Subject: Re: [RFC] [PATCH 2/2] namespace enter: introduce sys_hijack (v3)
Posted by serue on Tue, 04 Sep 2007 19:32:56 GMT
View Forum Message <> Reply to Message

Quoting Dave Hansen (haveblue@us.ibm.com):
> On Tue, 2007-09-04 at 07:50 -0500, Serge E. Hallyn wrote:
> > > What do you do if there are no processes in a particular container?
> > >
> > The nsproxy will have been released so you couldn't enter it anyway.
> >
> Yeah, we'd need some kind of other object to keep the nsproxy around and
> hold a reference to it.

We could of course have the ns_container subsystem do that. The ns_container generally stick around until the admin does a manual rm on its directory, so this way we could keep the nsproxy around.

> But, it also begs other questions about how we define the namespace
> boundaries vs. containers. What if we have a normal container with
> chroot'd process inside of it? Two such processes will not share an
> nsproxy because the chroot'd one has switched filesystem namespaces.

But then a chroot isn't really anything to do with a namespace. An equivalent would be clone(CLONE_NEWNS)+pivot_root(new_root,put_old). And that would cause a new namespace for the child. Which is why I think we need to be able to define a container as a set of nsproxies, either by introducing CLONE_NEWCONTAINER or using the ns_container subsystem.

Then a namespace enter would always be done into the namespace init process' nsproxy.

> Who is to say that the "container" is represented by one process's
> nsproxy more than another?

The admin who defined the container I guess :)

One major ugliness is that the definition of namespace boundaries is different with each ns.

Uts namespaces are completely distinct and creates as copies of the original.

Mounts namespaces are by default isolated, and created as copies of the original. But sharing can be done using mounts propagation.

Pid namespaces are hierarchical, with processes being visible in all ancestor namespaces.

IPC namespaces are isolated and created empty.

Network namespaces will be isolated and created empty, with network devices being shared between a parent and child namespace?

In the face of that straight namespace entering is the simplest way to administer a 'container'. Without namespace entering, every namespace may need to be administrated somewhat differently, if it is even possible. (i.e. utsname0

-serge

_____

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