
Subject: [PATCH] fix spurious EBUSY on memory cgroup removal

Posted by [yamamoto](#) on Fri, 01 Feb 2008 03:46:24 GMT

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

hi,

the following patch is to fix spurious EBUSY on cgroup removal.

YAMAMOTO Takashi

call mm_free_cgroup earlier.

otherwise a reference due to lazy mm switching can prevent cgroup removal.

Signed-off-by: YAMAMOTO Takashi <yamamoto@valinux.co.jp>

--- linux-2.6.24-rc8-mm1/kernel/fork.c.BACKUP 2008-01-23 14:43:29.000000000 +0900

+++ linux-2.6.24-rc8-mm1/kernel/fork.c 2008-01-31 17:26:31.000000000 +0900

@@ -393,7 +393,6 @@ void __mmdrop(struct mm_struct *mm)

```
{  
    BUG_ON(mm == &init_mm);  
    mm_free_pgd(mm);  
- mm_free_cgroup(mm);  
    destroy_context(mm);  
    free_mm(mm);  
}
```

@@ -415,6 +414,7 @@ void mmput(struct mm_struct *mm)

```
    spin_unlock(&mmlist_lock);  
}  
    put_swap_token(mm);  
+ mm_free_cgroup(mm);  
    mmdrop(mm);  
}  
}
```

Containers mailing list

Containers@lists.linux-foundation.org

<https://lists.linux-foundation.org/mailman/listinfo/containers>

Subject: Re: [PATCH] fix spurious EBUSY on memory cgroup removal

Posted by [Balbir Singh](#) on Sat, 02 Feb 2008 04:10:57 GMT

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

YAMAMOTO Takashi wrote:

> hi,

>

```

> the following patch is to fix spurious EBUSY on cgroup removal.
>
> YAMAMOTO Takashi
>
>
> call mm_free_cgroup earlier.
> otherwise a reference due to lazy mm switching can prevent cgroup removal.
>
> Signed-off-by: YAMAMOTO Takashi <yamamoto@valinux.co.jp>
> ---
>
> --- linux-2.6.24-rc8-mm1/kernel/fork.c.BACKUP 2008-01-23 14:43:29.000000000 +0900
> +++ linux-2.6.24-rc8-mm1/kernel/fork.c 2008-01-31 17:26:31.000000000 +0900
> @@ -393,7 +393,6 @@ void __mmdrop(struct mm_struct *mm)
> {
>     BUG_ON(mm == &init_mm);
>     mm_free_pgd(mm);
>     mm_free_cgroup(mm);
>     destroy_context(mm);
>     free_mm(mm);
> }
> @@ -415,6 +414,7 @@ void mmput(struct mm_struct *mm)
>     spin_unlock(&mmlist_lock);
> }
>     put_swap_token(mm);
> + mm_free_cgroup(mm);
>     mmdrop(mm);
> }
> }

```

The difference I see with the change is that mmput() will now call mm_free_cgroup() when mm->mm_users drop to 0 and __mmdrop() will call it when mm->mm_count drops to 0.

I think this change makes sense

Acked-by: Balbir Singh <balbir@linux.vnet.ibm.com>

--

Warm Regards,
Balbir Singh
Linux Technology Center
IBM, ISTL

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