IBM Cloud Object Storage System™
Version 3.15.3

Container Mode Guide

This edition applies to IBM Cloud Object Storage System and is valid until replaced by new editions.

Contents

Chapter 1. Container Mode Guide

The system supports two different modes of operations: Vault Mode and Container Mode. By default, the system operates in Vault Mode. In Vault Mode, all object I/O occurs at the vault level. Systems can contain a limited number of vaults (see *IBM Manager Administration Guide*). Container Mode can be enabled when more buckets than the vault limit would otherwise allow are required. In Container Mode, containers are created inside of vaults. Object I/O occurs on containers instead of vaults.

Table 1. Advantages of Container Mode

Container Mode	**Vault Mode**
Support millions of buckets	Support for a limited number of buckets
Support for millions of users	Support for thousands of users
Support for fast and high performance listing with index updates (system OPs greatly reduced for listing)	Slow listing and unreliable performance
Support for usage based billing based on top-edge requests via access logs	Billing is supported based on estimated usage for storage and network resources on a vault basis
Support for optional storage metrics cluster (SMC) with REST API support for billing	No support for billing via storage metrics cluster
Support for self service portal for millions of end users via service API	Provides portal for end users to manage buckets (vaults)
Support for S3 Compliant List-Only ACL	S3 ACLs are supported, but as Read-and-List ACL
Support for additional S3 APIs (for example, **GET SERVICE**)	Support for standard S3 APIs previously supported

Container Mode has the following advantages:

- Isolation of objects between users and tenants
- Security for objects
- Ability to control usage programmatically
- Ability to isolate issues for a user or tenant

Note: This document references other documents that are available in the *IBM Knowledge Center* (http://www.ibm.com/support/knowledgecenter/STXNRM). See: *IBM Container Mode Credentials Management API Developer Guide*, *IBM Container Mode Service API - Bucket Management Guide* and *IBM Manager REST API Developer Guide*.

Overview

In Container Mode, each container vault can store over 1 million containers. When enabled, I/O occurs at the container level instead of the container vault level. ACLs are enforced at the container level. Metrics are tracked at the container, storage account, and container vault levels. Containers and storage accounts are not visible on the Manager. Accesser® Devices can be used to create, update, and delete containers.

Container Mode supports the following APIs:

- *IBM Manager REST API Developer Guide* for administration and monitoring
- APIs for service providers
  - *IBM Container Mode Storage Account Management API Developer Guide*

  - *IBM Container Mode Credentials Management API Developer Guide*
  - *IBM Container Mode Service API - Bucket Management Guide*
- *IBM Cloud Storage Object API 2.5 Developer Guide* for end users to manage data I/O, containers, and objects

Limitations

Table 2. Limitations with Container Mode

Container Mode	Vault Mode
No support for Swift, SOH, DDN/WOS, or HDFS.	Top-edge support for Swift, SOH, DDN/WOS, and HDFS.
No support for S3 versioning.	S3 versioning support.
No support for data migration and proxy.	Data migration and proxy support.
No support for Keystone and AD for user accounts and credentials management.	Keystone and AD support for user accounts and credentials management.
No Manager visibility for user accounts and credentials.	Manager visibility for all user accounts and credentials.
No support for mirroring.	Vault mirroring support.
No support for locked or private vaults.	Private or locked vaults support.
No support for S3 bucket tagging.	S3 bucket tagging support.
No support for Accesser application.	Accesser application supported.
No support for Embedded Accesser.	Embedded Accesser supported.

The following requirement must be met before you enable Container Mode:

- Native File Interface cannot be enabled on the same system.

When in Container Mode, the system has the following restrictions:

- The only API type that is supported is Cloud Object Storage.
- Name index must be enabled on all container vaults. It is done automatically when container vaults are created.
- Versioning is not supported.
- Private vaults are not supported.
- Mirrors are not supported.
- Delete restricted vaults are not supported.
- Container Mode can be disabled only when all container vaults are deleted.
- Container vault deletion is only supported by assistance from IBM Support.
- The service vault can be deleted only after Container Mode is disabled.
- Use of the embedded Accessor service feature is not recommended.

Capabilities

A large cloud object storage (COS) system needs several crucial capabilities to offer storage as a service (STaaS).

These capabilities give an object storage system the flexibility to be deployed as private, public or hybrid cloud solution.

- Support for millions of containers

- Support for millions of users
- Self-provisioning capability for service
- Support for billing users based on usage
- Isolation of objects between users and tenants
- Security for objects
- Ability to set quotas for tenants
- Ability to isolate issues to a tenant
- Ability to dynamically scale system and add capacity

Container Mode allows an IBM COS system to be deployed for private, public or hybrid cloud with capability across all these areas to various degrees to offer storage as a service. Without these capabilities, a COS deployment for a cloud using the existing Vault Mode has the following constraints:

- A , which also limits the number of user buckets
- No more than a few thousand user accounts
- No ability for self provisioning service with a special service user role
- No detailed billing information in order to support all billing models (e.g. request based pricing)

Supporting large number of containers and large number users all using the system concurrently while tracking real time data usage requires additional focus on consistency and performance. To address these areas, several enhancements to current design were introduced as part of the feature to achieve the following:

- Consistency of index and usage updates
- Fast listing
- Better handling of contention

Container Mode brings the following capabilities to each of the areas listed above in the IBM COS solution:

Support for millions of containers
A new logical entity called container is introduced on top of vault (container vault). The containers are addressable externally via REST API requests. External requests cannot address or access vaults directly in Container Mode. The new entity has fewer overheads for creation and deletion within the system. Containers can be quickly created or deleted (in the order of milliseconds). There is no visibility for these containers on the Manager.

Support for millions of users
All end user accounts and credentials are now created and stored outside of the manager within the IBM COS system. This makes it possible to support millions of users without the limitations that come with the Manager. There is no visibility for these users on the Manager. The account and credential data is stored in a private vault called a service vault.

Self-provisioning capability for service
The IBM COS system now supports a new set of REST APIs with the Accesser appliance for storage account management and AWS credentials management. A self-service portal or another application using a service role only on a different port can use these APIs.

Support for billing users based on usage
The IBM COS system now has detailed logging that includes account, container and usage information at the different levels. These logs are available to business support systems to process them to create per-tenant bills.

Isolation of objects between users and tenants
The IBM COS system supports isolation of objects between users and tenants.

Security for objects
The IBM COS system now has an owner assigned to every container and object. In addition, the system enforces ACL permissions across writes and reads (including more granular permissions). An

enhanced ACL enforcement mode is also available to more closely match the S3 ACL model. This is enabled by default.

Ability to control usage programmatically

The IBM COS system continues to support hard and soft quotas at the vault level.

Ability to isolate issues

The IBM COS system has more detailed logging and additional log files when the system is in Container Mode. The detailed logging will help isolate issues for a user or tenant in a large deployment.

Ability to scale and grow, as capacity has to expand

The IBM COS system continues to support system expansion as the need arises to grow the system to increase capacity.

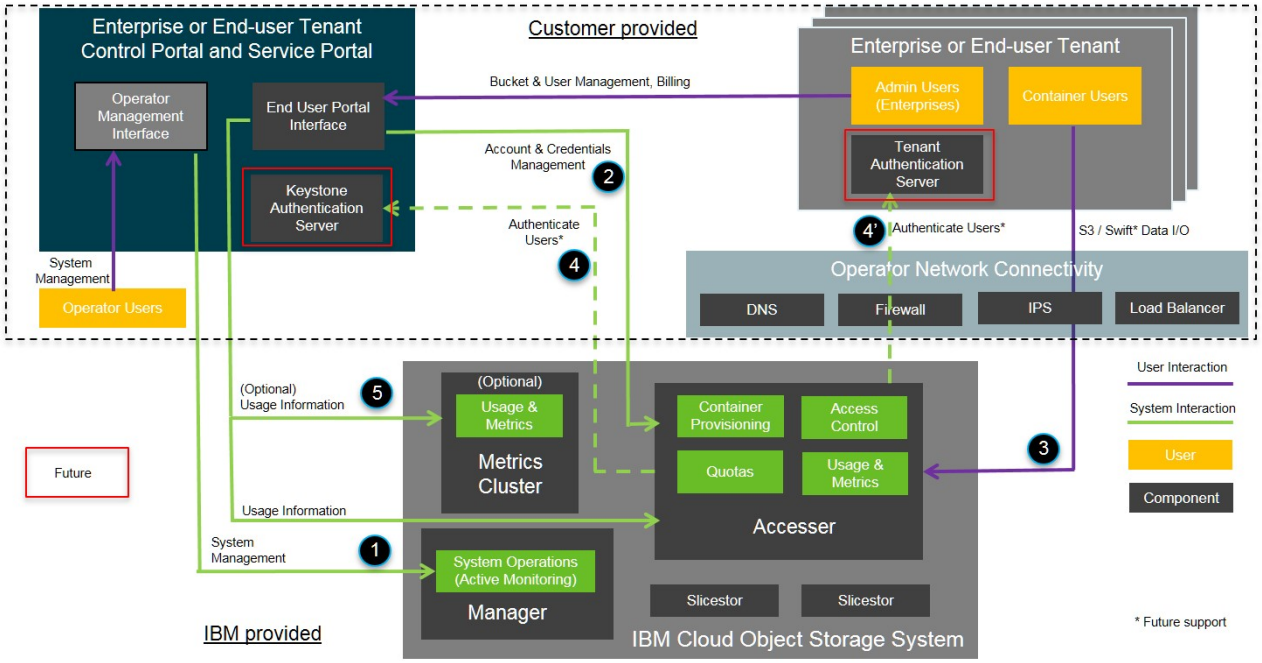

Figure 1. End-to-end STaaS system architecture

Table 3. API Interface points for IBM COS with STaaS

Callout in STaaS architecture diagram	Purpose	IBM element	APIs
1 - Container mode administration	To administer and monitor container mode	Manager	Create service vault, Enable container mode and service API, Manage service accounts, Manage container vaults
2 - Storage account management	To manage storage accounts and associate them to tenants (enterprises and users)	Accesser	Create, List, Head, Modify, Delete storage accounts. List containers
2 - AWS credentials management	To manage credentials that are associated with the storage accounts	Accesser	Create, List, Show, Delete, Update AWS credentials
2 - Bucket management	For service provider to manage container on behalf of clients	Accesser	Create, retrieve/, update bucket metadata, delete bucket

Table 3. API Interface points for IBM COS with STaaS (continued)			
Callout in STaaS architecture diagram	**Purpose**	**IBM element**	**APIs**
3 - S3 API	For data IO and container/object management	Accesser	Read, Write, List, Delete, Head for containers and objects
4 - Not supported	(future)		
4' - Not supported	(future)		
5 - SMC Interface S3 API	Refer to SMC documentation		

Terminology and components

ACL

Access control list (ACL) is a list of permissions attached to a container or an object. An ACL specifies which accounts are granted access to a given container or object.

AWS credentials

Amazon web services (AWS) credentials are security credentials to verify the user and if they are authorized to access the resources being requested.

Container
A new massively scalable logical entity that is accessible by users for storage.

Container vault
A vault that is hosting containers within an IBM COS system enabled for Container Mode. These vaults are similar to standard vaults in Vault Mode, except that they cannot be accessed directly by users. A container vault contains all the container metadata like ACLs, Usage information etc. A container vault also contains the actual objects and indexes. There could be many container vaults created in a system. The container vault that will host any given container is determined by the region used by the user in the request (the region needs to contain the provisioning code of the desired container vault).

Default Container Vault
A default container vault is a vault that will host containers for requests with no explicit request with a region (i.e. no provisioning code provided in the region request). This is configured as part of the access pool configuration on the Manager UI.

Index
A data structure used to perform efficient ordered listing of objects within a container.

IOP
Input output operations. Basically read, write, delete, head and other user requests.

List-Only ACL

Previously known as S3-Compliant ACL. When granted this ACL on a bucket, the user can only list objects. To read objects user needs to get explicit read access on required objects.

Management vault
A vault that is created by system administrators where access log and statistic files are uploaded periodically. These logs can be retrieved and processed for billing, issue isolation, support, etc.

Metadata
Information about account, credentials, containers and objects that are stored within an IBM COS system. The information could be system generated or user provided.

Read-and-List-ACL
Previously known as Legacy ACL. When granted this ACL on a bucket, the user can read and list objects.

Service role
A Manager user must have this role assigned in order to use the service API and interact with the service API ports on the Accesser.

Service vault
A vault that is created when a system administrator enables Container Mode. The vault contains system-generated data such as container pointers (references), storage account metadata; AWS credentials and Index for the different types of system data. There can be only a single instance of the service vault on a system in Container Mode.

Storage account
A billable entity on the IBM COS system that has a share of the available resources. AWS credentials are tied to storage accounts. Permissions are enforced for this entity using ACLs.

Usage
Metrics to track resources consumed by storage accounts and containers. This information is logged in the access logs as a separate entry type specific to usage to support billing by parsing and processing the logs.

System administrator
An end-user with an IBM COS credential with Manager Super User or System Administrator role who manages user and vault using Manager Web Interface or Manager REST API.

Service administrator
Service Admin is used generically to represent an IBM COS credential with “Service Account” role that can perform Service API to manage the account, key, and bucket resource. This user does not have access to user data and is not the same as the End-User who owns buckets.

End user
End user is used generically to represent an IBM COS end-user or an IBM Cloud service/application that is interacting with COS.

Workflow

You can enable Container Mode on a new or existing IBM Cloud Object Storage System. During Container Mode enablement, the system administrator is required to create a service vault.

Impact to system after enabling Container Mode

- All end user accounts must be managed through the Service API on the Accesser devices. There is no visibility to user accounts on the Manager. This includes storage accounts and AWS credentials.
- The maximum throughput (thereby IOPs) and minimum latency are negatively impacted by enabling Container Mode.

Storage account, AWS credentials, and container management
All credentials are associated with storage accounts, and are created after appropriate accounts are created.

After Container Mode is enabled, all user requests are executed on containers instead of vaults. This means that a user must have the correct storage account and credentials created through the Service API before they can successfully access an IBM COS system in Container Mode for reads and writes.

The Container Mode Storage Account Management and Container Mode Credentials Management APIs are Service APIs that support the following operations:

Container Mode Storage Account Management:

- Account listing
- Account creation
- Account retrieval
- Show account details
- Account modification
- Accounts deletion

Container Mode Credentials Management:

- Create credential
- List credential
- Show credential Details
- Delete credential
- Update credential

Additionally, IBM COS provides flexibility for service providers to have full control over bucket resources and to prevent end users from using the Cloud Storage Object API to perform a subset of container-level operations (such as create bucket, delete bucket, and access or update security information such as ACLs). These service providers can use the Service APIs to manage containers on behalf of their client, including:

- Create container
- Retrieve container metadata
- Update container metadata such as ACL, IP access control, and hard quota
- Delete container

Retrieval of access logs
All Service API requests are logged in the access log with the interface type "service". Each entry in this log corresponds to a request received on the Service API port of the S3 user request port. Access log entries have a different format in Container Mode. Additional fields with detailed usage information, such as bytes used and object count, can be used for billing. These fields can be aggregated for storage accounts and users.

Access logs are uploaded to the management vault periodically. The management vault has to be deployed to an appropriate access pool in order to allow a service user to download log files from a management vault.

Use cases

If a deployment must support a large number of S3 buckets (greater than 1K) and a large number of users (greater than 1K), then it makes sense to enable Container Mode. Enabling container mode triggers additional fields to get logged in the access logs and enables new internal structures and flows that can impact performance negatively. A system administrator must understand these impacts before deciding to enable Container Mode.

In addition, Container Mode does not support all functionality available in legacy Vault Mode. System administrators must understand and address these differences before choosing to enable Container Mode.

Enabling Container Mode is an irreversible process. An IBM COS system enabled for Container Mode cannot be taken back into Vault Mode.

Container Mode provides flexibility for service providers to choose whether to have full control over bucket resources and disable end user capabilities.

Design decisions and limitations

Migration from Vault Mode

This feature adds the ability to convert standard vaults to container vaults and support mixed-mode operation, with some access pools operating in Container Mode while others operate in Vault Mode. Options are available during conversion to specify Container Mode preference whether to use "DNS-compliant" Container Mode or "Create only container vault" whether to transfer IP and quota from standard vault to the first container and whether to only restricted the bucket level configuration to the service APText.

Migration of users and accounts
Manager users will not be able to read or write to containers (unlike in Vault Mode). These users have to be provisioned via the service API on the Accessers (storage accounts and credentials). Only AWS style authentication is supported in Container Mode. The credentials created via service API will have to be used by the users to make IO requests.

During Vault Mode conversion, the Manager account and credentials associated with standard vaults are migrated to the storage account and credentials on Accesser devices.

Migration of access control, IP access control, and hard quota
During the conversion, the bucket owners are identified, and the ACLs are migrated to the container. The user can determine whether to transfer the IP access control and the hard quota to the container, or to keep them in Vault Mode only.

If both IP access control and hard quota are specified for both the container vault and container, then IBM COS enforces both.

Compatibility with old clients
In Container Mode, only the AWS S3 API is supported. The format and methods supported in Container Mode is the same as in Vault Mode for AWS S3.

Be sure to use non-S3 compliant, or Read-and-List ACLs after conversion so that the ACL behavior for the converted container is compatible with the old client. In other words, an end user with READ permission can list objects from a bucket as well as can read the object from a bucket.

Performance implications

Enabling container mode will allow the system to scale for number of buckets and accounts supported in the system, as compared to vault mode. The performance characteristics of the system in container mode is expected to be different from the performance characteristics of the system in vault mode. In container mode, there are additional internal system operations needed in order to support significantly larger number of buckets and accounts. These additional internal system operations involve maintaining indices, and maintaining account information in internal vaults. Depending on the number of operations per second, number of objects in a bucket and object overwrite percentage for the system, the performance in container mode could be lower than the performance in vault mode for both operations per second and latency.

Vault Mode

Account

Cloud Object Storage accounts are defined in the Cloud Object Storage Manager and are used for vault level I/O.

- Basic authentication
- Access key authentication

The general roles that can be assigned to an account are:

- Super user
- System administrator
- Security officer
- Operator
- Service account

Except for the *Service account* role, each of these roles affects what the account can do and see in the Cloud Object Storage Manager, irrespective of whether the system is in Container Mode or not. However, the *Service account* role grants the account access to the Service API in Container Mode and cannot be set unless the system is in Container Mode.

If an account has access to a vault that is a part of a conversion to Container Mode, a *Storage account* is created for it (see "Storage accounts and credentials management" on page 10). In Container Mode, all I/O is performed by using access keys.

Note: In Container Mode, the *Service account* role is used for accessing the Service API.

Access control list (ACL)

The types of vault access are:

- Owner
- Read/write
- Read
- No access

These access types map to either the Read-and-List or List-Only ACL. The ACL is a sub-resource that is attached to every bucket and object. It grants users to read, write, or full-control permissions. The following table shows the common ACL behavior except for one explicitly mentioned that is not S3-compliant.

Table 4. Vault ACLs

Permissions	**ACL**
Read	Allow grantee to list object in the bucket. **Note:** The system's Object Access property determines the behavior of the READ ACL. It can be configured with one of the following: • Grant list access to the container or vault (S3 compliant behavior). • Grant list access or read-and-list access to all objects in the container or vault. See also: **Administration** > **Configuring system properties** in the *IBM Manager Administration Guide* and **Administration** > **System properties configuration** in the *IBM Manager REST API Development Guide*.
Write (Read/Write)	Allow grantee to create, overwrite, and delete any object in the bucket.
Read_ACP	Not supported. The default is full_control, implied by the bucket "owner" permission.
Write_ACP	Not supported. Default is full_control.
Full_control (owner)	Allows grantee read, write, read_ACP, and write_ACP permissions on the bucket.

A System Administrator can grant a user individual object READ permission using the Cloud Storage Object API's `PUT Object ACL` operation.

In Vault Mode, a vault cannot be granted to any grantee with "Owner" permission. A system administrator can also configure whether the end user can use storage APIs (for example: SOH and S3) to create new vaults or delete existing vaults using the Provisioning API defined in the *Manager REST API Development Guide*.

IP access control

IBM COS supports Allowed IP in Vault Mode to control bucket access only to the trusted IP addresses

A client can access a bucket in various ways, through a direct connection or a proxy connection. When a client connects to IBM COS Accesser® device directly, it is considered a direct connection, such that the client IP address is retrieved from the client transport connection information. There is a special case of the direct connection where the client connects the Accesser® device through a proxy server, such as a

load balancer, while the proxy server is configured to preserve the client source IP address, the client IP address can then be retrieved from the client transport connection information. The customer can whitelist the client public IP addresses to be allowed to access the bucket when the connection is a direct connection. IBM COS will enforce IP access control based on the client IP address. When the client connects to the Accesser® device through a proxy server such that the client IP address cannot be retrieved from the client transport connection information, the connection is considered as a proxy connection. To use the IP access control capability for the proxy connection, the system application must set the client originating public IP address at the rightmost proxy IP address in the X-Forwarded-For HTTP header, and the system admin must choose the "proxy" connection in Manager Web Interface. Please refer to IBM COS Manager Web interface -> Administration -> Network Transport Layer Configuration to set proper connection type and public client originating IP address.

Soft and hard quota

The IBM Cloud Object Storage System maintains both soft quota and hard quota at the vault level. A system administrator can specify these values through the Manager Web Interface on the **Create Vault**, **Create Vault Template**, or **Edit Vault** pages, or through the methods in the Manager REST API.

The Accesser® device prevents users from exceeding the hard quota of the vault. The hard quota is often used to manage usage for billing purposes. The system does not restrict physical usage according to soft quota, but exceeding the soft quota is a good indicator that you should plan for vault migration to a larger storage pool or to expand the storage pool.

The system enforces vault quota based on the vault usage. In a short window, a successful write might result in exceeding the hard quota. In this case, the system restricts additional write operations after the hard quota is exceeded. For instance, if a vault's hard quota is 100 GB and its current usage is 99GB with a 0.3 GB new write request, then the system allows the write or multipart upload request. However, if the usage is refreshed to 100.1 GB immediately after authorizing this request without counting the size of in-progressed writes, the user is not able to write more objects until usage is brought below 100 GB by deleting objects from the vault.

Container Mode

Storage accounts and credentials management

With the introduction of Container Mode, the concept of a storage account is introduced. A storage account is an entity on which users and containers are created. Storage accounts are often the entity that is billed for storage usage.

Each storage account in the system has at least one set of access keys, which are needed for all Cloud Storage Object requests on the container. All credentials are associated with storage accounts, and both are created through the Service APIs after enabling Container Mode, so that user read and write requests are executed on containers.

Access control list

During vault conversion, the system maintains the ACL behavior. You can change the ACL after vault conversion.

A System Administrator can grant a user individual object READ permission using the Cloud Storage Object API's PUT `Object` ACL operation. Then the user can switch between the ACL options that:

- Grant read access to all object data and metadata, and grant list access to the container or vault, or
- Grant list access only, to the container or vault.

See also: **Administration** > **Configuring system properties** in the *IBM Manager Administration Guide* and **Administration** > **System properties configuration** in the *IBM Manager REST API Development Guide*.

Bucket owner

In Container Mode, each bucket must have a bucket owner.

A bucket owner has implicit “Owner” permission such that there is no explicit ACL granted to the bucket owner if S3 ACL operation is used for authorization. On a newly created bucket (container) or object in Container Mode, the system grants the resource owner full-control permission over the resource.

Accounts and conversion to Container Mode

As part of Container Mode conversion, the IBM Cloud Object Storage System determines the storage account which owns a bucket according to the following criteria, and assigns the storage account to the bucket owner:

1. The first user chronologically granted Manager credentials with "owner" permission for the vault.
2. If no user has "owner" permission to the vault, then the first user chronologically granted with "read/write" permission to the vault.

   Note: If a standard vault does not have an owner assigned before conversion, a user with "read/write" permissions is promoted to the bucket "owner" during conversion and has full control permission. This promotion may result in a user having the following new permissions after conversion:

   - Delete a bucket
   - Set or retrieve ACL, CORS, or retention policy etc.
3. Conversion is not allowed if the vault does not have an assigned owner or user with read/write permission. In this case, a System Administrator must adjust permissions prior to enabling Container Mode.

The system creates storage accounts for other users and explicitly grants these users their original permission to the bucket in the bucket ACL. All storage accounts can create a bucket (container) using the S3 command in Container Mode unless this is disabled as described in “Restrict Bucket Owner S3 Operation in Container Mode.”

To avoid unanticipated additional permissions, a System Administrator must ensure that the bucket owner is the desired bucket owner after conversion. Otherwise, adjust the authorization on the **Vault Authorization** (see *IBM Manager Administration Guide*) page of the Manager Web Interface or the **`Edit vault authorization`** (see *IBM Manager REST API Developer Guide*) command of Manager REST API prior to conversion.

The following two tables illustrate the examples of storage account, bucket owner, and ACL assignment during conversion. The first table represents a scenario in which a user granted only with a read/write permission is assigned to the vault.

Table 5. Convert an account granted only read/write vault permission

Vault Mode			**Container Mode**				
Manager User	**Permission**	**Vault**	**Storage Account**	**Container**	**Permission**	**Bucket owner**	**Explicit ACL entry**
acct1	Read/Write	bucket1	acct1	bucket1	full_control	Yes	Yes

The following table represents a scenario in which multiple users are granted with either owner or read/write permission to access the vault in the order of acct2, acct3, acct4. In Container Mode, the system creates the storage account for all accounts in Vault Mode and explicitly adds ACL entries to the bucket for the storage accounts other than the bucket owner.

Table 6. Convert multiple accounts granted owner and Read/Write vault permission							
Vault Mode			**Container Mode**				
Manager User	**Permission**	**Vault**	**Storage Account**	**Container**	**Permission**	**Bucket owner**	**Explicit ACL entry**
acct2	Read/Write	bucket2	acct2	bucket2	write	No	Yes
acct3	Owner	bucket2	acct3	bucket2	full_control	Yes	No
acct4	Owner	2ucket2	acct4	bucket2	full_control	No	Yes

Restrict bucket owner Cloud Object Storage API operations

In Container Mode, by default, the user with bucket "full_control" permission can use S3 to retrieve security information (such as ACL or CORS) or create or delete a bucket.

In Vault Mode, the managerAdmin_administration_configure_provisioning_api.dita provides flexibility for a service provider to have full control for all buckets. Similarly, in Container Mode, the system allows a user to configure through the Manager Web interface or Manager REST API whether an end user can perform operations that are only allowed for those with the "full_control" permission.

A System administrator can restrict an End-User to not allow some bucket-level S3 operations, by selecting a checkbox on container mode configuration to allow create, delete, configure bucket operations only via service API.

It can be applied to all access pools or customized for each access pool. When set to false on the manager UI or API, the user from all access pools with the bucket "full_control" permission cannot use S3 to retrieve security information, such as ACL or CORS on a bucket or create/delete a bucket, and can only perform operations allowed for users with read/write permission. If the “full_control” permission is needed for some users to perform the S3 operation in some vaults, the System Admin needs to deploy these vaults into a separate access pool and contact IBM support to enable these bucket-level S3 operations on this access pool for the end users.

Table 7. Allowed S3 operations				
Standard Vault		**Converted to container Mode**		**Restricted S3 operation mode**
Permission	Supported S3 bucket Operation	Permission	Supported S3 bucket operation	Supported S3 bucket Operation
Read/Write (1st writer and no owner for the bucket)	HEAD bucket	Owner(full control)	Complete bucket-level S3 Operation set	HEAD bucket
Read/Write (Other users)	HEAD bucket	Read/Write	HEAD bucket	HEAD bucket
Owner (full_control)	Complete bucket-level S3 Operation Set	Owner (full_control)	Complete bucket-level S3 Operation Set	HEAD bucket

Account for IO and Service APIs

The following three tables show a scenario in which an account has access keys, permission to perform I/O on a vault, and the *Service account* role in the Cloud Object Storage manager.

Table 8. Account in Cloud Object Storage Manager		
Name	**Access key ID**	**Secret access key**
acct1	**GQn7t4SOu3LXrcCIRpKn**	**ShZAq0dEdrAX6D6B4ijUK**

Table 9. Standard vault permissions in Cloud Object Storage Manager		
Vault name	**Account name**	**Permission**
vault1	**acct1**	**readWrite**
vault2	**acct1**	**readWrite**

Table 10. General Cloud Object Storage Manager roles for accounts	
Account name	**General roles (list)**
acct1	**system administrator, service account**

When you perform a Container Mode conversion in this scenario, when **vault1** is converted to Container Mode, a *Storage account* is created for **acct1**. The access key credentials in the first table can be used to perform I/O at the container level.

Having a *Storage account* that is created for the account does not prevent the account from having access to the Service API. Access to the Service API is solely determined by the existence (or a lack thereof) of the *Service account* role. Thus, in this situation, the *Account* has access to the service API and the *Storage account* has access to container level I/O, both with the same set of access keys defined in Table 8 on page 13.

In addition, even after **vault1** is converted to a container vault, a standard vault **vault2** still presides on the system. Since **acct1** has permission to the vault, it can perform vault level I/O to vault **vault2** by using basic authentication or by using the access keys that are defined in Table 8 on page 13.

Access key credentials that are used by the *Storage account* for container level I/O can be changed. When new access keys are added or removed by using the Service API, the Cloud Object Storage Manager does not show the updates. The access keys that are used by the *Storage account* are not managed by the same entity as the ones used by the *Account* in the Cloud Object Storage Manager.

Similarly, if access keys are added to the account in the Cloud Object Storage Manager, these keys can be used for vault-level I/O or the Service API, but they cannot be used for container-level I/O.

For these reasons and to avoid confusion, it is not recommended that you use the same base account (**acct1** in this case) and its storage account for accessing the Service API and performing container I/O. It is recommended that you create new accounts in the Cloud Object Storage Manager for accessing the Service API. In most cases, confusion can be avoided when all vaults that a single account has access to are converted at the same time (for example, **vault1** and **vault2**). It avoids a mixed Vault Mode/ Container Mode scenario.

IP access control

In Container Mode, the system supports both allowed IP and denied IP addresses on the container level.

For some on-premise customers, vault-level specification provides efficient configuration when all containers belong to the same organization; on the other hand, the container-level IP specification offers flexibility when each container requires different client IP.

When the allowed IP addresses are specified on both vaults and containers, the system denies S3 operations for all users and applications if the client IP is outside the `allowedIp` for either vault or container.

IBM COS supports the enforcement of the container IP whitelisting or blacklisting when the client connects to the Accesser© device directly, or through a proxy server. Please refer to IBM COS Manager

Web interface -> Administration -> Network Transport Layer Configuration to set proper connection type and the client originating IP address.

The system also allows a Service Administrator to create lists of denied IP addresses for a container to block user to access the container from these IP addresses, such as those suspected to be malware or spam, or allow access from the wider `allowedIp` list except the narrow sub-ranges specified in `deniedIp`. There is no denied IP support at the vault level.

Transfer IP access control during conversion

Prior to conversion to Container Mode, a System Administrator must determine and configure whether to transfer IP access control from standard vault to a container vault or to the first container. For more information, see "Enabling Container Mode with an existing system" on page 26 or set corresponding "transferAllowedIpsToContainer" value using "editContainerModeSettings.adm" Manager REST API. IP access control is enforced during the conversion to Container Mode.

Container hard quota

In addition to supporting both soft quota and hard quota on the vault level, the system also enforces the hard quota on a container for customers who manage billing-based container usage.

If a hard quota is set in a container, and if an end user specifies the content-length in an HTTP request header, the system rejects new writes when the sum of content-length and container usage is greater than the container hard quota. However, if the content-length is not available from the write request header, the system rejects the next write operation after the quota is exceeded, based on the updated container usage.

For example, if bucket quota is 100 GB and its usage is 99GB with a new write request of 10 GB (size is unknown in the header), the write request is allowed, and the bucket usage after the request is then 109 GB. A user is not able to write more objects until the bucket usage is brought below the 100 GB quota by deleting objects from the bucket.

In streaming, such as the multiple-part upload command where the content-length is not in the HTTP request header, the system rejects current and remaining parts if the sum of the estimated chunk size and known bucket usage is greater than container hard quota.

The system enforces container quota based on the container usage, which is updated every 10-20 minutes in Container Mode.

Prior to Container Mode conversion, a system administrator can choose whether to transfer the hard quota from the standard vault to the container, or to keep it in the container vault. System administrators can accomplish this through the "Configure Container Mode" option on the Manager Web Interface or by specifying the value of "transferHardQuotaToContainer" using "editContainerModeSettings.adm" Manager REST API. Soft quota remains at the vault level during the conversion. The vault hard quota is enforced during conversion.

When system administrators specify hard quota on both the vault and container levels after enabling Container Mode, the system enforces both.

Table 11. Example requests for quotas at the container vault and container levels

	Container vault quota	**Container quota**	**Container vault usage**	**Container usage**	**New write to container**	**Result**
Example 1	100 TB	2 TB	99.999 TB	0.999 TB	0.002 TB	Rejected
Example 2	100 TB	2 TB	98 TB	0.999 TB	0.002 TB	Accepted

Using the Service APIs and the Manger REST API

When deploying Container Mode in an existing system, you can progressively convert sets of standard vaults. Therefore, you might need to use different APIs to perform account, key, and bucket management operations while the system is in vault, mixed, or Container Mode.

- In Vault Mode, use the Manager REST API or the Manager Web Interface.
- In mixed mode, use the Manager REST API for standard vaults and corresponding account and keys; use the Service APIs for containers and corresponding accounts and keys.
- In Container Mode, use the Service APIs.

The following table describes the equivalent operations for the Service API in Container Mode and the Manager REST API in Vault Mode for typical account, key, and bucket operations.

Note: You must use the Manager REST API or Manager Web Interface to create or delete a vault.

Table 12. Container Mode Service API operations vs. Vault Mode Manager REST API operations

Category	**Operation**	**Manager Role**	**Manager REST API in Vault Mode**	**Service APIs in Container Mode**
Account management	Create account	Super User, Security Officer	CreateAccount.adm	PUT <accesser>:8338/accounts/{account.id}
	Delete account	Super User, Security Officer	deleteAccount.adm	DELETE <accesser>:8338/accounts/{account.id}
	Edit account	Super User, Security Officer	editAccount.adm	POST <accesser>:8338/accounts/{account.id}
	Retrieve account	Super User, System Administrator, Operator	listVaults.adm	Retrieve account HEAD <accesser>:8338/accounts/{account.id}
	Container Listing	Super User, System Administrator, Operator	listVault.adm	GET<accesser>:8338/accounts/{account.id}/containers

Table 12. Container Mode Service API operations vs. Vault Mode Manager REST API operations (continued)

Category	Operation	Manager Role	Manager REST API in Vault Mode	Service APIs in Container Mode
AWS Credential mangement	Create an access key	Super User, Security Officer	editAccountAccessKey.adm	Create credential POST <accesser>:8338/credentials
	Update an access key	Super User, Security Officer	editAccountAccessKey.adm	PATCH <accesser>:8338/credentials/ [credential id]
	Delete an access key	Super User, Security Officer	editAccountAccessKey.adm	DELETE <accesser>:8338/credentials/{credential.id}
	List My access Keys	All	listMyAccessKey.adm	List credential or show credential deatails GET <accesser>:8338/credentials GET <accesser>:8338/credentials/ [credential id]

Table 12. Container Mode Service API operations vs. Vault Mode Manager REST API operations (continued)

Category	Operation	Manager Role	Manager REST API in Vault Mode	Service APIs in Container Mode
Bucket management	Retrieve bucket metadata (ACL, allowedIp etc)	Super user, System Administrator	view System.adm list Vault.adm	Retrieve bucket metadata GET <accesser>:8338/bucket/{bucket.name}
	Edit Bucket quota	Super user, System Administrator	editVault.adm	Update bucket hardQuota parameter PATCH <accesser>:8338/bucket/{bucket.name}
	Edit Bucket Access Control (ACL)	Super User, System Administrator	editVaultAuthorization.adm	Update bucket metadata, acl parameter PATCH <accesser>:8338/bucket/{bucket.name}
	Edit Bucket Allowed IP	Super User, System Administrator	editVaultAccessControl.adm	Update bucket metadata, firewall parameters such as the allowed or denied IP PATCH <accesser>:8338/bucket/{bucket.name}
	Setting bucket quota	Super User, System Administrator	editVault.adm	PATCH <accesser>:8338/bucket/{bucket.name}
	Delete bucket	Super User, System Administrator	deleteVault.adm	DELETE <accesser>:8338/bucket/{bucket.name

Log files

All requests through the Service APIs are logged in the access log with a different interface type.

Each entry for Service API requests is logged as `"interface_type": "service"`. For more information, see the *IBM Log File Reference Guide*.

To use the Service APIs:

- The Service API ports must be enabled for the access pools in use
- Only a user with the Service Account role can use the Service API ports
- All Service API requests must comply with the interface specifications
- Storage accounts cannot be deleted if they are not empty. This means that a storage account that is associated with AWS credentials, containers, or objects cannot be deleted.

Access log

Access log entries have a different format in Container Mode than in Vault Mode, and is defined in the *IBM Log File Reference Guide* . Each entry in this log corresponds to a request received on the Service API port or the S3 user request port. Container Mode also introduces additional fields that can be used for billing.

The interface `type` field specifies one of the following parameters for each entry in the access logs in Container Mode.

- `service`: All Service API requests received at the Service API port.
- `s3-service`: All vault-level access received at the Service API port (for example, management vault).
- `s3`: All user requests received.

The fields in the access logs must be processed and aggregated for storage accounts and users, as each entry in an access log is for a single request received at an Accesser device. The usage information includes both bytes used and object count. In addition, the legacy fields provide the size of the object for the current request.

A billing application can support any billing model based on the available information and processing of the logs. Each access log has the device UUID for which the access log has access information, and a timestamp associated with the name of the log. The logs are rotated as configured in the Manager Web Interface and appropriately are named with timestamps.

Retrieval of access logs

Access logs are uploaded to the management vault periodically. These logs include detailed usage information that can be used for billing. The management vault must be deployed to an appropriate access pool to allow a service user to download log files from the management vault.

A system administrator must configure a suitable access log rotation time (see *IBM Manager Administration Guide*) on the Manager Web Interface when creating or updating management vaults. All access logs are uploaded to the management vault periodically. An application can retrieve these logs from the management vault using the Service API port and service account credentials.

The URI format for management vault access in Container Mode differs from Vault Mode. The format is as follows:

```
http://<accesser-IP-address>:8337/vaults/s3/<management-vault-name>
```

```
https://<accesser IP address>:8338/vaults/s3/<management vault name>
```

Impact to system after enabling Container Mode

- Enabling Container Mode is an irreversible process. A system in Container Mode cannot be converted back into Vault Mode.
- All end user accounts must be managed through the Service APIs on the Accesser devices. There is no visibility to user accounts on the Manager, including storage accounts and AWS credentials.
- Any user or application using the Service APIs must have a new role assigned (service account role) to have permissions to manage storage accounts and credentials.
- If the system administer chooses to enforce DNS compliant container names, all new bucket names must be DNS compliant. Any existing bucket names that are non-DNS compliant will continue to be accessible.

- A storage account can have at least 1000 containers by default, but the maximum number of supported containers can vary. The S3 API's container listing command does not support pagination, meaning that having more than 1000 containers results in some containers not being listable by end users.
- Service vault availability and reliability is critical for the system to continue to operate in Container Mode and serve all user requests.
- Enabling Container Mode triggers additional fields to get logged in the access logs and enables new internal structures and flows that could impact performance negatively.

Enabling Container Mode

The Container Mode guide details the process of enabling Container Mode on an IBM COS system.

The common process for both new and existing systems includes:

- Creating a service vault
- Configure Container Mode, such as with DNS compliant only container names
- Enabling Container Mode
- Enabling service API ports

Existing systems must complete the following additional items:

- Ensure the correct bucket owner
- Control owner S3 operations
- Configure Container Mode conversion options
  - DNS compliant only containers
  - Create only container vaults
  - Whether to restrict the container management only to the service API
- Progressive conversion of these standard vaults of Container Mode.

These steps must be followed, in order, to successfully enable Container Mode. The steps can completed through the Manager Web Interface or the Manager REST API. See the relative chapters for specific examples.

The service vault is a critical vault that impacts system reliability and availability. Several workflows are impacted when a service vault is unavailable. Service vault availability is critical for the following use cases.

- Storage account and AWS credentials management
- Container related operations
- Generation of usage reports for billing

A system administrator must ensure that the service vault has the highest reliability and availability that the system can support. Several restrictions must also be addressed before Container Mode can be turned on for an IBM COS system. Some of these restrictions are enforced when Container Mode is being enabled on the system. The following restrictions are enforced when Container Mode is being enabled.

- The IBM COS system will not allow an operator or administrator to enable Container Mode if there exists vaults on the system. Any vault and their data must be deleted before Container Mode can be enabled.
- The IBM COS system will require an operator or an administrator to create a service vault that will contain the system data required to support Container Mode service, before enabling container mode

The following steps have to be taken by an operator or an administrator before end users are able to make requests to an IBM COS system:

- Appropriate service account roles must be created on the Manager. The service roles accounts can then be used to make service API requests. For example, a self-service portal built by the operator could use these accounts to provision users.

- The storage accounts and credentials for users have to be created via service API by a service account on the Manager, before any IO is possible in Container Mode.

Other restrictions will have to be understood by the system administrator and addressed appropriately before enabling Container Mode.

Table 13. Functions and level of support in Container Mode

Function	Container Vault Level	Container Level
IP allow and disallow	Yes	Yes
Device ACLs	Yes	No
User ACLs	No	Yes
Versioning	No	No
Delete restrictions	No	No
Quotas	Yes	Yes
Mirror	No	No
Proxy	No	No

Enabling Container Mode with a new system

About this task

Perform the following steps to enable and use Container Mode:

Procedure

1. Create a service vault.
2. Enable Container Mode.
3. Create a service account.
4. Create a container vault.
5. Configure the service API.
6. Create a Storage Account.
7. Create access keys.
8. Specify the ACL type for the new container.
9. Control owner S3 operation.
10. Create a container.
11. Specify container IP access control and bucket quota.

Creating a service vault

About this task

The service vault is used to store metadata that is related to containers, storage accounts, and access keys. Each IBM Cloud Object Storage System™ has a single service vault that only needs to be created once. It is important to select an IDA that ensures no data is lost from the service vault as data loss would lead to the loss of containers. Also, if the service vault is not accessible, all container I/O fails. However, selecting an IDA that is too wide results in reduced container I/O performance.

Note: The service vault must be created on a storage pool with packed storage enabled.

Procedure

1. Navigate to **Settings** > **Vaults** > **Container Mode**.

2. Enter the information for a new service vault and click **Create Service Vault**.

Enabling Container Mode

About this task

During this step, the Manager validates that the current configuration is compatible with Container Mode. Detected errors are displayed and must be fixed before you can continue.

You must abide by the following validation rules:

1. No standard vaults can exist.
2. The API type of all access pools must be cloud storage object.
3. A service vault must be present.

After creating the service vault, new container vaults can be created.

Procedure

Select the **Create only container vaults** option on the **Configure Container Mode** page.

All new vaults will be container vaults.

Note: To add new standard vaults to the system in addition to container vaults, you will be operating in mixed mode. More information on mixed-mode operation is below.

Creating a service account

About this task

Manager accounts must be used for authentication or authorization of service API requests and are needed to have the Service Account role. Complete the following steps to add the Service Account role to a manager account.

Procedure

1. Click the **Security** tab.
2. Create an account or modify an existing account in the **Accounts** section.
   - For a new account, click **Create Account**.
   - For an existing account, click the user name to whom you want to assign the role and then click **Change**.
3. Enable the **Service Account** role in the **Roles** section.

Creating a container vault

Create a container vault to house containers.

Procedure

1. Click the **Configure** tab.
2. Click **Create Vault** in the **Summary** section.
3. Choose the method by which you want to create a container vault:
   - **Using Template**: Select a template and click **Continue**.
   - **Custom Vault From Storage Pool**: Select a storage pool and click **Continue**.

     In the **Select vault type** dialog box, select **Container**.

     Optionally, select **Create only container vaults.** New vaults will be container vaults. If standard vaults are required, disable this option. This option will have no effect for vaults created on incompatible storage pools or vault templates.

Click **Continue**.

4. In the **General** section, complete the following fields:

   Tip: The provisioning code is located in the **General** section, and is used to specify the container vault in which you want to create containers.

Field Label	Acceptable Field Value
Name	Each vault must be uniquely named (maximum of 255 characters); this name is used by the Manager for all references to this vault. Vault names can include underscores and alphanumeric characters. The vault name can also contain periods (.), but the name cannot start or end with a period or contain more than one period in a row. The first character of the name must be a letter, underscore, or number.
Description	An optional free-form description can also be entered. Information that you might include in the description field might be initiator host name and IP address, names and phone numbers of administrators, and key users of the vault.
Tags	Tags can be created and or assigned to a vault before the vault is created. For more information, see the *Tags* section.
Organization	When you create a vault, you can assign it to an organization. The menu does not appear if you only have one organization. See *Creating an organization* in the Security chapter and *Editing an organization* in the Security chapter .

5. In the **Configuration** section, several options display.

Field Label	Acceptable Field Value
Width	This setting is referred to as the width of the vault and corresponds to the number of slices into which all data in the vault is split. Vault width must be a factor of the storage pool width. The Manager Web Interface allows any vault width greater than or equal to 6 and less than or equal to 60.
Threshold	The minimum number of slices that must be available to perform a read. Pre-defined, supported thresholds are presented when the drop-down list is clicked. The vault threshold, always less than the width, determines the reliability of the vault. If the set of available Slicestor® devices is such that the number of slices falls below this threshold, the vault content cannot be read, and the vault appears as red in the Monitor application. The Manager Web Interface allows any value between 1 and Vault Width, inclusive. If the vault is on a storage pool that spans multiple sites, the Manager Web Interface warns the user if the selected threshold is high enough such that a single site outage affects read and write availability.

Field Label	Acceptable Field Value
`Write Threshold`	The Manager Web Interface allows any value such that all the following are true: • Write Threshold > Threshold. **CAUTION:** Write Threshold = Threshold is allowed if Threshold = Vault Width or if Vault Width < 6. • Write Threshold ≤ Vault Width. • (Write Threshold + Threshold) > Vault Width. Write Threshold defaults to Threshold + 2, if that is within the allowed range. Otherwise, the selected Write Threshold is the halfway point between the minimum allowed Write Threshold and Vault Width, rounded up. This value is selected by default in the Write Threshold drop-down when Threshold is selected. This value is also used as the Write Threshold when a vault is created through the Manager REST API and a Write Threshold is not specified. If the vault is on a storage pool that spans multiple sites, the Manager Web Interface warns the user if the selected write threshold is high enough such that a single site outage affects write availability.
`Alert Level`	Optional. If the set of available Slicestor devices is such that the number of slices is between the write threshold and the alert level exclusive, the vault icon is yellow in the Monitor application. In this case, the vault is still fully functional.

Attention: If the **Threshold** is set such that the loss of one site would make Vaults either unusable or read-only, the Manager Web Interface displays a confirmation dialog box that asks the operator if they accept the settings with the risks they present.

If only the **Threshold** causes an issue:

```
Warning: This IDA configuration is susceptible to read availability issues during a
single site outage.
Do you still wish to continue?
```

If the **Threshold** and **Write Threshold** cause issues:

```
Warning: This IDA configuration is susceptible to read and write availability issues
during a single site outage.
Do you still wish to continue?
```

Click **Cancel** or **OK** to change or keep the settings.

6. If you enabled vault protection (see *IBM Manager Administration Guide*) on the system, choose a Retention setting.

   Note: This section displays only if **`Vault Protection Configuration`** is enabled in the **Configure** tab.

   - **Disabled**. The container vault does not support Retention.
   - **Enabled**: The container vault supports Retention for buckets in this container vault.
7. In the **Options** section, complete these fields:

Field Label	Description
Enable SecureSlice™ Technology.	Optional. SecureSlice™ provides extra encryption benefits that are combined with dispersal. This box is checked by default for newly created vaults. This feature can be cleared, although it is not recommended. If it is cleared, a warning message appears, and a confirmation is needed before proceeding. When a vault is created, the SecureSlice™ option cannot be changed.
Enable Server Side Encryption with Customer-Provided Keys	Optional. SSE-C is enforced on objects. Requests to read or write objects or their metadata using customer managed keys send the required encryption information as headers in the HTTP requests.
Enable Server Side Encryption with Key Protect (SSE-KP)	Optional. Use IBM Key Protect to create, add, and manage keys, which you can then associate with your instance of IBM® Cloud Object Storage to encrypt buckets.
Enable Archive Tiering (cannot be disabled)	Optional. This feature provides a service user the ability to create a bucket and assign an archive policy, ability to add an archive policy to an existing bucket, modify an archive policy for a bucket (only applied to newly written objects in the bucket), view the archive policy for a bucket, delete/disable archive policy for a bucket.

8. If you are using a vault-level quota, in the **Quotas** section, complete these fields:

Field	Value
Soft Quota	Optional. If wanted, select a value for a soft quota. A notification is sent to the **Event Console**, if the soft quota setting is exceeded. It does not cause restrictions to usage. Setting the quota higher than the total space available in one or more storage pools that are associated with this vault has no effect.
Hard Quota	Optional. If wanted, enter a hard quota value. The Accesser® device (or application) does not permit the user to exceed the hard quota value for this vault. A notification is also sent to the **Event Console** if the hard quota setting is exceeded. Setting the quota higher than the total space available in one or more storage pools that are associated with this vault has no effect.

9. Click **Save**.

   A new container vault is created.

Configuring the service API

You can create or edit an access pool to open the service API ports on at least one access pool.

About this task

By default, the service API ports are not opened because most access pools do not need to have these ports open.

Procedure

1. Click the **Configure** tab.
2. Perform one of the following steps to open the service API ports.
   - For a new access pool, click **Create Access Pool** in the **Summary** section. For more information, see *Create access pools* in the *Configuration* chapter of the *Manager Administration Guide*.

- For an existing access pool, click **Open All** above the left navigation tree and then click an access pool. Click **Change**.

3. Enable the service API ports in the **General** section.
4. Click **Save**.

Create a storage account
Storage accounts are entities on which users and containers are created. Often, storage accounts are the entity that is billed for the storage usage. The creation of the storage account is the first step in allowing access to the system.

Storage accounts are created through Service API requests, as in the following example:

```
curl "http://<accesser ip>:8337/accounts/<storage account name>" -X PUT -u <user
name>:<password>
```

- `<accesser ip>` is the IP address of an Accesser® Device that belongs to an access pool with the service API enabled. To use an IPv6 address, enclose the Accesser IP in square brackets: `[<accesser ip>]`.
- `<storage account name>` is the name of the storage account that is created. The storage account name must be unique.
- `<user name>` is the user name of a manager account that has the Service Account role.
- `<password>` is the password of the `<user name>`.

Create access keys
After you create a storage account, you can create access keys for users that have access to the storage account. Access keys are needed for all Cloud Object Storage requests on the container.

Access keys are created through Service API requests, as in the following example:

```
curl "http://<accesser ip>:8337/credentials" -X POST -u admin:password
-d '{"credential":{"project_id": "<storage account name>","type":"ec2"}}' -H "Content-Type:
application/json"
```

- `<accesser ip>` is the IP address of an Accesser® Device that belongs to an access pool with the service API enabled. To use an IPv6 address, enclose the Accesser IP in square brackets: `[<accesser ip>]`.
- `<storage account name>` is the name of the storage account the access key is associated with.

As part of the response, the key ID, `key`, and secret key, `secret`, are returned. These credentials must be used by the user to create and modify containers.

Specify the ACL type for the new container
By default, the ACL type is S3-compliant. A user with read permission can only list objects in the new container.

If you want to use the Read-and-List ACL so that users with read permission can list objects as well as read objects from a bucket, set the corresponding **`Object Access`** property in the Manager Web Interface or Manager API. Otherwise, users with `read` permission can only list objects and users with `full_control` permissions must grant individual users read object permission using the IBM Cloud Storage Object API.

Control bucket owner interface operations
Users with `full_control` permissions can create, configure, and delete buckets using the IBM Cloud Object Storage API.

To disable these operations for a user with `full_control` permissions, check the **Create, configure, and delete containers using only the Service APIs** checkbox on the Container Mode Configuration page in the Manager Web Interface or use the Manager REST API.

Create a container

After you create access keys, you can create containers.

Use the access key to determine which storage account to create the container in, as access keys are associated with a single storage account.

Using the Cloud Storage Object API

When the **Create, configure, and delete containers using only the Service APIs** checkbox on the Container Mode Configuration page is not enabled created with Cloud Object Storage **PUT** bucket requests, as shown in the "Create a new bucket" section of the *IBM Cloud Storage Object API Developer Guide.*

Note: In **PUT** bucket requests in Container Mode, the IP to use is the IP address of an Accesser® Device that belongs to an access pool with the service API enabled. To use an IPv6 address, enclose the Accesser IP in square brackets: [<accesser ip>].

The location constraint field should be the provisioning code of the container vault where the container is created. The provisioning code is configured on the create or edit vault page. This parameter is NOT required if a default container vault is configured for the Accesser® Device. The default container vault can be set on the create or edit access pool page.

Using the Service API

A Service Administrator can create containers using the Service API, as shown in the "Create a new bucket" section of the *IBM Cloud Storage Object API Developer Guide*.

Specify container ACL, IP access control, and quota

A System Administrator can use the Service API to specify the allowedIp, deniedIp, or hardQuota for the container, as shown in either “Create a bucket” or “Update a bucket” in the IBM COS Container Configuration Service API Developer Guide .

For more information, see “Service_API->Bucket provision and configuration” section in this document for example to specify ACL, IPaccess control, or hard quota during bucket creation or metadata modification

Enabling Container Mode with an existing system

About this task

The conversion process is only needed if one or more standard vaults exist in the system and there's a desire for them to be container vaults. Before you enable Container Mode and initiating a vault conversion, it is highly recommended that you familiarize yourself with the differences in how the system operates in Container Mode. Moving from a system that uses vault level I/O to one that uses container level I/O is a significant change. It cannot be undone.

Standard vaults are not converted directly to container vaults. Instead, access pools are selected for conversion. When an access pool is selected, all of its vaults are converted from standard to container vaults. If a single vault is deployed to multiple access pools, it is not possible to convert only a subset of the access pools. All of them must be converted. An alternative option is to remove one or more of the access pools from the vault's deployments before you start a conversion.

During the conversion process, the Cloud Object Storage Manager does not allow configuration changes to be made.

I/O to existing vaults do not operate any differently when the conversion process is ongoing. If the vault is being converted to a container vault as a part of the process, any I/O initiated while the vault was a standard vault (before conversion, for example) succeeds normally, even if the vault becomes a container vault before the I/O is completed. After the vault is converted into a container vault, traditional vault level I/O will no longer be possible: container level I/O must be used.

Procedure

1. Create a service vault.

   The service vault is needed for Container Mode. It is used to store metadata that is related to containers, storage accounts, and access keys. It must be created, but needs to be created only one time.
2. Configure the Service API.
3. Specify the ACL behavior for converted containers.

   For more information, see “Access control list” on page 10.
4. Ensure that the bucket owner is correct.
5. Optional: containermodeguide_containermode_restrictowneroperations.dita.
6. Ensure that all vaults that are deployed to the selected access pool satisfy all “Conversion compatibility requirements” on page 28.
7. Configure the Container Mode conversion options:
   - DNS-compliant only containers.
   - Create only container vaults.
   - Whether to transfer IP access control and hard quota to container vault or container.
8. Progressively convert standard vaults into container vaults

What to do next

After an access pool is converted, each of its deployed vaults will be container vaults. New I/O at the vault level is no longer possible on the vaults. Objects must be accessed by using container I/O methods, which require the use of account access keys. To ensure a smooth conversion, you should start the system by using access keys for vault I/O before you convert to Container Mode. Only container vaults can be deployed to the access pool.

Ensuring the correct bucket owner

Ensure the correct owner is assigned to a bucket after a vault is converted to a container.

About this task

In Container Mode, a container must have one bucket owner; other users can be granted to the account with “full_control” permission but not as a bucket owner.

Procedure

1. Click on the Change button in the “Access Control” section of the “Vault Configuration” page in Manager Web Interface.
2. On section to grant user access, identify the user who shall be the owner of the container after converting to Container Mode.
3. When there is no user granted with “Owner” or “Read/write” permission, move one of them into either “Owner” or “Read/Write” permission; otherwise, the vault is not allowed for conversion.
4. When there is no user granted to “Owner” permission, but at least one of the users with “Read/Write” permission, perform one of the following to ensure bucket owner is the one with chronologically granted with the “Owner” or “Read/Write” permission:
   - Move the desired bucket owner to “Owner” permission, or
   - Move the rest of users to “Read”, save, move them back to “Read/Write”, and save.
5. When there are multiple users under the “Owner”, ensure the desired one is the first user chronologically granted with the “Owner” permission.
   - Move other users from “Owner” to “Read/Write”, and save.
   - Move these users back to “Owner”, and save.

6. Record the user access information for the vault for verification after completing Vault Mode conversion.

Storage account credentials
A storage account and its credentials are created for each account with permissions to the vault.

Note: After a successful conversion, any changes (including deletion) that occur to the accounts/credentials on the Cloud Object Storage Manager will not propagate to the corresponding storage accounts/credentials that were created as part of the conversion. Likewise, any changes (including deletion) that occur to the storage accounts/credentials created in the service API as part of the conversion will not propagate to the corresponding accounts/credentials on the Cloud Object Storage Manager.

If multiple conversions take place over time and a storage account was created for an account during a previous conversion, the system will attempt to re-create the storage account and credentials. A conversion will fail if an account to be converted exists but differs (for example, the disabled status is different), or the a credential to be converted exists but differs (for example, the associated storage account is different). See the Conversion failures section below for more information regarding conversions that fail.

Conversion compatibility requirements
Not all vaults can be converted. Several compatibility requirements exist that could prevent a vault, or an access pool, from being compatible for conversion.

System-wide requirements.

- A service vault must exist in the system.

Requirements for each vault.

- At least one account must have **`readWrite`** or **`owner`** permissions assigned.
- Name index must be enabled [1].
- Vault proxy is not allowed.
- Recovery Listing is not allowed [1].
- Locked vault is not allowed [1].
- Delete restrictions not allowed [1].
- Versioning is not allowed [1].
- Being one side of a mirror is not allowed.

Requirements for each vault's storage pool.

- Packed or zone storage must be used [1].

Requirements for each access pool.

- API type must be 'Cloud Object Storage' or 'OpenStack Object Storage'.
- Mirrors cannot be deployed.

Requirements for each account with permission to a selected vault.

- One or more access keys must be created.

If a system does not obey these requirements, several of them can be reconfigured in the Cloud Object Storage Manager. For instance, if a vault proxy is configured on a vault, "fixing" the vault for conversion requires the vault remove the proxy.

[1] No simple fix for this requirement exists. In these cases, no way to convert the standard vault to a container vault exists, at least without first migrating data to a new vault by using the vault migration function.

Note: If a vault contains objects that are written with Simple Object (SOH), those objects cannot be read if the vault is converted to a container vault.

Conversion failures

If a conversion fails, a summary of the failures and any associated recovery actions appears on the Container Mode Configuration page.

Resolve each failure before re-attempting the conversion. The following non-exhaustive list of scenarios can cause conversion failures:

- Write threshold for service vault or vault being converted cannot be met.
- A container exists with the same name as a vault being converted.
- A storage account created from a previous conversion has been disabled when the associated manager account is enabled and is part of a subsequent conversion.
- An access key id to be converted exists in the system under a different storage account.

The service vault or any vault associated to a failed conversion cannot be deleted until a successful conversion (the successful conversion does not need to include the vault to be deleted) is performed. If a conversion has failed and you wish to delete the vault (or the service vault) and do not intend complete a successful conversion or move to Container Mode, a recovery operation must be performed. Contact customer support for more details.

If a conversion fails, any storage accounts/credentials converted will not be deleted even if the storage accounts/credentials are not associated to any subsequent conversions. The service API may be used to manually delete any unwanted storage accounts/credentials.

If a conversion fails, the listing of containers for a storage account may show containers created for vaults that were part of the failed conversion. The list of containers will be accurate once the next successful conversion is triggered. If you do not intend to complete a successful conversion a recovery operation must be performed. Contact customer support for more details.

Mixed operation

After the service vault is created, the Cloud Object Storage Manager changes in two distinct ways.

- Allows standard vaults to be converted to container vaults.
- Allows the creation of container vaults in the system.

New user-created vaults can be standard vaults or container vaults. The one exception to this is that a vault that is created on a storage pool that uses file storage or containing mirrors, must be a standard vault. In other cases, both types of vaults can be created. In these situations, when you create a new vault, the type of vault (standard or container) must be selected. The configuration options on the vault creation page differ if the vault is a container vault versus a standard vault. Standard vaults show these parameters; they cannot be set for container vaults:

- Name index.
- Recovery listing.
- Versioning.
- Delete restriction.
- Authorized users.

If you create a standard vault now, with the desire to convert it into a container vault in the future, take special care when you select any of these options. See the conversion compatibility requirements.

Accounts cannot be given direct access to a container vault in the Cloud Object Storage Manager. It is handled only at the container level. For more information, see the *Service API Manual*.

Progressively convert standard vaults into container
For systems with a large number of standard vaults, the IBM Cloud Object Storage System supports progressive migration and conversion of vaults over time.

About this task

To progressively convert a subset of the vaults deployed to an access pool, a System Administrator must utilize three access pools:

1. An access pool for the standard vaults in Vault Mode
2. An access pool for the conversion of standard vaults to container vaults
3. An access pool for the converted container vaults in Container Mode

Procedure

1. Specify the conversion options.
   a) To transfer the IP access control information from vault to container, check the "Migrate vault access control" option in the "Config Container Mode" Manager Web Interface or set "transferAllowedIpsToContainer" to *true* using "editContainerModeSettings.adm" Manager REST API; **otherwise leave the option unchecked or set the value to false in the API, and the allowedIp remains in the vault.**
   b) To transfer the hard quota from vault to container, check the "Migrate hard quota" option in the "Config Container Mode" Manager Web Interface or set "transferHardQuotaToContainer" to *true* using "editContainerModeSettings.adm" Manager REST API; otherwise leave the option unchecked or set the value to false in the API, and the hard quota remains in the vault
   c) To set restricted operation mode, check the "**Create, configure, and delete containers using only the Service APIs"** checkbox on the Container Mode Configuration page in the Manager Web Interface or "configureContainersViaServiceAPIOnly" to true using "editContainerModeSettings.adm" Manager REST API.
2. Before conversion, optionally follow below step when performing progressive Vault Mode conversion on a subset or all of the vaults, otherwise, go to step 3. The progressive Vault Mode conversion uses AP1 to represent the access pool for vaults staying in the Vault Mode, AP2 for those in the Container Mode, and access pool AP3 to be shifted between the two Vault Mode.
   a. Ensure ap1 has been deployed to the subset of vaults to be converted.
   b. Add access pool ap3 to these vaults according to "Configure Vault" in Manager Web Interface or "editVaultAccess.adm" Manager REST API and pause for a couple of minutes to ensure the client traffic is switched to ap3.
   c. Use the same method to remove ap1 from these vaults; now only ap3 is deployed to the vaults pending conversion.
3. Initiate conversion for a subset or all of vaults converting the access pool.
   a. When using Manager Web Interface, click on "Convert to Container Mode" on the specific access pool in Manager Web Interface.
   b. When using Manager API, execute the "systemContainerModeConfiguration.adm" to with "enable" set to *true* on the specific access pool.
   c. All of the vaults deployed to access pool are converted.
4. Wait for the conversion to complete. Configuration requests to the Cloud Object Storage Manager are permitted again.
   a. During an ongoing vault to container conversion, the Cloud Object Storage Manager rejects all configuration requests. It is to ensure that the new containers have current metadata. The conversion can take several minutes.
   b. If the conversion fails, view the Conversion Failure Report. Resolve any failures and re-attempt the conversion.

c. As result of the Vault Mode conversion, IBM COS

1) Create a container for each vault. The new container's name matches the container vault's name as it is configured in the Cloud Object Storage Manager.
2) Create a Storage Accounts for each account with permissions to a vault. Credentials are created for the storage account by using the access keys that are owned by the account.
3) **Identify bucket owner as the grantee chronologically granted with OWNER or READ/WRITE permission**
4) **Transfer bucket and object ACL from standard vault to container.**
5) Transfer IP access control and hard quota to the destination according to conversion preference.

5. Optional: Following the below steps post the conversion when performing progressive Vault Mode conversion.
   a) Add AP2 to the newly converted container vaults and pause for a couple of minutes to ensure client traffic is switched to AP2.
   b) Remove AP3 from these vault now, AP3 is not deployed to any vault and ready for next set of vaults conversion
6. Optional: Repeat above steps to process additional subsets of vaults.
   a) Optional:
7. Verify conversion results.

   A Service Admin uses Container Mode Service API to verify the result in Container Mode. An end user can use S3 API to verify bucket configuration and perform IO. Below is the expected result.

   - Storage Account Management Service API
     – “Account Listing” command returns the storage accounts for all users authorized to the standard vaults prior conversion.
     – “Specific Account Listing” returns the expected number of objects and usage, and one container shows up for each converted vault (location constraint).
     – “Container Listing” command returns the expected usage and number of objects for the first container for the converted vault.
     – Usage
       - Need to wait about 20 minutes for the usage to be refreshed after conversion.
       - Note: it is expected that the usage in a container is slightly different from that in the standard vault.
   - Credentials Management Service API
     – “List Credentials” command returns the original credential for each account.
   - Container Resource Configuration Service API
     – “Retrieve Container Metadata” returns the bucket name, ACL, the bucket owner as the service instance (or storage account), the vault name as the storage location, and the optional IP Access control and hard quota based conversion decision. Note: There is no value in Cross Origin Region Support (CORS) or retention policy as part of the conversion. If later a bucket enables these features and specifies them using S3 “PUT Bucket CORS” or “PUT bucket protection” commands, then these value can be retrieved using this service API, IBM COS does not support specification on these parameters using service API.
     – Compare the ACL with the record user access information for the original standard vault. The users other than the bucket owner shall be granted explicitly in “acl” with the correct permission.
     – Verify that a new bucket can be created through the “Create a Bucket” service API command under the corresponding container vault and storage account.
     – Verify that a Service User can set and retrieve a container’s IP access control and hard quota using service API “Modify/Retrieve Container Metadata”

- S3 API
  - Verify “Get Account” S3 operation returns only the containers owned by the account.
  - When restricted operation mode is set, verify that an end user cannot create a bucket, delete a bucket, set or retrieve ACL
  - Verify that S3 HEAD bucket operation does not return IP and hard quota information.

What to do next
After an access pool is converted, each of its deployed vaults will be container vaults. New I/O at the vault level is no longer possible on the vaults. Objects must be accessed by using container I/O methods, which require the use of account access keys. To ensure a smooth conversion, you should start the system by using access keys for vault I/O before you convert to Container Mode. Only container vaults can be deployed to the access pool.

Operations

After Container Mode is enabled, the behavior of the system will change slightly. I/O is now run on containers instead of vaults. Also, some manager screens are slightly modified (refer to the following section for details).

Manager operation

After you migrate the system to Container Mode, you can still create, configure, and monitor vaults through the Manager.

Create or edit a vault
The **Create vault** and **Edit vault** pages look slightly different. Certain configuration parameters are no longer visible as they cannot be changed on container vaults.

The following configuration parameters are no longer visible:

- Name index.
- Recovery listing.
- Versioning.
- Delete restriction.
- Authorized users.

Also, a new configuration parameter is added.

Provisioning Code
Used to specify in which container vault containers should be created. During PUT bucket requests with the COS API or Create bucket requests with the Service APIs, the provisioning code of the wanted container vaults should be specified as part of the LocationContraint in a COS API request or StorageLocation in a Service API request. During container vault creation, the provisioning code defaults to the vault name. It can be changed by clicking the provisioning code text box and making the wanted edits.

Create or edit an access pool
The **Create access pool** and **Edit access pool** pages are also modified to include an extra parameter, default container vault.

If a default container vault is specified, the LocationConstraint in the PUT bucket requests becomes optional. If no LocationContraint is specified, the container is created in the device's default container vault.

Also, the service ports can be opened or closed on the create access pool or edit access pool pages.

Configuring Container Mode
The **Configure Container Mode** page can be used to configure whether the container names must be DNS-compliant names. By default, this restriction is not enabled. It is also used to configure whether to

transfer the allowed IP and hard quota from vault to container. By default, they are set to transfer to container. Additionally, it can be used to set restricted operation mode.

Also, Container Mode can be disabled on the **Configure Container Mode** page. However, Container Mode can be disabled only when all container vaults are deleted from the system. Container vault deletion requires assistance from IBM® Customer Support.

Delete a vault

Container vaults cannot be deleted without assistance from IBM Customer Support.

The service vault cannot be deleted until Container Mode is disabled. Container Mode can be disabled when all container vaults are deleted.

Examples

The following examples demonstrate how to manage storage accounts , containers, and AWS credentials.

The examples use cURL for storage account management , container configuration and AWS credentials management. The end user IO requests are no different in Container Mode, as compared to Vault Mode.

Note: To make storage account management , container configuration, or AWS credentials management API requests, the user must have the Service User role. For more information, see the IBM Knowledge Center.

Storage account management

Account Listing

```
curl http://<Accesser IP>:8337/accounts -u user:password
```

Account Creation

```
curl -X PUT http://<Accesser IP>:8337/accounts/<Account ID> -u user:password
```

Account Specific Listing

```
curl http://<Accesser IP>:8337/accounts/<Account ID> -u user:password
```

Account Deletion

```
curl -X DELETE http://<Accesser IP>:8337/accounts/<Account ID> -u user:password
```

Container Listing for Account

```
curl http://<Accesser IP>:8337/accounts/<Account ID>/containers -u user:password
```

AWS credentials management

Credential Creation

```
curl -X POST http://<Accesser IP>:8337/credentials
     -u user:password
     -d '{"credential":{"project_id":"<Account ID>","type":"ec2"}}'
     -H "Content-Type: application/json"
```

Credential Listing

```
curl http://<Accesser IP>:8337/credentials?project_id=<Account ID> -u user:password
```

Credential Specific Listing

```
curl http://<Accesser IP>:8337/credentials/<AWS Key ID> -u user:password
```

Credential Delete

```
curl -X DELETE http://<Accesser IP>:8337/credentials/<AWS Key ID> -u user:password
```

Container provisioning and configuration

In addition to the Cloud Object Storage API method and format to access containers in Container Mode, the system offers a Service API to provision and configure containers.

Create container

Create a container with container vault and storage account information

```
curl -X PUT "http://<Accesser IP>:8337/container/<bucket.name>"
     -u <user name>:<password>
     -d '{"storageLocation":"<vault provisioning code>", "serviceInstance":"<storage
account id>"}'
     -H "Content-Type: application/json"
```

Retrieve container metadata

Returns all container metadata including ACL, IP access control, quota, storage location, storage account, CORS, and retention policies.

```
curl http://<Accesser IP>:8337/container/<bucket.name> -u user:password
```

Modify container metadata

Modify authorized IP addresses

```
curl -X PATCH "http://<accesser ip>:8337/container/<bucket.name>"
     -u <user name>:<password>
     -d '{"firewall":{"allowedIp": ["<IP1>", "<IP2>"]}}'
     -H "Content-Type: application/json"
```

Modify quota

```
curl -X PATCH "http://<accesser ip>:8337/container/<bucket.name>"
     -u <user name>:<password>
     -d '{"hardQuota":"<hard quota bytes>"}'
     -H "Content-Type: application/json"
```

Modify authorization

```
curl -X PATCH "http://<accesser ip>:8337/container/<bucket.name>"
     -u <user name>:<password>
     -d '{"acl":{"user1":["write"]}}'
     -H "Content-Type: application/json"
```

Delete container

```
curl -X DELETE "http://<accesser ip>:8337/container/<bucket.name>" -u user:password
```

Note: The API method and format to access buckets and objects in Container Mode is no different from Vault Mode.

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Homologation statement

This product may not be certified in your country for connection by any means whatsoever to interfaces of public telecommunications networks. Further certification may be required by law prior to making any such connection. Contact an IBM representative or reseller for any questions.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at Copyright and trademark information at www.ibm.com/legal/copytrade.shtml.

Accesser®, Cleversafe®, ClevOS™, Dispersed Storage®, dsNet®, IBM Cloud Object Storage Accesser®, IBM Cloud Object Storage Dedicated™, IBM Cloud Object Storage Insight™, IBM Cloud Object Storage Manager™, IBM Cloud Object Storage Slicestor®, IBM Cloud Object Storage Standard™, IBM Cloud Object

Storage System™, IBM Cloud Object Storage Vault™, SecureSlice™, and Slicestor® are trademarks or registered trademarks of Cleversafe, an IBM Company and/or International Business Machines Corp.

Other product and service names might be trademarks of IBM or other companies.

Printed in USA