Subject: Re: [PATCH v4 05/25] memcg: Always free struct memcg through schedule_work()
Posted by Glauber Costa on Mon, 18 Jun 2012 12:10:22 GMT
View Forum Message <> Reply to Message

On 06/18/2012 04:07 PM, Kamezawa Hiroyuki wrote:
> (2012/06/18 19:27), Glauber Costa wrote:
>> Right now we free struct memcg with kfree right after a
>> rcu grace period, but defer it if we need to use vfree() to get
>> rid of that memory area. We do that by need, because we need vfree
>> to be called in a process context.
>>
>> This patch unifies this behavior, by ensuring that even kfree will
>> happen in a separate thread. The goal is to have a stable place to
>> call the upcoming jump label destruction function outside the realm
>> of the complicated and quite far-reaching cgroup lock (that can't be
>> held when calling neither the cpu_hotplug.lock nor the jump_label_mutex)
>>
>> Signed-off-by: Glauber Costa<glommer@parallels.com>
>> CC: Tejun Heo<tj@kernel.org>
>> CC: Li Zefan<lizefan@huawei.com>
>> CC: Kamezawa Hiroyuki<kamezawa.hiroyu@jp.fujitsu.com>
>> CC: Johannes Weiner<hannes@cmpxchg.org>
>> CC: Michal Hocko<mhocko@suse.cz>
>
> How about cut out this patch and merge first as simple cleanu up and
> to reduce patch stack on your side ?
>
> Acked-by: KAMEZAWA Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com>

I believe this is already in the -mm tree (from the sock memcg fixes)

But actually, my main trouble with this series here, is that I am basing
it on Pekka's tree, while some of the fixes are in -mm already.
If I'd base it on -mm I would lose some of the stuff as well.

Maybe Pekka can merge the current -mm with his tree?

So far I am happy with getting comments from people about the code, so I
did not get overly concerned about that.