Subject: Re: Namespaces exhausted CLONE_XXX bits problem
Posted by serue on Tue, 15 Jan 2008 14:35:24 GMT
View Forum Message <> Reply to Message

Quoting Cedric Le Goater (clg@fr.ibm.com):
> Serge E. Hallyn wrote:
> > Quoting Cedric Le Goater (clg@fr.ibm.com):
> >> to be more precise :
> >>
> >> long sys_clone_something(struct clone_something_args args)
> >>
> >> and
> >>
> >> long sys_unshare_something(struct unshare_something_args args)
> >>
> >> The arg passing will be slower bc of the copy_from_user() but we will
> >> still have the sys_clone syscall for the fast path.
> >>
> >> C.
> >
> > I'm fine with the direction you're going, but just as one more option,
> > we could follow more of the selinux/lsm approach of first requesting
> > clone/unshare options, then doing the actual clone/unshare. So
> > something like
> >
> > sys_clone_request(extended_64bit_clone_flags)
> > sys_clone(usual args)
> >
> > or
> >
> > echo pid,mqueue,user,ipc,uts,net > /proc/self/clone_unshare
> > clone()
>
> For my information, why selinux/lsm chose that 2 steps approach ?
> What kind of issues are they trying to solve ?

Well an interface was needed to allow multiple LSMs to query and set
task information. Using a syscall (which was attempted) required
ioctl-like subcommands which was not accepted.

-serge

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