



Red Hat Ceph Storage 4

Data Security and Hardening Guide

Red Hat Ceph Storage Data Security and Hardening Guide

Red Hat Ceph Storage 4 Data Security and Hardening Guide

Red Hat Ceph Storage Data Security and Hardening Guide

Legal Notice

Copyright © 2021 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

This document provides data security and hardening information for Ceph Storage Clusters and their clients. 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
1.1. PREFACE	3
1.2. INTRODUCTION TO RHCS	3
1.3. SUPPORTING SOFTWARE	4
CHAPTER 2. THREAT AND VULNERABILITY MANAGEMENT	5
2.1. THREAT ACTORS	5
2.2. SECURITY ZONES	6
2.3. CONNECTING SECURITY ZONES	7
2.4. SECURITY-OPTIMIZED ARCHITECTURE	7
CHAPTER 3. ENCRYPTION AND KEY MANAGEMENT	9
3.1. SSH	9
3.2. SSL TERMINATION	9
3.3. ENCRYPTION IN TRANSIT	9
3.3.1. Enabling the messenger v2 protocol	11
3.4. ENCRYPTION AT REST	12
CHAPTER 4. IDENTITY AND ACCESS MANAGEMENT	14
4.1. CEPH STORAGE CLUSTER USER ACCESS	14
4.2. CEPH OBJECT GATEWAY USER ACCESS	15
4.3. CEPH OBJECT GATEWAY LDAP/AD AUTHENTICATION	15
4.4. CEPH OBJECT GATEWAY OPENSTACK KEYSTONE AUTHENTICATION	15
CHAPTER 5. INFRASTRUCTURE SECURITY	17
5.1. ADMINISTRATION	17
5.2. NETWORK COMMUNICATION	17
5.3. HARDENING THE NETWORK SERVICE	18
5.4. REPORTING	20
5.5. AUDITING ADMINISTRATOR ACTIONS	20
CHAPTER 6. DATA RETENTION	22
6.1. CEPH STORAGE CLUSTER	22
6.2. CEPH BLOCK DEVICE	22
6.3. CEPH FILESYSTEM	22
6.4. CEPH OBJECT GATEWAY	23
CHAPTER 7. FEDERAL INFORMATION PROCESSING STANDARD (FIPS)	24
CHAPTER 8. SUMMARY	25

CHAPTER 1. INTRODUCTION

Security is an important concern and should be a strong focus of any Red Hat Ceph Storage deployment. Data breaches and downtime are costly and difficult to manage, laws may require passing audits and compliance processes, and projects have an expectation of a certain level of privacy and security of their data. This document provides a general introduction to security for Red Hat Ceph Storage, as well as the role of Red Hat in supporting your system's security.

1.1. PREFACE

This document provides advice and good practice information for hardening the security of your Red Hat Ceph Storage deployment, with a focus on Ceph Ansible-based deployments. While following the instructions in this guide will help harden the security of your environment, we do not guarantee security or compliance from following these recommendations.

1.2. INTRODUCTION TO RHCS

Red Hat Ceph Storage (RHCS) is a highly scalable and reliable object storage solution, which is typically deployed in conjunction with cloud computing solutions like OpenStack, as a standalone storage service, or as network attached storage using interfaces such as iSCSI.

All RHCS deployments consist of a storage cluster commonly referred to as the Ceph Storage Cluster or RADOS (Reliable Autonomous Distributed Object Store), which consists of three types of daemons:

- **Ceph Monitors (`ceph-mon`):** Ceph monitors provide a few critical functions: first, they establish agreement about the state of the cluster; second, they maintain a history of the state of the cluster, such as whether an OSD is up and running and in the cluster; third, they provide a list of pools through which clients write and read data; and finally, they provide authentication for clients and the Ceph Storage Cluster daemons.
- **Ceph Managers (`ceph-mgr`):** Ceph manager daemons track the status of peering between copies of placement groups distributed across Ceph OSDs, a history of the placement group states, and metrics about the Ceph cluster. They also provide interfaces for external monitoring and management systems.
- **Ceph OSDs (`ceph-osd`):** Ceph Object Storage Daemons (OSDs) store and serve client data, replicate client data to secondary Ceph OSD daemons, track and report to Ceph Monitors on their health and on the health of neighboring OSDs, dynamically recover from failures and backfill data when the cluster size changes, among other functions.

All RHCS deployments store end-user data in the Ceph Storage Cluster or RADOS (Reliable Autonomous Distributed Object Store). Generally, end users **DO NOT** interact with the Ceph Storage Cluster directly. Rather, they interact with a Ceph client. There are three primary Ceph Storage Cluster clients:

- **Ceph Object Gateway (`ceph-radosgw`):** The Ceph Object Gateway—also known as RADOS Gateway, `radosgw` or `rgw`—provides an object storage service with RESTful APIs. Ceph Object Gateway stores data on behalf of its clients in the Ceph Storage Cluster or RADOS.
- **Ceph Block Device (`rbd`):** The Ceph Block Device provides copy-on-write, thin-provisioned and cloneable virtual block devices to a Linux kernel via Kernel RBD (`krbd`) or to cloud computing solutions like OpenStack via `librbd`.
- **Ceph Filesystem (`cephfs`):** The Ceph Filesystem consists of one or more Metadata Servers (`mds`), which store the inode portion of a filesystem as objects on the Ceph Storage Cluster.

Ceph filesystems can be mounted via a kernel client, a FUSE client, or via the **libcephfs** library for cloud computing solutions like OpenStack.

Additional clients include **librados**, which enables developers to create custom applications to interact with the Ceph Storage cluster, and command line interface clients for administrative purposes.

1.3. SUPPORTING SOFTWARE

An important aspect of Red Hat Ceph Storage security is to deliver solutions that have security built-in upfront and that Red Hat supports over time. Specific steps which Red Hat takes with Red Hat Ceph Storage include:

- Maintaining upstream relationships and community involvement to help focus on security from the start.
- Selecting and configuring packages based on their security and performance track records.
- Building binaries from associated source code (instead of simply accepting upstream builds).
- Applying a suite of inspection and quality assurance tools to prevent an extensive array of potential security issues and regressions.
- Digitally signing all released packages and distributing them through cryptographically authenticated distribution channels.
- Providing a single, unified mechanism for distributing patches and updates.

In addition, Red Hat maintains a dedicated security team that analyzes threats and vulnerabilities against our products, and provides relevant advice and updates through the Customer Portal. This team determines which issues are important, as opposed to those that are mostly theoretical problems. The Red Hat Product Security team maintains expertise in, and makes extensive contributions to the upstream communities associated with our subscription products. A key part of the process, Red Hat Security Advisories, deliver proactive notification of security flaws affecting Red Hat solutions—along with patches that are frequently distributed on the same day the vulnerability is first published.

CHAPTER 2. THREAT AND VULNERABILITY MANAGEMENT

Red Hat Ceph Storage (RHCS) is typically deployed in conjunction with cloud computing solutions, so it can be helpful to think about an Red Hat Ceph Storage deployment abstractly as one of many series of components in a larger deployment. These deployments typically have shared security concerns, which this guide refers to as *Security Zones*. Threat actors and vectors are classified based on their motivation and access to resources. The intention is to provide you with a sense of the security concerns for each zone, depending on your objectives.

2.1. THREAT ACTORS

A threat actor is an abstract way to refer to a class of adversary that you might attempt to defend against. The more capable the actor, the more rigorous the security controls that are required for successful attack mitigation and prevention. Security is a matter of balancing convenience, defense, and cost, based on requirements. In some cases it will not be possible to secure a Red Hat Ceph Storage deployment against all of the threat actors described here. When deploying Red Hat Ceph Storage, you must decide where the balance lies for your deployment and usage.

As part of your risk assessment, you must also consider the type of data you store and any accessible resources, as this will also influence certain actors. However, even if your data is not appealing to threat actors, they could simply be attracted to your computing resources.

- **Nation-State Actors:** This is the most capable adversary. Nation-state actors can bring tremendous resources against a target. They have capabilities beyond that of any other actor. It is very difficult to defend against these actors without incredibly stringent controls in place, both human and technical.
- **Serious Organized Crime:** This class describes highly capable and financially driven groups of attackers. They are able to fund in-house exploit development and target research. In recent years the rise of organizations such as the Russian Business Network, a massive cyber-criminal enterprise, has demonstrated how cyber attacks have become a commodity. Industrial espionage falls within the serious organized crime group.
- **Highly Capable Groups:** This refers to 'Hactivist' type organizations who are not typically commercially funded but can pose a serious threat to service providers and cloud operators.
- **Motivated Individuals Acting Alone:** These attackers come in many guises, such as rogue or malicious employees, disaffected customers, or small-scale industrial espionage.
- **Script Kiddies:** These attackers don't target a specific organization, but run automated vulnerability scanning and exploitation. They are often only a nuisance, however compromise by one of these actors is a major risk to an organization's reputation.

The following practices can help mitigate some of the risks identified above:

- **Security Updates:** You must consider the end-to-end security posture of your underlying physical infrastructure, including networking, storage, and server hardware. These systems will require their own security hardening practices. For your Red Hat Ceph Storage deployment, you should have a plan to regularly test and deploy security updates.
- **Access Management:** Access management includes authentication, authorization and accounting. Authentication is the process of verifying the user identity. Authorization is the process of granting permissions to an authenticated user. Accounting is the process of tracking which user performed an action. When granting system access to users, apply the *principle of*

least privilege, and only grant users the granular system privileges they actually need. This approach can also help mitigate the risks of both malicious actors and typographical errors from system administrators.

- **Manage Insiders:** You can help mitigate the threat of malicious insiders by applying careful assignment of role-based access control (minimum required access), using encryption on internal interfaces, and using authentication/authorization security (such as centralized identity management). You can also consider additional non-technical options, such as separation of duties and irregular job role rotation.

2.2. SECURITY ZONES

A security zone comprises users, applications, servers or networks that share common trust requirements and expectations within a system. Typically they share the same authentication and authorization requirements and users. Although you may refine these zone definitions further, this guide refers to four distinct security zones, three of which form the bare minimum that is required to deploy a security-hardened Red Hat Ceph Storage cluster. These security zones are listed below from least to most trusted:

- **Public Security Zone:** The public security zone is an entirely untrusted area of the cloud infrastructure. It can refer to the Internet as a whole or simply to networks that are external to your Red Hat OpenStack deployment over which you have no authority. Any data with confidentiality or integrity requirements that traverse this zone should be protected using compensating controls such as encryption. The public security zone **SHOULD NOT** be confused with the Ceph Storage Cluster's front- or client-side network, which is referred to as the **public_network** in RHCS and is usually **NOT** part of the public security zone or the Ceph client security zone.
- **Ceph Client Security Zone:** With RHCS, the Ceph client security zone refers to networks accessing Ceph clients such as Ceph Object Gateway, Ceph Block Device, Ceph Filesystem, or **librados**. The Ceph client security zone is typically behind a firewall separating itself from the public security zone. However, Ceph clients are not always protected from the public security zone. It is possible to expose the Ceph Object Gateway's S3 and Swift APIs in the public security zone.
- **Storage Access Security Zone:** The storage access security zone refers to internal networks providing Ceph clients with access to the Ceph Storage Cluster. We use the phrase 'storage access security zone' so that this document is consistent with the terminology used in the OpenStack Platform Security and Hardening Guide. The storage access security zone includes the Ceph Storage Cluster's front- or client-side network, which is referred to as the **public_network** in RHCS.
- **Ceph Cluster Security Zone:** The Ceph cluster security zone refers the internal networks providing the Ceph Storage Cluster's OSD daemons with network communications for replication, heartbeating, backfilling, and recovery. The Ceph cluster security zone includes the Ceph Storage Cluster's backside network, which is referred to as the **cluster_network** in RHCS.

These security zones can be mapped separately, or combined to represent the majority of the possible areas of trust within a given RHCS deployment. Security zones should be mapped out against your specific RHCS deployment topology. The zones and their trust requirements will vary depending upon whether Red Hat Ceph Storage is operating in a standalone capacity or is serving a public, private, or hybrid cloud.

For a visual representation of these security zones, see [Security Optimized Architecture](#).

Additional Resources

- See the [Network Communications](#) section in the *Red Hat Ceph Storage Data Security and Hardening Guide* for more details.

2.3. CONNECTING SECURITY ZONES

Any component that spans across multiple security zones with different trust levels or authentication requirements must be carefully configured. These connections are often the weak points in network architecture, and should always be configured to meet the security requirements of the highest trust level of any of the zones being connected. In many cases the security controls of the connected zones should be a primary concern due to the likelihood of attack. The points where zones meet do present an opportunity for attackers to migrate or target their attack to more sensitive parts of the deployment.

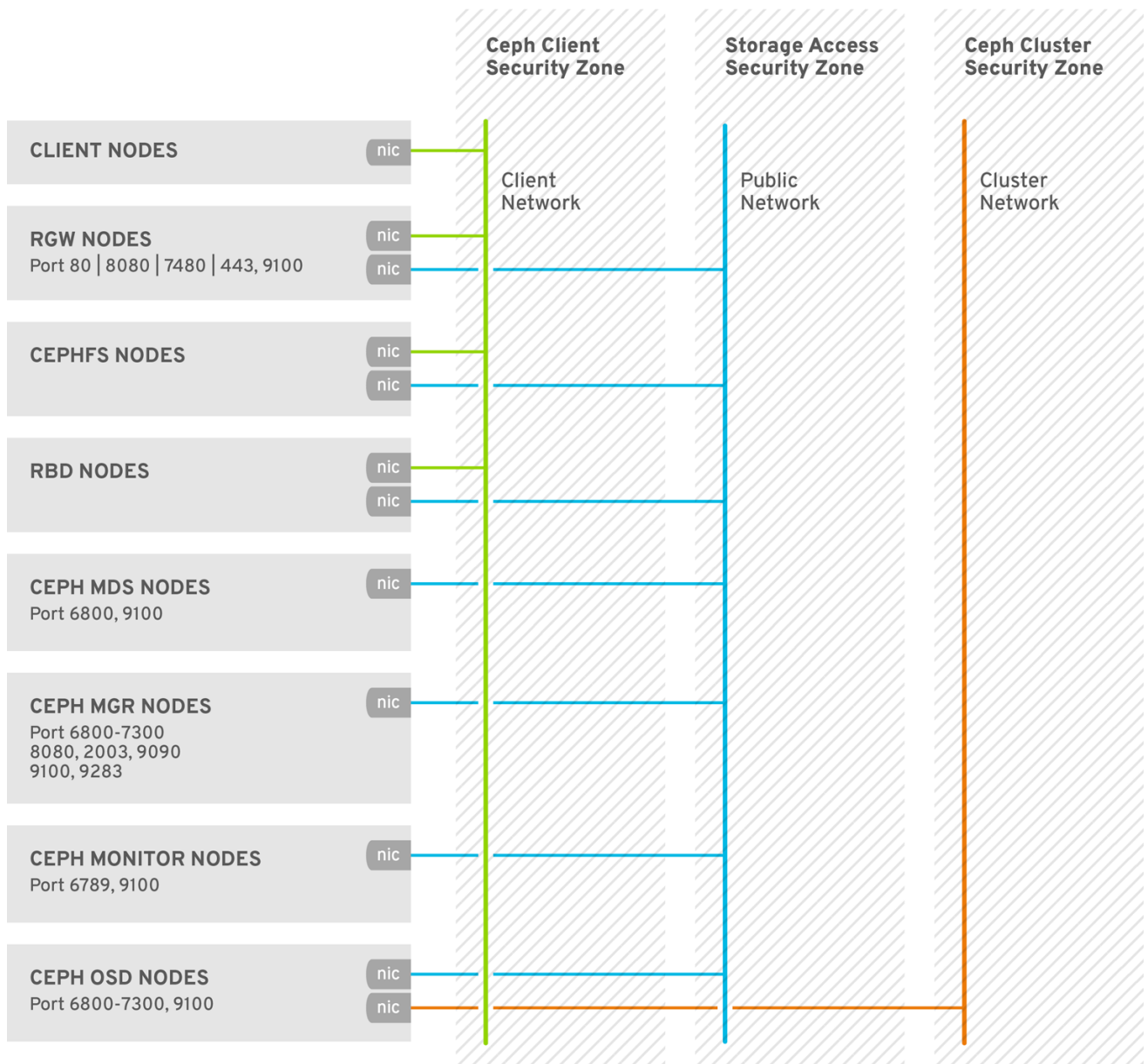
In some cases, Red Hat Ceph Storage administrators might want to consider securing integration points at a higher standard than any of the zones in which the integration point resides. For example, the Ceph Cluster Security Zone can be isolated from other security zones easily, because there is no reason for it to connect to other security zones. By contrast, the Storage Access Security Zone must provide access to port **6789** on Ceph monitor nodes, and ports **6800-7300** on Ceph OSD nodes. However, port **3000** should be exclusive to the Storage Access Security Zone, because it provides access to Ceph Graphana monitoring information that should be exposed to Ceph administrators only. A Ceph Object Gateway in the Ceph Client Security Zone will need to access the Ceph Cluster Security Zone's monitors (port **6789**) and OSDs (ports **6800-7300**), and may expose its S3 and Swift APIs to the Public Security Zone such as over HTTP port **80** or HTTPS port **443**; yet, it may still need to restrict access to the admin API.

The design of Red Hat Ceph Storage is such that separation of security zones is difficult. Because core services will usually span at least two zones, special consideration must be given when applying security controls to them.

2.4. SECURITY-OPTIMIZED ARCHITECTURE

A Red Hat Ceph Storage cluster's daemons typically run on nodes that are subnet isolated and behind a firewall, which makes it relatively simple to secure an RHCS cluster.

By contrast, Red Hat Ceph Storage clients such as Ceph Block Device (**rbd**), Ceph Filesystem (**cephfs**) and Ceph Object Gateway (**rgw**) access the RHCS storage cluster, but expose their services to other cloud computing platforms.



CEPH_476225_0818

CHAPTER 3. ENCRYPTION AND KEY MANAGEMENT

The Red Hat Ceph Storage cluster typically resides in its own network security zone, especially when using a private storage cluster network.



IMPORTANT

Security zone separation may be insufficient for protection if an attacker gains access to Ceph clients on the public network.

There are situations where there is a security requirement to assure the confidentiality or integrity of network traffic, and where Red Hat Ceph Storage uses encryption and key management, including:

- SSH
- SSL Termination
- Encryption in Transit
- Encryption at Rest

3.1. SSH

All nodes in the RHCS cluster use SSH as part of deploying the cluster. This means that on each node:

- An Ansible user exists with password-less root privileges.
- The SSH service is enabled and by extension port 22 is open.
- A copy of the Ansible user's public SSH key is available.



IMPORTANT

Any person with access to the Ansible user by extension has permission to exercise CLI commands as root on any node in the RHCS cluster.

See [Creating an Ansible User with `sudo` Access](#) and [Enabling Password-less SSH for Ansible](#) for additional details.

3.2. SSL TERMINATION

The Ceph Object Gateway may be deployed in conjunction with HAProxy and **keepalived** for load balancing and failover. The object gateway Red Hat Ceph Storage version 2 and 3 uses Civetweb. Earlier versions of Civetweb do not support SSL and later versions support SSL with some performance limitations. When using HAProxy and **keepalived** to terminate SSL connections, the HAProxy and **keepalived** components use encryption keys.

When using HAProxy and **keepalived** to terminate SSL, the connection between the load balancer and the Ceph Object Gateway is **NOT** encrypted.

See [Using SSL with Civetweb](#) and [HAProxy/keepalived Configuration](#) for details.

3.3. ENCRYPTION IN TRANSIT

Starting with Red Hat Ceph Storage 4 and later, you can enable encryption for all Ceph traffic over the network with the introduction of the messenger version 2 protocol. The **secure** mode setting for messenger v2 encrypts communication between Ceph daemons and Ceph clients, giving you end-to-end encryption.

Messenger v2 Protocol

The second version of Ceph's on-wire protocol, **msgr2**, includes several new features:

- A secure mode encrypting all data moving through the network.
- Encapsulation improvement of authentication payloads.
- Improvements to feature advertisement and negotiation.

The Ceph daemons bind to multiple ports allowing both the legacy, v1-compatible, and the new, v2-compatible, Ceph clients to connect to the same storage cluster. Ceph clients or other Ceph daemons connecting to the Ceph Monitor daemon will try to use the **v2** protocol first, if possible, but if not, then the legacy **v1** protocol will be used. By default, both messenger protocols, **v1** and **v2**, are enabled. The new v2 port is 3300, and the legacy v1 port is 6789, by default.

The **msgr2** protocol supports two connection modes:

- **crc**
 - Provides strong initial authentication when a connection is established with **cephx**.
 - Provides a **crc32c** integrity check to protect against bit flips.
 - Does not provide protection against a malicious man-in-the-middle attack.
 - Does not prevent an eavesdropper from seeing all post-authentication traffic.
- **secure**
 - Provides strong initial authentication when a connection is established with **cephx**.
 - Provides full encryption of all post-authentication traffic.
 - Provides a cryptographic integrity check.

The default mode is **crc**.

Ceph Object Gateway Encryption

Also, the Ceph Object Gateway supports encryption with customer-provided keys using its S3 API.



IMPORTANT

To comply with regulatory compliance standards requiring strict encryption in transit, administrators **MUST** deploy the Ceph Object Gateway with client-side encryption.

Ceph Block Device Encryption

System administrators integrating Ceph as a backend for Red Hat OpenStack Platform 13 **MUST** encrypt Ceph block device volumes using **dm_crypt** for RBD Cinder to ensure on-wire encryption within the Ceph storage cluster.



IMPORTANT

To comply with regulatory compliance standards requiring strict encryption in transit, system administrators **MUST** use **dmccrypt** for RBD Cinder to ensure on-wire encryption within the Ceph storage cluster.

Additional Resources

- See the [Red Hat Ceph Storage Data Security and Hardening Guide](#) for details on SSL termination.
- See the [Red Hat Ceph Storage Developer Guide](#) for details on S3 API encryption.

3.3.1. Enabling the messenger v2 protocol

For new installations of Red Hat Ceph Storage 4, the messenger v2 protocol, **msgr2**, is enabled by default. For Red Hat Ceph Storage 3 or earlier, the Ceph Monitors bind to the legacy v1 port **6789**. After upgrading you can enable the **msgr2** protocol to take advantage of its new features. Once the Ceph Monitors bind to the **msgr2** protocol they start advertising a v2 address, after restarting the Ceph Monitor service. The **msgr2** protocol's default connection mode is **crc**

Ensure that you consider cluster CPU requirements when you plan the Red Hat Ceph Storage cluster, to include encryption overhead.



IMPORTANT

Using **secure** mode is currently supported by Ceph kernel clients, such as CephFS and **krbd** on Red Hat Enterprise Linux 8.2. Using **secure** mode is supported by Ceph clients using **librbd**, such as OpenStack Nova, Glance, and Cinder.

Prerequisites

- A running Red Hat Ceph Storage 4 cluster.
- Open TCP port 3300 in the firewall.
- Root-level access to the Ceph Monitor nodes.

Procedure

1. Open the Ceph configuration file, by default **/etc/ceph/ceph.conf**, for editing.
2. Under the **[global]** section added the following to a new line:

```
ms_bind_msgr2 = true
```

- a. Optionally, to enable on-wire encryption between the Ceph daemons and to also enable on-wire encryption between Ceph clients and daemons, add the following options under the **[global]** sections of the Ceph configuration file:

```
[global]
ms_cluster_mode=secure
ms_service_mode=secure
ms_client_mode=secure
```

3. Save the changes to the Ceph configuration file.
4. Copy the updated Ceph configuration file to all nodes in the Red Hat Ceph Storage cluster.
5. Enable the **msgr2** protocol:

```
[root@mon ~]# ceph mon enable-msgr2
```

6. Restart the Ceph Monitor service on each Ceph Monitor node:

```
[root@mon ~]# systemctl restart ceph-mon.target
```

Additional Resources

- See the [Red Hat Ceph Storage Architecture Guide](#) for more details on the **msgr2** protocol.

3.4. ENCRYPTION AT REST

Red Hat Ceph Storage supports encryption at rest in a few scenarios:

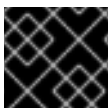
1. **Ceph Storage Cluster:** The Ceph Storage Cluster supports Linux Unified Key Setup or LUKS encryption of OSDs and their corresponding journals, write-ahead logs, and metadata databases. In this scenario, Ceph will encrypt all data at rest irrespective of whether the client is a Ceph Block Device, Ceph Filesystem, Ceph Object Storage cluster or a custom application built on **librados**.
2. **Ceph Object Gateway:** The Ceph Storage Cluster supports encryption of client objects. When the Ceph Object Gateway encrypts objects, they are encrypted independently of the Red Hat Ceph Storage cluster. Additionally, the data transmitted between the Ceph Object Gateway and the Ceph Storage Cluster is in encrypted form.

Ceph Storage Cluster Encryption

The Ceph Storage Cluster supports encrypting data stored on OSDs. Red Hat Ceph Storage can encrypt logical volumes with **lvm** by specifying **dmccrypt**; that is, **lvm**, invoked by **ceph-volume**, encrypts an OSD's logical volume not its physical volume, and may encrypt non-LVM devices like partitions using the same OSD key. Encrypting logical volumes allows for more configuration flexibility.

Ceph uses LUKS v1 rather than LUKS v2, because LUKS v1 has the broadest support among Linux distributions.

When creating an OSD, **lvm** will generate a secret key and pass the key to the Ceph monitors securely in a JSON payload via **stdin**. The attribute name for the encryption key is **dmccrypt_key**.



IMPORTANT

System administrators must explicitly enable encryption.

By default, Ceph does not encrypt data stored in OSDs. System administrators must enable **dmccrypt** in Ceph Ansible. See the appendix on [OSD Ansible settings](#) in the *Red Hat Ceph Storage Installation Guide* for details on setting the **dmccrypt** option in the **group_vars/osds.yml** file.

**NOTE**

LUKS and **dmccrypt** only address encryption for data at rest, not encryption for data in transit.

Ceph Object Gateway Encryption

The Ceph Object Gateway supports encryption with customer-provided keys using its S3 API. When using customer-provided keys, the S3 client passes an encryption key along with each request to read or write encrypted data. It is the customer's responsibility to manage those keys. Customers must remember which key the Ceph Object Gateway used to encrypt each object.

See [S3 API server-side encryption](#) in the *Red Hat Ceph Storage Developer Guide* for details.

CHAPTER 4. IDENTITY AND ACCESS MANAGEMENT

Red Hat Ceph Storage provides identity and access management for:

- Ceph Storage Cluster User Access
- Ceph Object Gateway User Access
- Ceph Object Gateway LDAP/AD Authentication
- Ceph Object Gateway OpenStack Keystone Authentication

4.1. CEPH STORAGE CLUSTER USER ACCESS

To identify users and protect against man-in-the-middle attacks, Ceph provides its **cephx** authentication system to authenticate users and daemons. For additional details on **cephx**, see [Ceph user management](#).

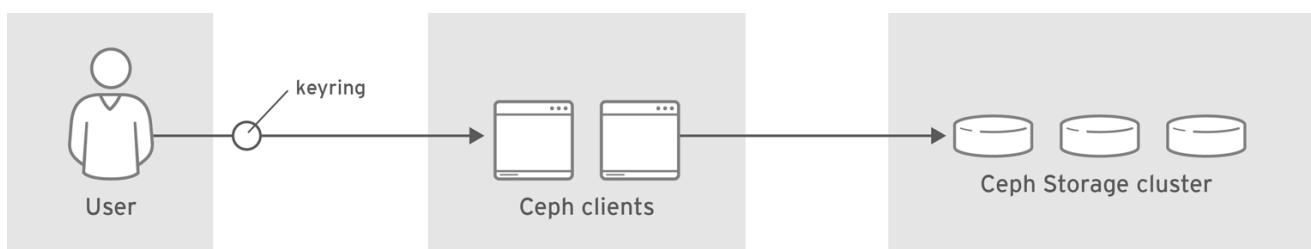


IMPORTANT

The **cephx** protocol **DOES NOT** address data encryption in transport or encryption at rest.

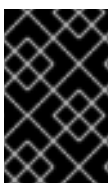
Cephx uses shared secret keys for authentication, meaning both the client and the monitor cluster have a copy of the client's secret key. The authentication protocol is such that both parties are able to prove to each other they have a copy of the key without actually revealing it. This provides mutual authentication, which means the cluster is sure the user possesses the secret key, and the user is sure that the cluster has a copy of the secret key.

Users are either individuals or system actors such as applications, which use Ceph clients to interact with the Red Hat Ceph Storage cluster daemons.



CEPH_459704_1017

Ceph runs with authentication and authorization enabled by default. Ceph clients may specify a user name and a keyring containing the secret key of the specified user—usually by using the command line. If the user and keyring are not provided as arguments, Ceph will use the **client.admin** administrative user as the default. If a keyring is not specified, Ceph will look for a keyring by using the **keyring** setting in the Ceph configuration.



IMPORTANT

To harden a Ceph cluster, keyrings **SHOULD ONLY** have read and write permissions for the current user and **root**. The keyring containing the **client.admin** administrative user key must be restricted to the **root** user.

For details on configuring the Red Hat Ceph Storage cluster to use authentication, see [Configuration Guide](#) for Red Hat Ceph Storage 4. More specifically, see [CephX Configuration Reference](#).

4.2. CEPH OBJECT GATEWAY USER ACCESS

The Ceph Object Gateway provides a RESTful API service with its own user management that authenticates and authorizes users to access S3 and Swift APIs containing user data. Authentication consists of:

- **S3 User:** An access key and secret for a user of the S3 API.
- **Swift User:** An access key and secret for a user of the Swift API. The Swift user is a subuser of an S3 user. Deleting the S3 'parent' user will delete the Swift user.
- **Administrative User:** An access key and secret for a user of the administrative API. Administrative users should be created sparingly, as the administrative user will be able to access the Ceph Admin API and execute its functions, such as creating users, and giving them permissions to access buckets or containers and their objects among other things.

The Ceph Object Gateway stores all user authentication information in Ceph Storage cluster pools. Additional information may be stored about users including names, email addresses, quotas and usage.

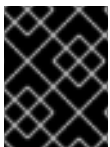
For additional details, see [User Management](#) and [Creating an Administrative User](#).

4.3. CEPH OBJECT GATEWAY LDAP/AD AUTHENTICATION

Red Hat Ceph Storage supports Light-weight Directory Access Protocol (LDAP) servers for authenticating Ceph Object Gateway users. When configured to use LDAP or Active Directory, Ceph Object Gateway defers to an LDAP server to authenticate users of the Ceph Object Gateway.

Ceph Object Gateway controls whether to use LDAP. However, once configured, it is the LDAP server that is responsible for authenticating users.

To secure communications between the Ceph Object Gateway and the LDAP server, Red Hat recommends deploying configurations with LDAP Secure or LDAPS.



IMPORTANT

When using LDAP, ensure that access to the `rgw_ldap_secret = <path-to-secret>` secret file is secure.

For additional details, see the [Ceph Object Gateway with LDAP/AD Guide](#).

4.4. CEPH OBJECT GATEWAY OPENSTACK KEYSTONE AUTHENTICATION

Red Hat Ceph Storage supports using OpenStack Keystone to authenticate Ceph Object Gateway Swift API users. The Ceph Object Gateway can accept a Keystone token, authenticate the user and create a corresponding Ceph Object Gateway user. When Keystone validates a token, the Ceph Object Gateway considers the user authenticated.

Ceph Object Gateway controls whether to use OpenStack Keystone for authentication. However, once configured, it is the OpenStack Keystone service that is responsible for authenticating users.

Configuring the Ceph Object Gateway to work with Keystone requires converting the OpenSSL certificates that Keystone uses for creating the requests to the **nss db** format.

See [Using Keystone to Authenticate Ceph Object Gateway Users](#) for details.

CHAPTER 5. INFRASTRUCTURE SECURITY

The scope of this guide is Red Hat Ceph Storage. However, a proper Red Hat Ceph Storage security plan **requires** consideration of the [Red Hat Enterprise Linux 7 Security Guide](#) and the [Red Hat Enterprise Linux 7 SELinux Users and Administration Guide](#), which by the foregoing hyperlinks are incorporated herein.



WARNING

No security plan for Red Hat Ceph Storage is complete without consideration of the foregoing guides.

5.1. ADMINISTRATION

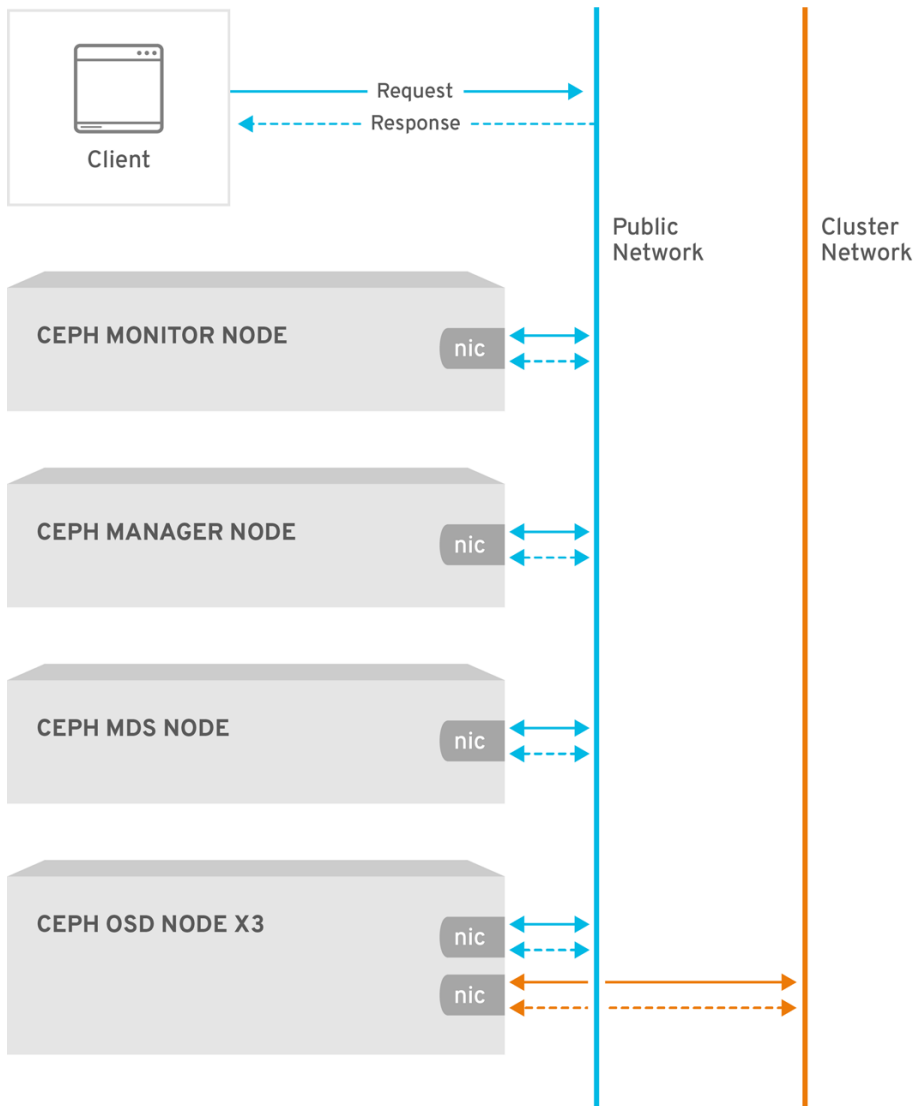
Administering a Red Hat Ceph Storage cluster involves using command line tools. The CLI tools require an administrator key for administrator access privileges to the cluster. By default, Ceph stores the administrator key in the `/etc/ceph` directory. The default file name is `ceph.client.admin.keyring`. Take steps to secure the keyring so that only a user with administrative privileges to the cluster may access the keyring.

5.2. NETWORK COMMUNICATION

Red Hat Ceph Storage provides two networks:

- A public network, and
- A cluster network.

All Ceph daemons and Ceph clients require access to the public network, which is part of the *storage access security zone*. By contrast, **ONLY** the OSD daemons require access to the cluster network, which is part of the *Ceph cluster security zone*.



CEPH_471750_0518

The Ceph configuration contains **public_network** and **cluster_network** settings. For hardening purposes, specify the IP address and the netmask using CIDR notation. Specify multiple comma-delimited IP/netmask entries if the cluster will have multiple subnets.

```
public_network = <public-network/netmask>[,<public-network/netmask>]
cluster_network = <cluster-network/netmask>[,<cluster-network/netmask>]
```

See the [Network Configuration Reference](#) of the Configuration Guide for details.

5.3. HARDENING THE NETWORK SERVICE

System administrators deploy Red Hat Ceph Storage clusters on Red Hat Enterprise Linux 7 Server. SELinux is on by default and the firewall blocks all inbound traffic except for the SSH service port **22**; however, you **MUST** ensure that this is the case so that no other unauthorized ports are open or unnecessary services enabled.

On each server node, execute the following:

1. Start the **firewalld** service, enable it to run on boot and ensure that it is running:

```
# systemctl enable firewalld
# systemctl start firewalld
# systemctl status firewalld
```

2. Take an inventory of all open ports.

```
# firewall-cmd --list-all
```

On a new installation, the **sources:** section should be blank indicating that no ports have been opened specifically. The **services** section should indicate **ssh** indicating that the SSH service (and port **22**) and **dhcpv6-client** are enabled.

```
sources:
services: ssh dhcpv6-client
```

3. Ensure SELinux is running and **Enforcing**.

```
# getenforce
Enforcing
```

If SELinux is **Permissive**, set it to **Enforcing**.

```
# setenforce 1
```

If SELinux is not running, enable it. See the [Red Hat Enterprise Linux 7 SELinux Users and Administration Guide](#) for details.

Each Ceph daemon uses one or more ports to communicate with other daemons in the Red Hat Ceph Storage cluster. In some cases, you may change the default port settings. Administrators typically only change the default port with the Ceph Object Gateway or **ceph-radosgw** daemon. See *Changing the CivetWeb port* in the [Object Gateway Configuration and Administration Guide](#).

Table 5.1. Ceph Ports

Port	Daemon	Configuration Option
6789, 3300	ceph-mon	N/A
6800-7300	ceph-osd	ms_bind_port_min to ms_bind_port_max
6800-7300	ceph-mgr	ms_bind_port_min to ms_bind_port_max
6800	ceph-mds	N/A
8080	ceph-radosgw	rgw_frontends

The Ceph Storage Cluster daemons include **ceph-mon**, **ceph-mgr** and **ceph-osd**. These daemons and their hosts comprise the Ceph cluster security zone, which should use its own subnet for hardening purposes.

The Ceph clients include **ceph-radosgw**, **ceph-mds**, **ceph-fuse**, **libcephfs**, **rbd**, **librbd** and **librados**. These daemons and their hosts comprise the storage access security zone, which should use its own subnet for hardening purposes.

On the Ceph Storage Cluster zone's hosts, consider enabling only hosts running Ceph clients to connect to the Ceph Storage Cluster daemons. For example:

```
# firewall-cmd --zone=<zone-name> --add-rich-rule="rule family="ipv4" \  
source address="<ip-address>/<netmask>" port protocol="tcp" \  
port="<port-number>" accept"
```

Replace **<zone-name>** with the zone name. Replace the **<ipaddress>** with the IP address and **<netmask>** with the subnet mask in CIDR notation. Replace the **<port-number>** with the port number or range. Repeat the process with the **--permanent** flag so that the changes persist after reboot. For example:

```
# firewall-cmd --zone=<zone-name> --add-rich-rule="rule family="ipv4" \  
source address="<ip-address>/<netmask>" port protocol="tcp" \  
port="<port-number>" accept" --permanent
```

See the [Firewalls](#) section of the Red Hat Ceph Storage installation guide for specific steps.

5.4. REPORTING

Red Hat Ceph Storage provides basic system monitoring and reporting with the **ceph-mgr** daemon plug-ins; namely, the RESTful API, the dashboard, and other plug-ins such as Prometheus and Zabbix. Ceph collects this information using **collectd** and sockets to retrieve settings, configuration details and statistical information.

In addition to default system behavior, system administrators may configure **collectd** to report on security matters such as configuring the **IP-Tables** or **ConnTrack** plug ins to track open ports and connections respectively.

System administrators may also retrieve configuration settings at runtime. See [Viewing the Ceph Runtime Configuration](#).

5.5. AUDITING ADMINISTRATOR ACTIONS

An important aspect of system security is to periodically audit administrator actions on the cluster. Red Hat Ceph Storage stores a history of administrator actions in the **/var/log/ceph/ceph.audit.log** file.

Each entry will contain:

- **Timestamp:** Indicates when the command was executed.
- **Monitor Address:** Identifies the monitor modified.
- **Client Node:** Identifies the client node initiating the change.
- **Entity:** Identifies the user making the change.
- **Command:** Identifies the command executed.

For example, a system administrator may set and unset the **nodown** flag. In the audit log, it will look something like this:

```
2018-08-13 21:50:28.723876 mon.reesi003 mon.2 172.21.2.203:6789/0 2404194 : audit [INF]  
from='client.? 172.21.6.108:0/4077431892' entity='client.admin' cmd=[{"prefix": "osd set", "key":
```



```
"nodown"}]: dispatch
2018-08-13 21:50:28.727176 mon.reesi001 mon.0 172.21.2.201:6789/0 2097902 : audit [INF]
from='client.348389421 -' entity='client.admin' cmd=[{"prefix": "osd set", "key": "nodown"}]: dispatch
2018-08-13 21:50:28.872992 mon.reesi001 mon.0 172.21.2.201:6789/0 2097904 : audit [INF]
from='client.348389421 -' entity='client.admin' cmd=[{"prefix": "osd set", "key": "nodown"}]: finished
2018-08-13 21:50:31.197036 mon.mira070 mon.5 172.21.6.108:6789/0 413980 : audit [INF]
from='client.? 172.21.6.108:0/675792299' entity='client.admin' cmd=[{"prefix": "osd unset", "key":
"nodown"}]: dispatch
2018-08-13 21:50:31.252225 mon.reesi001 mon.0 172.21.2.201:6789/0 2097906 : audit [INF]
from='client.347227865 -' entity='client.admin' cmd=[{"prefix": "osd unset", "key": "nodown"}]: dispatch
2018-08-13 21:50:31.887555 mon.reesi001 mon.0 172.21.2.201:6789/0 2097909 : audit [INF]
from='client.347227865 -' entity='client.admin' cmd=[{"prefix": "osd unset", "key": "nodown"}]: finished
```

In distributed systems such as Ceph, actions may begin on one instance and get propagated to other nodes in the cluster. When the action begins, the log indicates **dispatch**. When the action ends, the log indicates **finished**.

In the foregoing example, **entity='client.admin'** indicates that the user is the **admin** user. The command **cmd=[{"prefix": "osd set", "key": "nodown"}]** indicates that the admin user executed **ceph osd set nodown**.

CHAPTER 6. DATA RETENTION

Red Hat Ceph Storage stores user data, but usually in an indirect manner. Customer data retention may involve other applications such as the Red Hat OpenStack Platform.

6.1. CEPH STORAGE CLUSTER

The Ceph Storage Cluster—often referred to as the Reliable Autonomic Distributed Object Store or RADOS—stores data as objects within pools. In most cases, these objects are the atomic units representing client data such as Ceph Block Device images, Ceph Object Gateway objects, or Ceph Filesystem files. However, custom applications built on top of **librados** may bind to a pool and store data too.

Cephx controls access to the pools storing object data. However, Ceph Storage Cluster users are typically Ceph clients, and not end users. Consequently, end users generally **DO NOT** have the ability to write, read or delete objects directly in a Ceph Storage Cluster pool.

6.2. CEPH BLOCK DEVICE

The most popular use of Red Hat Ceph Storage, the Ceph Block Device interface, also referred to as RADOS Block Device or RBD, creates virtual volumes, images and compute instances and stores them as a series of objects within pools. Ceph assigns these objects to placement groups and distributes or places them pseudo-randomly in OSDs throughout the cluster.

Depending upon the application consuming the Ceph Block Device interface—usually Red Hat OpenStack Platform—end users may create, modify and delete volumes and images. Ceph handles the CRUD operations of each individual object.

Deleting volumes and images destroys the corresponding objects in an unrecoverable manner. However, residual data artifacts may continue to reside on storage media until overwritten. Data may also remain in back up archives.

6.3. CEPH FILESYSTEM

The Ceph Filesystem interface creates virtual filesystems and stores them as a series of objects within pools. Ceph assigns these objects to placement groups and distributes or places them pseudo-randomly in OSDs throughout the cluster.

Typically, the Ceph Filesystem uses two pools:

- **Metadata:** The metadata pool stores the data of the metadata server (mds), which generally consists of inodes; that is, the file ownership, permissions, creation date/time, last modified/accessed date/time, parent directory, etc.
- **Data:** The data pool stores file data. Ceph may store a file as one or more objects, typically representing smaller chunks of file data such as extents.

Depending upon the application consuming the Ceph Filesystem interface—usually Red Hat OpenStack Platform—end users may create, modify and delete files in a Ceph filesystem. Ceph handles the CRUD operations of each individual object representing the file.

Deleting files destroys the corresponding objects in an unrecoverable manner. However, residual data artifacts may continue to reside on storage media until overwritten. Data may also remain in back up archives.

6.4. CEPH OBJECT GATEWAY

From a data security and retention perspective, the Ceph Object Gateway interface has some important differences when compared to the Ceph Block Device and Ceph Filesystem interfaces. The Ceph Object Gateway provides a service to end users. So the Ceph Object Gateway may store:

- **User Authentication Information:** User authentication information generally consists of user IDs, user access keys and user secrets. It may also comprise a user's name and email address if provided. Ceph Object Gateway will retain user authentication data unless the user is explicitly deleted from the system.
- **User Data:** User data generally comprises user- or administrator-created buckets or containers, and the user-created S3 or Swift objects contained within them. The Ceph Object Gateway interface creates one or more Ceph Storage cluster objects for each S3 or Swift object and stores the corresponding Ceph Storage cluster objects within a data pool. Ceph assigns the Ceph Storage cluster objects to placement groups and distributes or places them pseudo-randomly in OSDs throughout the cluster. The Ceph Object Gateway may also store an index of the objects contained within a bucket or index to enable services such as listing the contents of an S3 bucket or Swift container. Additionally, when implementing multi-part uploads, the Ceph Object Gateway may temporarily store partial uploads of S3 or Swift objects. End users may create, modify and delete buckets or containers, and the objects contained within them in a Ceph Object Gateway. Ceph handles the CRUD operations of each individual Ceph Storage cluster object representing the S3 or Swift object.

Deleting S3 or Swift objects destroys the corresponding Ceph Storage cluster objects in an unrecoverable manner. However, residual data artifacts may continue to reside on storage media until overwritten. Data may also remain in back up archives.

- **Logging:** Ceph Object Gateway also stores logs of user operations that the user intends to accomplish and operations that have executed. This data provides traceability about who created, modified or deleted a bucket or container, or an S3 or Swift object residing in a an S3 bucket or Swift container. When users delete their data, the logging information is not effected and will remain in storage until deleted by a system administrator or removed automatically by expiration policy.

Bucket Lifecycle

Ceph Object Gateway also supports bucket lifecycle features, including object expiration. Data retention regulations like the General Data Protection Regulation may require administrators to set object expiration policies and disclose them to end users among other compliance factors.

Multisite

Ceph Object Gateway is often deployed in a multi-site context whereby a user stores an object at one site and the Ceph Object Gateway creates a replica of the object in another cluster possibly at another geographic location. For example, if a primary cluster fails, a secondary cluster may resume operations. In another example, a secondary cluster may be in a different geographic location, such as an edge network or content-delivery network such that a client may access the closest cluster to improve response time, throughput and other performance characteristics. In multisite scenarios, administrators must ensure that each site has implemented security measures. Additionally, if geographic distribution of data would occur in a multisite scenario, administrators must be aware of any regulatory implications when the data crosses political boundaries.

CHAPTER 7. FEDERAL INFORMATION PROCESSING STANDARD (FIPS)

Red Hat Ceph Storage uses FIPS validated cryptography modules when run on Red Hat Enterprise Linux 7.6 or Red Hat Enterprise Linux 8.1, or Red Hat Enterprise Linux 8.2.

- Enable FIPS mode on Red Hat Enterprise Linux either during system installation or after it.
 - For bare-metal deployments, follow the instructions in the [Red Hat Enterprise Linux 8 Security Hardening Guide](#).
 - For container deployments, follow the instructions in the [Red Hat Enterprise Linux 8 Security Hardening Guide](#).

Additional Resources

- Refer to the [US Government Standards](#) to know the latest information on FIPS validations.

CHAPTER 8. SUMMARY

This document has provided only a general introduction to security for Red Hat Ceph Storage. Contact the Red Hat Ceph Storage consulting team for additional help.