Subject: Re: [PATCH 1/7][QUOTA] Move sysctl management code under ifdef CONFIG_SYSCTL
Posted by akpm on Tue, 04 Dec 2007 09:45:49 GMT
View Forum Message <> Reply to Message

On Tue, 04 Dec 2007 12:31:37 +0300 Pavel Emelyanov <xemul@openvz.org> wrote:

> Andrew Morton wrote:
> > On Tue, 04 Dec 2007 11:58:30 +0300 Pavel Emelyanov <xemul@openvz.org> wrote:
> >
> >>>> +#ifdef CONFIG_SYSCTL
> >>>>   register_sysctl_table(sys_table);
> >>>> +#endif
> >>>>
> >>>>   dquot_cachep = kmem_cache_create("dquot",
> >>>>   sizeof(struct dquot), sizeof(unsigned long) * 4,
> >>> We should avoid the ifdefs around the register_sysctl_table() call.
> >>>
> >>> At present the !CONFIG_SYSCTL implementation of register_sysctl_table() is
> >>> a non-inlined NULL-returning stub. All we have to do is to inline that stub
> >>> then these ifdefs can go away.
> >> What if some code checks for the return value to be not-NULL? In case
> >> CONFIG_SYSCTL=n this code will always think, that the registration failed.
> >
> > The stub function should return success?
>
> Well, I think yes. If some functionality is turned off, then the
> caller should think that everything is going fine (or he should
> explicitly removes the call to it with some other ifdef).
>
> At least this is true for stubs that return the error code, not
> the pointer. E.g. copy_semundo() always returns success if SYSVIPC
> is off, or namespaces cloning routines act in a similar way.
>
> Thus I though, that routines, that return pointers should better
> report that everything is OK (somehow) to reduce the number of
> "helpers" in the outer code. No?
>

Dunno. Returning NULL should be OK. If anyone is dereferenceing that
pointer with CONFIG_SYSCTL=n then they might need some attention?