Subject: Global namespace naming, and monitoring.
Posted by ebiederm on Mon, 07 Jul 2008 11:54:26 GMT
View Forum Message <> Reply to Message

Daniel Lezcano <dlezcano@fr.ibm.com> writes:

>> Well, if we want to get the stats for each namespace separately, then
>> this ability is already present. Since this statistics is shown via the
>> /proc/net files and the /proc/net itself is seen via the /proc/<pid>/net,
>> then we can walk the init-s of all the containers in the system and dump
>> this info.
>>
>> The problem that is to be solved with this approach is how to get these
>> init-s :) But since finding any namespace by some task living in it is a
>> common practice now (netdev moving, sys_hijack) this one will be solved.

Yes. Finding the which the a single process in each namespace to look
at (the init-s) is something we have yet to refine.

The model for multiple namespace monitoring is definitely having
filesystems mounted that we can look at to get all of the information
we care about. /proc does a lot of this today, and with some cleanups
it should be able to display per namespace sysctls and a few other
goodies.

...

There is one class of user that we have yet to find a good solution for.
The people who want to use isolated network stacks within a single application.
Usually because there are duplicate routes between the stacks. In that case
indirect through processes falls down, as does being able to create a socket in
one namespace.

My latest brainstorm comes from asking how the problem would have been
solved in plan9. The idea is to create a filesystem we can mount that
holds a reference to a netns (netnsfs). Using a mounted filesystem
like that is a bit heavy weight, but just referring to it's the root
directory is enough to give us a name in the mount namespace. The
auto unmount and consequent release of the network namespace when the
mount namespace goes away is attractive.

> Shouldn't be interesting to handle the network namespaces directly with the
> iproute command ?
>
> * ip netns add <name>
> * ip netns del <name>
> * ip netns <oldname> set name <newname>
> * ip netns show

Ugh.  You have to be really careful with proposal like this to ensure that
they wind up in some namespace.  Otherwise you have created a new global
namespace, and have made nesting of containers much harder to implement.

> So having the netns binded, we can plug the known subcommand (link, ip, ...)
> with the netns features. For examples:
>
>  * ip addr add 1.2.3.4/24 dev eth0 netns foo

The only thing limiting that today is that we need someway to get a netlink
socket on the namespace in question.  Get me in a perverse mood and I will
grab the netlink socket from one of the init-s with ptrace.

Eric

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