Subject: Re: [PATCH v2 04/29] slub: always get the cache from its page in kfree
Posted by Glauber Costa on Fri, 11 May 2012 18:42:42 GMT
View Forum Message <> Reply to Message

On 05/11/2012 03:32 PM, Christoph Lameter wrote:
> On Fri, 11 May 2012, Glauber Costa wrote:
>
>> Thank you in advance for your time reviewing this!
>
> Where do I find the rationale for all of this? Trouble is that pages can
> contain multiple objects f.e. so accounting of pages to groups is a bit fuzzy.
> I have not followed memcg too much since it is not relevant (actual
> it is potentially significantly harmful given the performance
> impact) to the work loads that I am using.
>
It's been spread during last discussions. The user-visible part is
documented in the last patch, but I'll try to use this space here to
summarize more of the internals (it can also go somewhere in the tree
if needed):

We want to limit the amount of kernel memory tasks inside a memory
cgroup use. slab is not the only one of them, but it is quite significant.

For that, the least invasive, and most reasonable way we found to do it,
is to create a copy of each slab inside the memcg. Or almost: we lazy
create them, so only slabs that are touched by the memcg are created.

So we don't mix pages from multiple memcgs in the same cache - we
believe that would be too confusing.

/proc/slabinfo reflects this information, by listing the memcg-specific
slabs.

This also appears in a memcg-specific memory.kmem.slabinfo.

Also note that accounting is not done until kernel memory is limited.
And if no memcg is limited, the code is wrapped inside static_key
branches. So it should be completely patched out if you don't put stuff
inside memcg.