

---

Subject: [patch 0/8] mount ownership and unprivileged mount syscall (v4)  
Posted by [Miklos Szeredi](#) on Fri, 20 Apr 2007 10:25:32 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

This patchset has now been bared to the "lowest common denominator" that everybody can agree on. Or at least there weren't any objections to this proposal.

Andrew, please consider it for -mm.

Thanks,  
Miklos

----

v3 -> v4:

- simplify interface as much as possible, now only a single option ("user=UID") is used to control everything
- no longer allow/deny mounting based on file/directory permissions, that approach does not always make sense

----

This patchset adds support for keeping mount ownership information in the kernel, and allow unprivileged mount(2) and umount(2) in certain cases.

The mount owner has the following privileges:

- unmount the owned mount
- create a submount under the owned mount

The sysadmin can set the owner explicitly on mount and remount. When an unprivileged user creates a mount, then the owner is automatically set to the user.

The following use cases are envisioned:

- 1) Private namespace, with selected mounts owned by user.  
E.g. /home/\$USER is a good candidate for allowing unpriv mounts and unmounts within.
- 2) Private namespace, with all mounts owned by user and having the "nosuid" flag. User can mount and umount anywhere within the namespace, but suid programs will not work.
- 3) Global namespace, with a designated directory, which is a mount owned by the user. E.g. /mnt/users/\$USER is set up so that it is bind mounted onto itself, and set to be owned by \$USER. The user

can add/remove mounts only under this directory.

The following extra security measures are taken for unprivileged mounts:

- usermounts are limited by a sysctl tunable
- force "nosuid,nodev" mount options on the created mount

--

---

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

---