
Subject: Re: [RFC][PATCH] memory cgroup enhancements updated [1/10]
try_to_free_mem_cgroup_pages bugfix
Posted by [Balbir Singh](#) on Tue, 23 Oct 2007 04:00:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

KAMEZAWA Hiroyuki wrote:

> Because NODE_DATA(node)->node_zonelists[] is guaranteed to contain
> all necessary zones, it is not necessary to use for_each_online_node.
>
> And this for_each_online_node() makes reclaim routine start always
> from node 0. This is bad.
>

But with this change, we'll start reclaim on the node that the task
hit the limit on - right?

> Signed-off-by: KAMEZAWA Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com>

>

>

>

> mm/vmscan.c | 8 +++-----

> 1 file changed, 3 insertions(+), 5 deletions(-)

>

> Index: devel-2.6.23-mm1/mm/vmscan.c

> =====

> --- devel-2.6.23-mm1.orig/mm/vmscan.c

> +++ devel-2.6.23-mm1/mm/vmscan.c

> @@ -1375,15 +1375,13 @@ unsigned long try_to_free_mem_cgroup_pag

> .mem_cgroup = mem_cont,

> .isolate_pages = mem_cgroup_isolate_pages,

> };

> - int node;

> + int node = numa_node_id();

> struct zone **zones;

> int target_zone = gfp_zone(GFP_HIGHUSER_MOVABLE);

>

> - for_each_online_node(node) {

> - zones = NODE_DATA(node)->node_zonelists[target_zone].zones;

> - if (do_try_to_free_pages(zones, sc.gfp_mask, &sc))

> + zones = NODE_DATA(node)->node_zonelists[target_zone].zones;

> + if (do_try_to_free_pages(zones, sc.gfp_mask, &sc))

> return 1;

> - }

> return 0;

> }

> #endif

>

--

Warm Regards,
Balbir Singh
Linux Technology Center
IBM, ISTL

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