Subject: Re: [PATCH -mm 5/5] swapcgroup (v3): implement force_empty
Posted by yamamoto on Fri, 04 Jul 2008 07:48:28 GMT
View Forum Message <> Reply to Message

> Hi, Yamamoto-san.
>
> Thank you for your comment.
>
> On Fri, 4 Jul 2008 15:54:31 +0900 (JST), yamamoto@valinux.co.jp (YAMAMOTO Takashi) wrote:
> > hi,
> >
> > > +/*
> > > + * uncharge all the entries that are charged to the group.
> > > + */
> > > +void __swap_cgroup_force_empty(struct mem_cgroup *mem)
> > > +{
> > > + struct swap_info_struct *p;
> > > + int type;
> > > +
> > > + spin_lock(&swap_lock);
> > > + for (type = swap_list.head; type >= 0; type = swap_info[type].next) {
> > > + p = swap_info + type;
> > > +
> > > + if ((p->flags & SWP_ACTIVE) == SWP_ACTIVE) {
> > > +   unsigned int i = 0;
> > > +
> > > +   spin_unlock(&swap_lock);
> >
> > what prevents the device from being swapoff'ed while you drop swap_lock?
> >
> Nothing.
>
> After searching the entry to be uncharged(find_next_to_unuse below),
> I recheck under swap_lock whether the entry is charged to the group.
> Even if the device is swapoff'ed, swap_off must have uncharged the entry,
> so I don't think it's needed anyway.
>
> > YAMAMOTO Takashi
> >
> > > +   while ((i = find_next_to_unuse(p, i, mem)) != 0) {
> > > +   spin_lock(&swap_lock);
> > > +   if (p->swap_map[i] && p->memcg[i] == mem)
> Ah, I think it should be added !p->swap_map to check the device has not
> been swapoff'ed.

find_next_to_unuse seems to have fragile assumptions and
can dereference p->swap_map as well.

YAMAMOTO Takashi

>
>
> Thanks,
> Daisuke Nishimura.
>
> > > +   swap_cgroup_uncharge(p, i);
> > > +   spin_unlock(&swap_lock);
> > > +   }
> > > +   spin_lock(&swap_lock);
> > > + }
> > > + }
> > > + spin_unlock(&swap_lock);
> > > +
> > > + return;
> > > +}
> > > #endif

_____

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