

---

Subject: [PATCH 06/15] Introduce sysfs\_sd\_setattr and fix sysfs\_chmod

Posted by [ebiederm](#) on Fri, 04 Jul 2008 01:11:40 GMT

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

---

Currently sysfs\_chmod calls sys\_setattr which in turn calls inode\_change\_ok which checks to see if it is ok for the current user space process to change the attributes. Since sysfs\_chmod\_file has only kernel mode clients denying them permission if user space is the problem is completely inappropriate.

Therefore factor out sysfs\_sd\_setattr which does not call inode\_change\_ok and modify sysfs\_chmod\_file to call it.

In addition setting victim\_sd->s\_mode explicitly in sysfs\_chmod\_file is redundant so remove that as well.

Thanks to Tejun Heo <[htejun@gmail.com](mailto:htejun@gmail.com)>, and Daniel Lezcano <[dlezcano@fr.ibm.com](mailto:dlezcano@fr.ibm.com)> for working on this and spotting this case.

Signed-off-by: Eric W. Biederman <[ebiederm@xmission.com](mailto:ebiederm@xmission.com)>

---

```
fs/sysfs/file.c | 5 +----
fs/sysfs/inode.c | 23 ++++++-----
fs/sysfs/sysfs.h | 1 +
3 files changed, 18 insertions(+), 11 deletions(-)
```

```
diff --git a/fs/sysfs/file.c b/fs/sysfs/file.c
```

```
index cb5dd3f..1304b3a 100644
```

```
--- a/fs/sysfs/file.c
```

```
+++ b/fs/sysfs/file.c
```

```
@@ -600,13 +600,10 @@ int sysfs_chmod_file(struct kobject *kobj, struct attribute *attr, mode_t mode)
```

```
    newattrs.ia_mode = (mode & S_IALLUGO) | (inode->i_mode & ~S_IALLUGO);
```

```
    newattrs.ia_valid = ATTR_MODE | ATTR_CTIME;
```

```
    newattrs.ia_ctime = current_fs_time(inode->i_sb);
```

```
- rc = sysfs_setattr(victim, &newattrs);
```

```
+ rc = sysfs_sd_setattr(victim_sd, inode, &newattrs);
```

```
    if (rc == 0) {
        fsnotify_change(victim, newattrs.ia_valid);
- mutex_lock(&sysfs_mutex);
- victim_sd->s_mode = newattrs.ia_mode;
- mutex_unlock(&sysfs_mutex);
    }
```

```
    mutex_unlock(&inode->i_mutex);
```

```
diff --git a/fs/sysfs/inode.c b/fs/sysfs/inode.c
```

index eb53c63..80f8fd4 100644

--- a/fs/sysfs/inode.c

+++ b/fs/sysfs/inode.c

```
@@ -42,10 +42,9 @@ int __init sysfs_inode_init(void)
    return bdi_init(&sysfs_backing_dev_info);
}
```

```
-int sysfs_setattr(struct dentry * dentry, struct iattr * iattr)
```

```
+int sysfs_sd_setattr(struct sysfs_dirent *sd, struct inode *inode,
+ struct iattr * iattr)
```

```
{
- struct inode * inode = dentry->d_inode;
- struct sysfs_dirent * sd = dentry->d_fsdata;
  struct iattr * sd_iattr;
  unsigned int ia_valid = iattr->ia_valid;
  int error;
```

```
@@ -55,10 +54,6 @@ int sysfs_setattr(struct dentry * dentry, struct iattr * iattr)
```

```
sd_iattr = sd->s_iattr;
```

```
- error = inode_change_ok(inode, iattr);
```

```
- if (error)
```

```
- return error;
```

```
-
```

```
ia_valid &= ~ATTR_SIZE; /* ignore size changes */
```

```
error = inode_setattr(inode, iattr);
```

```
@@ -104,6 +99,20 @@ int sysfs_setattr(struct dentry * dentry, struct iattr * iattr)
```

```
return error;
```

```
}
```

```
+int sysfs_setattr(struct dentry *dentry, struct iattr *iattr)
```

```
+{
```

```
+ struct inode * inode = dentry->d_inode;
```

```
+ struct sysfs_dirent * sd = dentry->d_fsdata;
```

```
+ int error;
```

```
+
```

```
+ error = inode_change_ok(inode, iattr);
```

```
+ if (error)
```

```
+ return error;
```

```
+
```

```
+ return sysfs_sd_setattr(sd, inode, iattr);
```

```
+}
```

```
+
```

```
+
```

```
static inline void set_default_inode_attr(struct inode * inode, mode_t mode)
```

```
{
```

```
inode->i_mode = mode;
```

```
diff --git a/fs/sysfs/sysfs.h b/fs/sysfs/sysfs.h
index b1bdc6e..5ee5d0a 100644
--- a/fs/sysfs/sysfs.h
+++ b/fs/sysfs/sysfs.h
@@ -154,6 +154,7 @@ static inline void sysfs_put(struct sysfs_dirent *sd)
 * inode.c
 */
struct inode *sysfs_get_inode(struct sysfs_dirent *sd);
+int sysfs_sd_setattr(struct sysfs_dirent *sd, struct inode *inode, struct iattr *iattr);
int sysfs_setattr(struct dentry *dentry, struct iattr *iattr);
int sysfs_hash_and_remove(struct sysfs_dirent *dir_sd, const char *name);
int sysfs_inode_init(void);
--
1.5.3.rc6.17.g1911
```

---

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

---