A task will use twice of its own usage virtaually for a while.
> - some amount of cpu time will be necessary to move _Big_ apps.
> - It's difficut to move _Big_ apps to small memcg.
> - we have to add "special case" handling.
>
> 2. Don't move any usage at task move. (current implementation.)
> Pros.
> - no complication in the code.
> Cons.
> - A task's usage is chareged to wrong cgroup.
> - Not sure, but I believe the users don't want this.
>
> 3. Use Lazy Manner
> When the task moves, we can mark the pages used by it as
> "Wrong Charge, Should be dropped", and add them some penalty in the LRU.
> Pros.
> - no complicated ones.
> - the pages will be gradually moved at memory pressure.
> Cons.
> - A task's usage can exceed the limit for a while.
> - can't handle mlocked() memory in proper way.
>
> 4. Allow Half-moved state and abandon rollback.
> Pros.
> - no complicated ones in the code.
> Cons.
> - the users will be in chaos.

how about:

5. try to move charges as your patch does.
if the target cgroup's usage is going to exceed the limit,
try to shrink it. if it failed, just leave it exceeded.
(ie. no rollback)
for the memory subsystem, which can use its OOM killer,
the failure should be rare.

> After writing this patch, for me, "3" is attractive. now.
> (or using Lazy manner and allow moving of usage instead of freeing it.)
>
> 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.

i guess that moving long-running applications can be desirable
esp. for not so well-designed systems.

YAMAMOTO Takashi

_____

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