Hi,

I have a 3 DC multisite setup.

The replication is directional like HKG->SGP->US so the bucket is replicated from HKG to SGP and the same bucket is replicated further from SGP to US.

The HKG > SGP connection is pretty fast 12.5 millions objects (600 GB) transferred in 6.5 hours. Once the OSD crashed in SGP, it stopped the complete chain replication and made PG_DAMAGED cluster error. The pg can be repaired but the sync never started back only bucket sync disable/enable helped.

I got OSD crash also in HKG BUT in ASH. ASH no any error, the replication speed is 2 millions objects in 6.5 hours which is like 90 GB of data.

This is the crash of the osd:

```
{
    "backtrace": [
        "(()+0x12b20) [0x7f597d3fbb20]",
        "(gsignal()+0x10f) [0x7f597c067ff]",
        "(abort()+0x127) [0x7f597c050c35]",
        "((+0x9009b) [0x7f597c050c35]",
        "((+0x9653c) [0x7f597ca2597]",
        "((+0x96597) [0x7f597ca2597]",
        "((+0x967f8) [0x7f597ca2778]",
        "((+0x19d24) [0x7f597e168d24]",
        "("tmalloc::allocate_full_cpp_throw_oom(unsigned long)+0x146) [0x7f597e18b0d6]",
        "(rocksdb::Arena::AllocateNewBlock(unsigned long)+0x43) [0x5632f083b9]",
        "(rocksdb::Fallback::AllocateFallback(unsigned long, bool)+0x4b) [0x5632f083b3b]",
        "(rocksdb::ConcurrentArena::AllocateAligned(unsigned long, unsigned long, rocksdb::Logger*)+0x452) [0x5632f07f9e94]",
        "((+0x1103e7e) [0x5632f085a8e7e]",
        "(rocksdb::MemTable::Add(unsigned long, rocksdb::Value, rocksdb::Slice const&, rocksdb::Slice const&, bool, rocksdb::MemTablePostProcessInfo*)+0xc) [0x5632f07f8f9f]",
        "(rocksdb::MemTableAdder::PutCFImpl(unsigned int, rocksdb::Value, rocksdb::Slice const&, rocksdb::Slice const&, rocksdb::Value) +0x452) [0x5632f08520e2]",
        "(rocksdb::MemTableAdder::PutCF(unsigned int, rocksdb::Value, rocksdb::Slice const&, rocksdb::Slice const&) +0x17) [0x5632f08520e7]",
        "(rocksdb::WriteBatch::Iterate(rocksdb::WriteBatch::Handler*) const+0x80) [0x5632f084ac20]",
        "(rocksdb::WriteBatch::WriteBatch::WriteThread::WriteGroup&, unsigned long, rocksdb::ColumnFamilyMemTables*, rocksdb::FlushScheduler*, bool, unsigned long, rocksdb::DB*, bool, bool, bool)+0x149) [0x5632f084ee9]",
        "(rocksdb::DBImpl::WriteableImpl(rocksdb::WriteOptions const&, rocksdb::WriteBatch*, rocksdb::WriteCallback*, unsigned long*, unsigned long, bool, unsigned long*, unsigned long, rocksdb::PreReleaseCallback*)+0x1ac) [0x5632f078a03d]",
```
And couple of hours before it was am RGW crash also in HKG with tmalloc issue too.

"backtrace": [
  "(()+0x12b20) [0x7ff137833b20]
, "(gsignal()+0x10f) [0x7ff135e7b7ff]
, "(abort()+0x127) [0x7ff135e65c35]
, "(()+0x909b) [0x7ff13683109b]
, "(()+0x9653c) [0x7ff13683753c]
, "(()+0x96597) [0x7ff136837597]
, "(()+0x967f8) [0x7ff1368377f8]
, "(()+0x19d24) [0x7ff141006d24]
]
"ceph_version": "15.2.9",
"crash_id": "2021-03-04T14:55:45.094048Z_3d481fd3-7573-4cb7-9b22-20784b418e64",
"entity_name": "osd.5",
"os_id": "centos",
"os_name": "CentOS Linux",
"os_version": "8",
"os_version_id": "8",
"process_name": "ceph-osd",
"stack_sig": "9643c370a20c0d34f5e8965ae4461e2a7cf709ab4183929239bc63d0e1ee9f4",
"timestamp": "2021-03-04T14:55:45.094048Z",
"utsname_HOSTNAME": "hostname",
"utsname_MACHINE": "x86_64",
"utsname_RELEASE": "4.18.0-240.10.1.el8_3.x86_64",
"utsname_SYSNAME": "Linux",
"utsname_VERSION": "#1 SMP Mon Jan 18 17:05:51 UTC 2021"
]
Any idea what I should tune or is it a bug?

Thank you.

History

#1 - 03/05/2021 11:57 AM - Igor Fedotov

It look like out-of-memory case. Do you have any understanding what was the RAM usage prior to that? Aren't you observing unexpectedly high RAM usage (primarily for ceph-osd processes) since then?

#2 - 03/05/2021 04:11 PM - Ist Gab

Igor Fedotov wrote:

It look like out-of-memory case. Do you have any understanding what was the RAM usage prior to that? Aren't you observing unexpectedly high RAM usage (primarily for ceph-osd processes) since then?

Hi Igor,

Thank you very much to pickup my issue.

Some background information about the setup, maybe something wrong.

SGP and HKG has 6 osd nodes, each nodes have 6x15TB SSD, 1 NVME for index pool and each nodes have 2 nvme for wal+db (3 SSD for 1 NVME WAL+rocksdb).

The servers have 256GB memory and 88 Core cpu.

In the

/etc/sysctl.d/ceph-tuning.conf these are set based on redhat and ansible playbook:

fs.aio-max-nr=1048576
fs.file-max=26234859
vm.zone_reclaim_mode=0
vm.swappiness=10
vm.min_free_kbytes=4194303

In the /etc/sysconfig/ceph the tcmalloc 128MB

TCMALLOC_MAX_TOTAL_THREAD_CACHE_BYTES=134217728

This is the ceph.conf on all nodes included mgr, mon, osd. OSD memory target is around 31GB.

[global]
cluster network = 192.168.198.0/24
fsid = 347b7a4f-9225-4d96-8b28-3f866e28c424
mon initial members = server,server-6s02,server-6s03
osd pool default crush rule = -1
public network = 10.104.198.0/24
rgw_relaxed_s3_bucket_names = true
rgw_dynamic_resharding = false

[mon]
mon_allow_pool_delete = true
mon_pg_warn_max_object_skew = 0
mon_osd_nearfull_ratio = 70
As you've asked, please have a look the grafana dashboard, you can see has plenty of free memories all nodes. (each column is 1 node). You can see how it is smashing the SSDs and nvmes. Yellow and green the rocksdb+wal nvme drives.
This is SGP between March 4 0:00 and March 5 11:00pm

1.png
2.png

This is HKG between March 4 0:00 and March 5 11:00pm

hkg1.png
hkg2.png

And the SGP cluster after many times repaired the pg to take out the cluster from health error, now I have this:
HEALTH_WARN Too many repaired reads on 1 OSDs
[WRN] OSD_TOO_MANY_REPAIRS: Too many repaired reads on 1 OSDs
osd.41 had 16 reads repaired

This is the index pool nvme just FYI.I need to restart this osd to clear out this error. The cluster is new.
It look like out-of-memory case. Do you have any understanding what was the RAM usage prior to that? Aren't you observing unexpectedly high RAM usage (primarily for ceph-osd processes) since then?

I think we are seeing the same issue with 15.2.9, on a non-RGW deployment. We noticed on deploy OOM-killer would go wild and kill all of our servers, requiring a reboot. With the cluster up and running, we see periodic OSD crashes with this:

```
rpm -qa | grep ceph
```
These servers have plenty of RAM (128G, for 24x OSDs with 2x NVME for DB/WAL), and graphs do not show it being completely exhausted except during deploy. Our memory target is default:

```
root@ceph01:~# ceph config get osd osd_memory_target
Inferring fsid 508bbbd0-7067-11eb-baa9-bf10c01557e4
Inferring config /var/lib/ceph/508bbbd0-7067-11eb-baa9-bf10c01557e4/mon.ceph01/config
Using recent ceph image docker.io/ceph/ceph@sha256:4e710662986cf366c282323bfb4c4ca507d7e117c5ccf691a8273732073
297e5
4294967296
root@ceph01:~#
```

This is happening across multiple clusters deployed with 15.2.9, and does repeat; I'd suggest this is a critical bug.

Our deployment is via cephadm, using containers (Podman 3.0.1). Host OS is Ubuntu 20.04.2.

---

**#4 - 03/08/2021 03:51 PM - Dan van der Ster**

[https://tracker.ceph.com/issues/49387](https://tracker.ceph.com/issues/49387) is relevant, no?

Was 15.2.9 built against gperftools 2.8?

---

**#5 - 03/08/2021 03:56 PM - David Orman**

Dan van der Ster wrote:

[https://tracker.ceph.com/issues/49387](https://tracker.ceph.com/issues/49387) is relevant, no?

Was 15.2.9 built against gperftools 2.8?

---

It's definitely what's installed in the 15.2.9 container:

```
root@ceph01:~# cephadm shell
Inferring fsid 012540a0-670e-11eb-b1f8-79b447895d28
Inferring config /var/lib/ceph/012540a0-670e-11eb-b1f8-79b447895d28/mon.ceph01/config
Using recent ceph image docker.io/ceph/ceph@sha256:4e710662986cf366c282323bfb4c4ca507d7e117c5ccf691a8273732073
297e5
root@ceph01:~# rpm -qa |grep gperftools
gperftools-libs-2.8-1.el8.x86_64
root@ceph01:~#
```
#6 - 03/08/2021 04:41 PM - Neha Ojha
- Priority changed from Normal to Urgent

#7 - 03/08/2021 07:14 PM - Ken Dreyer
This build is in epel-testing: https://bodhi.fedoraproject.org/updates/FEDORA-EPEL-2021-0eda4297eb. Can you test it and provide feedback in Bodhi?

#8 - 03/09/2021 03:08 AM - Ist Gab
Ken Dreyer wrote:

This build is in epel-testing: https://bodhi.fedoraproject.org/updates/FEDORA-EPEL-2021-0eda4297eb. Can you test it and provide feedback in Bodhi?

You mean replace the current one, restart osds ...

#9 - 03/12/2021 03:39 PM - David Orman
Ken Dreyer wrote:

This build is in epel-testing: https://bodhi.fedoraproject.org/updates/FEDORA-EPEL-2021-0eda4297eb. Can you test it and provide feedback in Bodhi?

We've tested this against multiple tools that depend on it without issue (and left feedback). We use the container images for Ceph, so will have to wait for those to be rebuilt to validate it fixes our frequent OSD crashes - we definitely agree this is a urgent priority/critical fix - so hopefully new images will be built quickly and updated re: docker hub.

#10 - 03/17/2021 12:53 AM - David Orman
The new (old) version of gperftools has been promoted to stable in EPEL8 (thanks Ken!!). Now we just need a new build of the docker images with this version and hopefully we will stop seeing the OSDs fail.

#11 - 03/17/2021 06:39 PM - Nathan Cutler
So this is a bug in tcmalloc/gperftools 2.8? Has it been reported somewhere?

#12 - 03/18/2021 01:31 PM - Kenneth Waegeman
We are also seeing something that looks like this issue:

```
-42> 2021-03-18T13:17:28.128+0000 7efe89e09700 10 monclient: tick
```
OSDs are crashing now and then...

Thanks!

05/31/2021
#13 - 03/18/2021 02:13 PM - Josh Durgin
- Status changed from New to Resolved

This is fixed by the downgraded version of tcmalloc now in EPEL and the 15.2.10 container images.

#14 - 03/25/2021 04:18 PM - Sage Weil
- Crash signature (v1) updated