Subject: Re: [PATCH] BC: resource beancounters (v2)
Posted by Andrew Morton on Wed, 23 Aug 2006 17:05:32 GMT
View Forum Message <> Reply to Message

On Wed, 23 Aug 2006 14:46:19 +0400
Kirill Korotaev <dev@sw.ru> wrote:

> The following patch set presents base of
> Resource Beancounters (BC).
> BC allows to account and control consumption
> of kernel resources used by group of processes.
>
> Draft UBC description on OpenVZ wiki can be found at
> http://wiki.openvz.org/UBC_parameters
>
> The full BC patch set allows to control:
> - kernel memory. All the kernel objects allocatable
> on user demand should be accounted and limited
> for DoS protection.
> E.g. page tables, task structs, vmas etc.
>
> - virtual memory pages. BCs allow to
> limit a container to some amount of memory and
> introduces 2-level OOM killer taking into account
> container's consumption.
> pages shared between containers are correctly
> charged as fractions (tunable).
>
> - network buffers. These includes TCP/IP rcv/snd
> buffers, dgram snd buffers, unix, netlinks and
> other buffers.
>
> - minor resources accounted/limited by number:
> tasks, files, flocks, ptys, siginfo, pinned dcache
> mem, sockets, iptentries (for containers with
> virtualized networking)
>
> As the first step we want to propose for discussion
> the most complicated parts of resource management:
> kernel memory and virtual memory.

The patches look reasonable to me - mergeable after updating them for
today's batch of review commentlets.

I have two high-level problems though.

a) I don't yet have a sense of whether this implementation
   is appropriate/sufficient for the various other

applications which people are working on.

If the general shape is OK and we think this
implementation can be grown into one which everyone can
use then fine.

And...

> The patch set to be sent provides core for BC and
> management of kernel memory only. Virtual memory
> management will be sent in a couple of days.

We need to go over this work before we can commit to the BC
core. Last time I looked at the VM accounting patch it
seemed rather unpleasing from a maintainability POV.

And, if I understand it correctly, the only response to a job
going over its VM limits is to kill it, rather than trimming
it. Which sounds like a big problem?
