Subject: Re: [Q] missing ->d_delete() in shrink_dcache_for_umount()?
Posted by David Howells on Tue, 31 Oct 2006 15:06:57 GMT
View Forum Message <> Reply to Message

Vasily Averin <vvs@sw.ru> wrote:

> It looks like I've noticed yet one suspicious place in your patch. As far as I
> see you have removed dput(root) call in shrink_dcache_for_umount() function.
> However I would note that dput contains ->d_delete() call that is missing in
> your function:
>
> if (dentry->d_op && dentry->d_op->d_delete) {
> if (dentry->d_op->d_delete(dentry))
>   goto unhash_it;
>
> I'm not sure but it seems to me some (probably out-of-tree) filesystems can do
> something useful in this place.

Can they though? I'm not so sure. I've added Al to the To list to get his opinion.

It seems to me that d_op->d_delete() is asking a question of whether the dentry should be discarded immediately upon dput() reducing the usage count to 0. The documentation in vfs.txt isn't entirely clear on this point, but what it does say is that that op isn't allowed to sleep, so there's a limit as to what it can do.

Furthermore, d_op->d_delete() is probably the wrong hook to call. It returns an indication of whether the dentry should be retained or not, but we aren't interested in that at this point: we have to get rid of the dentry anyway, so the fs's opinion is irrelevant.

Finally, we do still call d_op->d_release() by way of d_free(), so it's not as if the fs doesn't get a chance to clean up the dentry.

David