

---

Subject: Re: Question : memrlimit cgroup's task\_move (2.6.26-rc5-mm3)

Posted by [Balbir Singh](#) on Thu, 19 Jun 2008 12:30:24 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

KAMEZAWA Hiroyuki wrote:

> On Thu, 19 Jun 2008 12:24:29 +0900

> KAMEZAWA Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com> wrote:

>

>> On Thu, 19 Jun 2008 08:43:43 +0530

>> Balbir Singh <balbir@linux.vnet.ibm.com> wrote:

>>

>>>> I think the charge of the new group goes to minus. right ?

>>>> (and old group's charge never goes down.)

>>>> I don't think this is "no problem".

>>>>

>>>> What kind of patch is necessary to fix this ?

>>>> task\_attach() should be able to fail in future ?

>>>>

>>>> I'm sorry if I misunderstand something or this is already in TODO list.

>>>>

>>> It's already on the TODO list. Thanks for keeping me reminded about it.

>>>

>> Okay, I'm looking forward to see how can\_attach and roll-back(if necessary)

>> is implemented.

>> As you know, I'm interested in how to handle failure of task move.

>>

> One more thing...

> Now, charge is done at

>

> - vm is inserted (special case?)

> - vm is expanded (mmap is called, stack growth...)

>

> And uncharge is done at

> - vm is removed (success of munmap)

> - exit\_mm is called (exit of process)

>

> But it seems charging at may\_expand\_vm() is not good.

> The mmap can fail after may\_expand\_vm() because of various reason,

> but charge is already done at may\_expand\_vm()....and no roll-back.

>

> == an easy example of leak in stack growth handling ==

> [root@iridium kamezawa]# cat /opt/cgroup/test/memrlimit.usage\_in\_bytes

> 71921664

> [root@iridium kamezawa]# ulimit -s 3

> [root@iridium kamezawa]# ls

> Killed

> [root@iridium kamezawa]# ls

> Killed

```
> [root@iridium kamezawa]# ls
> Killed
> [root@iridium kamezawa]# ls
> Killed
> [root@iridium kamezawa]# ls
> Killed
> [root@iridium kamezawa]# ulimit -s unlimited
> [root@iridium kamezawa]# cat /opt/cgroup/test/memrlimit.usage_in_bytes
> 72368128
> [root@iridium kamezawa]#
```

Aaah.. I see.. I had it in place earlier, but moved them to `may_expand_vm()` on review suggestions. I can move it out or try to unroll when things fail. I'll experiment a bit more. Is there any particular method you prefer?

--

Warm Regards,  
Balbir Singh  
Linux Technology Center  
IBM, ISTL

---

Containers mailing list  
[Containers@lists.linux-foundation.org](mailto:Containers@lists.linux-foundation.org)  
<https://lists.linux-foundation.org/mailman/listinfo/containers>

---