Subject: Re: [PATCH 2/2] Replace pid_t in autofs with struct pid reference
Posted by Herbert Poetzl on Thu, 22 Mar 2007 14:33:12 GMT
View Forum Message <> Reply to Message

On Thu, Mar 22, 2007 at 11:28:43AM +0900, Ian Kent wrote:
> On Wed, 2007-03-21 at 15:58 -0500, Serge E. Hallyn wrote:
> > Quoting Eric W. Biederman (ebiederm@xmission.com):
> > > "Serge E. Hallyn" <serue@us.ibm.com> writes:
> > >
> > > >> void autofs4_dentry_release(struct dentry *);
> > > >> extern void autofs4_kill_sb(struct super_block *);
> > > >> diff --git a/fs/autofs4/waitq.c b/fs/autofs4/waitq.c
> > > >> index 9857543..4a9ad9b 100644
> > > >> --- a/fs/autofs4/waitq.c
> > > >> +++ b/fs/autofs4/waitq.c
> > > >> @@ -141,8 +141,8 @@ static void autofs4_notify_daemon(struct
> > > >>   packet->ino = wq->ino;
> > > >>   packet->uid = wq->uid;
> > > >>   packet->gid = wq->gid;
> > > >> - packet->pid = wq->pid;
> > > >> - packet->tgid = wq->tgid;
> > > >> + packet->pid = pid_nr(wq->pid);
> > > >> + packet->tgid = pid_nr(wq->tgid);
> > > >>   break;
> > > >>
> > > >> I'm assuming we build the packet in the process context of the
> > > >> daemon we are sending it to. If not we have a problem here.
> > > >
> > > > Yes this is data being sent to a userspace daemon (Ian pls
> > > > correct me if I'm wrong) so the pid_nr is the only thing we can
> > > > send.
> > >
> > > Agreed. The question is are we in the user space daemon's process
> > > when we generate the pid_nr. Or do we stuff this in some kind of
> > > socket, and the socket switch locations of the packet.
> > >
> > > Basically I'm just trying to be certain we are calling pid_nr in the
> > > proper context. Otherwise we could get the wrong pid when we have
> > > multiple pid namespaces in play.
> >
> > We need to know what the userspace daemon being written to is doing
> > with autofs_ptype_{missing,expire}_{in,}direct() messages.
>
> At the moment autofs only uses the packet->pid for logging purposes.
> This solves an age old problem of not knowing who is causing mount
> requests.

probably I'm wrong, but that sounds like the packet->pid

is supposed to be the pid of the process _causing_ the mount,
not the user space daemon communicating with the kernel ...

> I'm not aware of any other applications that use version 5 yet but
> that of course could change. So we can't really know what will be done
> with these ids at some point in the future.
>
> > If I understand correctly, the pid being sent is of a process which
> > tried to automount some directory. The message is being sent to the
> > autofs daemon, which should be running in the root pid namespace.
>
> Yes, but it could be the autofs daemon itself in the expire case.
>
> Usually it doesn't make sense to run an automounting application as
> other than "root" but I'm not familiar with other possible userspace
> applications. Perhaps User Mode Linux could be an issue?
>
> >
> > So as it is, the pid_nr(wq->pid) should be done under the init
> > pid_namespace, since it's a kthread. So as long as the userspace
> > automount daemon is started in the root pid namespace, the pid it
> > gets will be the right one.
> >
> > Ian, does what I'm saying make sense, or am I wrong about how things
> > work for autofs?
>
> Yep. That's the way it is.

assumed we allow auto mounter mounts inside a context
(I see no immediate reason not to do that) we want to
know the name/pid space the userspace daemon is running
in as well as the name/pid space of the trigger task

> > thanks,
> > -serge
> >
> > PS
> > Note that if I'm right, but some machine starts autofs in a child
> > pid_namespace, the pid_nr() the way I have it is wrong. I'm not sure in
> > that case how we go about fixing that. Somehow we need to store the
> > autofs userspace daemon's pid namespace pointer to help us find the
> > proper pid_nr.
>
> In order for any user space application to use the module it must mount
> the autofs file system, passing a file handle for the pipe to use for
> communication. This must always be done. Can we grab the process pid
> namespace at that time and store it in the superblock info structure?

probably yes, but if my assumption above is correct, that
isn't necessarily the pid/space for the trigger process
(although it makes sense that it _should_ be)

best,
Herbert

> How does this affect getting ids for waitq request packets of other user
> space processes triggering mounts? I'm guessing that they would need to
> belong to the appropriate namespace for this mechanism to funtion
> correctly.
>
> Ian
>

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