Subject: Re: [PATCH 5/6 mm] memcgroup: fix zone isolation OOM
Posted by KAMEZAWA Hiroyuki on Fri, 09 Nov 2007 09:27:29 GMT
View Forum Message <> Reply to Message

On Fri, 9 Nov 2007 07:13:22 +0000 (GMT)
Hugh Dickins <hugh@veritas.com> wrote:

> mem_cgroup_charge_common shows a tendency to OOM without good reason,
> when a memhog goes well beyond its rss limit but with plenty of swap
> available. Seen on x86 but not on PowerPC; seen when the next patch
> omits swapcache from memcgroup, but we presume it can happen without.
>
> mem_cgroup_isolate_pages is not quite satisfying reclaim's criteria
> for OOM avoidance. Already it has to scan beyond the nr_to_scan limit
> when it finds a !LRU page or an active page when handling inactive or
> an inactive page when handling active. It needs to do exactly the same
> when it finds a page from the wrong zone (the x86 tests had two zones,
> the PowerPC tests had only one).
>
> Don't increment scan and then decrement it in these cases, just move
> the incrementation down. Fix recent off-by-one when checking against
> nr_to_scan. Cut out "Check if the meta page went away from under us",
> presumably left over from early debugging: no amount of such checks
> could save us if this list really were being updated without locking.
>
> This change does make the unlimited scan while holding two spinlocks
> even worse - bad for latency and bad for containment; but that's a
> separate issue which is better left to be fixed a little later.
>

Okay, I agree with this logic for scan.

I'll consider some kind of optimization for avoiding all list scan
because of a zone's page is not included in cgroup's lru.

Maybe counting the number of active/inactive per zone (or per node) will
be first help.

Thanks,
-Kame

_____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers