Subject: Re: [RFC][PATCH 1/6] Add struct pid_nr
Posted by serue on Sun, 11 Mar 2007 20:38:30 GMT
View Forum Message <> Reply to Message

Quoting Eric W. Biederman (ebiederm@xmission.com):
> "Serge E. Hallyn" <serue@us.ibm.com> writes:
>
> >> > +int attach_pid_nr(struct pid *pid, struct pid_nr *pid_nr)
> >> > +{
> >> > + spin_lock(&pid->lock);
> >> > + hlist_add_head_rcu(&pid_nr->node, &pid->pid_nrs);
> >> > + spin_unlock(&pid->lock);
> >>
> >> struct pid doesn't have a lock member.
> >>
> >> We should be able to add everything to struct pid at allocation time,
> >> so we should not need a lock.
> >>
> >> If you made struct pid_nr what the hash table entry it would probably
> >> make more sense, and gave it a struct pid pointer it would probably
> >> make more sense.
> >
> > i guess this is one reason to not support unshare for pidns. if we only
> > support clone thenj we don't need the lock.
> >
> > actually that's not true. whenever we start to allow clone pidns
> > without doing setsid first, we'll need to pull the existing processes
> > which are session and prgp leaders into the new pidns, so we will need
> > to add pidnr's to their struct pid.
>
> A better way to put this is that we already have a lock that attach_pid
> and detach_pid use. So we don't need another one for what should be
> a very rare case.

I think we'd at least need rcu if we supported unshare.

> I don't think we will need to add pids in the new pid namespace for
> sid and pgrp leaders into the new pid namespace.
>
> If we do need to pull in sid and pgrp leaders calling setsid() before
> the operation won't help (wrong namespace)

I'm pretty sure sid and pgrp leaders are a single task_struct pointer
each, so setting these to point to yourself before an unshare works.
But I guess for clone it does not work!

I agree with what you say in a later email - just returning '0' if sid or
pgrp is not in our pid_ns presents no problems, and not having a pgrp on

which to do kill -pgrp doesn't matter either...

So how about we

1. remove the setsid requirement before CLONE_NEWPID
2. punt on unshare(CLONE_NEWPID) support for now
3. remove pid->lock saving some space and time for now

> and the problem is the same
> for unshare and clone.

Disagree, but irrelevant if we do the above for now.

-serge

_____
Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers