CephFS - Bug #43943
qa: "[WRN] evicting unresponsive client smithi131:z (6314), after 304.461 seconds"
02/01/2020 12:56 PM - Patrick Donnelly

| Status:  | Resolved       | % Done:      | 0%       |
| Priority: | Urgent         |             |          |
| Assignee: | Venky Shankar  |             |          |
| Category: |                |             |          |
| Target version: | v16.0.0     |             |          |
| Source: | Q/A            |             |          |
| Tags: |                |             |          |
| Backport: | octopus,nautilus |             |          |
| Regression: | No             |             |          |
| Severity: | 3 - minor      |             |          |
| Reviewed: |                |             |          |
| Affected Versions: |            |             |          |

Description

/a/sage-2020-01-28_03:52:05-rados-wip-sage2-testing-2020-01-27-1839-distro-basic-smithi/4713589
description: rados/mgr/{clusters/{2-node-mgr.yaml} debug/mgr.yaml objectstore/bluestore-bitmap.yaml
supported-random-distro$/{ubuntu_latest.yaml} tasks/module_selftest.yaml}

Part 2 of #40867

Related issues:
- Related to Bug #40867: mgr: failover during in qa testing causes unresponsive client warnings added
- Resolved
- Copied to CephFS - Backport #46199: octopus: qa: "[WRN] evicting unresponsive... resolved
- Resolved
- Copied to CephFS - Backport #46200: nautilus: qa: "[WRN] evicting unresponsive... resolved

History

#1 - 02/01/2020 12:57 PM - Patrick Donnelly
- Related to Bug #40867: mgr: failover during in qa testing causes unresponsive client warnings added

#2 - 02/03/2020 04:36 PM - Venky Shankar

client: 172.21.15.131/0/4191323679 (cephfs instance), registers its addr with ceph-mgr:

2020-01-28T18:10:26.941+0000 7fccd75ba700 7fccc75ba700 7fccc75ba700 7fccc75ba700

but, before the active mgr instance can send its updated client list to monitor, manager transitions to standby

2020-01-28T18:10:29.837+0000 7f8790ea2ec0 0 ceph version 15.0.0-9869-g4b944a6 (4b944a6d8397907af1750fd52b641c
bb82a67ba2) octopus (dev), process ceph-mgr, pid 16726
2020-01-28T18:10:32.833+0000 7f8790ea2ec0 20 mgr send beacon standb
2020-01-28T18:10:32.833+0000 7f8790ea2ec0 10 mgr send beacon sending beacon as gid 8056

Between the call to register and the transitioning to standby, mgr didn't get a chance to call `send_beacon()` (which is called every `mgr_tick_period` seconds), but mds knows about this client.

Maybe, sending a "final" beacon to monitor w/ updated clients before transitioning to standby might work. I'm not sure. Any other approached others
Venky Shankar wrote:

client: 172.21.15.131:0/4191323679 (cephfs instance), registers its addr with ceph-mgr:

[...]

but, before the active mgr instance can send its updated client list to monitor, manager transitions to standby

[...]

Between the call to register and the transitioning to standby, mgr didn't get a chance to call `send_beacon()` (which is called every `mgr_tick_period` seconds), but mds knows about this client.

Maybe, sending a "final" beacon to monitor w/ updated clients before transitioning to standby might work. I'm not sure. Any other approached others can think of?

It's a little different. The mgr sends this beacon:

```
2020-01-28T18:10:26.877+0000 7fccea660700 20 mgr send_beacon active
2020-01-28T18:10:26.877+0000 7fccea660700 10 mgr send_beacon sending beacon as gid 7933
2020-01-28T18:10:26.881+0000 7fccea660700 4 mgr send_beacon going active, including 317 commands in beacon
```

Then it registers some client handles:

```
2020-01-28T18:10:26.905+0000 7fccea660700 7 mgr register_client registering msgr client handle v2:172.21.15.131:0/1320837425
... 2020-01-28T18:09:00.947+0000 7ff68a3c4700 7 mgr unregister_client unregistering msgr client handle v2:172.21.15.131:0/3910140031
... 2020-01-28T18:10:26.925+0000 7fccea660700 7 mgr register_client registering msgr client handle v2:172.21.15.131:0/416558227
... 2020-01-28T18:10:26.941+0000 7fccd75ba700 7 mgr register_client registering msgr client handle v2:172.21.15.131:0/4191323679
```

Then teuthology restarts the mgr:

```
2020-01-28T18:10:29.763 INFO:tasks.ceph.mgr.z:Started
```

Then the mgr starts:

```
2020-01-28T18:10:29.837+0000 7f8790ea2ec0 0 ceph version 15.0.0-9869-g4b944a6 (4b944a6d8397907af1750fd52b641cbb82a57ba2) octopus (dev), process ceph-mgr, pid 16726
```

No chance to send the new beacon before restart.

Besides whitelisting the warning (which I don't want to do), I see two possible solutions here:

1. Fire off a beacon whenever the mgr receives a fatal signal. For the MDS, we send the mons STATE_DNE so rapid failover occurs. Haven't yet looked how the mgr/MgrMonitor works for this case. I think ideally we could send one last beacon to the monitor with the latest client list.

2. Whenever the client instance list changes, send a beacon immediately.
I think (2) is troublesome has you'd need to wire up a notification mechanism so MgrStandby::send_beacon is called. (That method is not ideally placed BTW, the method is called even when the Mgr is active!) That would probably involve creating a dedicated beacon thread like what we have for the MDS which can wait on a condition variable with a timeout.

What do you think Venky?

---

### #4 - 02/04/2020 02:11 PM - Venky Shankar

Patrick Donnelly wrote:

Venky Shankar wrote:

client: 172.21.15.131:0/4191323679 (cephfs instance), registers its addr with ceph-mgr:

[...]

but, before the active mgr instance can send its updated client list to monitor, manager transitions to standby

[...]

Between the call to register and the transitioning to standby, mgr didn't get a chance to call `send_beacon()` (which is called every `mgr_tick_period` seconds), but mds knows about this client.

Maybe, sending a "final" beacon to monitor w/ updated clients before transitioning to standby might work. I'm not sure. Any other approached others can think of?

It's a little different. The mgr sends this beacon:

[...]

Then it registers some client handles:

[...]

Then teuthology restarts the mgr:

[...]

Then the mgr starts:

[...]

No chance to send the new beacon before restart.

Right, that's my understanding too -- maybe my description wasn't detailed.

Besides whitelisting the warning (which I don't want to do), I see two possible solutions here:

1. Fire off a beacon whenever the mgr receives a fatal signal. For the MDS, we send the mons STATE_DNE so rapid failover occurs. Haven't yet looked how the mgr/MgrMonitor works for this case. I think ideally we could send one last beacon to the monitor with the latest client list.

2. Whenever the client instance list changes, send a beacon immediately.

I think (2) is troublesome has you'd need to wire up a notification mechanism so MgrStandby::send_beacon is called. (That method is not ideally placed BTW, the method is called even when the Mgr is active!) That would probably involve creating a dedicated beacon thread like what we have for the MDS which can wait on a condition variable with a timeout.

What do you think Venky?

I would prefer option (1) -- sending a final beacon before going down would be enough.
While running with `--resolve-parent`, the script "backport-create-issue" noticed that all backports of this issue are in status "Resolved" or "Rejected".