Subject: [patch 1/1] selinux: do not clear f_op when removing entries
Posted by Stephen Smalley on Wed, 21 Nov 2007 14:01:36 GMT
View Forum Message <> Reply to Message

On Tue, 2007-11-20 at 15:17 +0000, Christoph Hellwig wrote:
> On Tue, Nov 20, 2007 at 10:05:05AM -0500, Stephen Smalley wrote:
> > > Nice, getting rid of this is a very good step formwards. Unfortunately
> > > we have another copy of this junk in
> > > security/selinux/selinuxfs.c:sel_remove_entries() which would need the
> > > same treatment.
> >
> > Can't just be dropped completely for selinux - we need a way to drop
> > obsolete entries from the prior policy when we load a new policy.
> >
> > Is the only real problem here the clearing of f_op? If so, we can
> > likely remove that from sel_remove_entries() without harm, and fix the
> > checks for it to use something more reliable.
>
> f_op removal is the biggest issue. It can't really work and this is the
> last instance. But in general having some half-backed attempts at revoke
> is never a good idea.

Do not clear f_op when removing entries since it isn't safe to do.

Signed-off-by: Stephen Smalley <sds@tycho.nsa.gov>

---

 security/selinux/selinuxfs.c | 28 +---------------------------
 1 file changed, 1 insertion(+), 27 deletions(-)

diff --git a/security/selinux/selinuxfs.c b/security/selinux/selinuxfs.c
index f5f3e6d..ac6fe99 100644
--- a/security/selinux/selinuxfs.c
+++ b/security/selinux/selinuxfs.c
@@ -838,10 +838,6 @@ static ssize_t sel_read_bool(struct file *filep, char __user *buf,

  ret = -EFAULT;

- /* check to see if this file has been deleted */
- if (!filep->f_op)
- goto out;
-
  if (count > PAGE_SIZE) {
   ret = -EINVAL;
   goto out;
@@ -882,10 +878,6 @@ static ssize_t sel_write_bool(struct file *filep, const char __user *buf,
  if (length)

```
   goto out;

- /* check to see if this file has been deleted */
- if (!filep->f_op)
-  goto out;
-
  if (count >= PAGE_SIZE) {
   length = -ENOMEM;
   goto out;
@@ -940,10 +932,6 @@ static ssize_t sel_commit_bools_write(struct file *filep,
  if (length)
   goto out;

- /* check to see if this file has been deleted */
- if (!filep->f_op)
-  goto out;
-
  if (count >= PAGE_SIZE) {
   length = -ENOMEM;
   goto out;
@@ -982,11 +970,9 @@ static const struct file_operations sel_commit_bools_ops = {
  .write          = sel_commit_bools_write,
 };

-/* partial revoke() from fs/proc/generic.c proc_kill_inodes */
 static void sel_remove_entries(struct dentry *de)
 {
- struct list_head *p, *node;
- struct super_block *sb = de->d_sb;
+ struct list_head *node;

  spin_lock(&dcache_lock);
  node = de->d_subdirs.next;
@@ -1006,18 +992,6 @@ static void sel_remove_entries(struct dentry *de)
  }

  spin_unlock(&dcache_lock);
-
- file_list_lock();
- list_for_each(p, &sb->s_files) {
-  struct file * filp = list_entry(p, struct file, f_u.fu_list);
-  struct dentry * dentry = filp->f_path.dentry;
-
-  if (dentry->d_parent != de) {
-   continue;
-  }
-  filp->f_op = NULL;
- }
```

- file_list_unlock();
 }

 #define BOOL_DIR_NAME "booleans"

--
Stephen Smalley
National Security Agency