
Subject: [PATCH 02/11] sysfs: sysfs_get_dentry add a sb parameter

Posted by [Benjamin Thery](#) on Wed, 18 Jun 2008 17:07:49 GMT

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

sysfs: sysfs_get_dentry add a sb parameter

In preparation for multiple mounts of sysfs add a superblock parameter to sysfs_get_dentry.

Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>

Signed-off-by: Benjamin Thery <benjamin.thery@bull.net>

fs/sysfs/dir.c | 12 ++++++-----

fs/sysfs/file.c | 2 +-

fs/sysfs/sysfs.h | 3 ++-

3 files changed, 10 insertions(+), 7 deletions(-)

Index: linux-mm/fs/sysfs/dir.c

=====

--- linux-mm.orig/fs/sysfs/dir.c

+++ linux-mm/fs/sysfs/dir.c

@@ -85,6 +85,7 @@ static void sysfs_unlink_sibling(struct

/**

* sysfs_get_dentry - get dentry for the given sysfs_dirent

+ * @sb: superblock of the dentry to return

* @sd: sysfs_dirent of interest

*

* Get dentry for @sd. Dentry is looked up if currently not

@@ -97,9 +98,10 @@ static void sysfs_unlink_sibling(struct

* RETURNS:

* Pointer to found dentry on success, ERR_PTR() value on error.

*/

-struct dentry *sysfs_get_dentry(struct sysfs_dirent *sd)

+struct dentry *sysfs_get_dentry(struct super_block *sb,

+ struct sysfs_dirent *sd)

{

- struct dentry *dentry = dget(sysfs_sb->s_root);

+ struct dentry *dentry = dget(sb->s_root);

while (dentry->d_fsdata != sd) {

struct sysfs_dirent *cur;

@@ -777,7 +779,7 @@ int sysfs_rename_dir(struct kobject * ko

goto out; /* nothing to rename */

/* get the original dentry */

- old_dentry = sysfs_get_dentry(sd);

+ old_dentry = sysfs_get_dentry(sysfs_sb, sd);

```

if (IS_ERR(old_dentry)) {
    error = PTR_ERR(old_dentry);
    old_dentry = NULL;
@@ -845,7 +847,7 @@ int sysfs_move_dir(struct kobject *kobj,
    goto out; /* nothing to move */

```

```

/* get dentries */
- old_dentry = sysfs_get_dentry(sd);
+ old_dentry = sysfs_get_dentry(sysfs_sb, sd);
if (IS_ERR(old_dentry)) {
    error = PTR_ERR(old_dentry);
    old_dentry = NULL;
@@ -853,7 +855,7 @@ int sysfs_move_dir(struct kobject *kobj,
}
old_parent = old_dentry->d_parent;

```

```

- new_parent = sysfs_get_dentry(new_parent_sd);
+ new_parent = sysfs_get_dentry(sysfs_sb, new_parent_sd);
if (IS_ERR(new_parent)) {
    error = PTR_ERR(new_parent);
    new_parent = NULL;

```

Index: linux-mm/fs/sysfs/file.c

=====

--- linux-mm.orig/fs/sysfs/file.c

+++ linux-mm/fs/sysfs/file.c

```

@@ -584,7 +584,7 @@ int sysfs_chmod_file(struct kobject *kobj
    goto out;

```

```

mutex_lock(&sysfs_rename_mutex);
- victim = sysfs_get_dentry(victim_sd);
+ victim = sysfs_get_dentry(sysfs_sb, victim_sd);
mutex_unlock(&sysfs_rename_mutex);
if (IS_ERR(victim)) {
    rc = PTR_ERR(victim);

```

Index: linux-mm/fs/sysfs/sysfs.h

=====

--- linux-mm.orig/fs/sysfs/sysfs.h

+++ linux-mm/fs/sysfs/sysfs.h

```

@@ -113,7 +113,8 @@ extern spinlock_t sysfs_assoc_lock;
extern const struct file_operations sysfs_dir_operations;
extern const struct inode_operations sysfs_dir_inode_operations;

```

```

-struct dentry *sysfs_get_dentry(struct sysfs_dirent *sd);
+struct dentry *sysfs_get_dentry(struct super_block *sb,
+ struct sysfs_dirent *sd);
struct sysfs_dirent *sysfs_get_active_two(struct sysfs_dirent *sd);
void sysfs_put_active_two(struct sysfs_dirent *sd);
void sysfs_addrm_start(struct sysfs_addrm_cxt *acxt,

```

--

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