Subject: Re: [PATCH v3 04/13] kmem accounting basic infrastructure
Posted by Glauber Costa on Thu, 27 Sep 2012 18:30:36 GMT
View Forum Message <> Reply to Message

On 09/27/2012 06:58 PM, Tejun Heo wrote:
> Hello, Mel.
>
> On Thu, Sep 27, 2012 at 03:43:07PM +0100, Mel Gorman wrote:
>>> I'm not too convinced. First of all, the overhead added by kmemcg
>>> isn't big.
>>
>> Really?
>>
>> If kmemcg was globally accounted then every __GFP_KMEMCG allocation in
>> the page allocator potentially ends up down in
>> __memcg_kmem_newpage_charge which
>>
>> 1. takes RCU read lock
>> 2. looks up cgroup from task
>> 3. takes a reference count
>> 4. memcg_charge_kmem -> __mem_cgroup_try_charge
>> 5. release reference count
>>
>> That's a *LOT* of work to incur for cgroups that do not care about kernel
>> accounting. This is why I thought it was reasonable that the kmem accounting
>> not be global.
>
> But that happens only when pages enter and leave slab and if it still
> is significant, we can try to further optimize charging. Given that
> this is only for cases where memcg is already in use and we provide a
> switch to disable it globally, I really don't think this warrants
> implementing fully hierarchy configuration.
>

Not totally true. We still have to match every allocation to the right
cache, and that is actually our heaviest hit, responsible for the 2, 3 %
we're seeing when this is enabled. It is the kind of path so hot that
people frown upon branches being added, so I don't think we'll ever get
this close to being free.