Subject: Re: [PATCH 5/7]: Determine pts_ns from a pty's inode. Posted by serue on Wed, 26 Mar 2008 02:50:38 GMT View Forum Message <> Reply to Message Quoting sukadev@us.ibm.com (sukadev@us.ibm.com): > Serge E. Hallyn [serue@us.ibm.com] wrote: > | Quoting Serge E. Hallyn (serue@us.ibm.com): > | > Quoting sukadev@us.ibm.com (sukadev@us.ibm.com): > | > > > | > > From: Sukadev Bhattiprolu < sukadev@us.ibm.com> > | > Subject: [PATCH 5/7]: Determine pts_ns from a pty's inode. > | > > The devpts interfaces currently operate on a specific pts namespace > | > > which they get from the 'current' task. > | > > With implementation of containers and cloning of PTS namespaces, we want > | > > to be able to access PTYs in a child-pts-ns from a parent-pts-ns. For > | > > instance we could bind-mount and pivot-root the child container on > | > > '/vserver/vserver1' and then access the "pts/0" of 'vserver1' using > | > \$ echo foo > /vserver/vserver1/dev/pts/0 > | > > > | > > The task doing the above 'echo' could be in parent-pts-ns. So we find > | > > the 'pts-ns' of the above file from the inode representing the above > | > > file rather than from the 'current' task. > | > > > | > Note that we need to find and hold a reference to the pts_ns to prevent > | > > the pts ns from being freed while it is being accessed from 'outside'. > | > > > | > This patch implements, 'pts_ns_from_inode()' which returns the pts_ns > | > using 'inode->i_sb->s_fs_info'. > | > > > | > Since, the 'inode' information is not visible inside devpts code itself, > | > > this patch modifies the tty driver code to determine the pts_ns and passes > | > > it into devpts. > | > > > | > > TODO: > | >> What is the expected behavior when '/dev/tty' or '/dev/ptmx' are > | > > accessed from parent-pts-ns. i.e: > | > > > | > \$ echo "foobar" > /vserver/vserver1/dev/tty) > | > This patch currently ignores the '/vserver/vserver1' part (that > | > > | > The way this is phrased it almost sounds like you're considering using > | > the pathnames to figure out the ptsns to use :).

> | > It's not clear to me what is the sane thing to do.

> | >

	> what you're doing here - have /dev/ptmx and /dev/tty always use > current->'s ptsns - isn't ideal.
>	•
	> It would be nicer to not have a 'devpts ns', and instead have a
	> full device namespace. However, then it still isn't clear how to tie
	/vs/vs1/dev/ptmx to vs1's device namespace, since there is no devicefs to which to tie the devns.
>	
	> We could tie the devns to a device inode on mknod, using the devns of
	> the creating task. Then when starting up vs1, you just have to always
	> let vs1 create /dev/ptmx and /dev/tty. I can't think of anything
>	> better offhand.
>	
- 1	> Other ideas?
>	Lauppage you could just greate /dev/sta/stmy and /dev/sta/thy
	I suppose you could just create /dev/pts/ptmx and /dev/pts/tty. Recommend that in containers /dev/ptmx and /dev/tty be symlinks
	into /dev/pts. Applications don't need to change. If
	ptmx_open() sees that inode->i_sb is a devptsfs, it gets the
	namespace from the sb. If not, then it was a device in /dev
	and it gets the nmespace from current.
>	
	But we would still depend on user-space remounting /dev/pts after
	the clone right? Until they do that we would access the parent
> (container's /dev/pts/ptmx ?
Υe	es. Which is the right thing to do imo.
-serge	
Containers mailing list Containers@lists.linux-foundation.org	
https://lists.linux-foundation.org/mailman/listinfo/containers	
110	ps.//lists.iiiidx_fouridation.org/maiiman/listimo/containers