

---

Subject: Re: [patch -mm 1/5] mqueue namespace : add struct mq\_namespace

Posted by [Cedric Le Goater](#) on Wed, 03 Oct 2007 07:44:40 GMT

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

---

sukadev@us.ibm.com wrote:

> Cedric Le Goater [clg@fr.ibm.com] wrote:

> |  
> | >> however, we have an issue with the signal notification in \_\_do\_notify()  
> | >> we could kill a process in a different pid namespace.  
> | >  
> | > So I took a quick look at the code as it is (before this patchset)  
> | > and the taking a reference to a socket and the taking a reference to  
> | > a struct pid should do the right thing when we intersect with other  
> | > namespaces. It certainly does not look like a fundamental issue.  
> |  
> | right. this should be covered when the pid namespace signal handling is  
> | complete. kill\_pid\_info() should fail to send a signal to a sibling or  
> | a parent pid namespace.  
> |  
> | I guess we should add a WARNING() to say that we're attempting to do so.  
> |  
> Just want to clarify how a signal is sent to a parent ns.  
> |  
> A process P1 sets itself up to be notified when a message arrives  
> on a queue.  
> |  
> P1 then clones P2 with CLONE\_NEWPID.  
> |  
> P2 writes to the message queue and thus signals P1  
> |  
> What should the semantics be here ?  
> |  
> I guess it makes less sense for two namespaces to be dependent on the same  
> message queue this way. But, if P2 writes to the queue, technically, the  
> queue is not empty, so P1 should be notified, no ?  
> |  
> This sounds similar to the SIGIO signal case (F\_SETOWN). My understanding  
> was that we would notify whoever was set to receive the notification, even  
> if they were in a parent ns (again my reasoning was its based on the state  
> of a file).

yes

> IOW, should we change kill\_pid\_info() ? If the caller can 'see' the  
> 'struct pid' they can signal it. The expectation was that callers would  
> call find\_vpid() and thus only see processes in their namespace.

I think we have to decide on some limitations with signals and make sure that we cannot send a signal to a sibling pid namespace. This can occur in some special namespaces unshare configuration which should never be used but to make sure, let's add a big WARNING when we detect such a pid namespace violation.

If it is what you mean, I agree :)

Thanks,

C.

---

Containers mailing list  
Containers@lists.linux-foundation.org  
<https://lists.linux-foundation.org/mailman/listinfo/containers>

---