
Subject: [patch 09/22] elevate writer count for chown and friends
Posted by [Cedric Le Goater](#) on Thu, 07 Jun 2007 15:25:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

From: Dave Hansen <hansendc@us.ibm.com>

chown/chmod,etc... don't call permission in the same way
that the normal "open for write" calls do. They still
write to the filesystem, so bump the write count during
these operations.

Signed-off-by: Dave Hansen <hansendc@us.ibm.com>

fs/open.c | 41 ++++++-----
1 file changed, 31 insertions(+), 10 deletions(-)

Index: 2.6.22-rc4-mm2-robindmount/fs/open.c

=====

--- 2.6.22-rc4-mm2-robindmount.orig/fs/open.c

+++ 2.6.22-rc4-mm2-robindmount/fs/open.c

@@ -596,12 +596,12 @@ asmlinkage long sys_fchmod(unsigned int

audit_inode(NULL, inode);

- err = -EROFS;

- if (IS_RDONLY(inode))

+ err = mnt_want_write(file->f_vfsmnt);

+ if (err)

goto out_putf;

err = -EPERM;

if (IS_IMMUTABLE(inode) || IS_APPEND(inode))

- goto out_putf;

+ goto out_drop_write;

mutex_lock(&inode->i_mutex);

if (mode == (mode_t) -1)

mode = inode->i_mode;

@@ -610,6 +610,8 @@ asmlinkage long sys_fchmod(unsigned int

err = notify_change(dentry, &newattrs);

mutex_unlock(&inode->i_mutex);

+out_drop_write:

+ mnt_drop_write(file->f_vfsmnt);

out_putf:

fput(file);

out:

@@ -629,13 +631,13 @@ asmlinkage long sys_fchmodat(int dfd, co

```

    goto out;
    inode = nd.dentry->d_inode;

- error = -EROFS;
- if (IS_RDONLY(inode))
+ error = mnt_want_write(nd.mnt);
+ if (error)
    goto dput_and_out;

    error = -EPERM;
    if (IS_IMMUTABLE(inode) || IS_APPEND(inode))
- goto dput_and_out;
+ goto out_drop_write;

    mutex_lock(&inode->i_mutex);
    if (mode == (mode_t) -1)
@@ -645,6 +647,8 @@ asmlinkage long sys_fchmodat(int dfd, co
    error = notify_change(nd.dentry, &newattrs);
    mutex_unlock(&inode->i_mutex);

+out_drop_write:
+ mnt_drop_write(nd.mnt);
dput_and_out:
    path_release(&nd);
out:
@@ -667,10 +671,7 @@ static int chown_common(struct dentry *
    printk(KERN_ERR "chown_common: NULL inode\n");
    goto out;
}
- error = -EROFS;
- if (IS_RDONLY(inode))
- goto out;
- error = -EPERM;
+ error = -EPERM;
    if (IS_IMMUTABLE(inode) || IS_APPEND(inode))
        goto out;
    newattrs.ia_valid = ATTR_CTIME;
@@ -699,7 +700,12 @@ asmlinkage long sys_chown(const char __u
    error = user_path_walk(filename, &nd);
    if (error)
        goto out;
+ error = mnt_want_write(nd.mnt);
+ if (error)
+ goto out_release;
    error = chown_common(nd.dentry, user, group);
+ mnt_drop_write(nd.mnt);
+out_release:
    path_release(&nd);

```

```

out:
    return error;
@@ -719,7 +725,12 @@ asmlinkage long sys_fchownat(int dfd, co
    error = __user_walk_fd(dfd, filename, follow, &nd);
    if (error)
        goto out;
+ error = mnt_want_write(nd.mnt);
+ if (error)
+ goto out_release;
    error = chown_common(nd.dentry, user, group);
+ mnt_drop_write(nd.mnt);
+out_release:
    path_release(&nd);
out:
    return error;
@@ -733,7 +744,12 @@ asmlinkage long sys_lchown(const char __
    error = user_path_walk_link(filename, &nd);
    if (error)
        goto out;
+ error = mnt_want_write(nd.mnt);
+ if (error)
+ goto out_release;
    error = chown_common(nd.dentry, user, group);
+ mnt_drop_write(nd.mnt);
+out_release:
    path_release(&nd);
out:
    return error;
@@ -750,9 +766,14 @@ asmlinkage long sys_fchown(unsigned int
    if (!file)
        goto out;

+ error = mnt_want_write(file->f_vfsmnt);
+ if (error)
+ goto out_fput;
    dentry = file->f_path.dentry;
    audit_inode(NULL, dentry->d_inode);
    error = chown_common(dentry, user, group);
+ mnt_drop_write(file->f_vfsmnt);
+out_fput:
    fput(file);
out:
    return error;

--

```

Containers mailing list
Containers@lists.linux-foundation.org

