
Subject: [PATCH 3/4] sysfs: Implement sysfs_delete_link and sysfs_rename_link
Posted by [ebiederm](#) on Thu, 19 Jul 2007 04:46:39 GMT

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

When removing a symlink sysfs_remove_link does not provide enough information to figure out which shadow directory the symlink falls in. So I need sysfs_delete_link which is passed the target of the symlink to delete.

Further half the time when we are removing a symlink the code is actually renaming the symlink but not doing so explicitly because we don't have a symlink rename method. So I have added sysfs_rename_link as well.

Both of these functions now have enough information to find a symlink in a shadow directory. The only restriction is that they must be called before the target kobject is renamed or deleted. If they are called later I loose track of which tag the target kobject was marked with and can no longer find the old symlink to remove it.

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

fs/sysfs/symlink.c | 31 ++++++

include/linux/sysfs.h | 18 ++++++

2 files changed, 49 insertions(+), 0 deletions(-)

diff --git a/fs/sysfs/symlink.c b/fs/sysfs/symlink.c

index e52d3ef..62b0aeb 100644

--- a/fs/sysfs/symlink.c

+++ b/fs/sysfs/symlink.c

@@ -126,6 +126,21 @@ addrm_finish:

```
/**
 * sysfs_delete_link - remove symlink in object's directory.
 * @kobj: object we're acting for.
 * @targ: object we're pointing to.
 * @name: name of the symlink to remove.
 *
 * Unlike sysfs_remove_link sysfs_delete_link has enough information
 * to successfully delete symlinks in shadow directories.
 */
+void sysfs_delete_link(struct kobject *kobj, struct kobject *targ,
+ const char *name)
+{
+ sysfs_hash_and_remove(targ, kobj->sd, name);
+}
+
```

```

+/**
 * sysfs_remove_link - remove symlink in object's directory.
 * @kobj: object we're acting for.
 * @name: name of the symlink to remove.
@@ -136,6 +151,22 @@ void sysfs_remove_link(struct kobject * kobj, const char * name)
 sysfs_hash_and_remove(kobj, kobj->sd, name);
}

+/**
+ * sysfs_rename_link - rename symlink in object's directory.
+ * @kobj: object we're acting for.
+ * @targ: object we're pointing to.
+ * @old: previous name of the symlink.
+ * @new: new name of the symlink.
+ *
+ * A helper function for the common rename symlink idiom.
+ */
+int sysfs_rename_link(struct kobject *kobj, struct kobject *targ,
+ const char *old, const char *new)
+{
+ sysfs_delete_link(kobj, targ, old);
+ return sysfs_create_link(kobj, targ, new);
+}
+
static int sysfs_get_target_path(struct sysfs_dirent * parent_sd,
 struct sysfs_dirent * target_sd, char *path)
{
diff --git a/include/linux/sysfs.h b/include/linux/sysfs.h
index 7cda047..e195cb5 100644
--- a/include/linux/sysfs.h
+++ b/include/linux/sysfs.h
@@ -127,6 +127,13 @@ sysfs_create_link(struct kobject * kobj, struct kobject * target, const char
 * n
extern void
sysfs_remove_link(struct kobject *, const char * name);

+extern int
+sysfs_rename_link(struct kobject *kobj, struct kobject *target,
+ const char *old_name, const char *new_name);
+
+extern void
+sysfs_delete_link(struct kobject *dir, struct kobject *targ, const char *name);
+
int __must_check sysfs_create_bin_file(struct kobject *kobj,
 struct bin_attribute *attr);
void sysfs_remove_bin_file(struct kobject *kobj, struct bin_attribute *attr);
@@ -202,6 +209,17 @@ static inline void sysfs_remove_link(struct kobject * k, const char *
name)

```

```
;
}
```

```
+static inline int
+sysfs_rename_link(struct kobject * k, struct kobject *t,
+  const char *old_name, const char * new_name)
+{
+ return 0;
+}
+
+static inline void
+sysfs_delete_link(struct kobject *k, struct kobject *t, const char *name)
+{
+}

static inline int sysfs_create_bin_file(struct kobject * k, struct bin_attribute * a)
{
--
1.5.1.1.181.g2de0
```

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