

---

Subject: Re: [PATCH 2/3] memcggroup: fix typo in VM\_BUG\_ON()

Posted by [Li Zefan](#) on Mon, 18 Feb 2008 07:14:25 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Balbir Singh wrote:

> Li Zefan wrote:

>> Balbir Singh wrote:

>>> YAMAMOTO Takashi wrote:

>>>> Li Zefan wrote:

>>>>> No need for VM\_BUG\_ON(pc), since 'pc' is the list entry. This should

>>>>> be VM\_BUG\_ON(page).

>>>>>

>>>>> Signed-off-by: Li Zefan <lizf@cn.fujitsu.com>

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

>>>>> pc is of type page\_cgrouop and we use list\_for\_each\_entry\_safe\_reverse. Not sure

>>>>> why we can't bug on pc.

>>>> pc is dereferenced before this VM\_BUG\_ON.

>>>>

>>>> YAMAMOTO Takashi

>>>>

>>> OK, so the VM\_BUG\_ON needs to move to an earlier location. Agreed.

>>>

>> No, 'pc' has been dereferenced in list\_for\_each\_entry\_safe\_reverse().

>>

>>

>> #define list\_for\_each\_entry\_safe\_reverse(pos, n, head, member) \

>> for (pos = list\_entry((head)->prev, typeof(\*pos), member), \

>> n = list\_entry(pos->member.prev, typeof(\*pos), member); \

>> ^^^^^^^^^

>> &pos->member != (head); \

>> ^^^^^^^^^

>> pos = n, n = list\_entry(n->member.prev, typeof(\*n), member))

>>

>

> Hmm.. We used to have a for loop with !list\_empty() as a termination condition

> and VM\_BUG\_ON(!pc) is a spill over. With the new loop, VM\_BUG\_ON(!pc) does not

> make sense.

>

>

I see, and I'll post a new patch to just remove it.

---

Containers mailing list

[Containers@lists.linux-foundation.org](mailto:Containers@lists.linux-foundation.org)

<https://lists.linux-foundation.org/mailman/listinfo/containers>

---