

Table of Contents

Linux AFS client file corruption.....	1
Peter's notes from meeting 13.09.2005.....	3
Jan's notes from meeting 13.09.2005.....	4
Invoking the test scripts.....	5
Hardware.....	6
Test Log.....	7

Linux AFS client file corruption

(from Rainer) A script "disk_stress" creates a file, copies that file n times to a number of directories and compares the results and the original file. If the files don't match then obviously something went wrong. After a "while" (usually after an out-of-quota error), files are erased and everything restarts. This is a vanilla stress test and runs in an endless loop. The file size is typically 2M but the script can operate on several sizes. The copying to the different directories is performed in (parallel) sub-processes, one per directory.

Now: on the several SMP i386es that I tested since over a year ago, on GB Ethernet, and after usually several days of running, one of the compares fails! The file is a text file therefore easily comparable with tdiff and usually it turns out that in the middle of the file a section of another place in the file (or from another file - it would be the easy part of the work to find out) overlays until the next 4096 boundary.

After how long: not deterministic. Perhaps the "hit" rate could even be increased by having still more entropy in the file, currently the test pattern repeats every 512 bytes.

A few more hints after more than a year of (occasional!) tests:

- I've run the test on a number of Solaris machines for weeks (e.g. throughout my summer holiday) and nothing odd happens.

hypothesis: Linux only.

- the correct test pattern usually restarts at a 4K boundary. This could be a side effect of the 512 pattern of course. But still:

hypothesis: problem with Linux VM system.

- Sometimes the file copied to is corrupted on the server, sometimes however an 'fs flush' issued immediately after the failing compare reveals a correct file.

hypothesis: the problem occurs on both the read and write caching.

- I **have** run this on Linux for weeks on the local (ext3 or xfs, don't remember) file systems: nothing odd happened.

hypothesis: this is an AFS-only problem.

A few things to discuss:

1. a neutral eye should look at the script: it would be disastrous to spend weeks or months of investigation to find a trivial bug in the test script.
2. should we set up a serious 'test farm' to increase the frequency of the occurrence of the problem first? On the client side this is probably easy but for more than 4-5 the server will become a limit. Setting up a test AFS cell with a series of servers is an option but means a day or so of startup work.
3. the test pattern should be refined to include the file name as to find out whether the overlying buffer/pages comes from a different file. This is obviously trivial but shouldn't be forgotten.
4. the test pattern itself should be revisited: increasing the entropy in the file might increase the frequency of observed corruptions.
5. somebody should understand/be prepared to investigate the Linux VM system and the way the AFS client interacts with it - at least while there are hints that the problem is around there. For you experts this is of course daily bread, I spent days already trying to find out when a page actually gets marked dirty and when/how/why it gets paged out and what locks you are/are not supposed to hold while doing something with it. But I might be of help trying to give an overview of how the AFS clients

handles writes and reads.

Peter's notes from meeting 13.09.2005

- Gigabit-level traffic
- multiple clients
- run for several days
- AFS disk cache was always used (how about memcache?)
- once a corruption was detected, it was observed multiple times
- corruption location: both on-server and on-client-only has been observed
- corrupted data seems to be part of the corrupted file from a different offset
- no public records of observation of the problem by someone else than CERN

Jan's notes from meeting 13.09.2005

- Server on (Solaris or Linux), Linux client will have cause the problem
- problem occurs both on 2.4 and 2.6 kernels (2.6 has other(?) issues as well)
- offset of the corruption is varying, but "large" (not off-by-one or similar)
- debug utility: AFS `fstrace` (invoked by test script, output is hard to read)
- corruption occurred with various file sizes (<5GB), now concentrating on 2MB files
- pattern:
 - ◆ should have something that can be checked programmatically (instead of diff/cmp), so we can move the check into the kernel or check AFS cache chunks directly
 - ◆ suggest pattern with "prime" repeating factor and unlimited length. Something like `<!16bitNumericIPLSB><!8bitStreamNo><!32bitCounter><` (IP repeats every 7 bytes)
- OpenAFS version to use for experimenting: 1.3.8x from lxcert-amd64

Invoking the test scripts

(from Rainer)

1. `~rtb/bin/disk_stress -ri ~rtb/t[45]/d[01]`

runs the stress test on 2 AFS volumes (both on afs32), each volume has two directories (d[01]) => 4 "streams".

In order to run it on several clients, better mkdir more subdirectories and start disk_stress with different subdirectory names on the different clients.

-r means repeatedly, -i means "compare immediately after the copy". disk_stress has other options: one that **may** be interesting is to not do the test in AFS alone, but always copy **from** a file in /tmp.

2. `fstrace setlog -log cmfx -buffer 256 fstrace setset cm -active`

turns on fstrace. The script will do itself an 'fstrace dump' as soon as a compare has failed.

Hardware

lxs5011 (flaky)

lxs5012

lxs5013

lxs5014

lxs5015

lxs5016

lxs5017

lxs5018

Test Log

(please record any tests you do here, including time for the corruption to happen)

1. Peter: reproducing using Rainer's scripts, on 7 lxs machines (day 1: no errors)

day 2: lxs5013 and lxs5016 has exited with cmp fail, both at the same time: Thu Sep 15 17:39:57 2005.
lxs5013 was cmp'ing ~rtb/t4/d2/disk_stress.569 with ~rtb/t4/d2/disk_stress.0, while lxs5016 was cmp'ing
~rtb/t5/d1/disk_stress.88 with ~rtb/t5/d1/disk_stress.0. Manually cmp'ing those pairs on the original nodes and
on other nodes returns no errors. The incidents happened exactly at the same time on two different volumes on
the same server!

This topic: LinuxSupport > AFSCacheCorruption

Topic revision: r7 - 2005-09-17 - unknown



Copyright &© 2008-2024 by the contributing authors. All material on this
collaboration platform is the property of the contributing authors.
or Ideas, requests, problems regarding TWiki? use Discourse or Send feedback