

rgw - Bug #8766

multipart minimum size error should be EntityTooSmall

07/07/2014 03:05 PM - Josh Durgin

Status: Resolved	% Done: 80%
Priority: High	Spent time: 0.00 hour
Assignee: Luis Pabon	
Category:	
Target version:	
Source: Community (user)	Affected Versions:
Tags:	ceph-qa-suite:
Backport:	Pull request ID:
Regression: No	Crash signature (v1):
Severity: 3 - minor	Crash signature (v2):
Reviewed:	

Description

This is already in the code as ERR_TOO_SMALL, but is reported to stop any response instead of sending the error on firefly: <http://thread.gmane.org/gmane.comp.file-systems.ceph.user/11367/focus=11433>

There should be an s3test for this too.

History

#1 - 07/29/2014 09:38 AM - Sage Weil

- Status changed from New to 12
- Priority changed from Normal to High

#2 - 08/26/2014 09:24 AM - Ian Colle

- Assignee set to Luis Pabon

#3 - 09/15/2014 06:50 AM - Luis Pabon

Starting to look into this...

#4 - 10/07/2014 12:52 PM - Luis Pabon

- % Done changed from 0 to 80

I have added a test to s3-test to check for EntityTooSmall and it **passes** on the current code. According to AWS an S3 server can return HTTP code 400 with the error code set to EntityTooSmall, and that is what the current RadosGW does.

Please let me know if I am mistaken, but it seems that the code returns the correct information. I will add the new test to s3-test.

For your information, here is the patch I am going to be sending to s3-test:

```
diff --git a/s3tests/functional/test_s3.py b/s3tests/functional/test_s3.py
index 44db1f1..31a0f0d 100644
--- a/s3tests/functional/test_s3.py
+++ b/s3tests/functional/test_s3.py
@@ -4113,14 +4113,12 @@ def transfer_part(bucket, mp_id, mp_keyname, i, part):
     part_out = StringIO(part)
     mp.upload_part_from_file(part_out, i+1)

-def generate_random(size):
+def generate_random(size, part_size=5*1024*1024):
     """
-    Generate the specified number of megabytes of random data.
```

```

+   Generate the specified number random data.
   (actually each MB is a repetition of the first KB)
   """
-   mb = 1024 * 1024
   chunk = 1024
-   part_size = 5 * mb
   allowed = string.ascii_letters
   for x in range(0, size, part_size):
       strpart = ''.join([allowed[random.randint(0, len(allowed) - 1)] for _ in xrange(chunk)])
@@ -4133,14 +4131,14 @@ def generate_random(size):
   if (x == size):
       return

-def _multipart_upload(bucket, s3_key_name, size, do_list=None, headers=None, metadata=None):
+def _multipart_upload(bucket, s3_key_name, size, part_size=5*1024*1024, do_list=None, headers=None, metadata=
None):
   """
   generate a multi-part upload for a random file of specified size,
   if requested, generate a list of the parts
   return the upload descriptor
   """
-   upload = bucket.initiate_multipart_upload(s3_key_name, headers=headers, metadata=metadata)
-   for i, part in enumerate(generate_random(size)):
+   for i, part in enumerate(generate_random(size, part_size)):
       transfer_part(bucket, upload.id, upload.key_name, i, part)

   if do_list is not None:
@@ -4196,6 +4194,19 @@ def test_multipart_upload_multiple_sizes():

   @attr(resource='object')
   @attr(method='put')
+@attr(operation='check failure on multiple multi-part upload with size too small')
+@attr(assertion='fails 400')
+def test_multipart_upload_size_too_small():
+   bucket = get_new_bucket()
+   key="mymultipart"
+   upload = _multipart_upload(bucket, key, 100 * 1024, part_size=10*1024)
+   e = assert_raises(boto.exception.S3ResponseError, upload.complete_upload)
+   eq(e.status, 400)
+   eq(e.error_code, u'EntityTooSmall')
+
+@attr(resource='object')
+@attr(method='put')
   @attr(operation='check contents of multi-part upload')
   @attr(assertion='successful')
   def test_multipart_upload_contents():

```

#5 - 10/07/2014 12:56 PM - Yehuda Sadeh

Maybe the problem is that we don't send the xml body with the appropriate error?

#6 - 10/07/2014 05:30 PM - Luis Pabon

Here is the response from the gateway:

```
=====
ERROR: s3tests.functional.test_s3.test_multipart_upload_size_too_small
=====
Traceback (most recent call last):
  File "/home/lpabon/git/ceph/s3-tests/virtualenv/lib/python2.7/site-packages/nose/case.py", line 197, in runTest
    self.test(*self.arg)
  File "/home/lpabon/git/ceph/s3-tests/s3tests/functional/test_s3.py", line 4203, in test_multipart_upload_size_too_small
    upload.complete_upload()
  File "/home/lpabon/git/ceph/s3-tests/virtualenv/lib/python2.7/site-packages/boto/s3/multipart.py", line 319, in complete_upload
    self.id, xml)
  File "/home/lpabon/git/ceph/s3-tests/virtualenv/lib/python2.7/site-packages/boto/s3/bucket.py", line 1806, in complete_multipart_upload
    response.status, response.reason, body)
S3ResponseError: S3ResponseError: 400 Bad Request
<?xml version="1.0" encoding="UTF-8"?><Error><Code>EntityTooSmall</Code></Error>
```

According to <http://docs.aws.amazon.com/AmazonS3/latest/API/ErrorResponses.html> the error above seems to be fine. It can also be parsed correctly by boto.

#7 - 10/07/2014 05:48 PM - Josh Durgin

The issue was reported on firefly - does it have the same behavior as master, or is there something that should be backported?

#8 - 10/07/2014 10:01 PM - Luis Pabon

Josh Durgin wrote:

The issue was reported on firefly - does it have the same behavior as master, or is there something that should be backported?

Good question, I'll take a look.

#9 - 10/13/2014 07:22 AM - Luis Pabon

I have submitted the following patches:

Update s3-tests with the new small size multipart tests:

<https://github.com/ceph/s3-tests/pull/21>

Update vstart.sh to setup the users for s3-test in ceph master branch:

<https://github.com/ceph/ceph/pull/2712>

#10 - 11/05/2014 01:23 PM - Yehuda Sadeh

- *Status changed from 12 to Resolved*

Tested on firefly, seem to work.