Subject: Re: [ckrm-tech] [PATCH] BC: resource beancounters (v4) (added user memory)
Posted by Pavel Emelianov on Fri, 15 Sep 2006 07:21:39 GMT
View Forum Message <> Reply to Message

Chandra Seetharaman wrote:
> On Thu, 2006-09-14 at 17:02 +0400, Pavel Emelianov wrote:
>
> <snip>
>
>> Reserving in advance means that sometimes you won't be able to start a
>> new group without taking back some of reserved pages. This is ... strange.
>>
>
> I do not see it strange. At the time of creation, user sees the failure
> (that there isn't enough resource to provide the required/requested
> guarantee) and can act accordingly.
>
> BTW, VMware does it this way.
>
And VPS density in VMware is MUCH lower than in
OpenVZ with beancounters :)
>
>> I think that a satisfactory solution now would be:
>> - limit unreclaimable memory during mmap() against soft limit to prevent
>>   potential rejects during page faults;
>>
>
> we can have guarantee and still handle it this way.
>
>> - reclaim memory in case of hitting hard limit;
>> - guarantees are done via setting soft and hard limits as I've shown
>> before.
>>
>
> complexity is high in doing that.
>
Nope. I've already said in another letter that a program of 60 lines
does this in a single loop.
>> The question still open is wether or not to account fractions.
>> I propose to skip fractions for a while and try to charge the page to
>> it's first user.
>>
>
> sounds fine
>
>
>> So final BC design is:

>> 1. three resources:
>> - kernel memory
>> - user unreclaimable memory
>> - user reclaimable memory
>>
>
> should be able to get other controllers also under this framework.
>
OK. But note, that it's easy to add new resource to current BC code.
The most difficult thing is placing 'charge/uncharge' calls over the kernel.
>
>> 2. unreclaimable memory is charged "in advance", reclaimable
>> is charged "on demand" with reclamation if needed
>> 3. each object (kernel one or user page) is charged to the
>> first user
>> 4. each resource controller declares it's own
>> - meaning of "limit" parameter (percent/size/bandwidth/etc)
>> - behaviour on changing limit (e.g. reclamation)
>> - behaviour on hitting the limit (e.g. reclamation)
>> 5. BC can be assigned to any task by pid (not just current)
>> without recharging currently charged resources.
>>
>
> Please see the emails i sent earlier in this context:
> http://marc.theaimsgroup.com/?l=ckrm-tech&m=115593001810 616&w=2
>
> We would need at least:
> - BC should be created/deleted explicitly by the user
> - cleaner interface for controller writers
>
OK.
Next week we'll try to send a new set of patches.