
Subject: Re: [ckrm-tech] [PATCH 4/13] BC: context handling
Posted by [Paul Menage](#) on Sat, 25 Nov 2006 00:09:03 GMT
[View Forum Message](#) <> [Reply to Message](#)

On 11/24/06, Pavel Emelianov <xemul@openvz.org> wrote:

> I've got it! That's what will work:

```
>
> struct task_struct {
>     ...
>     struct beancounter *exec_bc;
>     struct beancounter *tmp_exec_bc; /* is set to NULL on
>                                     * tsk creation
>                                     */
> };
>
> struct beancounter get_exec_bc(void)
> {
>     if (current->tmp_exec_bc)
>         return current->tmp_exec_bc;
>     return rcu_dereference(current->exec_bc);
> }
```

Don't forget that this means all callers need to be in an
rcu_read_lock() section.

>
> I will implement this in the next beancounter patches.

This is looking remarkably like the mechanism in use for my generic
containers patches (inherited from Paul Jackson's cpusets code). In
the last set of patches that I posted on Wednesday night, I included
the example of the beancounters core and numfiles counter implemented
on top of the generic containers - basically pulling out the hash
table, reccounting and most of the configs code (since that's handled
by the generic containers), and moving the attribute management
configs code to the use the containerfs filesystem interface instead.
The rest is pretty much unchanged.

I think you could continue to use the tmp_exec_bc idea with this, and
have get_exec_bc() use the tmp_exec_bc if it existed, or else get the
bc pointer via the container system.

I'd appreciate any feedback you had on that approach.

Paul
