
Subject: [PATCH 03/11] sysfs: Implement __sysfs_get_dentry
Posted by Benjamin Thery on Fri, 06 Jun 2008 15:47:24 GMT
[View Forum Message](#) <> [Reply to Message](#)

sysfs: Implement __sysfs_get_dentry

This function is similar but much simpler to sysfs_get_dentry
returns a sysfs dentry if one currently exists.

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

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

fs/sysfs/dir.c | 39 +++
1 file changed, 39 insertions(+)

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

=====

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

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

```
@@ -777,6 +777,45 @@ void sysfs_remove_dir(struct kobject * kobj,
    __sysfs_remove_dir(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. Only return a dentry if one currently
+ * exists.

+ *

+ * LOCKING:

+ * Kernel thread context (may sleep)

+ *

+ * RETURNS:

+ * Pointer to found dentry on success, NULL on failure.

+ */

```
+static struct dentry *__sysfs_get_dentry(struct super_block *sb,
```

```
+    struct sysfs_dirent *sd)
```

```
{
```

```
+ struct inode *inode;
```

```
+ struct dentry *dentry = NULL;
```

```
+
```

```
+ inode = ilookup5_nowait(sysfs_sb, sd->s_ino, sysfs_ilookup_test, sd);
```

```
+ if (inode && !(inode->i_state & I_NEW)) {
```

```
+ struct dentry *alias;
```

```
+ spin_lock(&dcache_lock);
```

```
+ list_for_each_entry(alias, &inode->i_dentry, d_alias) {
```

```
+ if (!IS_ROOT(alias) && d_unhashed(alias))
+   continue;
+ if (alias->d_sb != sb)
+   continue;
+ dentry = alias;
+ dget_locked(dentry);
+ break;
+ }
+ spin_unlock(&dcache_lock);
+ }
+ iput(inode);
+ return dentry;
+}
+
int sysfs_rename_dir(struct kobject * kobj, const char *new_name)
{
    struct sysfs_dirent *sd = kobj->sd;

--
```

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