Subject: Re: [patch -mm] update mq_notify to use a struct pid
Posted by ebiederm on Mon, 11 Sep 2006 11:09:19 GMT
View Forum Message <> Reply to Message

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

> Eric W. Biederman wrote:
>> Cedric Le Goater <clg@fr.ibm.com> writes:
>>
>>> message queues can signal a process waiting for a message.
>>>
>>> this patch replaces the pid_t value with a struct pid to avoid pid wrap
>>> around problems.
>>>
>>> Signed-off-by: Cedric Le Goater <clg@fr.ibm.com>
>>> Cc: Eric Biederman <ebiederm@xmission.com>
>>> Cc: Andrew Morton <akpm@osdl.org>
>>> Cc: containers@lists.osdl.org
>>
>> Signed-off-by: Eric Biederman <ebiederm@xmission.com>
>>
>> I was just about to send out this patch in a couple more hours.
>
> Well, you did the same with the usb/devio.c and friends :)

Good. The you should be familiar enough with it to review my patch
and make certain I didn't do anything stupid :)

>> So expect the fact we wrote the same code is a good sign :)
>
> How does oleg feel about it ? I've seen some long thread on possible race
> conditions with put_pid() and solutions with rcu. I didn't quite get all of
> it ... it will need another run for me.

Short. Oleg felt it was a shame that locking was needed to use a
struct pid.

While parsing that I realized my second vt patch that deals with
vt_pid (the pid for console switching) has a subtle race, and
that patch needs to be reworked.

We confused each other. :)

> On the "pid_t to struct pid*" topic:
>
> * I started smbfs and realized it was useless.

Killing the user space part is useless?

I thought that is what I saw happening.

Of course I don't frequently mount smbfs.

> * in the following, the init process is being killed directly using 1. I'm
> not sure how useful it would be to use a struct pid. To begin with, may be
> they could use a :
>
> kill_init(int signum, int priv)

An interesting notion. The other half of them use cad_pid.
Converting that is going to need some sysctl work, so I have been
ignoring it temporarily.

Filling in a struct pid through sysctl is extremely ugly at the
moment, plus cad_pid needs some locking.

> ./arch/mips/sgi-ip32/ip32-reset.c
> ./arch/powerpc/platforms/iseries/mf.c
> ./drivers/parisc/power.c
> ./drivers/char/snsc_event.c
> ./kernel/sys.c
> ./kernel/sysctl.c
> ./drivers/char/nwbutton.c
> ./drivers/s390/s390mach.c
>
> * some more drivers,
> * some more kthread to convert

Ok. Time to exchange some status information, before I roll over and
go back to sleep.

My patch todo list (almost a series file) currently looks like:
> n_r396r
> fs3270-Change-to-use-struct-pid.txt
> smbfs-Make-conn_pid-a-struct-pid.txt
> ncpfs-Use-struct-pid-to-track-the-userspace-watchdog-process.txt
>
> Don-t-use-kill_pg-in-the-sunos-compatibility-code.txt
>
> usbatm-use-kthread-api (I think I have this one)
I did usbatm mostly to figure out why kthread conversions seem
to be so hard, and got lucky this one wasn't too ugly.

> The-dvb_core-needs-to-use-the-kthread-api-not-kernel-threads.txt
> nfs-Note-we-need-to-start-using-the-kthreads-api.txt

dvb-core I have only started looking at.
nfs I noticed it is the svc stuff that matters.

usbatm, dvb-core, and nfs are the 3 kernel_thread users
that also use kill_proc, and thus are high on my immediate hit list.

> pid-Replace-session_of_pgrp-with-pgrp_in_current_session.txt
> pid-Use-struct-pid-for-talking-about-process-groups-in-exit.c.txt
> pid-Replace-is_orphaned_pgrp-with-is_current_pgrp_orphaned.txt
>
> tty-Update-the-tty-layer-to-work-with-struct-pid.txt
I need to ensure I don't have a race with task->signal->tty_old_pgrp.
tty_old_pgrp is a weird notion that I haven't fully wrapped my head
around yet.

> pid-Remove-use-of-old-do_each_task_pid-while_each_task_pid.txt
>
> Rewrite-kill_something_info-so-it-uses-newer-helpers.txt
>
> pid-Remove-now-unused-do_each_task_pid-and-while_each_task_pid.txt
> Remove-the-now-unused-kill_pg-kill_pg_info-and-__kill_pg_info.txt
>
>
> pid-Better-tests-for-same-thread-group-membership.txt
> pid-Cleanup-the-pid-equality-tests.txt
> pid-Track-the-sending-pid-of-a-queued-signal.txt
> proc-Use-pid_nr-in-array.c-so-the-code-is-foobar-safe.txt
>
> sysctl-Implement-get_data-put_data.txt
>
> cad-pid (killing init)

Once the above list is processed none of the old none of the signal
functions that take a pid_t is needed anymore.
i.e. kill_proc, kill_pg, and do_each_task_pid will be removable.

I have at least a first draft of everything on my list except for the
kthread conversions which I just started messing with yesterday. But
don't worry about beating me to something if you feel you have it
complete. That just means I will have enough of a clue to review your
code :)

Eric

_____
Containers mailing list
Containers@lists.osdl.org