Subject: Re: [patch -mm] update mq_notify to use a struct pid
Posted by ebiederm on Mon, 11 Sep 2006 19:01:18 GMT
View Forum Message <> Reply to Message

Cedric you mentioned a couple of other patches that are in flight.
In the future could you please Cc: the containers list so independent
efforts are less likely to duplicate work, and we are more likely
to review each others patches instead?

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

> Eric W. Biederman wrote:
>
>>>> 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 :)
>
> well, the least i can try ...
>

>>> * I started smbfs and realized it was useless.
>>
>> Killing the user space part is useless?
>> I thought that is what I saw happening.
>
> smb_fill_super() says :
>
> if (warn_count < 5) {
>   warn_count++;
>   printk(KERN_EMERG "smbfs is deprecated and will be removed in"
>   " December, 2006. Please migrate to cifs\n");
> }
>
> So, i guess we should forget about it and spend our time on the cifs
> kthread instead.

Sure. Although in this instance the changes are simple enough I will probably
send the patch anyway :) That at least explains why you figured it was
useless work.

>> 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.
>
> yes.
>
>> 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.
>
> Which distros use /proc/sys/kernel/cad_pid and why ? I can image the need
> but i didn't find much on the topic.

I'm not at all certain, and I'm not even certain I care. The concept
is there in the code so it needs to be dealt with. Although if I we
extend the cad_pid concept it may make a difference.

>> My patch todo list (almost a series file) currently looks like:
>>> n_r396r
>>> fs3270-Change-to-use-struct-pid.txt
>
> done that. will send to martin for review.

Added to my queue of pending patches to look at review.

>>> 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.
>
> argh. i've done also and i just send my second version of the patch to the
> maintainer Duncan Sands.
>
> This one might just be useless also because greg kh has a patch in -mm to
> enable multithread probing of USB devices.

Added to my queue of pending patches to track down and reivew.

>>> 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.
>
> suka and i have sent patches to fix :
>
> drivers/media/video/tvaudio.c
> drivers/media/video/saa7134/saa7134-tvaudio.c
>
> we are no waiting for the maintainer feedback.

Ok. I think I saw a little of that.

>> 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.
>
> nfs is also full of signal_pending() ...

Yes, there is a lot to read and understand before I can confidently
do much with nfs.

>>> 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
>
> is that about updating the siginfos in collect_signal() to hold the right
> pid value depending on the pid namespace they are being received ?

Yes in send_signal, and in collect signal. To make it work easily I needed
to add a struct pid to struct sigqueue. So in send_signal I generate
the struct pid from the pid_t value and in collect signal I regenerate
the numeric value.

Eric