
Subject: Re: [PATCH 02/11] memcg: Reclaim when more than one page needed.
Posted by [Suleiman Souhlal](#) on Mon, 25 Jun 2012 23:33:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Mon, Jun 25, 2012 at 7:15 AM, Glauber Costa <glommer@parallels.com> wrote:

> From: Suleiman Souhlal <ssouhlal@FreeBSD.org>

>

> mem_cgroup_do_charge() was written before slab accounting, and expects

> three cases: being called for 1 page, being called for a stock of 32 pages,

> or being called for a hugepage. If we call for 2 or 3 pages (and several

> slabs used in process creation are such, at least with the debug options I

> had), it assumed it's being called for stock and just retried without reclaiming.

>

> Fix that by passing down a minsize argument in addition to the csize.

>

> And what to do about that (csize == PAGE_SIZE && ret) retry? If it's

> needed at all (and presumably is since it's there, perhaps to handle

> races), then it should be extended to more than PAGE_SIZE, yet how far?

> And should there be a retry count limit, of what? For now retry up to

> COSTLY_ORDER (as page_alloc.c does), stay safe with a cond_resched(),

> and make sure not to do it if __GFP_NORETRY.

The commit description mentions COSTLY_ORDER, but it's not actually used in the patch.

-- Suleiman
