Subject: Re: [PATCH 14/25] sysfs: Don't use lookup_one_len_kern
Posted by ebiederm on Wed, 08 Aug 2007 15:26:48 GMT
View Forum Message <> Reply to Message

Tejun Heo <htejun@gmail.com> writes:

> On Tue, Aug 07, 2007 at 03:23:57PM -0600, Eric W. Biederman wrote:
>>
>> Upon inspection it appears that there is no looking of the
>> inode mutex in lookup_one_len_kern and we aren't calling
>> it with the inode mutex and that is wrong.
>>
>> So this patch rolls our own dcache insertion function that
>> does exactly what we need it to do. As it turns out this
>> is pretty trivial to do and it makes the code easier to
>> audit.
>>
>> Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>
>> ---
>> fs/sysfs/dir.c | 41 ++--
>> 1 files changed, 39 insertions(+), 2 deletions(-)
>>
>> diff --git a/fs/sysfs/dir.c b/fs/sysfs/dir.c
>> index a9bdb12..1d53c2a 100644
>> --- a/fs/sysfs/dir.c
>> +++ b/fs/sysfs/dir.c
>> @@ -765,6 +765,44 @@ static struct dentry *__sysfs_get_dentry(struct
> super_block *sb, struct sysfs_di
>> return dentry;
>> }
>>
>> +static struct dentry *sysfs_add_dentry(struct dentry *parent, struct
> sysfs_dirent *sd)
>> +{
>> + struct qstr name;
>> + struct dentry *dentry;
>> + struct inode *inode;
>> +
>> + mutex_lock(&parent->d_inode->i_mutex);
>> + mutex_lock(&sysfs_mutex);
>> + dentry = ERR_PTR(-EINVAL);
>> + if (parent->d_fsdata != sd->s_parent)
>> + goto out;
>> +
>> + name.name = sd->s_name;
>> + name.len = strlen(sd->s_name);
>> + dentry = d_hash_and_lookup(parent, &name);
>> + if (dentry)

```
>> +  goto out;
>> +
>> + dentry = d_alloc(parent, &name);
>> + if (!dentry) {
>> +  dentry = ERR_PTR(-ENOMEM);
>> +  goto out;
>> + }
>> +
>> + inode = sysfs_get_inode(sd);
>> + if (!inode) {
>> +  dput(dentry);
>> +  dentry = ERR_PTR(-ENOMEM);
>> +  goto out;
>> + }
>> + d_instantiate(dentry, inode);
>> + sysfs_attach_dentry(sd, dentry);
>> +out:
>> + mutex_unlock(&sysfs_mutex);
>> + mutex_unlock(&parent->d_inode->i_mutex);
>> + return dentry;
>> +}
>
> This is virtually identical to
>
>  mutex_lock(&parent_dentry->d_inode->i_mutex);
>  dentry = lookup_one_len_kern(cur->s_name, parent_dentry,
>          strlen(cur->s_name));
>  mutex_unlock(&parent_dentry->d_inode->i_mutex);
>
> right?  I don't think we need to duplicate the code here.  Or is it
> needed for later multi-sb thing?
```

Right. We can do that as well. In practice in working code
there is no real difference.

There is a little extra uniformity in rolling it ourselves, but
not enough to worry about either way.

In the review/debug etc cycle it just wound up being a lot easier
to roll the code myself.

By the time we get to lookup_one_len_kern it is almost impenetrable
code in namei.c where sysfs_add_dentry tends is easier to comprehend,
and to modify for debugging.

Eric

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