

---

Subject: Re: Problem: LTP linkat01 test fails on nfs directory (NFS v3)

Posted by [gblond](#) on Fri, 21 Sep 2007 14:39:35 GMT

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

---

On the Friday 21 September 2007 16:37 Trond Myklebust, wrote:

> On Fri, 2007-09-21 at 13:13 +0400, Vitaliy Gusev wrote:

> > Hello.

> >

> > Tested kernels: 2.6.18, 2.6.22, 2.6.23-rc2

> >

> > Steps to reproduce: Suppose that we have mounted some directory from nfs

> > v3 server with default options. Also we have the two directories in this

> > mountpoint and each directory has hard linked file. Try to open those

> > files and write to one and read from another. Data will not be equal.

> > (Testcase: attached hardlink\_test.c)

>

> Please retry after re-exporting the filesystem using the highly

> recommended "no\_subtree\_check" option. The default subtree\_check option

> is broken: it changes the filehandle of the file upon a cross-directory

> rename() of that file.

Ok, problem goes out. Default option depends on nfs-utils release.

> >

> > Please explain why nfs\_find\_actor() function compares file handles?

>

> 'cos this is the only way to know that two files are the same. fileid is

> not always supported by servers: it is an optional NFSv4 attribute, and

> From RFC3010 ( NFSv4 ) section 2.4.1 :

"For example, if paths /a/b/c and /a/d/c refer to the same file, the server SHOULD return the same filehandle for both path names traversals".

For NFSv4 problem should not be shown.

> on NFSv3, most non-posix filesystems will fake it using something like

> iunique().

>

> Comparing filehandles allows you to be certain that two files are the

> same if the filehandles are equal. If they are not equal, then that does

> not guarantee that the files are different, but then how else are you

> going to determine it?

>

--

Thanks,

Vitaliy Gusev

---