Subject: Re: NFS causing oops when freeing namespace
Posted by Oleg Nesterov on Wed, 17 Jan 2007 18:58:23 GMT
View Forum Message <> Reply to Message

On 01/17, Daniel Hokka Zakrisson wrote:
>
> >> Call Trace:
> >> [<c03be6f0>] _spin_lock_irqsave+0x20/0x90
> >> [<c01f6115>] lockd_down+0x125/0x190
> >> [<c01d26bd>] nfs_free_server+0x6d/0xd0
> >> [<c01d8e9c>] nfs_kill_super+0xc/0x20
> >> [<c0161c5d>] deactivate_super+0x7d/0xa0
> >> [<c0175e0e>] release_mounts+0x6e/0x80
> >> [<c0175e86>] __put_mnt_ns+0x66/0x80
> >> [<c0132b3e>] free_nsproxy+0x5e/0x60
> >> [<c011f021>] do_exit+0x791/0x810
> >> [<c011f0c6>] do_group_exit+0x26/0x70
> >> [<c0103142>] sysenter_past_esp+0x5f/0x85
> >> [<c03b0033>] rpc_wake_up+0x3/0x70
>
> It was the only semi-plausible explanation I could come up with. I added a
> printk in do_exit right before exit_task_namespaces, where sighand was
> still set, and one right before the spin_lock_irq in lockd_down, where it
> had suddenly been set to NULL.

I can't reproduce the problem, but

```
 do_exit:
  exit_notify(tsk);
  exit_task_namespaces(tsk);
```

the task could be reaped by its parent in between.

We should not use ->signal/->sighand after exit_notify().

Can we move exit_task_namespaces() up?

Oleg.