CephFS - Fix #46645

librados|libcephfs: use latest MonMap when creating from CephContext

07/20/2020 05:03 PM - Patrick Donnelly

Status:	Resolved	**% Done:**	0%
Priority:	Urgent		
Assignee:	Shyamsundar Ranganathan		
Category:			
Target version:	v16.0.0		
Source:	other	**ceph-qa-suite:**	
Tags:		**Component(FS):**	libcephfs
Backport:	octopus,nautilus	**Labels (FS):**	
Reviewed:		**Pull request ID:**	36533
Affected Versions:		**Crash signature:**	

Description

If the monitor IPs are given at startup via the `--mon_host` switch, those IPs are used for the duration of the process lifetime even if the monitors have been moved elsewhere (e.g. by Rook).

Teach the MonClient to update the CephContext with the current monitor addresses so future instantiations of libcephfs/librados get the correct mon IPs.

Related issues:

Related to RADOS - Bug #47180: qa/standalone/mon/mon-handle-forward.sh failure	**Resolved**
Copied to CephFS - Backport #47013: nautilus: librados|libcephfs: use latest ...	**Resolved**
Copied to CephFS - Backport #47014: octopus: librados|libcephfs: use latest M...	**Resolved**

History

#1 - 07/29/2020 02:48 PM - Shyamsundar Ranganathan

Capturing a mail conversation on direction of fix:

> Shyamsundar

Patrick

> MonClient::handle_monmap handles the message CEPH_MSG_MON_MAP. As a part
> of handling the above message, we should also update global
> CephContext->_conf such that newer instances of MonClient in the same
> address space, is able to pick up the updated MONs from the CephContext
> during initialization, rather than relying on bootstrap settings.
> e.g from:
> - libcephfs::ceph_mount_info::init
> - librados::RadosClient::connect

Right. Keep in mind that CephContext used to be global (I believe) but
recently became instantiated with some library state (like librados)..
librados/libcephfs allow you to use an existing CephContext during
init so CephContext can be shared.

> MonMap::build_initial is responsible to look at various sources for the
> MON map and build an initial MonMap from the same. This method looks at
> the CephCluster configuration section "monmap" first, and then with
> "mon_host", file(entire configs), "mon_dns_srv_name" sections till one
> is found.
>
> We would need to either overwrite one of the above conf sections with

the updated monmap, add a new section to the config (that is looked up first by MonMap::build-initial) or, store it in as a class variable(?).

I think the right approach here is to
have MonClient store the latest MonMap in the CephContext (or just the Mons entity_addrvec_t's if MonMap is generally large). Use that if available before using configs/"monmap".

Questions/Thoughts:
1) Where to store the updated MonMap

a) Add another conf section to CephContext, say "monmap-updates", and store updated contents in there, reading from this conf section first for future calls into MonMap::build_initial
- Preserves original config sections as is

b) Update the "monmap" section
- Requires converting the MMonMap message into "monmap" format for future use
- Overwrites whatever was present in the initial "monmap"

c) (for completeness) MonMap class variable?

2) Format of the data to store

a) Store the contents of MonMap::decode into a stream in CephCluster config using 1.a/c scheme

b) We could store the entire MMonMap message, as is, in the config (assuming 1.a/c above)

c) In case of 1.b we would need to parse and store it in the existing "monmap" format
- Decode MMonMap message extracting required information and store it as in 1.b
- Seems lossy in terms of data that we could store

3) We would also need to handle this in crimson::Client::handle_monmap, right?

4) For tests enhance, src/test/mon/MonMap.cc
- test build_initial with a refreshed MonMap, and ensure initial list is built correctly

#2 - 08/10/2020 02:11 PM - Shyamsundar Ranganathan

- Status changed from New to In Progress

- Pull request ID set to 36533

#3 - 08/11/2020 08:17 PM - Shyamsundar Ranganathan

- Status changed from In Progress to Fix Under Review

#4 - 08/18/2020 04:01 PM - Patrick Donnelly

- Status changed from Fix Under Review to Pending Backport

#5 - 08/18/2020 04:03 PM - Patrick Donnelly

- Copied to Backport #47013: nautilus: librados|libcephfs: use latest MonMap when creating from CephContext added

#6 - 08/18/2020 04:03 PM - Patrick Donnelly

- Copied to Backport #47014: octopus: librados|libcephfs: use latest MonMap when creating from CephContext added

#7 - 09/02/2020 09:46 PM - Patrick Donnelly

- Related to Bug #47180: qa/standalone/mon/mon-handle-forward.sh failure added

#8 - 09/30/2020 03:41 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".