Subject: Re: Re: [ckrm-tech] containers development plans (July 10 version)
Posted by Herbert Poetzl on Wed, 11 Jul 2007 18:59:07 GMT
View Forum Message <> Reply to Message

On Wed, Jul 11, 2007 at 11:04:06AM -0700, Dave Hansen wrote:
> On Wed, 2007-07-11 at 21:18 +0900, Takenori Nagano wrote:
> > I think Balbir's idea is very simple and reasonable way to develop
> > per container swapping. Because kernel needs the information that
> > a target page belongs to which container. Fortunately, we already
> > had page based memory management system which included in RSS
> > controller. I think it is appropriate that we develop per container
> > swapping on page based memory management system.
>
> There are a couple of concepts being thrown about here, so let's
> separate them out a bit.
>
> 1. Limit a container's usage of swap.
>   - Keep track of how many swap pages a container uses
>   - go OOM on the container when it exceeds its allowed usage
>   - tracking will be on a container's use of swap globally, no matter
>   what swap device or file it is actually allocated in
>   - all containers share all swapfiles

this is probably what Linux-VServer would prefer,
but the aim would be to allow certain contexts to
keep contexts from swapping out even when over
their memory limits as long as there is enough
memory available (could be reservation or best
efford based)

> 2. Keep separate lists of swap devices for each container
>   - each container is allowed to use a subset of the system's
>   swap files

sounds okay too ...

>   eventually:
>   - keep a per-container list of which pte values correspond
>   to which swapfiles
>   - pte swap values are only valid inside of one container

smells like additional memory and cpu overhead

> 3. Use a completely isolated set of swapfiles from (2) for
>   checkpoint/restart
>   - ensures that any swapfile will only contain data from one container
>
> The idea in (1) is not very useful for checkpoint/restart, but it would

> be useful to solve the cpuset OOM problem described in the VM BOF.
> (That problem is basically that a cpuset with available memory but a
> large amount in swap can cause another cpuset to go OOM. The memory
> footprint in the system is under RAM+swap, but the OOM still happens.)

best,
Herbert

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

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