Subject: Re: [PATCH 2/2] Replace pid_t in autofs with struct pid reference
Posted by serue on Mon, 19 Mar 2007 21:19:59 GMT
View Forum Message <> Reply to Message

Quoting Eric W. Biederman (ebiederm@xmission.com):
> "Serge E. Hallyn" <serue@us.ibm.com> writes:
>
> >>
> >> >> Index: 2.6.20/fs/autofs4/waitq.c
> >> >> ===
> >> >> --- 2.6.20.orig/fs/autofs4/waitq.c
> >> >> +++ 2.6.20/fs/autofs4/waitq.c
> >> >> @@ -292,8 +292,8 @@ int autofs4_wait(struct autofs_sb_info *
> >> >> wq->ino = autofs4_get_ino(sbi);
> >> >> wq->uid = current->uid;
> >> >> wq->gid = current->gid;
> >> >> - wq->pid = current->pid;
> >> >> - wq->tgid = current->tgid;
> >> >> + wq->pid = pid_nr(task_pid(current));
> >> >> + wq->tgid = pid_nr(task_tgid(current));
> >> >> wq->status = -EINTR; /* Status return if interrupted */
> >> >> atomic_set(&wq->wait_ctr, 2);
> >> >> mutex_unlock(&sbi->wq_mutex);
> >>
> >> I have a concern with this bit as I my quick review said the wait queue
> >> persists, and if so we should be cache the struct pid pointer, not the
> >> pid_t value. Heck the whol pid_nr(task_xxx(current)) idiom I find very
> >> suspicious.
> >
> > Based just on what I see right here I agree it seems like we would want
> > to store a ref to the pid, not store the pid_nr(pid) output, so in this
> > context it is suspicious.
>
> So that far we are in agreement.
>
> > OTOH if you're saying that using pid_nr(task_pid(current)) anywhere
> > should always be 'wrong', then please explain why, as I think we have a
> > disagreement on the meanings of the structs involved. In other words,
> > at some point I expect the only way to get a "pid number" out of a task
> > would be using this exact idiom, "pid_nr(task_pid(current))".
>
> Dealing with the current process is very common, and
> "pid_nr(task_pid(current)" is very long winded. Therefore I think it
> makes sense to have a specialized helper for that case.
>
> I don't think "current->pid" and "current->tgid" are necessarily
> wrong.

True, current->pid can probably always be legitimately taken as the pid
number in the current task's cloning namespace. But task->pid is wrong.
So if as you say it's worth caching (not saying I doubt you, just that I
haven't verified), then ideally we could cache current->pid but only
access it using current_pid(). Does that seem worth doing?

In any case, certainly adding a task_pid_nr() helper which for starters
returns pid_nr(task_pid(task)) seems reasonable. Note that Suka's about
ready to send a new iteration of the pidns patchset, so I'd like this to
be considered something to clean up on top of that patchset.

-serge

> For "process_session(current)", and "process_group(current)" I think
> they are fine but we might optimize them to something like:
> "current_session()" and "current_group()".
>
> The important part is that we have clearly detectable idioms for
> finding the pid values. So we can find the users and audit the code.
> Having a little more change so that the problem cases don't compile
> when they comes from a patch that hasn't caught up yet with the changes
> is also useful.
>
> The only advantage I see in making everything go through something
> like: pid_nr(task_pid(current)) is that we don't have the problem of
> storing the pid value twice. However if we have short hand helper
> functions for that case it will still work and we won't be horribly
> wordy.
>
> Further I don't know how expensive pid_nr is going to be, I don't
> think it will be very expensive. But I still think it may be
> reasonable to cache the answers for the current process on the
> task_struct. Fewer cache lines and all of that jazz.
>
> Mostly I just think pid_nr(task_pid(xxx)) looks ugly is rarely needed
> and is frequently associated with a bad conversion.
>
> Eric

_____