

Ceph - Tasks #4560

unit tests for src/os/LFNIndex.cc

03/26/2013 02:09 PM - Loic Dachary

Status:	Closed	Start date:	03/26/2013
Priority:	Normal	Due date:	03/29/2013
Assignee:	Loic Dachary	% Done:	100%
Category:	qa	Estimated time:	0.00 hour
Target version:		Spent time:	8.00 hours
Tags:		Affected Versions:	
Reviewed:		Pull request ID:	
Description			
work in progress			
Create a set of unit tests in <code>src/test/os/TestLFNIndex.cc</code> covering the code src/os/LFNIndex.cc			

Associated revisions

Revision c8ac1ee9 - 03/28/2013 07:58 AM - Loic Dachary

unit test LFNIndex::lfn_get_name

The escape logic is tested for

- leading . => \.
- / => \s
- \ => \\
- leading DIR_ => \d

The file names for small object names (size < FILENAME_PREFIX_LEN) are created with CEPH_NOSNAP and checked to contain the _head string and not the _long string.

The file names for long object names (size >= FILENAME_PREFIX_LEN) are tested to contain the _long string. A matching file is created to check that it is removed unless it contains the expected extended attribute.

If the SHA1 of two long object names collide and they have the same prefix, lfn_get_name increments an anticollision counter to differentiate them. This condition is engineered because it would be really difficult to find two long names that actually create such a collision.

The lfn_get_name method is private and the get_mangled_name method is used to access it. The out_path argument is not available and cannot be tested. However it is a trivial concatenation of the strings in the path vector.

A TestIndex class is derived from the LFNIndex class to set the pure virtual functions. The TestLFNIndex fixture is derived from it so that the tests get access to the protected methods of LFNIndex

The SetUp method of the fixture creates a PATH directory to be used by all tests as the base path for all object files.

<http://tracker.ceph.com/issues/4560> refs #4560

Signed-off-by: Loic Dachary <loic@dachary.org>

Revision 972f0eb0 - 03/29/2013 08:46 AM - Loic Dachary

unit test LFNIndex::remove_object and LFNIndex::lfn_unlink

When the object name is short, check that the corresponding file is ::unlink()ed. When the object name is long, there may be multiple files with the same name, modulo the anti-collision number showing just before the FILENAME_COOKIE. The following scenarii are tested:

- there only is one file
- there are multiple files and the last one is removed
- there are multiple files and the last one is moved in place of the file that is to be removed

lfn_unlink and remove_object are tested together because lfn_unlink is a private function and remove_object is a protected function that does very little beside calling lfn_unlink

<http://tracker.ceph.com/issues/4560> refs #4560

Signed-off-by: Loic Dachary <loic@dachary.org>

History

#1 - 03/26/2013 02:10 PM - Loic Dachary

- Description updated

#2 - 03/26/2013 02:52 PM - Loic Dachary

```
(05:29:28 PM) loicd: sjust: Hi ! I see you authored a many commits from https://github.com/ceph/ceph/blob/master/src/os/LFNIndex.cc . Would you have an advice regarding unit tests ? It is fine if you don't I'll just go a head and write them ;-)  
(05:43:35 PM) sjust: loicd: the pure functions in LFNIndex.h are good candiates for unit testing  
(05:43:54 PM) sjust: unfortunately, that's mostly just the methods in the manglers/demanglers section  
(05:43:56 PM) sjust: *section  
(05:44:02 PM) sjust: at the bottom  
(05:44:08 PM) sjust: the rest are thin wrappers over IO  
(05:45:04 PM) ***loicd looking  
(05:45:24 PM) sjust: lfn_get_name along with lfn_unlink are the least trivial pieces of machinery  
(05:45:49 PM) sjust: and I don't think they have explicit test coverage  
(05:47:09 PM) loicd: I'll start with them. It's a little tricky because they are private.  
(05:48:15 PM) sjust: loicd: urgh, would prefer they not be public, but there's probably no real harm  
(05:48:32 PM) sjust: ideally, you could make a public debug_runtests() method  
(05:48:41 PM) sjust: not sure  
(05:49:25 PM) sjust: test/filestore/store_test.cc has some testing for the collection index machinery  
(05:49:31 PM) loicd: I'll try to do with the protected/public methods. If I'm stuck I'll figure out a minimal way to create the conditions for a good unit test.  
(05:49:38 PM) sjust: though I wouldn't classify it as a unit test
```

#3 - 03/27/2013 01:37 PM - Loic Dachary

- % Done changed from 0 to 10

#4 - 03/29/2013 02:39 PM - Loic Dachary

- % Done changed from 10 to 20

#5 - 05/27/2013 07:05 AM - Loic Dachary

- Status changed from In Progress to Closed

- % Done changed from 20 to 100

There is still more work to be done but another ticket can be re-opened if someone wants to work on it.