Subject: [RFC] [PATCH 0/4] uid_ns: introduction
Posted by serue on Tue, 07 Nov 2006 04:18:14 GMT
View Forum Message <> Reply to Message

Cedric has previously sent out a patchset
(http://lists.osdl.org/pipermail/containers/2006-August/000078.html)
impplementing the very basics of a user namespace. It ignores filesystem
access checks, so that uid 502 in one namespace could access files
belonging to uid 502 in another namespace, if the containers were so set
up.

This isn't necessarily bad, since proper container setup should prevent
problems. However there has been concern, so here is a patchset which
takes one course in addressing the concern.

It adds a user namespace pointer to every superblock, and to enhances
fsuid equivalence checks with a (inode->i_sb->s_uid_ns ==
current->nsproxy->uid_ns) comparison.

I've tested this as follows:

Created a bare-minimum loopback filesystem which has su, ps, touch, and sh
and requisites (like /etc/pam.d). Under that, created a user hallyn with
the same uid as user hallyn on the root filesystem. Under both /home/hallyn
and /mnt/0/home/hallyn (/home/hallyn on the loopbackfs) created a directory
'priv' with 0700 perms.

unsharens -U /bin/sh
su hallyn
ls /home/hallyn/priv
(permission denied)
mount -o loop /usr/src/disk.img /mnt/0
mount -t proc none /mnt/0/proc
mount -t devpts none /mnt/0/dev/pts
chroot /mnt/0
su hallyn
ls /home/hallyn/priv
ab

And, finally, of course

mount -o loop /usr/src/disk.img /mnt/0
mount -t proc none /mnt/0/proc
mount -t devpts none /mnt/0/dev/pts
unsharens -U /bin/sh
chroot /mnt/0
su hallyn
ls /home/hallyn/priv

(permission denied)

This is only a rough prototype to start some discussion. i.e. I ignore groups, so kernel/sys.c:in_group_p() for instance will need to be updated.

A few issues to be discussed:

1. I am not doing anything about root access. There are several ways we can address this.

a. implement CAP_NS_OVERRIDE, without which cross-ns access is not allowed
b. just don't allow any cross-ns access at all
c. a more complicated scheme where root process in parent and child namespaces can access each other until somehow the parent-ns cuts off the child's access.

2. This patch takes the easy route of adding user_ns pointers to the superblock. It would be very nice to add it to the vfsmount instead, so that admins could simply mount --bind into various namespaces, rather than having to use completely separate filesystems. However several fsuid equivalence checks happen with only an inode available. The hardest to address so far appear to be fs/namei.c:generic_permission as called from, say nfs, fs/generic_acl.c:generic_acl_set, and fs/attr.c:inode_change_ok called from jffs2.

Still, putting the user_ns in the superblock and forcing the use of separate filesystems (i.e. through a lightweight stackable read-only filesystem) isn't *so* bad, is it?

thanks,
-serge

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