Red Hat Ceph Storage 7

Data Security and Hardening Guide

Red Hat Ceph Storage Data Security and Hardening Guide

Last Updated: 2024-03-08

Red Hat Ceph Storage 7 Data Security and Hardening Guide

Red Hat Ceph Storage Data Security and Hardening Guide

Legal Notice

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 TO DATA SECURITY

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 data privacy and security. 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 Red Hat Ceph Storage, with a focus on the Ceph Orchestrator using **cephadm** for Red Hat Ceph Storage 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 RED HAT CEPH STORAGE

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.

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 such as establishing an agreement about the state of the cluster, maintaining a history of the state of the cluster such as whether an OSD is up and running and in the cluster, providing a list of pools through which clients write and read data, and providing 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 c hanges, among other functions.

All RHCS deployments store end-user data in the Ceph Storage Cluster or RADOS (Reliable Autonomous Distributed Object Store). Generally, 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 (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 File System (cephfs):** The Ceph File System consists of one or more Metadata Servers

(**mds**), which store the inode portion of a file system as objects on the Ceph Storage Cluster. Ceph file systems 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, 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 is typically deployed in conjunction with cloud computing solutions, so it can be helpful to think about a 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's impossible to secure a Red Hat Ceph Storage deployment against all 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's difficult to defend against these actors without 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 'Hacktivist' 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 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.

- **Product Updates:** Red Hat recommends running product updates as they become available. Updates are typically released every six weeks (and occasionally more frequently). Red Hat endeavors to make point releases and z-stream releases fully compatible within a major release in order to not require additional integration testing.

- **Access Management:** Access management includes authentication, authorization, and

accounting. Authentication is the process of verifying the user's 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, 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 to 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.

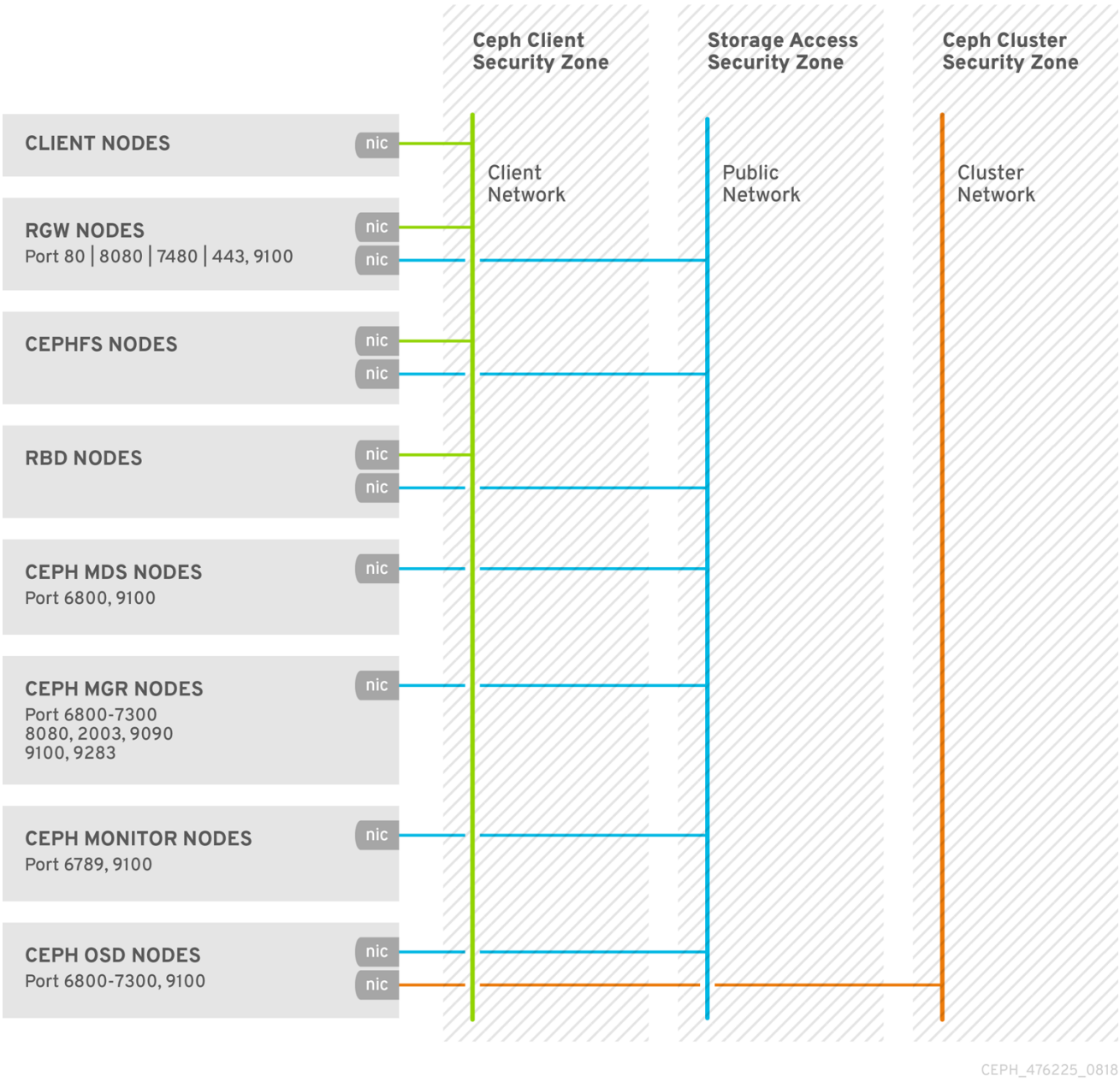
In some cases, Red Hat Ceph Storage administrators might want to consider securing integration points at a higher standard than any of the zones in which the integration point resides. For example, the Ceph Cluster Security Zone can be isolated from other security zones easily, because there is no reason 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 Grafana 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 the separation of security zones is difficult. As core services usually span at least two zones, special consideration must be given when applying security controls to them.

2.4. SECURITY-OPTIMIZED ARCHITECTURE

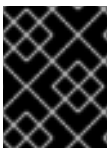
A Red Hat Ceph Storage cluster's daemons typically run on nodes that are subnet isolated and behind a firewall, which makes it relatively simple to secure an RHCS cluster.

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 Client Security Zone
Storage Access Security Zone
Ceph Cluster Security Zone
CLIENT NODES
nic
RGW NODES
Port 80 | 8080 | 7480 | 443, 9100
nic
nic
CEPHFS NODES
nic
nic
RBD NODES
nic
nic
CEPH MDS NODES
Port 6800, 9100
nic
CEPH MGR NODES
Port 6800-7300
8080, 2003, 9090
9100, 9283
nic
CEPH MONITOR NODES
Port 6789, 9100
nic
CEPH OSD NODES
Port 6800-7300, 9100
nic
nic
Client Network
Public Network
Cluster Network
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 might be insufficient for protection if an attacker gains access to Ceph clients on the public network.

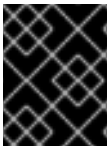
There are situations where there is a security requirement to assure the confidentiality or integrity of network traffic, and where Red Hat Ceph Storage uses encryption and key management, including:

- SSH
- SSL Termination
- Messenger v2 protocol
- Encryption in Transit
- Encryption at Rest
- Key rotation

3.1. SSH

All nodes in the Red Hat Ceph Storage cluster use SSH as part of deploying the cluster. This means that on each node:

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



IMPORTANT

Any person with access to the **cephadm** user by extension has permission to run commands as **root** on any node in the Red Hat Ceph Storage cluster.

Additional Resources

- See the *How **cephadm** works* section in the *Red Hat Ceph Storage Installation Guide* for more information.

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 versions 2 and 3 use Civetweb. Earlier versions of Civetweb do not support SSL and later versions support SSL with some performance limitations.

The object gateway Red Hat Ceph Storage version 5 uses Beast. You can configure the Beast front-end web server to use the OpenSSL library to provide Transport Layer Security (TLS).

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 *Configuring SSL for Beast* and *HAProxy and keepalived* for details.

3.3. MESSENGER V2 PROTOCOL

The second version of Ceph's on-wire protocol, **msgr2**, has the following features:

- A secure mode encrypting all data moving through the network.
- Encapsulation improvement of authentication payloads, enabling future integration of new authentication modes.
- Improvements to feature advertisement and negotiation.

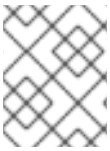
The Ceph daemons bind to multiple ports allowing both the legacy v1-compatible, and the new, v2-compatible Ceph clients to connect to the same storage cluster. Ceph clients or other Ceph daemons connecting to the Ceph Monitor daemon uses the **v2** protocol first, if possible, but if not, then the legacy **v1** protocol is 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 messenger v2 protocol has two configuration options that control whether the v1 or the v2 protocol is used:

- **ms_bind_msgr1** - This option controls whether a daemon binds to a port speaking the v1 protocol; it is **true** by default.
- **ms_bind_msgr2** - This option controls whether a daemon binds to a port speaking the v2 protocol; it is **true** by default.

Similarly, two options control based on IPv4 and IPv6 addresses used:

- **ms_bind_ipv4** - This option controls whether a daemon binds to an IPv4 address; it is **true** by default.
- **ms_bind_ipv6** - This option controls whether a daemon binds to an IPv6 address; it is **true** by default.



NOTE

The ability to bind to multiple ports has paved the way for dual-stack IPv4 and IPv6 support.

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

- **crc**
  - Provides strong initial authentication when a connection is established with **cephx**.

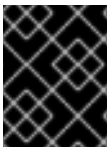
- Provides a **crc32c** integrity check to protect against bit flips.
- Does not provide protection against a malicious man-in-the-middle attack.
- Does not prevent an eavesdropper from seeing all post-authentication traffic.

- **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 **dmcrypt** for RBD Cinder to ensure on-wire encryption within the Ceph storage cluster.

Additional resources

- See the *Connection mode configuration options* in the *Red Hat Ceph Storage Configuration Guide* for more details.

3.4. ENCRYPTION IN TRANSIT

Starting with Red Hat Ceph Storage 5 and later, encryption for all Ceph traffic over the network is enabled by default, with the introduction of the messenger version 2 protocol. The **secure** mode setting for messenger v2 encrypts communication between Ceph daemons and Ceph clients, providing end-to-end encryption.

You can check for encryption of the messenger v2 protocol with the **ceph config dump** command, **netstat -lp | grep ceph-osd** command, or verify the Ceph daemon on the v2 ports.

Additional resources

- See the *SSL Termination* for details on SSL termination.

- See the *S3 server-side encryption* for details on S3 API encryption.

3.5. COMPRESSION MODES OF MESSENGER V2 PROTOCOL

Starting with Red Hat Ceph Storage 6 and later, messenger v2 protocol supports the compression feature.

This feature is not enabled by default. Compressing and encrypting the same message is not recommended as the level of security of messages between peers is reduced. If encryption is enabled, a request to enable compression is ignored until the configuration option **ms_osd_compress_mode** is set to **true**.

It supports two compression modes:

- **force**
  - In multi-availability zones deployment, compresses replication messages between OSDs saves latency.
  - In the public cloud, minimizes message size, thereby reducing network costs to cloud provider.
  - Instances on public clouds with NVMe provides low network bandwidth relative to the device bandwidth. Does not provide protection against a malicious man-in-the-middle attack.
- **none**
  - The messages are transmitted without compression.

To ensure that compression of the message is enabled, run the **debug_ms** command and check some debug entries for connections. Also, you can run the **ceph config get** command to get details about the different configuration options for the network messages.

Additional resources

- See the *Compression mode configuration options* in the *Red Hat Ceph Storage Configuration Guide* for more details.

3.6. 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 Ceph 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, 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 is 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 in Ceph OSDs. Red Hat Ceph Storage can

encrypt logical volumes with **lvm** by specifying **dmcrypt**; that is, **lvm**, invoked by **ceph-volume**, encrypts an OSD's logical volume, not its physical volume. It can 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 **dmcrypt_key**.

IMPORTANT

System administrators must explicitly enable encryption.

By default, Ceph does not encrypt data stored in Ceph OSDs. System administrators must enable **dmcrypt** to encrypt data stored in Ceph OSDs. When using a Ceph Orchestrator service specification file for adding Ceph OSDs to the storage cluster, set the following option in the file to encrypt Ceph OSDs:

Example

```
...
encrypted: true
...
```

NOTE

LUKS and **dmcrypt** 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.

Additional Resources

- See S3 API server-side encryption in the *Red Hat Ceph Storage Developer Guide* for details.

3.7. ENABLING KEY ROTATION

Ceph and Ceph Object Gateway daemons within the Ceph cluster have a secret key. This key is used to connect to and authenticate with the cluster. You can update an active security key within an active Ceph cluster with minimal service interruption, by using the key rotation feature.

NOTE

The active Ceph cluster includes nodes within the Ceph client role with parallel key changes.

Key rotation helps ensure that current industry and security compliance requirements are met.

Prerequisites

- A running Red Hat Ceph Storage cluster.
- User with **admin** privileges.

Procedure

1. Rotate the key:

   Syntax

   ```
   ceph orch daemon rotate-key NAME
   ```

   Example

   ```
   [ceph: root@host01 /]# ceph orch daemon rotate-key mgr.ceph-key-host01
   Scheduled to rotate-key mgr.ceph-key-host01 on host 'my-host-host01-installer'
   ```

2. If using a daemon other than MDS, OSD, or MGR, restart the daemon to switch to the new key. MDS, OSD, and MGR daemons do not require daemon restart.

   Syntax

   ```
   ceph orch restart SERVICE_TYPE
   ```

   Example

   ```
   [ceph: root@host01 /]# ceph orch restart rgw
   ```

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

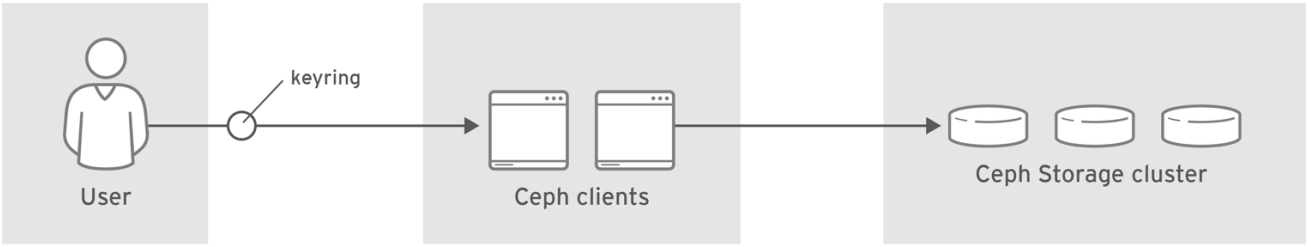
To identify users and protect against man-in-the-middle attacks, Ceph provides its **cephx** authentication system to authenticate users and daemons. For additional details on **cephx**, see *Ceph user management*.

IMPORTANT

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

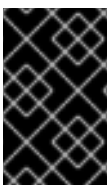
Cephx uses shared secret keys for authentication, meaning both the client and the monitor cluster have a copy of the client's secret key. The authentication protocol is such that both parties are able to prove to each other they have a copy of the key without actually revealing it. This provides mutual 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 the *Configuration Guide* for Red Hat Ceph Storage 7. More specifically, see *Ceph authentication configuration*.

4.2. CEPH OBJECT GATEWAY USER ACCESS

The Ceph Object Gateway provides a RESTful application programming interface (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 OR 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 (AD), 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_FILE*** secret file is secure.

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.

Additional Resources

- See *The Ceph Object Gateway and OpenStack Keystone* section of the *Red Hat Ceph Storage Object Gateway Guide* for more information.

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 following prerequisites.

Prerequisites

- Review the *Using SELinux Guide* within the Product Documentation for Red Hat Enterprise Linux for your OS version, on the Red Hat Customer Portal.
- Review the *Security Hardening Guide* within the Product Documentation for Red Hat Enterprise Linux for your OS version, on the Red Hat Customer Portal.

5.1. ADMINISTRATION

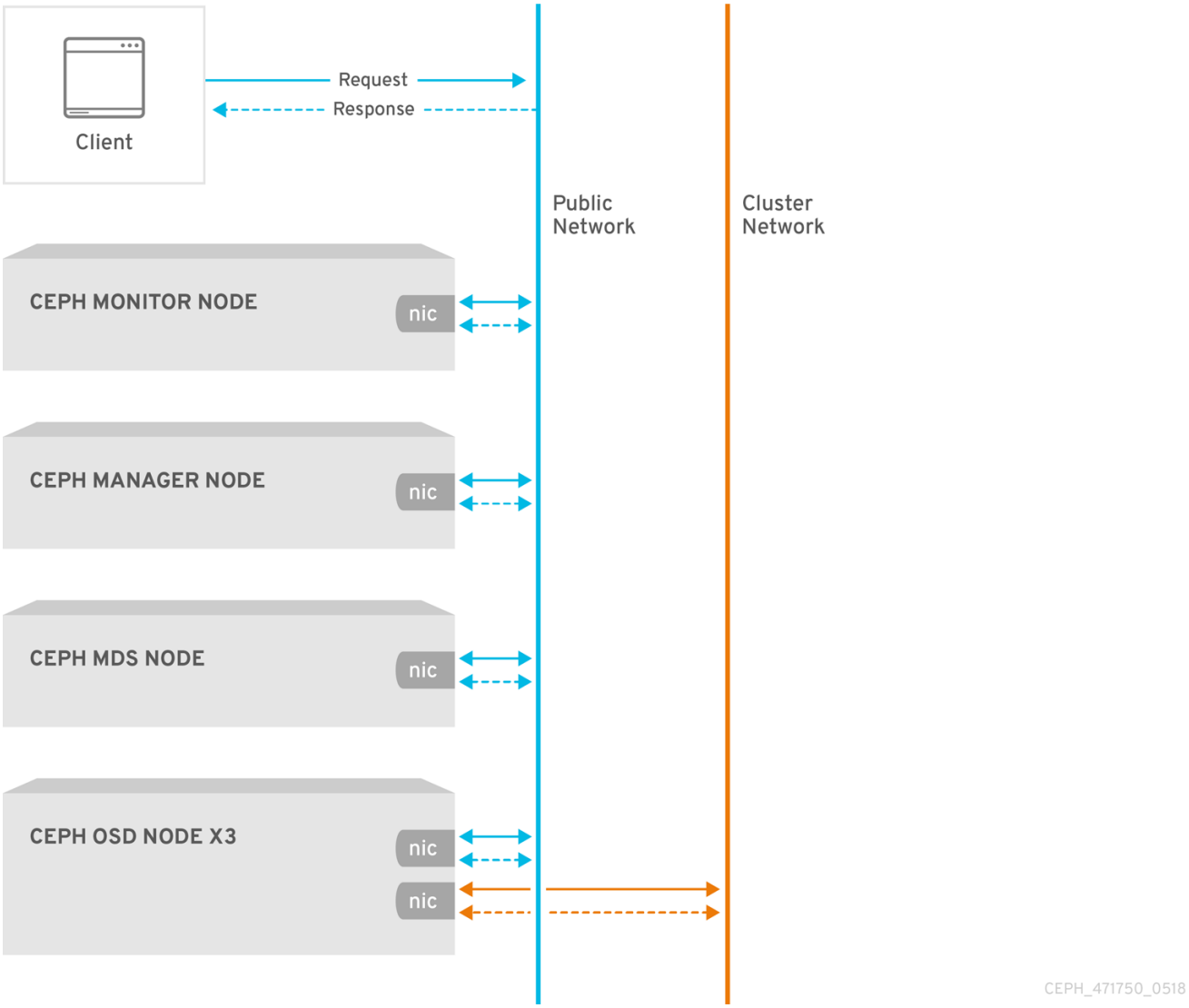
Administering a Red Hat Ceph Storage cluster involves using command line tools. The CLI tools require an administrator key for administrator access privileges to the cluster. By default, Ceph stores the administrator key in the **/etc/ceph** directory. The default file name is **ceph.client.admin.keyring**. 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.
- 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* .



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 address and 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 *Ceph network configuration* section of the *Red Hat Ceph Storage Configuration Guide* for details.

5.3. HARDENING THE NETWORK SERVICE

System administrators deploy Red Hat Ceph Storage clusters on Red Hat Enterprise Linux 8 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 are 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 *Using SELinux Guide* within the *Security Hardening Guide* within the *Configuring basic system settings* guide within the Product Documentation for Red Hat Enterprise Linux for your OS version, on the Red Hat Customer Portal.

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.

Table 5.1. Ceph Ports

TCP/UDP 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, **<ipaddress>** with the IP address, **<netmask>** with the subnet mask in CIDR notation, and **<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
```

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 configuration at runtime*.

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/CLUSTER_FSID/ceph.audit.log** file. Run the following command on the monitor host.

Example

```
[root@host04 ~]# cat /var/log/ceph/6c58dfb8-4342-11ee-a953-fa163e843234/ceph.audit.log
```

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.

The following is an output of the Ceph audit log:

```
2023-09-01T10:20:21.445990+0000 mon.host01 (mon.0) 122301 : audit [DBG] from='mgr.14189
10.0.210.22:0/1157748332' entity='mgr.host01.mcadea' cmd=[{"prefix": "config generate-minimal-
conf"}]: dispatch
2023-09-01T10:20:21.446972+0000 mon.host01 (mon.0) 122302 : audit [INF] from='mgr.14189
10.0.210.22:0/1157748332' entity='mgr.host01.mcadea' cmd=[{"prefix": "auth get", "entity":
"client.admin"}]: dispatch
2023-09-01T10:20:21.453790+0000 mon.host01 (mon.0) 122303 : audit [INF] from='mgr.14189
10.0.210.22:0/1157748332' entity='mgr.host01.mcadea'
2023-09-01T10:20:21.457119+0000 mon.host01 (mon.0) 122304 : audit [DBG] from='mgr.14189
10.0.210.22:0/1157748332' entity='mgr.host01.mcadea' cmd=[{"prefix": "osd tree", "states":
["destroyed"], "format": "json"}]: dispatch
2023-09-01T10:20:30.671816+0000 mon.host01 (mon.0) 122305 : audit [DBG] from='mgr.14189
10.0.210.22:0/1157748332' entity='mgr.host01.mcadea' cmd=[{"prefix": "osd blocklist ls", "format":
"json"}]: dispatch
```

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**.

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 users. Consequently, 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, users may create, modify, and delete volumes and images. Ceph handles the create, retrieve, update, and delete 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 backup archives.

6.3. CEPH FILE SYSTEM

The Ceph File System interface creates virtual file systems 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 File System uses two pools:

- **Metadata:** The metadata pool stores the data of the Ceph Metadata Server (MDS), which generally consists of inodes; that is, the file ownership, permissions, creation date and time, last modified or accessed date and time, parent directory, and so on.

- **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 File System interface, usually Red Hat OpenStack Platform, users may create, modify, and delete files in a Ceph File System. Ceph handles the create, retrieve, update, and delete 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 backup 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 users. 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.
  Users may create, modify, and delete buckets or containers, and the objects contained within them in a Ceph Object Gateway. Ceph handles the create, retrieve, update, and delete 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 backup archives.

- **Logging**: Ceph Object Gateway also stores logs of user operations that the user intends to accomplish and operations that have been executed. This data provides traceability about who created, modified or deleted a bucket or container, or an S3 or Swift object residing in an S3 bucket or Swift container. When users delete their data, the logging information is not affected 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 users among other compliance factors.

Multi-site

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 multi-site scenarios, administrators must ensure that each site has implemented security measures. Additionally, if geographic distribution of data would occur in a multi-site 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 the latest certified Red Hat Enterprise Linux version.

- Enable FIPS mode on Red Hat Enterprise Linux either during system installation or after it.
    - For container deployments, follow the instructions in the *Red Hat Enterprise Linux 9 Security Hardening Guide*.

Additional Resources

- Refer to the *US Government Standards* to know the latest information on FIPS validations.
- Refer to the *Red Hat Ceph Storage Compatibility Guide*.

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.