

---

Subject: Re: [RFC] [-mm PATCH] Memory controller fix swap charging context in  
unuse\_pte()

Posted by [Hugh Dickins](#) on Mon, 22 Oct 2007 18:51:33 GMT

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

---

On Mon, 15 Oct 2007, Balbir Singh wrote:

> Hugh Dickins wrote:

> >

> > --- 2.6.23-rc8-mm2/mm/swapfile.c 2007-09-27 12:03:36.000000000 +0100

> > +++ linux/mm/swapfile.c 2007-10-07 14:33:05.000000000 +0100

> > @@ -507,11 +507,23 @@ unsigned int count\_swap\_pages(int type,

> > \* just let do\_wp\_page work it out if a write is requested later - to

> > \* force COW, vm\_page\_prot omits write permission from any private vma.

> > \*/

> > -static int unuse\_pte(struct vm\_area\_struct \*vma, pte\_t \*pte,

> > +static int unuse\_pte(struct vm\_area\_struct \*vma, pmd\_t \*pmd,

> > unsigned long addr, swp\_entry\_t entry, struct page \*page)

...

>

> I tested this patch and it seems to be working fine. I tried swapon -a

> in the middle of tests consuming swap. Not 100% rigorous, but a good

> test nevertheless.

>

> Tested-by: Balbir Singh <balbir@linux.vnet.ibm.com>

Thanks, Balbir. Sorry for the delay. I've not forgotten our agreement that I should be splitting it into before-and-after mem cgroup patches. But it's low priority for me until we're genuinely assigning to a cgroup there. Hope to get back to looking into that tomorrow, but no promises.

I think you still see no problem, where I claim that simply omitting the mem charge mods from mm/swap\_state.c leads to OOMs? Maybe our difference is because my memhog in the cgroup is using more memory than RAM, not just more memory than allowed to the cgroup. I suspect that arrives at a state (when the swapcache pages are not charged) where it cannot locate the pages it needs to reclaim to stay within its limit.

Hugh

---

Containers mailing list

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

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

---