ceph-volume - Feature #47295

Optimize ceph-volume inventory to reduce runtime

09/03/2020 09:17 PM - Paul Cuzner

Status:	Rejected	**% Done:**	0%
Priority:	Normal		
Assignee:	Paul Cuzner		
Category:			
Target version:	v16.0.0		
Source:		**Reviewed:**	
Tags:		**Affected Versions:**	
Backport:	octopus,nautilus	**Pull request ID:**	37274

Description

The inventory process currently relies on repeated invocation of subprocess calls, which are expensive. On my test system (16 drives), the inventory command issued over 160 calls and took 7 secs to complete.

The goal of this feature is to optimise how the data is gathered to reduce the overheads which in turn will reduce the runtime to the user/caller.

Related issues:

Related to ceph-volume - Bug #37490: ceph-volume lvm list is O(n^2)	**Resolved**	**11/30/2018**

History

#1 - 09/04/2020 05:26 AM - Paul Cuzner

there's a couple of things that impact the runtime that I need some background on

for every block device,
we run the ceph-bluestore-tool, but bluestore is configured on LV's so every command fails anyway
we query for the first lv - but we pass the physical device during an inventory not the vg/lv - so again this returns nothing and soaks time

I've batched up some of the lsblk, and pvs commands, and skipped the two scenarios above and on my test system this brings the runtime of an inventory down from >7s to ~3 (16 devices)

Can anyone comment on the above?

#2 - 09/04/2020 12:03 PM - Jan Fajerski

- Related to Bug #37490: ceph-volume lvm list is O(n^2) added

#3 - 09/04/2020 12:10 PM - Jan Fajerski

Paul Cuzner wrote:

> there's a couple of things that impact the runtime that I need some background on
>
> for every block device,
> we run the ceph-bluestore-tool, but bluestore is configured on LV's so every command fails anyway

This was introduced with the new raw mode, which can deploy OSDs on raw block devices. To identify these we call ceph-bluestore-tool.

> we query for the first lv - but we pass the physical device during an inventory not the vg/lv - so again this returns nothing and soaks time

I think this is due to the fairly new different availability notions. Look for available_lvm and available_raw in util/device.py

> I've batched up some of the lsblk, and pvs commands, and skipped the two scenarios above and on my test system this brings the runtime of an inventory down from >7s to ~3 (16 devices)
>
> Can anyone comment on the above?

We started work to improve this already a while ago, see the related issue.
It comes down to the Device class in util/device.py. This class is widely used and was extended for various purposes, so there is a lot of bloat. I would love a major refactor of this class, but due to time constraints and complexity of the task this is still on the back burner. I'm pretty sure we could also optimize the way we dispatch to the subprocess module.

tl;dr: THis is part of the significant tech dept ceph-volume in ceph-volume. I don't think there is a quick fix, since this class is used everywhere but a major rewrite of it would probably pay off.

#4 - 09/06/2020 10:37 PM - Paul Cuzner

Agree a rewrite is probably the better long term goal - but ultimately, if the code relies on lvs/pvs/vgs/blkid/lsblk and ceph-bluestore-tool it's going to be problematic anyway.

The simplest and least risk way to reduce inventory runtime is to multi-thread the Device object creation. In my tests this cuts the runtime by 1/2, with 4 threads (more than 4 doesn't yield further gains, so I suspect contention somewhere...parhaps in lvm)

I think that this be worthwhile as an interim step.

#5 - 09/06/2020 11:17 PM - Paul Cuzner

https://github.com/ceph/ceph/pull/37013

#6 - 09/07/2020 08:16 PM - Nathan Cutler

- Status changed from New to Fix Under Review

- Assignee set to Paul Cuzner

- Pull request ID set to 37013

#7 - 09/25/2020 12:41 AM - Paul Cuzner

change aborted. Jan decided to reimplement my parallelism using his own approach.

#8 - 09/25/2020 12:43 AM - Paul Cuzner

Paul Cuzner wrote:

> change aborted. Jan decided to reimplement my parallelism using his own approach.

https://github.com/ceph/ceph/pull/37274

#9 - 10/01/2020 12:10 PM - Jan Fajerski

- Backport set to octopus,nautilus

- Pull request ID changed from 37013 to 37274

Any specific backport requirements here? Targetting the usual for now.

#10 - 12/03/2020 02:43 AM - Paul Cuzner

- Status changed from Fix Under Review to Rejected

rejected - an alternate approach was implemented