RADOS - Bug #20545

erasure coding = crashes

07/07/2017 05:43 PM - Bob Bobington

Status:	Duplicate	**% Done:**	0%
Priority:	High	**Spent time:**	0.00 hour
Assignee:			
Category:			
Target version:	v12.1.0		
Source:	Community (user)	**Affected Versions:**	v12.1.0
Tags:		**ceph-qa-suite:**	
Backport:		**Component(RADOS):**	
Regression:	No	**Pull request ID:**	
Severity:	2 - major	**Crash signature:**	
Reviewed:			

Description

Steps to reproduce:

- Create 4 OSDs and a mon on a machine (4TB disk per OSD, Bluestore, using dm-crypt too), using Luminous RC built from the tag on Github
- ceph osd erasure-code-profile set myprofile k=2 m=1 ruleset-failure-domain=osd ceph osd pool create imagesrep 256 256 erasure myprofile
- Open an ioctx using rados.py
- Walk a large directory, use ioctx.aio_write(key, data, offset=offset) with 4MB chunks, don't bother waiting for a response

The attached log.txt is the result.

Each time I've tried this, a different OSD has crashed but all display similar tracebacks.

Related issues:

Related to RADOS - Bug #20295: bluestore: Timeout in tp_osd_tp threads when r...	**Resolved**	**06/14/2017**

History

#1 - 07/09/2017 05:49 PM - Bob Bobington

Actually I thought to test this with filestore on BTRFS and it fails there in the same way as well. This seems to be an issue with Ceph rather than Bluestore.

#2 - 07/09/2017 06:27 PM - Bob Bobington

I ran Rados bench on the same cluster and it seems to be working fine, so it seems that something about my Python code is causing the crash. Here it is:

```
import rados
cluster = rados.Rados(conffile='/etc/ceph/ceph.conf')
cluster.connect()
block_size = 1024*1024*4
for root, dirnames, filenames in os.walk('/path/to/stuff'):
    for filename in filenames:
        fullpath = os.path.join(root, filename)
        key = os.path.relpath(fullpath, '/path/to/stuff')
        with open(fullpath, 'r') as infile:
            offset = 0
            while True:
                data = infile.read(block_size)
                ioctx.aio_write(key, data, offset=offset)
                if len(data) < block_size:
                    break
                offset += block_size
```

#3 - 07/09/2017 06:34 PM - Bob Bobington

Sorry, forgot a line of the code. Here's the exact process I'm using to do this:

Shell:

```
for x in k l g h
  sudo wipefs -a /dev/sd$x
  sudo ceph-disk prepare --fs-type btrfs --dmcrypt /dev/sd$x
  sudo ceph-disk activate /dev/sd"$x"1
end
ceph osd erasure-code-profile set myprofile k=2 m=1 ruleset-failure-domain=osd
ceph osd pool create pool 256 256 erasure myprofile
```

Python

```
import rados
cluster = rados.Rados(conffile='/etc/ceph/ceph.conf')
cluster.connect()
ioctx = cluster.open_ioctx('pool')
block_size = 1024*1024*4
for root, dirnames, filenames in os.walk('/path/to/stuff'):
    for filename in filenames:
        fullpath = os.path.join(root, filename)
        key = os.path.relpath(fullpath, '/path/to/stuff')
        with open(fullpath, 'r') as infile:
            offset = 0
            while True:
                data = infile.read(block_size)
                ioctx.aio_write(key, data, offset=offset)
                if len(data) < block_size:
                    break
                offset += block_size
```

#4 - 07/12/2017 03:32 PM - Josh Durgin

From the log the backtrace is:

```
-- Logs begin at Thu 2017-07-06 16:44:09 PDT, end at Fri 2017-07-07 00:10:13 PDT. --
Jul 07 00:03:36 hostname ceph-osd[14090]: *** Caught signal (Aborted) **
Jul 07 00:03:36 hostname ceph-osd[14090]:  in thread 7fa734b32700 thread_name:tp_osd_tp
Jul 07 00:03:36 hostname ceph-osd[14090]:  ceph version 12.1.0 (262617c9f16c55e863693258061c5b25dea5b086) lumi
nous (dev)
Jul 07 00:03:36 hostname ceph-osd[14090]:  1: (()+0x95cd16) [0x55eeb36d4d16]
Jul 07 00:03:36 hostname ceph-osd[14090]:  2: (()+0x11940) [0x7fa7509ca940]
Jul 07 00:03:36 hostname ceph-osd[14090]:  3: (pthread_cond_wait()+0x1fd) [0x7fa7509c639d]
Jul 07 00:03:36 hostname ceph-osd[14090]:  4: (Throttle::_wait(long)+0x33e) [0x55eeb370f38e]
Jul 07 00:03:36 hostname ceph-osd[14090]:  5: (Throttle::get(long, long)+0x2a2) [0x55eeb3710102]
Jul 07 00:03:36 hostname ceph-osd[14090]:  6: (BlueStore::queue_transactions(ObjectStore::Sequencer*, std::vec
tor<ObjectStore::Transaction, std::allocator<ObjectStore::Transaction> >&, boost::intrusive_ptr<TrackedOp>, Th
readPool::TPHandle*)+0xde4) [0x55eeb35dc684]
Jul 07 00:03:36 hostname ceph-osd[14090]:  7: (non-virtual thunk to PrimaryLogPG::queue_transactions(std::vect
or<ObjectStore::Transaction, std::allocator<ObjectStore::Transaction> >&, boost::intrusive_ptr<OpRequest>)+0x6
8) [0x55eeb332c2c8]
Jul 07 00:03:36 hostname ceph-osd[14090]:  8: (ECBackend::handle_sub_write(pg_shard_t, boost::intrusive_ptr<Op
Request>, ECSubWrite&, ZTracer::Trace const&, Context*)+0x960) [0x55eeb3455ca0]
Jul 07 00:03:36 hostname ceph-osd[14090]:  9: (ECBackend::handle_message(boost::intrusive_ptr<OpRequest>)+0x32
1) [0x55eeb346dd21]
Jul 07 00:03:36 hostname ceph-osd[14090]:  10: (PrimaryLogPG::do_request(boost::intrusive_ptr<OpRequest>&, Thr
eadPool::TPHandle&)+0x64a) [0x55eeb32d437a]
Jul 07 00:03:36 hostname ceph-osd[14090]:  11: (OSD::dequeue_op(boost::intrusive_ptr<PG>, boost::intrusive_ptr
<OpRequest>, ThreadPool::TPHandle&)+0x1bc) [0x55eeb317080c]
Jul 07 00:03:36 hostname ceph-osd[14090]:  12: (PGQueueable::RunVis::operator()(boost::intrusive_ptr<OpRequest
> const&)+0x5a) [0x55eeb3170c3a]
Jul 07 00:03:36 hostname ceph-osd[14090]:  13: (OSD::ShardedOpWQ::_process(unsigned int, ceph::heartbeat_handl
e_d*)+0x1ae7) [0x55eeb3194037]
Jul 07 00:03:36 hostname ceph-osd[14090]:  14: (ShardedThreadPool::shardedthreadpool_worker(unsigned int)+0x67
2) [0x55eeb371e3d2]
Jul 07 00:03:36 hostname ceph-osd[14090]:  15: (ShardedThreadPool::WorkThreadSharded::entry()+0x10) [0x55eeb37
21b70]
Jul 07 00:03:36 hostname ceph-osd[14090]:  16: (()+0x7297) [0x7fa7509c0297]
Jul 07 00:03:36 hostname ceph-osd[14090]:  17: (clone()+0x3f) [0x7fa74fe501ef]
```

#5 - 07/12/2017 03:33 PM - Greg Farnum

- Project changed from Ceph to RADOS

- Subject changed from Bluestore + erasure coding = crashes to erasure coding = crashes

- Category deleted (OSD)

- Priority changed from Normal to High

So this looks like you're just killing the cluster by overflowing it with infinite IO. The crash is distressing, though.

#6 - 07/20/2017 11:47 PM - Daniel Oliveira

Trying to reproduce this issue in my lab

#7 - 08/02/2017 03:11 PM - Sage Weil

- Related to Bug #20295: bluestore: Timeout in tp_osd_tp threads when running RBD bench in EC pool w/ overwrites added

#8 - 08/02/2017 03:12 PM - Sage Weil

- Status changed from New to Duplicate

I think this is the same as #20295, which we can now reproduce.

Files

log.txt	9.33 KB	07/07/2017	Bob Bobington