

---

Subject: Re: [PATCH v3 12/13] execute the whole memcg freeing in rcu callback  
Posted by [Tejun Heo](#) on Fri, 21 Sep 2012 17:23:55 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hello, Glauber.

On Tue, Sep 18, 2012 at 06:04:09PM +0400, Glauber Costa wrote:

> A lot of the initialization we do in mem\_cgroup\_create() is done with softirqs  
> enabled. This include grabbing a css id, which holds &ss->id\_lock->rlock, and  
> the per-zone trees, which holds rtpz->lock->rlock. All of those signal to the  
> lockdep mechanism that those locks can be used in SOFTIRQ-ON-W context. This  
> means that the freeing of memcg structure must happen in a compatible context,  
> otherwise we'll get a deadlock.

Lockdep requires lock to be softirq or irq safe iff the lock is  
actually acquired from the said context. Merely using a lock with bh  
/ irq disabled doesn't signal that to lockdep; otherwise, we'll end up  
with enormous number of spurious warnings.

> The reference counting mechanism we use allows the memcg structure to be freed  
> later and outlive the actual memcg destruction from the filesystem. However, we  
> have little, if any, means to guarantee in which context the last memcg\_put  
> will happen. The best we can do is test it and try to make sure no invalid  
> context releases are happening. But as we add more code to memcg, the possible  
> interactions grow in number and expose more ways to get context conflicts.  
>  
> We already moved a part of the freeing to a worker thread to be context-safe  
> for the static branches disabling. I see no reason not to do it for the whole  
> freeing action. I consider this to be the safe choice.

And the above description too makes me scratch my head quite a bit. I  
can see what the patch is doing but can't understand the why.

\* Why was it punting the freeing to workqueue anyway? ISTR something  
about static\_keys but my memory fails. What changed? Why don't we  
need it anymore?

\* As for locking context, the above description seems a bit misleading  
to me. Synchronization constructs involved there currently doesn't  
require softirq or irq safe context. If that needs to change,  
that's fine but that's a completely different reason than given  
above.

Thanks.

--  
tejun

---