

---

Subject: Re: [PATCH 2/6] IPC namespace - utils  
Posted by [Cedric Le Goater](#) on Mon, 12 Jun 2006 21:05:34 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Eric W. Biederman wrote:

> Cedric Le Goater <clg@fr.ibm.com> writes:

>

>> I've used the ipc namespace patchset in rc6-mm2. Thanks for putting this  
>> together, it works pretty well ! A few questions when we clone :

>>

>> \* We should do something close to what exit\_sem() already does to clear the  
>> sem\_undo list from the task doing the clone() or unshare().

>

> Possibly which case are you trying to prevent?

task records a list of struct sem\_undo each containing a semaphore id. When we unshare ipc namespace, we break the 'reference' between the semaphore id and the struct sem\_array because the struct sem\_array are cleared and freed in the new namespace. When the task exit, that inconsistency could lead to unexpected results in exit\_sem(), task locks, BUG\_ON, etc. Nope ?

>> \* I don't like the idea of being able to unshare the ipc namespace and keep  
>> some shared memory from the previous ipc namespace mapped in the process mm.  
>> Should we forbid the unshare ?

>

> No. As long as the code handles that case properly we should be fine.

what is the proper way to handle that case ? the current patchset is not protected : a process can be in one ipc namespace and use a shared segment from a previous ipc namespace. This situation is not desirable in a migration scenario. May be asking too much for the moment ... and I agree this can be fixed by the way namespaces are created.

> As a general principle we should be able to keep things from other namespaces  
> open if we get them. The chroot or equivalent binary is the one that needs  
> to ensure these kinds of issues don't exist if we care.

>

> Speaking of we should put together a small test application probably similar  
> to chroot so people can access these features at least for testing.

are you thinking about a command unshare()ing each namespace or some kind of create\_nsproxy ?

> Ack. For the unshare fix below. Could you resend this one separately with  
> patch in the subject so Andrew sees it and picks up?

done.

thanks,

C.

---