Subject: Re: A consideration on memory controller.
Posted by Balbir Singh on Mon, 21 Jan 2008 09:50:36 GMT
View Forum Message <> Reply to Message

* KAMEZAWA Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com> [2008-01-21 18:19:20]:

> On Mon, 21 Jan 2008 13:58:52 +0530
> Balbir Singh <balbir@linux.vnet.ibm.com> wrote:
>
> > > If memory controller is used, we can limit maximum usage of memory per
> > > applications. Workload can be isolated per cgroup.
> > > This is good one progress. But maybe I need more features for my purpose....maybe.
> > >
> > > One consideration is...
> > > Now, memory controller can tamper LRU/relcaim handling but cannot do
> > > free memory. For guaranteing amount of usable memory for an applicatons,
> > > using VM is the best answer.
> >
> > This is a hard question? In the past it has been suggested that we use
> > hard limits to implement guarantees. Once we have the kernel memory
> > controller, guarantees might be easier to implement (we need account
> > for non-reclaimable resources)
> >
> yes, I'm looking forward to see the kernel memory controller.
> But maybe guarantee amount of *immediately usable* memory (like mempool)
> for cgroup is not the same issue as to guarantee free-cache for kernel
> memory.
>
>
> >
> > But sometimes it can't be used.
> > > I'm wondering whether we can add free-memory controller or not. It will
> > > gather free memory for some cgroup with low <-> min <-> high + page-order setup
> > > and work as buffer within cgroup <-> system workload.
> > > But I'm not sure this idea is good or not ;)
> > >
> >
> > I think it might be good to explore it more. The other idea is to
> > limit a soft-limit, such that memory is only reclaimed when there is
> > memory pressure.
> >
> thanks, I'll dig more.
>
> > > - back ground reclaim (Maybe it's better to wait for RvR's LRU set merge.)
> > > - guarantee some amount of memory not to be reclaimed by global LRU.
> > > - per cgroup swappiness.
> > > - swap controller. (limit swap usage...maybe independet from memory
> > >   controller.)

> > >
> > > belows are no patch, no plan topics.
> > > - limit amount of mlock.
> > > - limit amount of hugepages.
> > > - more parameters for page reclaim.
> > > - balancing on NUMA (if we can find good algorythm...)
> > > - dirty_ratio per cgroup.
> > >
> > > - multi-level memory controller.
> > >
> > We might also need to consider the following
> >
> > 1. Implementation of shares
> > 2. Implementation of virtual memory limit
> limiting virtual memory like vm.overcommit_memory ?
>

Sort of, yes. The main idea is to limit paging rate and swap usage of
the control group.

>
> > > If you have feature-lists against memory controller, I'd like to see.
> > >
> > >
> > > Note:
> > > In last year, limit size of page-cache was posted but denied. It is said that
> > > free memory is bad memory. Now, I never think anything just for limitig
> > > page-cache will be accepted.
> > >
> >
> > This topic needs more discussion, we have some form of page-cache
> > control built into the memory controller.
> >
> Hmm. ok. I'looking forward to see.
>

Could you elaborate on what sort of page-cache control you need, is it
global page-cache control?

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

--

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