

Hitachi Content Platform for Cloud Scale

v1.0.0

Object Storage Management Guide

© 2019 Hitachi, Ltd. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including copying and recording, or stored in a database or retrieval system for commercial purposes without the express written permission of Hitachi, Ltd., or Hitachi Vantara Corporation (collectively “Hitachi”). Licensee may make copies of the Materials provided that any such copy is: (i) created as an essential step in utilization of the Software as licensed and is used in no other manner; or (ii) used for archival purposes. Licensee may not make any other copies of the Materials. “Materials” mean text, data, photographs, graphics, audio, video and documents.

Hitachi reserves the right to make changes to this Material at any time without notice and assumes no responsibility for its use. The Materials contain the most current information available at the time of publication.

Some of the features described in the Materials might not be currently available. Refer to the most recent product announcement for information about feature and product availability, or contact Hitachi Vantara Corporation at <https://support.hitachivantara.com/en-us/contact-us.html>.

Notice: Hitachi products and services can be ordered only under the terms and conditions of the applicable Hitachi agreements. The use of Hitachi products is governed by the terms of your agreements with Hitachi Vantara Corporation.

By using this software, you agree that you are responsible for:

1. Acquiring the relevant consents as may be required under local privacy laws or otherwise from authorized employees and other individuals; and
2. Verifying that your data continues to be held, retrieved, deleted, or otherwise processed in accordance with relevant laws.

Notice on Export Controls. The technical data and technology inherent in this Document may be subject to U.S. export control laws, including the U.S. Export Administration Act and its associated regulations, and may be subject to export or import regulations in other countries. Reader agrees to comply strictly with all such regulations and acknowledges that Reader has the responsibility to obtain licenses to export, re-export, or import the Document and any Compliant Products.

Preface

This document contains information about using the Object Storage Management application, which is available as part of the Hitachi Content Platform for cloud scale (HCP for cloud scale) software.

This document matches the information in the online Help available in the Object Storage Management application.

Intended audience

This book is intended for those who use the Object Storage Management application, one of the applications available as part of Hitachi Content Platform for cloud scale.

Product version

This document revision applies to HCP for cloud scale v1.0.0.

Related documents

Referenced documents


- *Hitachi Content Platform for Cloud Scale System Management Guide*, MK-HCPCS001
- *Installing Hitachi Content Platform for Cloud Scale*, MK-HCPCS002
- *Hitachi Content Platform for Cloud Scale Copyrights and Third-Party Licenses*, MK-HCPCS003
- *Hitachi Content Platform for Cloud Scale Release Notes*, RN-HCPCS004




Document conventions

This document uses the following typographic conventions:

Convention	Description
Bold	<ul style="list-style-type: none"> Indicates text in a window, including window titles, menus, menu options, buttons, fields, and labels. Example: Click OK. Indicates emphasized words in list items.
<i>Italic</i>	<ul style="list-style-type: none"> Indicates a document title or emphasized words in text. Indicates a variable, which is a placeholder for actual text provided by the user or for output by the system. Example: <code>pairedisplay -g group</code> <p>(For exceptions to this convention for variables, see the entry for angle brackets.)</p>
Monospace	Indicates text that is displayed on screen or entered by the user. Example: <code>pairedisplay -g oradb</code>
< > angle brackets	<p>Indicates variables in the following scenarios:</p> <ul style="list-style-type: none"> Variables are not clearly separated from the surrounding text or from other variables. Example: <code>Status-<report-name><file-version>.csv</code> Variables in headings.
[] square brackets	Indicates optional values. Example: [a b] indicates that you can choose a, b, or nothing.
{ } braces	Indicates required or expected values. Example: { a b } indicates that you must choose either a or b.
vertical bar	<p>Indicates that you have a choice between two or more options or arguments. Examples:</p> <p>[a b] indicates that you can choose a, b, or nothing.</p> <p>{ a b } indicates that you must choose either a or b.</p>

This document uses the following icons to draw attention to information:

Icon	Label	Description
	Note	Calls attention to important or additional information.

Icon	Label	Description
	Tip	Provides helpful information, guidelines, or suggestions for performing tasks more effectively.
	Caution	Warns the user of adverse conditions and/or consequences (for example, disruptive operations, data loss, or a system crash).
	WARNING	Warns the user of a hazardous situation which, if not avoided, could result in death or serious injury.

Accessing product downloads

Product software, drivers, and firmware downloads are available on Hitachi Vantara Support Connect: <https://support.hitachivantara.com/>.

Log in and select Product Downloads to access the most current downloads, including important updates that may have been made after the release of the product.

Accessing product documentation

Product user documentation is available on Hitachi Vantara Support Connect: <https://knowledge.hitachivantara.com/Documents>. Check this site for the most current documentation, including important updates that may have been made after the release of the product.

Getting help

[Hitachi Vantara Support Connect](https://support.hitachivantara.com/en_us/contact-us.html) is the destination for technical support of products and solutions sold by Hitachi Vantara. To contact technical support, log on to Hitachi Vantara Support Connect for contact information: https://support.hitachivantara.com/en_us/contact-us.html.

[Hitachi Vantara Community](https://community.hitachivantara.com) is a global online community for Hitachi Vantara customers, partners, independent software vendors, employees, and prospects. It is the destination to get answers, discover insights, and make connections. **Join the conversation today!** Go to community.hitachivantara.com, register, and complete your profile.

Comments

Please send us your comments on this document to doc.comments@hitachivantara.com. Include the document title and number, including the revision level (for example, -07), and refer to specific sections and paragraphs whenever possible. All comments become the property of Hitachi Vantara Corporation.

Thank you!

Chapter 1: Introducing Hitachi Content Platform for cloud scale

This section introduces Hitachi Content Platform for cloud scale and its major features.

Hitachi Content Platform for cloud scale (HCP for cloud scale) is a software-defined object storage solution that is based on a massively parallel microservice architecture, and is compatible with the Amazon S3 application programming interface (API). HCP for cloud scale is especially well suited to service applications requiring high bandwidth and compatibility with Amazon S3 APIs.

HCP for cloud scale has the ability to federate S3-compatible storage from virtually any private or public source, and present the combined capacity in a single, centrally managed, global namespace.

You can install HCP for cloud scale on any server, in the cloud or on premise, that supports the minimum requirements.

HCP for cloud scale lets you manage and scale storage components. You can add storage components, monitor their states, and take them online or offline for purposes of maintenance and repair. The HCP for cloud scale system provides functions to send notification of alerts, track and monitor throughput and performance, and trace actions through the system.

Data access

HCP for cloud scale supports the Amazon Simple Storage Service (S3) application programming interface (API), which allows client applications to store and retrieve unlimited amounts of data from configured storage services.

High availability

HCP for cloud scale provides high availability for multi-instance sites. High availability requires at least four instances. The best practice is to run at least three master instances, which run essential services, on separate physical hardware (or, if running on virtual machines, on at least three separate physical hosts), and to run HCP for cloud scale services on more than one instance.

Site availability

An HCP for cloud scale site has three master instances, and can tolerate the failure of one master instance without interruption of service. Even if two or all three master instances fail, HCP for cloud scale services may be functional (but you cannot move or scale service instances until master instances are restored).

Service availability

HCP for cloud scale services provide high availability as follows:

- The Metadata Gateway service always has three service instances. When the system starts up, the nodes "elect a leader" using the raft consensus algorithm. The leader processes all GET and PUT requests. If the followers cannot identify the leader, they elect a new leader. The Metadata Gateway service can tolerate one service instance failure, and service remains available without loss of data, so long as at least two service instances are healthy.
- The Metadata Coordination service always has one service instance. If that instance fails, HCP for cloud scale automatically starts another instance. Until startup is complete, the Metadata Gateway service cannot scale.
- The Metadata Cache service always has one service instance. If that instance fails, HCP for cloud scale automatically starts another instance. Until startup is complete, performance decreases.

The rest of the HCP for cloud scale services remain available if HCP for cloud scale instances or service instances fail as long as at least one service instance remains healthy. Even if a service that only has one service instance fails, HCP for cloud scale will automatically start a new service instance.

Metadata availability

Metadata is available as long as two services are available:

- S3 Gateway
- Metadata Gateway

Object data availability

Object data is available as long as these items are available:

- S3 Gateway service (at least one instance)
- The storage component containing the requested data
- At least two functioning Metadata Gateway service instances (of the required three)

The availability of object data depends on the storage component. For high availability of object data, you should use a storage component with high availability, such as HCP, HCP-S, and AWS S3. This is true as well for data protection.

Network availability

You can install each HCP for cloud scale instance with an internal and an external network interface. If you want to avoid networking single points of failure, you can:

- Configure two external network interfaces in each HCP for cloud scale instance
- Use two switches, and connect each network interface to one of them
- Bind the two network interfaces (that is, as Active-Passive) into one virtual network interface
- Install HCP for cloud scale using the virtual network interface

Failure recovery

HCP for cloud scale actively monitors the health and performance of the system and its resources, provides real-time visual health representations, issues alert messages when needed, and can automatically take action to recover from the following types of failures:

- Instances (nodes)
- Product services (software processes)
- System services (software processes)
- Storage components

Instance failure recovery

If an instance (a compute node) fails, HCP for cloud scale automatically adds new service instances to other available instances (compute nodes) to maintain the recommended minimum number of service instances. Data on the failed instance is not lost and remains consistent. However, while the instance is down, data redundancy may degrade.

HCP for cloud scale only adds new service instances automatically for floating services. Depending on the remaining number of instances and service instances running, you may need to add new service instances or deploy a new instance.

Service failure recovery

HCP for cloud scale monitors service instances and automatically restarts them if they are not healthy.

For floating services, you can configure a pool of eligible HCP for cloud scale instances and the number of service instances that should be running at any time. You can also set the minimum and maximum number of instances running each service. If a service instance failure causes the number of service instances to go below the minimum, HCP for cloud scale brings up another one on one of the HCP for cloud scale instances in the pool that doesn't already have that service instance running.

Persistent services run on the specific instances that you specify. If one of those service instances fails, HCP for cloud scale restarts the service instance in the same HCP for cloud scale instance. HCP for cloud scale does not automatically bring up a new service instance on a different HCP for cloud scale instance.

Storage component failure recovery

HCP for cloud scale performs regular health checks to detect storage component failures.

If HCP for cloud scale detects a failure, it sets the storage component state to INACCESSIBLE, so that HCP for cloud scale will not try to write new objects to it. You can configure HCP for cloud scale to send an alert when this event happens. While a storage component is down, the data in it is not accessible.

HCP for cloud scale keeps checking a failed storage component and, when it detects that the storage component is healthy again, automatically sets its state to ACTIVE. You can configure HCP for cloud scale to send an alert when this event happens as well. Once the storage component is repaired and brought back online, the data its contains is again accessible, and you can write new objects to it.

Storage components, buckets, and objects

A storage component is an Amazon S3-compatible storage system, running independently, that is manageable by HCP for cloud scale as a back end to store object data. To an S3 client, the existence, type, and state of storage components are transparent.

HCP for cloud scale supports the following storage systems:

- Amazon S3
- Hitachi Content Platform (HCP)
- HCP S Series Nodes
- Any Amazon S3-compatible storage service

An HCP for cloud scale bucket is modeled on a storage service bucket. A bucket is a logical collection of secure data objects that is created and managed by a client application. HCP for cloud scale uses buckets to manage storage components, and an HCP for cloud scale site can be thought of as a logical collection of secure buckets. Buckets have associated metadata such as ownership and lifecycle status. HCP for cloud scale buckets are owned by an HCP for cloud scale user, and access is controlled on a per-bucket basis by Amazon ACL support using S3 APIs. Buckets are contained in a specific region; HCP for cloud scale supports one region.

**Note:**

1. HCP for cloud scale buckets are not stored in storage components, so HCP for cloud scale clients can create buckets even before adding storage components.
2. Storage component buckets are created by storage component administrators, and are not visible to HCP for cloud scale clients.

An object consists of data and associated metadata. The metadata is a set of name-value pairs that describe the object. Every object is contained in a bucket. An object is handled as a single unit by all HCP for cloud scale transactions, services, and internal processes.

For information about Amazon S3, see [Introduction to Amazon S3](#).

Security and authentication

HCP for cloud scale controls access to system functions by means of user accounts, roles, and OAuth tokens, where user accounts reside in an external identity provider. HCP for cloud scale controls access to data (S3 APIs) by means of S3 credentials, ownership, and access control lists. HCP for cloud scale supports in-flight encryption (HTTPS) for all external communications.

User accounts

The initial user account, which has all permissions, is created when you install HCP for cloud scale. The initial user account can perform all functions. After the initial user account is created, you can change its password any time, but you cannot disable it and you cannot change its permissions.

The initial user is the only local account allowed, and is only intended to let you configure an identity provider (IdP). HCP for cloud scale can communicate to IdPs using HTTP or HTTPS. HCP for cloud scale supports multiple IdPs:

- Active Directory
- OpenLDAP
- 389 Directory Server
- LDAP compatible

HCP for cloud scale supports external users defined in the IdP. External users with the appropriate permissions can perform any or all of these functions:

- Log in to the Object Storage Management application and use all functions
- Log in to the System Management application and use all functions
- Get an OAuth token to use all API calls for the Object Storage Management and System management applications
- Log in to the S3 User Credentials application and get S3 credentials to use S3 APIs

HCP for cloud scale discovers the groups in each IdP, and allows assigning roles to groups.

HCP for cloud scale uses OAuth2 as a service provider to authenticate single sign-on (SSO) access. SSO lets you use one set of login credentials for all HCP for cloud scale applications, and you can switch between applications without logging in again.

Data access control

HCP for cloud scale uses ownership and access control lists (ACLs) as data access control mechanisms in S3 APIs.

Ownership is implemented as follows:

- An HCP for cloud scale bucket is owned by the user who creates the bucket, and the owner cannot be changed
- A user has full control of the buckets that user owns
- A user has full control of the objects that user creates
- A user can only list the buckets that user owns

ACLs allow the assignment of privileges (read, write, or full control) to other user accounts besides the owner to access bucket and objects.

API access

The Object Storage Management application APIs require a valid OAuth access token for a user account with suitable permissions; otherwise, the requests are rejected. With one exception, the System Management application APIs also require a valid OAuth access token for a user account with suitable permissions; otherwise, the requests are rejected. (The API call to generate an OAuth token requires only a username and password in the body of the request.)

Before using either the Object Storage Management or System Management APIs, you need to obtain an OAuth token. You can generate an OAuth token by sending a request to the OAuth server with your account credentials. Then you can supply the OAuth token in the Authorization header in the request. OAuth tokens are valid for five hours.



Note: A user can revoke all OAuth tokens for any other HCP for cloud scale user. You would do this if an employee leaves the company, you delete the user account, and you do not want to wait for the account tokens to expire. For information about revoking OAuth tokens for a user, see [Revoking OAuth tokens for a user \(on page 60\)](#).

S3 API requests generally require valid S3 credentials for users with the right privileges, that is, access control lists (ACLs). (Exception are operations configured to allow anonymous access and the use of pre-signed requests.) HCP for cloud scale supports AWS Signature version 4 authentication to include S3 credentials in S3 requests.

A valid user account with suitable permissions can generate S3 credentials. You can generate an unlimited number of S3 credentials, but only the last credentials generated are valid. These credentials are associated only with your account. S3 credentials do not have an expiration date, so they are valid unless and until revoked.

A valid user account with suitable permissions can revoke all S3 credentials of any user. (That is, you can revoke your own S3 credentials or the S3 credentials of any other user.) Revocation removes all S3 credentials associated with the account.



Note: Deleting a user account from the IdP does not revoke S3 credentials, and if a user's S3 credentials are revoked the user can generate new credentials. The best practice is to delete the user account from the IdP and then revoke the S3 credentials.

For information about generating S3 user credentials, see [S3 User Credentials \(on page 55\)](#). For information about revoking S3 user credentials, see [Revoking S3 credentials \(on page 56\)](#).

Data security

HCP for cloud scale supports encryption of data sent between systems ("in flight") and data stored persistently within the system ("at rest").

Certificate management

HCP for cloud scale uses Secure Sockets Layer (SSL) to provide security for both incoming and outgoing communications. To enable SSL security, two certificates are required:

- System certificate: the certificate HCP for cloud scale uses for its GUI and APIs (incoming communications)
- Client certificate: the certificates of IDPs, storage components, and SMTP servers (outgoing communications)

For a system certificate, HCP for cloud scale comes with its own self-signed SSL server certificate, which is generated and installed automatically when the system is installed. This certificate is not automatically trusted by web browsers. You can choose to trust this self-signed certificate or replace it by using one of three options:

1. Upload a PKCS12 certificate chain and password and apply it as the active system certificate.
2. Download a certificate signing request (CSR), then use it to obtain, upload, and apply a certificate signed by a certificate authority (CA).
3. Generate a new self-signed certificate and apply it as the active system certificate.

For a client certificate, you need to upload the certificate of the clients HCP for cloud scale needs to access using SSL.

You can manage certificates, as well as view the installed certificates and their details, using the System Management application.

Data-in-flight encryption

HCP for cloud scale supports data-in-flight encryption (HTTPS) for all external communications. Data-in-flight encryption is always enabled for these data paths:

- S3 API (HTTP is also enabled on a different port)
- Management API
- System Management App user interface (GUI)
- Object Storage Management App GUI

You can enable or disable data-in-flight encryption for these data paths:

- Between HCP for cloud scale and an identity provider (IDP) server
- Between HCP for cloud scale and each application using TLS or SSL
- Between HCP for cloud scale and each managed storage component
- Between HCP for cloud scale and each SMTP server using SSL or STARTTLS

Communication among HCP for cloud scale instances are without data-in-flight encryption. Depending on your security requirements, you may need to set up an isolated internal network for your HCP for cloud scale site.

Data-at-rest encryption

HCP for cloud scale stores three kinds of data persistently:

1. HCP for cloud scale services data
2. HCP for cloud scale metadata and user-defined metadata
3. User data (object data)

The first two kinds of data are handled by the hardware on which HCP for cloud scale instances are installed. If needed, you can install HCP for cloud scale on servers with encrypted disks. Data of the last kind is handled by storage components. If needed, you can use storage components that support data-at-rest encryption. Storage components can self-manage their keys, or HCP for cloud scale can facilitate customer-supplied keys following the S3 API specification.

Network isolation and port mapping

When you install HCP for cloud scale, you can achieve network isolation by configuring it with one external network and one internal network.

HCP for cloud scale software creates a cluster using commodity x86 Linux servers that are networked using Ethernet. The software uses two networks constructed on the operating system hosting the HCP for cloud scale software. These networks may additionally employ link aggregation defined by the OS administrator. While two networks provide optimal traffic isolation, it is possible to deploy the software using a single network. These networking decisions are made by the OS administrator. These network topology decisions must be completed and already in place when you install HCP for cloud scale. HCP for cloud scale uses a variety of network ports identified during the installation process. You will have this one opportunity to adjust or alter the default ports used by any service.

When you install HCP for cloud scale, you can also configure it to use specific ports instead of the default ports.

For information about installing HCP for cloud scale, see *Installing Hitachi Content Platform for Cloud Scale*.

Scalability of instances, service instances, and storage components

You can increase or decrease the capacity, performance, and availability of an HCP for cloud scale site by adding or removing the following:

- Instances: Additional physical computer nodes or virtual machines
- Service instances: Copies of services running on additional instances
- Storage components: S3-compatible object storage used to store object data

In a multi-instance site, you might add additional instances if you want to improve system performance or you are running out of disk space on one or more instances. You might remove instances if you are retiring hardware, an instance is down and cannot be recovered, or you want to run a site with fewer instances.

In a multi-instance site, you can change where a service instance runs:

- You can configure it to run on additional instances. For example, you can increase the number of instances of the S3-Gateway service to improve throughput of S3 API transactions without having to add a compute instance.
- You can configure it to run on fewer instances. For example, you can free up resources on an instance to run other services.
- You can configure it to run on different instances. For example, you can move the service instances off a hardware instance to retire it.
- For a floating service, instead of specifying a specific instance on which it runs, you can specify a pool of eligible instances, any of which can run the service.

Some services have a fixed number of instances and therefore cannot be scaled. These include:

- Metadata-Coordination
- Metadata-Gateway
- Metadata-Cache

You might add additional storage components to a site under these circumstances:

- The existing storage components are running out of available capacity
- The existing storage components do not provide the performance you require
- The existing storage components do not provide the functionality you require

Supported limits

HCP for cloud scale limits the number of instances (nodes) in a system to 160.

HCP for cloud scale does not limit the number of the following entities.

Entity	Minimum	Maximum	Notes
Buckets	None	Unlimited	
Users (external)	None	Unlimited	The local user can do all operations including MAPI calls and S3 API calls. However, it is recommended that HCP for cloud scale be configured with an identity provider (IdP) with users to enforce role-based access control.
Groups (external)		Unlimited	
Roles		Unlimited	
Objects	None	Unlimited	
Storage components	1	Unlimited	

Logging in

User accounts reside in an external identity provider (IdP). To log in you need this information:

- The IP address of the HCP for cloud scale instance that you're using
- Your user name as assigned by your system administrator
- Your password as assigned by your system administrator
- The security realm where your user account is defined

Procedure

1. Open a web browser and go to `https://instance_ip_address:8000`
instance_ip_address is the IP address of the HCP for cloud scale instance you're using
2. Enter your username and password.

3. In the **Security Realm** field, select the location where your user account is defined. To log in using the local administrator account, without using an external IdP, select **Local**. If no IdP is configured yet, **Local** is the only available option,
4. Click **LOGIN**.

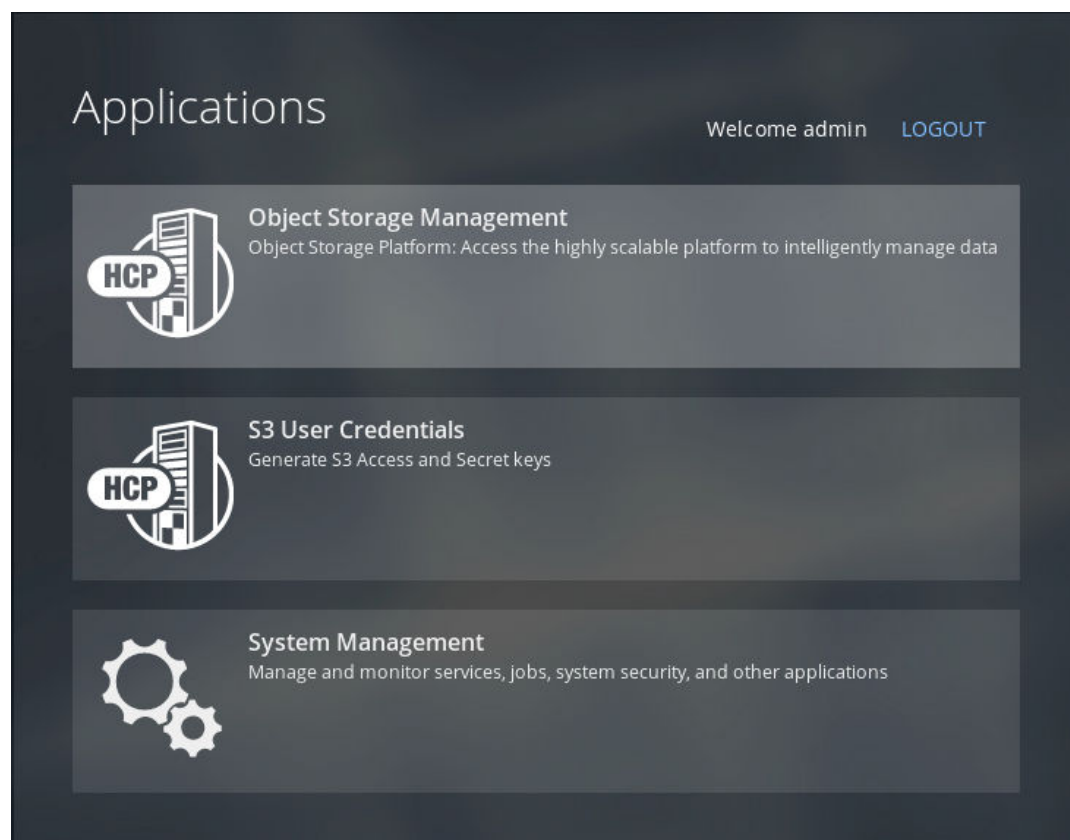
Result

The Applications page opens.

HCP for cloud scale applications

After you log in, HCP for cloud scale presents you with launchable applications:

- Object Storage Management: Manage and monitor storage components, data objects, alerts, and regions
- S3 User Credentials: Generate S3 access and secret keys
- System Management (sometimes referred to in the application as the Admin App): Manage and monitor cluster instances, software services, system security, user accounts, and other cluster configuration parameters



You can return to the Applications page to switch back and forth between these applications as needed.

Switching between applications

HCP for cloud scale uses OAuth2 as a service provider to authenticate single sign-on (SSO) access. You only need one set of login credentials for all HCP for cloud scale applications, and you can switch between applications without logging in again.

To switch between applications:

Procedure

1. Click the HCP icon, in the left corner of the top navigation bar.
You are returned to the **Applications** page.
2. Click **Object Storage Management**, **System Management**, or **S3 User Credentials**, as appropriate.

Chapter 2: Dashboard

This section describes the Dashboard functions.

Hitachi Content Platform for cloud scale (HCP for cloud scale) provides functions that let you monitor the activity and performance of the system and the objects stored in it in real time. Your starting point is the **Dashboard** page.

Entering your serial number

You can use the Object Storage Management application or APIs to enter your HCP for cloud scale serial number.

The serial number is required to activate the HCP for cloud scale software. You must enter the serial number before proceeding further.

Object Storage Management application instructions

To enter your product serial number:

Procedure

1. Select **Dashboard** and click on the Edit icon next to the **Serial Number** field. The **Add Serial Number** window opens.
2. Enter your serial number and click **Add**.

System reports

The **Dashboard** page includes a System Reports section that displays the current counts of active objects and alerts in the system.

Displaying the active object count

The Object Storage Management application displays a count of active objects stored in the system.

Object Storage Management application instructions

To display the Active Object Count report, select Dashboard.

The report displays a line graph showing the total number of active objects in the system over the past week.

Displaying the alert count

You can use the Object Storage Management application or APIs to display a count of active alerts.

For information about alerts, see [Alerts \(on page 35\)](#).

Object Storage Management application instructions

To display the Alert Count report, select Dashboard.

The report displays the number of active alerts, if any. If there are no active alerts, this infographic is not displayed.

In addition the alert icon, in the upper right corner of the page, displays a badge with the current count of active alerts.

Related API method

```
POST /alert/list
```

For information about specific API methods, in the Object Storage Management application, click the profile icon and select REST API.

Metrics

HCP for cloud scale uses a third-party, open-source software tool, running as a service, to provide storage component metrics through a browser.

The Metrics service collects metrics for these HCP for cloud scale services:

- S3 Gateway
- MAPI Gateway
- Metadata Policy Engine
- Metadata Cache
- Metadata Coordination
- Metadata Gateway

By default the Metrics service collects all storage component metrics, and you cannot disable collection. By default, the Metrics service collects data every ten seconds (the Scrape Interval), and retains data for 15 days (the Database Retention); you can configure these values in the service by using the System Management application.



Note: Metrics related to the operation of HCP for cloud scale instances and services are collected and provided by the System Management application. Collection of these metrics cannot be disabled. For information about these metrics, see the Help available in that application.

Available metrics

Metrics from all services

The following metrics are available from all services.

Metric	Description
http_monitoring_requests_total	Count of the total number of requests made to the monitoring API
metadata_clientobject_active_count	
metadata_clientobject_part_active_count	
scrape_duration_seconds	Duration in seconds of the scrape (collection interval)
scrape_samples_post_metric_relabeling	Number of samples remaining after metric relabeling was applied
scrape_samples_scraped	Number of samples the target exposed
up	1 if the instance is healthy (reachable) or 0 if collection of metrics from the instance failed
update_queue_inprogress	
update_queue_size	

S3 Gateway

The following metrics are available from the S3 Gateway service.

Metric	Description
http_s3_servlet_errors_total	Count of total number of errors returned by the s3 servlet
http_s3_servlet_get_object_response_bytes_total	Count of total bytes in the body of S3 GET Object responses
http_s3_servlet_operations_total	Count of total number of S3 operations made to the s3 servlet (for each endpoint)

Metric	Description
http_s3_servlet_put_object_bytes_total	Count of total bytes of objects put to S3
http_s3_servlet_put_object_part_bytes_total	
http_s3_servlet_requests_histogram_latency_seconds_bucket	Latency in seconds as measured by a histogram timer
http_s3_servlet_requests_histogram_latency_seconds_count	Count of S3 servlet request observations; used with sum to determine average
http_s3_servlet_requests_histogram_latency_seconds_sum	Sum of S3 servlet request latency; used with count to determine average
http_s3_servlet_requests_latency_seconds	Latency in seconds as measured by a summary timer
http_s3_servlet_requests_latency_seconds:hour_average	Latency in seconds over the last hour as measured by a summary timer
http_s3_servlet_requests_latency_seconds_count	
http_s3_servlet_requests_latency_seconds_sum	
http_s3_servlet_requests_total	Count of total number of requests made to the s3 servlet

Metadata Policy Engine

The following metrics are available from the Metadata Policy Engine:

- BUCKET
- BUCKET_count
- BUCKET_sum
- CLIENT_OBJECT
- CLIENT_OBJECT_count
- CLIENT_OBJECT_sum
- CLIENT_OBJECT_PART
- CLIENT_OBJECT_PART_count
- CLIENT_OBJECT_PART_sum
- CONFIG_REPLICATION
- CONFIG_REPLICATION_count
- CONFIG_REPLICATION_sum
- DUQ_query_latency
- DUQ_query_latency_count
- DUQ_query_latency_sum
- SCHEDULED_JOB
- SCHEDULED_JOB_count
- SCHEDULED_JOB_sum
- STORED_OBJECT
- STORED_OBJECT_count
- STORED_OBJECT_sum

Displaying metrics

You can use the metrics service to display or graph metrics, or use the service APIs to obtain metrics.

Object Storage Management application instructions

You can display and graph metrics using the metrics GUI.

To display metrics, select Dashboard and then click the Metrics panel. The metrics tool opens in a separate browser window.

The metrics tool is a third-party, open-source package. For information about using the metrics tool, see the documentation provided with the tool.

Tracing requests and operations

HCP for cloud scale uses an open-source software tool, running as a service, to provide service tracing through a browser.

The Tracing service provides end-to-end, distributed tracing of S3 requests and operations by HCP for cloud scale services. By tracing requests and operations you can monitor performance and troubleshoot possible issues.

Tracing involves three service instances:

- Tracing Query: serves up the traces
- Tracing Agent: receives spans from tracers
- Tracing Collector: receives spans from Tracing Agent service using Tchannel

Displaying traces

You can display traces using the tracing service GUI.

To begin tracing, select Dashboard and then click the Tracing panel. The tracing tool opens in a separate browser window.

When tracing, you can specify:

- Service to trace
- Operation to trace (all or specific) for each service
- Tags
- Lookback period (by default, over the last hour)
- Minimum duration
- Number of results to display (by default, 20)

The service displays all found traces with a chart giving the time duration for each trace. You can click on a trace to display how the trace is served by difference services in cascade and the time spent on each service.

For information about the tracing tool, see the documentation provided with the tool.

Traceable operations

The following operations are traceable.

Component	Operation
asynch-policy-engine	Bucket Lookup
	Bucket Name To Id Map
	Bucket Name To Owner Id Map

Component	Operation
	Dequeue
	Metadata
	BUCKET
client-access-service	Bucket Count Limit
	Bucket Create
	Not Anonymous Authorization
	User Lookup Buckets
foundry-auth-client	Foundry Authorize
	Foundry List Users
	Foundry Validate
jaeger-query	/api/dependencies
	/api/services
	/api/services/{service}/operations
mapi-service	GET
	POST
metadata-client	BucketService/Create
	BucketService/ListBucketOwnerListing
	BucketService/LookupBucketNameById
	BucketService/LookupByName
	BucketService/NameToIdMap
	BucketService/NameToOwnerIdMap
	ConfigService/List
	ConfigService/LookupById
	ConfigService/Set
	ControlApiService/AddNode
	UpdateQueueService/Dequeue
	UpdateQueueService/GetWork
	UpdateQueueService/ListInProgress

Component	Operation
	UserService/Create
	UserService/LookupByAuthId
	UserService/LookupById
	UserService/UpdateAddAuthToken
	UserService/ UpdateInvalidateAllAuthTokens
metadata-coordination-client	ScaleGatewayService/AddNode
metadata-coordination-service	ScaleGatewayService/AddNode
metadata-gateway-service	BucketService/Create
	BucketService/ListBucketOwnerListing
	BucketService/LookupBucketNameById
	BucketService/LookupByName
	BucketService/NameToIdMap
	BucketService/NameToOwnerIdMap
	ConfigService/List
	ConfigService/LookupById
	ConfigService/Set
	ControlApiService/AddNode
	StatusService/GetStatus
	UpdateQueueService/Dequeue
	UpdateQueueService/GetWork
	UserService/Create
	UserService/LookupByAuthId
	UserService/LookupById
	UserService/UpdateAddAuthTokens
	UserService/ UpdateInvalidateAllAuthTokens
	raftrpc.MultiRaftRpc/PartitionAppendEntry
storage-component-client	S3 Storage Component Verify

Component	Operation
tomcat-servlet	GET
	PUT
	S3 operation

Chapter 3: Storage components

This section describes the Storage Management functions.

Hitachi Content Platform for cloud scale (HCP for cloud scale) provides functions to let you manage and monitor storage components. Your starting point is the **Storage Component** page.

Storage components, buckets, and objects

A storage component is an Amazon S3-compatible storage system, running independently, that is manageable by HCP for cloud scale as a back end to store object data. To an S3 client, the existence, type, and state of storage components are transparent.

HCP for cloud scale supports the following storage systems:

- Amazon S3
- Hitachi Content Platform (HCP)
- HCP S Series Nodes
- Any Amazon S3-compatible storage service

An HCP for cloud scale bucket is modeled on a storage service bucket. A bucket is a logical collection of secure data objects that is created and managed by a client application. HCP for cloud scale uses buckets to manage storage components, and an HCP for cloud scale site can be thought of as a logical collection of secure buckets. Buckets have associated metadata such as ownership and lifecycle status. HCP for cloud scale buckets are owned by an HCP for cloud scale user, and access is controlled on a per-bucket basis by Amazon ACL support using S3 APIs. Buckets are contained in a specific region; HCP for cloud scale supports one region.



Note:

1. HCP for cloud scale buckets are not stored in storage components, so HCP for cloud scale clients can create buckets even before adding storage components.
2. Storage component buckets are created by storage component administrators, and are not visible to HCP for cloud scale clients.

An object consists of data and associated metadata. The metadata is a set of name-value pairs that describe the object. Every object is contained in a bucket. An object is handled as a single unit by all HCP for cloud scale transactions, services, and internal processes.

For information about Amazon S3, see [Introduction to Amazon S3](#).

Displaying storage component analytics

The **Storage Component** page includes an Analytics section that displays counts of active, inactive, and unverified storage components.

The states displayed are:

- **ACTIVE:** Available to serve requests
- **INACTIVE:** Not available to serve requests (access is administratively paused)
- **UNVERIFIED:** Not available to serve requests (unreachable by specified parameters, or awaiting administrative activation)

Displaying counts of storage components

You can use the Object Storage Management application or APIs to display counts of storage components in the system.

Object Storage Management application instructions

To display storage counts, select Storage.

The infographic displays the count of active, inactive, and unverified storage components.

Related API method

```
POST /storage_component/list
```

For information about specific API methods, in the Object Storage Management application, click the profile icon and select REST API.

Viewing storage components

You can use the Object Storage Management application or APIs to list the storage components defined in the system.

For each storage component, the list gives its name, type, region, and state.

The storage component types are:

- **AMAZON_S3:** An Amazon Web Services S3-compatible node
- **HCP_S3:** A Hitachi Content Platform node
- **HCPS_S3:** An HCP S Series node
- **GENERIC_S3:** An S3-compatible node

The possible storage component states are:

- **ACTIVE:** Available to serve requests
- **INACTIVE:** Not available to serve requests (access is administratively paused)
- **INACCESSIBLE:** Available to serve requests, but HCP for cloud scale is having issues (for example, network, authentication, or certificate issues) accessing it
- **UNVERIFIED:** Not available to serve requests (unreachable by specified parameters, or awaiting administrative activation)

You can activate or deactivate a storage component. For information on activation and deactivation, see [Activating a storage component \(on page 33\)](#) and [Deactivating a storage component \(on page 33\)](#).

You can modify the configuration of a storage component. For more information, see [Modifying a storage component \(on page 32\)](#).

Object Storage Management application instructions

The storage components defined in the HCP for cloud scale system are listed in the Storage Components section of the Storage Components page.

Related API method

```
POST /storage_component/list
```

For information about specific API methods, in the Object Storage Management application, click the profile icon and select REST API.

Adding a storage component

You can use the Object Storage Management application or APIs to add a storage component to the system.



Tip: To improve performance and availability, and to avoid transfer fees, register storage components local to the HCP for cloud scale site.

Before adding a storage component, you must have created an S3 bucket on it.

To define a storage component, you need the following information:

- Storage component type and connection endpoint information
- Proxy connection information
- The access key and secret key you use for access to the storage component



Note: HCP for cloud scale does not support proxy authentication.

Object Storage Management application instructions


The Add Storage Component wizard helps you define a storage component.

The storage component must include an HCP for cloud scale bucket.

To add a storage component:

Procedure

1. From the **Storage Component** page, click **Add Storage Component**.
The **Add Storage Component** wizard opens. The first page describes the information needed.
2. Click **Start**.
The **Connection** page opens.
3. Enter the following information:
 - a. **Storage Component Name** (optional): The display name you choose for the storage component. Enter up to 1024 alphanumeric characters.
 - b. **Storage Type**: Select **AMAZON_S3**, **HCP_S3**, **HCPS_S3**, or **GENERIC_S3**.
 - c. **Region** (optional): Enter a region name of up to 1024 characters.
HCP for cloud scale doesn't validate this field except for its length.
 - d. **Host**: Enter the host name of the storage component.
4. Click **Next**.
The **Connection Advanced** page opens.
5. Enter the following information:
 - a. Select **HTTP** or **HTTPS**
 - b. Enter the **Port**.
 - c. If you select **Proxy**, enter values for the fields **Proxy Host** and **Proxy Port**.
 - d. (Optional) Select **Use Path Style Always**.
6. Click **Next**.
The **Activation** page opens.
7. Enter the following information:
 - a. **Bucket Name**: The name of the bucket on the storage component. Enter a name up to 1024 characters long.

 **Note:** The bucket must already exist on the storage component.

 - b. **Authenticate**: Select the AWS Signature version: Select **V2** or **V4**.
 - c. Enter your **Access Key**.
 - d. Enter your **Secret Key**.
8. Click **Next**.
The **Review** page opens.
9. Review the configuration of the storage component.
 - If the information is not correct, click **Back** to return to the wizard page with the information to correct.
 - If the information is correct, click **Create**.

Result

The storage component is defined. The Storage Component page is refreshed, and the storage component is added to the list.



Note: After you define the storage component, if its state is UNVERIFIED, check the parameters you used when adding it.

Related API method

POST /storage_component/create

**Note:**

1. HCP for cloud scale does not support proxy authentication.
2. After you define the storage component, if its state is UNVERIFIED, check the parameters you used when adding it.

For information about specific API methods, in the Object Storage Management application, click the profile icon and select REST API.

Modifying a storage component

You can use the Object Storage Management application or APIs to modify a storage component.

Object Storage Management application instructions

You can modify the configuration of a storage component.

Procedure

1. From the **Storage Component** page, click the **Edit Component** icon by the storage component you want to modify.
The **Edit Storage Component** wizard opens. For information about the configurable fields, see [Object Storage Management application instructions \(on page 31\)](#).
2. Edit connection information as needed. When you're finished click **Next**.
The **Connection Advanced** page opens.
3. Edit advanced connection information as needed. When you're finished click **Next**.
The **Activation** page opens.
4. Edit activation information as needed. When you're finished click **Next**.
The **Review** page opens.
5. Review the edited configuration of the storage component.
 - If the information is not correct, click **Back** to return to the wizard page with the information to correct.
 - If the information is correct, click **Create**.

Related API method

```
POST /storage_component/update
```

For information about specific API methods, in the Object Storage Management application, click the profile icon and select REST API.

Activating a storage component

You can use the Object Storage Management application or APIs to activate a storage component.

Object Storage Management application instructions

You can activate a storage container that is in the state INACTIVE.

To activate a storage component:

Procedure

1. Select **Storage**.
The **Storage Component** page opens.
2. For the storage component you want to activate, click **Activate Now**.
The storage component state changes to **ACTIVE**.

Related API method

```
POST /storage_component/update_state
```

For information about specific API methods, in the Object Storage Management application, click the profile icon and select REST API.

Deactivating a storage component

You can use the Object Storage Management application or APIs to deactivate a storage component.

Object Storage Management application instructions

You can deactivate a storage container that is in the state ACTIVE.

To deactivate a storage component:

Procedure

1. Select **Storage**.
The **Storage Component** page opens.
2. For the storage component you want to deactivate, click **Yes, Inactivate**.
The storage component state changes to **INACTIVE**.

Related API method

```
POST /storage_component/update_state
```

For information about specific API methods, in the Object Storage Management application, click the profile icon and select REST API.

Chapter 4: Notifications and user profiles

This section describes the notification and user profile functions.

Within Hitachi Content Platform for cloud scale (HCP for cloud scale), the Object Storage Management application provides functions to display notifications, user profiles, online help, and API reference information.

Alerts

An alert is a message to notify you of an event that may require your attention. The Object Storage Management application displays alerts about storage components. Alerts are triggered by events, and remain active until the condition that caused the event is resolved. Once the condition is resolved, the alert is cleared.

If an alert is raised the alert icon turns red and displays a badge with the number of active alerts. Click the icon to display a panel listing alert text.



Note: System alerts are generated by the System Management application to help you monitor overall system health and status of your HCP for cloud scale system. For information about system alerts and how to configure email notifications, see the Help in the System Management application.

Storage component alerts

The storage component alerts are:

- Certificate for Storage component *id* has expired
- Certificate for Storage component *id* is about to expire in *days* days
- Storage component *id* is unavailable

User profiles

The profile icon, on the right end of the top navigation bar, provides access to these functions:

- Help: information about the Object Storage Management application
- REST API: information about the Object Storage Management APIs
- Logout: Log out of the Object Storage Management application and return to the **Login** page

Chapter 5: Services

This section describes the Hitachi Content Platform for cloud scale services.

Services perform functions essential to the health and function of the Hitachi Content Platform for cloud scale (HCP for cloud scale) system. For example, the S3 Gateway service serves S3 API endpoints and communicates with storage components, while the Watchdog service ensures that other services remain running.

Services provide cluster management and coordination, metadata coordination and caching, and external gateways.

Internally, services run in Docker containers on the instances of the system. The container orchestration framework supports cloud or on-premise deployment.

HCP for cloud scale supports an adaptive service deployment model that can change based on workload.

Service categories

Services are grouped into these categories depending on what actions they perform:

- Product services enable HCP for cloud scale functions. For example, the S3 Gateway service serves S3 API endpoints and communicates with storage components. You can scale, move, and reconfigure product services.
- System services maintain the health and availability of the HCP for cloud scale system. For example, the Watchdog service ensures that other services remain running. You cannot scale, move, or reconfigure system services.

HCP for cloud scale services

The table below describes the services that HCP for cloud scale runs. Each service runs within its own Docker container. For each service, the table lists:

- RAM needed per instance: The amount of RAM that, by default, the service needs on each instance on which it's deployed. For all services except for System services, this value is also the default Docker value of Container Memory for the service.
- Number of instances: Shows both:
 - The required number of instances on which a service must run to function properly.
 - The recommended number of instances on which a service should run. These are recommended minimums; if your system includes more instances, you should take advantage of them by running services on them.
- Whether the service is persistent (that is, it must run on a specific instance) or supports floating (that is, it can run on any instance).
- Whether the service is scalable or not.



Note: For HCP for cloud scale services, you cannot set the Container Memory size larger than the Max Heap Size setting. For other services, you should not set the Container Memory size larger than the Max Heap Size setting.

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
Product services: These services perform HCP for cloud scale functions. You can move and reconfigure these services.			
Cassandra Decentralized database that can be scaled across large numbers of hardware nodes.	Container Options: Default <ul style="list-style-type: none">Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 2.4 GB.CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1.	RAM needed per instance:	2.4 GB
	Service Options	Number of instance s:	Required: 3 Recommend ed: All

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
	<ul style="list-style-type: none"> Max heap: Maximum amount of memory to allocate to the Java heap for each instance of the service. Enter an integer number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is 1800m. Heap new size: Heap size for the Java virtual machine. Valid values are integers representing a number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is 512 MB. 	Service units:	10
		Persistent or floating?	Persistent
		Supports volume configuration?	No
		Single or multiple types?	Single
		Scalable?	Yes
Chronos Job scheduling.	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 712 MB. CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	RAM needed per instance:	712 MB
		Service units:	1

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
	Service Options <ul style="list-style-type: none"> Heap settings: Maximum amount of memory to allocate to the Java heap for each instance of the service. Enter an integer number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is 512 MB. 	Number of instances:	<ul style="list-style-type: none"> 1 required 1 recommended
		Persistent or floating?	Floating
		Supports volume configuration?	Yes
		Single or multiple types?	Single
		Scalable?	No
Elasticsearch Data indexing and search platform.	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 2 GB. CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	RAM needed per instance:	2 GB
		Service units:	25
	Service Options <ul style="list-style-type: none"> Heap settings: Maximum amount of memory to allocate to the Java heap for each instance of the service. Enter an integer number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is 1024 MB. 	Number of instances:	Required: 3 Recommended: All
		Persistent or floating?	Persistent

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
	<ul style="list-style-type: none"> Days to keep logs: The number of days to keep service logs, including access and metrics indexes. The default is 30 days. Index Protection Level: The number of additional replicas (copies) to keep of each index file (shard). Replicas are kept on separate instances. You can set this value for every shard. The default is 1 replica (which means that two copies are kept). The maximum is the number of instances less one. 	Supports volume configuration?	Yes
		Single or multiple types?	Single
		Scalable?	Yes
Kafka Stream processing platform for handling real-time data streams.	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 2 GB. CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	RAM needed per instance:	2 GB
		Service units:	5
	Service Options <ul style="list-style-type: none"> Heap settings: Maximum amount of memory to allocate to the Java heap for each instance of the service. Enter an integer number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is -Xmx1800m-Xms512m. 	Number of instances:	Required: 3 Recommended: All
		Persistent or floating?	Persistent
		Supports volume configuration?	Yes
		Single or multiple types?	Single

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
		Scalable ?	Yes
Logstash Logging.	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 700 MB. CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	RAM needed per instance:	700 MB
		Number of instances:	Required: 1 Recommended: 1
		Persistent or floating?	Floating
		Supports volume configuration?	No
		Single or multiple types?	Single
		Scalable ?	No
MAPI Gateway Serves MAPI endpoints.	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 768 MB. 	RAM needed per instance:	768 MB

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
	<ul style="list-style-type: none"> CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	Service units:	5
		Number of instances:	Required: 1 Recommended: All
		Persistent or floating?	Floating
		Supports volume configuration?	No
		Single or multiple types?	Single
		Scalable?	Yes
Metadata Cache Cache for system metadata.	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 768 MB. CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	RAM needed per instance:	768 MB
		Service units:	10

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
	Service Options <ul style="list-style-type: none"> Max Heap Size: Maximum amount of memory to allocate to the Java heap for each instance of the service. Enter an integer number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is 512 MB. 	Number of instances:	Required: 1 Recommended: 1
		Persistent or floating?	Persistent
		Supports volume configuration?	No
		Single or multiple types?	Single
		Scalable?	No
Metadata Coordination Coordinates the different metadata gateway services, and coordinates metadata partitions.	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 768 MB. CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	RAM needed per instance:	768 MB
		Number of instances:	Required: 1 Recommended: 1
	Service Options <ul style="list-style-type: none"> Max Heap Size: Maximum amount of memory to allocate to the Java heap for each instance of the service. Enter an integer number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is 64 MB. 	Persistent or floating?	Floating

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
		Supports volume configuration?	No
		Single or multiple types?	Single
		Scalable?	No
Metadata Gateway Stores and protects metadata and serves it to other services.	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 4096 MB. CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	RAM needed per instance:	768 MB
		Service units:	50
	Service Options <ul style="list-style-type: none"> Max Heap Size: Maximum amount of memory to allocate to the Java heap for each instance of the service. Enter an integer number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is 2048 MB. 	Number of instances:	Required: 3 Recommended: 3
		Persistent or floating?	Persistent
		Supports volume configuration?	No
		Single or multiple types?	Single
		Scalable?	No

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
Metrics	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 768 MB. CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	RAM needed per instance:	768 MB
		Service units:	10
	Service Options <ul style="list-style-type: none"> Prometheus Scrape Interval: The time interval between runs of the metrics collection task. Enter an integer number of seconds. You can optionally specify the suffix s (seconds). The default is 10 seconds. Prometheus Database Path: Storage location for prometheus local time-series db. Enter a path. The default is tsdb/. Prometheus Database Retention: The number of days to retain files. Enter an integer number of days. You can optionally specify the suffix d (days). The default is 15 days. 	Number of instances:	Required: 1 Recommended: All
		Persistent or floating?	Persistent
		Supports volume configuration?	No
		Single or multiple types?	Single
		Scalable?	Yes
Metadata Policy Engine Executes asynchronous metadata updates.	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 2048 MB. 	RAM needed per instance:	768 MB

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
	<ul style="list-style-type: none"> CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	Service units:	25
		Number of instances:	Required: 1 Recommended: 1
		Persistent or floating?	Floating
		Supports volume configuration?	No
		Single or multiple types?	Single
		Scalable?	No
S3 Gateway Serves S3 API endpoints and communicates with storage components.	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 768 MB. CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	RAM needed per instance:	768 MB
		Service units:	25

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
	Service Options <ul style="list-style-type: none"> Max Heap Size: Maximum amount of memory to allocate to the Java heap for each instance of the service. Enter an integer number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is 512 MB. 	Number of instances:	Required: 1 Recommended: All
	HTTP Options: <ul style="list-style-type: none"> Enable HTTP: Select to enable HTTP connections. Max Http Request Headers: The maximum number of HTTP request headers to allow. Enter an integer. The default is 100 request headers. 	Persistent or floating?	Floating
		Supports volume configuration?	No
		Single or multiple types?	Single
		Scalable?	Yes
Tracing Agent Provides end-to-end distributed tracing for S3 API calls and MAPI calls.	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 768 MB. CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	RAM needed per instance:	768 MB
		Service units:	1
	Service Options <ul style="list-style-type: none"> Collector TChannel Hostname: Enter a host name. The default is localhost. Collector TChannel Port: Enter a port number. The default is 14267. 	Number of instances:	Required: All

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
		Persistent or floating?	Floating
		Supports volume configuration?	No
		Single or multiple types?	Single
		Scalable?	Yes
Tracing Collector Provides end-to-end distributed tracing for S3 API calls and MAPI calls.	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 768 MB. CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	RAM needed per instance:	768 MB
	Service Options <ul style="list-style-type: none"> ElasticSearch Hostname: Enter a host name. The default is localhost. ElasticSearch Port: Enter a port number. The default is 9200. Sampling Rate: The sampling rate for all clients implementing remote sampling. Enter a number between 0 and 1 inclusive. The default is 1. 	Number of instances:	Required: 1 Recommended: All
		Persistent or floating?	Floating
		Supports volume configuration?	No
		Single or multiple types?	Single

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
	<ul style="list-style-type: none"> Max open Index age: How long to keep tracing indexes open in the database, in days. Enter a value from 1 to 365 days inclusive. The default is 30 days. Max Index age: How long to keep tracing indexes in the database, in days. Enter a value from 1 to 365 days inclusive. The default is 60 days. 	Scalable ?	Yes
Tracing Query Provides end-to-end distributed tracing for S3 API calls and MAPI calls.	Container Options: Default <ul style="list-style-type: none"> Container Memory: The hard memory limit for the service's Docker container, in MB. The default is 768 MB. CPU: The relative CPU usage weight for the service's Docker container. Generally, a higher value means that the container received more CPU resources than other processes (including other service Docker containers) running on the instance. Enter a decimal number. The default is 0.1. 	RAM needed per instance:	768 MB
	Service Options <ul style="list-style-type: none"> ElasticSearch Hostname: Enter a host name. The default is localhost. ElasticSearch Port: Enter a port number. The default is 9200. 	Number of instances:	Required: 1 Recommended: All
		Persistent or floating?	Floating
		Supports volume configuration?	No
		Single or multiple types?	Single
		Scalable ?	Yes
System services: These services manage system resources and ensure that the HCP for cloud scale system remains available and accessible. These services cannot be moved or reconfigured.			

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
Admin App The system management application.	Service Options <ul style="list-style-type: none"> Max heap size: Maximum amount of memory to allocate to the Java heap for each instance of the service. Enter an integer number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is 512 MB. 	RAM needed per instance:	N/A
		Number of instances:	N/A
		Persistent or floating?	Persistent
		Supports volume configuration?	Yes
		Single or multiple types?	Single
		Scalable?	No
Cluster Coordination Manages hardware resource allocation.	N/A	RAM needed per instance:	N/A
		Number of instances:	N/A
		Persistent or floating?	Persistent
		Supports volume configuration?	No
		Single or multiple types?	Single

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
		Scalable ?	No
Cluster Worker Receives and performs work from other services.	N/A	RAM needed per instance:	N/A
		Number of instances:	N/A
		Service units:	5
		Persistent or floating?	Persistent
		Supports volume configuration?	Yes
		Single or multiple types?	Single
		Scalable ?	No
Network Proxy Network request load balancer.	Security Protocol: Select which TLS versions to use: <ul style="list-style-type: none"> ▪ TLS 1.0 ▪ TLS 1.1 ▪ TLS 1.2 ▪ TLS 1.3 	RAM needed per instance:	N/A
	SSL Ciphers: If you want to provide your own cipher suite, enter it here.	Number of instances:	N/A
		Service units:	1

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
	Custom Global Configuration:Select Enable Advanced Global Configuration to enable adding your own parameters to the HAProxy "global" section.	Persistent or floating?	Persistent
		Supports volume configuration?	Yes
		Single or multiple types?	Single
		Scalable?	No
Sentinel Runs internal system processes and monitors the health of the other services.	Service Options ▪ Max heap size: Maximum amount of memory to allocate to the Java heap for each instance of the service. Enter an integer number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is 256 MB.	RAM needed per instance:	N/A
		Number of instances:	N/A
		Persistent or floating?	persistent
		Supports volume configuration?	Yes
		Single or multiple types?	Single
		Scalable?	No
Service Deployment Handles deployment of high-level services (that is,	N/A	RAM needed per instance:	N/A

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
the services that you can configure).		Number of instances:	N/A
		Persistent or floating?	Persistent
		Supports volume configuration?	Yes
		Single or multiple types?	Single
		Scalable?	No
Synchronization Coordinates service configuration settings and other information across instances.	Service Options <ul style="list-style-type: none"> Max heap size: Maximum amount of memory to allocate to the Java heap for each instance of the service. Enter an integer number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is 1024 MB. 	RAM needed per instance:	N/A
		Number of instances:	N/A
		Persistent or floating?	Persistent
		Supports volume configuration?	Yes
		Single or multiple types?	Single
		Scalable?	No

Service name and description	Configuration settings (changes cause the service to redeploy)	Properties	
Watchdog Monitors the other System services and restarts them if necessary. Also responsible for initial system startup.	Service Options <ul style="list-style-type: none"> Max heap size: Maximum amount of memory to allocate to the Java heap for each instance of the service. Enter an integer number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is 1024 MB. Heap new size: Heap size for the Java virtual machine. Valid values are integers representing a number of bytes. You can optionally specify the suffixes k (kilobytes), m (megabytes), or g (gigabytes). The default is 512 MB. 	RAM needed per instance:	N/A
		Number of instances:	N/A
		Service units:	5
		Persistent or floating?	Persistent
		Supports volume configuration?	Yes
		Single or multiple types?	Single
		Scalable?	No

Listing service ports

You can list service port information for ports available for customer use.

You can list public service ports using an API without an access token.

Related API method

```
POST /public/discovery/get_service_port
```

For information about specific API methods, in the Object Storage Management application, click the profile icon and select REST API.

Chapter 6: S3 User Credentials

This section describes the S3 User Credentials application.

Amazon Web Services uses security credentials, called S3 credentials, to authenticate and authorize data requests. The credentials consist of an access key and a secret key. Client applications that post S3 requests, such as uploading documents, reading documents, and adding buckets, to Hitachi Content Platform for cloud scale (HCP for cloud scale) also need these credentials. HCP for cloud scale provides a simple application, S3 User Credentials, to obtain these credentials for registered users of the system. It obtains an OAuth token from system services once you log in.

Obtaining S3 credentials

You can use the S3 User Credentials application or APIs to obtain S3 credentials.

The S3 User Credentials application retrieves credentials (access and secret key) to access Amazon S3 bucket services. These credentials are linked to the username and password supplied in the API call. Thus, each unique user will retrieve a unique set of credentials.

If a user makes multiple, repeated API calls, only the last set of credentials remain active. Previously retrieved credentials become invalidated and will no longer work. Credentials expire automatically when the user changes the password held in the identity provider.

S3 application instructions

Use the S3 User Credentials application to obtain S3 access credentials.

Obtaining credentials nullifies any pre-existing S3 credentials you may already have.

To obtain S3 credentials:

Procedure

1. From the **Applications** page, select the application **S3 User Credentials**.
2. Click **Generate S3 Credentials**.
You are warned that any existing credentials will be nullified.
3. Click **Generate**.

Result

The application generates and displays an Access Key and a Secret Key.

Next steps

You can copy and paste these credentials into the client application you use to post S3 requests to HCP for cloud scale.

Related API method

```
POST /s3/user/generate_credentials
```

For information about specific API methods, in the Object Storage Management application, click the profile icon and select REST API.

Revoking S3 credentials

Amazon S3 credentials can be revoked by the associated user or by other users with appropriate permissions. If you have permissions you can revoke all Amazon S3 credentials belonging to a specific user. Use the endpoint `/user/list` to look up the ID of the user for whom you want to revoke credentials.

Related API methods

```
POST /user/list
```

```
POST /user/revoke_credentials
```

For information about specific API methods, in the Object Storage Management application, click the profile icon and select REST API.

Chapter 7: HCP for cloud scale APIs

This section describes how to use the Hitachi Content Platform for cloud scale APIs.

The Hitachi Content Platform for cloud scale (HCP for cloud scale) system provides a set of RESTful application programming interfaces (APIs) that you can use for writing applications that can exercise its functions and manage the system. Anything you can do in the Object Storage Management, System Management, or S3 User Credentials applications can also be done using APIs.

HCP for cloud scale management APIs

The HCP for cloud scale management API provides a RESTful HTTPS interface to administrative functions such as managing Amazon S3 settings and storage components. The rest of this section describes how you can get started with this API.

System Management management APIs

You can access system management functions, such as system monitoring and configurations, through the REST API within the System Management application. For more information on the System Management API, see the Help in the System Management application.

Getting started with Object Storage Management APIs

HCP for cloud scale provides a RESTful HTTPS interface for the following functions:

- Configuring Amazon Simple Storage Service (Amazon S3) settings
- Managing storage components

The Object Storage Management APIs are served by the MAPI Gateway service from any HCP for cloud scale node.

You can execute all functions supported in the Object Storage Management application using RESTful APIs.



Note: The system configuration, management, and monitoring functions provided through the System Management application can be performed using the System Management APIs.

All URLs for the APIs have the following base, or root, uniform resource identifier (URI):

`https://hcpcs_ip_address:9099/mapi/v1`

Input and output formats

The API accepts and returns JSON.

The REST API accepts and returns JavaScript Object Notation (JSON). It does not support HTTPS 1.0 requests; all HTTPS 1.0 requests are denied. When the body of the request has contents, the MAPI accepts and returns JSON; when the body is empty, JSON format is unnecessary.

Access

S3 API requests require valid S3 credentials for users with the right privileges, that is, ACLs. An exception are the operations configured to allow anonymous access, and the use of pre-signed requests. HCP for cloud scale supports AWS Signature Version 4 authentication to include the S3 credentials in S3 requests.




Note: AWS Signature Version 2 is deprecated by AWS and not supported by HCP for cloud scale.

Authentication

API requests require a valid OAuth access token for a user account with the suitable permissions. Before performing API requests, you need to generate an OAuth token by sending a request to the OAuth server using the System Management application REST API with your user account credentials. Then you need to supply your OAuth token in an authorization header in your API requests.

The following table describes the life cycle of an OAuth token:

Stage	Description
Generation	A valid user account can generate an OAuth token using a System Management application REST API. A user can generate multiple OAuth tokens, and they are all valid.
Expiration	OAuth tokens expire two hours after they are generated.  Note: You can configure this value using the System Management application.
Refresh	Access token refresh is not supported.
Revocation	You can revoke all OAuth tokens for any user in the system. When an employee leaves the company, for instance, the system access of that employee should be terminated immediately. Revocation lets you terminate the tokens for a user without having to wait for them to expire. Revocation can be reversed as long as the user's account is not deleted from the identity provider. In this case, the user can re-authenticate and obtain a new OAuth token.

Requesting and submitting an access token

You need to request an access token from the system. Send an `HTTP POST` request to the endpoint `/auth/oauth`.

To use the API interface, the account you're using must have the appropriate permissions assigned, and you need a valid OAuth access token. For information about the required permissions for making API calls, see the System Management Help. The security access token, known as a Bearer token, authorizes all requests made to access and manage storage components and S3 settings in an HCP for cloud scale system.

To request an access token:



Note:

- To get a list of security realms for the system, send an `HTTP GET` request to the endpoint `/setup`. For example, to do this with `cURL`, send this command:

```
curl -k -X GET --header 'Accept: application/json' 'https://
mysystem.example.com:admin-app-port/api/admin/setup'
```

- To get an access token for the local admin user account, you can omit the realm option for the request, or specify a realm value of `Local`.

Procedure

1. Send an HTTP `POST` request to the endpoint `/auth/oauth` in the System Management application API.
The payload of the request requires your username, password, and realm for a user account.
The system sends you a JSON response body containing an `access_token` field.
The value of this field is your token.
2. Include your access token as part of all REST API requests that you make by submitting an `Authorization` header along with your request.

Example

Here's an example of requesting an authentication token using the cURL command-line tool:

```
curl -ik -X POST https://mysystem.example.com:8000/auth/oauth/ \
-d grant_type=password \
-d username=user1 \
-d password=password1 \
-d scope=* \
-d client_secret=my-client \
-d client_id=my-client \
-d realm=marketingUsers
```

In response to this request, you receive a JSON response body containing an `access_token` field. The value of this field is your token. For example:

```
{
  "access_token": "eyJr287bjle..."
  "expires_in": 7200
}
```

Here's an example that uses cURL of including an access token as part of a request:

```
curl -X GET --header "Accept:application/json"
      https://mysystem.example.com:admin_app_port/api/admin/instances --
header "Authorization:
      Bearer eyJr287bjle..."
```

Revoking OAuth tokens for a user

Related API method

```
POST /user/revoke_tokens
```

For information about specific API methods, in the Object Storage Management application, click the profile icon and select REST API.

Viewing and using API methods

Your system provides web-based documentation pages where you can view all supported API methods, including the request bodies, request URLs, response bodies, and return codes for each. You can also use these pages to run each API method.

You can use the API documentation pages to experiment with the API. Any requests you submit on the REST API page take effect on the system.



Note: If you specify UUIDs when creating resources, the UUIDs are ignored.

To use the API page to run a method:

Procedure

1. In either the Object Storage Management App or the System Management App, click the user profile icon, in the upper right portion of the page.
2. Select **API**.
3. Click on the row for the method you want.
4. If the method you want requires that you specify a UUID:
 - a. Click the row for the GET method for the resource type that you want.
 - b. Click **Try It Out!**
 - c. In the JSON response body, copy the value for the `uuid` field for the resource that you want.
5. If the method you want requires that you specify a request body, in the **Parameters** section, under **Model Schema**, click inside the JSON text box. The JSON text is added to the **Value** field.



Note: Some methods may require other information in addition to or instead of UUIDs or JSON-formatted text. Some require particular string values or require that you browse for and select a file to upload.

6. Click **Try It Out!**

HTTP status codes

When an HTTP request is sent, the server sends back an HTTP response message. The HTTP response message consists of an HTTP header and, optionally, a message body. The response header contains an HTTP status code that provides a status of the request.

The following table contains a list of returned status codes, descriptions, and the type of HTTP request that can generate the status code.

Status code	HTTP name	Description	Service
200	OK	Success.	GET HEAD Bucket PUT
204	Successful Operation	No content.	DELETE POST Object
400	Bad Request	<p>The request body contains one or more of these:</p> <ul style="list-style-type: none"> ▪ An invalid entry ▪ An invalid value for an entry ▪ Invalidly formatted JSON <p>If the request includes a UUID, the UUID may be invalidly formatted.</p>	GET Bucket DELETE Bucket POST Object PUT
403	Forbidden	<p>Your access is denied. Possible reasons:</p> <ul style="list-style-type: none"> ▪ The credentials provided with the request are invalid. ▪ You do not have permission to list the contents of the specified bucket. ▪ The S3 compatible API is currently disabled for the specified bucket. 	DELETE GET HEAD POST Object
404	Not Found	The resource you are trying to retrieve, edit, or delete cannot be found.	GET DELETE HEAD Bucket POST Object

Status code	HTTP name	Description	Service
			PUT
405	Method Not Allowed	A request was made of a resource using a request method not supported by that resource; for example, using GET on a form which requires data to be presented via POST, or using PUT on a read-only resource.	
409	Conflict	Possible reasons: <ul style="list-style-type: none"> ▪ Bucket not empty. ▪ Object data is currently being written. ▪ Specified object is under retention. 	DELETE PUT Bucket
411	Missing Content Length	No content length specified. You must provide the Content-Length HTTP header.	PUT
413	Request Entity Too Large	The object is too large.	POST Object
500	Internal Server Error	An internal error occurred. If this error persists, contact Support.	DELETE GET

Status code	HTTP name	Description	Service
503	Service Unavailable	<p>The service is temporarily unable to handle the request, probably due to system overload, maintenance, or upgrade. Try the request again, gradually increasing the delay between each successive attempt.</p> <p>If this error persists, contact Support.</p>	DELETE GET

For information about the status codes for a particular method, view the REST API Web interface.

Support for Amazon S3 API

HCP for cloud scale is compatible with the Amazon Simple Storage Service (Amazon S3) REST API, which allows clients to store objects in containers called buckets. A bucket is a collection of objects and has its own individual settings, such as ownership and lifecycle. Using HCP for cloud scale, you can perform common read and write operations on objects and buckets, and manage ACL settings through the client access data service.

For information about using Amazon S3, see the [Amazon S3 API documentation](#).

For information about obtaining S3 user credentials, see [S3 User Credentials \(on page 55\)](#).

The following tables list the supported Amazon S3 API features and describes any implementation differences between Amazon and HCP for cloud scale S3 APIs.

Authentication and addressing operations

Feature	Implementation differences
Authentication with AWS Signature Version 4	Fully implemented
Addressing virtual host (like http://bucket.server/object)	Fully implemented

Feature	Implementation differences
Addressing Path style (like http://server/bucket/object)	Fully implemented
Signed/Unsigned payload	Fully implemented
Chunked request	Fully implemented
Pre-signed URL	Fully implemented

Service operations

Feature	Implementation differences
GET service (list buckets)	Fully implemented

Bucket operations

Feature	Implementation differences
GET Bucket (list objects) V1	Fully implemented
GET Bucket (list objects) V2	Fully implemented
PUT Bucket	When anonymous requests to create or delete a bucket use an invalid bucket name, Amazon S3 performs an access check first and returns 403. HCP for cloud scale returns 400 if the bucket name validation check fails.
DELETE Bucket	
HEAD Bucket	
PUT Bucket ACL	ACL email address grantee types are not supported. In AWS each grantee is specified as a type=value pair, where the type is one of the following: <ul style="list-style-type: none"> ▪ <code>emailAddress</code> if the value specified is the email address of an AWS account ▪ <code>id</code> if the value specified is the canonical user ID of an AWS account ▪ <code>uri</code> if granting permission to a predefined group HCP for cloud scale supports only <code>id</code> and <code>uri</code> .
GET Bucket ACL	
List Multipart Uploads	Fully implemented

Feature	Implementation differences
GET Bucket Lifecycle (except transition action)	HCP for cloud scale does not support Object Transition actions. If these actions are included it will throw a Malformed XML exception.
PUT Bucket Lifecycle (except transition action)	
DELETE Bucket Lifecycle (except transition action)	
GET Bucket Versioning	Version Listing Requests do not strictly comply to documented behavior for NextKeyMarker/NextVersionIdMarker. S3 documentation currently states that these values "specifies the first key not returned that satisfies the search criteria." However, HCP for cloud scale specifies the last key returned in the current response. S3 V1 object listings do not call out as specific a requirement and V2 object listings utilize a continuation token (opaque to the caller); internally, HCP for cloud scale shares the same listing logic across all three listing types.
GET Bucket Object Versions	Fully implemented

Object operations

Feature	Implementation differences
GET Object	If a lifecycle policy is configured for a bucket, HCP for cloud scale displays the expiration date of an object (in the x-amz-expiration header) fetched using the ?versionId subresource. Amazon only displays this when performing unversioned GET requests.
HEAD Object	If a lifecycle policy is configured for a bucket, HCP for cloud scale displays the expiration date of an object (in the x-amz-expiration header) fetched using the ?versionId subresource. Amazon only displays this when performing unversioned HEAD requests.

Feature	Implementation differences
PUT Object	Content-Type Validations: Amazon is extremely liberal in what is accepted for the Content-Type of an object. HCP for cloud scale adds additional checks for what is allowed.
Object and Version Encoding	Amazon AWSS3 Object and Version listing documentation mentions the ability to pass an encoding parameter (url). This is so the object name XML in the response to the client can be escaped to avoid names containing invalid XML characters. This encoding is only documented as applied to object names and not Owner/DisplayNames. Additionally, there is no mention of escaping for Bucket Listing requests. The Owner/DisplayName is a concern as there is a possibility that user display names may not be able to contain characters that could cause XML parsing issues. Amazon may be able to restrict this, though it does not currently return a display name for all regions. HCP for cloud scale utilizes Foundry IDPs, thus controlling restriction is not in the realm of HCP for cloud scale. Bucket name restrictions should prevent problematic bucket names from being created. For security, HCP for cloud scale passes the user display name through a uri encoder before returning it in XML responses.
Object tagging	<p>Amazon wraps eTags in double-quotes. For XML listings (v1 object, v2 object, version) it escapes these, for example:</p> <pre data-bbox="894 1465 1409 1528"><ETag>&quot;32c81604d07395b1aa39a7e206c3af06\$&quot;</ETag></pre> <p>It's not necessary for HCP for cloud scale to perform this because double-quotes do not need to be escaped within content, only attributes.</p>

Feature	Implementation differences
	<p>Expiration Date URL Encoding (x-amz-expiration header)</p> <p>The RuleID portion of the x-amz-expiration header is URL-encoded by HCP for cloud scale using the same encoding strategy that Amazon suggests for V4 authentication. This may result in encoded strings that do not exactly match how Amazon encodes RuleIDs in general. However, decoding them should always return the original string.</p>
GET Object ACL	<p>ACL email address grantee types are not supported. In AWS each grantee is specified as a type=value pair, where the type is one of the following:</p> <ul style="list-style-type: none"> ▪ <code>emailAddress</code> if the value specified is the email address of an AWS account ▪ <code>id</code> if the value specified is the canonical user ID of an AWS account ▪ <code>uri</code> if granting permission to a predefined group <p>HCP for cloud scale supports only <code>id</code> and <code>uri</code>.</p>
PUT Object ACL	
DELETE Multiple Objects	Fully implemented
POST Object	Fully implemented
Initiate/Complete/Abort Multipart Upload	Fully implemented
Upload Part	Fully implemented
List Multipart Uploads	Fully implemented

Unsupported S3 APIs

The following lists are the unsupported Amazon S3 API features.

Authentication API

- Authentication v2 (deprecated by AWS)

Bucket APIs

- GET/PUT/DELETE Bucket Website
- GET/PUT/DELETE Bucket Policy
- GET/PUT/DELETE Bucket Tagging

- GET/PUT/DELETE Bucket CORS (cross-origin resource sharing)
- GET Bucket Location
- PUT Bucket Versioning (versioning is always On)
- GET/PUT Bucket Logging
- GET Bucket Notification
- GET/PUT Bucket requestPayment
- GET/PUT/DELETE Bucket Inventory
- List Bucket Inventory Configurations
- GET/PUT/DELETE Bucket Replication
- GET/DELETE Bucket Metrics
- List Bucket Metrics Configurations
- GET/PUT/DELETE Bucket Analytics
- List Bucket Analytics Configurations
- PUT/GET Bucket Accelerate
- Server-Side Encryption with Customer-Provided Encryption Keys (SSE-C)
- Server-Side Encryption with Storage-Managed Encryption Keys (SSE-S3)

Object APIs

- PUT Object (Copy)
- Options Object
- GET/POST Object Torrent
- SELECT Object Content (SQL)
- Upload Part - Copy

Hitachi Vantara



Corporate Headquarters

2535 Augustine Drive

Santa Clara, CA 95054 USA

HitachiVantara.com | community.HitachiVantara.com

Contact Information

USA: 1-800-446-0744

Global: 1-858-547-4526

HitachiVantara.com/contact