bluestore - Bug #45613
ObjectStore/StoreTestSpecificAUSize.SyntheticMatrixCompression/2 failed
05/20/2020 12:18 AM - Neha Ojha

Status: Resolved
Priority: Urgent
Assignee: Category:
Target version:
Source:
Tags: Backport: octopus
Regression: No
Severity: 3 - minor
Reviewed:

Affected Versions:
ceph-qa-suite:
Pull request ID: 35497
Crash signature (v1):
Crash signature (v2):

Description
---------------------- 1 / 8

2020-05-19T09:58:56.724 INFO:teuthology.orchestra.run.smithi135.stdout: bluestore_min_alloc_size = 4096
2020-05-19T09:58:56.724 INFO:teuthology.orchestra.run.smithi135.stdout: max_write = 1048576
2020-05-19T09:58:56.725 INFO:teuthology.orchestra.run.smithi135.stdout: bluestore_default_buffered_write = false
2020-05-19T09:58:56.725 INFO:teuthology.orchestra.run.smithi135.stdout: bluestore_sync_submit_transaction = true
2020-05-19T09:58:56.726 INFO:teuthology.orchestra.run.smithi135.stderr: seeding object 0
2020-05-19T09:58:56.862 INFO:teuthology.orchestra.run.smithi135.stderr: seeding object 500
2020-05-19T09:58:56.988 INFO:teuthology.orchestra.run.smithi135.stderr: available_objects: 1000 in_flight_objects: 0 total objects: 1000 in_flight 0
thread 7f8dfa42f40 time 2020-05-19T10:00:14.820590+0000
2020-05-19T10:00:14.824 INFO:teuthology.orchestra.run.smithi135.stderr: ceph version 16.0.0-1670-g1f885b6 (1f885b6c2fdde1f3ab249a56f7b4b76356f4373) pacific (dev)
2020-05-19T10:00:14.824 INFO:teuthology.orchestra.run.smithi135.stderr: 1: (ceph::__ceph_assert_failure(char const*, char const*, int, char const*)+0x158) [0x7f8de662d90]
2020-05-19T10:00:14.824 INFO:teuthology.orchestra.run.smithi135.stderr: 2: ()+0x275faa [0x7f8def62faa]
2020-05-19T10:00:14.825 INFO:teuthology.orchestra.run.smithi135.stderr: 3: (StoreTest::doSyntheticTest(int, unsigned long, unsigned long, unsigned long)+0x15af) [0x5570f36f107f]
2020-05-19T10:00:14.825 INFO:teuthology.orchestra.run.smithi135.stderr: 4: (StoreTestSpecificAUSize::do_matrix_choose(char const* [*] [10], int, int, int, std::function<void (unsigned long, unsigned long, unsigned long, unsigned long)+0x1f6) [0x5570f36f7486]
2020-05-19T10:00:14.825 INFO:teuthology.orchestra.run.smithi135.stderr: 5: (StoreTestSpecificAUSize::do_matrix_choose(char const* [*] [10], int, int, int, std::function<void (unsigned long, unsigned long, unsigned long, unsigned long)+0x148) [0x5570f36f73d8]
2020-05-19T10:00:14.825 INFO:teuthology.orchestra.run.smithi135.stderr: 6: (StoreTestSpecificAUSize::do_matrix_choose(char const* [*] [10], int, int, int, std::function<void (unsigned long, unsigned long, unsigned long, unsigned long)+0x148) [0x5570f36f73d8]
2020-05-19T10:00:14.825 INFO:teuthology.orchestra.run.smithi135.stderr: 7: (StoreTestSpecificAUSize::do_matrix_choose(char const* [*] [10], int, int, int, std::function<void (unsigned long, unsigned long, unsigned long, unsigned long)+0x148) [0x5570f36f73d8]
2020-05-19T10:00:14.825 INFO:teuthology.orchestra.run.smithi135.stderr: 8: (StoreTestSpecificAUSize::do_matrix_choose(char const* [*] [10], int, int, int, std::function<void (unsigned long, unsigned long, unsigned long, unsigned long)+0x148) [0x5570f36f73d8]
2020-05-19T10:00:14.825 INFO:teuthology.orchestra.run.smithi135.stderr: 9: (StoreTestSpecificAUSize::do_matrix_choose(char const* [*] [10], int, int, int, std::function<void (unsigned long, unsigned long, unsigned long, unsigned long)+0x148) [0x5570f36f73d8]
2020-05-19T10:00:14.826 INFO:teuthology.orchestra.run.smithi135.stderr: 10: (StoreTestSpecificAUSize::do_matrix_choose(char const* [*] [10], int, int, int, std::function<void (unsigned long, unsigned long, unsigned long, unsigned long)+0x148) [0x5570f36f73d8]
2020-05-19T10:00:14.826 INFO:teuthology.orchestra.run.smithi135.stderr: 11: (StoreTestSpecificAUSize::do_matrix(char const* [*] [10], std::function<void (unsigned long, unsigned long, unsigned long, unsigned long)>)+0xeef) [0x5570f36f7d1e]
2020-05-19T10:00:14.827 INFO:teuthology.orchestra.run.smithi135.stderr: 12: (StoreTestSpecificAUSize_SyntheticMatrixCompression_Test::TestBody()+0x20b) [0x5570f3641aeb]
2020-05-19T10:00:14.827 INFO:teuthology.orchestra.run.smithi135.stderr: 13: (void testing::internal::HandleExceptionsInMethodIfSupported<testing::Test, void>(testing::Test*, void (testing::Test::*)(), char const*+0x8f) [0x5570f399a9e5f]
2020-05-19T10:00:14.827 INFO:teuthology.orchestra.run.smithi135.stderr: 14: (testing::Test::Run()+0xcb) [0x5570f399f7ab]
2020-05-19T10:00:14.827 INFO:teuthology.orchestra.run.smithi135.stderr: 15: (testing::TestInfo::Run()+0x135) [0x5570f399f915]
2020-05-19T10:00:14.827 INFO:teuthology.orchestra.run.smithi135.stderr: 16: (testing::TestSuite::Run()+0xb5) [0x5570f399f9f5]
2020-05-19T10:00:14.827 INFO:teuthology.orchestra.run.smithi135.stderr: 17: (testing::internal::UnitTestImpl::RunAllTests()+0x425) [0x5570f399f5e6b5]
2020-05-19T10:00:14.827 INFO:teuthology.orchestra.run.smithi135.stderr: 18: (bool testing::internal::HandleExceptionsInMethodIfSupported<testing::internal::UnitTestImpl, bool>(testing::internal::UnitTestImpl*, bool (testing::internal::UnitTestImpl::*)(void) [0x5570f399a3cf]
2020-05-19T10:00:14.828 INFO:teuthology.orchestra.run.smithi135.stderr: 19: (testing::UnitTest::Run()+0xa0) [0x5570f39a0160]
2020-05-19T10:00:14.828 INFO:teuthology.orchestra.run.smithi135.stderr: 20: (main()+0x700) [0x5570f3620bf0]
2020-05-19T10:00:14.828 INFO:teuthology.orchestra.run.smithi135.stderr: 21: (__libc_start_main()+0xf3) [0x7f8ded9a2873]
2020-05-19T10:00:14.828 INFO:teuthology.orchestra.run.smithi135.stderr: 22: (start()+0xe2) [0x5570f363faae]

/a/nojha-2020-05-19_00:53:41-rados-wip-revert-34894-distro-basic-smithi/5068255

Related issues:
Duplicated by bluestore - Bug #48827: Ceph Bluestore OSDs fail to start on WA... New
Copied to bluestore - Backport #46350: octopus: ObjectStore/StoreTestSpecific... Resolved

History
#1 - 05/20/2020 12:14 PM - Igor Fedotov
store_test's log contains:

-17> 2020-05-19T10:00:14.304+0000 7f8dfa42f240  3 rocksdb: [db/db_impl_open.cc:518] db.wal/000037.log: dropping 534816 bytes; Corruption: error in middle of record
-16> 2020-05-19T10:00:14.304+0000 7f8dfa42f240  3 rocksdb: [db/db_impl_open.cc:518] db.wal/000037.log: dropping 32757 bytes; Corruption: 06/05/2021
missing start of fragmented record(1)
-15> 2020-05-19T10:00:14.304+0000 7f8dfa42f240  3 rocksdb: [db/db_impl_open.cc:518] db.wal/000037.log: dropping 32757 bytes; Corruption: missing start of fragmented record(1)
-14> 2020-05-19T10:00:14.304+0000 718dafa421240  3 rocksdb: [db/db_impl_open.cc:518] db.wal/000037.log: dropping 27445 bytes; Corruption: missing start of fragmented record(2)

which looks to some degree similar to the report at:
https://lists.ceph.io/hyperkitty/list/ceph-users@ceph.io/thread/CX5PRFGL6UBFOMJC6CLUMPLMT4B2CXVQ/

So I presume this is a recent regression unrelated to compression or test case itself.

---

#2 - 05/21/2020 01:18 AM - Neha Ojha
Reproduces pretty easily on master: /a/nojha-2020-05-20_19:45:21-rados-master-distro-basic-smithi/5073678. Haven't been able to reproduce it on octopus yet.

#3 - 05/21/2020 10:00 PM - Brad Hubbard
/a/bhubbard-2020-05-20_04:16:41-rados-wip-badone-testing-7-distro-basic-smithi/5071455

#4 - 05/21/2020 10:35 PM - Brad Hubbard
[Redacted]

#5 - 05/22/2020 03:53 PM - Igor Fedotov
- Status changed from New to Fix Under Review
- Backport set to octopus
- Pull request ID set to 35201

#6 - 05/22/2020 04:02 PM - Igor Fedotov
So the bug is caused by submitting overlapping write requests for BlueFS WAL via libaio. BlueFS::_flush_range might extend the first request with zero filled data block to fill tail of RocksDB WAL when preextended WALs are enabled. This request might race with the subsequent request adding data to a WAL since it looks like libaio might execute input requests in arbitrary order. As a result valid WAL data might be overwritten with zeros. Aside from WAL preextending the final trigger for the bug to appear was recent disabling bluefs buffer io which brought libaio write path into an action.

#7 - 05/22/2020 04:27 PM - Dan van der Ster
@Igor: So setting bluefs_preextend_wal_files=false and/or bluefs_buffered_io=true should workaround the issue until the patch is published?

#8 - 05/22/2020 05:26 PM - Igor Fedotov
@Dan - right. Setting bluefs_preextend_wal_files seems better to me (one can even try to do that on the fly) but I'd like to hear about the actual experience from the field - still not 100% certain it's absolutely safe.

#9 - 05/22/2020 05:30 PM - Igor Fedotov
At the same time IIUC it's OSD restart which reveals data corruption - while OSD is running it doesn't read from WAL and hence corruption isn't effective. OSD even can heal itself by recycling WAL file.

#10 - 05/22/2020 07:37 PM - Dan van der Ster
OK thanks. I'm also just confirming that nautilus would be immune, even though 14.2.10 will change bluefs_bufferio_io to false. (Because
bluefs_preextend_wal_files still defaults to false).

#11 - 05/28/2020 02:01 PM - Neha Ojha
- Priority changed from Normal to Urgent

#12 - 06/01/2020 03:48 PM - Neha Ojha
  https://github.com/ceph/ceph/pull/35293 - Fix for octopus has been merged and released in 15.2.3.

#13 - 06/01/2020 09:30 PM - Igor Fedotov
- Status changed from Fix Under Review to Resolved

#14 - 06/05/2020 04:52 PM - Igor Fedotov
- Status changed from Resolved to Fix Under Review

#15 - 06/09/2020 08:51 AM - Igor Fedotov
- Pull request ID changed from 35201 to 35497

Original fix was https://github.com/ceph/ceph/pull/35201 but it was decided to kill WAL preextending instead, see https://github.com/ceph/ceph/pull/35497

#16 - 07/02/2020 01:59 PM - Igor Fedotov
- Status changed from Fix Under Review to Pending Backport

#17 - 07/03/2020 03:37 PM - Nathan Cutler
- Copied to Backport #46350: octopus: ObjectStore/StoreTestSpecificAUSize.SyntheticMatrixCompression/2 failed added

#18 - 09/30/2020 03:42 PM - Nathan Cutler
- Status changed from Pending Backport to Resolved

While running with --resolve-parent, the script "backport-create-issue" noticed that all backports of this issue are in status "Resolved" or "Rejected".

#19 - 03/25/2021 02:21 PM - Neha Ojha
- Duplicated by Bug #48827: Ceph Bluestore OSDs fail to start on WAL corruption added