Subject: Re: [patch -mm 08/17] nsproxy: add hashtable
Posted by serue on Tue, 12 Dec 2006 15:29:12 GMT
View Forum Message <> Reply to Message

Quoting Eric W. Biederman (ebiederm@xmission.com):
> "Serge E. Hallyn" <serue@us.ibm.com> writes:
>
> > Quoting Eric W. Biederman (ebiederm@xmission.com):
> >> I actually have code that will let me fork a process in a new namespace today
> >> with out needing bind_ns. What is more I don't even have to be root
> >> to use it.
> >
> > Can you elaborate? The user namespace patches don't enforce ptrace
> > yet, so you could unshare as root, become uid 500, then as uid 500
> > in the original namespace ptrace the process in the new namespace.
> > Is that what you're doing? If (when) ptrace enforces the uid namespace,
> > will that stop what you're doing?
>
> sys_ptrace is allowed in 2 situations.
> - The user and group identities are the same.
> - The calling process has CAP_SYS_PTRACE capability.
>
> So currently if the uid namespace enforces the user and group checks
> that will prevent the first case, and is very desirable. But it won't
> stop someone with CAP_SYS_PTRACE. Which given the normal case seems
> reasonable.

Yes, I was forgetting that intra-container ptrace is generally
inhibited by lack of a handle to processes in the other container.
So:

 . in checkpoint/restart usage, the normal CAP_SYS_PTRACE
   semantics is fine
 . inside a vserver, the normal CAP_SYS_PTRACE is fine
 . in general, a process inside one vserver cannot reference
   a process in another vserver, so we don't need to worry
   about ptrace permissions at all
 . however, if we want to (as per emails yesterday) provide
   some bit of enforcement of limits from parent namespaces
   to child namespaces - where a pid is in fact available for
   at least the init process (and, depending on our final
   implementation, perhaps all processes) - then we need
   something more.

As you say, selinux permissions would be one way to obtain this.

> Getting to the point where you can't trace what a process is doing
> would probably require some additional interprocess firewalling

> from something like selinux.

Yup.

thanks,
-serge

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