Red Hat Ceph Storage 5.2

Release Notes

Release notes for Red Hat Ceph Storage 5.2

Last Updated: 2022-08-17

Red Hat Ceph Storage 5.2 Release Notes

Release notes for Red Hat Ceph Storage 5.2

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 5.2 release up to the current release.

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:

- For simple comments on specific passages, make sure you are viewing the documentation in the multi-page HTML format. Highlight the part of text that you want to comment on. Then, click the **Add Feedback** pop-up that appears below the highlighted text, and follow the displayed instructions.
- For submitting more complex feedback, create a Bugzilla ticket:
    1. Go to the *Bugzilla* website.
    2. In the Component drop-down, select **Documentation**
    3. Select the appropriate version of the document.
    4. Fill in the **Summary** and **Description** field with your suggestion for improvement. Include a link to the relevant part(s) of documentation.
    5. Optional: Add an attachment, if any.
    6. 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/red-hat-ceph-storage/.

CHAPTER 2. ACKNOWLEDGMENTS

Red Hat Ceph Storage 5 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.

3.1. THE CEPHADM UTILITY

The Cephadm-Ansible modules

The **Cephadm-Ansible** package provides several modules that wrap the new integrated control plane, **cephadm**, for those users that wish to manage their entire datacenter with Ansible. It does not intend to provide backward compatibility with **Ceph-Ansible**, but it aims to deliver a supported set of playbooks that customers can use to update their Ansible integration.

See *The **cephadm-ansible** modules* for more details.

Bootstrap Red Hat Ceph Storage cluster is supported on Red Hat Enterprise Linux 9

With this release, **cephadm** bootstrap is available on Red Hat Enterprise Linux 9 hosts to enable Red Hat Ceph Storage 5.2 support for Red Hat Enterprise Linux 9. Users can now bootstrap a Ceph cluster on Red Hat Enterprise Linux 9 hosts.

cephadm rm-cluster command cleans up the old systemd unit files from host

Previously, the **rm-cluster** command would tear down the daemons without removing the systemd unit files.

With this release, **cephadm rm-cluster** command , along with purging the daemons, cleans up the old systemd unit files as well from the host.

cephadm raises health warnings if it fails to apply a specification

Previously, failures to apply a specification would only be reported as a service event which the users would often not check.

With this release, **cephadm** raises health warnings if it fails to apply a specification, such as an incorrect pool name in an iscsi specification, to alert users.

Red Hat Ceph Storage 5.2 supports staggered upgrade

Starting with Red Hat Ceph Storage 5.2, you can selectively upgrade large Ceph clusters in **cephadm** in multiple smaller steps.

The **ceph orch upgrade start** command accepts the following parameters:

- **--daemon-types**
- **--hosts**
- **--services**
- **--limit**

These parameters selectively upgrade daemons that match the provided values.

NOTE

These parameters are rejected if they cause **cephadm** to upgrade daemons out of the supported order.

NOTE

These upgrade parameters are accepted if your active Ceph Manager daemon is on a Red Hat Ceph Storage 5.2 build. Upgrades to Red Hat Ceph Storage 5.2 from an earlier version does not support these parameters.

fs.aio-max-nr is set to 1048576 on hosts with OSDs

Previously, leaving **fs.aio-max-nr** as the default value of **65536** on hosts managed by **Cephadm** could cause some OSDs to crash.

With this release, **fs.aio-max-nr** is set to 1048576 on hosts with OSDs and OSDs no longer crash as a result of the value of **fs.aio-max-nr** parameter being too low.

ceph orch rm <service-name> command informs users if the service they attempted to remove exists or not

Previously, removing a service would always return a successful message, even for non-existent services causing confusion among users.

With this release, running **ceph orch rm *SERVICE_NAME*** command informs users if the service they attempted to remove exists or not in **cephadm**.

A new playbook rocksdb-resharding.yml for resharding procedure is now available in cephadm-ansible

Previously, the **rocksdb** resharding procedure entailed tedious manual steps.

With this release, the cephadm-ansible playbook, **rocksdb-resharding.yml** is implemented for enabling **rocksdb** resharding which makes the process easy.

cephadm now supports deploying OSDs without an LVM layer

With this release, to support users who do not want a LVM layer for their OSDs, **cephadm** or **ceph-volume** support is provided for raw OSDs. You can include "method: raw" in an OSD specification file passed to Cephadm, to deploy OSDs in raw mode through Cephadm without the LVM layer.

With this release, cephadm supports using method: raw in the OSD specification yaml file to deploy OSDs in raw mode without an LVM layer.

See *Deploying Ceph OSDs on specific devices and hosts* for more details.

3.2. CEPH DASHBOARD

Start, stop, restart, and redeploy actions can be performed on underlying daemons of services

Previously, orchestrator services could only be created, edited, and deleted. No action could be performed on the underlying daemons of the services

With this release, actions such as starting, stopping, restarting, and redeploying can be performed on the underlying daemons of orchestrator services.

OSD page and landing page on the Ceph Dashboard displays different colours in the usage bar of OSDs

Previously, whenever an OSD would reach near full or full status, the cluster health would change to WARN or ERROR status, but from the landing page, there was no other sign of failure.

With this release, when an OSD turns to near full ratio or full, the OSD page for that particular OSD, as well as the landing page, displays different colours in the usage bar.

Dashboard displays onode hit or miss counters

With this release, the dashboard provides details pulled from the Bluestore stats, to display the onode hit or miss counters to help you deduce whether increasing the RAM per OSD could help improve the cluster performance.

Users can view the CPU and memory usage of a particular daemon

With this release, you can see CPU and memory usage of a particular daemon on the Red Hat Ceph Storage Dashboard under Cluster > Host> Daemons.

Improved Ceph Dashboard features for rbd-mirroring

With this release, the RBD Mirroring tab on the Ceph Dashboard is enhanced with the following features that were previously present only in the command-line interface (CLI):

- Support for enabling or disabling mirroring in images.
- Support for promoting and demoting actions.
- Support for resyncing images.
- Improve visibility for editing site names and create bootstrap key.
- A blank page consisting a button to automatically create an rbd-mirror appears if none exist.

User can now create OSDs in simple and advanced mode on the Red Hat Ceph Storage Dashboard

With this release, to simplify OSD deployment for the clusters with simpler deployment scenarios, “Simple” and “Advanced” modes for OSD Creation are introduced.

You can now choose from three new options:

- Cost/Capacity-optimized: All the available HDDs are used to deploy OSDs.
- Throughput-optimized: HDDs are supported for data devices and SSDs for DB/WAL devices.
- IOPS-optimized: All the available NVMEs are used as data devices.

See *Management of OSDs using the Ceph Orchestrator* for more details.

Ceph Dashboard Login page displays customizable text

Corporate users want to ensure that anyone accessing their system is acknowledged and is committed to comply with the legal/security terms.

With this release, a placeholder is provided on the Ceph Dashboard login page, to display a customized banner or warning text. The Ceph Dashboard admin can set, edit, or delete the banner with the following commands:

Example

```
[ceph: root@host01 /]# ceph dashboard set-login-banner -i filename.yaml
[ceph: root@host01 /]# ceph dashboard get-login-banner
[ceph: root@host01 /]# ceph dashboard unset-login-banner
```

When enabled, the Dashboard login page displays a customizable text.

Major version number and internal Ceph version is displayed on the Ceph Dashboard

With this release, along with the major version number, the internal Ceph version is also displayed on the Ceph Dashboard, to help users relate Red Hat Ceph Storage downstream releases to Ceph internal versions. For example, **Version: 16.2.9-98-gccaadd**. Click the top navigation bar, click the question mark menu (?), and navigate to the *About* modal box to identify the Red Hat Ceph Storage release number and the corresponding Ceph version.

3.3. CEPH FILE SYSTEM

New capabilities are available for CephFS subvolumes in ODF configured in external mode

If CephFS in ODF is configured in external mode, users like to use volume/subvolume metadata to store some Openshift specific metadata information, such as the PVC/PV/namespace from the volumes/subvolumes.

With this release, the following capabilities to set, get, update, list, and remove custom metadata from **CephFS** subvolume are added.

Set custom metadata on the subvolume as a key-value pair using:

Syntax

```
ceph fs subvolume metadata set VOLUME_NAME SUBVOLUME_NAME KEY_NAME VALUE [--group-name SUBVOLUME_GROUP_NAME]
```

Get custom metadata set on the subvolume using the metadata key:

Syntax

```
ceph fs subvolume metadata get VOLUME_NAME SUBVOLUME_NAME KEY_NAME [--group-name SUBVOLUME_GROUP_NAME ]
```

List custom metadata, key-value pairs, set on the subvolume:

Syntax

```
ceph fs subvolume metadata ls VOLUME_NAME SUBVOLUME_NAME [--group-name SUBVOLUME_GROUP_NAME ]
```

Remove custom metadata set on the subvolume using the metadata key:

Syntax

```
ceph fs subvolume metadata rm VOLUME_NAME SUBVOLUME_NAME KEY_NAME [--group-name SUBVOLUME_GROUP_NAME ] [--force]
```

-

Reason for clone failure shows up when using clone status command

Previously, whenever a clone failed, the only way to check the reason for failure was by looking into the logs.

With this release, the reason for clone failure is shown in the output of the **clone status** command:

Example

```
[ceph: root@host01 /]#  ceph fs clone status cephfs clone1
{
  "status": {
    "state": "failed",
    "source": {
      "volume": "cephfs",
      "subvolume": "subvol1",
      "snapshot": "snap1"
      "size": "104857600"
    },
    "failure": {
      "errno": "122",
      "errstr": "Disk quota exceeded"
    }
  }
}
```

The reason for a clone failure is shown in two fields:

- **errno** : error number
- **error_msg** : failure error string

3.4. CEPH MANAGER PLUGINS

CephFS NFS export can be dynamically updated using the ceph nfs export apply command

Previously, when updating a CephFS NFS export, the NFS-Ganesha servers were always restarted. This temporarily affected all the client connections served by the ganesha servers including those exports that were not updated.

With this release, a CephFS NFS export can now be dynamically updated using the **ceph nfs export apply** command. The NFS servers are no longer restarted every time a CephFS NFS export is updated.

3.5. THE CEPH VOLUME UTILITY

Users need not manually wipe devices prior to redeploying OSDs

Previously, users were forced to manually wipe devices prior to redeploying OSDs.

With this release, post zapping, physical volumes on devices are removed when no volume groups or logical volumes are remaining, thereby users are not forced to manually wipe devices anymore prior to redeploying OSDs.

3.6. CEPH OBJECT GATEWAY

Ceph Object Gateway can now be configured to direct its Ops Log to an ordinary Unix file.

With this release, Ceph Object Gateway can be configured to direct its Ops Log to an ordinary Unix file, as a file-based log is simpler to work with in some sites, when compared to a Unix domain socket. The content of the log file is identical to what would be sent to the Ops Log socket in the default configuration.

Use the **radosgw lc process** command to process a single bucket's lifecycle

With this release, users can now use the **radosgw-admin lc process** command to process only a single bucket's lifecycle from the command line interface by specifying its name **--bucket** or ID **--bucket-id**, as processing the lifecycle for a single bucket is convenient in many situations, such as debugging.

User identity information is added to the Ceph Object Gateway Ops Log output

With this release, user identity information is added to the Ops Log output to enable customers to access this information for auditing of S3 access. User identities can be reliably tracked by S3 request in all versions of the Ceph Object Gateway Ops Log.

Log levels for Ceph Object Gateway's HTTP access logging can be controlled independently with **debug_rgw_access** parameter

With this release, log levels for Ceph Object Gateway's HTTP access logging can be controlled independently with the **debug_rgw_access** parameter to provide users the ability to disable all Ceph Object Gateway logging such as **debug_rgw=0** except for these HTTP access log lines.

Level 20 Ceph Object Gateway log messages are reduced when updating bucket indices

With this release, the Ceph Object Gateway level 20 log messages are reduced when updating bucket indices to remove messages that do not add value and to reduce size of logs.

3.7. MULTI-SITE CEPH OBJECT GATEWAY

current_time field is added to the output of several **radosgw-admin** commands

With this release, **current_time** field is added to several **radosgw-admin** commands, specifically **sync status**, **bucket sync status**, **metadata sync status**, **data sync status**, and **bilog status**.

Logging of HTTP client

Previously, Ceph Object Gateway was neither printing error bodies of HTTP responses nor was there a way to match the request to the response.

With this release, a more thorough logging of HTTP client is implemented by maintaining a tag to match a HTTP request to a HTTP response for the async HTTP client and error bodies. When the Ceph Object Gateway debug is set to twenty, error bodies and other details are printed.

Read-only role for OpenStack Keystone is now available

The OpenStack Keystone service provides three roles: **admin**, **member**, and **reader**. In order to extend the role based access control (RBAC) capabilities to OpenStack, a new read-only Admin role can now be assigned to specific users in the Keystone service.

The support scope for RBAC is based on the OpenStack release.

3.8. PACKAGES

New version of grafana container provides security fixes and improved functionality

With this release, a new version of grafana container, rebased with **grafana v8.3.5** is built, which provides security fixes and improved functionality.

3.9. RADOS

MANY_OBJECTS_PER_PG warning is no longer reported when pg_autoscale_mode is set to on

Previously, Ceph health warning **MANY_OBJECTS_PER_PG** was reported in instances where **pg_autoscale_mode** was set to **on** with no distinction between the different modes that reported the health warning.

With this release, a check is added to omit reporting **MANY_OBJECTS_PER_PG** warning when **pg_autoscale_mode** is set to **on**.

OSDs report the slow operations details in an aggregated format to the Ceph Manager service

Previously, slow requests would overwhelm a cluster log with too many details, filling up the monitor database.

With this release, slow requests get logged in the cluster log by operation type and pool information and is based on OSDs reporting aggregated slow operations details to the manager service.

Users can now blocklist a CIDR range

With this release, you can blocklist a CIDR range, in addition to individual client instances and IPs. In certain circumstances, you would want to blocklist all clients in an entire data center or rack instead of specifying individual clients to blocklist. For example, failing over a workload to a different set of machines and wanting to prevent the old workload instance from continuing to partially operate. This is now possible using a "blocklist range" analogous to the existing "blocklist" command.

3.10. THE CEPH ANSIBLE UTILITY

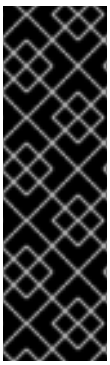
A new Ansible playbook is now available for backup and restoring Ceph files

Previously, users had to manually backup and restore files when either upgrading the OS from Red Hat Enterprise Linux 7 to Red Hat Enterprise Linux 8 or reprovisioning their machines, which was quite inconvenient especially in case of large cluster deployments.

With this release, the **backup-and-restore-ceph-files.yml** playbook is added to backup and restore Ceph files, such as **/etc/ceph** and **/var/lib/ceph** that eliminates the need for the user to manually restore files.

CHAPTER 4. 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 *Technology Preview Features Support Scope*

HA backed NFS for improved availability of NFS deployments

With this release, NFS is backed by HA to improve availability of NFS deployments. You can deploy NFS backed by **haproxy** and **keepalived**. If the placement specifies more hosts, but limits the number of hosts being used with the **count** property, NFS daemons are deployed on other hosts when an NFS host goes offline.

See *Management of NFS Ganesha gateway using the Ceph orchestrator* for more details.

4.1. CEPH OBJECT GATEWAY

Ceph Object Gateway technology preview support for S3 transparent encryption

With this release, Ceph Object Gateway provides technology preview support for S3 transparent encryption using **SSE-S3** and **S3 PutBucketEncryption** APIs.

4.2. RADOS BLOCK DEVICE

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.

CHAPTER 5. 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.

IMPORTANT

Deprecated functionality continues to be supported until the end of life of Red Hat Ceph Storage 5. Deprecated functionality will likely not be supported in future major releases of this product and is not recommended for new deployments. For the most recent list of deprecated functionality within a particular major release, refer to the latest version of release documentation.

NFS support for CephFS is now deprecated

NFS support for CephFS is now deprecated. Red Hat Ceph Storage support for NFS in OpenStack Manila is not affected. Deprecated functionality will receive only bug fixes for the lifetime of the current release, and may be removed in future releases. Relevant documentation around this technology is identified as "Limited Availability".

iSCSi support is now deprecated

iSCSi support is now deprecated in favor of future NVMEoF support. Deprecated functionality will receive only bug fixes for the lifetime of the current release, and may be removed in future releases. Relevant documentation around this technology is identified as "Limited Availability".

CHAPTER 6. 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.

6.1. THE CEPHADM UTILITY

Container process number limit set to max

Previously, the process number limit, 2048, on the containers prevented new processes from being forked beyond the limit.

With this release, the process number limit is set to **max**, which allows you to create as many luns as required per target. However, the number is still limited by the server resources.

(BZ#1976128)

Unavailable devices are no longer passed when creating OSDs in a batch

Previously, devices with GPT headers were not marked as unavailable. Cephadm would attempt to create OSDs on those devices, along with other valid devices, in a batch leading to failure of the batch OSD creation, since OSDs cannot be created on devices with GPT headers. This would not create OSDs.

With this fix, unavailable devices are no longer passed when creating OSDs in a batch and having devices with GPT headers no longer blocks creating OSDs on valid devices.

(BZ#1962511)

Users providing --format argument with unsupported formats received a traceback

Previously, the orchestrator would throw an exception whenever it received a **--format** argument that it did not support, causing users who passed **--format** with unsupported formats to receive a traceback.

With this fix, unsupported formats are now properly handled and users providing an unsupported format get a message explaining that the format is unsupported.

(BZ#2006214)

The ceph-common packages can now be installed without dependency errors

Previously, after upgrading Red Hat Ceph Storage 4 to Red Hat Ceph Storage 5, a few packages were left out which caused dependency errors.

With this fix, the left out Red Hat Ceph Storage 4 packages are removed and the **ceph-common** packages can now be installed during preflight playbook execution without any errors.

(BZ#2008402)

The tcmu-runner daemons are no longer reported as stray daemons

Previously, **tcmu-runner** daemons were not actively tracked by **cephadm** as they were considered part of iSCSI. This resulted in **tcmu-runner** daemons getting reported as stray daemons since **cephadm** was not tracking them.

With this fix, when a **tcmu-runner** daemon matches up with a known iSCSI daemon, it is not marked as a stray daemon.

(BZ#2018906)

Users can re-add host with an active manager without an explicit IP

Previously, whenever **cephadm** attempted to resolve the IP address of the current host from within a container, there was a chance of it resolving to a loopback address. An explicit IP was required if the user wished to re-add the host with the active Ceph Manager, and users would receive an error message if they did not provide it.

With the current fix, **cephadm** reuses the old IP when re-adding the host if it is not explicitly provided and name resolution returns a loopback address. Users can now re-add the host with the active manager without an explicit IP.

(BZ#2024301)

cephadm verifies if the fsid of the daemon it was inferring a config from matches the expected fsid

Previously, in **cephadm**, there was no check to verify if the **fsid** of the daemon it was inferring a configuration from matched the expected **fsid**. Due to this, if users had a **/var/lib/ceph/*FSID*/*DAEMON_NAME*** directory with an **fsid** other than the expected one, the configuration from that daemon directory would still be inferred.

With this fix, checking is done to verify if the **fsid** matches what is expected and users no longer get a "failed to probe daemons or devices" error.

(BZ#2024720)

cephadm supports copying client keyrings with different names

Previously, **cephadm** would enforce a file name at the destination, when copying the client keyring **ceph.keyring**.

With the current fix, **cephadm** supports copying the client keyring with a different name, eliminating the issue of automatic renaming when copied.

(BZ#2028628)

User can bootstrap a cluster with multiple public networks with -c ceph.conf option

Previously, **cephadm** would not parse multiple public networks during bootstrap, when they were provided as part of the **-c ceph.conf** option. Due to this, it was not possible to bootstrap a cluster with multiple public networks.

With the current fix, from the provided **ceph.conf** file, the **public network** field is correctly parsed and can now be used to populate the **public_network mon config** field, enabling the user to bootstrap a cluster providing multiple public networks by using the **-c ceph.conf** option.

(BZ#2035179)

Setting up a MDS service with a numeric service ID throws an error to alert user

Previously, setting up a MDS service with a numeric service ID would result in crashing of the MDS daemons.

With this fix, if an attempt is made to create a MDS service with a numeric service ID, an error is immediately thrown to alert and warn the users to not use a numeric service ID.

(BZ#2039669)

The ceph orch redeploy mgr command redeploys the active Manager daemon last

Previously, the **ceph orch redeploy mgr** command would cause the Ceph Manager daemons to continually redeploy themselves without clearing the scheduled redeploy action which would result in the Ceph Manager daemons endlessly flapping.

With this release, the ordering of the redeployment was adjusted so that the active manager daemon is always redeployed last and the command **ceph orch redeploy mgr** now only redeploys each Ceph Manager once.

(BZ#2042602)

Adopting clusters with custom name is now supported

Previously, adopting Ceph OSD containers from a Ceph cluster with custom name failed as **cephadm** would not propagate custom clusters in the **unit.run** file.

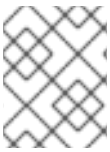
With this release, **cephadm** changes the LVM metadata and enforces the default cluster name "Ceph" thereby adopting a cluster with custom cluster names works as expected.

(BZ#2058038)

cephadm no longer adds docker.io to the image name provided for the ceph orch upgrade start command

Previously, **cephadm** would add **docker.io** to any image from an unqualified registry, thereby it was impossible to pass an image from an unqualified registry, such as a local registry, to upgrade, as it would fail to pull this image.

Starting with Red Hat Ceph Storage 5.2, , **docker.io** is no longer added to the image name, unless the name is a match for an upstream ceph image such as **ceph/ceph:v17**. On running the **ceph orch upgrade** command, users can pass images from local registries and **Cephadm** can upgrade to that image.



NOTE

This is ONLY applicable to upgrades starting from 5.2. Upgrading from 5.1 to 5.2 is still affected by this issue.

(BZ#2077843)

Cephadm no longer infers configuration files from legacy daemons

Previously, **Cephadm** would infer config files from legacy daemons, regardless of whether the daemons were still present, based on the existence of a **/var/lib/ceph/{mon|osd|mgr}** directory. This caused certain tasks, such as refreshing the disks, to fail on nodes where these directories existed, as **Cephadm** would throw an error when it attempts to infer the non-existent configuration file.

With the current fix, **Cephadm** no longer infers configuration files from legacy daemons; instead it checks for existing configuration files before inferring. **Cephadm** no longer encounters issues when refreshing daemons or devices on a host, due to the existence of a legacy daemon directory.

(BZ#2080242)

.rgw.root pool is no longer created automatically

Previously, an additional check for Ceph Object Gateway for multi-site existed, which caused the automatic creation of the **.rgw.root** pool even when the user had deleted it.

Starting with Red Hat Ceph Storage 5.2 the multi-site check is removed and the **.rgw.root** pool is no longer automatically created, unless the user takes Ceph Object Gateway -related actions that results in its creation.

(BZ#2083885)

The Ceph Manager daemon is removed from a host that is no longer specified in the placement specification in cephadm

Previously, the current active manager daemon would not be removed from **cephadm** even if it no longer matched the placement specified in the manager service specification. Whenever users changed the service specification to exclude the host where the current active manager was, they would end up with an extra manager until they caused a failover.

With this fix, **cephadm** fails over the manager if a standby is available and the active manager is on a host that no longer matches the service specification. Ceph Manager daemon is removed from a host that is no longer specified in the placement specification in **cephadm** even if the manager is the active one.

(BZ#2086438)

A 404 error due to a malformed URL was causing tracebacks in the logs.

Previously, **cephadm** would incorrectly form the URL for the prometheus receiver, causing a traceback to be printed in the log due to a 404 error that would occur when trying to access the malformed URL.

With this fix, the URL formatting has been fixed and the 404 error is avoided. Tracebacks are no longer logged.

(BZ#2087736)

cephadm no longer removes osd_memory_target config settings at host level

Previously, if **osd_memory_target_autotune** was turned off globally, **cephadm** would remove the values that the user set for **osd_memory_target** at the host level. Additionally, for hosts with FQDN name, even though the CRUSH map uses a short name, **cephadm** would still set the config option using the FQDN. Due to this, users could not manually set **osd_memory_target** at the host level and **osd_memory_target** auto tuning would not work with FQDN hosts.

With this fix, the **osd_memory_target** config settings is not removed from **cephadm** at the host level if **osd_memory_target_autotune** is set to **false**. It also always users a short name for hosts when setting host level **osd_memory_target**. If at the host level **osd_memory_target_autotune** is set to **false**, users can manually set the **osd_memory_target** and have the options not be removed by **cephadm**. Additionally, autotuning should now work with hosts added to **cephadm** with FQDN names.

(BZ#2092089)

Cephadm uses the FQDN to build the alertmanager webhook URLs

Previously, **Cephadm** picked alertmanager webhook URLs based on the IP address it had stored for the hosts. This caused issues since these webhook URLs would not work for certain deployments.

With this fix, **Cephadm** uses FQDNs to build the alertmanager webhook URLs, enabling webhook URLs to work for some deployment situations which were previously broken.

(BZ#2099348)

6.2. CEPH DASHBOARD

Drain action on the Ceph dashboard ensures safe removal of host

Previously, whenever a user removed a host on the Ceph dashboard without moving out all the daemons, the host transitioned to an unusable state or a ghost state.

With this fix,users can use the drain action on the dashboard to move all the daemons out from the host. Upon successful completion of the drain action, the host can be safely removed.

(BZ#1889976)

Performance details graphs show the required data on the Ceph Dashboard

Previously, due to related metrics being outdated, performance details graphs for a daemon were showing no data even when **put**/**get** operations were being performed.

With this fix, the related metrics are up-to-date and performance details graphs show the required data.

(BZ#2054967)

Alertmanager shows the correct MTU mismatch alerts

Previously, **Alertmanager** was showing false **MTU mismatch** alerts for cards that were in **down** state as well.

With this fix, **Alertmanager** shows the correct **MTU mismatch** alerts.

(BZ#2057307)

PG status chart no longer displays unknown placement group status

Previously, **snaptrim_wait** placement group (PG) state was incorrectly parsed and split into 2 states, **snaptrim** and **wait**, which are not valid PG states. This caused the PG status chart to incorrectly show a few PGs in unknown states, even though all of them were in known states.

With this fix, **snaptrim_wait** and all states containing an underscore are correctly parsed and the unknown PG status is no longer displayed in the PG states chart.

(BZ#2077827)

Ceph Dashboard improved user interface

Previously, the following issues were identified in the Ceph Dashboard user interface, causing it to be unusable when tested with multi-path storage clusters:

- In clusters, with multi-path storage devices, if a disk was selected in the *Physical Disks* page, multiple disks would be selected and the selection count of the table would start incrementing, until the table stopped responding within a minute.
- The *Device Health* page showed errors while fetching the SMART data.
- *Services* column in the *Hosts* page showed a lot of entries, thereby reducing readability.

With this release, the following fixes are implemented, resulting in improved user interface:

- Fixed the disk selection issue in the *Physical Disks* page.
- An option to fetch the scsi devices SMART data is added.
- *Services* column is renamed as *Service Instances* and just the instance name and instance count of that service is displayed in a badge.

(BZ#2097487)

6.3. CEPH FILE SYSTEM

Fetching ceph.dir.layout for any directory returns the closest inherited layout

Previously, the directory paths did not traverse to the root to find the closest inherited layout causing the system to return a "No such attribute" message for directories that did not have a layout set specifically on them.

With this fix, the directory paths traverse to the root to find the closest inherited layout and fetches the **ceph.dir.layout** for any directory from the directory hierarchy.

(BZ#1623330)

The subvolumegroup ls API filters the internal trash directory _deleting

Previously, the **subvolumegroup ls** API would not filter internal trash directory **_deleting**, causing it to be listed as a **subvolumegroup**.

With this fix, the **subvolumegroup ls** API filters the internal trash directory **_deleting** and the **subvolumegroup ls** API doesn't show the internal trash directory **_deleting**.

(BZ#2029307)

Race condition no longer causes confusion among MDS in a cluster

Previously, a race condition in MDS, during messenger setup, would result in confusion among other MDS in the cluster, causing other MDS to refuse communication.

With this fix, the race condition is rectified, establishing successful communication among the MDS.

(BZ#2030540)

MDS can now trigger stray reintegration with online scrub

Previously, stray reintegrations were triggered only on client requests, resulting in the process of clearing out stray inodes to require expensive recursive directory listings by a client.

With this fix, MDS can now trigger stray reintegration with online scrub.

(BZ#2041563)

MDS reintegrates strays if target directories are full

Previously, MDS would not reintegrate strays if the target directory of the link was full causing the stray directory to fill up in degenerate situations.

With this fix, MDS proceeds with stray integration even when target directories are full as no change in size occurs.

(BZ#2041571)

Quota is enforced on the clone after the data is copied

Previously, the quota on the clone would be set prior to copying the data from the source snapshot and the quota would be enforced before copying the entire data from the source. This would cause the subvolume snapshot clone to fail if the quota on the source exceeded. Since the quota is not strictly enforced at the byte range, this is a possibility.

With this fix, the quota is enforced on the clone after the data is copied. The snapshot clone always succeeds irrespective of the quota.

(BZ#2043602)

Disaster recovery automation and planning resumes after ceph-mgr restart

Previously, schedules would not start during **ceph-mgr** startup which affected the disaster recovery plans of users who presumed that the snapshot schedule would resume at **ceph-mgr** restart time.

With this fix, schedules start on **ceph-mgr** restart and the disaster recovery automation and planning, such as snapshot replication, immediately resumes after **ceph-mgr** is restarted, without the need for manual intervention.

(BZ#2055173)

The mdlog is flushed immediately when opening a file for reading

Previously, when opening a file for reading, MDS would revoke the Fw capability from the other clients and when the Fw capability was released, the MDS could not flush the **mdlog** immediately and would block the Fr capability. This would cause the process that requested for a file to be stuck for about 5 seconds until the **mdlog** was flushed by MDS periodically every 5 seconds.

With this release, the **mdlog** flush is triggered immediately when there is any capability wanted when releasing the Fw capability and you can open the file for reading quickly.

(BZ#2076850)

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#2081596)

New clients are compatible with old Ceph cluster

Previously, new clients were incompatible with old Ceph clusters causing the old clusters to trigger **abort()** to crash the MDS daemons when receiving unknown metrics.

With this fix, ensure to check the feature bits in the client and collect and send only those metrics that are supported by MDSs. New clients are compatible with old cephs.

(BZ#2081929)

Ceph Metadata Server no longer crashes during concurrent lookup and unlink operations

Previously, an incorrect assumption of an assert placed in the code, which gets hit on concurrent lookup and unlink operations from a Ceph client, caused Ceph Metadata Server crash.

The latest fix moves the assertion to the relevant place where the assumption, during concurrent lookup and unlink operation, is valid, resulting in the continuation of Ceph Metadata Server serving the Ceph client operations without crashing.

(BZ#2093065)

MDSs no longer crash when fetching unlinked directories

Previously, when fetching unlinked directories, the projected version would be incorrectly initialized, causing MDSs to crash when performing sanity checks.

With this fix, the projected version and the inode version are initialized when fetching an unlinked directory, allowing the MDSs to perform sanity checks without crashing.

(BZ#2108656)

6.4. CEPH MANAGER PLUGINS

The missing pointer is added to the **PriorityCache** perf counters builder and perf output returns the **prioritycache** key name

Previously, the PriorityCache perf counters builder was missing a necessary pointer, causing the perf counter output, **ceph tell *DAEMON_TYPE.DAEMON_ID* perf dump** and **ceph tell *DAEMON_TYPE.DAEMON_ID* perf schema**, to return an empty string instead of the **prioritycache** key. This missing key caused a failure in the **collectd-ceph** plugin.

With this fix, the missing pointer is added to the **PriorityCache** perf counters builder. The perf output returns the **prioritycache** key name.

(BZ#2064627)

Vulnerability with OpenStack 16.x Manila with Native CephFS and external Red Hat Ceph Storage 5

Previously, customers who were running OpenStack 16.x (with Manila) and external Red Hat Ceph Storage 4, who upgraded to Red Hat Ceph Storage 5.0, 5.0.x, 5.1, or 5.1.x, were potentially impacted by a vulnerability. The vulnerability allowed an OpenStack Manila user/tenant (owner of a Ceph File System share) to maliciously obtain access (read/write) to any Manila share backed by CephFS, or even the entire CephFS file system. The vulnerability is due to a bug in the "volumes" plugin in Ceph Manager. This plugin is responsible for managing Ceph File System subvolumes which are used by OpenStack Manila services as a way to provide shares to Manila users.

With this release, this vulnerability is fixed. Customers running OpenStack 16.x (with Manila providing native CephFS access) who upgraded to external Red Hat Ceph Storage 5.0, 5.0.x, 5.1, or 5.1.x should upgrade to Red Hat Ceph Storage 5.2. Customers who only provided access via NFS are not impacted.

(BZ#2056108)

6.5. THE CEPH VOLUME UTILITY

Missing backport is added and OSDs can be activated

Previously, OSDs could not be activated due to a regression caused by a missing backport.

With this fix, the missing backport is added and OSDs can be activated.

(BZ#2093022)

6.6. CEPH OBJECT GATEWAY

Lifecycle policy for a versioned bucket no longer fails in between reshards

Previously, due to an internal logic error, lifecycle processing on a bucket would be disabled during bucket resharding causing the lifecycle policies for an affected bucket to not be processed.

With this fix, the bug has been rectified and the lifecycle policy for a versioned bucket no longer fails in between reshards.

(BZ#1962575)

Deleted objects are no longer listed in the bucket index

Previously, objects would be listed in the bucket index if the delete object operations did not complete normally, causing the objects that should have been deleted to still be listed.

With this release, the internal "dir_suggest" that finalizes incomplete transactions is fixed and deleted objects are no longer listed.

(BZ#1996667)

Zone group of the Ceph Object Gateway is sent as the awsRegion value

Previously, the value of **awsRegion** was not populated with the zonegroup in the event record.

With this fix, the zone group of the Ceph Object Gateway is sent as the **awsRegion** value.

(BZ#2004171)

Ceph Object Gateway deletes all notification topics when an empty list of topics is provided

Previously, in Ceph Object Gateway, notification topics were deleted accurately by name, but would not follow AWS behavior to delete all topics when given an empty topic name, causing a few customer bucket notification workflows to be unusable with Ceph Object Gateway.

With this fix, explicit handling for empty topic lists has changed and Ceph Object Gateway deletes all the notification topics when an empty list of topics is provided.

(BZ#2017389)

Crashes in bucket listing, bucket stats, and similar operations are not seen for indexless buckets

Previously, several operations, including general bucket listing, would incorrectly attempt to access index information from indexless buckets causing a crash.

With this fix, new checks for indexless buckets are added, thereby crashes in bucket listing, bucket stats, and similar operations are not seen.

(BZ#2043366)

Internal table index is prevented from becoming negative

Previously, an index into an internal table was allowed to become negative after a period of continuous operation, which caused the Ceph Object Gateway to crash.

With this fix, the index is prevented from becoming negative and the Ceph Object Gateway no longer crashes.

(BZ#2079089)

Usage of MD5 in a FIPS-enabled environment is explicitly allowed and S3 multipart operations can be completed

Previously, in a FIPS-enabled environment, the usage of MD5 digest was not allowed by default, unless explicitly excluded for non-cryptographic purposes. Due to this, a segfault occurred during the S3 complete multipart upload operation.

With this fix, the usage of MD5 for non-cryptographic purposes in a FIPS-enabled environment for S3 complete multipart **PUT** operations is explicitly allowed and the S3 multipart operations can be completed.

(BZ#2088602)

Result code 2002 of radosgw-admin commands is explicitly translated to 2

Previously, a change in the S3 error translation of internal **NoSuchBucket** result inadvertently changed the error code from the **radosgw-admin bucket stats** command causing the programs checking the shell result code of those **radosgw-admin** commands to see a different result code.

With this fix, the result code 2002 is explicitly translated to 2 and users can see the original behavior.

(BZ#2100967)

Usage of MD5 in a FIPS-enabled environment is explicitly allowed and S3 multipart operations can be completed

Previously, in a FIPS-enabled environment, the usage of MD5 digest was not allowed by default, unless explicitly excluded for non-cryptographic purposes. Due to this, a segfault occurred during the S3 complete multipart upload operation.

With this fix, the usage of MD5 for non-cryptographic purposes in a FIPS-enabled environment for S3 complete multipart **PUT** operations is explicitly allowed and the S3 multipart operations can be completed.

6.7. MULTI-SITE CEPH OBJECT GATEWAY

radosgw-admin bi purge command works on deleted buckets

Previously, **radosgw-admin bi purge** command required a bucket entrypoint object, which does not exist for deleted buckets causing **bi purge** to be unable to clean up after deleted buckets.

With this fix, **bi purge** accepts **--bucket-id** to avoid the need for a bucket entry point and the command works on deleted buckets.

(BZ#1910503)

Null pointer check no longer causes multi-site data sync crash

Previously, a null pointer access would crash the multisite data sync.

With this fix, null pointer check is successfully implemented, preventing any possible crashes.

(BZ#1967901)

Metadata sync no longer gets stuck when encountering errors

Previously, some errors in metadata sync would not retry, causing sync to get stuck when some errors occurred in a Ceph Object Gateway multi-site configuration.

With this fix, retry behaviour is corrected and metadata sync does not get stuck when errors are encountered.

(BZ#2068039)

Special handling is added for **rgw_data_notify_interval_msec=0** parameter

Previously, **rgw_data_notify_interval_msec** had no special handling for 0, resulting in the primary site flooding the secondary site with notifications.

With this fix, special handling for **rgw_data_notify_interval_msec=0** is added and async data notification can now be disabled.

(BZ#2102365)

6.8. RADOS

PGs no longer get incorrectly stuck in **remapped+peering** state in stretch mode

Previously, due to a logical error, when operating a cluster in stretch mode, it was possible for some placement groups (PGs) to get permanently stuck in **remapped+peering** state under certain cluster conditions, causing the data to be unavailable until the OSDs were taken offline.

With this fix, PGs choose stable OSD sets and they no longer get incorrectly stuck in **remapped+peering** state in stretch mode.

(BZ#2042417)

OSD deployment tool successfully deploys all the OSDs while making changes to the cluster

The KVMonitor paxos services manages the keys being added, removed, or modified when performing changes to the cluster. Previously, while adding new OSDs using the OSD deployment tool, the keys would be added without verifying whether the service could write to it. Due to this, assertion failure would occur in the paxos code causing the monitor to crash.

The latest fix ensures that the KVMonitor service is able to write prior to adding new OSDs, otherwise, the command back is pushed back into the relevant queue to be retried at a later point. The OSD deployment tool successfully deploys all the OSDs without any issues.

(BZ#2086419)

Corrupted dups entries of a PG Log can be removed by off-line and on-line trimming

Previously, trimming of PG log dups entries could be prevented during the low-level PG split operation, which is used by the PG autoscaler with far higher frequency than by a human operator. Stalling the trimming of dups resulted in significant memory growth of PG log, leading to OSD crashes as it ran out of memory. Restarting an OSD did not solve the problem as the PG log is stored on disk and reloaded to RAM on startup.

With this fix, both off-line (using the **ceph-objectstore-tool** command) and on-line (within OSD) trimming is able to remove corrupted dups entries of a PG Log that jammed the on-line trimming machinery and were responsible for the memory growth. A debug improvement is implemented that prints the number of dups entries to the OSD's log to help future investigations.

(BZ#2093031)

6.9. RBD MIRRORING

last_copied_object_number value is properly updated for all images

Previously, due to an implementation defect, **last_copied_object_number** value was properly updated only for fully allocated images. This caused the **last_copied_object_number** value to be incorrect for any sparse image and the image replication progress to be lost in case of abrupt rbd-mirror daemon restart.

With this fix, **last_copied_object_number** value is properly updated for all images and upon rbd-mirror daemon restart, image replication resumes from where it had previously stopped.

(BZ#2019909)

Existing schedules take effect when an image is promoted to primary

Previously, due to an ill-considered optimization, existing schedules would not take effect following an image's promotion to primary resulting in the snapshot-based mirroring process to not start for a recently promoted image.

With this release, the optimization causing this issue is removed and the existing schedules now take effect when an image is promoted to primary and the snapshot-based mirroring process starts as expected.

(BZ#2020618)

Snapshot-based mirroring process no longer gets cancelled

Previously, as a result of an internal race condition, the **rbd mirror snapshot schedule add** command would be cancelled out. The snapshot-based mirroring process for the affected image would not start, if no other existing schedules were applicable.

With this release, the race condition is fixed and the snapshot-based mirroring process starts as expected.

(BZ#2069720)

Replay or resync is no longer attempted if the remote image is not primary

Previously, due to an implementation defect, replay or resync would be attempted even if the remote image was not primary, that is, there is nowhere to replay or resync from. This caused the snapshot-based mirroring to run into a livelock and to continuously report "failed to unlink local peer from remote image" error.

With this fix, the implementation defect is fixed and replay or resync is not attempted if the remote image is not primary, thereby no errors are reported.

(BZ#2081715)

Mirror snapshots that are in use by rbd-mirror daemon on the secondary cluster are not removed

Previously, as a result of an internal race condition, the mirror snapshot that was in use by the rbd-mirror daemon on the secondary cluster would be removed, causing the snapshot-based mirroring process for the affected image to stop, reporting a "split-brain" error.

With this fix, the mirror snapshot queue is extended in length and the mirror snapshot cleanup procedure is amended accordingly. Mirror snapshots that are in use by the rbd-mirror daemon on the secondary cluster are no longer removed and the snapshot-based mirroring process does not stop.

(BZ#2092838)

Logic no longer causes RBD mirror to crash if owner is locked during **schedule_request_lock()**

Previously, during **schedule_request_lock()**, for an already locked owner, the block device mirror would crash and image syncing would stop.

With this fix, if the owner is already locked, **schedule_request_lock()** is gracefully aborted and the block device mirroring does not crash.

(BZ#2102227)

Image replication no longer stops with **incomplete local non-primary snapshot** error

Previously, due to an implementation defect, upon an abrupt rbd-mirror daemon restart, image replication would stop with **incomplete local non-primary snapshot** error.

With this fix, image replication no longer stops with **incomplete local non-primary snapshot** error and works as expected.

(BZ#2105454)

6.10. THE CEPH ANSIBLE UTILITY

Correct value is set for **autotune_memory_target_ratio** when migrating to **cephadm**

Previously, when migrating to **cephadm**, nothing would set a proper value for **autotune_memory_target_ratio** depending on the kind of deployment, HCI or non_HCI. Due to this, no ratio was set and there would be no difference between the two deployments.

With this fix, the **cephadm-adopt playbook** sets the right ratio depending on the kind of deployment and the right value is set for **autotune_memory_target_ratio** parameter.

(BZ#2028693)

CHAPTER 7. KNOWN ISSUES

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

7.1. THE CEPHADM UTILITY

Crash daemon might not be able to send crash reports to the storage cluster

Due to an issue with the crash daemon configuration, it might not be possible to send crash reports to the cluster from the crash daemon.

(BZ#2062989)

Users are warned while upgrading to Red Hat Ceph Storage 5.2

Previously, buckets resharded in Red Hat Ceph Storage 5, might not have been understandable by a Red Hat Ceph Storage 5.2 Ceph Object Gateway daemon, due to which an upgrade warning/blocker was added to make sure all users upgrading to Red Hat Ceph Storage 5.2 are aware of the issue and can downgrade if they were previously using Red Hat Ceph Storage 5.1 with object storage.

As a workaround, users not using object storage or upgrading from a version other than 5.1 can run **ceph config set mgr mgr/cephadm/no_five_one_rgw --force** to remove the warning/blocker and return all operations to normal. By setting this config option users have acknowledged that they are aware of the Ceph Object Gateway issue before they upgrade to Red Hat Ceph Storage 5.2.

(BZ#2104780)

HA-backed I/O operations on the virtual IP that the NFS daemon is on, are not maintained across the failover HAProxy configurations are not updated with NFS daemons

The HAProxy configurations are not updated when failing over NFS daemons from an offline to an online host. As a result, the HA-backed I/O operations are directed to the virtual IP that the NFS daemon is on and is not maintained across the failover.

(BZ#2106849)

7.2. CEPH DASHBOARD

Creating ingress service with SSL from the Ceph Dashboard is not working

The ingress service creation with SSL from the Ceph Dashboard is not working as the form expects the user to populate a **Private key** field, which is not a required field anymore.

To workaround this issue, the ingress service is successfully created using the Ceph orchestrator CLI.

(BZ#2080276)

"Throughput-optimized" option is recommended for clusters containing SSD and NVMe devices

Previously, whenever the cluster had either only SSD devices or both SSDs and NVMe devices, the "Throughput-optimized" option would be recommended, even though it shouldn't be and it had no impact either on the user or the cluster.

As a workaround, users can use the "Advanced" mode for deploying OSDs according to their desired specifications and all the options in the "Simple" mode are still usable apart from this UI issue.

(BZ#2101680)

7.3. CEPH FILE SYSTEM

The getpath command causes automation failure

An assumption that the directory name returned by the **getpath** command is the directory under which snapshots would be created causes automation failure and confusion.

As a workaround, it is recommended to use the directory path that is one level higher, to add to the **snap-schedule add** command. Snapshots are available one level higher than at the level returned by the **getpath** command.

(BZ#2053706)

7.4. CEPH OBJECT GATEWAY

Upgrading to Red Hat Ceph Storage 5.2 from Red Hat Ceph Storage 5.1 with Ceph Object Gateway configuration is not supported

Upgrading to Red Hat Ceph Storage 5.2 from Red Hat Ceph Storage 5.1 on any Ceph Object Gateway (RGW) clusters (single-site or multi-site) is not supported due to a known issue BZ#2100602.

For more information, see Support Restrictions for upgrades for RGW.

WARNING

Do not upgrade Red Hat Ceph Storage clusters running on Red Hat Ceph Storage 5.1 and Ceph Object Gateway (single-site or multi-site) to the Red Hat Ceph Storage 5.2 release.

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/
- For Red Hat Enterprise Linux 9:
  http://ftp.redhat.com/redhat/linux/enterprise/9Base/en/RHCEPH/SRPMS/