Subject: Re: [PATCH 5/7] shrink support for memcg kmem controller Posted by Glauber Costa on Wed, 22 Feb 2012 14:00:00 GMT

View Forum Message <> Reply to Message

```
On 02/22/2012 03:35 AM, Suleiman Souhlal wrote:
> On Tue, Feb 21, 2012 at 3:34 AM, Glauber Costa<glommer@parallels.com> wrote:
>
>> @ @ -5055,8 +5117,21 @ @ int memcg_kmem_newpage(struct mem_cgroup *memcg, struct
page *page, unsigned lon
>> {
>>
       unsigned long size = pages<< PAGE_SHIFT;
        struct res counter *fail:
>>
        int ret;
>> +
        bool do_softlimit;
>> +
>> +
>> +
        ret = res_counter_charge(memcg_kmem(memcg), size,&fail);
        if (unlikely(mem cgroup event ratelimit(memcg,
>> +
                                MEM_CGROUP_TARGET_THRESH))) {
>> +
>> +
             do softlimit = mem cgroup event ratelimit(memcg,
>> +
                                MEM_CGROUP_TARGET_SOFTLIMIT);
>> +
             mem cgroup threshold(memcg);
>> +
             if (unlikely(do_softlimit))
>> +
                  mem_cgroup_update_tree(memcg, page);
>> +
>> +
        }
>>
        return res_counter_charge(memcg_kmem(memcg), size,&fail);
>> -
        return ret;
>> +
>> }
> It seems like this might cause a lot of kernel memory allocations to
> fail whenever we are at the limit, even if we have a lot of
> reclaimable memory, when we don't have independent accounting.
> Would it be better to use __mem_cgroup_try_charge() here, when we
> don't have independent accounting, in order to deal with this
> situation?
>
Yes, it would.
I'll work on that.
```