Subject: Re: [PATCH] BC: resource beancounters (v2)
Posted by Andrew Morton on Fri, 25 Aug 2006 14:30:03 GMT
View Forum Message <> Reply to Message

On Fri, 25 Aug 2006 15:49:15 +0400
Kirill Korotaev <dev@sw.ru> wrote:

> > 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.
> hmmm... in which regard?

Little changes all over the MM code which might get accidentally broken.

> > 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?
> No, UBC virtual memory management refuses occur on mmap()'s.

That's worse, isn't it? Firstly it rules out big sparse mappings and secondly

 mmap_and_use(80% of container size)
 fork_and_immediately_exec(/bin/true)

will fail at the fork?

> Andrey Savochkin wrote already a brief summary on vm resource management:
>
> ------------- cut ----------------
> The task of limiting a container to 4.5GB of memory bottles down to the
> question: what to do when the container starts to use more than assigned
> 4.5GB of memory?
>
> At this moment there are only 3 viable alternatives.
>
> A) Have separate memory management for each container,
>   with separate buddy allocator, lru lists, page replacement mechanism.
>   That implies a considerable overhead, and the main challenge there
>   is sharing of pages between these separate memory managers.
>
> B) Return errors on extension of mappings, but not on page faults, where
>   memory is actually consumed.
>   In this case it makes sense to take into account not only the size of used
>   memory, but the size of created mappings as well.
>   This is approximately what "privvmpages" accounting/limiting provides in
>   UBC.
>

> C) Rely on OOM killer.
>   This is a fall-back method in UBC, for the case "privvmpages" limits
>   still leave the possibility to overload the system.
>

D) Virtual scan of mm's in the over-limit container

E) Modify existing physical scanner to be able to skip pages which
   belong to not-over-limit containers.

F) Something else ;)