

---

Subject: Re: [PATCH 04/10] memcg: Introduce \_\_GFP\_NOACCOUNT.  
Posted by [Glauber Costa](#) on Tue, 06 Mar 2012 10:36:43 GMT

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

---

On 03/04/2012 04:10 AM, Suleiman Souhlal wrote:

> On Sat, Mar 3, 2012 at 3:24 PM, Glauber Costa<glommer@parallels.com> wrote:

>> On 03/03/2012 01:38 PM, Suleiman Souhlal wrote:

>>> Another possible example might be the skb data, which are just kcalloc

>>> and are already accounted by your TCP accounting changes, so we might

>>> not want to account them a second time.

>>

>>

>> How so?

>>

>> struct sk\_buff \* \_\_alloc\_skb(unsigned int size, gfp\_t gfp\_mask,  
>> int fclone, int node)

>> {

>> [ ... ]

>> cache = fclone ? skbuff\_fclone\_cache : skbuff\_head\_cache;

>>

>> /\* Get the HEAD \*/

>> skb = kmem\_cache\_alloc\_node(cache, gfp\_mask & ~\_\_GFP\_DMA, node);

>

> Just a few lines below:

>

> data = kcalloc\_node\_track\_caller(size, gfp\_mask, node);

>

> -- Suleiman

Can't we just make sure those come from the root cgroup's slabs?

Then we need no flag.

---

---

Subject: Re: [PATCH 04/10] memcg: Introduce \_\_GFP\_NOACCOUNT.  
Posted by [Suleiman Souhlal](#) on Tue, 06 Mar 2012 16:13:12 GMT

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

---

On Tue, Mar 6, 2012 at 2:36 AM, Glauber Costa <glommer@parallels.com> wrote:

> On 03/04/2012 04:10 AM, Suleiman Souhlal wrote:

>>

>> Just a few lines below:

>>

>> data = kcalloc\_node\_track\_caller(size, gfp\_mask, node);

>>

>> -- Suleiman

>

> Can't we just make sure those come from the root cgroup's slabs?

> Then we need no flag.

Do you mean make it so that all kmallocs come from the root cgroup's slabs?  
We would really like to account kmallocs in general (and all the other slab types) to the right cgroup...

That said, I'm probably going to concentrate on accounting specially marked caches only, for now, since there seems to be a strong opposition on accounting everything, even though I don't understand this point of view.

-- Suleiman

---