Subject: [PATCH 03/15] sysfs: sysfs_get_dentry add a sb parameter
Posted by ebiederm on Fri, 04 Jul 2008 01:08:02 GMT
View Forum Message <> Reply to Message

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>
Signed-off-by: Daniel Lezcano <dlezcano@fr.ibm.com>
Acked-by: Tejun Heo <tj@kernel.org>
---
 fs/sysfs/dir.c | 12 +++++++-----
 fs/sysfs/file.c | 2 +-
 fs/sysfs/sysfs.h | 3 ++-
 3 files changed, 10 insertions(+), 7 deletions(-)

diff --git a/fs/sysfs/dir.c b/fs/sysfs/dir.c
index 146b86a..69c40ed 100644
--- a/fs/sysfs/dir.c
+++ b/fs/sysfs/dir.c
@@ -85,6 +85,7 @@ static void sysfs_unlink_sibling(struct sysfs_dirent *sd)

 /**
  * 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 sysfs_dirent *sd)
  * 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 * kobj, const char *new_name)
   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;
@@ -841,7 +843,7 @@ int sysfs_move_dir(struct kobject *kobj, struct kobject *new_parent_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;
@@ -849,7 +851,7 @@ int sysfs_move_dir(struct kobject *kobj, struct kobject *new_parent_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;
diff --git a/fs/sysfs/file.c b/fs/sysfs/file.c
index 15ff679..cb5dd3f 100644
--- a/fs/sysfs/file.c
+++ b/fs/sysfs/file.c
@@ -585,7 +585,7 @@ int sysfs_chmod_file(struct kobject *kobj, struct attribute *attr, mode_t
mode)
   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);
diff --git a/fs/sysfs/sysfs.h b/fs/sysfs/sysfs.h
index 0fdc3de..b1bdc6e 100644
--- a/fs/sysfs/sysfs.h
+++ b/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,
--
1.5.3.rc6.17.g1911

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