Subject: Re: [PATCH 1/7][QUOTA] Move sysctl management code under ifdef CONFIG_SYSCTL
Posted by ebiederm on Tue, 04 Dec 2007 11:40:26 GMT
View Forum Message <> Reply to Message

Andrew Morton <akpm@linux-foundation.org> writes:

```
> 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?
```

We do have some current code in the network stack that fails miserably
when register_sysctl_table returns NULL, and there are explicit
checks for that.

Grr.

I had forgotten about that.

I expect the right answer is to simply have code ignore the fact
that register_sysctl_xxxx returns NULL, and not error on it.

The alternative is to get fancy and have everyone check the
return code and make the return type an IS_ERR thing. That seems
a lot more trouble then it is worth.

We can probably define it as register_sysctl_xxxx always returns
a token that must be passed to unregister_sysctl, and no errors
will be reported except to dmesg. That at sounds simple sane
and supportable from where we are now.

Eric