



Red Hat Ceph Storage 3.3

Release Notes

Release notes for Red Hat Ceph Storage 3.3

Red Hat Ceph Storage 3.3 Release Notes

Release notes for Red Hat Ceph Storage 3.3

Legal Notice

Copyright © 2020 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

The Release Notes document describes the major features and enhancements implemented in Red Hat Ceph Storage in a particular release. The document also includes known issues and bug fixes.

Table of Contents

CHAPTER 1. INTRODUCTION	3
CHAPTER 2. ACKNOWLEDGMENTS	4
CHAPTER 3. NEW FEATURES	5
3.1. THE CEPH-ANSIBLE UTILITY	5
3.2. CEPH MANAGEMENT DASHBOARD	5
3.3. CEPH MANAGER PLUGINS	6
3.4. THE CEPH-VOLUME UTILITY	6
3.5. OBJECT GATEWAY	6
3.6. PACKAGES	7
3.7. RADOS	7
CHAPTER 4. BUG FIXES	10
4.1. THE CEPH-ANSIBLE UTILITY	10
4.2. CEPH MANAGEMENT DASHBOARD	15
4.3. CEPH FILE SYSTEM	16
4.4. CEPH MANAGER PLUGINS	17
4.5. THE CEPH-VOLUME UTILITY	17
4.6. ISCSI GATEWAY	18
4.7. OBJECT GATEWAY	19
4.8. OBJECT GATEWAY MULTISITE	23
4.9. RADOS	24
4.10. BLOCK DEVICES (RBD)	25
CHAPTER 5. TECHNOLOGY PREVIEWS	26
5.1. BLOCK DEVICES (RBD)	26
5.2. CEPH FILE SYSTEM	26
5.3. OBJECT GATEWAY	26
CHAPTER 6. KNOWN ISSUES	27
6.1. CEPH MANAGEMENT DASHBOARD	27
6.2. OBJECT GATEWAY	27
6.3. RADOS	27
CHAPTER 7. DEPRECATED FUNCTIONALITY	28
7.1. THE CEPH-ANSIBLE UTILITY	28
CHAPTER 8. SOURCES	29

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/red-hat-ceph-storage/>.

CHAPTER 2. ACKNOWLEDGMENTS

Red Hat Ceph Storage version 3.3 contains many contributions from the Red Hat Ceph Storage team. Additionally, 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
- UbuntuKylin
- Mellanox
- CERN
- Deutsche Telekom
- Mirantis
- SanDisk
- SUSE

CHAPTER 3. NEW FEATURES

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

3.1. THE CEPH-ANSIBLE UTILITY

Setting ownership is faster when using `switch-from-non-containerized-to-containerized-ceph-daemons.yml`

Previously, the `chown` command in the `switch-from-non-containerized-to-containerized-ceph-daemons.yml` playbook unconditionally re-applied the ownership of Ceph directories and files causing a lot of write operations. With this update, the command has been improved to run faster. This is especially useful on a Red Hat Ceph Storage cluster with a significant amount of directories and files in the `/var/lib/ceph/` directory.

The new `device_class` Ansible configuration option

With the `device_class` feature, you can alleviate post deployment configuration by updating the `groups_vars/osd.yml` file in the desired layout. This feature offers you multi-backend support by avoiding to comment out sections after deploying Red Hat Ceph Storage.

Removing iSCSI targets using Ansible

Previously, the iSCSI targets had to be removed manually before purging the storage cluster. Starting with this release, the `ceph-ansible` playbooks remove the iSCSI targets as expected.

For bare-metal Ceph deployments, see the *Removing the Configuration* section in the the Red Hat Ceph Storage 3 [Block Device Guide](#) for more details.

For Ceph container deployment, see the Red Hat Ceph Storage 3 [Container Guide](#) for more details.

`osd_auto_discovery` now works with the `batch` subcommand

Previously, when `osd_auto_discovery` was activated, the `batch` subcommand did not create OSDs as expected. With this update, when `batch` is used with `osd_auto_discovery`, all the devices found by the `ceph-ansible` utility become OSDs and are passed in `batch` as expected.

The Ceph Ansible playbooks are compatible with Ansible 2.7

Starting with this release, users can install Ansible 2.7 and run the latest `ceph-ansible` playbooks for `{storage-product}`.

3.2. CEPH MANAGEMENT DASHBOARD

New options to use pre-downloaded container images

Previously, it was not possible to install Red Hat Ceph Storage Dashboard and the Prometheus plug-in without access to the Red Hat Container Registry. This update adds the following Ansible options that allow you to use pre-downloaded container images:

`prometheus.pull_image`

Set to `false` to not pull the Prometheus container image

`prometheus.trust_image_content`

Set to `true` to not contact the Registry for Prometheus container image verification

grafana.pull_image

Set to **false** to not pull the Dashboard container image

grafana.trust_image_content

Set to **true** to not contact the Registry for Dashboard container image verification

Set these options in the Ansible **group_vars/all.yml** file to use the pre-downloaded container images.

3.3. CEPH MANAGER PLUGINS

The RESTful plug-in now exposes performance counters

The RESTful plug-in for the Ceph Manager (**ceph-mgr**) now exposes performance counters that include a number of Ceph Object Gateway metrics. To query the performance counters through the REST API provided by the RESTful plug-in, access the **/perf** endpoint.

3.4. THE CEPH-VOLUME UTILITY

New ceph-volume subcommand: inventory

The **ceph-volume** utility now supports a new **inventory** subcommand. The subcommand describes every device in the system, reports if it is available or not and if it is used by the **ceph-disk** utility.

The ceph-volume tool can now set the sizing of journals and block.db

Previously, sizing for journals and block.db volumes could only be set in the **ceph.conf** file. With this update, the **ceph-volume** tool can set the sizing of journals and block.db. This exposes sizing right on the command line interface (CLI) so the user can use tools like **ceph-ansible** or the CLI directly to set or change sizing when creating an OSD.

New ceph-volume lvm zap options: --osd.id and --osd-fsid

The **ceph-volume lvm zap** command now supports the **--osd.id** and **--osd-fsid** options. Use these options to remove any devices for an OSD by providing its ID or FSID, respectively. This is especially useful if you are not aware of the actual device names or logical volumes in use by that OSD.

3.5. OBJECT GATEWAY

Renaming users is now supported

This update of Red Hat Ceph Storage adds the ability to rename the Ceph Object Gateway users. For details, see the *Rename a User* section in the Object Gateway Guide for [Red Hat Enterprise Linux](#) or for [Ubuntu](#).

The Ceph Object Gateway now supports the use of SSE-S3 headers

Clients and applications can successfully negotiate SSE-S3 encryption using the global, default encryption key, if one has been configured. Previously, the default key only used SSE-KMS encryption.

The x-amz-version-id header is now supported

The **x-amz-version-id** header is now returned by PUT operations on versioned buckets to conform to the S3 protocol. With this enhancement, clients now know the version ID of the objects they create.

New commands to view the RADOS objects and orphans

This release adds two new commands to view how Object Gateway maps to RADOS objects and

produce a potential list of orphans for further processing. The **radosgw-admin bucket radoslist --bucket=<bucket_name>** command lists all RADOS objects in the bucket. The **rgw-orphan-list** command lists all orphans in a specified pool. These commands keep intermediate results on the local file system.

Ability to associate one email address to multiple user accounts

This update adds the ability to create multiple Ceph Object Gateway (RGW) user accounts with the same email address.

Ability to search for users by access-key

This update adds the ability to search for users by the access-key as a search string when using the **radosgw-admin** utility:

```
radosgw-admin user info --access-key key
```

Keystone S3 credential caching has been implemented

The Keystone S3 credential caching feature permits using AWSv4 request signing (**AWS_HMAC_SHA256**) with Keystone as an authentication source, and accelerates Keystone authentication using S3. This also enables AWSv4 request signing, which increases client security.

3.6. PACKAGES

nfs-ganesha has been updated to the latest version

The **nfs-ganesha** package is now based on the upstream version 2.7.4, which provides a number of bug fixes and enhancements from the previous version.

3.7. RADOS

OSD BlueStore is now fully supported

BlueStore is a new back end for the OSD daemons that allows for storing objects directly on the block devices. Because BlueStore does not need any file system interface, it improves performance of Ceph Storage Clusters.

To learn more about the BlueStore OSD back end, see the [OSD BlueStore](#) chapter in the Administration Guide for Red Hat Ceph Storage 3.

A new configuration option: **osd_map_message_max_bytes**

The monitoring function can sometimes send messages via the Ceph File system kernel client to the cluster which are too large, causing a traffic problem. A configuration option named **osd_map_message_max_bytes** was added with a default value of 10MiB. This allows the cluster to respond in a more timely manner.

The default BlueStore and BlueFS allocator is now **bitmap**

Previously, the default allocator for BlueStore and BlueFS was the **stupid** allocator. This allocator spreads allocations over the entire device because it allocates the first extent it finds that is large enough, starting from the last place it allocated. The **stupid** allocator tracks each extent in a separate B-tree, so the amount of memory used depends on the number of extents. This behavior causes more fragmentation and requires more memory to track free space. With this update, the default allocator has been changed to **bitmap**. The **bitmap** allocator allocates based on the first extent possible from the

start of the disk, so large extents are preserved. It uses a fixed-size tree of bitmaps to track free space, thus using constant memory regardless of number of extents. As a result, the new allocator causes less fragmentation and requires less memory.

osdmapprool has a new option for the Ceph upmap balancer

The new **--upmap-active** option for **osdmapprool** command calculates and displays the number of rounds that the active balancer must complete to optimize all **upmap** items. The balancer completes one round per minute. The **upmap.out** file contains a line for each upmap item.

Example

```
$ ceph osd getmap > mymap
got osdmap epoch #####
$ osdmapprool --upmap upmap.out --upmap-active mymap
osdmapprool: osdmap file 'mymap'
writing upmap command output to: upmap.out
checking for upmap cleanups
upmap, max-count 10, max deviation 5
pools .....
....
prepared 0/10 changes
Time elapsed ##### secs
Unable to find further optimization, or distribution is already perfect
osd.0 pgs ###
osd.1 pgs ###
osd.2 pgs ###
....
Total time elapsed ##### secs, ## rounds
```

The ability to inspect BlueStore fragmentation

This update adds the ability to inspect fragmentation of the BlueStore back end. To do so, use the **ceph daemon** command or the **ceph-bluestore-tool** utility.

For details see the Red Hat Ceph Storage 3 [Administration Guide](#).

Updated the Ceph debug log to include the source IP address on failed incoming CRC messages

Previously, when a failed incoming Cyclic Redundancy Check (CRC) message was getting logged into the Ceph debug log, only a warning about the failed incoming CRC message was logged. With this release, the source IP address is added to this warning message. This helps system administrators identify which clients and daemons might have some networking issues.

New omap usage statistics per PG and OSD

This update adds a better reporting of **omap** data usage on a per placement group (PG) and per OSD level. PG-level data is gathered opportunistically during a deep scrub. Additional fields have been added to the output of the **ceph osd df** and various **ceph pg** commands to display the new values.

Listing RADOS objects in a specific PG

The **rados ls** command now accepts the **--pgid** option to list the RADOS objects in a specific placement group (PG).

PG IDs added to omap log messages

The **large omap** log messages now include placement group IDs to aid in locating the object.

The rocksdb_cache_size option default is now 512 MB

BlueStore OSD **rocksdb_cache_size** option default value has been changed to 512 MB to help with compaction.

The RocksDB compaction threads default value has changed

The new default value for the **max_background_compactions** option is **2**. As a result, this change improves performance for write heavy OMAP workloads. This option controls the number of concurrent background compaction threads. The old default value was **1**.

CHAPTER 4. BUG FIXES

This section describes bugs with significant impact on users that 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 CEPH-ANSIBLE UTILITY

It is now possible to use Ansible playbooks without copying them to the root **ceph-ansible** directory

Due to the missing **library** variable in the Ansible configuration, the custom Ansible modules were not detected when executed playbooks were present in the **infrastructure-playbooks** directory. Consequently, it was not possible to run the infrastructure playbooks without copying them into the root **ceph-ansible** directory. This update adds the **library** variable to the Ansible configuration. As a result, it is possible to use playbooks in the **infrastructure-playbooks** without copying them, for example:

```
# ansible-playbook infrastructure-playbooks/purge-cluster.yml -i inventory_file
```

([BZ#1668478](#))

The **purge-cluster.yml** playbook no longer fails when initiated a second time

The **purge-cluster.yml** playbook would fail if the **ceph-volume** binary was not present. Now the presence of the **ceph-volume** binary is checked, allowing for the **purge-cluster.yml** playbook to be initiated multiple times successfully.

([BZ#1722663](#))

An increase to the CPU allocation for containerized Ceph MDS deployments

Previously, for container-based deployments, the CPU allocation for the Ceph MDS daemons was set to **1** as the default. In some scenarios, this caused slow performance when compared to a bare-metal deployment. With this release, the Ceph MDS daemon CPU allocation default is **4**.

([BZ#1695850](#))

Redeploying OSDs using the same device name works as expected

Previously, the **shrink-osd.yml** playbook did not remove containers generated as part of the **prepare containers** task that were launched during initial development. As a consequence, an attempt to redeploy a container using the same device name failed, because the container was already present. The **shrink-osd.yml** playbook now properly removes containers generated as part of the **prepare containers** task, and redeploying OSDs using the same device name works as expected.

([BZ#1728132](#))

The BlueStore WAL and DB partitions are now only created when dedicated devices are specified for them

Previously, in containerized deployments using the **non-collocated** scenario, the BlueStore WAL partition was created by default on the same device as the BlueStore DB partition when it was not required. With this update, the **bluestore_wal_devices** variable is no longer set to **dedicated_devices** by default, and the BlueStore WAL partition is no longer created on the BlueStore DB device.

([BZ#1685253](#))

Ceph Ansible can configure RBD mirroring as expected

Previously, the configuration of RADOS Block Device (RBD) mirroring was incomplete and only available for non-containerized deployment. Consequently, the **ceph-ansible** utility was unable to configure RBD mirroring properly. The RBD mirroring configuration has been improved and has added support for containerized deployments. As a result, **ceph-ansible** can now configure the mirror pool mode and add the remote peer as expected on both deployments.

(BZ#1665877)

The **ceph-handler** script no longer restarts all OSDs regardless of if the **limit** parameter is provided

Previously, the **ceph-handler** script executed on all OSD nodes even if the **ceph-ansible limit** parameter was provided. This meant all OSDs were restarted, ignoring the **limit** parameter. With this update, the **ceph-handler** script only targets the OSD nodes included by the **limit** parameter, and the OSDs are restarted properly according to the **ceph-ansible limit** parameter.

(BZ#1535960)

Ansible first completes **configure_iscsi.yml** tasks and then starts the daemons

Previously, during a rolling update, the **ceph-ansible** utility started the Ceph iSCSI daemons and ran the **configure_iscsi.yml** playbook in parallel. Consequently, the daemon operations could conflict with the **configure_iscsi.yml** tasks that set up objects and the system could terminate unexpectedly due to the kernel being in an unsupported state. With this update, **ceph-ansible** first completes the `configure_iscsi.yml` tasks of creating iSCSI targets and then starts the daemons to avoid potential conflicts.`

(BZ#1795806)

Using custom repositories to install Red Hat Ceph Storage

Previously, using custom software repositories to install Ceph was disabled. Having a custom software repository can be useful for environments where Internet access is not allowed. With this release, the ability to use custom software repositories are enabled for Red Hat signed packages only. Custom third-party software repositories are not supported.

(BZ#1673254)

Ceph Ansible can now successfully activate OSDs that use NVMe devices

Due to an incorrect parsing of Non-volatile Memory Express (NVMe) drives, the **ceph-ansible** utility could not activate an OSD that used NVMe devices. This update fixes the parsing of the NVMe drives, and **ceph-ansible** can now successfully activate OSDs that use NVMe devices.

(BZ#1523464)

Rolling update works as expected.

Previously, when using **rolling_update.yml** to update the Red Hat Ceph Storage cluster, it could fail due to a Python module import failure. The error printed was **ERROR! Unexpected Exception, this is probably a bug: cannot import name to_bytes**. With this update, the correct import is used and no error occurs.

(BZ#1598763)

ceph-ansible now reports an error if an unsupported Ansible version is used

The **ceph-ansible** utility supports only Ansible versions from 2.3.x to 2.4.x. Previously, when Ansible version was higher than 2.4.x, the installation process failed with an error. With this update, **ceph-ansible** checks the Ansible version and reports an error if an unsupported Ansible version is used.

(BZ#1631563)

ceph_release is no longer automatically being reset to ceph_stable_release when ceph_repository is set to rhcs

Previously, **ceph_release** was automatically being reset to **ceph_stable_release** even when **ceph_repository** was set to **rhcs** in the **all.yml** file. **ceph_stable_release** is not needed when using the **rhcs** repository, and was being set to the automatic default value **dummy**. This caused the **allow multi mds** task to fail with the error **has no attribute** because **ceph_release_num** has no key **dummy**. With this update, **ceph_release** is no longer reset when **ceph_repository** is set to **rhcs**, and the task **allow multi mds** can be executed properly.

(BZ#1765230)

The shrink-osd.yml playbook removes partitions from NVMe disks in all situations

Previously, the Ansible playbook **infrastructure-playbooks/shrink-osd.yml** did not properly remove partitions on NVMe devices when used with the **osd_scenario: non-collocated** option in containerized environments. This bug has been fixed with this update, and the playbook removes the partitions as expected.

(BZ#1572933)

The ceph -w process is no longer running after canceling the command

The **ceph** aliases were not using the interactive session options for the docker commands in {storage-product} container environments. This was leaving a running **ceph** process, which was waiting on the user's input. With this release, the interactive session options, **-it**, have been added to the docker commands being referenced by the **ceph** aliases.

(BZ#1797874)

The shrink-osd.yml playbook stops OSD services as expected

A bug in the **shrink-osd.yml** playbook caused the **stopping osd service** task to attempt to connect to an incorrect node. Consequently, the task could not stop the OSD services properly. With this update, the bug has been fixed, and the playbook delegates the task on the correct node. As a result, OSD services are stopped properly.

(BZ#1686306)

Adding a new Ceph Manager node will no longer fail when using the Ansible limit option

Previously, adding a new Ceph Manager to an existing storage cluster when using the **limit** option would fail the Ansible playbook. With this release, you can now use the **limit** option when adding a new Ceph Manager and the newly generated keyring to be copied successfully.

(BZ#1552210)

The radosgw_address variable can be set to 0.0.0.0

Previously, the default value for **radosgw_address** was **0.0.0.0**. If you did not change the default value from **0.0.0.0**, then **ceph-ansible** would fail validation. However, this is a valid value for RADOS Gateway. With this update to Red Hat Ceph Storage, the default value was changed to **x.x.x.x**, so you can change the value to **0.0.0.0** and it will pass validation.

([BZ#1782743](#))

The **ceph-ansible** playbooks are no longer missing certain tags

Previously, the **ceph-ansible** playbooks were missing some tags, so running **ceph-ansible** with those specific tags was failing. With this update, the Ceph roles are tagged correctly in the **ceph-ansible** playbooks, and running **ceph-ansible** with those specific tags works as expected.

([BZ#1754432](#))

The **group_vars** files now correctly refer to RHCS 3.x instead of 2.x

Previously, the Red Hat Ceph Storage (RHCS) documentation URL and default value were referring to RHCS 2.x instead of 3.x. This meant deploying with the default value on baremetal using the CDN repositories would configure RHCS 2.x repositories instead of 3.x. The documentation in the configuration files were also referring to 2.x. With this update, the default RHCS version value and URL are now referring to RHCS 3, and there are no 2.x references.

([BZ#1702732](#))

- The playbook cannot shrink OSDs when using FQDN in inventory

When using FQDN in inventory, some tasks use the shortname returned by the Ceph OSD tree to add data to `hostvars[]`. This renders the playbook unable to shrink the OSDs.

The fix for this issue directs the tasks to use the `inventory_hostname` instead of the shortname.

([BZ#1779021](#))

Ansible removes the **chronyd** service after Ceph installation

The **chronyd** service is another implementation of the Network Time Protocol (NTP) and was enabled after rebooting from the initial installation. With this release, the **chronyd** service is disabled and the default NTP service is enabled.

([BZ#1651875](#))

Virtual IPv6 addresses are no longer configured for MON and RGW daemons

Previously, virtual IPv6 addresses could be configured in the Ceph configuration file for MON and RGW daemons because virtual IPv6 addresses are the first value present in the Ansible IPv6 address fact. The underlying code has been changed, and the last value in the Ansible IPv6 address fact is now used, and MON and RGW IPv6 configurations are set to the right value.

([BZ#1680155](#))

Faster OSD creation when deploying on containers

Previously, when creating an OSD in a container using the **lvm** OSD scenario, the container was allowed to set the number of open files to a value higher than the default host value. This behavior caused slower **ceph-volume** performance when compared to running **ceph-volume** on bare metal. With this release, the maximum number of open files is set to a lower value (**1024**) on the container during OSD creation. This results in faster OSD creation in container-based deployment.

([BZ#1702285](#))

The **rolling_update.yml** playbook now restarts **tcmu-runner** and **rbd-target-api**

Previously, the iSCSI gateway infrastructure playbooks, specifically **rolling_update.yml**, only restarted the **rbd-target-gw** daemon. With this update, the playbook also restarts the **tcmu-runner** and **rbd-target-api** daemons so the updated versions of those daemons are used.

([BZ#1659611](#))

Ceph Ansible can now successfully updates and restarts NFS Ganesha container when a custom suffix is used for the container name

Previously, the value set for the **ceph_nfs_service_suffix** variable was not considered when checking the status and version of the Ceph NFS Ganesha (**ceph-nfs**) container for restart or update. Consequently, the **ceph-nfs** container was not updated or restarted because the **ceph-ansible** utility could not determine that the container was running. With this update, **ceph-ansible** uses the value of **ceph_nfs_service_suffix** to determine the status of the **ceph-nfs** container. As a result, **ceph-nfs** container is successfully updated or restarted as expected.

([BZ#1750005](#))

The **purge-cluster.yml** playbook no longer causes issues with redeploying a cluster

Previously the **purge-cluster.yml** Ansible playbook did not clean all Red Hat Ceph Storage kernel threads as it should and could leave CephFS **mountpoint** mounted and Ceph Block Devices mapped. This could prevent redeploying a cluster. With this update, the **purge-cluster.yml** Ansible playbook cleans all Ceph kernel threads, unmounts all Ceph related **mountpoint** on client nodes, and unmaps Ceph Block Devices so the cluster can be redeployed.

([BZ#1337915](#))

Upgrading OSDs is no longer unresponsive for a long period of time

Previously, when using the **rolling_update.yml** playbook to upgrade an OSD, the playbook waited for the **active+clean** state. When data and **no of retry** count was large, the upgrading process became unresponsive for a long period of time because the playbook set the **noout** and **norebalance** flags instead of the **nodeep-scrub** flag. With this update, the playbook sets the correct flag, and the upgrading process is no longer unresponsive for a long period of time.

([BZ#1740463](#))

Ansible now enables the fragmentation flag when upgrading from Red Hat Ceph Storage 2 to 3

Previously, when upgrading from Red Hat Ceph Storage 2 to 3, the Ceph File System (CephFS) directories were not fragmented. With this update, the **ceph-ansible** utility enables the **allow_dirflag** flag that allows fragmentation during the upgrade.

([BZ#1776233](#))

Deploying NFS Ganesha gateway on Ubuntu IPv6 systems works as expected

When deploying NFS Ganesha gateway on Ubuntu IPv6 systems, the **ceph-ansible** utility failed to start the **nfs-ganesha** service. As a consequence, the installation process failed as well. This bug has been fixed, and the installation process proceeds as expected.

([BZ#1656908](#))

The **ceph-volume** execution time has been adjusted

On containerized deployment, the **ceph-volume** commands that were executed inside the OSD containers were taking more time than expected. Consequently, the OSD daemon could take several

minutes to start because **ceph-volume** was executed before the **ceph-osd** process. The value of the **ulimit nofile** variable has been adjusted on the OSD container process to reduce the execution time of the **ceph-volume** commands. As a result, the OSD daemon starts faster.

(BZ#1744390)

The value of **osd_memory_target** for HCI deployment is calculated properly

Previously, the calculation of the number of OSDs was not implemented for containerized deployment; the default value was **0**. Consequently, the calculation of the value of the BlueStore **osd_memory_target** option for Hyper-converged infrastructure (HCI) deployment was not correct. With this update, the number of OSDs is reported correctly for containerized deployment, and the value of **osd_memory_target** for the HCI configuration is calculated properly.

(BZ#1664112)

4.2. CEPH MANAGEMENT DASHBOARD

Alerts are sent to the Dashboard when the cluster status changes from **HEALTH_WARN** to **HEALTH_ERR**

Previously, when the cluster status changed from **HEALTH_WARN** to **HEALTH_ERR**, no alert was sent to the Dashboard Alert tab. With this update, sending alerts works as expected in the described scenario.

(BZ#1609381)

The Prometheus exporter port is now opened on all **ceph-mgr** nodes

Previously, the **ceph-mgr** playbook was not run on each **ceph-mgr** node, which meant the **ceph-mgr** Prometheus exporter port was not being opened on each node. With this update, the **ceph-mgr** playbook runs on all the **ceph-mgr** nodes, and the Prometheus exporter port is opened on all **ceph-mgr** nodes.

(BZ#1744549)

The dashboard can now be configured in a containerized cluster

Previously, in a containerized Ceph environment, the Red Hat Ceph Storage dashboard failed because the **cephmetric-ansible** playbook failed to populate the container name. With this update, the playbook populates the container name, and the dashboard can be configured as expected.

(BZ#1731919)

The *MDS Performance* dashboard now displays the correct number of CephFS clients

The *MDS Performance* dashboard displayed an incorrect value for *Clients* after increasing and decreasing the number of active Metadata Servers (MDS) and clients multiple times. This bug has been fixed, and the *MDS Performance* dashboard now displays the correct number of Ceph File System (CephFS) clients as expected.

(BZ#1652896)

The TCP port for the Ceph exporter is opened during the Ansible deployment of the Ceph Dashboard

Previously, the TCP port for the Ceph exporter was not opened by the Ansible deployment scripts on all the nodes in the storage cluster. Opening TCP port 9283 had to be done manually on all nodes for the metrics to be available to the Ceph Dashboard. With this release, the TCP port is now being opened by

the Ansible deployment scripts for Ceph Dashboard.

([BZ#1677269](#))

The Red Hat Ceph Storage Dashboard includes information for Disk IOPS and Disk Throughput as expected

The Red Hat Ceph Storage Dashboard did not show any data for Disk IOPS and Disk Throughput. This bug has been fixed, and the Dashboard includes information for Disk IOPS and Disk Throughput as expected.

([BZ#1753942](#))

No data alerts are no longer generated

The Red Hat Ceph Storage Dashboard generated a **No data** alert when a query returns no data. Previously, this alert sent an email to the administrator whenever there was a network outage or a node was down for maintenance. With this update, these **No data** alert are no longer generated.

([BZ#1663289](#))

4.3. CEPH FILE SYSTEM

The drop cache command completes as expected

Previously, when executing the administrative **drop cache** command, the Metadata Server (MDS) did not detect that the clients could not return more capabilities, and the command would not complete. With this update, the MDS now detects the clients cannot return any more capabilities, and the command completes.

([BZ#1685734](#))

The MDS no longer tries many log segments after restart

Previously, the Ceph Metadata Server (MDS) would sometimes try many log segments after restart. The MDS would then send too many OSD requests in a short period of time which could harm the Ceph cluster. This update limits the number of log segments, and the cluster is no longer harmed.

([BZ#1714814](#))

An issue with the `_lookup_parent()` function no longer causes `nfs-ganesha` to fail

Under certain circumstances, the `_lookup_parent()` function in the Red Hat Ceph Storage `userland` client libraries could return `0`, but not zero out the parent return pointer, which would remain uninitialized. Later, an assertion that the parent pointer be `NULL` would trip, and cause `nfs-ganesha` to fail. With this update, the error checking and return of `_lookup_parent()` has been refactored, and the situation is avoided.

([BZ#1715086](#))

A new ASOK command prevents server outages caused by client eviction

This release introduces a new `ceph-daemon mds.x session config <client_id> timeout <seconds>` ASOK command. Use this command to configure a timeout for individual clients to prevent or delay the client from getting evicted. This is especially useful to prevent server outages caused by client evictions.

([BZ#1729353](#))

Partially flushed **ESessions** log event no longer cause the MDS to fail

Previously, when a Ceph Metadata Server (MDS) had more than 1024 client sessions, sessions in the **ESessions** log event could get flushed partially. The journal replay code expects sessions in the **ESessions** log event to either be all flushed or not flushed at all, so this would cause the MDS to fail. With this update, the journal replay code can handle a partially flushed **ESessions** log event.

([BZ#1718135](#))

Heartbeat packets are reset as expected

Previously, the Ceph Metadata Server (MDS) did not reset heartbeat packets when it was busy in a large loop. This prevented the MDS from sending a beacon to the Monitor. With this update, the Monitor replaces the busy MDS, and the heartbeat packets are reset when the MDS is busy in a large loop.

([BZ#1714810](#))

4.4. CEPH MANAGER PLUGINS

The RESTful API `/osd` endpoint returns the full list of OSDs

Previously, the OSD traversal algorithm incorrectly handled data structures. As consequence, an internal server error was returned when listing OSDs by using the RESTful API `/osd` endpoint. With this update, the algorithm now properly traverses the OSD map, and the `/osd` endpoint returns the full list of OSDs as expected.

([BZ#1764919](#))

Using several `ceph-mgr` modules at the same time no longer causes random segmentation faults

Previously, random segmentation faults of the `ceph-mgr` daemon were occurring. This was because the shared memory in `ceph-mgr` Python modules was being accessed without proper locks, and the memory was not being dereferenced properly. For these cases, the locking mechanisms in `ceph-mgr` has been improved, and random segmentation faults when using several `ceph-mgr` modules at the same time no longer occurs.

([BZ#1717199](#))

Ceph-balancer status requests respond immediately

Previously, the status requests could become unresponsive due to CPU bound balance calculation. With this update, locks are released when they are not needed, and the CPU bound balance calculation has been fixed. As a result, status requests respond immediately.

([BZ#1761839](#))

4.5. THE CEPH-VOLUME UTILITY

`ceph-volume` now returns a more accurate error message when deploying OSDs on devices with GPT headers

The `ceph-volume` utility does not support deploying OSDs on devices with GUID Partition Table (GPT) headers. Previously, after attempting to do so, an error similar to the following one was returned:

```
Device /dev/sdb excluded by a filter
```

With this update, the **ceph-volume** utility returns a more accurate error message instructing the users to remove GPT headers:

GPT headers found, they must be removed on: \$device_name

([BZ#1644321](#))

ceph-volume can determine if a device is rotational or not even if the device is not in the /sys/block/ directory

If the device name did not exist in the **/sys/block/** directory, the **ceph-volume** utility could not acquire information on if a device was rotational or not. This was for example the case for loopback devices or devices listed in the **/dev/disk/by-path/** directory. Consequently, the **lvm batch** subcommand failed. With this update, **ceph-volume** uses the **lsblk** command to determine if a device is rotational if no information is found in **/sys/block/** for the given device. As a result, **lvm batch** works as expected in this case.

([BZ#1666822](#))

An error is now returned when the WAL and DB partitions are defined but not present

Due to a race condition, after restarting a Nonvolatile Memory Express (NVMe) device containing the WAL and DB devices, the symbolic links for WAL and DB were missing. Consequently, the NVMe node could not be mounted. The underlying source code has been modified to return an error if WAL or DB devices are defined but the symbolic links are missing on the system, which allows trying for up to 30 times at 5 second intervals and increasing the chances of finding the devices as the system boots.

([BZ#1719971](#))

4.6. ISCSI GATEWAY

The Ceph iSCSI gateway no longer fails to start when an RBD image cannot be found in a pool

During initialization, the **rbd-target-gw** daemon configures RBD images for use with the Ceph iSCSI gateway. The **rbd-target-gw** daemon did only a partial pool name match, potentially causing the incorrect pool to be used when opening an RBD image. As a consequence, the **rbd-target-gw** daemon failed to start. With this release, the **rbd-target-gw** daemon does a full pool name match, and the **rbd-target-gw** daemon starts as expected.

([BZ#1719772](#))

The rbd-target-gw service no longer fails to start when there are expired blacklist entries

When the **rbd-target-gw** service starts, it removes blacklist entries for the node. Previously, if a blacklist entry expired at the same time the daemon was removing it, the **rbd-target-gw** service would fail to detect the race and fail to start up. With this update, the **rbd-target-gw** service now checks for the error code indicating the blacklist entry no longer exists, ignores the error, and starts as expected.

([BZ#1732393](#))

Synchronization between ceph-ansible and the ceph-iscsi daemon

Prior to this update, when using the **ceph-ansible** utility, the **python-rtslib** back end device cache used by the **ceph-iscsi** daemons, and the kernel could become out of sync. Consequently, the **ceph-ansible** and **ceph-iscsi** daemon operations failed, and the daemons terminated unexpectedly. With this update,

the **ceph-iscsi** operations executed by **ceph-ansible** and the daemons that access the cache are forced to be updated. As a result, the daemons no longer fail in the described scenario.

(BZ#1785288)

The **rbd-target-api** service is started and stopped with respect to the **rbd-target-gw** service status

Previously, the **rbd-target-api** service did not start after starting the **rbd-target-gw** service. Consequently, the **rolling_update.yml** playbook stopped at **TASK [stop ceph iscsi services]**, and the updating process did not continue. With this update, the **rbd-target-api** service is started and stopped with respect to the **rbd-target-gw** service status, and the updating process works as expected.

(BZ#1670785)

4.7. OBJECT GATEWAY

A performance decrease when listing buckets with large object counts due to a regression was resolved

RADOS Gateway introduced a performance regression as a byproduct of changes in Red Hat Ceph Storage 3.2z2, which added support for multicharacter delimiters. This could cause S3 clients to time out. The regression has been fixed, restoring the original performance when listing buckets with large object counts. S3 clients no longer time out due to this issue.

(BZ#1717135)

Removing non-existent buckets from the reshard queue works as expected

When a bucket was added to the reshard queue and then it was deleted, an attempt to remove the bucket from the queue failed because the removal process tried to modify the bucket record, which did not exist. Additionally, during reshard processing, when a non-existent bucket was encountered on the queue, the reshard process stopped early and possibly never got to other buckets on the queue. This behavior kept happening because the reshard process is scheduled to run at a specified time interval. The underlying source code has been modified, and removing non-existent buckets from the reshard queue works as expected.

(BZ#1749124)

Ability to cancel resharding of tenanted buckets

Previously, it was not possible to cancel the resharding process of a tenanted bucket because the **radosgw-admin reshard cancel** did not support this scenario. With this update, a new **--tenant** option has been added, and it is now possible to cancel resharding of the tenanted buckets as expected.

(BZ#1756149)

The S3 client no longer times out when listing buckets with millions of objects

Previously, a change to the behavior of ordered bucket listing allowed support for multi-character delimiter searching, but this change did not include important listing optimizations. This caused a large performance loss. With this release, the logic controlling delimiter handling has been optimized, resulting in better performance.

(BZ#1718328)

Multi-character delimiter searches now take an expected amount of time to complete

Sometimes multi-character delimiter searches took an excessive amount of time. The logic has been corrected and now searches take an expected amount of time.

([BZ#1720741](#))

Getting the versioning state on a nonexistent bucket now returns an error

Previously, when getting the bucket version on a nonexistent bucket, the HTTP response was successful, for example:

'HTTPStatusCode': 200

Because the bucket does not exist, the correct HTTP response must be an error. With this release, when getting the bucket version on a nonexistent bucket, the Ceph Object Gateway code returns the following error:

ERR_NO_SUCH_BUCKET

([BZ#1705922](#))

The RADOS configuration URL is now able to read objects larger than 1000 bytes

The RADOS configuration URL was unable to read configuration objects greater than 1000 bytes because they were truncated. This behavior has been fixed and now larger objects are read properly.

([BZ#1725521](#))

All visible bucket index entries are listed as expected

Previously, interaction of legacy filtering rules with new sharded listing optimizations in the Object Gateway bucket listing code was incorrect. As a consequence, bucket listings could skip some of or even all entries in a bucket with a sharded index when multiple filtered entries, such as uncompleted multipart uploads, were present in the index. The iteration and filtering logic has been fixed, and all visible bucket index entries are listed as expected.

([BZ#1778217](#))

Swift object expiration is no longer effected by resharding

The Swift object expiration code was not compatible with bucket index resharding. This behavior could stall object expiration for the buckets. The Swift object expiration code has been updated to identify buckets using a tenant and bucket name. This update allows the removal of expired objects from an already resharded and stalled bucket. As a result, the object expiration is no longer effected by bucket index resharding.

([BZ#1703557](#))

Large or changed directories are now handled properly

Due to several underlying problems in the Ceph Object Gateway, the listing of very large directories could fail, and changed directories could become stale. With this update, the underlying problems have been fixed, allowing listing of large directories without failures, and reliable expiration of cached directory contents. Additionally, for the RADOS Gateway NFS interface, further changes were made allowing large directories to be listed at least 10 times faster than in Red Hat Ceph Storage 2.x.

([BZ#1708587](#))

Dynamic bucket index resharding no longer uses unnecessarily high system resources

Previously, during bucket index sharding, the code built a large JSON object even if it was not needed. During bucket listing, the Ceph Object Gateway requested too many entries from each bucket index. This behavior caused high CPU, memory, and network usage. Together, this caused the time for resharding to complete to be unnecessarily long. With this release, the large JSON object is only built if required and dynamic bucket index resharding only shards up to 2000 entries at a time. The default maximum can be overridden using a configuration option. With these changes Red Hat Ceph Storage uses less memory during resharding and ordered bucket listing is more efficient so it takes less time.

(BZ#1753588)

A new bucket life-cycle policy will overwrite the existing life-cycle policy

Because of an encoding error with the Ceph Object Gateway, storing a new bucket life-cycle policy on a bucket that already had an existing one would fail. Previously, working around the failure was done by deleting the old policy first, before storing the new one. With this release, this encoding error was fixed.

(BZ#1708650)

Enabling the `rgw_enable_ops_log` option would result in unbound memory growth

Previously, there was no process for consuming log entries, which lead to unbound memory growth for the Ceph Object Gateway. With this release, the process discards new messages when the number of outstanding messages in the data buffer exceeds a threshold, resulting in a smaller memory footprint.

(BZ#1708346)

Enabling the `enable_experimental_unrecoverable_data_corrupting_features` flag is no longer required when using the Beast web server

To use the Beast web server, it was required to enable the `enable_experimental_unrecoverable_data_corrupting_features` flag even though Beast was fully supported and not a Technology Preview anymore. With this update, enabling `enable_experimental_unrecoverable_data_corrupting_features` is no longer required to use Beast.

(BZ#1749754)

Space is no longer leaked when deleting objects via NFS

Previously, the Ceph Object Gateway NFS implementation incorrectly set a value used to construct a key subsequently used to set garbage collection (GC) on shadow objects. Deleting an object via NFS, as opposed to S3 or Swift, could cause space to be leaked. With this update, the GC tag is now set correctly and space is not leaked when deleting objects via NFS.

(BZ#1715946)

Ceph Object Gateway daemons no longer crash after upgrading to the latest version

Latest update to Red Hat Ceph Storage introduces a bug that caused Ceph Object Gateway daemons to terminate unexpectedly with a segmentation fault after upgrading to the latest version. The underlying source code has been fixed, and Ceph Object Gateway daemons work as expected after the upgrade.

(BZ#1766448)

Entries are now placed on the correct bucket index shard

Previously, certain objects in the sharded bucket index were in an incorrect shard because their hash source was set incorrectly. Consequently, entries could not be found when the correct shard was consulted in the sharded bucket index. The hash source has been set correctly for such objects, and

entries are now placed on the correct bucket index shard as expected.

([BZ#1766731](#))

Different life-cycle rules for different objects no longer display the same rule applied to all objects

The S3 life-cycle expiration tags are a key-value pair, such that a valid match must match both the key and the value. However, the Ceph Object Gateway only matched the key when computing the **x-amz-expiration** headers, causing tag rules with a common key, but different values, to match incorrectly. With this release, the key and value are both checked when matching a tag rules in the expiration header computation. As a result, objects are displayed with the correct tag rules.

([BZ#1731486](#))

Swift requests no longer cause the "HTTP/1.1 401 Unauthorized" error

Certain Swift requests with headers that contained non-strictly-compliant HTTP 1.1 line termination character in the "X-Auth-Token:" line were rejected with the "HTTP/1.1 401 Unauthorized" error. On Red Hat Ceph Storage version 2.5 those requests were processed despite their non-compliance. After upgrade to version 3.3 those requests began to return an error. With this update, the non-compliant line termination characters have been removed from the HTTP headers, and the aforementioned Swift requests no longer cause errors.

([BZ#1742993](#))

Bucket creation failed with a non-default location constraint

The default value is not set for the zone **api_name** option. This was causing the default zone group name to not be added properly, even when explicitly defining the zone group name. As a consequence, buckets could not be created with a non-default location constraint when referencing a non-default placement target. In this release, buckets can be created with a non-default location constraint when referencing a non-default placement target.

([BZ#1744766](#))

The clean-up process no longer fails after an aborted upload

When a multipart upload was aborted part way through, the clean-up process assumed some artifacts were present. If they were not present, it caused an error and the clean-up process stopped. The logic has been updated so if the artifacts are not present, the clean-up process still continues until it finishes.

([BZ#1722664](#))

Ceph Object Gateway no longer terminates when there are many open file descriptors

Previously, the Ceph Object Gateway with Beast front end terminated with an uncaught exception if there were many open file descriptors. With this update, the Ceph Object Gateway no longer terminates.

([BZ#1740668](#))

The Ceph Object Gateway returns the correct error code when accessing a S3 bucket

The Ceph Object Gateway authorization subsystem was changed in a previous release, and the LDAP error code for failed authentication was not updated. Because of this, the incorrect error code of **AccessDenied** was returned instead of **InvalidAccessKeyId** when trying to access a S3 bucket with non-existing credentials. With this release, the correct error code is returned when trying to access a S3 bucket with non-existing credentials.

(BZ#1721033)

Removing entries from the reshard log that refer to tenanted buckets is possible

Previously, when using **radosgw-admin** to remove a bucket from the reshard log, the tenant information was not passed down to the corresponding code, this limited the ability to remove entries from the reshard log that referenced tenanted buckets. With this update to Red Hat Ceph Storage, entries that reference tenanted buckets can be removed as expected.

(BZ#1794429)

Bucket resharding status is now displayed in plain language

Previously, the **radosgw-admin reshard status --bucket *bucket_name*** command used identifier-like tokens as follows to display the resharding status of a bucket:

- CLS_RGW_RESHARD_NONE
- CLS_RGW_RESHARD_IN_PROGRESS
- CLS_RGW_RESHARD_DONE

With this update, the command uses plain language to display the status:

- not-resharding
- in-progress
- done

(BZ#1639712)

4.8. OBJECT GATEWAY MULTISITE

The **radosgw-admin bucket sync status** command works for single-direction sync set up

Previously, if a zone was set up with the **sync_from_all** parameter set to **false**, the command **radosgw-admin bucket sync status** reported as "not in sync_from" because the underlying function expected a zone name instead of zone ID that was provided. The underlying source code has been modified, and **radosgw-admin bucket sync status** works as expected in the described situation.

(BZ#1713798)

Ceph Object Gateway multisite data sync issue is fixed by removing the filtering step from datalog processing

Previously, multisite data sync could get reported as behind one or more shards during a change, causing incorrect filtering of certain duplicate bucket names. With this fix, Ceph Object Gateway removes a filtering step from datalog processing that detects duplicate bucket names, and the data sync proceeds as expected in the described situation.

(BZ#1782508)

Bucket creation time remains consistent between zones in a multisite environment

Previously, a metadata sync in a multisite environment did not always update bucket creation time, and bucket creation times could become inconsistent between zones. With this update, the metadata sync now updates creation time even if the bucket already exists, and bucket creation time remains

consistent between zones.

(BZ#1702288)

radosgw-admin bucket rm --bypass-gc now stores timestamps for deletions

Previously, objects deleted with **radosgw-admin bucket rm --bypass-gc** did not store a timestamp for the deletion. Because of this, data sync did not apply these object deletions on other zones. With this update, proper timestamps are stored for deletions, and **bucket rm** with **--bypass-gc** correctly deletes objects on all zones.

(BZ#1599852)

The radosgw-admin bilog trim command now fully trims the bucket index log

Previously, the **radosgw-admin bilog trim** command only trimmed 1000 entries from the log, because only one OSD request was sent. With this release, the **radosgw-admin bilog trim** command now sends OSD requests in a loop until the bucket index log is completely trimmed.

(BZ#1713779)

Enhanced log trimming

Previously, the **radosgw-admin datalog trim** and **radosgw-admin mdlog trim** commands trimmed only 1000 entries. This was inconvenient when doing extended log trimming. With this update, the aforementioned commands loop until no log records are available to trim.

(BZ#1732101)

Ceph Object Gateway multisite zones of the same objects are in the same order

Previously, writing versions of the same object from different zones of Ceph Object Gateway multisite were in different orders after sync. With this correction, the versions are sorted, and all the zones are in order as expected.

(BZ#1779334)

4.9. RADOS

The Ceph Balancer now works with erasure-coded pools

The **maybe_remove_pg_upmaps** method is meant to cancel invalid placement group items done by the **upmap** balancer, but this method incorrectly canceled valid placement group items when using erasure-coded pools. This caused a utilization imbalance on the OSDs. With this release, the **maybe_remove_pg_upmaps** method is less aggressive and does not invalidate valid placement group items, and as a result, the **upmap** balancer works with erasure-coded pools.

(BZ#1715577)

ceph osd in any no longer marks permanently removed OSDs as in

Previously, running the **ceph osd in any** command on a Red Hat Ceph Storage cluster marked all historic OSDs that were once part of the cluster as **in**. With this update, **ceph osd in any** no longer marks permanently removed OSDs as **in**.

(BZ#1696691)

4.10. BLOCK DEVICES (RBD)

Operations against the RBD object map now utilize significantly less OSD CPU and I/O resources

The RADOS Block Device (RBD) object map support logic within the OSD daemons inefficiently handled object updates for multi-TiB RBD images. As a consequence, for such images, updating the RBD object map led to high CPU usage and unnecessary I/O within the OSDs. With this update, OSDs no longer pre-initialize the in-memory object map prior to reading the object map from disk. Additionally, now OSDs only perform read-modify-writes operations on portions of the object map Cyclic Redundancy Check (CRC) that are potentially affected by the updated state. As a result, operations against the RBD object map now utilize significantly less OSD CPU and I/O resources.

(BZ#1683751)

The Ceph v12 client (luminous) takes significantly longer to export an image to the Ceph cluster than do the Ceph v13 (mimic) or v14 (nautilus) clients on the same cluster

The **rbd export** command in the v12 client had a read queue depth of 1, which means that the command issued only one read request at a time to the cluster when exporting to STDOUT. The v12 client now supports up to 10 concurrent read requests to the cluster, resulting in a significant increase in speed.

(BZ#1794704)

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 to use 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 <https://access.redhat.com/support/offerings/techpreview/>.

5.1. BLOCK DEVICES (RBD)

Erasure Coding for Ceph Block Devices

Erasure coding for Ceph Block Devices is supported as a Technology Preview. For details, see the [Erasure Coding with Overwrites \(Technology Preview\)](#) section in the Storage Strategies Guide for Red Hat Ceph Storage 3.

5.2. CEPH FILE SYSTEM

Erasure Coding for Ceph File System

Erasure coding for Ceph File System is now supported as a Technology Preview. For details, see the [Creating Ceph File Systems with erasure coding](#) section in the Ceph File System Guide for Red Hat Ceph Storage 3.

5.3. OBJECT GATEWAY

Improved interoperability with S3 and Swift by using a unified tenant namespace

This enhancement allows buckets to be moved between tenants. It also allows buckets to be renamed.

In Red Hat Ceph Storage 2 the **rgw_keystone_implicit_tenants** option only applied to Swift. As of Red Hat Ceph Storage 3 this option applies to s3 also. Sites that used this feature with Red Hat Ceph Storage 2 now have outstanding data that depends on the old behavior. To accommodate that issue this enhancement also expands **rgw_keystone_implicit_tenants** so it can be set to any of "none", "all", "s3", or "swift".

For more information, see Bucket management in the [Object Gateway Guide for Red Hat Enterprise Linux](#) or [Object Gateway Guide for Ubuntu](#) depending on your distribution. The **rgw_keystone_implicit_tenants** setting is documented in the [Using Keystone to Authenticate Ceph Object Gateway Users](#) guide.

Ceph Object Gateway now supports Elasticsearch 5 and 6 APIs as a Technology Preview feature

Support has been added for using the Elasticsearch 5 and 6 application programming interfaces (APIs) with the Ceph Object Gateway.

CHAPTER 6. KNOWN ISSUES

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

6.1. CEPH MANAGEMENT DASHBOARD

The dashboard shows no data while the cluster is updating

Due to a known issue, the Red Hat Ceph Storage dashboard does not show any data while the cluster is updating.

([BZ#1731330](#))

6.2. OBJECT GATEWAY

Invalid bucket names

There are some S3 bucket names that are invalid in AWS, and therefore cannot be replicated by the Ceph Object Gateway multisite. For more information about these bucket names, see the [AWS documentation](#).

([BZ#1724106](#))

6.3. RADOS

Upgrading from Red Hat Ceph Storage 3 to 4 can cause Ceph Monitors to crash

Doing an upgrade from Red Hat Ceph Storage 3 to 4 can cause newer Ceph Monitors to send an incompatible message to older Ceph Monitors. This can cause older Ceph Monitors to crash while trying to read the incompatible message. To work around this issue, start all Ceph Monitors running Red Hat Ceph Storage 4. As a result, all Ceph Monitors will understand the new message.

([BZ#1872593](#))

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.

7.1. THE CEPH-ANSIBLE UTILITY

The `rgw_dns_name` parameter

The `rgw_dns_name` parameter is deprecated. Instead, configure the RADOS Gateway (RGW) zonegroup with the RGW DNS name. For more information, see: [Ceph - How to add hostnames in RGW zonegroup](#) in the Red Hat Customer Portal.

CHAPTER 8. SOURCES

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

- For Red Hat Enterprise Linux:
<http://ftp.redhat.com/redhat/linux/enterprise/7Server/en/RHCEPH/SRPMS/>
- For Ubuntu: <https://rhcs.download.redhat.com/ubuntu/>