

---

Subject: Re: [RFC][v2][patch 0/12][CFQ-cgroup]Yet another I/O bandwidth controlling subsystem for CGroups bas

Posted by [Andrea Righi](#) on Sun, 22 Jun 2008 17:04:33 GMT

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

---

Satoshi UCHIDA wrote:

> Hi, Andrea.

>

> Thanks for bug reports.

> I fix this problem.

>

> This problem causes by miss of trace for children groups.

> Please adopt and test this patch.

OK, I can confirm the fix resolves the problem.

Tested-by: Andrea Righi <[righi.andrea@gmail.com](mailto:righi.andrea@gmail.com)>

>

> If OK, this amendment is adopted when I release new patch-set.

>

>

> Regards,

> Satoshi Uchida.

>

>

> diff --git a/block/cfq-cgroup.c b/block/cfq-cgroup.c

> index f868f4f..64561f5 100644

> --- a/block/cfq-cgroup.c

> +++ b/block/cfq-cgroup.c

> @@ -183,7 +184,7 @@ static void \*cfq\_cgroup\_init\_cfq\_data(struct cfq\_cgroup \*cfqc, struct cfq\_data \*

>

> /\* check and create cfq\_data for children \*/

> if (cfqc->css.cgroup)

> - list\_for\_each\_entry(child, &cfqc->css.cgroup->children, children){

> + list\_for\_each\_entry(child, &cfqc->css.cgroup->children, sibling){

> cfq\_cgroup\_init\_cfq\_data(cgroup\_to\_cfq\_cgroup(child), cfqd);

> }

>

>

>> -----Original Message-----

>> From: Andrea Righi [<mailto:righi.andrea@gmail.com>]

>> Sent: Friday, May 30, 2008 7:37 PM

>> To: Satoshi UCHIDA

>> Cc: 'Ryo Tsuruta'; [axboe@kernel.dk](mailto:axboe@kernel.dk); [vtaras@openvz.org](mailto:vtaras@openvz.org);

>> [containers@lists.linux-foundation.org](mailto:containers@lists.linux-foundation.org); [tom-sugawara@ap.jp.nec.com](mailto:tom-sugawara@ap.jp.nec.com);

>> [linux-kernel@vger.kernel.org](mailto:linux-kernel@vger.kernel.org)

```

>> Subject: Re: [RFC][v2][patch 0/12][CFQ-cgroup]Yet another I/O bandwidth
>> controlling subsystem for CGroups based on CFQ
>>
>> Satoshi UCHIDA wrote:
>>> Hi, Tsuruta-san.
>>>>
>>>> I'm looking forward to your report.
>> Hi Satoshi,
>>
>> I'm testing your patch againsts latest Linus git and I've got the
>> following bug. It can be easily reproduced creating a cgroup, switching
>> the i/o scheduler from cfq to any other and switch back to cfq again.
>>
>> -Andrea
>>
>> BUG: unable to handle kernel paging request at fffffeb
>> IP: [

```

```
>> f0 75 5f
>> 8d 47 10 89 45 ec 89 c3 31 c0 eb 11 8b 56 7c 8d 41 04 <3b> 51 ec 8d 59
>> 08 0f 43
>> d8 89 c8 8b 0b 85 c9 75 e9 89 86 90 00
>> EIP: [<c0212dc6>] cfq_cgroup_sibling_tree_add+0x36/0x90 SS:ESP
>> 0068:c7afbe80
>> ---[ end trace 9701f4859bb53d27 ]---
```

---

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

---