
Subject: Re: [patch -mm] update mq_notify to use a struct pid

Posted by [ebiederm](#) on Tue, 12 Sep 2006 16:03:32 GMT

[View Forum Message](#) <> [Reply to Message](#)

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

> Eric W. Biederman wrote:

>

> [...]

>

>> There is also the case that should not come up with signals where

>> we have a pid from a child namespace, that we should also be able to

>> compute the pid for.

>

> I don't understand how a signal can come from a child pid namespace ?

SIG_CHLD is the only case where I think we will be sending a signal from the child pid namespace.

Reading pids from the status files in /proc, from a parent namespace, is another example where we need to deal with the pid of children.

>> In essence I intend to have a list of pid_namespace, pid_t pairs connected

>> to a struct pid that we can look through to find the appropriate pid.

>

> yes, that's the purpose of pid_nr() I guess.

>

> This list would contain in nearly all cases a single pair (current pid

> namespace, pid value). It will contain 2 pairs for a task that has unshared

> its pid namespace : a pair for the current pid namespace, that needs to

> allocated when unshare() is called, and one pair for the ancestor pid

> namespace which is already allocated.

>

> Do you see more ?

I don't see the list getting longer until we get into a nested pid namespaces.

As long as the interface is well defined for the container in a container case I don't mind having additional restrictions.

I will note that you can get some extremely weird interactions if you do things like open a file descriptor in the parent pid namespace.

Fork two children each child in a different pid_namespaces.

fcntl(F_SETOWN) is called in one child, and fcntl(F_GETOWN) is called in the other child.

So we can't just call BUG_ON, if we have can't find the namespace.

But returning 0 from pid_nr should be fine.

Eric

Containers mailing list

Containers@lists.osdl.org

<https://lists.osdl.org/mailman/listinfo/containers>
