Ceph - Bug #40104

maybe_remove_pg_upmap can be super inefficient for large clusters

06/01/2019 01:33 AM - xie xingguo

Status:	Resolved	**% Done:**	0%
Priority:	Normal	**Spent time:**	0.00 hour
Assignee:	xie xingguo		
Category:	OSDMap		
Target version:			
Source:	Community (dev)	**Affected Versions:**	
Tags:		**ceph-qa-suite:**	
Backport:	luminous,mimic,nautilus	**Pull request ID:**	28373
Regression:	No	**Crash signature (v1):**	
Severity:	3 - minor	**Crash signature (v2):**	
Reviewed:			

Description

Report from Tom Byrne, Senior storage system administrator at the Rutherford Appleton Laboratory (RAL), part of STFC:

> I wasn't sure if I had managed to explain my problem well enough, so I would like to explain it below in writing and get your thoughts on it.
>
> I am worried about the amount of time to create a new OSD map on our large 5000 OSD, 25000 PG cluster that uses upmap.
>
> Before we used upmap, our OSD map creation time was 2-3 seconds. After the upmap balancer had balanced the cluster and added ~14000 upmap item entries, the OSD map creation time after any cluster change (stopping OSD, reweighting OSD) took about 15 seconds, which was significantly longer and is causing us issues with the monitors hanging, and blocked requests as the cluster continues to try and talk to the down OSDs.
>
> Looking at the logs with debugging turned up while the leader monitor generates a new OSD map, I traced the extra OSDmap creation time to the maybe_remove_pg_upmap function. It appears that for any change to the cluster, no matter how small, the maybe_remove_pg_upmap function checks all upmap entries in the OSD map for validity when creating the new OSD map. It seems to do this in a single thread, so the time taken scales with more upmap entries.
>
> Does this sound like the behaviour you expect? I'm not massively familiar with the Ceph codebase so I may be wrong about this.
>
> It seems to me that there are two possible ways to improve the situation:
>
> - Reduce the amount of pg_upmaps that have to be checked for removal. Possibly only check for removal on OSDs that have changed state?
>
> Are either of these options sensible? Or do you think I have a different problem than I have described.
>
> Thank you,
> Tom

Related issues:

Copied to Ceph - Backport #40229: luminous: maybe_remove_pg_upmap can be supe...	**Resolved**
Copied to Ceph - Backport #40230: mimic: maybe_remove_pg_upmap can be super i...	**Resolved**
Copied to Ceph - Backport #40231: nautilus: maybe_remove_pg_upmap can be supe...	**Resolved**

History

#1 - 06/01/2019 01:34 AM - xie xingguo

- Description updated

#2 - 06/04/2019 01:26 AM - xie xingguo

- Pull request ID set to 28373

#3 - 06/06/2019 10:20 AM - xie xingguo

- Status changed from 12 to Pending Backport

#4 - 06/10/2019 10:28 AM - Nathan Cutler

- Copied to Backport #40229: luminous: maybe_remove_pg_upmap can be super inefficient for large clusters added

#5 - 06/10/2019 10:28 AM - Nathan Cutler

- Copied to Backport #40230: mimic: maybe_remove_pg_upmap can be super inefficient for large clusters added

#6 - 06/10/2019 10:28 AM - Nathan Cutler

- Copied to Backport #40231: nautilus: maybe_remove_pg_upmap can be super inefficient for large clusters added

#7 - 08/26/2019 02:57 PM - Nathan Cutler

- Status changed from Pending Backport to Resolved