Subject: dead code: EXT3_IOC_WAIT_FOR_READONLY?
Posted by vaverin on Fri, 16 Mar 2007 11:16:28 GMT
View Forum Message <> Reply to Message

Hi all,

I've found some strange ext3 ioctl under CONFIG_JBD_DEBUG:
EXT3_IOC_WAIT_FOR_READONLY.

I had DEBUG_SPINLOCK enabled and my kernel was crashed on wait-for-readonly.c
test from ext3-tools because ioctl handler tries to use uninitialized
EXT3_SB(sb)->ro_wait_queue and waits EXT3_SB(sb)->turn_ro_timer that nobody uses
in kernel.

I do not understand the function of this ioctl and believe this code is dead
long time ago. From my POV it makes sense to remove it from kernel instead of
fixing it. Has somebody any objections?

Thank you,
Vasily Averin

Subject: Re: dead code: EXT3_IOC_WAIT_FOR_READONLY?
Posted by Andrew Morton on Fri, 16 Mar 2007 12:03:21 GMT
View Forum Message <> Reply to Message

On Fri, 16 Mar 2007 14:16:28 +0300 Vasily Averin <vvs@sw.ru> wrote:

> Hi all,
>
> I've found some strange ext3 ioctl under CONFIG_JBD_DEBUG:
> EXT3_IOC_WAIT_FOR_READONLY.
>
> I had DEBUG_SPINLOCK enabled and my kernel was crashed on wait-for-readonly.c
> test from ext3-tools because ioctl handler tries to use uninitialized
> EXT3_SB(sb)->ro_wait_queue and waits EXT3_SB(sb)->turn_ro_timer that nobody uses
> in kernel.
>
> I do not understand the function of this ioctl and believe this code is dead
> long time ago. From my POV it makes sense to remove it from kernel instead of
> fixing it. Has somebody any objections?

It exists to support my original ext3 recovery-testing code. The idea is
that you mount the filesytem with ro_after=1000, then after 1000 jiffies a
timer fires and the underlying device driver starts ignoring writes. You
then run the EXT3_IOC_WAIT_FOR_READONLY ioctl to wait for the timer to have
fired, then kill off the the stresstest programs, then unmount the fs, then
fsck it.

Yes, it's kind of dead code now - I don't think anyone has used it since 2001 or so.

We do need to resurrect the functionality: it was very powerful and found several problems.

People recently reported serious-looking ext3 recovery bugs but nobody seems interested in investigating them.

A reimplementation probaby wouldn't use any of the old code so sure, feel free to send in a patch which removes it.
