Subject: Re: [RFC][PATCH] allow "unlimited" limit value.
Posted by Paul Menage on Wed, 26 Sep 2007 00:06:21 GMT
View Forum Message <> Reply to Message

On 9/25/07, David Rientjes <rientjes@google.com> wrote:
>
> Having the limit expressed and configurable in bytes suggests that it can
> be defined on that granularity which is obviously wrong.

One of the other options suggested was that you can write a value in
bytes, and the value you can read back from there will reflect the
real limit, with any associated granularity/rounding.

>
> > > So by expressing it in terms of bytes instead of kilobytes, you're just
> > > making the largest amount of memory allowed via this interface smaller
> > > that is should have to be.
> >
> > Yes, that's true. With a 64-bit count in bytes, we can only limit
> > people to 16 exabytes of memory. We're going to bump up against that
> > limit in no time.
> >
>
> So, by your logic, it would be fine to express it in bits too.

I don't think it would be much of a scalability limit to express it in
bits, no. Of course, it would be a bit silly. Bytes are the natural
counting units for memory - e.g. they're the units you get back when
you call sizeof(), or you pass to malloc().

>
> Please cite examples of other memory controllers that you can imagine
> would actually support (not expose to userspace, but support) memory
> limits in terms of anything smaller than kilobytes

Pavel's kernel memory controller, posted to this list this morning,
appears to charge for each object based on its size in bytes.

I could also imagine that a filesystem that packs short files or tails
into partial pages could charge based on those partial pages, although
I don't know of any such controller.

Paul

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