Subject: Re: [PATCH v4 09/14] memcg: kmem accounting lifecycle management
Posted by Michal Hocko on Fri, 12 Oct 2012 08:41:01 GMT
View Forum Message <> Reply to Message

On Fri 12-10-12 11:47:17, Glauber Costa wrote:
> On 10/11/2012 05:11 PM, Michal Hocko wrote:
> > On Mon 08-10-12 14:06:15, Glauber Costa wrote:
> >> Because kmem charges can outlive the cgroup, we need to make sure that
> >> we won't free the memcg structure while charges are still in flight.
> >> For reviewing simplicity, the charge functions will issue
> >> mem_cgroup_get() at every charge, and mem_cgroup_put() at every
> >> uncharge.
> >>
> >> This can get expensive, however, and we can do better. mem_cgroup_get()
> >> only really needs to be issued once: when the first limit is set. In the
> >> same spirit, we only need to issue mem_cgroup_put() when the last charge
> >> is gone.
> >>
> >> We'll need an extra bit in kmem_accounted for that: KMEM_ACCOUNTED_DEAD.
> >> it will be set when the cgroup dies, if there are charges in the group.
> >> If there aren't, we can proceed right away.
> >>
> >> Our uncharge function will have to test that bit every time the charges
> >> drop to 0. Because that is not the likely output of
> >> res_counter_uncharge, this should not impose a big hit on us: it is
> >> certainly much better than a reference count decrease at every
> >> operation.
> >>
> >> [v3: merged all lifecycle related patches in one]
> >>
> >> Signed-off-by: Glauber Costa <glommer@parallels.com>
> >> CC: Kamezawa Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com>
> >> CC: Christoph Lameter <cl@linux.com>
> >> CC: Pekka Enberg <penberg@cs.helsinki.fi>
> >> CC: Michal Hocko <mhocko@suse.cz>
> >> CC: Johannes Weiner <hannes@cmpxchg.org>
> >> CC: Suleiman Souhlal <suleiman@google.com>
> >
> > OK, I like the optimization. I have just one comment to the
> > memcg_kmem_dead naming but other than that
> >
> > Acked-by: Michal Hocko <mhocko@suse.cz>
> >
> > [...]
> >> +static bool memcg_kmem_dead(struct mem_cgroup *memcg)
> >
> > The name is tricky because it doesn't tell you that it clears the flag
> > which made me scratch my head when reading comment in kmem_cgroup_destroy

> >
> memcg_kmem_finally_kill_that_bastard() ?

memcg_kmem_test_and_clear_dead? I know long but at least clear that the
flag is cleared. Or just open code it.
--
Michal Hocko
SUSE Labs
