

---

Subject: [PATCH 24/25] honor r/w changes at do\_remount() time

Posted by [Dave Hansen](#) on Mon, 11 Dec 2006 22:30:19 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Originally from: Herbert Poetzl <herbert@13thfloor.at>

This is the core of the read-only bind mount patch set.

Note that this does not add a "ro" option directly to the bind mount operation. If you require such a mount, you must first do the bind, then follow it up with a 'mount -o remount,ro' operation.

Signed-off-by: Dave Hansen <[haveblue@us.ibm.com](mailto:haveblue@us.ibm.com)>

---

lxc-dave/fs/namespace.c | 24 ++++++-----

lxc-dave/fs/open.c | 2 +-

2 files changed, 23 insertions(+), 3 deletions(-)

diff -puN fs/namespace.c~23-24-honor-r-w-changes-at-do-remount-time fs/namespace.c

--- lxc/fs/namespace.c~23-24-honor-r-w-changes-at-do-remount-time 2006-12-11

14:22:12.000000000 -0800

+++ lxc-dave/fs/namespace.c 2006-12-11 14:22:12.000000000 -0800

@@ -399,7 +399,7 @@ static int show\_vfsmnt(struct seq\_file \*

seq\_path(m, mnt, mnt->mnt\_root, " \t\n\\");

seq\_putc(m, '');

mangle(m, mnt->mnt\_sb->s\_type->name);

- seq\_puts(m, mnt->mnt\_sb->s\_flags & MS\_RDONLY ? " ro" : " rw");

+ seq\_puts(m, \_\_mnt\_is\_READONLY(mnt) ? " ro" : " rw");

for (fs\_infop = fs\_info; fs\_infop->flag; fs\_infop++) {

if (mnt->mnt\_sb->s\_flags & fs\_infop->flag)

seq\_puts(m, fs\_infop->str);

@@ -1023,6 +1023,23 @@ out:

}

EXPORT\_SYMBOL\_GPL(mnt\_want\_write);

+static int change\_mount\_flags(struct vfsmount \*mnt, int ms\_flags)

+

+ int error = 0;

+ int readonly\_request = 0;

+

+ if (ms\_flags & MS\_RDONLY)

+ readonly\_request = 1;

+ if (readonly\_request == \_\_mnt\_is\_READONLY(mnt))

+ return 0;

+

+ if (readonly\_request)

```

+ error = mnt_make_readonly(mnt);
+ else
+ __mnt_unmake_readonly(mnt);
+ return error;
+}
+
/*
 * change filesystem flags. dir should be a physical root of filesystem.
 * If you've mounted a non-root directory somewhere and want to do remount
@@ -1044,7 +1061,10 @@ static int do_remount(struct nameidata *
    return -EINVAL;

down_write(&sb->s_umount);
- err = do_remount_sb(sb, flags, data, 0);
+ if (flags & MS_BIND)
+ err = change_mount_flags(nd->mnt, flags);
+ else
+ err = do_remount_sb(sb, flags, data, 0);
if (!(sb->s_flags & MS_RDONLY))
    mnt_flags |= MNT_SB_WRITABLE;
if (!err)
diff -puN fs/open.c~23-24-honor-r-w-changes-at-do-remount-time fs/open.c
--- lxc/fs/open.c~23-24-honor-r-w-changes-at-do-remount-time 2006-12-11 14:22:12.000000000
-0800
+++ lxc-dave/fs/open.c 2006-12-11 14:22:12.000000000 -0800
@@ -401,7 +401,7 @@ asmlinkage long sys_faccessat(int dfd, c
    special_file(nd.dentry->d_inode->i_mode))
    goto out_path_release;

- if(IS_RDONLY(nd.dentry->d_inode))
+ if(__mnt_is_READONLY(nd.mnt) || IS_RDONLY(nd.dentry->d_inode))
    res = -EROFS;

out_path_release:
-
```

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