

---

Subject: Re: [PATCH 02/10] memcg: Uncharge all kmem when deleting a cgroup.  
Posted by [Suleiman Souhlal](#) on Wed, 29 Feb 2012 19:00:23 GMT

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

---

On Tue, Feb 28, 2012 at 10:22 PM, KAMEZAWA Hiroyuki  
<kamezawa.hiroyu@jp.fujitsu.com> wrote:  
> On Mon, 27 Feb 2012 14:58:45 -0800  
> Suleiman Souhlal <[ssouhlal@FreeBSD.org](mailto:ssouhlal@FreeBSD.org)> wrote:  
>  
>> A later patch will also use this to move the accounting to the root  
>> cgroup.  
>>  
>> Signed-off-by: Suleiman Souhlal <[suleiman@google.com](mailto:suleiman@google.com)>  
>> ---  
>> mm/memcontrol.c | 30 ++++++-----  
>> 1 files changed, 29 insertions(+), 1 deletions(-)  
>>  
>> diff --git a/mm/memcontrol.c b/mm/memcontrol.c  
>> index 11e31d6..6f44fcb 100644  
>> --- a/mm/memcontrol.c  
>> +++ b/mm/memcontrol.c  
>> @@ -378,6 +378,7 @@ static void mem\_cgroup\_get(struct mem\_cgroup \*memcg);  
>> static void mem\_cgroup\_put(struct mem\_cgroup \*memcg);  
>> static void memcg\_kmem\_init(struct mem\_cgroup \*memcg,  
>> struct mem\_cgroup \*parent);  
>> +static void memcg\_kmem\_move(struct mem\_cgroup \*memcg);  
>>  
>> /\* Writing them here to avoid exposing memcg's inner layout \*/  
>> #ifdef CONFIG\_CGROUP\_MEM\_RES\_CTLR\_KMEM  
>> @@ -3674,6 +3675,7 @@ static int mem\_cgroup\_force\_empty(struct mem\_cgroup \*memcg,  
bool free\_all)  
>> int ret;  
>> int node, zid, shrink;  
>> int nr\_retries = MEM\_CGROUP\_RECLAIM\_RETRIES;  
>> + unsigned long usage;  
>> struct cgroup \*cgrp = memcg->css.cgroup;  
>>  
>> css\_get(&memcg->css);  
>> @@ -3693,6 +3695,8 @@ move\_account:  
>> /\* This is for making all \*used\* pages to be on LRU. \*/  
>> lru\_add\_drain\_all();  
>> drain\_all\_stock\_sync(memcg);  
>> + if (!free\_all)  
>> memcg\_kmem\_move(memcg);  
>> ret = 0;  
>> mem\_cgroup\_start\_move(memcg);  
>> for\_each\_node\_state(node, N\_HIGH\_MEMORY) {  
>> @@ -3714,8 +3718,13 @@ move\_account:

```
>>         if (ret == -ENOMEM)
>>             goto try_to_free;
>>         cond_resched();
>> +
>>         usage = memcg->res.usage;
>> +#ifdef CONFIG_CGROUP_MEM_RES_CTLR_KMEM
>> +
>>         if (free_all && !memcg->independent_kmem_limit)
>>             usage -= memcg->kmem_bytes.usage;
>> +#endif
>
> Why we need this even if memcg_kmem_move() does uncharge ?
```

We need it when manually calling force\_empty.

-- Suleiman

---