Red Hat Ceph Storage 5.0

Release Notes

Release notes for Red Hat Ceph Storage 5.0z4

Last Updated: 2023-11-07

Red Hat Ceph Storage 5.0 Release Notes

Release notes for Red Hat Ceph Storage 5.0z4

Legal Notice

Abstract

The release notes describe the major features, enhancements, known issues, and bug fixes implemented for the Red Hat Ceph Storage 5 product release. This includes previous release notes of the Red Hat Ceph Storage 5.0 release up to the current release. Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright's message .

Table of Contents

MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright's message.

PROVIDING FEEDBACK ON RED HAT CEPH STORAGE DOCUMENTATION

We appreciate your input on our documentation. Please let us know how we could make it better. To do so, create a Bugzilla ticket:

+ . Go to the *Bugzilla* website. . In the Component drop-down, select **Documentation**. . In the Sub-Component drop-down, select the appropriate sub-component. . Select the appropriate version of the document. . Fill in the **Summary** and **Description** field with your suggestion for improvement. Include a link to the relevant part(s) of documentation. . Optional: Add an attachment, if any. . Click **Submit Bug**.

CHAPTER 1. INTRODUCTION

Red Hat Ceph Storage is a massively scalable, open, software-defined storage platform that combines the most stable version of the Ceph storage system with a Ceph management platform, deployment utilities, and support services.

The Red Hat Ceph Storage documentation is available at https://access.redhat.com/documentation/en-us/red_hat_ceph_storage/5.

CHAPTER 2. ACKNOWLEDGMENTS

Red Hat Ceph Storage version 5.0 contains many contributions from the Red Hat Ceph Storage team. In addition, the Ceph project is seeing amazing growth in the quality and quantity of contributions from individuals and organizations in the Ceph community. We would like to thank all members of the Red Hat Ceph Storage team, all of the individual contributors in the Ceph community, and additionally, but not limited to, the contributions from organizations such as:

- Intel®
- Fujitsu ®
- UnitedStack
- Yahoo ™
- Ubuntu Kylin
- Mellanox ®
- CERN ™
- Deutsche Telekom
- Mirantis ®
- SanDisk ™
- SUSE Linux® Enterprise Server (SLES)

CHAPTER 3. NEW FEATURES

This section lists all major updates, enhancements, and new features introduced in this release of Red Hat Ceph Storage.

The main features added by this release are:

- **Containerized Cluster**
  Red Hat Ceph Storage 5 supports only containerized daemons. It does not support non-containerized storage clusters. If you are upgrading a non-containerized storage cluster from Red Hat Ceph Storage 4 to Red Hat Ceph Storage 5, the upgrade process includes the conversion to a containerized deployment.

  For more information, see the *Upgrading a Red Hat Ceph Storage cluster from RHCS 4 to RHCS 5* section in the *Red Hat Ceph Storage Installation Guide* for more details.

- **Cephadm**
  Cephadm is a new containerized deployment tool that deploys and manages a Red Hat Ceph Storage 5.0 cluster by connecting to hosts from the manager daemon. The **cephadm** utility replaces **ceph-ansible** for Red Hat Ceph Storage deployment. The goal of Cephadm is to provide a fully-featured, robust, and well installed management layer for running Red Hat Ceph Storage.

  The **cephadm** command manages the full lifecycle of a Red Hat Ceph Storage cluster.

  Starting with Red Hat Ceph Storage 5.0, **ceph-ansible** is no longer supported and is incompatible with the product. Once you have migrated to Red Hat Ceph Storage 5.0, you must use **cephadm** and **cephadm-ansible** to perform updates.

  The **cephadm** command can perform the following operations:

- Bootstrap a new Ceph storage cluster.

- Launch a containerized shell that works with the Ceph command-line interface (CLI).

- Aid in debugging containerized daemons.
  The **cephadm** command uses **ssh** to communicate with the nodes in the storage cluster and add, remove, or update Ceph daemon containers. This allows you to add, remove, or update Red Hat Ceph Storage containers without using external tools.

  The **cephadm** command has two main components:

- The **cephadm** shell launches a **bash** shell within a container. This enables you to run storage cluster installation and setup tasks, as well as to run **ceph** commands in the container.

- The **cephadm** orchestrator commands enable you to provision Ceph daemons and services, and to expand the storage cluster.
  For more information, see the *Red Hat Ceph Storage Installation Guide* .

- **Management API**
  The management API creates management scripts that are applicable for Red Hat Ceph Storage 5.0 and continues to operate unchanged for the version lifecycle. The incompatible versioning of the API would only happen across major release lines.

  For more information, see the *Red Hat Ceph Storage Developer Guide*.

- **Disconnected installation of Red Hat Ceph Storage**

Red Hat Ceph Storage 5.0 supports the disconnected installation and bootstrapping of storage clusters on private networks. A disconnected installation uses custom images and configuration files and local hosts, instead of downloading files from the network.

You can install container images that you have downloaded from a proxy host that has access to the Red Hat registry, or by copying a container image to your local registry. The bootstrapping process requires a specification file that identifies the hosts to be added by name and IP address. Once the initial monitor host has been bootstrapped, you can use Ceph Orchestrator commands to expand and configure the storage cluster.

See the *Red Hat Ceph Storage Installation Guide* for more details.

- **Ceph File System geo-replication**
  Starting with the Red Hat Ceph Storage 5 release, you can replicate Ceph File Systems (CephFS) across geographical locations or between different sites. The new **cephfs-mirror** daemon does asynchronous replication of snapshots to a remote CephFS.

  See the *Ceph File System mirrors* section in the *Red Hat Ceph Storage File System Guide* for more details.

- **A new Ceph File System client performance tool**
  Starting with the Red Hat Ceph Storage 5 release, the Ceph File System (CephFS) provides a **top**-like utility to display metrics on Ceph File Systems in realtime. The **cephfs-top** utility is a **curses**-based Python script that uses the Ceph Manager **stats** module to fetch and display client performance metrics.

  See the *Using the **cephfs-top** utility* section in the *Red Hat Ceph Storage File System Guide* for more details.

- **Monitoring the Ceph object gateway multisite using the Red Hat Ceph Storage Dashboard**
  The Red Hat Ceph Storage dashboard can now be used to monitor an Ceph object gateway multisite configuration.

  After the multi-zones are set-up using the **cephadm** utility, the buckets of one zone is visible to other zones and other sites. You can also create, edit, delete buckets on the dashboard.

  See the *Management of buckets of a multisite object configuration on the Ceph dashboard* chapter in the *Red Hat Ceph Storage Dashboard Guide* for more details.

- **Improved BlueStore space utilization**
  The Ceph Object Gateway and the Ceph file system (CephFS) stores small objects and files as individual objects in RADOS. With this release, the default value of BlueStore's **min_alloc_size** for SSDs and HDDs is 4 KB. This enables better use of space with no impact on performance.

  See the *OSD BlueStore* chapter in the *Red Hat Ceph Storage Administration Guide* for more details.

3.1. THE CEPHADM UTILITY

Red Hat Ceph Storage can now automatically tune the Ceph OSD memory target

With this release, **osd_memory_target_autotune** option is fixed, and works as expected. Users can enable Red Hat Ceph Storage to automatically tune the Ceph OSD memory target for the Ceph OSDs in the storage cluster for improved performance without explicitly setting the memory target for the Ceph OSDs. Red Hat Ceph Storage sets the Ceph OSD memory target on a per-node basis by evaluating the total memory available, and the daemons running on the node.

Users can enable the memory auto-tuning feature for the Ceph OSD by running the following command:

```
ceph config set osd osd_memory_target_autotune true
```

3.2. CEPH DASHBOARD

A new Grafana Dashboard to display graphs for Ceph Object Gateway multi-site setup

With this release, a new Grafana dashboard is now available and displays graphs for Ceph Object Gateway multisite sync performance including two-way replication throughput, polling latency, and unsuccessful replications.

See the *Monitoring Ceph object gateway daemons on the dashboard* section in the *Red Hat Ceph Storage Dashboard Guide* for more information.

The Prometheus Alertmanager rule triggers an alert for different MTU settings on the Red Hat Ceph Storage Dashboard

Previously, mismatch in MTU settings, which is a well-known cause of networking issues, had to be identified and managed using the command-line interface. With this release, when a node or a minority of them have an MTU setting that differs from the majority of nodes, an alert is triggered on the Red Hat Ceph Storage Dashboard. The user can either mute the alert or fix the MTU mismatched settings.

See the *Management of Alerts on the Ceph dashboard* section in the *Red Hat Ceph Storage Dashboard Guide* for more information.

User and role management on the Red Hat Ceph Storage Dashboard

With this release, user and role management is now available. It allows administrators to define fine-grained role-based access control (RBAC) policies for users to create, update, list, and remove OSDs in a Ceph cluster.

See the *Management of roles on the Ceph dashboard* in the *Red Hat Ceph Storage Dashboard Guide* for more information.

The Red Hat Ceph Storage Dashboard now supports RBD v1 images

Previously, the Red Hat Ceph Storage Dashboard displayed and supported RBD v2 format images only.

With this release, users can now manage and migrate their v1 RBD images to v2 RBD images by setting the **RBD_FORCE_ALLOW_V1** to **1**.

See the *Management of block devices using the Ceph dashboard* section in the *Red Hat Ceph Storage Dashboard Guide* for more information.

Users can replace the failed OSD on the Red Hat Ceph Storage Dashboard

With this release, users can identify and replace the failed OSD by preserving the *OSD_ID* of the OSDs on the Red Hat Ceph Storage Dashboard.

See *Replacing the failed OSDs on the Ceph dashboard* in the *Red Hat Ceph Storage Dashboard Guide* for more information.

Specify placement target when creating a Ceph Object Gateway bucket on the Red Hat Ceph Storage Dashboard

With this release, users can now specify a placement target when creating a Ceph Object Gateway bucket on the Red Hat Ceph Storage Dashboard.

See the *Creating Ceph object gateway buckets on the dashboard* section in the *Red Hat Ceph Storage Dashboard Guide* for more information.

The Multi-Factor Authentication deletes feature is enabled on the Red Hat Ceph Storage Dashboard

With this release, users can now enable Multi-Factor Authentication deletes (MFA) for a specific bucket from the Ceph cluster on the Red Hat Ceph Storage Dashboard.

See the *Editing Ceph object gateway buckets on the dashboard* section in the *Red Hat Ceph Storage Dashboard Guide* for more information.

The bucket versioning feature for a specific bucket is enabled on the Red Hat Ceph Storage Dashboard

With this release, users can now enable bucket versioning for a specific bucket on the Red Hat Ceph Storage Dashboard.

See the *Editing Ceph object gateway buckets on the dashboard* section in the *Red Hat Ceph Storage Dashboard Guide* for more information.

The object locking feature for Ceph Object Gateway buckets is enabled on the Red Hat Ceph Storage Dashboard

With this release, users can now enable object locking for Ceph Object Gateway buckets on the Red Hat Ceph Storage Dashboard.

See the *Creating Ceph object gateway buckets on the dashboard* section in the *Red Hat Ceph Storage Dashboard Guide* for more information.

The Red Hat Ceph Storage Dashboard has the vertical navigation bar

With this release, the vertical navigation bar is now available. The heartbeat icon on the Red Hat Ceph Storage Dashboard menu changes color based on the cluster status that is green, yellow, and red. Other menus for example Cluster>Monitoring and Block>Mirroring display a colored numbered icon that shows the number of warnings in that specific component.

The "box" page of the Red Hat Ceph Storage dashboard displays detailed information

With this release, the "box" page of Red Hat Ceph Storage Dashboard displays information about the Ceph version, the hostname where the **ceph-mgr** is running, username,roles, and the browser details.

Browser favicon displays the Red Hat logo with an icon for a change in the cluster health status

With this release, the browser favicon now displays the Red Hat logo with an icon that changes color based on cluster health status that is green, yellow, or red.

The error page of the Red Hat Ceph Storage Dashboard works as expected

With this release, the error page of the Red Hat Ceph Storage Dashboard is fixed and works as expected.

Users can view Cephadm workflows on the Red Hat Ceph Storage Dashboard

With this release, the Red Hat Ceph Storage displays more information on inventory such as nodes defined in the Ceph Orchestrator and services such as information on containers. The Red Hat Ceph Storage dashboard also allows the users to manage the hosts on the Ceph cluster.

See the *Monitoring hosts of the Ceph cluster on the dashboard* section in the *Red Hat Ceph Storage Dashboard Guide* for more information.

Users can modify the object count and size quota on the Red Hat Ceph Storage Dashboard

With this release, the users can now set and modify the object count and size quota for a given pool on the Red Hat Ceph Storage Dashboard.

See the *Creating pools on the Ceph dashboard* section in the *Red Hat Ceph Storage Dashboard Guide* for more information.

Users can manage Ceph File system snapshots on the Red Hat Ceph Storage Dashboard

With this release, the users can now create and delete Ceph File System (CephFS) snapshots, and set and modify per-directory quotas on the Red Hat Ceph Storage Dashboard.

Enhanced account and password policies for the Red Hat Ceph Storage Dashboard

With this release, to comply with the best security standards, strict password and account policies are implemented. The user passwords need to comply with some configurable rules. User accounts can also be set to expire after a given amount of time, or be locked out after a number of unsuccessful log-in attempts.

Users can manage users and buckets on any realm, zonegroup or zone

With this release, users can now manage users and buckets not only on the default zone but any realm, zone group, or zone that they configure.

To manage multiple daemons on the Red Hat Ceph Storage Dashboard, see the *Management of buckets of a multi-site object gateway configuration on the Ceph dashboard* in the *Red Hat Ceph Storage Dashboard Guide*.

Users can create a tenanted S3 user intuitively on the Red Hat Ceph Storage Dashboard

Previously, a tenanted S3 user could be created using a user friendly syntax that is "tenant$user" instead of the intuitive separate input fields for each one.

With this release, users can now create a tenanted S3 user intuitively without using "tenant$user" on the Red Hat Ceph Storage Dashboard.

The Red Hat Ceph Storage Dashboard now supports host management

Previously, the command-line interface was used to manage hosts in a Red Hat Ceph Storage cluster.

With this release, users can enable or disable the hosts by using the maintenance mode feature on the Red Hat Ceph Storage Dashboard.

Nested tables can be expanded or collapsed on the Red Hat Ceph Storage Dashboard

With this release, rows that contain nested tables can be expanded or collapsed by clicking on the row on the Red Hat Ceph Storage Dashboard.

3.3. CEPH FILE SYSTEM

CephFS clients can now reconnect after being blocklisted by Metadata Servers (MDS)

Previously, Ceph File System (CephFS) clients were blocklisted by MDS because of network partitions or other transient errors.

With this release, the CephFS client can reconnect to the mount with the appropriate configurations turned ON for each client as manual remount is not needed.

Users can now use the ephemeral pinning policies for automated distribution of subtrees among MDS

With this release, the export pins are improved by introducing efficient strategies to pin subtrees, thereby enabling automated distribution of subtrees among Metadata Servers (MDS) and eliminating user intervention for manual pinning.

See the *Ephemeral pinning policies* section in the *Red Hat Ceph Storage File System Guide* for more information.

mount.ceph has an additional option of recover_session=clean

With this release, an additional option of **recover_session=clean** is added to **mount.ceph**. With this option, the client reconnects to the Red Hat Ceph Storage cluster automatically when it detects that it is blocklisted by Metadata servers (MDS) and the mounts are recovered automatically.

See the *Removing a Ceph File System client from the blocklist* section in the *Red Hat Ceph Storage File System Guide* for more information.

Asynchronous creation and removal of metadata operations in the Ceph File System

With this release, Red Hat Enterprise Linux 8.4 kernel mounts now asynchronously execute file creation and removal on Red Hat Ceph Storage clusters. This improves performance of some workloads by avoiding round-trip latency for these system calls without impacting consistency. Use the new **-o nowsync** mount option to enable asynchronous file creation and deletion.

Ceph File System (CephFS) now provides a configuration option for MDS called mds_join_fs

With this release, when failing over metadata server (MDS) daemons, the cluster's monitors prefer standby daemons with **mds_join_fs** equal to the file system name with the failed **rank**.

If no standby exists with **mds_join_fs** equal to the file system **name**, it chooses an unqualified standby for the replacement, or any other available standby, as a last resort.

See the *File system affinity* section in the *Red Hat Ceph Storage File System Guide* for more information.

Asynchronous replication of snapshots between Ceph Filesystems

With this release, the mirroring module, that is the manager plugin, provides interfaces for managing directory snapshot mirroring. The mirroring module is responsible for assigning directories to the mirror daemons for the synchronization. Currently, a single mirror daemon is supported and can be deployed using **cephadm**.

Ceph File System (CephFS) supports asynchronous replication of snapshots to a remote CephFS through the **cephfs-mirror** tool. A mirror daemon can handle snapshot synchronization for multiple file systems in a Red Hat Ceph Storage cluster. Snapshots are synchronized by mirroring snapshot data followed by creating a snapshot with the same name for a given directory on the remote file system, as the snapshot being synchronized.

See the *Ceph File System mirrors* section in the *Red Hat Ceph Storage File System Guide* for more information.

The cephfs-top tool is supported

With this release, the **cephfs-top** tool is introduced.

Ceph provides a **top(1)** like utility to display the various Ceph File System(CephFS) metrics in realtime. The **cephfs-top** is a curses based python script that uses the **stats** plugin in the Ceph Manager to fetch and display the metrics.

CephFS clients periodically forward various metrics to the Ceph Metadata Servers (MDSs), which then forward these metrics to MDS rank zero for aggregation. These aggregated metrics are then forwarded to the Ceph Manager for consumption.

Metrics are divided into two categories; global and per-mds. Global metrics represent a set of metrics for the file system as a whole for example client read latency, whereas per-mds metrics are for a specific MDS rank for example the number of subtrees handled by an MDS.

Currently, global metrics are tracked and displayed. The **cephfs-top** command does not work reliably with multiple Ceph File Systems.

See the *Using the **cephfs-top** utility* section in the *Red Hat Ceph Storage File System Guide* for more information.

MDS daemons can be deployed with mds_autoscaler plugin

With this release, a new ceph-mgr plugin, **mds_autoscaler** is available which deploys metadata server (MDS) daemons in response to the Ceph File System (CephFS) requirements. Once enabled, **mds_autoscaler** automatically deploys the required standbys and actives according to the setting of **max_mds**.

For more information, see the *Using the MDS autoscaler module* section in *Red Hat Ceph Storage File System Guide*.

Ceph File System (CephFS) scrub now works with multiple active MDS

Previously, users had to set the parameter **max_mds=1** and wait for only one active metadata server (MDS) to run Ceph File System (CephFS) scrub operations.

With this release, irrespective of the value of **mds_max**, users can execute scrub on rank **0** with multiple active MDS.

See the *Configuring multiple active Metadata Server daemons* section in the *Red Hat Ceph Storage File System Guide* for more information.

Ceph File System snapshots can now be scheduled with snap_schedule plugin

With this release, a new ceph-mgr plugin, **snap_schedule** is now available for scheduling snapshots of the Ceph File System (CephFS). The snapshots can be created, retained, and automatically garbage collected.

3.4. CONTAINERS

The cephfs-mirror package is included in the ceph-container ubi8 image

With this release, the cephfs-mirror package is now included in the ceph-container ubi8 image to support the mirroring Ceph File System (CephFS) snapshots to a remote CephFS. The command to configure CephFS-mirror is now available.

See the *Ceph File System mirrors* section in the *Red Hat Ceph Storage File System Guide* for more information.

3.5. CEPH OBJECT GATEWAY

Bucket name or ID is supported in the **radosgw-admin bucket stats** command.

With this release, the bucket name or ID can be used as an argument in the **radosgw-admin bucket stats** command. Bucket stats reports the non-current bucket instances which can be used in debugging a class of large OMAP object warnings that is the Ceph OSD log.

Six new performance counters added to the Ceph Object Gateway's perfcounters

With this release, six performance counters are now available in the Ceph Object Gateway. These counters report on the object expiration and lifecycle transition activity through the foreground and background processing of the Ceph Object Gateway lifecycle system. The **lc_abort_mpu**, **lc_expire_current**, **lc_expire_noncurrent** and **lc_expire_dm** counters permit the estimation of object expiration. The **lc_transition_current** and **lc_transition_noncurrent** counters provide information for lifecycle transitions.

Users can now use object lock to implement WORM-like functionality in S3 object storage

The S3 Object lock is the key mechanism supporting write-once-read-many (WORM) functionality in S3 Object storage. With this release, Red Hat Ceph Storage 5 supports Amazon Web Services (AWS) S3 Object lock data management API and the users can use Object lock concepts like retention period, legal hold, and bucket configuration to implement WORM-like functionality as part of the custom workflow overriding data deletion permissions.

3.6. RADOS

The Red Hat Ceph Storage recovers with fewer OSDs available in an erasure coded (EC) pool

Previously, erasure coded (EC) pools of size **k+m** required at least **k+1** copies for recovery to function. If only **k** copies were available, recovery would be incomplete.

With this release, Red Hat Ceph Storage cluster now recovers with **k** or more copies available in an EC pool.

For more information on erasure coded pools, see the *Erasure coded pools* chapter in the *Red Hat Ceph Storage Storage Strategies Guide*.

Sharding of RocksDB database using column families is supported

With the BlueStore admin tool, the goal is to achieve less read and write amplification, decrease DB (Database) expansion during compaction, and also improve IOPS performance.

With this release, you can reshard the database with the BlueStore admin tool. The data in RocksDB (DB) database is split into multiple Column Families (CF). Each CF has its own options and the split is performed according to type of data such as omap, object data, delayed cached writes, and PGlog.

For more information on resharding, see the *Resharding the RocksDB database using the BlueStore admin tool* section in the *Red Hat Ceph Storage Administration Guide*.

The **mon_allow_pool_size_one** configuration option can be enabled for Ceph monitors

With this release, users can now enable the configuration option **mon_allow_pool_size_one**. Once enabled, users have to pass the flag **--yes-i-really-mean-it** for **osd pool set size 1**, if they want to configure the pool size to **1**.

The osd_client_message_cap option has been added back

Previously, the **osd_client_message_cap** option was removed. With this release, the **osd_client_message_cap** option has been re-introduced. This option helps control the maximum number of in-flight client requests by throttling those requests. Doing this can be helpful when a Ceph OSD flaps due to an overwhelming amount of client-based traffic.

Ceph messenger protocol is now updated to msgr v2.1.

With this release, a new version of Ceph messenger protocol, msgr v2.1, is implemented, which addresses several security, integrity and potential performance issues with the previous version, msgr v2.0. All Ceph entities, both daemons and clients, now default to msgr v2.1.

The new default osd_client_message_cap value is 256

Previously, the **osd_client_message_cap** had a default value of **0**. The default value of **0** disables the flow control feature for the Ceph OSD and does not prevent Ceph OSDs from flapping during periods of heavy client traffic.

With this release, the default value of **256** for **osd_client_message_cap** provides better flow control by limiting the maximum number of inflight client requests.

The set_new_tiebreaker command has been added

With this release, storage administrators can set a new tiebreak Ceph Monitor when running in a storage cluster in stretch mode. This command can be helpful if the tiebreaker fails and cannot be recovered.

3.7. RADOS BLOCK DEVICES (RBD)

Improved librbd small I/O performance

Previously, in an NVMe based Ceph cluster, there were limitations in the internal threading architecture resulting in a single librbd client struggling to achieve more than 20K 4KiB IOPS.

With this release, librbd is switched to an asynchronous reactor model on top of the new ASIO-based neorados API thereby increasing the small I/O throughput potentially by several folds and reducing latency.

Built in schedule for purging expired RBD images

Previously, the storage administrator could set up a cron-like job for the **rbd trash purge** command.

With this release, the built-in schedule is now available for purging expired RBD images. The **rbd trash purge schedule add** and the related commands can be used to configure the RBD trash to automatically purge expired images based on a defined schedule.

See the *Defining an automatic trash purge schedule* section in the *Red Hat Ceph Storage Block Device Guide* for more information.

Servicing reads of immutable objects with the new ceph-immutable-object-cache daemon

With this release, the new **ceph-immutable-object-cache** daemon can be deployed on a hypervisor node to service the reads of immutable objects, for example a parent image snapshot. The new **parent_cache** librbd plugin coordinates with the daemon on every read from the parent image, adding

the result to the cache wherever necessary. This reduces latency in scenarios where multiple virtual machines are concurrently sharing a golden image.

For more information, see the *Management of `ceph-immutable-object-cache` daemons* chapter in the *Red Hat Ceph Storage Block device guide*.

Support for sending compressible or incompressible hints in librbd-based clients

Previously, there was no way to hint to the underlying OSD object store backend whether data is compressible or incompressible.

With this release, **the rbd_compression_hint** configuration option can be used to hint whether data is compressible or incompressible, to the underlying OSD object store backend. This can be done per-image, per-pool or globally.

See the *Block device input and output options* section in the *Red Hat Ceph Storage Block Device Guide* for more information.

Overriding read-from-replica policy in librbd clients is supported

Previously there was no way to limit the inter-DC/AZ network traffic, as when a cluster is stretched across data centers, the primary OSD may be on a higher latency and cost link in comparison with other OSDs in the PG.

With this release, the **rbd_read_from_replica_policy** configuration option is now available and can be used to send reads to a random OSD or to the closest OSD in the PG, as defined by the CRUSH map and the client location in the CRUSH hierarchy. This can be done per-image, per-pool or globally.

See the *Block device input and output options* section in the *Red Hat Ceph Storage Block Device Guide* for more information.

Online re-sparsification of RBD images

Previously, reclaiming space for image extents that are zeroed and yet fully allocated in the underlying OSD object store was highly cumbersome and error prone. With this release, the new **rbd sparsify** command can now be used to scan the image for chunks of zero data and deallocate the corresponding ranges in the underlying OSD object store.

ocf:ceph:rbd cluster resource agent supports namespaces

Previously, it was not possible to use ocf:ceph:rbd cluster resource agent for images that exist within a namespace.

With this release, the new **pool_namespace** resource agent parameter can be used to handle images within the namespace.

RBD images can be imported instantaneously

With the **rbd import** command, the new image becomes available for use only after it is fully populated.

With this release, the image live-migration feature is extended to support external data sources and can be used as an alternative to **rbd import**. The new image can be linked to local files, remote files served over HTTP(S) or remote Amazon S3-compatible buckets in **raw**, **qcow** or **qcow2** formats and becomes available for use immediately. The image is populated as a background operation which can be run while it is in active use.

LUKS encryption inside librbd is supported

Layering QEMU LUKS encryption or dm-crypt kernel module on top of librbd suffers a major limitation

that a copy-on-write clone image must use the same encryption key as its parent image. With this release, support for LUKS encryption has been incorporated within librbd. The new "rbd encryption format" command can now be used to format an image to a **luks1** or **luks2** encrypted format.

3.8. RBD MIRRORING

Snapshot-based mirroring of RBD images

The journal-based mirroring provides fine-grained crash-consistent replication at the cost of double-write penalty where every update to the image is first recorded to the associated journal before modifying the actual image.

With this release, in addition to journal-based mirroring, snapshot-based mirroring is supported. It provides coarse-grained crash-consistent replication where the image is mirrored using the mirror snapshots which can be created manually or periodically with a defined schedule. This is supported by all clients and requires a less stringent recovery point objective (RPO).

3.9. ISCSI GATEWAY

Improved tcmu-runner section in the ceph status output

Previously, each iSCSI LUN was listed individually resulting in cluttering the **ceph status** output.

With this release, the **ceph status** command summarizes the report and shows only the number of active portals and the number of hosts.

3.10. THE CEPH ANSIBLE UTILITY

The cephadm-adopt.yml playbook is idempotent

With this release, the **cephadm-adopt.yml** playbook is idempotent, that is the playbook can be run multiple times. If the playbook fails for any reason in the first attempt, you can rerun the playbook and it works as expected.

For more information, see the *Upgrading from Red Hat Ceph Storage 4 to Red Hat Ceph Storage 5 using `ceph-ansible`* section in the *Red Hat Ceph Storage Installation Guide*.

1. The **pg_autoscaler** and **balancer** modules are now disabled during upgrades Previously Red Hat Ceph Storage did not support disabling the **pg_autoscaler** and **balancer** modules during the upgrade process. This can result in the placement group check failing during the upgrade process, because the **pg_autoscaler** continues adjusting the placement group numbers.

With this release, **ceph-ansible** disables the **pg_autoscaler** and **balancer** modules before upgrading a Ceph OSD node, and then re-enables them after the upgrade completes.

Improvement to the Ceph Ansible rolling_update.yml playbook

Previously, the Ceph Ansible **rolling_update.yml** playbook checked the Ceph version requirement of a container image later during the upgrade process. This resulted in the playbook failing in the middle of the upgrade process.

With this release, the **rolling_update.yml** playbook will fail early, if the container image does not meet the Ceph version requirement.

CHAPTER 4. BUG FIXES

This section describes bugs with significant user impact, which were fixed in this release of Red Hat Ceph Storage. In addition, the section includes descriptions of fixed known issues found in previous versions.

4.1. THE CEPHADM UTILITY

The **ceph-volume** commands do not block OSDs and devices and runs as expected

Previously, the **ceph-volume** commands like **ceph-volume lvm list** and **ceph-volume inventory** were not completed thereby preventing the execution of other **ceph-volume** commands for creating OSDs, listing devices, and listing OSDs.

With this update, the default output of these commands are not added to the Cephadm log resulting in completion of all **ceph-volume** commands run in a container launched by the cephadm binary.

(BZ#1948717)

Searching Ceph OSD id claim matches a host's fully-qualified domain name to a host name

Previously, when replacing a failed Ceph OSD, the name in the CRUSH map appeared only as a host name, and searching for the Ceph OSD id claim was using the fully-qualified domain name (FQDN) instead. As a result, the Ceph OSD id claim was not found. With this release, the Ceph OSD id claim search functionality correctly matches a FQDN to a host name, and replacing the Ceph OSD works as expected.

(BZ#1954503)

The **ceph orch ls** command correctly displays the number of daemons running for a given service

Previously, the **ceph orch ls --service-type *SERVICE_TYPE*** command incorrectly reported 0 daemons running for a service that had running daemons, and users were unable to see how many daemons were running for a specific service. With this release, the **ceph orch ls --service-type *SERVICE_TYPE*** command now correctly displays how many daemons are running for that given service.

(BZ#1964951)

Users are no longer able to remove the Ceph Manager service using **cephadm**

Previously, if a user ran a **ceph orch rm mgr** command, it would cause **cephadm** to remove all the Ceph Manager daemons in the storage cluster, making the storage cluster inaccessible.

With this release, attempting to remove the Ceph Manager, a Ceph Monitor, or a Ceph OSD service using the **ceph orch rm *SERVICE_NAME*** command displays a warning message stating that it is not safe to remove these services, and results in no actions taken.

(BZ#1976820)

The **node-exporter** and **alert-manager** container versions have been updated

Previously, the Red Hat Ceph Storage 5.0 **node-exporter** and **alert-manager** container versions defaulted to version 4.5, when version 4.6 was available, and in use in Red Hat Ceph Storage 4.2.

With this release, using the **cephadm** command to upgrade from Red Hat Ceph Storage 5.0 to Red Hat Ceph Storage 5.0z1 results in the **node-exporter** and **alert-manager** container versions being updated to version 4.6.

(BZ#1996090)

4.2. CEPH DASHBOARD

Secure cookie-based sessions are enabled for accessing the Red Hat Ceph Storage Dashboard

Previously, storing information in LocalStorage made the Red Hat Ceph Storage dashboard accessible to all sessions running in a browser, making the dashboard vulnerable to XSS attacks. With this release, LocalStorage is replaced with secure cookie-based sessions and thereby the session secret is available only to the current browser instance.

(BZ#1889435)

4.3. CEPH FILE SYSTEM

The MDS daemon no longer crashes when receiving unsupported metrics

Previously, the MDS daemon could not handle the new metrics from the kernel client causing the MDS daemons to crash on receiving any unsupported metrics.

With this release, the MDS discards any unsupported metrics and works as expected.

(BZ#2030451)

Deletion of data is allowed when the storage cluster is full

Previously, when the storage cluster was full, the Ceph Manager hung on checking pool permissions while reading the configuration file. The Ceph Metadata Server (MDS) did not allow write operations to occur when the Ceph OSD was full, resulting in an **ENOSPACE** error. When the storage cluster hit full ratio, users could not delete data to free space using the Ceph Manager volume plugin.

With this release, the new FULL capability is introduced. With the FULL capability, the Ceph Manager bypasses the Ceph OSD full check. The **client_check_pool_permission** option is disabled by default whereas, in previous releases, it was enabled. With the Ceph Manager having FULL capabilities, the MDS no longer blocks Ceph Manager calls. This results in allowing the Ceph Manager to free up space by deleting subvolumes and snapshots when a storage cluster is full.

(BZ#1910272)

Ceph monitors no longer crash when processing authentication requests from Ceph File System clients

Previously, if a client did not have permission to view a legacy file system, the Ceph monitors would crash when processing authentication requests from clients. This caused the Ceph monitors to become unavailable. With this release, the code update fixes the handling of legacy file system authentication requests and authentication requests work as expected.

(BZ#1976915)

Fixes KeyError appearing every few milliseconds in the MGR log

Previously, **KeyError** was logged to the Ceph Manager log every few milliseconds. This was due to an

attempt to remove an element from **client_metadata[in_progress]** dictionary with a non-existent key, resulting in a **KeyError**. As a result, locating other stack traces in the logs was difficult. This release fixes the code logic in the Ceph File System performance metrics and **KeyError** messages in the Ceph Manager log.

(BZ#1979520)

Deleting a subvolume clone is no longer allowed for certain clone states

Previously, if you tried to remove a subvolume clone with the force option when the clone was not in a **COMPLETED** or **CANCELLED** state, the clone was not removed from the index tracking the ongoing clones. This caused the corresponding cloner thread to retry the cloning indefinitely, eventually resulting in an **ENOENT** failure. With the default number of cloner threads set to four, attempts to delete four clones resulted in all four threads entering a blocked state allowing none of the pending clones to complete.

With this release, unless a clone is either in a **COMPLETED** or **CANCELLED** state, it is not removed. The cloner threads no longer block because the clones are deleted, along with their entry from the index tracking the ongoing clones. As a result, pending clones continue to complete as expected.

(BZ#1980920)

The ceph fs snapshot mirror daemon status command no longer requires a file system name

Previously, users were required to give at least one file system name to the **ceph fs snapshot mirror daemon status** command. With this release, the user no longer needs to specify a file system name as a command argument, and daemon status displays each file system separately.

(BZ#1988338)

Stopping the cephfs-mirror daemon can result in an unclean shutdown

Previously, the **cephfs-mirror** process would terminate uncleanly due to a race condition during **cephfs-mirror** shutdown process. With this release, the race condition was resolved, and as a result, the **cephfs-mirror** daemon shuts down gracefully.

(BZ#2002140)

The Ceph Metadata Server no longer falsely reports metadata damage, and failure warnings

Previously, the Ceph Monitor assigned a rank to standby-replay daemons during creation. This behavior can lead to the Ceph Metadata Servers (MDS) reporting false metadata damage, and failure warnings. With this release, Ceph Monitors no longer assign rank to standby-replay daemons during creation, eliminating false metadata damage, and failure warnings.

(BZ#2002398)

4.4. CEPH MANAGER PLUGINS

The pg_autoscaler module no longer reports failed op error

Previously, the **pg-autoscaler** module reported KeyError for **op** when trying to get the pool status if any pool had the CRUSH rule **step set_chooseleaf_vary_r 1**. As a result, the Ceph cluster health displayed HEALTH_ERR with **Module 'pg_autoscaler' has failed: op** error. With this release,only steps with **op** are iterated for a CRUSH rule while getting the pool status and the **pg_autoscaler** module no longer reports the failed **op** error.

(BZ#1874866)

4.5. CEPH OBJECT GATEWAY

S3 lifecycle expiration header feature identifies the objects as expected

Previously, some objects without a lifecycle expiration were incorrectly identified in GET or HEAD requests as having a lifecycle expiration due to an error in the logic of the feature when comparing object names to stored lifecycle policy. With this update, the S3 lifecycle expiration header feature works as expected and identifies the objects correctly.

(BZ#1786226)

The radosgw-admin user list command no longer takes a long time to execute in Red Hat Ceph Storage cluster 4

Previously, in Red Hat Ceph Storage cluster 4, the performance of many **radosgw-admin** commands were affected because the value of **rgw_gc_max_objs** config variable ,which controls the number of GC shards, was increased significantly. This included **radosgw-admin** commands that were not related to GC. With this release, after an upgrade from Red Hat Ceph Storage cluster 3 to Red Hat Ceph Storage cluster 4 , the **radosgw-admin user list** command does not take a longer time to execute. Only the performance of **radosgw-admin** commands that require GC to operate is affected by the value of the **rgw_gc_max_objs** configuration.

(BZ#1927940)

Policies with invalid Amazon resource name elements no longer lead to privilege escalations

Previously, incorrect handling of invalid Amazon resource name (ARN) elements in IAM policy documents, such as bucket policies, can cause unintentional permissions granted to users who are not part of the policy. With this release, this fix prevents storing policies with invalid ARN elements, or if already stored, correctly evaluates the policies.

(BZ#2007451)

4.6. RADOS

Setting bluestore_cache_trim_max_skip_pinned to 10000 enables trimming of the object's metadata

The least recently used (LRU) cache is used for the object's metadata. Trimming of the cache is done from the least recently accessed objects. Objects that are pinned are exempted from eviction, which means they are still being used by Bluestore..

Previously, the configuration variable **bluestore_cache_trim_max_skip_pinned** controlled how many pinned objects were visited, thereby the scrubbing process caused objects to be pinned for a long time. When the number of objects pinned on the bottom of the LRU metadata cache became larger than **bluestore_cache_trim_max_skip_pinned** , then trimming of cache was not completed.

With this release, you can set **bluestore_cache_trim_max_skip_pinned** to **10000** which is larger than the possible count of metadata cache. This enables trimming and the metadata cache size adheres to the configuration settings.

(BZ#1931504)

Upgrading storage cluster from Red Hat Ceph Storage 4 to 5 completes with HEALTH_WARN state

When upgrading a Red Hat Ceph Storage cluster from a previously supported version to Red Hat Ceph Storage 5, the upgrade completes with the storage cluster in a HEALTH_WARN state stating that monitors are allowing insecure **global_id** reclaim. This is due to a patched CVE, the details of which are available in the *CVE-2021-20288*.

Recommendations to mute health warnings:

1. Identify clients that are not updated by checking the **ceph health detail** output for the **AUTH_INSECURE_GLOBAL_ID_RECLAIM** alert.
2. Upgrade all clients to Red Hat Ceph Storage 5.0 release.
3. If all the clients are not upgraded immediately, mute health alerts temporarily:

   Syntax

   ```
   ceph health mute AUTH_INSECURE_GLOBAL_ID_RECLAIM 1w  # 1 week
   ceph health mute AUTH_INSECURE_GLOBAL_ID_RECLAIM_ALLOWED 1w  # 1 week
   ```

4. After validating all clients have been updated and the *AUTH_INSECURE_GLOBAL_ID_RECLAIM* alert is no longer present for a client, set **auth_allow_insecure_global_id_reclaim** to **false**

   Syntax

   ```
   ceph config set mon auth_allow_insecure_global_id_reclaim false
   ```

5. Ensure that no clients are listed with the **AUTH_INSECURE_GLOBAL_ID_RECLAIM** alert.

(BZ#1953494)

The trigger condition for RocksDB flush and compactions works as expected

BlueStore organizes data into chunks called blobs, the size of which is 64K by default. For large writes, it is split into a sequence of 64K blob writes.

Previously, when the deferred size was equal to or more than the blob size, all the data was deferred and they were placed under the “L” column family. A typical example is the case for HDD configuration where the value is 64K for both **bluestore_prefer_deferred_size_hdd** and **bluestore_max_blob_size_hdd** parameters. This consumed the “L” column faster resulting in the RocksDB flush count and the compactions becoming more frequent. The trigger condition for this scenario was **data size in blob ⇐ minimum deferred size**.

With this release, the deferred trigger condition checks the size of extents on disks and not blobs. Extents smaller than **deferred_size** go to a deferred mechanism and larger extents are written to the disk immediately. The trigger condition is changed to **data size in extent < minimum deferred size**.

The small writes are placed under the “L” column and the growth of this column is slow with no extra compactions.

The **bluestore_prefer_deferred_size** parameter controls the deferred without any interference from the blob size and works as per it’s description of “writes smaller than this size”.

(BZ#1991677)

The Ceph Manager no longer crashes during large increases to pg_num and pgp_num

Previously, the code that adjusts placement groups did not handle large increases to **pg_num** and **pgp_num** parameters correctly, and led to an integer underflow that can crash the Ceph Manager.

With this release, the code that adjusts placement groups was fixed. As a result, large increases to placement groups do not cause the Ceph Manager to crash.

(BZ#2001152)

4.7. RADOS BLOCK DEVICES (RBD)

The librbd code honors the CEPH_OSD_FLAG_FULL_TRY flag

Previously, you could set the **CEPH_OSD_FLAG_FULL_TRY** with the **rados_set_pool_full_try()** API function. In Red Hat Ceph Storage 5, **librbd** stopped honoring this flag. This resulted in write operations stalling on waiting for space when a pool became full or reached a quota limit, even if the **CEPH_OSD_FLAG_FULL_TRY** was set.

With this release, **librbd** now honors the **CEPH_OSD_FLAG_FULL_TRY** flag, and when set, and a pool becomes full or reaches quota, the write operations either succeed or fail with **ENOSPC** or **EDQUOT** message. The ability to remove RADOS Block Device (RBD) images from a full or at-quota pool is restored.

(BZ#1969301)

4.8. RBD MIRRORING

Improvements to the rbd mirror pool peer bootstrap import command

Previously, running the **rbd mirror pool peer bootstrap import** command caused **librados** to log errors about a missing key ring file in cases where a key ring was not required. This can confuse site administrators, because it appears as though the command failed due to a missing key ring. With this release, **librados** no longer log errors in cases where a remote storage cluster's key ring is not required, such as when the bootstrap token contains the key.

(BZ#1981186)

4.9. ISCSI GATEWAY

The gwcli tool now shows the correct erasure coded pool profile

Previously, the **gwcli** tool would show the incorrect **k+m** values of the erasure coded pool.

With this release, the **gwcli** tool pulls the information from the erasure coded pool settings from the associated erasure coded profile and the Red Hat Ceph Storage cluster shows the correct erasure coded pool profile.

(BZ#1840721)

The upgrade of the storage cluster with iSCSI configured now works as expected

Previously, the upgrade of the storage cluster with iSCSI configured would fail as the latest ceph-iscsi packages would not have the ceph-iscsi-tools packages that were deprecated.

With this release, the **ceph-iscsi-tools** package is marked as obsolete in the RPM specification file and the upgrade succeeds as expected.

(BZ#2026582)

The tcmu-runner no longer fails to remove “blocklist” entries

Previously, the **tcmu-runner** would execute incorrect commands to remove the “blocklist” entries resulting in a degradation in performance for iSCSI LUNs.

With this release, the **tcmu-runner** was updated to execute the correct command when removing blocklist entries. The blocklist entries are cleaned up by **tcmu-runner** and the iSCSI LUNs work as expected.

(BZ#2041127)

The tcmu-runner process now closes normally

Previously, the **tcmu-runner** process incorrectly handled a failed path, causing the release of uninitialized **g_object** memory. This can cause the **tcmu-runner** process to terminate unexpectedly. The source code has been modified to skip the release of uninitialized **g_object** memory, resulting in the **tcmu-runner** process exiting normally.

(BZ#2007683)

The RADOS Block Device handler correctly parses configuration strings

Previously, the RADOS Block Device (RBD) handler used the **strtok()** function while parsing configuration strings, which is not thread-safe. This caused incorrect parsing of the configuration string of image names when creating or reopening an image. This resulted in the image failing to open. With this release, the RBD handler uses the thread-safe **strtok_r()** function, allowing for the correct parsing of configuration strings.

(BZ#2007687)

4.10. THE CEPH ANSIBLE UTILITY

The cephadm-adopt playbook now enables the pool application on the pool when creating a new nfs-ganesha pool

Previously, when the **cephadm-adopt** playbook created a new **nfs-ganesha** pool, it did not enable the pool application on the pool. This resulted in a warning that one pool did not have the pool application enabled. With this update, the **cephadm-adopt** playbook sets the pool application on the created pool, and a warning after the adoption no longer occurs.

(BZ#1956840)

The **cephadm-adopt** playbook does not create default realms for multisite configuration

Previously, it was required for the **cephadm-adopt** playbook to create the default realms during the adoption process, even when there was no multisite configuration present.

With this release, the **cephadm-adopt** playbook does not enforce the creation of default realms when there is no multisite configuration deployed.

(BZ#1988404)

The Ceph Ansible **cephadm-adopt.yml** playbook can add nodes with a host's fully-qualified domain name

Previously, the task that adds nodes in **cephadm** using the Ceph Ansible **cephadm-adopt.yml** playbook, was using the short host name, and was not matching the current fully-qualified domain name (FQDN) of a node. As a result, the adoption playbook failed because no match to the FQDN host name was found.

With this release, the playbook uses the **ansible_nodename** fact instead of the **ansble_hostname** fact, allowing the adoption playbook to add nodes configured with a FQDN.

(BZ#1997083)

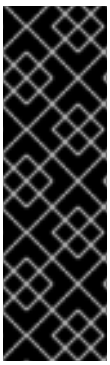
The Ceph Ansible **cephadm-adopt** playbook now pulls container images successfully

Previously, the Ceph Ansible **cephadm-adopt** playbook was not logging into the container registry on storage clusters that were being adopted. With this release, the Ceph Ansible **cephadm-adopt** playbook logs into the container registry, and pulls container images as expected.

(BZ#2000103)

CHAPTER 5. TECHNOLOGY PREVIEWS

This section provides an overview of Technology Preview features introduced or updated in this release of Red Hat Ceph Storage.



IMPORTANT

Technology Preview features are not supported with Red Hat production service level agreements (SLAs), might not be functionally complete, and Red Hat does not recommend using them for production. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process.

For more information on Red Hat Technology Preview features support scope, see the link:https:

- **Bucket granular multi-site replication [Technology Preview]**
  Previously, all buckets within a zone group were mirror copies that contained the same data. Multi-site data flow occurred within and between zones. With this release, bucket granular multi-site replication enables you to control the flow and replication of data at the bucket level. Buckets within a zone may contain different data, and can pull data from other buckets in other zones.

- **Document how to filter content with a query via a comma-separated values (CSV) file when retrieving data with S3 objects [Technology Preview]**
  The S3 Select Object Content API is now supported as a Technology Preview. This API filters the content of an S3 object through the structured query language (SQL). In the request you must specify the data serialization format that is comma-separated values (CSV) of the S3 object to retrieve the specified content. Aws CLI Select Object Content uses the CSV format to parse object data into records and returns only the records specified in the query.

5.1. CEPH OBJECT GATEWAY

Ceph object gateway in multisite replication setup now supports a subset of AWS bucket replication API functionality

With this release, Ceph Object Gateway now supports a subset of AWS bucket replication API functionality including {Put, Get, Delete} Replication operations. This feature enables bucket-granularity replication and additionally provides end-user replication control with the caveat that currently, buckets can be replicated within zones in an existing CephObject Gateway multisite replication setup.

Technology preview support for KMIP-protocol key management servers

With this release, technology preview support is available for KMIP-protocol key management servers like IBM SKLM thereby expanding the range of popular key management software used with Ceph object gateway's managed encryption feature.

5.2. RADOS BLOCK DEVICES (RBD)

librbd PMEM-based persistent write-back cache to reduce latency

With this release, the new **pwl_cache** librbd plugin provides a log-structured write-back cache targeted at PMEM devices thereby reducing the latency. The updates to the image are batched and flushed in-order, retaining the actual image in a crash-consistent state. If the PMEM device is lost, the image is still accessible though it may appear outdated.

Snapshot quiesce hook support for **rbd-nbd** devices

With this release, librbd API now offers quiesce and unquiesce hooks that enable coordinated snapshot creation. The **rbd-nbd** daemon optionally freezes and thaws the file system mounted on top of the mapped device to create file system consistent snapshots. This behavior can be customized by editing the **rbd-nbd_quiesce** shell script or by replacing it with a custom executable.

CHAPTER 6. KNOWN ISSUES

This section documents known issues found in this release of Red Hat Ceph Storage.

6.1. THE CEPHADM UTILITY

NFS-RGW issues in Red Hat Ceph Storage post-upgrade

It is recommended that customers using RGW-NFS defer their upgrade until Red Hat Ceph Storage 5.1.

(BZ#1842808)

The ceph orch host rm command does not remove the Ceph daemons in the host of a Red Hat Ceph Storage cluster

The **ceph orch host rm** command does not provide any output. This is expected behavior to avoid the accidental removal of Ceph daemons resulting in the loss of data.

To workaround this issue, the user has to remove the Ceph daemons manually. Follow the steps in the *Removing hosts using the Ceph Orchestrator* section in the *Red Hat Ceph Storage Operations Guide* for removing the hosts of the Red Hat Ceph Storage cluster.

(BZ#1886120)

The Ceph monitors are reported as stray daemons even after removal from the Red Hat Ceph Storage cluster

Cephadm reports the Ceph monitors as stray daemons even though they have been removed from the storage cluster.

To work around this issue, run the **ceph mgr fail** command, which allows the manager to restart and clear the error. If there is no standby manager, **ceph mgr fail** command makes the cluster temporarily unresponsive.

(BZ#1945272)

Access to the Cephadm shell is lost when monitor/s are moved to node/s without _admin label

After the bootstrap, access to the Cephadm shell is lost when the monitors are moved to other nodes if there is no **_admin** label. To workaround this issue, ensure that the destination hosts have the **_admin** label.

(BZ#1947497)

Upgrade of Red Hat Ceph Storage using Cephadm gets stuck if there are no standby MDS daemons

During an upgrade of a Red Hat Ceph Storage with an existing MDS service and with no active standby daemons, the process gets stuck.

To workaround this issue, ensure that you have at least one standby MDS daemon before an upgrade through Cephadm.

Run **ceph fs status *FILE_SYSTEM_NAME*** .

If there are no standby daemons, add MDS daemons and then upgrade the storage cluster. The upgrade works as expected when standby daemons are present.

(BZ#1959354)

The ceph orch ls command does not list the correct number of OSDs that can be created in the Red Hat Ceph Storage cluster

The command **ceph orch ls** gives the following output:

Example

```
# ceph orch ls

osd.all-available-devices    12/16  4m ago     4h   *
```

As per the above output, four OSDs have not started which is not correct.

To workaround this issue, run the **ceph -s** command to see if all the OSDs are up and running in a Red Hat Ceph Storage cluster.

(BZ#1959508)

The ceph orch osd rm help command gives an incorrect parameter description

The **ceph orch osd rm help** command gives **ceph orch osd rm *SVC_ID* ... [--replace] [--force]** parameter instead of **ceph orch osd rm *OSD_ID*... [--replace] [--force]**. This prompts the users to specify the ***SVC_ID*** while removing the OSDs.

To workaround this issue, use the OSD identification ***OSD_ID*** parameter to remove the OSDs of a Red Hat Ceph Storage cluster.

(BZ#1966608)

The configuration parameter osd_memory_target_autotune can be enabled

With this release, **osd_memory_target_autotune** is disabled by default. Users can enable OSD memory autotuning by running the following command:

```
ceph config set osd osd_memory_target_autotune true
```

(BZ#1939354)

6.2. CEPH DASHBOARD

Remove the services from the host before removing the hosts from the storage cluster on the Red Hat Ceph Storage Dashboard

Removing the hosts on the Red Hat Ceph Storage Dashboard before removing the services causes the hosts to be in a stale, dead, or a ghost state.

To workaround this issue, manually remove all the services running on the host and then remove the host from the storage cluster using the Red Hat Ceph Storage Dashboard. If you remove the host without removing the services, then to add the host again, you will have to use the command-line interface. If you remove the hosts without removing the services , you need to use the command-line interface to add the hosts again.

(BZ#1889976)

Users cannot create snapshots of subvolumes on the Red Hat Ceph Storage Dashboard

With this release, users cannot create snapshots of the subvolumes on the Red Hat Ceph Storage Dashboard. If the user creates a snapshot of the subvolumes on the dashboard, the user gets a 500 error instead of a more descriptive error message.

(BZ#1950644)

The Red Hat Ceph Storage Dashboard displays OSDs of only the default CRUSH root children

The Red Hat Ceph Storage Dashboard considers the default CRUSH root children ignoring other CRUSH types like datacenter, zones, rack, and other types. As a result, the CRUSH map viewer on the dashboard does not display OSDs which are not part of the default CRUSH root.

The tree view of OSDs of the storage cluster on the Ceph dashboard now resembles the **ceph osd tree** output.

(BZ#1953903)

Users cannot log in to the Red Hat Ceph Storage Dashboard with chrome extensions or plugins

Users cannot log into the Red Hat Ceph Storage Dashboard if there are Chrome extensions for the plugins used in the browser.

To work around this issue, either clear the cookies for a specific domain name in use or use the Incognito mode to access the Red Hat Ceph Storage Dashboard.

(BZ#1913580)

The graphs on the Red Hat Ceph Storage Dashboard are not displayed

The graphs on the Red Hat Ceph Storage Dashboard are not displayed because the grafana server certificate is not trusted on the client machine.

To workaround this issue, open the Grafana URL directly in the client internet browser and accept the security exception to see the graphs on the Ceph dashboard.

(BZ#1921092)

Incompatible approaches to manage NFS-Ganesha exports in a Red Hat Ceph Storage cluster

Currently, there are two different approaches to manage NFS-Ganesha exports in a Ceph cluster. One is using the dashboard and the other is using the command-line interface. If exports are created in one way, users might not be able to manage the exports in the other way.

To work around this issue, Red hat recommends to adhere to one way of deploying and managing the NFS thereby avoiding the potential duplication or management of non-modifiable NFS exports.

(BZ#1939480)

Dashboard related URL and Grafana API URL cannot be accessed with short hostnames

To workaround this issue, on the Red Hat Ceph Storage dashboard, in the *Cluster* drop-down menu, click Manager modules. Change the settings from short hostnames URL to FQDN URL. Disable the

dashboard using **ceph mgr module disable dashboard** command and re-enable the dashboard module using **ceph mgr module enable dashboard** command.

Dashboard should be able to access the Grafana API URL and the other dashboard URLs.

(BZ#1964323)

HA-Proxy-RGW service management is not supported on the Red Hat Ceph Storage Dashboard

The Red Hat Ceph Storage Dashboard does not support HA proxy service for Ceph Object Gateway.

As a workaround, HA proxy-RGW service can be managed using the Cephadm CLI. You can only view the service on the Red Hat Ceph Storage dashboard.

(BZ#1968397)

Red Hat does not support NFS exports over the Ceph File system in the back-end on the Red Hat Ceph Storage Dashboard

Red Hat does not support management of NFS exports over the Ceph File System (CephFS) on the Red Hat Ceph Storage Dashboard. Currently, NFS exports with Ceph object gateway in the back-end are supported.

(BZ#1974599)

6.3. CEPH FILE SYSTEM

Backtrace now works as expected for CephFS scrub operations

Previously, backtrace was unwritten to stable storage. Scrub activity reported a failure if the backtrace did not match the in-memory copy for a new and unsynced entry. Backtrace mismatch also happened for a stray entry that was about to be purged permanently since there was no need to save the backtrace to the disk. Due to the ongoing metadata I/O, it might have happened that the raw stats would not match if there was heavy metadata I/O because the raw stats accounting is not instantaneous.

To workaround this issue, rerun the scrub when the system is idle and has had enough time to flush in-memory state to disk. As a result, once the metadata has been flushed to the disk, these errors are resolved. Backtrace validation is successful if there is no backtrace found on the disk and the file is new, and the entry is stray and about to be purged.

See the KCS *Ceph status shows HEALTH_ERR with MDSs report damaged metadata* for more details.

(BZ#1794781)

NFS mounts are now accessible with multiple exports

Previously, when multiple CephFS exports were created, read/write to the exports would hang. As a result the NFS mounts were inaccessible. To workaround this issue, single exports are supported for Ganesha version 3.3-2 and below. With this release, multiple CephFS exports are supported when Ganesha version 3.3-3 and above is used.

(BZ#1909949)

The cephfs-top utility displays wrong mounts and missing metrics

The **cephfs-top** utility expects a newer kernel than what is currently shipped with Red Hat Enterprise Linux 8. The complete set of performance statistics patches are required by the **cephfs-top** utility. Currently, there is no workaround for this known issue.

(BZ#1946516)

6.4. CEPH OBJECT GATEWAY

The LC policy for a versioned bucket fails in between reshards

Currently, the LC policy fails to work, after suspending and enabling versioning on a versioned bucket, with reshards in between.

(BZ#1962575)

The **radosgw-admin user stats** command displays incorrect values for the **size_utilized** and **size_kb_utilized** fields

When a user runs the **radosgw-admin user stats** command after adding buckets to the Red Hat Ceph Storage cluster, the output displays incorrect values in the **size_utilized** and **size_kb_utilized** fields; they are always displayed as zero.

There is no workaround for this issue and users can ignore these values.

(BZ#1986160)

6.5. MULTI-SITE CEPH OBJECT GATEWAY

▌[5.0][rgw-multisite][Scale-testing][LC]: Deleting 16.5M objects via LC from the primary, does not delete the respective number of objects from secondary. |

TODO https://bugzilla.redhat.com/show_bug.cgi?id=1976874

▌[rgw-multisite][swift-cosbench]: Size in index not reliably updated on object overwrite, leading to ambiguity in stats on primary and secondary. |

TODO https://bugzilla.redhat.com/show_bug.cgi?id=1986826

6.6. THE CEPH ANSIBLE UTILITY

The rbd-mirroring does not work as expected after the upgrade from Red Hat Ceph Storage 4 to Red Hat Ceph Storage 5

The **cephadm-adopt** playbook does not bring up rbd-mirroring after the migration of the storage cluster from Red Hat Ceph Storage 4 to Red Hat Ceph Storage 5.

To work around this issue, add the peers manually:

Syntax

```
rbd mirror pool peer add POOL_NAME CLIENT_NAME@CLUSTER_NAME
```

Example

```
[ceph: root@host01 /]# rbd --cluster site-a mirror pool peer add image-pool client.rbd-mirror-peer@site-b
```

For more information, see the *Adding a storage cluster peer* section in the Red Hat Ceph Storage Block Device Guide.

(BZ#1967440)

The cephadm-adopt.yml playbook currently fails when the dashboard is enabled on the Grafana node

Currently, the **cephadm-adopt.yml** playbook fails to run as it does not create the **/etc/ceph** directory on nodes deployed only with a Ceph monitor.

To work around this issue, manually create the **/etc/ceph** directory on the Ceph monitor node before running the playbook. Verify the directory is owned by the **ceph** user's UID and GID.

(BZ#2029697)

6.7. KNOWN ISSUES WITH DOCUMENTATION

- **Documentation for users to manage Ceph File system snapshots on the Red Hat Ceph Storage Dashboard**
  Details for this feature will be included in the next version of the *Red Hat Ceph Storage Dashboard Guide*.

- **Documentation for users to manage hosts on the Red Hat Ceph Storage Dashboard**
  Details for this feature will be included in the next version of the *Red Hat Ceph Storage Dashboard Guide*.

- **Documentation for users to import RBD images instantaneously**
  Details for the **rbd import** command will be included in the next version of the *Red Hat Ceph Storage Block Device Guide*.

CHAPTER 7. DEPRECATED FUNCTIONALITY

This section provides an overview of functionality that has been deprecated in all minor releases up to this release of Red Hat Ceph Storage.

Ceph configuration file is now deprecated

The Ceph configuration file (**ceph.conf**) is now deprecated in favor of new centralized configuration stored in Ceph Monitors. For details, see the *The Ceph configuration database* section in the *Red Hat Ceph Storage Configuration Guide*.

The min_compat_client parameter for Ceph File System (CephFS) is now deprecated

The **min_compat_client** parameter is deprecated for Red Hat Ceph Storage 5.0 and new client features are added for setting-up the Ceph File Systems (CephFS). For details, see the *Client features* section in the *Red Hat Ceph Storage File System Guide*.

The snapshot of Ceph File System subvolume group is now deprecated

The snapshot feature of Ceph File System (CephFS) subvolume group is deprecated for Red Hat Ceph Storage 5.0. The existing snapshots can be listed and deleted, whenever needed. For details, see the *Listing snapshots of a file system subvolume group* and *Removing snapshots of a file system subvolume group* sections in the *Red Hat Ceph Storage Ceph File System guide*.

The Cockpit Ceph Installer is now deprecated

Installing a Red Hat Ceph Storage cluster 5 using Cockpit Ceph Installer is not supported. Use Cephadm to install a Red Hat Ceph Storage cluster. For details, see the *Red Hat Ceph Storage Installation guide*.

CHAPTER 8. SOURCES

The updated Red Hat Ceph Storage source code packages are available at the following location:

- For Red Hat Enterprise Linux 8:
  http://ftp.redhat.com/redhat/linux/enterprise/8Base/en/RHCEPH/SRPMS/