Release notes for Red Hat Ceph Storage 6.1z1
Abstract

The release notes describes the major features, enhancements, known issues, and bug fixes implemented for the Red Hat Ceph Storage 6.1z1 product 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

CHAPTER 1. INTRODUCTION ......................................................................................... 3
CHAPTER 2. ACKNOWLEDGMENTS ............................................................................. 4
CHAPTER 3. NEW FEATURES ....................................................................................... 5
  3.1. THE CEPHADM UTILITY ................................................................................. 7
  3.2. CEPH FILE SYSTEM ......................................................................................... 8
  3.3. CEPH OBJECT GATEWAY .............................................................................. 8
  3.4. RADOS BLOCK DEVICES (RBD) ................................................................... 9
CHAPTER 4. BUG FIXES ............................................................................................. 10
  4.1. THE CEPHADM UTILITY ............................................................................... 10
  4.2. CEPH FILE SYSTEM ....................................................................................... 11
  4.3. CEPH MANAGER PLUGINS ......................................................................... 13
  4.4. RADOS ........................................................................................................... 13
  4.5. RADOS BLOCK DEVICES (RBD) ................................................................. 14
  4.6. THE CEPH VOLUME UTILITY ...................................................................... 14
CHAPTER 5. KNOWN ISSUES .................................................................................... 15
  5.1. MULTI-SITE CEPH OBJECT GATEWAY ..................................................... 15
CHAPTER 6. SOURCES ............................................................................................... 16
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.

CHAPTER 2. ACKNOWLEDGMENTS

Red Hat Ceph Storage version 6.1 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®
CHAPTER 3. NEW FEATURES

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

The below main features were added in Red Hat Ceph Storage 6.1:

Compression on-wire with msgr2 protocol is now available

With this release, in addition to encryption on wire, compression on wire is also supported to secure network operations within the storage cluster.

Python notifications are more efficient

Previously, there were some unused notifications that no modules needed at the moment. This caused inefficiency.

With this release, the NotifyType parameter is introduced. It is annotated, which events modules consume at the moment, for example NotifyType.mon_map, NotifyType.osd_map, and the like. As a consequence, only events that modules ask for are queued. The events that no modules consume are issued. Because of these changes, python notifications are now more efficient.

The changes to pg_num are limited

Previously, if drastic changes were made to pg_num that outpaced pgp_num, the user could hit the per-osd placement group limits and cause errors.

With this release, the changes to pg_num are limited to avoid the issue with per-osd placement group limits.

New pg_progress item is created to avoid dumping all placement group statistics for progress updates

Previously, the pg_dump item included unnecessary fields that wasted CPU if it was copied to python-land. This tended to lead to long ClusterState::lock hold times, leading to long ms_dispatch delays and generally slowing the processes.

With this release, a new pg_progress item is created to dump only the fields that mgr tasks or progress needs.

The mgr_ip is no longer re-fetched

Previously, the mgr_ip had to be re-fetched during the lifetime of an active Ceph manager module.

With this release, the mgr_ip does not change for the lifetime of an active Ceph manager module, thereby, there is no need to call back into Ceph Manager for re-fetching.

QoS in the Ceph OSD is based on the mClock algorithm, by default

Previously, the scheduler defaulted to the Weighted Priority Queue (WPQ). Quality of service (QoS) based on the mClock algorithm was in an experimental phase and was not yet recommended for production.

With this release, the mClock based operation queue enables QoS controls to be applied to Ceph OSD specific operations, such as client input and output (I/O) and recovery or backfill, as well as other background operations, such as pg scrub, snap trim, and pg deletion. The allocation of resources to each of the services is based on the input and output operations per second (IOPS) capacity of each Ceph OSD and is achieved using built-in mClock profiles.
Also, this release includes the following enhancements:

- Hands-off automated baseline performance measurements for the OSDs determine Ceph OSD IOPS capacity with safeguards to fallback to default capacity when an unrealistic measurement is detected.
- Setting sleep throttles for background tasks is eliminated.
- Higher default values for recoveries and max backfills options with the ability to override them using an override flag.
- Configuration sets using mClock profiles hide complexity of tuning mClock and Ceph parameters.

See *The mClock OSD scheduler* in *Red Hat Ceph Storage Administration Guide* for details.

**WORM compliance certification is now supported**

Red Hat now supports WORM compliance certification.

See the *Enabling object lock for S3* for more details.

**Set rate limits on users and buckets**

With this release, you can set rate limits on users and buckets based on the operations in a Red Hat Ceph Storage cluster. See the *Rate limits for ingesting data* for more details.

**librbd plugin named persistent write log cache to reduce latency**

With this release, the new *librbd* plugin named Persistent Write Log Cache (PWL) provides a persistent, fault-tolerant write-back cache targeted with SSD devices. It greatly reduces latency and also improves performance at low *io_depths*. This cache uses a log-ordered write-back design which maintains checkpoints internally, so that writes that get flushed back to the cluster are always crash consistent. Even if the client cache is lost entirely, the disk image is still consistent; but the data will appear to be stale.

**Ceph File System (CephFS) now supports high availability asynchronous replication for snapshots**

Previously, only one *cephfs-mirror* daemon would be deployed per storage cluster, thereby a CephFS supported only asynchronous replication of snapshots directories.

With this release, multiple *cephfs-mirror* daemons can be deployed on two or more nodes to achieve concurrency in snapshot synchronization, thereby providing high availability.

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

**BlueStore is upgraded to V3**

With this release, BlueStore object store is upgraded to V3. Following are the two features:

- The allocation metadata is removed from RocksDB and now performs a full de-stage of the allocator object with the OSD allocation.
- With cache age binning, older onodes might be assigned a lower priority than the hot workload data. See the *Ceph BlueStore* for more details.

**Use cephadm to manage operating system tuning profiles**
With this release, you can use **cephadm** to create and manage operating system tuning profiles for better performance of the Red Hat Ceph Storage cluster. See the *Managing operating system tuning profiles with cephadm* for more details.

**A direct upgrade from Red Hat Ceph Storage 5 to Red Hat Ceph Storage 7 will be available**

For upgrade planning awareness, directly upgrading Red Hat Ceph Storage 5 to Red Hat Ceph Storage 7 (N=2) will be available.

**The new cephfs-shell option is introduced to mount a filesystem by name**

Previously, cephfs-shell could only mount the default filesystem.

With this release, a CLI option is added in cephfs-shell that allows the mounting of a different filesystem by name, that is, something analogous to the `mds_namespace=` or `fs=` options for `kclient` and `ceph-fuse`.

**Day-2 tasks can now be performed through the Ceph Dashboard**

With this release, in the Ceph Dashboard, a user can perform every day-2 tasks that require daily or weekly frequency of actions. This enhancement improves the Dashboard’s assessment capabilities, customer experience, and strengthens its usability and maturity. In addition to this, new on-screen elements are also included to help and guide the user in retrieving additional information to complete a task.

### 3.1. THE CEPHADM UTILITY

**public_network parameter can now have configuration options, such as global or mon**

Previously, in **cephadm**, the `public_network` parameter was always set as a part of the `mon` configuration section during a cluster bootstrap without providing any configuration option to alter this behavior.

With this enhancement, you can specify the configuration options, such as `global` or `mon` for the `public_network` parameter during cluster bootstrap by utilizing the Ceph configuration file.

Bugzilla:2156919

**The Cephadm commands that are run on the host from the cephadm Manager module now have timeouts**

Previously, one of the Cephadm commands would occasionally hang indefinitely, and it was difficult for users to notice and sort the issue.

With this release, timeouts are introduced in the Cephadm commands that are run on the host from the Cephadm mgr module. Users are now alerted with a health warning about eventual failure if one of the commands hangs. The timeout is configurable with the `mgr/cephadm/default_cephadm_command_timeout` setting, and defaults to 900 seconds.

Bugzilla:2151908

**cephadm support for CA signed keys is implemented**

Previously, CA signed keys worked as a deployment setup in Red Hat Ceph Storage 5, although their working was accidental, untested, and broken in changes from Red Hat Ceph Storage 5 to Red Hat Ceph Storage 6.
With this enhancement, **cephadm** support for CA signed keys is implemented. Users can now use CA signed keys rather than typical pubkeys for SSH authentication scheme.

Bugzilla:2182941

### 3.2. CEPH FILE SYSTEM

**Switch the unfair Mutex lock to fair mutex**

Previously, the implementations of the Mutex, for example, `std::mutex` in C++, would not guarantee fairness and would not guarantee that the lock would be acquired by threads in the order called `lock()`. In most cases, this worked well but in an overloaded case, the client requests handling thread and `submit` thread would always successfully acquire the `submit_mutex` in a long time, causing `MDLog::trim()` to get stuck. That meant the MDS daemons would fill journal logs into the metadata pool, but could not trim the expired segments in time.

With this enhancement, the unfair Mutex lock is switched to fair mutex and all the `submit_mutex` waiters are woken up one by one in FIFO mode.

Bugzilla:2158304

### 3.3. CEPH OBJECT GATEWAY

**The bucket listing feature enables the rgw-restore-bucket-index tool to complete reindexing**

Previously, the `rgw-restore-bucket-index` tool would restore the bucket’s index partially until the next user listed out the bucket. Due to this, the bucket’s statistics would report incorrectly until the reindexing completed.

With this enhancement, the bucket listing feature is added which enables the tool to complete the reindexing and the bucket statistics are reported correctly. Additionally, a small change to the build process is added that would not affect end-users.

Bugzilla:2182456

**Lifecycle transition no longer fails for objects with modified metadata**

Previously, setting an ACL on an existing object would change its `mtime` due to which lifecycle transition failed for such objects.

With this fix, unless it is a copy operation, the object’s `mtime` remains unchanged while modifying just the object metadata, such as setting ACL or any other attributes.

Bugzilla:2213801

**Blocksize is changed to 4K**

Previously, Ceph Object Gateway GC processing would consume excessive time due to the use of a 1K blocksize that would consume the GC queue. This caused slower processing of large GC queues.

With this fix, blocksize is changed to 4K, which has accelerated the processing of large GC queues.

Bugzilla:2212446

**Users can now enable data transition to Azure**
With this enhancement, users can enable data transition to a remote cloud service, such as Azure, as part of the lifecycle configuration. See Transitioning data to Azure cloud service for more details.

Bugzilla:2181555

3.4. RADOS BLOCK DEVICES (RBD)

Object map for the snapshot accurately reflects the contents of the snapshot

Previously, due to an implementation defect, a stale snapshot context would be used when handling a write-like operation. Due to this, the object map for the snapshot was not guaranteed to accurately reflect the contents of the snapshot in case the snapshot was taken without quiescing the workload. In differential backup and snapshot-based mirroring, use cases with object-map and/or fast-diff features enabled, the destination image could get corrupted.

With this fix, the implementation defect is fixed and everything works as expected.

Bugzilla:2216186
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 CEPHADM UTILITY

The config or keyring files are no longer temporarily removed from the nodes

Previously, in cephadm, the calculation of where client config/keyrings files should be written would occasionally end up empty. Due to this, config/keyring files are temporarily removed from the nodes where they should be left alone until the next checks and the correct calculation.

With this fix, the timing of the calculation is altered and the keyring/conf files are no longer randomly removed from the nodes temporarily.

Bugzilla:2161545

Message about the limit policy is lowered to debug-level

Previously, cephadm would log about hitting the limit policy of an OSD specification at info-level. Due to this, every time the limit policy was hit, cephadm would check if any new OSDs should be deployed to match the specification, resulting in the logs to be spammed with messages about hitting the limit.

With this fix, the message about the limit policy is lowered to debug-level and the logs are no longer spammed with messages about limit policy unless they set the log level to debug.

Bugzilla:2207480

Error-handling is updated to work with the python version used in Red Hat Ceph Storage 6.1

Previously, a new timeout on cephadm commands added in Red Hat Ceph Storage 6.1 would not work correctly due to a python version difference affecting the type of error generated. Due to this, whenever one of these timeouts actually happened, the generated error was not handled and the cephadm module would crash.

With this fix, the error-handling is updated to work with the python version used in Red Hat Ceph Storage 6.1. When one of these timeouts occurs, it results in a health warning about what timed out, and the cephadm module no longer crashes.

Bugzilla:2209493

Setting or getting the global configuration works when using the ceph config get command

Previously, running the ceph config get command would not retrieve any value when the entity was set as global. Due to this, the module would fail because the module first tried to get the current value of the option for idempotency concern.

With this fix, checks are made by running the ceph config dump command and setting or getting the global configuration works.

Bugzilla:2190187

Special lines are no longer included in the host’s /etc/hosts file when mounting into the container
Previously, podman versions added a special line to the `/etc/hosts` file inside the container which messed with the host name resolution within the container, causing it to think that the FQDN of the current host is `host.containers.internal`.

With this fix, the host’s `/etc/hosts` file is mounted into the container without the special lines and users can use `/etc/hosts` for hostname resolution. Users will no longer encounter errors related to being unable to find the IP for `host.container.internal` while accessing Grafana graphs in the Ceph dashboard.

Bugzilla:2216295

### 4.2. CEPH FILE SYSTEM

**Directory permissions are correctly mirrored from source clusters to remote clusters**

Previously, `cephfs-mirror` would not update top-level directory permissions after the initial directory creation. Due to this, top-level directory permissions on remote clusters were not mirrored correctly when permissions were updated on source clusters.

With this fix, top-level directory permissions are synced on every snapshot sync and directory permissions are perfectly mirrored.

Bugzilla:2160542

**The specified path is validated when creating a Ceph File System export over NFS**

Previously, anything could be specified as a path, such as a file, symlink, or a non-existent path and it would not be validated while creating a Ceph File System export over NFS and the export would be successfully created. Due to this, mounting the export would fail with an error message.

With this fix, the specified path is validated when creating the export and the operator is informed with a proper error message.

Bugzilla:2170739

**The `fallocate` path clears the `suid/sgid` if an unprivileged user changes the file**

Previously, the `fallocate` path would not clear the `suid/sgid` if an unprivileged user changed the file. There is no Posix item that requires clearing the `suid/sgid` in `fallocate` path but this is the default behaviour for most of the filesystems and the VFS layer. So, the user space `libcephfs` client would not comply with most filesystems in the kernel and this could be easily hacked.

With this fix, the `fallocate` path clears the `suid/sgid` if an unprivileged user changes the file, making the user space `libcephfs` client comply with most other filesystems and fix the attack hole.

Bugzilla:2185710

**The recovered files under the `lost+found` directory can now be deleted in Ceph File System**

With this fix, after recovering a Ceph File System post the disaster recovery the recovered files under the `lost+found` directory can be deleted.

Bugzilla:2222231

**`snap-schedule` module no longer throws any traceback**

Previously, due to an incorrect subvolume path resolution, traceback was seen on command-line whenever a bad path was provided along with a subvolume argument.
With this fix, users are recommended to ignore the user-specified path for subvolumes and snap-schedule module does not throw any traceback on the command-line interface.

Bugzilla:2196748

MDS no longer crashes when allocating CInode

Previously, when replaying the journals, if the inodetable or the sessionmap versions did not match, the CNode would be added to the inode_map. But the ino may still be in the inodetable or sessions’ prealloc inos list. Due to this, when allocating a new ino number, if the corresponding CInode was already in the inode_map, the MDS would crash.

With this fix, allocating ino# is skipped when allocating the new CInode and the MDS does not crash.

Bugzilla:2189132

Creation of pool-level snaps for pools actively associated with a filesystem is disallowed

Previously, the ceph osd pool mksnap command allowed the creation of pool-level snaps for pools actively associated with a filesystem. Due to this, there would be possible data loss when snapshots were deleted from either the filesystem or the pool due to pool ID collision.

With this fix, creation of pool-level snaps for pools actively associated with a filesystem is disallowed and no data loss occurs.

Bugzilla:2189787

Link requests no longer fail with -EXDEV

Previously, if an inode had more than one link and after one of its dentries was unlinked, it would be moved to a stray directory. Before the link merge/migrate finished, if a link request came, it would fail with -EXDEV error. While in non-multiple link cases, it was possible that the clients could pass one invalidate ino, which is still under unlinking. Due to this, some link requests would fail directly.

With this fix, if users wait for the link merge, migrate or purge to finish, no link requests fail with -EXDEV.

Bugzilla:2196405

The minimum compatible python version for cephfs-top is 3.6.0

With this fix, the minimum compatible python version for cephfs-top is 3.6.0. It checks if the current python version is greater than or equal to 3.6.0 during the build, and thus ensures that the cephfs-top curses display launches successfully.

Bugzilla:2203165

Structure variables are no longer stale or unsafe when accessed after session reconnection

Previously, the Ceph File System user-space clients could access stale/unsafe structure variables when rebuilding a request and this would lead to the clients misbehaving sometimes after reconnecting to the Ceph Manager daemons while re-issuing requests.

With this fix, the structure variables are no longer stale or unsafe when accessed after session reconnection. This is ensured by deep-copying them instead of shallow-copying and the Ceph File System (CephFS) user-space clients work as expected.

Bugzilla:2203906

Calls to Ceph Manager daemons and volumes no longer return -ESHUTDOWN
Previously, calls to Ceph Manager daemons and volumes would return -ESHUTDOWN when the ceph-mgr process was shutting down, which was not necessary.

With this fix, Ceph Manager plugin handles the shutdown without returning a special error code and -ESHUTDOWN is never returned.

Bugzilla:2150306

4.3. CEPH MANAGER PLUGINS

A new Ceph manager module option, exclude_perf_counters, is introduced

Previously, after the introduction of the new Ceph exporter, metrics coming from Ceph daemon performance counters were exposed by Ceph exporter and Prometheus Manager module. Due to this, metrics based on Ceph daemon performance counters were duplicated.

With this fix, a new Ceph Manager module option, exclude_perf_counters, is introduced. By default, it is set to True, preventing the Prometheus Manager module from exporting the metrics coming from the performance counters.

Bugzilla:2186549

Python tasks no longer wait for the GIL

Previously, the Ceph manager daemons held the Python Global Interpreter Lock (GIL) during some RPCs. Due to this, other python tasks in the queue starved waiting for the GIL.

With this fix, the GIL is released during all libcephfs or librbd calls and other Python tasks can acquire the GIL normally.

Bugzilla:2219440

4.4. RADOS

QoS parameters are restored to built-in default values whenever an attempt is made to modify them

Previously, the modifications to the QoS parameters would not reset to the profile defaults because the OSD would not have permissions to remove the modified entry. Due to this, the QoS parameters of the built-in were shown as "modified" by the configuration subsystem even though the intended change would not come into effect. This gave erroneous information to the user.

With this fix, the required permissions for the OSD to remove the modified QoS parameters from the configuration store are enabled. The QoS parameters, such as reservation, weight, and limit for a built-in profile are restored to the built-in default values whenever an attempt is made to modify them.

Bugzilla:2124137

Users can check for accurate number of placement groups with the crush rule

Previously, the check_pg_num() function would not take into account the root OSDs used by the crush rule. This resulted in an inaccurate placement group number per OSD count.

With this fix, check_pg_num() counts the projected placement group number which are part of the pools affected by the crush rule. The same applies to the number of OSDs as well; instead of dividing the projected placement group number total by all the osdmap’s OSDs, it is divided only by the OSDs used by the crush rule.
Bugzilla:2155766

**Manager continues to send beacons in the event of an error during authentication check**

Previously, if an error was encountered when performing an authentication check with a monitor, the manager would get into a state where it would no longer have an active connection. Due to this, the manager could no longer send beacons and the monitor would mark it as lost.

With this fix, a session (active connection) is reopened in the event of an error and the manager is able to continue to send beacons and is no longer marked as lost.

Bugzilla:2171847

**MonClient no longer fails to authenticate with EAGAIN**

Previously, if MonClient failed to authenticate with EAGAIN, there was a possibility that it would reach a prohibited state in which it would not have an active connection to ceph-mon and would not try further to acquire one. Due to this, even though the Ceph Manager daemon was technically alive, it became invisible to monitors in the cluster.

With this fix, the authentication with EAGAIN is handled properly and it works as expected.

Bugzilla:2187258

**4.5. RADOS BLOCK DEVICES (RBD)**

**profile rbd-read-only OSD capability allows opening an image in read-only mode**

Previously, due to an implementation defect, the profile rbd-read-only OSD capability would disallow opening an image even in read-only mode. Due to this, a bogus "Operation not permitted" error was returned for images in custom namespaces, although images in a default namespace were not affected.

With this fix, the implementation defect is fixed and the profile rbd-read-only OSD capability allows opening an image in read-only mode regardless of the images in the namespaces.

Bugzilla:2209652

**4.6. THE CEPH VOLUME UTILITY**

**Devices already used by Ceph are filtered out in ceph-volume**

Previously, due to a bug, ceph-volume would not filter out devices already used by Ceph. Due to this, adding new OSDs with ceph-volume failed when using pre-created LVs.

With this fix, devices already used by Ceph are filtered out in ceph-volume as expected and new OSDs with pre-created LVs can now be added.

Bugzilla:2188246
CHAPTER 5. KNOWN ISSUES

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

5.1. MULTI-SITE CEPH OBJECT GATEWAY

Deleting objects in versioned buckets causes statistics mismatch

Due to versioned buckets having a mix of current and non-current objects, deleting objects might cause bucket and user statistics discrepancies on local and remote sites. This does not cause object leaks on either site, just statistics mismatch.

Bugzilla: 1871333

Multisite replication may stop during upgrade

Multisite replication may stop if clusters are on different versions during the process of an upgrade. We would need to suspend sync until both clusters are upgraded to the same version.

Bugzilla: 2178909
CHAPTER 6. SOURCES

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

- For Red Hat Enterprise Linux 8:

- For Red Hat Enterprise Linux 9: