

---

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>  
---

```
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_infolp = fs_info; fs_infolp->flag; fs_infolp++) {
        if (mnt->mnt_sb->s_flags & fs_infolp->flag)
            seq_puts(m, fs_infolp->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>

---