
Subject: Re: [PATCH] Masquerade sender information

Posted by [serue](#) on Fri, 02 Nov 2007 13:45:45 GMT

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

Quoting Cedric Le Goater (clg@fr.ibm.com):

> Serge E. Hallyn wrote:

> > Quoting Eric W. Biederman (ebiederm@xmission.com):

> > > sukadev@us.ibm.com writes:

> > >

> > > > From: Sukadev Bhattiprolu <sukadev@us.ibm.com>

> > > > Subject: [PATCH] Masquerade sender information

> > > >

> > > > With multiple pid namespaces, sender of a signal could be in an ancestor

> > > > namespace of the receiver and so the sender will not have a valid 'pid_t'

> > > > in the receiver's namespace.

> > > >

> > > > In this case, masquerade the 'siginfo' for the signal to pretend that the

> > > > signal originated from the kernel.

> > > At first glance this looks ok. I think the only case where we can

> > > be sending a signal from inside a pid namespace to something not

> > > in a child pid namespace is if we are the kernel. In which case

> > >

> > > Are we now blocking F_SETOWN|F_SETSIG signals to outside our pid

> > > namespace? mq_notify? (I didn't think we were)

> > >

> > My understanding is that we're not blocking and that a process killing

> > another process in a sibling pid namespace will have a si_pid = 0.

And I think I'm fine with that, I was just wondering about Eric's claim that only the kernel can send signals from inside a pidns to something not in a child pidns. We can treat these cases as being from the kernel, but it's not in fact the case that the signals came from the kernel.

-serge

Containers mailing list

Containers@lists.linux-foundation.org

<https://lists.linux-foundation.org/mailman/listinfo/containers>
