

Re: [RFC PATCH] file as directory

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2007-05/msg10318.html>

- *From:* Miklos Szeredi <miklos@xxxxxxxxxx>
 - *Date:* Wed, 23 May 2007 11:03:08 +0200
-

On Wed, May 23, 2007 at 10:05:21AM +0200, Miklos Szeredi wrote:

Er... These mounts might not be propagated, but what about a bind over another instance of such file in master tree?

So your question is, which mount takes priority on the lookup? It probably should be the propagated real mount, rather than the dir-on-file one, shouldn't it?

There might be dragons in that area...

I think they should be the same superblock, same dentry. What would be the advantage of doing otherwise?

Then you are going to have interesting time with locking in `final mntput()`.

Final `mntput` of what?

When the last reference to your mount goes away.

I still don't get it where the superblock comes in. The locking is "interesting" in there, yes. And I haven't completely convinced myself it's right, let alone something that won't easily be screwed up in the future. So there's definitely room for thought there.

But how does it matter if two different paths have the same sb or a different sb mounted over them?

Re: [RFC PATCH] file as directory

BTW, what about having several links to the same file? You have `i_mutex` on the inode, so serialization of those is not a problem, but...

Sorry, I lost it...

Say `/foo/bar/a` is such a file.

```
cd /foo/bar
ln a b
```

now do lookups on `a/` and `b/`

What happens?

The same dentry is mounted over each one. The contents of the directory should only depend on the contents of the underlying inode. The path leading up to it is completely irrelevant.

Miklos

–

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to `majordomo@xxxxxxxxxxxxxxxxx`

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>