

zadara

zStorage NextGen Object Storage

Release 23.09-SP3

Zadara

Mar 27, 2025

GETTING STARTED

- 1 Introduction 1**
 - 1.1 What is Object Storage? 1
 - 1.2 Object Storage vs. Block and File Storage 1
 - 1.3 NextGen Object Storage Components 1
 - 1.4 NextGen Object Storage Hierarchy 5
 - 1.5 Functionality differences between NextGen Object Storage and VPSA Object Storage 6

- 2 First steps 9**
 - 2.1 Register a Zadara Account 9
 - 2.2 Creating a NextGen Object Storage 9
 - 2.3 System notifications 13

- 3 Management interface 15**
 - 3.1 Language localization 15
 - 3.2 Object Storage Administrator view 16
 - 3.3 Account administrator view 16
 - 3.4 Object Storage in a dark-site 16

- 4 Console 19**
 - 4.1 The Object Storage Console Window 19
 - 4.2 Encrypted Containers 19
 - 4.3 Containers pane 20
 - 4.4 Folders/Objects pane 22
 - 4.5 Details Pane 25
 - 4.6 Large objects support 37

- 5 Using Object Storage Clients 39**
 - 5.1 AWS S3 Compatible clients 39
 - 5.2 Openstack Swift Interface 56

- 6 Main Dashboard 63**
 - 6.1 Object Storage administrator dashboard 63
 - 6.2 Account administrator dashboard 64

- 7 Resources Management 67**
 - 7.1 Monitoring Drives 67
 - 7.2 Monitoring Virtual Controllers 68
 - 7.3 Managing Storage Policies 71

- 8 Accounts and Users 77**
 - 8.1 Managing Accounts 77
 - 8.2 Managing Users 82

8.3	Dual Factor Authentication	86
9	Managing Permissions	89
9.1	Understanding Permissions	89
9.2	Setting Account Permissions	90
10	Usage Reports	93
10.1	Usage Reports - Exporting a Summary Report	94
10.2	Usage Reports - Exporting a Detailed Report	94
11	Logs	95
11.1	Access Log	95
11.2	Events Log	95
12	Performance Monitoring	97
12.1	Understanding Performance Monitoring	97
12.2	The Performance Monitor	97
12.3	Customizing the Performance Monitor	98
13	Settings	101
13.1	General & Connectivity settings	101
13.2	Security settings	105
13.3	Pricing settings	106
13.4	Network settings	107
14	Network Diagnostics	109
15	Load Balancing	111
15.1	How load is balanced in the Object Storage?	111
16	Troubleshooting	117
16.1	Management interface access	117

INTRODUCTION

1.1 What is Object Storage?

Object Storage is an alternative way to store, organize and access units of data. It provides a reasonable balance between performance and functionality versus simplicity and scalability. Object Storage enables a minimal set of features: store, retrieve, copy, and delete objects. These basic operations are done via REST APIs that allow programmers to work with the objects. The HTTP interface to Object Storage systems allows fast and easy access to the data for users from anywhere in the world.

1.2 Object Storage vs. Block and File Storage

Object Storage is much more scalable than file storage because it is vastly simpler. Objects are not organized in hierarchical folders, but in a flat organization of containers or buckets. Each object is assigned a unique ID or key. Their keys, regardless of where the objects are stored, retrieve objects. Access is via APIs at the application level, rather than via OS at the file system level. As a result, Object Storage requires less metadata, and less management overhead than file systems. This means Object Storage can be scaled out with almost no limits. Object Storage is easier to use than block storage and overcomes the limitation of fixed size LUNs. It also removes file system limitations such as the folder size or path name length. Unlike block or file, Object Storage does not use RAID for data protection. It simply keeps a number of copies of each object.

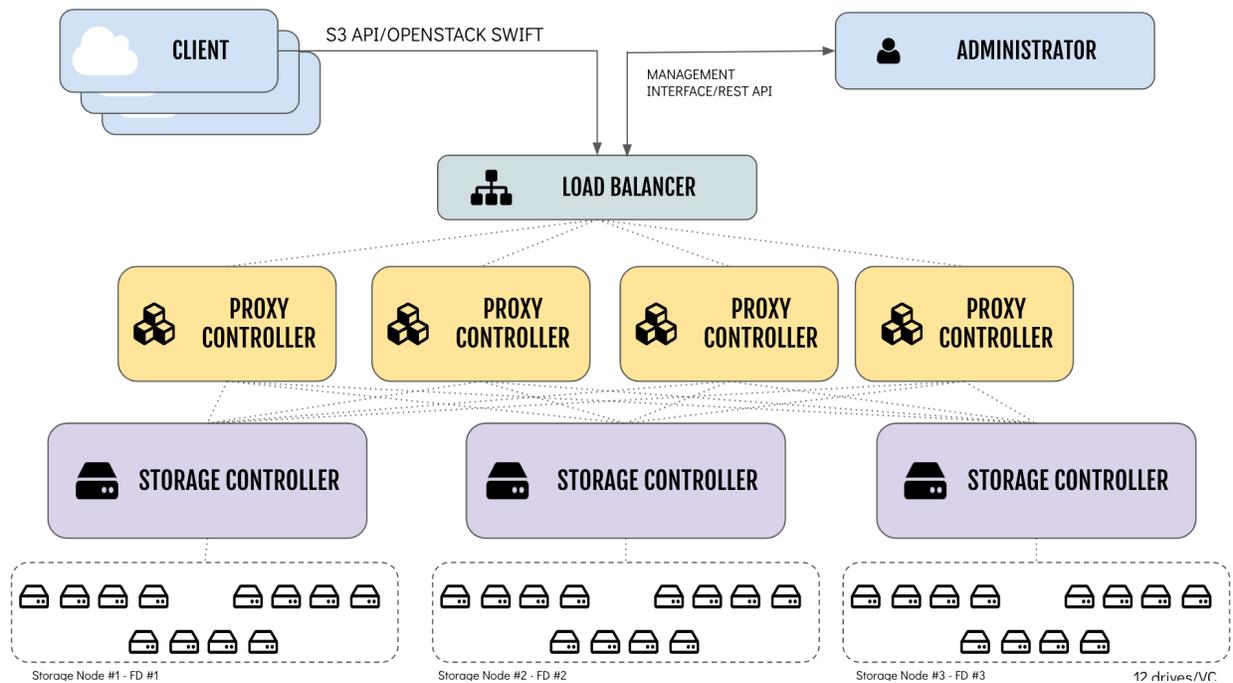
NextGen Object Storage is Zadara's object storage service. It is provided on Zadara clouds, side by side with the VPSA that provides block and file storage services.

A single Zadara Storage Cloud can host several virtual object stores and this makes it truly disruptive and unique, as each NextGen Object Storage has entirely provisioned resources of CPU/RAM/networking/disks & runs the object stack in isolated Virtual Machines (i.e. there is no sharing of resources anywhere across VPSAs) thereby providing complete performance and fault isolation.

1.3 NextGen Object Storage Components

NextGen Object Storage architecture is a scale out cluster of Virtual Controllers that together provides the object storage service.

Capacity & performance can independently scaled up/down by adding/removing disks and proxy-VCs respectively. The Object Storage is provisioned with an internal load-balancer to distribute REST API traffic across the Proxy REST API Layers. The Object Storage is a multi-tenant solution which allows creation of multiple accounts, where each account has its users who can work with the object interface (GET/PUT objects) and the management interface.



1.3.1 Object Storage Fault Domains

In order to ensure the Object Storage survival in case a complete storage node is lost, the data is distributed between Fault Domains. “Object Storage Fault Domains” are manually populated for the cloud Storage Nodes by the cloud admin.

Object Storage VCs are created in “VC-Sets” according to the desired policy protection type (2-way/Erasure Coding protection). Each VC in a Set is created in a different Fault Domain.

Drives are added to the the Object Storage in sets as well. And allocated only to VCs within the same Fault Domain.

1.3.2 The Ring

A ring represents a mapping between the names of entities stored on disk and their physical location. There are separate rings for accounts, containers, and one object ring per storage policy. When any components need to perform any operation on an object, container, or account, they need to interact with the appropriate ring to determine its location in the cluster.

The objects rings are stores in each Policy. The accounts and containers rings are stored in dedicated Policy named Metadata Policy.

One of the Virtual controllers (called Ring Master), runs the Rings, in addition to its other responsibilities. In case of failure of the Ring Master, another VC (called Ring Slave) will take its place.

1.3.3 Virtual Controller

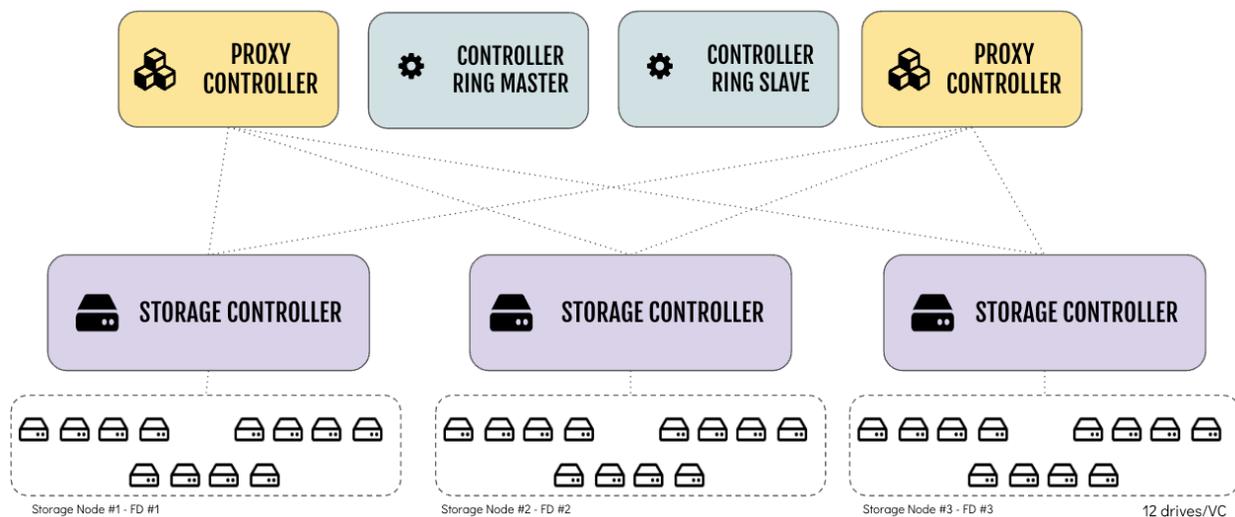
A Virtual Controller (VC) is a Virtual Machine with dedicated CPUs & RAM, which runs the NextGen Object Storage IO stack and control stack. The number of VCs in a configuration is determined by the number of drives assigned, starting with a minimal configuration of 2 VCs, and can grow to hundreds.

With Zadara's NextGen Object Storage, VCs are classified into three groups:

- Cluster controller VC - mainly responsible for object storage administration services and cluster health. The controller VC also exposes the web management interface endpoint along with Zadara's RestAPI server as well as managing authentication services and load-balancing services. Its main responsibilities are:
 - Query Cloud Controller and Storage Nodes for resource assignments and changes.
 - Provide Authentication/Authorization framework with which individual accounts/users can be managed and these account/users being able to work with objects within their account
 - Automatically reconfigure/redistribute object data across available disks on addition/removal of disks, failure/recovery
 - Provide management GUI and REST API to manipulate the system entities and also to work with the object store
 - Provide metering visibility in object request flows, capacity trend utilization
 - Billing based on capacity/throughput usage for each of the tenants
 - Provide internal load balancing service
- Proxy VC - The Proxy layer is the interface to the users or the application using the data objects. Proxy VCs can be added/removed on-the-fly. The proxy exposes both S3 compatible API and Openstack Swift API.
- Storage VC - The storage Layer is responsible for storing the objects on the drives, and updating the metadata in the databases. The physical drives allocated to the NextGen Object Storage instance will be attached to the Storage VC (up-to 12 drives). Storage VCs will be provisioned automatically by the system.

The following diagram describes the structure of Zadara's NextGen Object Storage:

NEXTGEN OBJECT STORAGE – STRUCTURE (EC 4+2, 3FD)



1.3.4 Dedicated Drives

The zStorage Cloud Orchestrator assigns dedicated drives for each each VPSA/Object Storage instance. The drives are provisioned from different Storage Nodes (SNs) for maximum redundancy and performance. Each drive is exposed as a separate iSCSI target from the SN and is LUN masked only to the VPSA's VCs. The instance QoS is guaranteed, because neighbors, with provisioned drives adjacent to yours, cannot access your drives, impact your performance, or compromise your privacy and security.

1.3.5 Object Storage data policies

Zadara Object Storage has two data protection types:

- 2-Way Mirror - with 50% capacity utilization and oriented for performance. For small scale object storage deployments (up to 1 PiB).
- Erasure-Code 4+2 - with 67% capacity utilization oriented for high-durability. For large scale object storage deployments.

The data protection will form the layout of the object storage instance. Each Object Storage instance supports only one data policy.

Important: Data policy type cannot be changed post creation.

Each Data Policy type has a unique set of characteristics in terms of performance, durability and scale, consider the following guidelines when you plan your Object Storage instance.

Display Name	Performance	Recommended Scale	FD	Utilization
2-Way Mirror	High	Up To 1 PiB Of Usable Capacity	2	50%
4+2 Erasure Coding	Moderate	Beyond 1 PiB Of Usable Capacity	3	67%

Important: The availability of Data Protection policies may differ in different Zadara deployment, as it is dependent on the amount of nodes the cloud is structured from.

Minimal drives required for Data Policy creation

Policy type	Drive count Required for creation	Drive count Required for expansion
2 Way Mirror	2	2
Erasure Coding 4+2	6	6

1.4 NextGen Object Storage Hierarchy

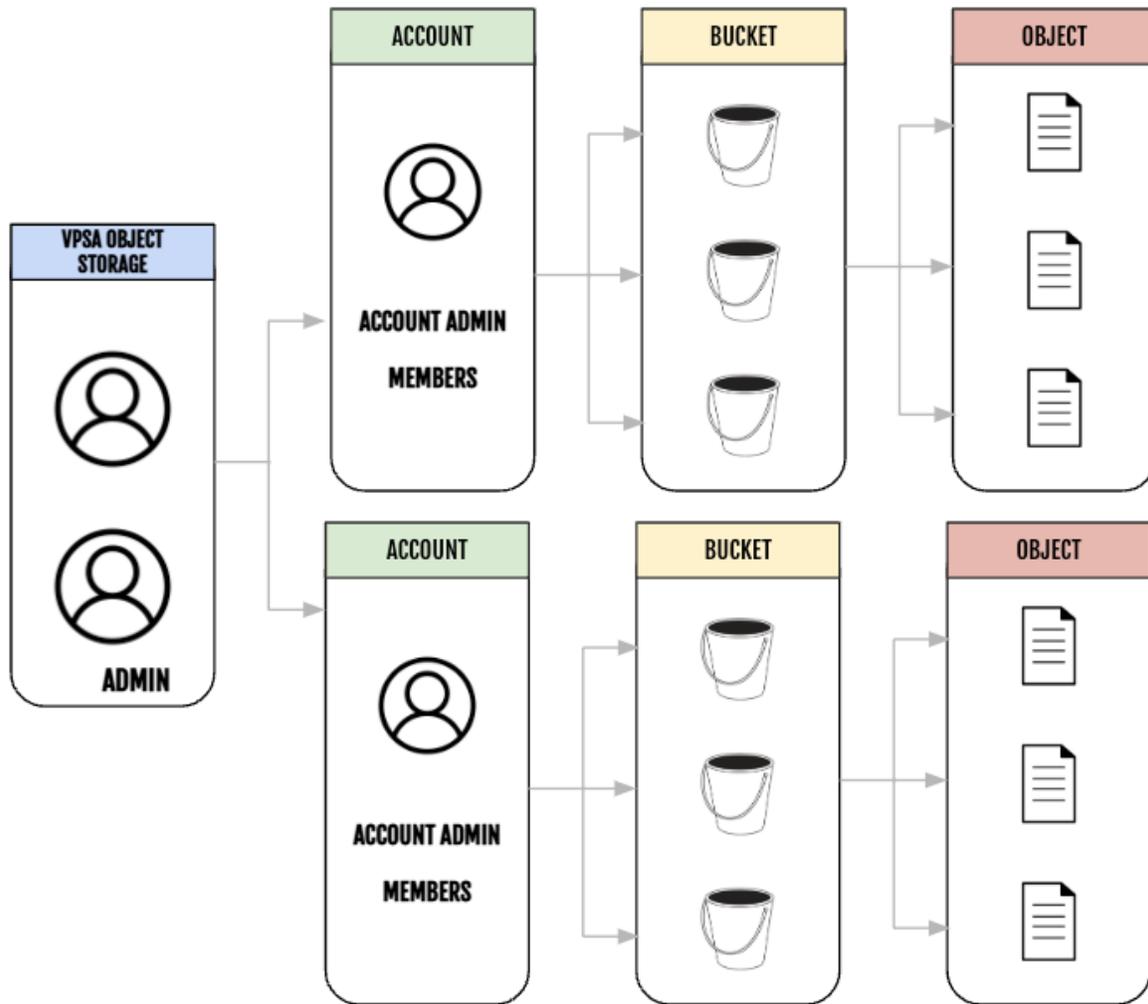
The Object Storage system organizes data in a hierarchy, as follows:

- **Account** (also referred to as Tenant). Represents the top-level of the hierarchy. Usually created by the service provider. The account admin owns all resources in that account. The account defines a namespace for containers. Containers in two different accounts, might have the same name. Accounts are also used to control users access to objects and containers.
- **Container** (also referred to as Bucket). Defines a namespace for objects. Objects in two different containers, may have the same name. Each container can have a unique access rule policy (ACL) or inherit the account default policy.
- **Object** Stores data content, such as documents, images, log files etc.

1.4.1 NextGen Object Storage Users and Roles

There are four types of Roles assigned to NextGen Object Storage Users:

- **Object Storage Administrator** (`zios_admin`) responsible for the administration of the NextGen Object Storage. The user (registered in Zadara Provisioning Portal) that provisions the NextGen Object Storage will be set as its Administrator. By default, the NextGen Object Storage is created with one account (administrator account) and the NextGen Object Storage Administrator is a member of this account. NextGen Object Storage Administrators can add other users with the same role. NextGen Object Storage Administrator is a super-user with privileges to create accounts and users of any role. Users with NextGen Object Storage Administrator role can define policies, add/ remove drives and assign drives to policies. Users with the NextGen Object Storage Administrator role can perform containers and objects operations across accounts. The NextGen Object Storage administrator is also responsible for the NextGen Object Storage settings (like IP addresses, SSL certification, etc.), and has access to the metering and usage information.
- **Object Storage Administrator (zios_admin) - Read Only** a dedicated Read-Only account for cross-accounts monitoring and reporting purposes. The Read-Only role is available for the `zios_admin` account only. A Read-Only user will have access to the NextGen Object Storage RestAPI, however it will not have data access. The user role is designated for monitoring and reporting purposes, such as:
 - Performance monitoring
 - Capacity monitoring
 - Usage reports & billing automation
- **Account Admin** can create an account (using the Self Account Creation Wizard) and can manage their own accounts. They can perform any user management and containers/objects operations.
- **Member** can do object storage operations according to the permission given by the account administrator, within the limits of that account. These operations include create/delete/list containers and create/delete/list objects.



1.5 Functionality differences between NextGen Object Storage and VPSA Object Storage

1.5.1 Data policy protection

VPSA Object Storage	NextGen Object Storage
2-Way Mirroring	2-Way Mirroring
Erasure Coding 2+1, 4+2	Erasure Coding 4+2
Erasure Coding 6+2, 9+3	
Multizone High Availability	

1.5.2 Functionality

The following features are not included in NextGen Object Storage 23.0-SP1:

- Account level rate limit
- Container level replication
- External authentication (Openstack Keystone integration)

The missing features will be added to NextGen Object Storage, based on Zadara's zStorage roadmap.

FIRST STEPS

This chapter contains step-by-step instructions to create a NextGen Object Storage instance and then to configure its storage properties from the Zadara's Provisioning Portal.

2.1 Register a Zadara Account

To register for a new Zadara account, go to <https://manage.zadarastorage.com/register/> and complete the registration form. In case you wish to provision your new Object Storage instance in a private location please use the URL provided by Zadara for the local Provisioning Portal instance.

2.2 Creating a NextGen Object Storage

1. Log on to your **Zadara Provisioning Portal** at <https://manage.zadarastorage.com>, or at your private cloud, using your username/email & password.

Important: It is recommended to enable MFA (Multi-Factor Authentication) in order to add an additional layer of security to your account.

2. Click **Create New Service** button
3. Select the service - **Next Gen Object Storage**
4. Select the location of for deployment by selecting the cloud provider from the dropdown list.
5. Provide a display name and a description for the new Object Storage
6. Select the redundancy level for the storage policy
7. Provide a display name. This name will be used for display purposes in Zadara Provisioning Portal console and in the Object Storage management interface. If you are planning on having multiple NextGen Object Storage configurations, you might want to give it meaningful name.

✓ **Note:** Name should contain only letters, numbers, and underscores, and cannot contain spaces.

8. Provide a description - free text description (i.e. Object Storage - Staging US East).
9. In the step **Capacity and Data Protection** phase, select the following:
 - **Redundancy Level for Default Storage Policy**
 - Erasure-Coding 4+2 (using 3 Fault Domains) (default)

- 2-Way Protection

Please refer to [Object Storage data policies](#) for additional information and requirements regarding Object Storage data policies.

- **Drive Quantities** - select the type and number of Drives that you would like to allocated to your new NextGen Object Storage. The number of drives that can be selected depends on the protection level required.
 - For 2-way protection, an even number of drives must be selected.
 - For Erasure Coding protection the number of drives must be in multiple of 6 drives (i.e. 6,12,18,24 etc.)

10. In the **Confirmation** step, you can review the summary for your new service creation. To change anything, click **Back** and return to the desires step and correct as needed. Once you are ready click **Create**
11. The requested Object Storage, will appear in the “Awaiting Approval” list until approved by a Zadara Cloud admin.
12. Once approved, the new system only takes a few minutes to launch. During that time the Object Storage status will be changes to “Launching”.
13. You’ll receive an email with a temporary password to the registered email address once the object storage is ready for use.
14. The NextGen Object Storage web management interface is accessible using the Management URL displayed in the portal.

✓ **Note:** By default, the object storage interfaces are accessible to the storage front-end network only. If you wish to access it using a public IP please refer to the [Assigning Public IPs](#) section in this guide.

15. Use your registered username or email address, and the temporary password, to enter the management interface. You will be immediately prompted to set a new password for your NextGen Object Storage Admin account.

✓ **Note:** It is recommended to enable MFA (Multi-Factor Authentication) in order to add an additional layer of security to your account.

Congratulations! You have successfully created and provisioned a new NextGen Object Storage.

The newly created NextGen Object Storage already has a single account named “zios_admin”, and you are the only user defined “admin” with the role of “NextGen Object Storage Admin”. You are now able to start using your your NextGen Object Storage - create containers and start uploading objects, or create additional accounts and users for others to use.

Please take a look at the [Management interface](#) section to familiarize yourself with the object storage administrator capabilities.

Except provisioning Zadara’s Provisioning Portal allows you to perform multiple management operations on an existing object storage service, such as:

1. Change your instance display name and description
2. Add capacity (drives) to the data policy
3. Assign/detach a Public IP to allow external connectivity
4. Request an additional Proxy Virtual Controller (VC)
5. Add a virtual network interface
6. Hibernate the instance (shutdown data services)
7. Delete your object storage

2.2.1 Adding drives

To add drives to your NextGen Object Storage, open the Zadara Provisioning Portal, select the NextGen Object Storage, click on the **Actions** button and then press the **Add Drives**.

- Select the number of Drives of the relevant type you wish to add to your NextGen Object Storage, and press **Add**. The number of drives added to the “Storage Policy” should match its characteristics, as described in the previous section of this guide.
- This operation requires the approval of a Zadara Storage Cloud Admin. Once approved, you’ll see the number of Drives in the Provisioning portal updated according to the request. The new drives will be automatically assigned to your object storage’s data policy.

2.2.2 Assigning Public IPs

For security and privacy reasons, by default you cannot access the Object Storage from a public network (i.e. internet). The Front-End IP address, used for management (via GUI and REST API) and for data IO workload (S3/Swift API), is allocated on the Zadara Storage Cloud “Front-End” network 10/40/100 GbE which is routable only from the Cloud Servers network. As this is an internal IP address, servers outside of your cloud network will not be able to reach this IP address. This means you cannot access your Object Storage from the Internet or any network with no routing to the Front-End network.

To assign a Public IP address to your Object Storage, for internet inbound connectivity, open the Provisioning Portal, select the Object Storage, and click the **Actions** button and then the **Allocate Public IP** option. This operation requires Zadara cloud admin approval. Once approved, the IP address will be added to the NextGen Object Storage characteristics. And In the NextGen Object Storage web management interface, under **Settings > General > Public IP**

To remove it, simply click the **Deallocate Public IP**

2.2.3 Adding Proxy Virtual Controllers

The object Storage REST API is exposed through the Proxy virtual controllers. For each request, it will look up the location of the account, container, or object and route the request accordingly. Failures are also handled in the Proxy. For example, if an object server is unavailable for an object PUT request, it will find an alternate route there instead.

On top of the Proxy VCs that are provided out of the box, it is possible to add additional Proxy VC in order to improve performance.

 **Note:** Please note that the inclusion of additional controllers will result in additional costs

To assign additional Proxy VC’s, go to the Zadara Provisioning Portal, select the NextGen Object Storage system, and press the Add Proxy Virtual Controllers button. Similarly to adding drives, this operation requires the approval of a Zadara cloud admin.

Upon approval, the Proxy VC will be added automatically to the array and will start handling clients workloads.

In case the additional proxies are no longer needed - it can be removed directly and immediately (without administrator approval) from the Object Storage management interface by the Object Storage administrator or from the Object Storage management RestAPI.

2.2.4 Add a virtual network interface

The object storage can support up to 10 additional network interfaces (on subnets/VLANs) in order to allow client connectivity coming from different networks for with different use-case or network topology.

The system will create a fully-qualified domain name matches to the new interface dynamically that matches the default domain name in order to allow seamless connectivity to the object storage endpoint.

The operation of adding an additional network interface can be handled directly from the provisioning portal and so as network management (VLAN allocation and network creation)

✓ **Note:** The network interfaces assigned to the object storage instance must be on different subnet/VLAN (i.e. network 192.168.0.0/24 cannot be used in two different interfaces regardless of the network VLAN)

2.2.5 Hibernate

Zadara cloud allows you to “hibernate” your object storage instance and by doing so you are gracefully shutting down its data services and management interfaces. It is extremely helpful in cases a complete hardware maintenance is underway like complete site relocation, networking gear replacement where all appliances needs to be replaced in parallel etc.

Important: There’s no need to hibernate your instance during standard Zadara’s software upgrades or expansions, the system is designed for seamless upgrades/maintenance.

Unlike the VPSA Storage Array/Flash Array, the Virtual Controllers will not be removed once the instance is hibernated. There will be no impact on billing once the object storage is hibernated.

2.2.6 Deleting your Object Storage

 **Warning:** Please note that deleting the Object Storage is a significant action and should be done with caution. Ensure that you have backed up any critical data and that you are certain you want to proceed with the deletion

The Object Storage owner can delete their instance using Zadara’s Provisioning Portal in case it is no longer needed. The delete operation will delete all underlying entities such as data-policy,accounts,users,containers and objects.

In order to delete the Object Storage select the **Delete** action from the **Actions** dropdown menu in Zadara’s Provisioning Portal. The delete operation will require an additional authentication confirmation.

Due to the sensitive nature of the delete operation, when the owner initiates a delete operation, it will generate a delete request that necessitates approval from a cloud administrator. Once approved, the Object Storage will be permanently deleted.

✓ **Note:** While the delete request is still pending, the Object Storage will continue to operate normally and will continue to report its consumption to the cloud’s billing services

2.3 System notifications

Object Storage notifications, both informational and critical, necessitating user action, will be communicated via email to the service owner and other designated users set to receive notifications.

System notifications are categorized based on the following priorities:

- Urgent
- High
- Normal
- Low

2.3.1 Urgent priority notification

An alert that requires an immediate Object Storage administrator action to ensure the storage service health or to restore the Object Storage to normal operation, for example:

- Data Policy free capacity state
- System is pending for Master Encryption key from the administrator

2.3.2 High priority notification

An alert that requires awareness of the Object Storage administrator to ensure the storage service health or other service issues that are currently being handled by Zadara's support team. In some cases VPSA administrator action is required.

Example for high priority notifications:

- Custom (customer provided) TLS certificate expired
- Duplicate Front-End IP discovery
- Account reached it's quota usage

2.3.3 Normal priority notification

An alert that requires the administrator's attention, yet does not necessarily have an immediate impact on the service. For example:

- New user account creation

2.3.4 Low priority notification

A low-priority message with no impact on service health. While the message may require attention or action from administrators, its lower priority status indicates that it poses minimal or no threat to the current state of the service and can be addressed at the convenience of the administrators without causing service disruptions.

MANAGEMENT INTERFACE

Zadara's object storage exposes a rich, user friendly, role based management interface.

The web management interface changes according to the context of the user that logs in (role based). The user's role determines the actions available for each specific user.

The following section will describe the management user interface for the following roles:

1. Object Storage Administrator (zios_admin) - labeled Object Storage Administrator throughout this user guide
2. Account administrator - labeled Account Administrator throughout this user guide
3. Account Member - labeled Account Member throughout this user guide

Important:

- Zadara's web applications allow only TLS 1.2 and higher, which is the recommended TLS level by industry standards. The TLS (Transport Layer Security) protocol secures transmission of data over the internet using standard encryption technology.
 - The VPSA management interface web application is supported in all modern browsers. We recommend using Google Chrome, Firefox or Microsoft Edge for an optimal user experience.
-

3.1 Language localization

The object storage management user interface is available in the following languages (you can use the top drop down to change the displayed language):

- English
- Japanese
- Korean
- Deutsche
- Portuguese

3.2 Object Storage Administrator view

The web management interface as seen by the NextGen Object Storage (**zios_admin**) account users includes the following:

- Dashboard - a “snapshot” of the Object Storage instance with a high-level overview of all the instance building blocks. The administrator can review the ongoing performance trends among capacity utilization.
- Resources (Drives, Policies, Reports, Object Storage Console)
- System -
 - Usage reports
 - Performance
 - Settings
 - Diagnostics
- Accounts Management (Accounts, Users, Roles, Requests)
- Logs (Access Log, Event Log)

3.3 Account administrator view

The web management interface as seen by the **Account admin** account users includes the following views:

- Object Storage Console - In this view, you can create/delete and view containers and folders, and perform other functions, to help organize and manage the storage objects. For more information, see [The Object Storage Console Window](#).
- Account Management - In this view, you can view/configure account properties, permissions, and storage usage, and see lists of users associated with the account. For more information, see [Managing Accounts](#).
- Users Management - In this view, you can create/delete/enable/disable users. You can reset user passwords and see their usage statistics. For more information, see [Managing Users](#).

The **Account Member** user will have the Object Storage Console view only.

3.4 Object Storage in a dark-site

The Object Storage can be created in a dark-site where Internet connectivity is not available.

The Object Storage includes two main interfaces:

- Management console (available over port 8443)
- S3/Openstack Swift API endpoint (available over port 443) & Openstack Keystone authentication endpoint (available over port 5000)

Both are provisioned with Zadara’s default TLS certificate (zadarazios.com domain name) to allow proper end-to-end TLS client connectivity).

In an isolated environment, there is no automatic DNS registration of the zadarazios.com alias, hence the certificate will not match the FQDN of the object Storage when opening the management console or trying to connect using object storage client to the s3 api endpoint directly.

There are two approaches to adjust the Object Storage within an isolated environment:

1. Import a custom domain certificate to the object storage (recommended). additional information about using custom certificates can be found in the settings section of this user-guide.
2. Use the Object Storage with IP only

CONSOLE

Scope: Object Storage Administrator Account Administrator Account Member

The Object Storage Console provides management access for Object Storage accounts. It is not a tool for read/write operations from or to the Object Storage. Using the console, you can create, delete and view containers, and list their contents. Within a container, you can create and delete folders to better organize the objects, and set permissions and other management configurations.

4.1 The Object Storage Console Window

The Console Window comprises:

1. [Containers pane](#) - lists the containers in the Object Storage.
2. [Folders/Objects pane](#) - Note that the view shows either folders or objects. Click on a container to display the content folders. Click on a folder to display the content objects. To return from object view to folder view, or in the case of nested folders, to move back within the folder hierarchy click on .. above the object pane.
3. [Details Pane](#) - shows different properties and permissions depending on whether a container, folder, or object is selected in the top pane.

4.2 Encrypted Containers

Encryption management of Data-at-Rest (data on the disk drives) is applied by the Object Storage on a per-container basis. Encrypted and unencrypted containers can coexist in the same account.

An Object Storage generates a random 256-bit unique Encryption Key per encrypted container and uses the Advanced Encryption Standard (AES) to encrypt and decrypt the objects' data.

The Encryption Keys are stored on disk as ciphertext, using AES with a 256-bit Master Encryption Key, which is generated from a user-supplied **Master Encryption Password**.

The user owns the Master Encryption Password. It is never stored on any persistent media. Instead, only its SHA3 hash-sum is saved on disk for password validation.



Caution: Since the system does not keep the Master Encryption Password, you are **fully responsible to retain and protect the Master Encryption Password**.

During Object Storage operation, the Master Encryption Password itself is held in kernel memory of the Object Storage. Core-dumping any User Mode process within the Object Storage will not reveal the Master Encryption Key.

This method ensures that encrypted Data-at-Rest cannot be accessed without explicitly knowing the user-supplied Master Encryption Password, thus providing you full protection if you enable Data-at-Rest Volume encryption.

 **Caution:** The encryption attribute of a container cannot be changed! If you'd like to encrypt the objects of a non-encrypted container, or vice versa, you will need to create a new container and copy the data.

4.2.1 Setting Encryption Password

Scope: Object Storage Administrator

To create a Master Encryption Password, go to the **Settings** page, **Security** tab and click **Edit** in the **Encryption** section. Read the instructions and warning. Enter your password and click **Save**.

Store your Master Encryption Password in a secure place.

4.3 Containers pane

4.3.1 Search for Containers

In the Console screen, you can search and filter the list of containers in the Containers pane to find a specific container or a list of containers. The search locates the container names that match a user-provided case-sensitive prefix.

1. In the navigation tree, select **Resources > Console**.

The Console screen displays.

2. In the **Search Containers** search box, in the toolbar above the Containers pane, enter the leading characters of the container that you want to find.

As each character is entered, the containers list filters to display container names that match the entered prefix.

Important:

- The search is case-sensitive.
 - Wildcards are not supported.
 - The search filter stays active until manually cleared.
-

To clear the search filter and redisplay the full list, click **X** in the search box's right edge.

Sorting the Containers list

In the Console screen's Containers pane, you can sort the list of containers to display in ascending or descending order, by clicking the **Container Name** column header.

✓ Note:

- By default, containers are listed in ascending order.
 - When containers are listed in descending order, a down-arrow displays on the column header.
-

4.3.2 Create Containers

To create a new container in the account, open the console, and click **Add** in the toolbar above the console pane. The **Create Container** dialog will open.

Enter the following information:

- **Name** - Enter the container name.

✓ **Note:** A container name must comprise only lowercase letters, numbers, periods and dashes.

The creation wizard verifies the proposed container name. A warning message is displayed if a non-S3-compatible name is chosen. This restriction can be overridden by checking the **Override S3 naming rules** option.

The Swift API is less restrictive; a container name can start with any character and contain any pattern, except for the slash (/) character.

It is highly recommended to align with the S3 naming rules to avoid S3 client compatibility issues.

- **Storage Policy** - the target storage policy for the container creation
- **Encrypted** - select if the container should be encrypted.
- **Object Lock** - selecting object lock will prevent the deletion or modification of any object prior to its retention period expiration. For more information on Object Lock, see [Object Lock Containers](#).
- **Container logging** - selecting container logging generates audit logs for all object operations.
 - **Target Container** - from the dropdown, select the target container for storing the audit logs.

Important: The target container requires write permissions.

- **Apply logging prefix** - define a prefix to simplify locating and identifying log objects.

For more information, see [Container logging](#).

Click **Create**. The new container will be displayed in the Containers pane.

4.3.3 Delete Containers

To remove a container, open the console, go to the containers pane, select the container to be deleted and click **Delete**. The system will prompt you for deletion confirmation.

Delete a non-empty container

1. Object-Lock enabled container which contains objects cannot be deleted from the console. In order to delete it all contents (including object versions) should be deleted. Empty container can be deleted from the console.
2. Container without object-lock enabled can be deleted directly from the console, once the operation is confirmed by the user, the system will empty the container's data and will delete the container itself.

✓ **Note:** The delete operation of a container may take a while to complete. During the delete operation all new writes to the container will fail. Once the operation completes, the container will no longer be listed in the console.

 **Caution:** After deletion confirmation, the container **with all its content** will be deleted. This operation is permanent and the data can not be recovered.

4.4 Folders/Objects pane

By definition, containers are flat and there is no hierarchy structure for storing the objects. However, since many users are accustomed to the folders tree concept of file systems, the Object Storage Console gives you an option to simulate a hierarchical structure within the Object Storage Containers.

4.4.1 Search for Folders or Objects

In the Console screen's Objects pane for a selected container, you can search and filter the currently displayed folder tree branch level to find folders or objects with names that match a user-provided case-sensitive prefix.

1. In the navigation tree, select **Resources > Console**.

The Console screen displays.

2. In the Containers pane, select a container.

Optionally, use the [Search for Containers](#) function to locate a container.

The selected container's list of top-level folders and objects displays in the Objects pane.

3. In the **Search Objects** search bar, in the toolbar above the Objects pane, enter the leading characters of the folders or objects that you want to find.

As each character is entered, the folders and objects list filters to display folder and object names that match the entered prefix.

Important:

- The search is case-sensitive.
- Wildcards are not supported.
- The search filter stays active until manually cleared.

When navigating to a different folder, the search filter is reapplied in the new current folder.

To clear the search filter and redisplay the full list, click **X** in the search box's right edge.

Sorting a Container Folder's content list

In the Console screen's Objects pane, you can sort the current folder's contents list to display in ascending or descending order, by clicking the respective **Container Name**, **Size** or **Last Modified** column header.

 **Note:** By default, folders are listed in **Name** ascending order.

- Sorting is automatically disabled for folders containing more than 1000 objects.
-

4.4.2 Adding folders

To create a folder, open the console, select a container in the containers pane, navigate to the hierarchy level where you want to create the new folder, and click **Add Folder**. Give it a name and click **Submit**.

Navigation within the container's folders tree is done in a way similar to the common user experience of file systems explorer. By double clicking a folder, you enter it and see its content (objects and sub-folders). By double clicking the .. at the top of the objects pane, you navigate one level up to the parent Folder. The path indicator above the objects pane always show you current position in the tree.

4.4.3 Uploading Folders or Objects

You can upload an entire folder or individual files to a container or to a folder in a container.

✓ **Note:**

- Empty folders are ignored and are not uploaded.
 - A file can be up to a maximum size of 2GB.
 - A folder can contain a maximum of 10,000 files.
-

To upload a folder or file:

1. In the navigation tree, select **Resources > Console**.

The Console screen displays.

2. In the Containers pane, select a container.

Optionally, use the [Search for Containers](#) function to locate a container.

The selected container's list of top-level folders and objects displays in the Objects pane.

If a folder or files should be uploaded below another folder in the container, navigate to the container's destination folder.

3. In the toolbar above the Objects pane, click **Upload**, and then **Folder** or **File**, depending on the scope of the upload.

An explorer window opens.

✓ **Note:** If **Folder** was selected, the explorer filters to display folders only.

1. In the explorer window, navigate to the folder or files to upload.
 - **Folder:** Click the folder or enter its name in the **Folder** field.
 - **File:** Selected a file or files, so that their names are listed in the **File name** field.

✓ **Note:** A maximum of 10,000 files can be selected for upload to a folder.

A folder can contain a maximum of 10,000 files.

Click **Upload**.

The explorer window closes.

An upload status window opens, displaying the uploading progress.

✓ **Note:** Uploads of folders and files to multiple destination containers can take place concurrently, by repeating the upload procedure while another upload is in progress.

Important: While an upload is in progress, keep its browser tab open. Closing or refreshing an upload's browser tab will abort the upload.

4.4.4 Downloading Objects

The Console supports single file download only.

To download a file:

1. In the Containers pane, select the container containing the file.

Optionally, use the [Search for Containers](#) function to locate a container.

The selected container's list of top-level folders and objects displays in the Objects pane.

2. In the Objects pane, select the file to download.

Optionally, use the [Search for Folders or Objects](#) function to locate a folder or file in the current folder.

If the file that should be downloaded is below another folder in the container, navigate to that folder and select the file.

3. In the toolbar above the Objects pane, click **Download**.

✓ **Note:** If the container is a [Versioned Container](#), refer to [Downloading a Versioned Object](#).

A download status window opens, displaying the downloading progress.

✓ **Note:** Downloads of multiple files can take place concurrently, by repeating the download procedure while another download is in progress.

Important: While a download is in progress, keep its browser tab open. Closing or refreshing a download's browser tab will abort the upload.

4.4.5 Removing folders

To remove a folder, navigate to its parent folder, select the folder to be removed and click **Delete** from the toolbar above the folder pane.

After confirmation, the folder with all its content will be deleted.

4.5 Details Pane

The details pane at the bottom of the console screen includes the following tabs:

- **Properties** - displays read only properties of selected container, folder, or object.
- **HTTP Headers** - display, edit, add, or delete the HTTP headers used in the object storage operations.
- **Permissions** - for information on assigning permissions to containers, see [Setting Container Permissions](#).
- **Quotas** - quotas allow placement of limits on size (in GiB) per container (**Console** view) or per account (**Account** view). For more information on quotas, see [Account Quota Management](#).
- **Versioning** - if enabled, versioning supports storing multiple versions of an object in the same container thus allowing recovery from unintended actions or failures. Note that once versioning is enabled for a container it cannot be disabled.
- **Versions** - displays versions for all folders and objects within a container. Note this tab is only available for containers for which versioning is enabled.
- **Container Logging** - display, enable, edit or disable container logging settings. If container logging is enabled, audit logs are generated in the selected **Target Container** for all object operations. **Apply logging prefix** to simplify locating and identifying log objects. For more information, see [Create Containers](#) and [Container logging](#).
- **Object Lifecycle Rules** Version: 23.09
 Display, configure and manage the rules of a container's Object Lifecycle Policy. For more information, see [Object Lifecycle Policy](#).
- **Event Log** - displays log of events related to selected container.

4.5.1 Container Quota Management

Scope: Account Administrator

Quotas are a useful way to control capacity consumption on a specific account or container.

For more information, see [Account Quota Management](#).

1. Navigate to **Console**.
2. In the top pane select the desired container, and open the **Quotas** tab in the bottom details pane.
3. Mark the **Enable capacity quota** checkbox.
4. Enter the **Capacity (GiB)** quota. The minimum is 1 GiB.

✓ **Note:** The sum of actual usage capacities of all the containers in an account are tracked, so that cumulatively they do not exceed the account's quota.

For purposes such as future planning, it is also possible to specify container quotas such that their sum or even an individual container's quota can be higher than the account quota. Although it is possible to specify higher quotas at container level, the system will prevent consumption of extra storage when the account quota has been reached.

5. Click **Update**.

✓ **Note:** When the quota is enabled, the actual **Used capacity (GiB)** also displays in the same tab.

See [Account Administrator Quota Alerts](#) to configure the system to issue alert notifications to the Account Administrator when the quota's warning, emergency and 100% utilization thresholds are reached.

4.5.2 Versioned Container

Zadara's Object Storage supports container versioning which is the ability to maintain multiple versions of an object instance in the same container.

Object Storage PUT operation which normally would replace (overwrite) an existing object would yield an additional a new object version. The versioning mechanism intends to protect against unintended deletion.

Versioning is a pre-requisite for enabling [Object Lock Containers](#).

"Versioned" container would add a version ID attribute to each object and a "Latest" flag to the most recent object's version which is retrieved by default.

S3 API allows the user to list, get or delete a specific version while using its version ID.

Enable Versioning

To enable versioning on a container:

1. Locate the container in the Containers pane, and select it.
2. In the Details pane, select the container's Versioning tab.
3. Activate the **Enable container versioning** toggle and click **Save**.

Downloading a Versioned Object

The Console supports single file download only.

To download a versioned file:

1. In the Containers pane, select a container.
Optionally, use the [Search for Containers](#) function to locate a container.
The selected container's list of top-level folders and objects displays in the Objects pane.
2. In the Objects pane, select the file to download.
Optionally, use the [Search for Folders or Objects](#) function to locate a folder or file in the current folder.
If the file that should be downloaded is below another folder in the container, navigate to that folder and select the file.
3. In the selected file's Details pane, select the **Versions** tab.
All of the versions of the selected file display, including deleted versions.
4. In the **Versions** tab, select a specific version of the file to download.
5. In the Details pane toolbar, click **Download**.
A download status window opens, displaying the downloading progress.

✓ **Note:** Downloads of multiple files can take place concurrently, by repeating the download procedure while another download is in progress.

Important: While an download is in progress, keep its browser tab open. Closing or refreshing a download's browser tab will abort the upload.

Deleting an object

Deleting an object in a container with versioning enabled would yield a new 0-byte version of the same object with a DeleteMarker tombstone. The DeleteMarker version will list the object as deleted however, all previous versions are retained.

In the following example, we have a container with two versions of the same object:

```
$ aws s3api list-object-versions --endpoint-url=$ENDPOINT --profile=$PROFILE \
  --bucket $BUCKET --output yaml
```

```
Versions:
- ETag: '"49e5c77426e2e3f5b635f7965f0020e3"'
  IsLatest: true
  Key: my-data.csv
  LastModified: '2023-04-11T21:07:45.053000+00:00'
  Owner:
    DisplayName: <truncated>
    ID: <truncated>
  Size: 22032
  StorageClass: STANDARD
  VersionId: '1681247265.05321'
- ETag: '"49e5c77426e2e3f5b635f7965f0020e3"'
  IsLatest: false
  Key: my-data.csv
  LastModified: '2023-04-11T21:05:02.306000+00:00'
  Owner:
    DisplayName: <truncated>
    ID: <truncated>
  Size: 22032
  StorageClass: STANDARD
  VersionId: '1681247102.30656'
```

Deleting the object without specifying a version ID would remove the object from the bucket listing while retaining the existing versions.

Delete the object:

```
$ aws s3api delete-object --endpoint-url=$ENDPOINT --profile=$PROFILE --bucket $BUCKET --key my-data.csv
```

Listing the container's object would output an empty list.

```
$ aws s3api list-objects --endpoint-url=$ENDPOINT --profile=$PROFILE --bucket $BUCKET --output yaml
```

While listing the object's versions:

```
$ aws s3api list-object-versions --endpoint-url=$ENDPOINT --profile=$PROFILE --bucket $BUCKET --output \
  ↪yaml
```

```
DeleteMarkers:
- IsLatest: true
```

(continues on next page)

(continued from previous page)

```

Key: my-data.csv
LastModified: '2023-04-11T21:14:41.045000+00:00'
Owner:
  DisplayName: <truncated>
  ID: <truncated>
VersionId: '1681247681.04512'
Versions:
- ETag: '"49e5c77426e2e3f5b635f7965f0020e3"'
  IsLatest: false
  Key: my-data.csv
  LastModified: '2023-04-11T21:07:45.053000+00:00'
  Owner:
    DisplayName: <truncated>
    ID: <truncated>
  Size: 22032
  StorageClass: STANDARD
  VersionId: '1681247265.05321'
- ETag: '"49e5c77426e2e3f5b635f7965f0020e3"'
  IsLatest: false
  Key: my-data.csv
  LastModified: '2023-04-11T21:05:02.306000+00:00'
  Owner:
    DisplayName: <truncated>
    ID: <truncated>
  Size: 22032
  StorageClass: STANDARD
  VersionId: '1681247102.30656'

```

4.5.3 Object Lock Containers

Zadara Object Storage Immutability ensures data integrity by stopping stored objects from being deleted or overwritten during a specific retention timeframe. When an object is locked in compliance mode, its retention mode can't be changed, and its retention period can't be shortened. Immutability ensures object version integrity and availability throughout the defined retention period.

This feature can be leveraged directly from the S3 Compatible backup software (i.e. Veeam Backup and Replication) to ensure the integrity and availability of the backup as required. A configuration guide for Veeam Backup & Replication can be found in [Zadara's Knowledge-Base portal](#).

S3 Object Lock

The NextGen Object Storage uses the S3 Object Lock feature (Compliance Mode) in order to set a retention period to a given object and mark it as an immutable object. Deleting an object will be blocked until its retention period has expired. Object Lock should be enabled when creating a new container, directly from the management interface or by using AWS S3 Tools (CLI/SDK).

✓ Note:

- Retention ensures that an object is WORM protected (cannot be deleted or overwritten) for a defined period of time that can be extended. It can be expressed in seconds, days or years.
- Object Lock Retention is set at the object level. Retention period on an object version can be set either explicitly on the object level, or through a container default setting. Configuring a default container setting is supported via an S3 compatible client. A default lock configuration set at the container level does not apply retroactively to versions

of objects created earlier. It only applies to objects that are created from the time that the configuration is applied. When a retention period is applied to an object version explicitly, you specify a Retain Until Date for the object version. The Retain Until Date setting is stored in the object version's metadata and ensures the object version's integrity until the retention period expires. A retention period can be extended by submitting a new lock request (put-object-retention).

- Zadara's Object Storage supports Object Lock Compliance mode, which is restrictive. It cannot be undone within the retention period. As a result, no user including the object storage administrator user, will be able to delete objects during their retention period.
 - Although the system's s3api allows setting Object Lock Governance mode, it assumes the same restrictions as Compliance mode.
 - Object Lock cannot be enabled for existing containers.
-

Enable Object Lock from the management interface

Object Lock can be enabled for a new container during its creation. In order to create a new container with Object Lock:

1. Log in to the management interface.
2. Navigate to the Object Storage Console section.
3. In the upper options menu, click on the Add button.
4. Enter a new container name.
5. Check the "Object Lock" option.
6. Clicking Create to create the new container.

On creation, the Versioning feature will be automatically enabled for the new container.

✔ **Note:** Automatic enabling of versioning for the new container could lead to additional storage consumption. Object Lock will prevent the deletion or modification of any object prior to its retention period expiry.

A container's Object Lock property set to "true" identifies the container as being Object Lock enabled.

Enable Object Lock using the AWS S3 CLI

In the following examples, Object Lock is enabled using AWS Tools for PowerShell.

Currently, Object Lock can be enabled and reviewed only from the NextGen Object Storage S3 API interface.

✔ **Note:** The following examples use PowerShell syntax. Equivalent API calls will achieve the same results using the language of your choice.

Enabling Object Lock

Object Lock should be enabled on the container level at creation time. Object versioning will be enabled automatically.

Make sure that the Object Storage credentials are set.

Define the NextGen Object Storage as an endpoint:

```
$ENDPOINT="https://vsa-0000000b-zadara-qa13.zadara.com"
```

Container creation

```
$BUCKET="immutable-container"
aws s3api --endpoint-url=$ENDPOINT create-bucket --bucket $BUCKET --object-lock-enabled-for-bucket
```

The expected result should be:

```
{
  "Location": "/immutable-container"
}
```

Confirm that Object Lock was enabled for the newly created container

```
aws s3api --endpoint-url=$ENDPOINT get-object-lock-configuration --bucket $BUCKET
```

The expected result should be:

```
{
  "ObjectLockConfiguration": {
    "ObjectLockEnabled": "Enabled"
  }
}
```

Upload a new object

```
$OBJECT="new-object-with-lock.log"
aws s3api --endpoint-url=$ENDPOINT put-object --bucket $BUCKET --key $OBJECT --body $OBJECT

#Response
{
  "ETag": "\"c6125a47483a2823d993da3d31ba6a50\"",
  "VersionId": "MzMxNjlmNzItOWQ3Ni00MmWI0LTl1OGYtZDQyN2RkMjRlN2Jk"
}
```

Set Object retention mode and date

```
aws s3api --endpoint-url=$ENDPOINT put-object-retention --bucket $BUCKET --key $OBJECT --retention
↪ Mode=COMPLIANCE, RetainUntilDate=2020-04-01
```

Retrieve Object Lock configuration

```
aws s3api --endpoint-url=$ENDPOINT get-object-retention --bucket $BUCKET --key $OBJECT

#Response
{
  "Retention": {
```

(continues on next page)

(continued from previous page)

```

    "Mode": "COMPLIANCE",
    "RetainUntilDate": "2020-04-01T00:00:00"
  }
}

```

In this example, the object will remain locked until April 1st, 2020.

List an object's versions and attempt to delete a specific version

List an object's versions:

```

aws s3api --endpoint-url=$ENDPOINT list-object-versions --bucket $BUCKET --prefix $OBJECT
{
  "Versions": [
    {
      "ETag": "%22c6125a47483a2823d993da3d31ba6a50%22",
      "Size": 14871255,
      "StorageClass": "STANDARD",
      "Key": "new-object-with-lock.log",
      "VersionId": "MzMxNjlmNzItOWQ3Ni00MmWI0LTl1OGYtZDQyN2RkMjRlN2Jk",
      "IsLatest": true,
      "LastModified": "2020-03-08T16:54:30.225Z",
      "Owner": {
        "DisplayName": "veeam:client",
        "ID": "veeam:client"
      }
    }
  ]
}

```

Attempt to delete a specific version of an object:

```
aws s3api --endpoint-url=$ENDPOINT delete-object --bucket=$BUCKET --key=$OBJECT --version-id=$VERSION
```

An error occurred (AccessDenied) when calling the DeleteObject operation: Access Denied.

Configure container's default retention values with object lock

Via the CLI, there is the option to configure an object lock and its default retention values at the bucket level, in a single command.

✓ **Note:** This option is only available via the CLI.

Configure a bucket's object lock with default retention values

```

aws s3api --profile $AWSUSER --endpoint-url=$ENDPOINT put-object-lock-configuration --bucket $BUCKET --
↳ object-lock-configuration '{ "ObjectLockEnabled": "Enabled", "Rule": { "DefaultRetention": { "Mode":
↳ "COMPLIANCE", "Days": 50 } } }'

```

Retrieve the bucket's object lock configuration:

```
aws s3api --profile $AWSUSER --endpoint-url=$ENDPOINT get-object-lock-configuration --bucket $BUCKET
```

Response

(continues on next page)

(continued from previous page)

```
{
  "ObjectLockConfiguration": {
    "ObjectLockEnabled": "Enabled",
    "Rule": {
      "DefaultRetention": {
        "Mode": "COMPLIANCE",
        "Days": 50
      }
    }
  }
}
```

4.5.4 Container logging

Version: 23.09

The Container Logging Feature is designed to provide logging capabilities for objects stored in the object storage system. This feature will provide insights into object-level activities and audit trails, which will help them understand access patterns, usage statistics, and security incidents.

Logging can be enabled at container level, for the purpose of auditing and analyzing all operations on objects.

Logs are generated into a separate target container in the same account. A container that has logging enabled cannot be used as the target for any audit logs, neither its own logs, nor audit logs for any other container.

A target container containing audit logs can be used to collect logs for multiple containers that have logging enabled. Write permissions on the target container are required.

Enabling logging is activated either when creating a container, or in the **Container Logging** tab in the details pane for existing containers. See [Create Containers](#) and [Details Pane](#).

The configuration includes an option to define a logging prefix, so that locating and identifying log objects is simplified.

In order to avoid additional workload on the Object Storage platform and to avoid any impact on client performance, the logs will be delivered once a day. The interval for sending the logs to the target container will be shortened in the Object Storage next release.

There is no extra charge for enabling Container logging. However, the logs that are stored in the system will increase storage consumption. The logs collected and stored can be deleted at any time.

Enabling Container Logging

Container Logging can be enabled/disabled at any time from the following interfaces:

1. From Zadara's Object Storage Object Storage management console
2. While using S3 API (aws cli package or S3 compatible software that supports enabling S3 Server Logging)

Log record structure

In order to make use of existing tools and avoid the overhead when using the audit logs Zadara log structure is similar to the AWS S3 Server Logging option, while not all attributes can match due to the differences in the platform, logs that doesn't have a Zadara Object Storage presence or are not currently implemented will have the value of - (dash).

Log filenames have the format:

```
<user defined prefix>-<ISO formatted log delivery date>
```

Example (no user defined prefix):

```
2023-12-02-22-07-22
```

The log record structure:

Field name	Description
Container Owner	The canonical user ID of the source container's owner.
Container	The name of the container that the request was processed against. Malformed requests will not appear in the log.
Time	The date and time that the request was received in UTC format. In <code>strftime()</code> terminology: [%d/%b/%Y:%H:%M:%S %z]. For example, [06/Feb/2019:00:00:38 +0000]
Remote IP	The requester's IP address. Proxies and firewalls might obscure the actual IP address of the requesting machine.
Requester ID	The user ID of the request, and "-" for unauthenticated requests.
Request ID	A generated string that uniquely identifies the request.
Operation	The object operation type: GET/POST/PUT/DELETE/HEAD
Key	The request's object name (key). For example, /objectpath/2023/08/object-name.key
Request-URI	The Request-URI part of the HTTP request message.
HTTP status	The numeric HTTP status code of the response.
Error Code	The error code, or "-" if no error occurred.
Bytes Sent	The number of response bytes sent, excluding HTTP protocol overhead, or "-" if zero.
Object Size	The object's total size in bytes.
Total Time	The number of milliseconds that the request was in flight from the server's perspective, measured from the time that the request is received to the time that the last byte of the response is sent. Measurements made from the client's perspective might be longer because of network latency.
User-Agent	The value of the HTTP <code>User-Agent</code> header. For example, <code>curl/8.1.2</code>
Version Id	The version ID in the request, or "-" if the operation doesn't take a "versionId" parameter.
Host Id	The VC that handled the request.

Container Logging log format examples

Container logging example for HEAD request

```
- savihou-manual [23/Nov/2023:02:23:44 +0000] 172.23.224.102 7c6d15bdf0d74802a9c751dd5c55bfa2
↳ txe6a729193f9e479f876cd-00655eb7b0 HEAD - "HEAD / HTTP/1.1" 200 - - - 0.0106 - - "APN/1.0 Veeam/1.0
↳ Backup/12.0" - - - - -
```

Container logging example for GET request

```
- savihou-manual [23/Nov/2023:06:58:01 +0000] 172.23.224.102 7c6d15bdf0d74802a9c751dd5c55bfa2
↳ tx39562231422843d99a776-00655ef7f9 GET - "GET /?delimiter=%2F&max-keys=1000&prefix= HTTP/1.1" 200 - 320
↳ - 0.0216 - - "S3 Browser/11.4.5 (https://s3browser.com)" - - - - -
```

Container logging example for PUT request

```
- savihou-manual [23/Nov/2023:02:24:36 +0000] 172.23.224.102 7c6d15bdf0d74802a9c751dd5c55bfa2
↳ tx8ebb2ddeb8fc4535826de-00655eb7e4 PUT 10k-objects-1/randobj-828 "PUT /10k-objects-1/randobj-828 HTTP/1.
↳ 1" 200 - - 4096 0.0712 - - "aws-sdk-go/1.45.18 (go1.10.4; linux; amd64)" - - - - -
```

4.5.5 Object Lifecycle Policy

Version: 23.09-SP1

An optional Object Lifecycle Policy can be configured for a container, to determine the retention period for the container's objects.

One or more Object Lifecycle Rules establish an Object Lifecycle Policy. A container can have a maximum of one Object Lifecycle Policy. The rules can be viewed, configured and managed in the container's south pane **Object Lifecycle Rules** tab.

An Object Lifecycle Rule defines object retention on the basis of a specific date or number of days since object creation. A single rule can be configured to apply across all objects in a container, or can be limited to objects according to a specific prefix such as folder or filename prefixes, or even filename prefixes within a specified folder tree. Each rule can be enabled or disabled.

Managing Object Lifecycle Policy Rules

A container's Object Lifecycle Policy is configured and maintained in the set of rules in the container's south pane **Object Lifecycle Rules** tab. For examples of managing Object Lifecycle Policy Rules using the AWS S3 API CLI, see [Object Lifecycle Policy configuration examples](#).

Creating or Editing an Object Lifecycle Rule

Both the **Add** and the **Edit** functions invoke the same **Object Lifecycle Rule** modal, for creating a new rule or editing an existing rule, respectively.

1. In the **Console** pane, select the container.
2. In the south pane, click the **Object Lifecycle Rules** tab.
3. In the **Object Lifecycle Rules** tab's toolbox menu:
 - To create a new rule, click **Add**:
 - To edit an existing rule, select the rule and click **Edit**.

4. In the **Object Lifecycle Rule** modal that opens, enter or edit the following fields:

1. **Rule name:** A unique freetext string describing or identifying the rule.
2. **Status:** Select **Enabled** (default) or **Disabled**.
3. **Object name prefix:** Enter a filter to limit the rule to matching objects, or leave it empty to apply the rule across all objects in the container.

For example:

- **2023-06-:** All objects that have names beginning with **2023-06-**.
- **/log/2023/abc:** All objects that have names beginning with **abc**, in the **/log/2023** folder.

✓ **Note:**

- When the prefix is left empty, a note displays a directive to mark an acknowledgement checkbox before saving the rule, as a confirmation that the rule will apply to all objects in the container.
- If the prefix overlaps another rule's filter, a note displays an alert that the rule with the earlier expiration will be honored.

If the prefix is left empty indicating that the rule applies to all objects in the container, and if the container already has another rule, the alert displays to indicate the overlap.

4. At least one of the following options must be configured:

- **Expire current version of objects:** Configure the rule to expire matching objects, either a number of **Days** after object creation, or at 00:00 on a specified **Date**.
 - Mandatory in a non-versioned container:
If the container is not versioned, there is no possibility to deselect the **Expire current version of objects** option.
 - Optional in a [Versioned Container](#):
 - * Unchecked by default.
 - * If **Expire current version of objects** is selected, the **Delete expired object delete markers** option cannot be selected.
- **Permanently delete noncurrent versions of objects** ([Versioned Container](#) only):
 - Unchecked by default.
 - Option to apply the rule to permanently delete versions of matching objects that are not current.
- **Delete expired object delete markers** ([Versioned Container](#) only):

✓ **Note:** A delete marker is a placeholder for a versioned object that was specified in a DELETE request that did not specify a version. In a versioned container, the object is marked as deleted and treated as such, even though it is not actually deleted.

- Unchecked by default.
- Option to apply the rule to delete expired delete markers of matching objects.
- If **Delete expired object delete markers** is selected, the **Expire current version of objects** option cannot be selected.

5. To save the rule:

- For a new rule, click **Create**.
- For an existing rule, click **Update**.

Enabling or Disabling an Object Lifecycle Rule

Every Object Lifecycle Rule is created with its **Status** configured as either **Enabled** or **Disabled**.

It is possible to switch the rule's **Status** at any time, from **Enabled** to **Disabled**, or vice versa.

To switch the rule's **Status**:

1. In the **Console** pane, select the container.
2. In the south pane, click the **Object Lifecycle Rules** tab.
3. In the **Object Lifecycle Rules** tab's toolbox menu, click **Enable/Disable**.
4. The **Confirm Status Change** modal displays.

A note displays, informing that it could take some time to update the Object Lifecycle Policy's configuration.

Click **Yes** to confirm changing the selected rule's **Status**.

Deleting an Object Lifecycle Rule

To delete an Object Lifecycle Rule:

1. In the **Console** pane, select the container.
2. In the south pane, click the **Object Lifecycle Rules** tab.
3. In the **Object Lifecycle Rules** tab's toolbox menu, click **Delete**.
4. The **Confirm Deletion** modal displays.

A note displays, informing that it could take some time to update the Object Lifecycle Policy's configuration.

Click **Yes** to confirm deleting the selected rule.

4.6 Large objects support

Zadara's Object Storage has a 5GB limit on the size of a single uploaded object. However, leveraging segmentation a single object size is unlimited. Segments of the larger object are uploaded and a special manifest file is created that, when downloaded, sends all the segments concatenated as a single object. This also offers much greater upload speed with the possibility of parallel uploads of the segments.

Dynamic Large Object (DLO) is supported out-of-the-box.

The majority of object storage clients support multi-part upload and the allows the user to set the segment size.

4.6.1 Failed upload handling

In case the multipart upload doesn't complete, the NextGen Object Storage will not assemble the object parts and will not create any object. The parts will remain stored in the Object Storage for a period of 15 days, this until the Object Storage segment tracker will cleanup automatically the orphan parts. In this case aborted/failed uploads incomplete parts will be considered as part of the account used capacity.

USING OBJECT STORAGE CLIENTS

Standard client tools can be used to browse objects in the Object Storage. This section will help configuring Object Storage Client Tools to work against Object Storage. In order to access the Object Storage the client tool must be configured with the user's authentication credentials.

The Object Storage support two API interfaces:

1. AWS S3 API
2. Openstack Swift API

The required parameters can be found in the Object Storage User Information page. Information for the user currently logged in to the Object Storage management interface displayed by clicking the user name on the management interface's upper right corner.



5.1 AWS S3 Compatible clients

5.1.1 Supported S3 APIs

The Object Storage is utilizing Openstack Swift's S3 Middleware. As S3 is an AWS product, it includes some features that are AWS oriented and are outside of the scope of Zadara's Object Storage offering.

Zadara supports the following S3 operations:

Object operations

- GET Object
- HEAD Object
- PUT Object
- PUT Object Copy
- DELETE Object
- DELETE Multiple Objects
- GET Object ACL

- PUT Object ACL
- Initiate Multipart Upload
- Upload Part
- Upload Part Copy
- Complete Multipart Upload
- Abort Multipart Upload
- List Parts
- List Multipart Uploads
- Object Retention Support

Bucket operations

- GET Bucket List Objects
- HEAD Bucket
- PUT Bucket
- DELETE Bucket
- GET Bucket ACL
- PUT Bucket ACL
- GET Bucket Location
- GET Bucket Versioning
- GET Bucket Logging
- PUT Bucket Logging
- GET Bucket CORS Version: 23.09
- PUT Bucket CORS Version: 23.09
- DELETE Bucket CORS Version: 23.09
- OPTIONS Object Version: 23.09
- PUT Bucket Lifecycle Configuration Version: 23.09
- GET Bucket Lifecycle Configuration Version: 23.09
- DELETE Bucket Lifecycle Configuration Version: 23.09

5.1.2 Authentication information

For Object Storage connectivity, it is required to gather the following information from the Object Storage management UI:

1. Object Storage Endpoint - an endpoint entry will be available according to the networking layout of the Object Storage (i.e. for Object Storage with a Public IP interface we'll have a record matching to the public network interfaces)
2. Object Storage region.
3. S3 API Access Key/Secret Key

In the Object Storage GUI, navigate to the User Information section (top right corner, by clicking the logged in username).

zadara
NEXTGEN OBJECT STORAGE

DASHBOARD

RESOURCES

- Drives
- Virtual Controllers
- Policies
- Console

SYSTEM

- Usage Reports
- Performance
- Settings
- Diagnostics

ACCOUNT MANAGEMENT

LOGS

TIME ZON

USER INFORMATION

CHANGE PASSWORD

Account information

Username:	object-admin
Email:	object-admin@zadarastorage.com
Account:	object-storage
User ID:	2c52d86b326349d9a1a07695fbbd0857
Account ID:	0db70cda645043b9af13f4eba5f89a9c
Dual Factor Authentication::	Deactivated Activate

Authentication

S3 Access Key:	00e80de45e3f472abee62ba24335f549 3
S3 Secret Key:	***** Copy Refresh
Region:	us-east-1 2
API Token:	***** Copy Refresh

Connectivity - Front End Network

API Endpoint:	https://vsa-00000100-zcloud-01.zadarazios.com:443 1
V3 Auth Endpoint:	https://vsa-00000100-zcloud-01.zadarazios.com:5000/v3
Account URL:	https://vsa-00000100-zcloud-01.zadarazios.com:443/v1/AUTH_0db70cda645043b9af13f4eba5f89a9c

Connectivity - Public Network

Public IP:	18.19.20.21
Public API Endpoint:	vsa-00000100-public-zcloud-01.zadarazios.com 1
Public V3 Auth Endpoint:	https://vsa-00000100-public-zcloud-01.zadarazios.com:5000/v3
Public Account URL:	vsa-00000100-public-zcloud-01.zadarazios.com:443/v1/AUTH_0db70cda645043b9af13f4eba5f89a9c

5.1.3 S3 Browser

S3 Browser can be used to administrate and perform object operations against ZADARA's Object Storage. The account information in S3 Browser should be configured according to the following example (S3 Compatible Storage):

Edit Account
Edit account details and click Save changes

Account Name:
Zadara VPSA Object Storage
Assign any name to your account.

Account Type:
S3 Compatible Storage
Choose the storage you want to work with. Default is Amazon S3 Storage.

REST Endpoint:
vsa-0000003f-zadara-iop-01.zadara.com
Specify S3-compatible API endpoint. It can be found in storage documentation. Example: rest.server.com:8080

Access Key ID:
74a8cb9470e546dd965af3aaaac788a9
Required to sign the requests you send to Amazon S3, see more details at <https://s3browser.com/keys>

Secret Access Key:
.....
Required to sign the requests you send to Amazon S3, see more details at <https://s3browser.com/keys>

Encrypt Access Keys with a password:
.....
Turn this option on if you want to protect your Access Keys with a master password.

Use secure transfer (SSL/TLS)
If checked, all communications with the storage will go through encrypted SSL/TLS channel

[Advanced S3-compatible storage settings](#)

Once the Endpoint and authentication details are configured properly, click on the Advanced S3-compatible storage settings

In the advanced settings select the following:

1. Signature version - Signature V4
2. Addressing model - Path style
3. Override storage regions - specify the Object Storage region name; the format is `Region Name=<region name>`.

Close and save the account information.

✓ **Note:** S3 Browser client is hard-coded to use `us-east-1` as the default region, In order to use Object Storage v4 signatures, ensure the same region value is configured in your Object Storage or override the default S3Browser region name in the Advanced Settings options.

5.1.4 S3cmd

The credentials can be retrieved from the Object Storage logged in “User Information” properties.

`/etc/.s3cfg`

```
[default]
access_key = <S3 Access Key>
secret_key = <S3 Secret Key>
host_base = vsa-00000001-cloud-01.zadara.com
host_bucket = vsa-00000001-cloud-01.zadara.com
use_https = True
```

✓ **Note:**

- `access_key` is the user S3 Access Key
 - `secret_key` is the user S3 Secret Key
 - `host_base` is the HTTPS path to the Object Storage being accessed
-

5.1.5 AWS Command Line Interface

Update the default/create new profile for the Object Storage within aws configuration file.

`~/.aws/config`

```
[profile zadara]
s3 =
    signature_version = s3v4
```

✓ **Note:** It is possible to use both AWS v4/v2 signatures with S3-compatible storage such as Zadara Object Storage.

`~/.aws/credentials`

```
[zadara]
aws_access_key_id = <S3 Access Key>
aws_secret_access_key = <S3 Secret Key>
```

The credentials can be retrieved from the Object Storage logged in “User Information” properties.

Example of usage:

```
$ aws s3 --profile=zadara --endpoint-url=https://vsa-00000001-cloud-01.zadara.com --region=us-east-1 ls
↳ s3://zadara-test

2018-04-01 19:00 mytestfile1
```

(continues on next page)

(continued from previous page)

```
2018-04-01 19:10 mytestfile2
2018-04-01 19:20 mytestfile3
```

✓ Note:

- `profile` is the name of the credentials and config profile specified above (in this case, “zadara”)
 - `endpoint-url` is the HTTPS path to the Object Storage being accessed
 - `region` should match the Region defined in the Object Storage [settings page](#) (Zadara default: `us-east-1`)
-

Common operations examples

- Creating a presigned URL - this allows anyone who receives the presigned URL to retrieve the object with HTTP GET request. The operation uses the S3 credentials and region field to generate the pre-signed URL.

```
$ aws s3 presign --profile <AWS CLI profile name> --endpoint \
https://<object storage api endpoint> \
s3://<container/bucket name>/<object name> --expires-in <expiration in seconds>
```

For more information please refer to the official [AWS CLI Command Reference](#).

CORS configuration examples

Version: 23.09

CORS configurations can be configured using AWS and Swift APIs.

Important: For best practice, it is highly recommended not to use both AWS and Swift methods for CORS configurations.

CORS settings that are configured using AWS APIs take precedence over CORS settings that are configured via Swift APIs.

When using an AWS API to update a CORS configuration, all existing Swift CORS headers are removed. However, if a CORS configuration that was configured using AWS APIs is updated using Swift APIs, the Swift CORS headers are ignored. For CORS settings configured via AWS APIs, Swift CORS headers are always ignored, irrespective of whether Swift or AWS APIs were used first.

PUT Bucket CORS

To configure a bucket to allow cross-region requests, use the AWS S3 `PutBucketCors` bucket operation.

For example:

```
aws s3api put-bucket-cors --bucket my-bucket --profile=zadara --endpoint-url=https://vsa-00000103-public-
↪zadara-qa19.zadarazios.com --cors-configuration file://cors.json
```

A CORS configuration is a list of CORS rules to apply to a bucket. It is mandatory to define at least one rule in a CORS configuration.

Command line parameters:

- **--bucket**: The name of the NGOS object container on which the CORS rules will be configured.
- **--profile**: The profile name of the configuration in the `~/.aws/config` and `~/.aws/credentials` files, for account credentials as described above.
- **--endpoint**: The account's API Endpoint or Public API Endpoint.
- **--cors-configuration**: The file comprising the CORS configuration rules.

In this example, the CORS configuration rules are defined in the `cors.json` configuration file:

```
{
  "CORSRules": [
    {
      "ID": "xyz-abc-def-wxy",
      "AllowedOrigins": ["https"],
      "AllowedHeaders": ["*"],
      "AllowedMethods": ["PUT", "POST", "DELETE"],
      "MaxAgeSeconds": 3000,
      "ExposeHeaders": ["x-amz-server-side-encryption"]
    },
    {
      "AllowedOrigins": ["*"],
      "AllowedHeaders": ["x-abc-*"],
      "AllowedMethods": ["GET"]
    },
    {
      "AllowedOrigins": ["http://*.example.com", "https://xyz.com"],
      "AllowedHeaders": ["x-def"],
      "AllowedMethods": ["PUT", "POST", "DELETE"],
      "MaxAgeSeconds": 4000
    }
  ]
}
```

CORS Rule Properties:

CORSRules: A CORS configuration is a list of CORS rules. It is mandatory to define at least one rule.

Each rule in a CORS configuration list must have the following **mandatory** properties:

- **ID** (mandatory)
 - A unique string identifying the rule, up to a maximum length of 255 characters.
- **AllowedMethods** (mandatory)
 - A comma-separated list of one or more permitted REST methods allowed on the bucket for the current rule.
 - Only the "GET", "PUT", "POST", "DELETE" and "HEAD" methods can be specified, and the AllowedMethods list must comprise at least one method.
 - Wildcards and empty strings are not permitted.
- **AllowedOrigins** (mandatory)
 - A comma-separated list of origin sites allowed access to the bucket for the current rule.
 - An asterisk (*) wildcard can be specified in an origin string.
 - * A maximum of one asterisk can be specified in an origin string.
 - * An origin string that includes an asterisk specifies permitted access to the bucket from all origins that match the permutations. For example, "AllowedOrigins": ["http://*.example.com"] specifies permitted access from all subdomains of `example.com`.

- * AllowedOrigins with the value of a single origin comprising only an asterisk ("AllowedHeaders": ["*"]) specifies permitted access from all origins.
- AllowedOrigins comprising strings without an asterisk must be fully defined, and only CORS requests from origins with an exact match are permitted access.

Each rule in a CORS configuration list can have the following **optional** properties:

- **ID** (optional)
 - A string to identify the rule, up to 255 characters.
- **AllowedHeaders** (optional)
 - A comma-separated list of headers that are allowed on this bucket for the current rule.
 - An asterisk (*) wildcard can be specified in a header.
 - * A maximum of one asterisk can be specified in a header.
 - * An allowed header string that includes an asterisk specifies that all matching permutations of the header are allowed. For example, "AllowedHeaders": ["x-abc- *"] specifies that all headers prefixed by "x-abc-" are allowed.
 - * AllowedHeaders with the value of only an asterisk ("AllowedHeaders": ["*"]) specifies that headers with any value are permitted.
 - AllowedHeaders comprising strings without an asterisk must be fully defined, and only CORS requests with headers having an exact match are permitted.
- **ExposeHeaders** (optional)
 - A comma-separated list of headers that can be exposed to the client from a CORS request.
 - Wildcards are not permitted.
- **MaxAgeSeconds** (optional)
 - A positive integer specifying the maximum number of seconds that an OPTIONS request result can be cached by the browser for the current rule.

GET Bucket CORS

To view the CORS configuration on a bucket, use the AWS S3 `GetBucketCors` bucket operation.

For example:

```
aws s3api get-bucket-cors --bucket my-bucket --profile=zadara --endpoint-url=https://vsa-00000103-public-
↳zadara-qa19.zadarazios.com
```

The CORS configuration is returned in JSON format.

Sample response:

```
{
  "CORSRules": [
    {
      "ID": "xyz-abc-def-wxy",
      "AllowedHeaders": [
        "*"
      ],
      "AllowedMethods": [
        "PUT",
```

(continues on next page)

(continued from previous page)

```

    "POST",
    "DELETE"
  ],
  "AllowedOrigins": [
    "https"
  ],
  "ExposeHeaders": [
    "x-amz-server-side-encryption"
  ],
  "MaxAgeSeconds": 3000
},
{
  "AllowedHeaders": [
    "x-abc-*"
  ],
  "AllowedMethods": [
    "GET"
  ],
  "AllowedOrigins": [
    "*"
  ]
},
{
  "AllowedHeaders": [
    "x-def"
  ],
  "AllowedMethods": [
    "PUT",
    "POST",
    "DELETE"
  ],
  "AllowedOrigins": [
    "http://*.example.com",
    "https://xyz.com"
  ],
  "MaxAgeSeconds": 4000
}
]
}

```

✓ **Note:** If the CORS configuration was configured using a Swift API and not by an AWS API, then the **GetBucketCors** operation does not return any data.

DELETE Bucket CORS

To delete a CORS configuration from a bucket, use the AWS S3 **DeleteBucketCors** bucket operation.

For example:

```
aws s3api delete-bucket-cors --bucket my-bucket --profile=zadara --endpoint-url=https://vsa-00000103-
↳public-zadara-qa19.zadaracloud.com
```

✓ **Note:** Only AWS CORS configurations are erased from the container collection.

CORS headers configured using Swift APIs are not affected.

OBJECT Options (CORS)

To determine whether the server will permit a request on a CORS-enabled bucket or on any of its objects to proceed, an OPTIONS request can be invoked to direct the browser to send a preflight request (i.e. a preliminary probe) to the same URL.

Examples of typical signatures for an OPTIONS request:

- For a CORS-enabled bucket

```
curl -i -XOPTIONS -H "X-Auth-Token: <token>" \  
-H "Origin: http://abc.com" \  
-H "Access-Control-Request-Method: POST" \  
-H "Access-Control-Request-Headers: <header1>" "https://(endpoint)/(bucket-name)"
```

- For an object in a CORS-enabled bucket

```
curl -i -XOPTIONS -H "X-Auth-Token: <token>" \  
-H "Origin: http://abc.com" \  
-H "Access-Control-Request-Method: POST" \  
-H "Access-Control-Request-Headers: <header1>" "https://(endpoint)/(bucket-name)/(object-key)"
```

✓ **Note:** Some versions of curl might require the headers in single quotes, and the URL in double quotes.

OPTIONS request headers:

- **Mandatory** header:

- **Origin** header (mandatory):

The Origin header is checked against the list of AllowedOrigins in the CORS configuration rules.

If there are no rules that match this header, the OPTIONS request fails with the HTTP_UNAUTHORIZED (401) status.

- **Access-Control-Request-Method** header (mandatory):

The Access-Control-Request-Method header is checked against the list of AllowedMethods in the CORS configuration rules.

If there are no rules that match this header, the OPTIONS request fails with the HTTP_UNAUTHORIZED (401) status.

Each rule in the CORS configuration is checked for matches to both AllowedOrigins and AllowedMethods. If there is no match, the check proceeds to the next rule.

Success is determined when a match is found. No further rules are checked, and the browser can proceed with invoking the request.

If none of the rules match, the HTTP_UNAUTHORIZED status is raised on OPTIONS request, indicating to the browser that it cannot invoke the request on the target CORS-enabled bucket using the requested Origin and method.

- **Optional** header:

- **Access-Control-Request-Headers** header (optional):

The Access-Control-Request-Headers header is checked against the list of the optional AllowedHeaders in the CORS configuration rule that matches the mandatory headers (AllowedOrigins and AllowedMethods).

If AllowedHeaders is specified in the CORS rule, but if any of the Access-Control-Request-Headers do not match, the OPTIONS request returns a success status, but without any Access-Control-* headers in the response.

In addition to a success status code, a successful OPTIONS request response should also contain all Access-Control-* headers, indicating the origin, method and request headers that are allowed on the bucket.

OPTIONS request example:

```
curl -i -XOPTIONS -H 'X-Auth-Token: <token>' \
  -H 'Origin: https://example.com' \
  -H 'Access-Control-Request-Method: PUT' \
  -H 'Access-Control-Request-Headers: x-123-abc' "https://vsa-0000003d-zadara-qa21.zadarazios.com:443/v1/
  ↪AUTH_2eb509f93b0c4790890061007cdd62a4/corsbucket/delete.json"
```

OPTIONS request response example for a method that is configured in the CORS rule:

```
HTTP/1.1 200 OK
Allow: HEAD, POST, GET, OPTIONS, DELETE, PUT
access-control-allow-origin: https://example.com
vary: Origin, Access-Control-Request-Headers
access-control-max-age: 3000
Access-Control-Allow-Methods: PUT, POST, DELETE
access-control-allow-headers: x-123-abc
x-trans-id: txb7997e22016c4e449733e-006527a288
x-openstack-request-id: txb7997e22016c4e449733e-006527a288
Server: Zadara
Content-Length: 0
Date: Thu, 18 Apr 2024 07:38:49 GMT
```

In the response example, the following matching elements have been returned in response headers:

Returned response header	Returned elements
access-control-allow-origin	Origin in the request that matches the CORS rule's AllowedOrigins .
access-control-allow-methods	AllowedMethods in the matching CORS rule.
access-control-allow-headers	AllowedHeaders in the CORS rule.
access-control-max-age	MaxAgeSeconds , if specified in the CORS rule that matches the origin and method request headers.

Important: If even only one of the request headers does not match **AllowedHeaders** in the CORS rule, the OPTIONS response does not return any of the access-control-* headers, including the access-control-allow-origin and access-control-allow-methods.

ExposeHeaders is not returned in an OPTIONS response.

OPTIONS request response example for a method that is not authorized in the CORS rule:

```
HTTP/1.1 401 Unauthorized
Content-Type: text/html; charset=UTF-8
Allow: OPTIONS, HEAD, PUT, POST, DELETE, GET
Content-Length: 131
```

(continues on next page)

(continued from previous page)

```
WWW-Authenticate: Keystone uri='<Keystone URI>'
x-trans-id: txf9e61f89a7424a84827e9-006641c489
x-openstack-request-id: txf9e61f89a7424a84827e9-006641c489
Server: Zadara
Date: Thu, 18 Apr 2024 07:36:23 GMT

<html><h1>Unauthorized</h1><p>This server could not verify that you are authorized to access the document
↳you requested.</p></html>
```

Object Lifecycle Policy configuration examples

Version: 23.09

An optional [Object Lifecycle Policy](#) can be configured for a container, to determine the retention period for the container's objects.

By creating one or more policy rules for a container, an Object Lifecycle Policy is established for the container.

The AWS S3 API examples in this section depict operations creating and managing a container's Object Lifecycle Policy, with the following assumptions:

- The container is named `my-container`.
- The container is in an account that has the public endpoint URL `https://vsa-00000103-public-zadara-qa19.zadarazios.com`.
- The account and its credentials are defined in a profile named `zadara`, configured in `~/.aws/config` and `~/.aws/credentials` as described in [AWS Command Line Interface](#).

PUT Bucket Lifecycle Configuration

To configure rules for a container's Object Lifecycle Policy, use the AWS S3 `PutBucketLifecycleConfiguration` bucket operation.

The AWS S3 CLI can invoke a configuration file that specifies lifecycle rules. For example, the following lifecycle configuration file `lifecycle.json` specifies two rules for the container's Object Lifecycle Policy:

- The first rule specifies:
 - Objects in the container have a retention period of 30 days from object creation.
 - In this example, there is no `Filter` parameter, indicating that the rule applies to all objects in the container.
- The second rule specifies:
 - Objects in the container have a retention period up to 00:00 on 1st July 2024, irrespective of their age.
 - In this example, the `Filter` parameter applies this rule only to objects in the `log` folder.
 - Note that this rule is set initially to be disabled.

```
{
  "Rules": [
    {
      "Expiration": {
        "Days": 30
      },
      "ID": "30-day expiration rule",
```

(continues on next page)

(continued from previous page)

```

        "Status": "Enabled"
    },
    {
        "Expiration": {
            "Date": "2024-07-01T12:00:00"
        },
        "ID": "2024 first half year expiration",
        "Filter": {
            "Prefix": "log/"
        },
        "Status": "Disabled"
    }
}
]
}

```

An AWS S3 CLI example that applies the rules specified in the lifecycle configuration file `lifecycle.json`, to configure a Lifecycle Policy for a container named `my-container`:

```

aws s3api put-bucket-lifecycle-configuration \
--bucket my-container \
--profile=zadara \
--endpoint-url=https://vsa-00000103-public-zadara-qa19.zadarazios.com \
--lifecycle-configuration file://lifecycle.json

```

GET Bucket Lifecycle Configuration

To retrieve a container's Object Lifecycle Policy configuration, use the AWS S3 `GetBucketLifecycleConfiguration` bucket operation.

For example, to retrieve the Object Lifecycle Policy configuration in JSON output format for a container named `my-container`:

```

aws s3api get-bucket-lifecycle-configuration \
--bucket=my-container \
--output=json \
--profile=zadara \
--endpoint-url=https://vsa-00000103-public-zadara-qa19.zadarazios.com

```

JSON response:

```

{
  "Rules": [
    {
      "Expiration": {
        "Days": 30
      },
      "ID": "30-day expiration rule",
      "Status": "Enabled"
    },
    {
      "Expiration": {
        "Date": "2024-07-01T12:00:00"
      },
      "ID": "2024 first half year expiration",
      "Filter": {

```

(continues on next page)

(continued from previous page)

```

        "Prefix": "log/"
      },
      "Status": "Disabled"
    }
  ]
}

```

DELETE Bucket Lifecycle

To delete a container's Object Lifecycle Policy configuration, use the AWS S3 `DeleteBucketLifecycle` bucket operation. For example, to delete the Object Lifecycle Policy configuration for a container named `my-container`:

```

aws s3api delete-bucket-lifecycle \
--bucket=my-container \
--profile=zadara \
--endpoint-url=https://vsa-00000103-public-zadara-qa19.zadarazios.com

```

Multipart Object example

Multipart object calls are used in scenarios where large objects need to be uploaded or downloaded efficiently. Multipart uploads are recommended for files larger than 100MB, and are mandatory for files larger than 5GB. Parts can be uploaded in parallel, and are reassembled into a single object at the destination container folder.

The following example illustrates the steps to upload a large file `largeFile.dat` that has a size of 725,135,360 bytes (approximately 700MB):

1. Split the file into smaller parts, in this example, three parts (files) of 200MB each, and one part (file) of the remainder.

✓ **Note:** This example uses the linux `split` command that inputs a file, outputting chunks of a specified size, to files with default names starting with `xa`.

```

$ ls -l largeFile.dat

-rwxrwxrwx 1 user1 zadara 725135360 Feb  3 15:22 largeFile.dat

$ split -b 200m largeFile.dat

$ ls -l xa*

-rwxrwxrwx 2 user1 zadara 209715200 Feb 10 08:27 xaa*
-rwxrwxrwx 2 user1 zadara 209715200 Feb 10 08:27 xab*
-rwxrwxrwx 2 user1 zadara 209715200 Feb 10 08:27 xac*
-rwxrwxrwx 2 user1 zadara  95989760 Feb 10 08:27 xad*

```

2. Initialize the upload process with directives that the multipart file `largeFile.dat` must be assembled into the destination folder `partsExample` in the bucket `mybucket`.

```

$ aws s3api create-multipart-upload \
--profile=zadara \

```

(continues on next page)

(continued from previous page)

```
--endpoint-url=https://vsa-00000001-cloud-01.zadara.com \
--bucket mybucket \
--key 'partsExample/largeFile.dat'
```

Response:

```
{
  "Bucket": "mybucket",
  "Key": "partSexample/largeFile.dat",
  "UploadId": "0WM1YTdl0WUt0Dd1MC00NGY1LTk3ZWYt0WNiNTE2ZjBj0DJj"
}
```

- From the response, note the value of `UploadId`, which is needed for all the subsequent calls of this multipart upload.

Upload the first part, `--part-number 1`, the `xaa` file, and specify the `UploadId`:

```
$ aws s3api upload-part --profile=zadara \
--endpoint-url=https://vsa-00000001-cloud-01.zadara.com \
--bucket mybucket \
--key "partsExample/largeFile.dat" \
--part-number 1 \
--body xaa \
--upload-id 0WM1YTdl0WUt0Dd1MC00NGY1LTk3ZWYt0WNiNTE2ZjBj0DJj
```

Response:

```
{
  "ETag": "\"6eac0b608eee111876931ddb6907429\""
}
```

- From the response, note the value of `ETag` for each part. These will be required in the multipart upload completion step, where the parts are reassembled into a single object at the destination.

Upload the second part, `--part-number 2`, the `xab` file, taking note of the value of its `ETag` from the response:

```
$ aws s3api upload-part --profile=zadara \
--endpoint-url=https://vsa-00000001-cloud-01.zadara.com \
--bucket mybucket \
--key "partsExample/largeFile.dat" \
--part-number 2 \
--body xab \
--upload-id 0WM1YTdl0WUt0Dd1MC00NGY1LTk3ZWYt0WNiNTE2ZjBj0DJj
```

Response:

```
{
  "ETag": "\"aaffcae340799e25223970903f917cd6\""
}
```

- Upload the third part, `--part-number 3`, the `xac` file, taking note of the value of its `ETag` from the response:

```
$ aws s3api upload-part --profile=zadara \
--endpoint-url=https://vsa-00000001-cloud-01.zadara.com \
--bucket mybucket \
--key "partsExample/largeFile.dat" \
--part-number 3 \
--body xac \
--upload-id 0WM1YTdl0WUt0Dd1MC00NGY1LTk3ZWYt0WNiNTE2ZjBj0DJj
```

Response:

```
{
  "ETag": "\"d6d7687673a728c8dc5683ce8ada395f\""
}
```

6. Upload the fourth part, `--part-number 4`, the xad file, taking note of the value of its ETag from the response:

```
$ aws s3api upload-part --profile=zadara \
--endpoint-url=https://vsa-00000001-cloud-01.zadara.com \
--bucket mybucket \
--key "partsExample/largeFile.dat" \
--part-number 4 \
--body xad \
--upload-id OWM1YtdlOWUtODd1MC00NGY1LTk3ZWYtOWNiNTE2ZjBjODJj
```

Response:

```
{
  "ETag": "\"ab246aba52b0c3e78ee3be1f43843d3b\""
}
```

7. Edit and save a JSON file, ensuring that it includes the ETag value returned from the responses from each `upload-part` of the multipart object.

In this example, the file is named `parts.json` containing the following:

```
{
  "Parts": [
    { "ETag": "6eac0b608eee111876931ddbd6907429", "PartNumber":1 },
    { "ETag": "aaffcae340799e25223970903f917cd6", "PartNumber":2 },
    { "ETag": "d6d7687673a728c8dc5683ce8ada395f", "PartNumber":3 },
    { "ETag": "ab246aba52b0c3e78ee3be1f43843d3b", "PartNumber":4 }
  ]
}
```

8. Complete the multipart upload, using the `parts.json` file to specify the file parts reassembly.

```
$ aws s3api complete-multipart-upload \
--profile=zadara \
--endpoint-url=https://vsa-00000001-cloud-01.zadara.com \
--bucket mybucket \
--key "partsExample/largeFile.dat" \
--upload-id OWM1YtdlOWUtODd1MC00NGY1LTk3ZWYtOWNiNTE2ZjBjODJj \
--multipart-upload file://parts.json
```

Response:

```
{
  "Location": "https://vsa-00000103-public-zadara-qa19.zadarazios.com/tmpstest/partsExample/
↩largeFile.dat",
  "Bucket": "mybucket",
  "Key": "partsExample/largeFile.dat",
  "ETag": "\"2e0b93de0eb3f28c6734b864dd157ccd-4\""
}
```

9. Use the `head-object` command to return information on the uploaded multipart object:

```
$ aws s3api head-object --profile=zadara \
--endpoint-url=https://vsa-00000001-cloud-01.zadara.com \
--bucket mybucket \
--key "partsExample/largeFile.dat"
--part-number 2
```

Response:

```
{
  "AcceptRanges": "bytes",
  "LastModified": "Mon, 10 Feb 2025 07:35:18 GMT",
  "ContentLength": 209715200,
  "ETag": "\"2e0b93de0eb3f28c6734b864dd157ccd-4\"",
  "ContentType": "application/octet-stream",
  "Metadata": {},
  "PartsCount": 4
}
```

Included in the response above:

- "ContentLength": 209715200:

The size of this part (--part-number 2) is 200 MB (209715200 bytes), matching the size of the original uploaded part 2, xab.

- "PartsCount": 4:

The total number of parts in the uploaded multipart object.

5.1.6 boto3 python library

Update the default/create new profile for the Object Storage within aws configuration file.

~/.aws/config

```
[profile zadara]
s3 =
  signature_version = s3v4
```

✓ **Note:** It is possible to use both AWS v4/v2 signatures with S3-compatible storage such as Zadara Object Storage.

~/.aws/credentials

```
[zadara]
aws_access_key_id = <S3 Access Key>
aws_secret_access_key = <S3 Secret Key>
```

The credentials can be retrieved from the Object Storage logged in “User Information” properties.

In your python code:

```
#!/usr/bin/env python

import boto3

session = boto3.session.Session(profile_name='zadara')
```

(continues on next page)

(continued from previous page)

```
s3_client = session.client(
    service_name='s3',
    region_name='us-east-1',
    endpoint_url='https://vsa-00000001-cloud-01.zadara.com',
)

print('Buckets')
print(s3_client.list_buckets())

print('')

print('Objects')
print(s3_client.list_objects(Bucket='test'))
```

✓ Note:

- `profile_name` is the name of the credentials and config profile specified above (in this case, “zadara”)
 - `endpoint_url` is the HTTPS path to the Object Storage being accessed
 - `region` should match the Region defined in the Object Storage [settings page](#) (Zadara default: `us-east-1`)
-

5.1.7 AWS S3 Java SDK (`aws-java-sdk`)

AWS Provides a comprehensive S3 Java SDK that can be used with Zadara’s Object Storage. Getting started guide is available in Zadara’s Support Knowledge Base article - [How to use AWS S3 Java SDK with Object Storage](#).

5.1.8 AWS S3 PHP SDK (`aws-sdk-php`)

AWS Provides a comprehensive S3 PHP SDK that can be used with Zadara’s VPSA Object Storage. Getting started guide is available in Zadara’s Support Knowledge Base article - [How to use AWS S3 PHP SDK with Object Storage](#).

5.1.9 AWS S3 JavaScript SDK (`aws-sdk`)

AWS Provides a comprehensive S3 JavaScript SDK that can be used with Zadara’s VPSA Object Storage. Getting started guide is available in Zadara’s Support Knowledge Base article - [How to use AWS S3 JavaScript SDK with Object Storage](#).

5.2 Openstack Swift Interface

The management interface generates a new Swift API token upon login. This means that if you logout and login again you’ll notice a new token. The token presented by the management interface is always the latest and valid to be used.

The API tokens created by the management interface are generated based on the Object Storage global configuration for token validity (default: 24 hours).

Example of validating an API token using the CLI:

```

# Get two consecutive API tokens from the management interface and store it
$ TOKEN1=gAAAAABiuwu4P55M2V...
$ TOKEN2=gAAAAABiuwvc8BEYc8...

$ curl -X GET -H "Content-Type: application/json" \
  -H "X-Access-Key: $TOKEN1" \
  "https://<object storage endpoint>:8443/api/zios/accounts/AUTH_<account ID>/users.json"

{"response":{"users":[],"count":0,"status":0}}%

curl -X GET -H "Content-Type: application/json" \
  -H "X-Access-Key: $TOKEN2" \
  "https://<object storage endpoint>:8443/api/zios/accounts/AUTH_<account ID>/users.json"

{"response":{"users":[],"count":0,"status":0}}%

# We can validate these tokens using the Openstack Keystone auth service as well:

$ curl -s -H "X-Subject-Token: $TOKEN2" -H "X-Auth-Token: $TOKEN1" \
  "https://<object storage endpoint>:5000/v3/auth/tokens" | python3 -m json.tool
{
  "token": {
    "is_domain": false,
    "methods": [
      "password"
    ],
    "roles": [
      {
        "id": "fedeff6db6df47959e96d8dd33963cfe",
        "name": "zios_admin"
      }
    ],
    "expires_at": "2022-06-29T14:10:36.000000Z",
    ....
    "issued_at": "2022-06-28T14:10:36.000000Z"
  }
}

```

Important: By default, the API token is valid for 24 hours. the preferred option to identify/renew the API token via an API call is to use a Keystone authentication request and not using Object Storage command indicated in the [Zadara Object Storage REST API Guide](#). Example for authentication against the Keystone service is provided in the next section.

5.2.1 cURL (swift API)

cURL can be used for Object Storage operations. The connectivity information is available in the User Information view.

Authentication

S3 Access Key:	12345le7373d42218cb96f8f8c612345
S3 Secret Key:	*****
Region:	us-east-1
API Token:	gAAAAABilyf6cEXTLuo8nVjRi8agmFIVqYmON1uWib8-ISGfGj8U-IXAsCttFOK8hmlJpEjbbeQyKwzclSLNmPIcR344GQMF6vmuFZ1aK8MQZcNT2fPIBiKRBrw72vc8G_rg2cyuzHVurqmOuW

▲ Connectivity - Front End Network

API Endpoint:	https://vsa-00000004-zadara-iop-01.zadarazios.com:443
V3 Auth Endpoint:	https://vsa-00000004-zadara-iop-01.zadarazios.com:5000/v3
Account URL:	https://vsa-00000004-public-zadara-iop-01.zadarazios.com:443/v1/AUTH123456789

▲ Connectivity - Public Network

Public IP:	192.168.13.180
Public API Endpoint:	vsa-00000004-public-zadara-iop-01.zadarazios.com
Public V3 Auth Endpoint:	https://vsa-00000004-public-zadara-iop-01.zadarazios.com:5000/v3
Public Account URL:	https://vsa-00000004-public-zadara-iop-01.zadarazios.com:443/v1/AUTH_123456789

In this example, we will use the **Front End Network Account URL** or **Public API Network Account URL**, and **API Token** in order to create a new container:

```
$ curl -H "x-auth-token: <user_token>" -X PUT <account_url>/test-bucket/
```

For example:

```
$ URL=<Front End Network Account URL or Public API Network Account URL>
$ TOKEN=<MYAPI TOKEN>
$ curl -H "x-auth-token: $TOKEN" -X PUT $URL/test-bucket/
```

The following example describes how to get the token programmatically using the Swift API:

```
$ curl -i -H "Content-Type: application/json" \
-d '{ "auth": \
{ "identity": { "methods": ["password"], "password": \
{ "user": { "name": "<USERNAME>", "domain": { "id": "default" }, \
"password": "<USER PASSWORD>" } } }, "scope": { "project": \
{ "name": "<ACCOUNT_NAME>", "domain": { "id": "default" } } } }' \
"https://vsa-00000001-mycloud-01.zadara.com:5000/v3/auth/tokens" ;
```

and use the returned token for the subsequent API calls.

```
HTTP/1.1 201 Created
Date: Thu, 19 Nov 2020 16:05:28 GMT
Server: Apache/2.4.29 (Ubuntu)
```

(continues on next page)

(continued from previous page)

```
Content-Length: 1114
X-Subject-Token: gAAAAABftpfIAiuo2tRZP8VvtomU1knVG7xNU0NV4b2u...
```

Presigned URL for POST

The following example uses the boto3 S3 Client `generate_presigned_post` to build the URL and form fields used for a presigned S3 POST.

1. Create a python script that uses boto3 functions to presign a URL for POST.

In this example, the python script is `post_presign.py`.

```
# post_presign.py

import boto3

s3_client = boto3.client(
    's3',
    aws_access_key_id='<S3 Access Key>',
    aws_secret_access_key='S3 Secret Key',
    endpoint_url='https://vsa-00000103-public-zadara-qa19.zadarazios.com',
    region_name='us-east-1',
    config=boto3.session.Config(signature_version='s3v4')
)
url = s3_client.generate_presigned_post(
    Bucket='Container1',
    Key='obj1',
    ExpiresIn=3600
)

print(url)
```

2. Run the python script to presign the URL for POST:

```
python3 post_presign.py
```

Response:

```
{
  'url': 'https://vsa-00000103-public-zadara-qa19.zadarazios.com:443/Container1',
  'fields': {
    'key': 'obj1',
    'x-amz-algorithm': 'AWS4-HMAC-SHA256',
    'x-amz-credential_request',
    'x-amz-date': '20250217T055444Z',
    'policy':
↵ 'eyJleHBpcmF0aW9uIjogIjIwMjU0MjE3T055444Z',
↵
    }
}
```

3. Using cURL, upload the object to the presigned URL, using the parameter values returned in the response from the script:

```
curl -i 'https://vsa-00000103-public-zadara-qa19.zadarazios.com:443/Container1' \
-F 'key=obj1' -F 'x-amz-algorithm=AWS4-HMAC-SHA256' \
```

(continues on next page)

(continued from previous page)

```

-F 'st-1/s3/aws4_request' \
-F 'x-amz-date=20250217T055444Z' \
-F
↪ 'policy=eyJleHBpcmF0aW9uIjogIjIwMjU0MDItMTdUMDY6NTQ6NDRaIiwgImNvbmRpdGlbnMiOiBbeyJidWnrZXQiOiAidG1wNC1ITUFDLVNIQTI
↪ '

```

4. Verify the upload by listing the objects in the container:

```

aws s3api list-objects \
--bucket Container1 \
--profile zadara
--endpoint https://vsa-00000103-public-zadara-qa19.zadarazios.com:443

```

Response:

```

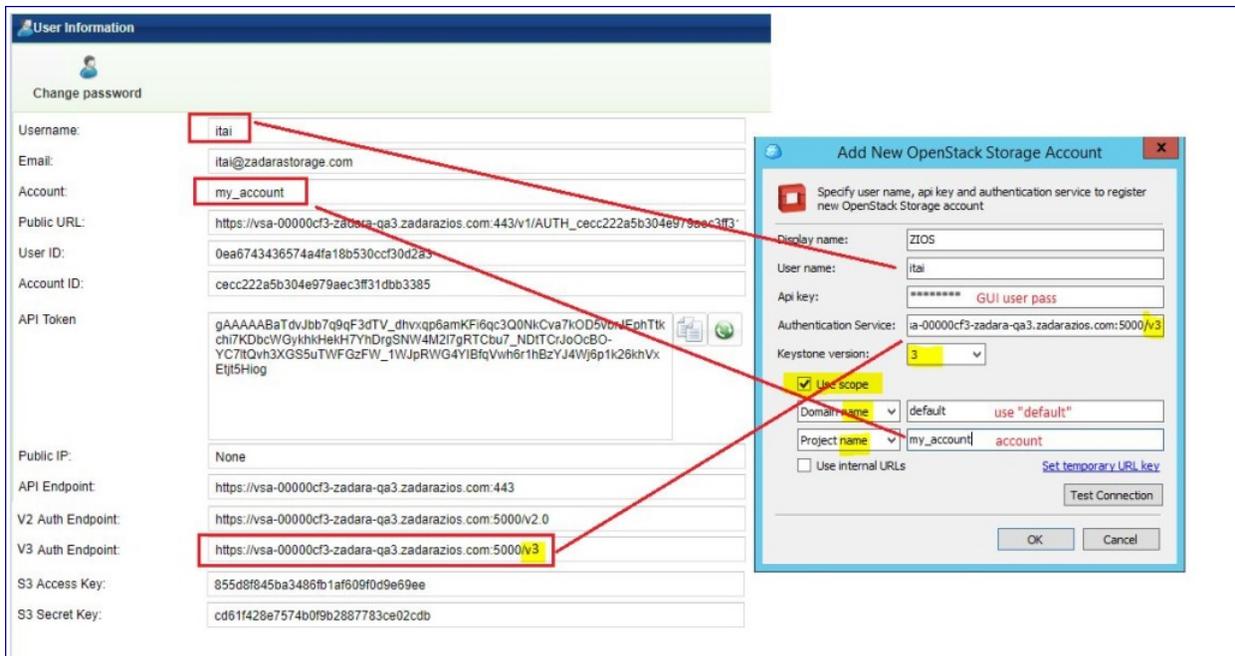
{
  "Contents": [
    {
      "Key": "obj1",
      "LastModified": "2025-02-17T05:57:43.000Z",
      "ETag": "\"d41d8cd98f00b204e9800998ecf8427e\"",
      "Size": 65959,
      "StorageClass": "STANDARD",
      "Owner": {
        "DisplayName": "<NGOS account:username>",
        "ID": "<NGOS account:username>"
      }
    }
  ],
  "RequestCharged": null,
  "Prefix": ""
}

```

Additional examples of using the Openstack Swift API can be found at the [Openstack Swift API documentation](#).

5.2.2 Cloudberry Explorer for OpenStack (v3 authentication)

Use the logged-in User Information properties to set the authentication fields of Cloudberry Explorer



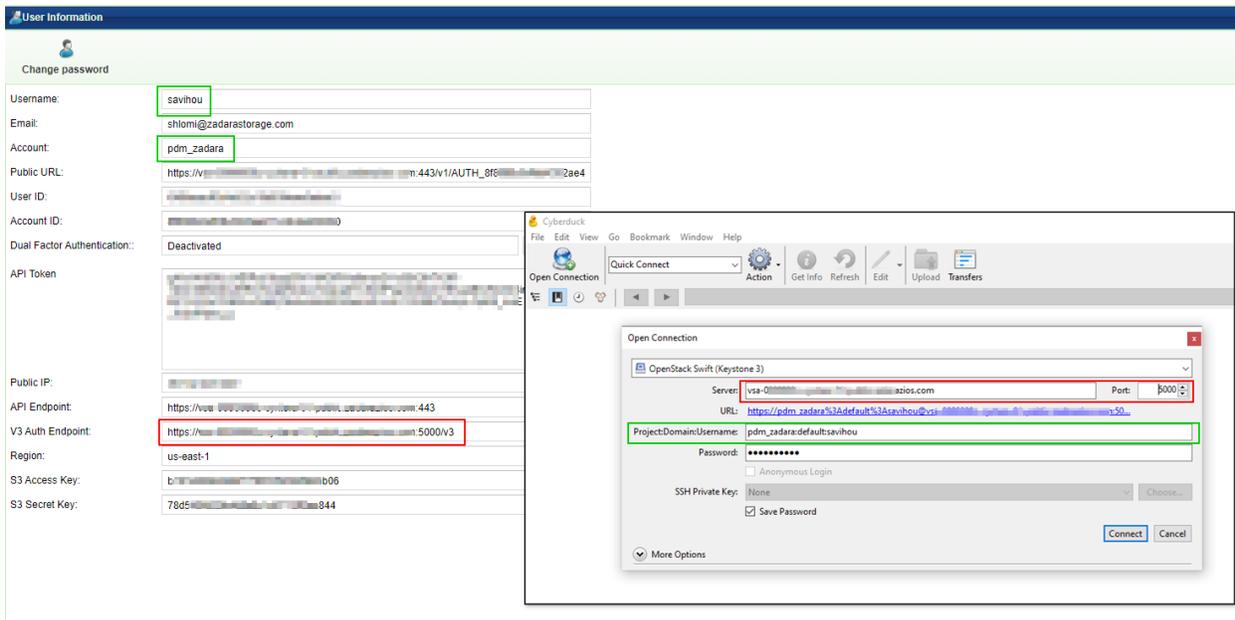
5.2.3 CyberDuck

Cyberduck version: 7.7.1 (33788)

Cyberduck client support “Openstack Swift (Keystone 3)” API interface.

Use the logged-in User Information properties to set the authentication field of CyberDuck client.

1. Server - the Object Storage v3 Auth Endpoint.
2. Port - 5000
3. Project:Domain:Username - <Object Storage Account>:default:<Object Storage Username>



MAIN DASHBOARD

Object Storage administrators and Object Storage account administrators have dedicated dashboard views that provides a “snapshot” of the current status, activities and trends in their scope (global array and account level respectively).

Refer to the relevant dashboard according to your Object Storage role:

- [Object Storage Administrator dashboard](#)
- [Account Administrator dashboard](#)

6.1 Object Storage administrator dashboard

Scope: Object Storage Administrator

The Object Storage dashboard allows administrators to get the current health, stats and capacity information at a glance. The dashboard is visible to the Object Storage administrator (`zios_admin` account) only.

The dashboard has 5 sections:

1. **General information** - lists general Object Storage information such as:
 - **Name** - Object Storage display name as provided during its creation process
 - **Region** - the region attribute defined by the administrator (default: `us-east-1`), used primarily for AWSv4 signatures
 - **Load Balancer Type** - Internal (default) or ZELB (Zadara Elastic Load Balancer)
 - **Management IP** - the frontend network IP address assigned to the Object Storage and used as the access endpoint. The DNS name registered automatically will match this IP address.
 - **Internal name** - the cloud's UUID
 - **Cloud name** - the Zadara zStorage cloud name
 - **Deployment type** - `Single Region` or `Multi-Zone`
 - **Management URL** - the DNS name for the Object Storage
 - **Version** - the Object Storage software version
 - **Public IP** - the public IP assigned to the Object Storage (optional)
 - **Creation date** - the Object Storage creation date
2. **Performance** - breakdown of the Object Storage's current performance metrics:
 - **Total Throughput** - the total throughput being served currently by the Object Storage controllers

- **Total OP/s** - object operations per second, aggregated value for all operations served by the object storage controllers
- **OP/s** - breakdown of object operations per type, listing the “common” operations as PUT/GET/DELETE.

✓ **Note:** For a more detailed breakdown use the Performance section in the management section

- **Active Connections** - the current number of sessions connected to the Object Storage
3. **System health** - object storage system inventory and status:
 - **Virtual Controller** - VC storage and proxy headcounts for the Object Storage, followed by average storage CPU usage and average proxy CPU usage as percentages
 - **Drives** - data drive count
 - **Accounts** - active account count
 - **Containers** - containers (buckets) count across all system accounts
 - **Objects** - object count across all object storage accounts/containers
 - **Average Object Size** - the average object size within the account
 - **New pending requests** - account creation requests that are pending for administrator approval
 4. **Capacity** – This chart shows the accumulated used capacity from the data storage policies over time. The bar shows the current used/free capacity along with a historical trend of the used capacity.
 5. **Policies** – Lists all storage policies with their health index as calculated by the system.

✓ **Note:** The object count/capacity information undergoes periodic refreshment to reflect the latest statistics stored in the system. However, since the updates occur once an hour, the information may not always present real-time values.

6.2 Account administrator dashboard

Scope: Account Administrator

Similar to the Object Storage administrator, the Object Storage dashboard allows account administrators to get the overall status of the system at a glance. The dashboard is visible to the Object Storage account administrators.

The dashboard has 5 sections:

1. **General information** - lists general Object Storage information such as:
 - **Account Name** - the Object Storage account name as provided during its creation process
 - **Version** - the Object Storage software version
 - **Deployment type** - **Single Region** or **Multi-Zone**
 - **Public management URL** - the public endpoint assigned to the Object Storage (optional)
 - **Name** - Object Storage display name as provided during the creation process
 - **Region** - the region attribute defined by the administrator (default: **us-east-1**), used primarily for AWSv4 signatures
 - **Management URL** - the DNS name for the Object Storage
 - **Internal name** - the cloud's UUID

- **Account creation date** - the object storage account creation date
2. **Performance** - breakdown of the Object Storage's current performance metrics:
 - **Total Throughput** - the total throughput being served currently by the Object Storage controllers
 - **Total OP/s** - object operations per second, aggregated value for all operations served by the object storage controllers
 - **OP/s** - breakdown of object operations per type, listing the "common" operations as PUT/GET/DELETE.
 3. **Account status** - account level snapshot:
 - **Total capacity** - total used capacity for the account
 - **Containers** - containers (bucket) count
 - **Objects** - object count across all containers within the account
 - **Average object size** - the average object size within the account
 - **Users** - account user count
 - **Quota status** - indication whether a capacity quota was enabled by the Object Storage administrator, and the current used capacity
 4. **Capacity** - capacity usage trend
 5. **Top containers** - list of the top containers (sorted by capacity)

✓ **Note:** The object count/capacity information undergoes periodic refreshment to reflect the latest statistics stored in the system. However, since the updates occur once an hour, the information may not always present real-time values.

RESOURCES MANAGEMENT

Scope: Object Storage Administrator

7.1 Monitoring Drives

To monitor drives in your Object Storage system, open **Resources > Drives**.

7.1.1 Viewing Drives Properties

The Drives details (properties and metering), are shown in the South Panel tabs:

Properties

Each Drive includes the following properties:

Property	Description
ID	An internally assigned unique ID
Name	Automatically assigned name.
Capacity	The Drive capacity in GiB
Storage Node	The Storage Node that contains the selected Drive
Virtual Controller	The virtual controller that owns the selected drives and performs IO operations on it
Storage Policy	The Storage Policy where the selected Drive belongs
Fault Domain	The Zadara cloud Fault Domain this Drive resides belongs to
Protection Zone	The Zadara cloud protection zone this drive is physically located at
Type	Drive type: SATA, SAS, SSD
UUID	The unique identifier of the drive
Status	<ul style="list-style-type: none">• Normal – All drives are in sync• Failed – The drive does not function• Absent – The drive does not exist
Added	Date & time when the drive was added
Modified	Date & time when the drive was last modified

Disk Metering

The Metering Charts provide live metering of the IO workload associated with the selected Drive.

The charts display the last captured performance samples. An interval length can be one of the following: 10 second, 1 minute, 10 minutes, 1 hour, 1 day, 1 week.

The **Auto** button allows to switch to continuously-update live metering info.

The following charts are displayed:

Chart	Description
IOPs	The number of read and write commands issued to the selected Drive per second
Bandwidth (MB/s)	Total throughput (in MB) of read and write commands issued to the selected Drive per second
Latency (ms)	Average response time of all read and write commands issued to the selected Drive per selected interval

Backend Metering

The following charts are displayed:

Chart	Description
Throughput (OP/s)	The number of operations (PUT/GET/DELETE) that were sent to the selected Drive per second
Bandwidth (MB/s)	Total throughput (in MB) of read and write commands that were sent to the selected Drive per second
Latency (ms)	Average response time of all operations (PUT/GET/DELETE) that were sent to the selected Drive per selected interval

7.2 Monitoring Virtual Controllers

Virtual Controllers are Virtual Machines running on the Zadara zStorage cloud that serve client operations on the Object Storage. For a full list of the VC responsibilities refer to [Virtual Controller](#). Virtual Controllers are automatically created and added/removed to the Object Storage configuration, depending on the number of the allocated drives. In case the workload have changed and additional performance are required - the object storage owner can add Proxy only VCs from the Zadara Provisioning Portal as described in [Adding Proxy Virtual Controllers](#).

7.2.1 Viewing VCs Properties

Properties

Each Virtual Controller has the following properties:

Property	Description
ID	An internally assigned unique ID
Storage Role	Proxy+Storage / Proxy-Only
Management Role	<ul style="list-style-type: none"> • Ring Master - Runs the Object Storage Rings • Ring Slave - Standby to run the Object Storage Rings • VC - Regular Object Storage VC
Status	<ul style="list-style-type: none"> • Created - VC is running normally • Failed - VC is not running • Passivating - VC is shutting down • Deleting - in the process of being removed from the cluster
Storage Node	The Storage Node hosting selected VC
Fault Domain	The Zadara cloud Fault Domain this VC resides belongs to
Frontend IP	The IPv4 or IPv6 address allocated to the VC
Backend IP	The VC IP address on the backend network that connects to the Drives
Added	Date & time when the VC was added
Modified	Date & time when the VC was last modified

Drives

List the drives assigned to the selected Storage Policy.

Virtual Networks

A set of available IP addresses within a specific network segment. Virtual networks are allocated for a specific cloud tenant and within a specific available cloud VLAN.

The following properties are displayed for each virtual network:

Property	Description
Virtual Network In-Interface (VNI)	The virtual adapter that connects a device to the virtual network. This interface enables network communication within the virtual environment.
IP	The IP address assigned to the VNI.
Network Mask	The IP definition that defines the number of available host addresses.
Gateway	The IP address of the network node that routes traffic from the virtual network to external networks.
VLAN ID	The VLAN to which the VNI belongs.

System Usage

This chart shows the CPU utilization of the selected VC.

Backend Metering

The Metering Charts provide live metering of the IO workload at the backend of the selected VC.

The charts display the metering data as it was captured in the past 20 intervals. An interval length can be one of the following: 10 second, 1 minute, 10 minutes, or 1 hour, 1 day, 1 week. The **Auto** button lets you see continuously-updating live metering info.

The following charts are displayed:

Chart	Description
IOPS	The number of operations (PUT/GET/DELETE) issued to objects and handled by the selected VC per second
Bandwidth (MB/s)	Total throughput (in MB) of read and write commands issued by the selected VC per second
Latency (ms)	Average response time of all operations (PUT/GET/DELETE) issued to objects and handled by the selected VC per selected interval

Account Metadata Metering

The Metering Charts provide live metering of the IO workload on the accounts database at the backend of the selected VC.

The charts display the metering data as it was captured in the past 20 intervals. An interval length can be one of the following: 10 second, 1 minute, 10 minutes, or 1 hour, 1 day, 1 week. The **Auto** button lets you see continuously-updating live metering info.

The following charts are displayed:

Chart	Description
IOPS	The number of operations (PUT/GET/DELETE) issued to the accounts database and handled by the selected VC per second
Latency (ms)	Average response time of all operations (PUT/GET/DELETE) issued to the accounts database and handled by the selected VC per selected interval

Container Metadata Metering

The Metering Charts provide live metering of the IO workload on the containers database at the backend of the selected VC.

The charts display the metering data as it was captured in the past 20 intervals. An interval length can be one of the following: 10 second, 1 minute, 10 minutes, or 1 hour, 1 day, 1 week. The **Auto** button lets you see continuously-updating live metering info.

The following charts are displayed:

Chart	Description
IOPS	The number of operations (PUT/GET/DELETE) issued to containers and handled by the selected VC per second
Latency (ms)	Average response time of all operations (PUT/GET/DELETE) issued to containers and handled by the selected VC per selected interval

Object Metadata Metering

The Metering Charts provide live metering of the IO workload on the object database at the backend of the selected VC .

The charts display the metering data as it was captured in the past 20 intervals. An interval length can be one of the following: 10 second, 1 minute, 10 minutes, or 1 hour, 1 day, 1 week. The **Auto** button lets you see continuously-updating live metering info.

The following charts are displayed:

Chart	Description
IOPS	The number of operations (PUT/GET/DELETE) issued to objects and handled by the selected VC per second
Latency (ms)	Average response time of all operations (PUT/GET/DELETE) issued to objects and handled by the selected VC per selected interval

Frontend Metering

The Metering Charts provide live metering of the IO workload at the frontend of the selected VC.

The charts display the metering data as it was captured in the past 20 intervals. An interval length can be one of the following: 10 second, 1 minute, 10 minutes, or 1 hour, 1 day, 1 week. The **Auto** button lets you see continuously-updating live metering info.

The following charts are displayed:

Chart	Description
Throughput (OP/s)	The number of operations (PUT/GET/DELETE) issued to objects and handled by the proxy of the selected VC per second
Bandwidth (MB/s)	Total throughput (in MB) of read and write commands issued to proxy of the selected VC per second
Latency (ms)	Average response time of all operations (PUT/GET/DELETE) issued to objects and handled by proxy of the selected VC per selected interval

7.3 Managing Storage Policies

The Storage Policy provide a way for object storage providers to differentiate service levels, features and behaviors of an Object Storage deployment.

Policies can be thought of as a group of drives, with a redundancy level policy assigned to it.

Before placing object data into the Object Storage, users create a container which holds the listing of all objects stored under the container's namespace. Users can select the Storage Policy that will be used when storing data objects under a container's namespace when they create the container. All objects stored in a container will be placed according the configuration of the Storage Policy which was set upon the Object Storage creation.

To ensure availability of the Object Storage data, the drives assigned to a Storage Policy are evenly distributed between Object Storage Fault Domains. The cloud administrator defines the Fault Domain of each Storage Node. The system allocates drives across zones, based on the Storage Policy type.

Storage Policies allow objects to be stored based on the following criteria:

- Quality of Service: By using different disk drives for different policies, tiers of storage performance can be created. For example, an SSD-only policy can be created and used to implement a low-latency/high performance tier.
- Number of Replicas:
 - 2 Way replication offers protection for one FD failure, at the cost of 50% storage utilization.
 - Erasure Coding (4+2) offers protection for one FD failure, at the cost of 67% storage utilization.

The following Storage Policies are supported:

Table 1: Object Storage Data Protection Policies

Policy Type	Redundancy	Minimal Configuration
2 Way	x 2	2 Storage VCs on 2 SNs
Erasure Coding 4+2	x 1.5	3 Storage VCs on 3 SNs

Each drive in the system is assigned to one Storage Policy.

Object Storage is created with a default data Storage Policy for objects, and another system Storage Policy for metadata. The Object Storage administrator (zios_admin) can later expand/shrink the storage policy based on their needs.

7.3.1 Storage Policies

When the Object Storage system is created, 2 policies are created by default:

- MetadataPolicy: Used to store the Accounts and Containers' metadata
- 2-way/EC-protection-policy: Used to store the users' objects, usually contains all the drives that were assigned to the Object Storage at creation time, with 2-way replication or EC protection, according to the initial selection on the provisioning portal.

7.3.2 Policies Properties

You can view the following properties and metering information in the Policies Details south panel tabs:

Properties

Each Policy includes the following properties:

Property	Description
General	
ID	An internally assigned unique ID
Name	The name of policy
Type	Object or Account/Container
State	Not Configured / Configuring / Initialized
Description	A user defined policy description
Ring Version	Ring Database version
Default	Yes/No
Redundancy Level	2-way/EC
Policy Capacity	Total usable capacity of the storage policy
Policy Used Capacity	Total capacity used in the storage policy
Data Usage	Amount of written data in the storage policy by the Object Storage account's users
Containers Count	Total amount of containers created within the policy
Objects Count	Total amount of objects created within the policy
GB per Month Price	Price of used capacity for charge back purposes
Added	The date and time when the policy was added
Modified	The date and time when the policy was last modified
Policy Health	
Health Status	Normal / Degraded / Critical
Balance	Indicates the progress of the rebalance process
Rebalance Paused	Yes/No
Last Data Rebalance	The date and time of the last data rebalance
Last Ring Rebalance	The date and time of the last ring rebalance

✔ **Note:** Please note that **Policy Used Capacity** attribute will always reflect higher capacity consumption than **Data Usage**. The **Policy Used Capacity** also include the underlying object allocated metadata (4K) and the actual object storage allocation (the minimal allocation unit is 4K). **Data Usage** is the actual account's data usage representation.

Drives

List the drives assigned with the selected Storage Policy.

Capacity Metering

The Metering Charts provide live metering of the capacity usage associated with the selected Policy.

The charts display the metering data as it was captured in the past 20 intervals. An interval length can be one of the following: 10 minutes, or 1 hour, 1 day, 1 week. The **Auto** button lets you see continuously-updating live metering info.

The following charts are displayed:

Chart	Description
Used Capacity	Total storage capacity consumed in the selected policy
Containers	Total numbers of containers that store their objects in the selected policy
Objects	Total numbers of objects stored in the selected policy

Backend Metering

The Metering Charts provide live metering of the IO workload associated with the selected Policy.

The charts display the metering data as it was captured in the past 20 intervals. An interval length can be one of the following: 10 second, 1 minute, 10 minutes, or 1 hour, 1 day, 1 week. The **Auto** button lets you see continuously-updating

live metering info.

The following charts are displayed:

Chart	Description
Throughput (OP/s)	The number of operations (PUT/GET/DELETE) issued to the Drives of the selected policy per second
Bandwidth (MB/s)	Total throughput (in MB) of read and write commands issued to the Drives of selected policy per second
Avg. Drive Latency	Average response time of all operations (PUT/GET/DELETE) issued to objects in the selected policy per selected interval

Frontend Metering

The Metering Charts provide live metering of the IO workload associated with the traffic coming to the selected Policy.

The charts display the metering data as it was captured in the past 20 intervals. An interval length can be one of the following: 10 second, 1 minute, 10 minutes, or 1 hour, 1 day, 1 week. The **Auto** button lets you see continuously-updating live metering info.

The following charts are displayed:

Chart	Description
Throughput (OP/s)	The number of operations (PUT/GET/DELETE) issued to objects in the selected policy per second
Bandwidth (MB/s)	Total throughput (in MB) of read and write commands issued to the selected policy per second
Avg. Latency (ms)	Average response time of all operations (PUT/GET/DELETE) issued to objects in the selected policy per selected interval

Capacity Alerts

The Object Storage administrator can set their own custom configuration for capacity monitoring.

Alert Threshold

Send alert when it is estimated that the policy will be at full capacity within the given time period (default 43,200 minutes - 30 days).

Alert Interval

The period of the sample collection window to calculate the rate at which the free space is consumed in the storage policy.

- Default 1,440 minutes
- Minimal value is 10 minutes
- Minimum used capacity to trigger the alert is 60%

Emergency Threshold

The threshold to trigger administrator capacity emergency state notifications

- Default is 85%
- Minimal value is 60%
- Alert will be triggered hourly as long as the policy is in "Emergency state"

7.3.3 Adding Drives to Policy

Drives are added to an Object Storage policy via the Provisioning Portal. To add drives into a policy, go to the Provisioning Portal, select the Object Storage of interest and click **Add Storage**. Follow the instruction here: [Adding drives](#).

✓ **Note:** Drive-related operations in a storage policy will require rebalance that might take several hours/days until completion.

7.3.4 Removing Drives from Policy

If there is a need to reduce the total available capacity of a given policy, or to remove some failed drives that were detached from the policy, you may remove drives from the policy and return them to the cloud for a different use. To remove drives from a policy, go to **GUI > Policies**, select the policy of interest, and click **Remove Drives**.

The dialog that opens will list all the drives types and quantities that currently belong to the policy. The system support removing a pair of drives in a single operation. In case a larger volumes of drives needs to be removed please contact Zadara's support.

✓ **Note:** Drive-related operations in a storage policy will require rebalance that might take several hours until completion.

ACCOUNTS AND USERS

8.1 Managing Accounts

Object Storage Accounts are a collection of containers and are typically associated with a tenant. Object Storage Account Management allows you to view/configure account properties, permissions, and storage usage, and see lists of users associated with the account.

8.1.1 Creating an account

Scope: Object Storage Administrator

When the system is first built, a default account is created, called `zios_admin`. At that point only the Object Storage Admin has access to this account. In order to provision Object Storage to customers, the Object Storage Admin needs to create accounts.

To create additional accounts, first select the **Accounts** entity in the Main Navigation Panel (left panel) under **Account Management**, and then click the **Create** button in the top toolbar above the account pane.

In the dialog that opens, give a name to the new account and click **Add**. The new account will be added.

✓ **Note:** An account name can comprise only the following characters, or any combination of them up to a maximum of 128 characters in length:

- Uppercase and lowercase English letters (A-Z, a-z)
- Numbers (0-9)
- . period
- _ underscore
- + plus
- - dash/minus
- @ at

An account cannot contain spaces, other special characters and other language letters.

8.1.2 Accounts Properties

Scope: Object Storage Administrator Account Administrator

- **Properties** - the following account properties are displayed in the account pane in the **Account Management > Account** view.

✓ **Note:** Parameters marked with (*) in table below are only available to Object Storage Administrators.

Property	Description
ID	An internally assigned unique ID
Name	The name of the account
Status (*)	Normal / Deleting / Deleted, awaiting cleanup
Enabled (*)	Yes/No
Public URL	The URL that identifies this account. To be used by the REST API
Containers	Number of containers in the selected account
Objects	Number of objects stored in the selected account
Used Capacity	Amount of written data in the account
Policies	Show statistics per each policy (e.g. 2-way protection) used by this account. Details include: <ul style="list-style-type: none"> - Containers: Number of containers this account keeps in this policy - Objects: Number of objects this account keeps in this policy - Used Capacity: Capacity consumed by this account, kept in this policy

- **Permissions** - account permissions are displayed in the details pane, permission tab in the **Account Management > Account** view. For more information on account permissions, see [Setting Account Permissions](#).
- **Users** - lists of users per account are displayed in the users pane in the **Account Management > Users** view, and in the Users tab in the **Account Management > Account** view.
- **Capacity Metering** - provide live metering of the capacity usage associated with the selected account.

The charts display the metering data as it was captured in the past 20 intervals. An interval length can be one of the following: 10 minutes, or 1 hour, 1 day, 1 week. The **Refresh** button forces a refresh of the data displayed in the graphs. The **Auto** button lets you see continuously-updating live metering info.

The following charts are displayed:

Chart	Description
Used Capacity	Total storage capacity consumed in the selected account
Containers	Total numbers of containers belonging to the selected account, by storage policy
Objects	Total numbers of objects belonging to the selected account, by storage policy

- **Frontend Metering** - provide live metering of the IO workload at the Object Storage frontend associated with the selected account.

The charts display the metering data as it was captured in the past 20 intervals. An interval length can be one of the following: 10 second, 1 minute, 10 minutes, or 1 hour, 1 day, 1 week. The **Refresh** button forces a refresh of the data displayed in the graphs. The **Auto** button lets you see continuously-updating live metering info.

The following charts are displayed:

Chart	Description
Throughput (OP/s)	The number of operations (PUT/GET/DELETE) issued to objects that belong to the selected account.
Bandwidth (MB/s)	Total throughput (in MB) of read and write commands issued to proxy for the selected account.
Latency (ms)	Average response time of all operations (PUT/GET/DELETE) issued to objects of the selected Account per selected interval.

- **Account Metering** - provide live metering of the IO workload at the Object Storage frontend associated with the selected account.

The charts display the metering data as it was captured in the past 20 intervals. An interval length can be one of the following: 10 second, 1 minute, 10 minutes, or 1 hour, 1 day, 1 week. The **Refresh** button forces a refresh of the data displayed in the graphs. The **Auto** button lets you see continuously-updating live metering info.

The following charts are displayed:

Chart	Description
Throughput (OP/s)	The number of operations (PUT/GET/DELETE) issued to objects that belong to the selected account.
Latency (ms)	Average response time of all operations (PUT/GET/DELETE) issued to objects of the selected Account per selected interval.

- **Container Metering** - provide live metering of the IO workload at the Object Storage frontend associated with the selected account.

The charts display the metering data as it was captured in the past 20 intervals. An interval length can be one of the following: 10 second, 1 minute, 10 minutes, or 1 hour, 1 day, 1 week. The **Refresh** button forces a refresh of the data displayed in the graphs. The **Auto** button lets you see continuously-updating live metering info.

The following charts are displayed:

Chart	Description
Throughput (OP/s)	The number of operations (PUT/GET/DELETE) issued to objects that belong to the selected account.
Latency (ms)	Average response time of all operations (PUT/GET/DELETE) issued to objects of the selected Account per selected interval.

8.1.3 Account Quota Management

Version: 23.09

Scope: Object Storage Administrator Account Administrator

Quotas are a useful way to control capacity consumption on a specific account or container.

Capacity quotas can be set:

- Per container by the Account Administrator
- Globally per account by the Object Storage Administrator

✓ **Note:** The sum of actual usage capacities of all the containers in an account are tracked, so that cumulatively they do not exceed the account's quota.

For purposes such as future planning, it is also possible to specify container quotas such that their sum or even an individual container's quota can be higher than the account quota. Although it is possible to specify higher quotas at container level, the system will prevent consumption of extra storage when the account quota has been reached.

Configurations are available for alert notifications when the quota's warning, emergency and 100% utilization thresholds are reached:

- Quota alerts to Object Storage Administrator: see [Quota Alerts](#) on the Settings page.
 - Quota alerts to Account Administrator: see [Account Administrator Quota Alerts](#).
-

✓ **Note:** Once enabled, it will take up to 10 minutes for the quota management to be activated.

Account Level Quota Management

Scope: Object Storage Administrator

1. Navigate to **Account Management > Accounts**.
 2. In the top pane select the desired account, and open the **Quotas** tab in the bottom **Details** pane.
 3. Mark the **Enable capacity quota** checkbox.
 4. Enter the **Capacity (GiB)** quota. The minimum is 1 GiB.
 5. Click **Update**.
-

✓ **Note:**

- When the quota is enabled, the actual **Used capacity (GiB)** also displays in the same tab.
 - In the **Account Management > Accounts > Quotas** tab, an Account Administrator cannot configure the account's capacity quota, but can view:
 - Whether the capacity quota feature is enabled or disabled for the account.
 - If enabled, the capacity quota and used capacity amounts.
-

Account Administrator Quota Alerts

Scope: Account Administrator

Quota alerts to the Object Storage Administrator are configured in the account's Settings. See [Quota Alerts](#) on the Settings page.

By default, alert notifications are not sent to the Account Administrator.

To configure the system to issue alert notifications to the Account Administrator when the quota's warning, emergency and 100% utilization thresholds are reached:

1. Navigate to **Account Management > Accounts**.
 2. In the bottom account details pane, open the **Quota Alerts** tab.
-

3. Mark the **Notify the account administrator(s) with quota alerts** checkbox.
4. Select the **Alert frequency** option to determine notification repetition on reaching a quota alert threshold:
 - **Single alert** (default) notification without further repetition, when the usage capacity reaches the threshold.
 - **Once a day**, for as long as the usage capacity reaches the threshold, repeat the notification alert.
5. Click **Update**.

8.1.4 Deleting an account

Scope: Object Storage Administrator

To delete an account, navigate to **Account Management > Account**, select the account to be deleted, and click **Delete** in the top toolbar.

 **Note:**

- Deleting an account is an irreversible operation, and requires double confirmation
 - Once an account is deleted, all account user data is removed. However account billing information still exists in the system for usage report generation. Click **Cleanup** in top toolbar to completely remove it from the system.
-

8.1.5 Disabling an account

Scope: Object Storage Administrator

To disable an account, navigate to **Account Management > Account**, select the account to be deleted, and click **Disable** in the top toolbar.

 **Note:** Once an account is disabled, the account is no longer available for read or write operations. However, Object Storage maintains the account entities (users, access rights, etc.), as well as all the containers and objects.

8.1.6 Self Service Account Creation

Scope: Account Administrator

In addition to creation of a new account by the Object Storage administrator as described in [Creating an account](#), a user can be given permission to create his own account. In this case, a user will request creation of a new account via a provided URL. The Object Storage Admin will receive and must then approve the request. The account will then be created and the user who initiated the request will be set as the Account Administrator.

The detailed procedure for account self-creation is as follows:

1. Use the GUI URL received from Object Storage Admin to access the login screen.
2. On the login screen, click **Create new account**. In the overlay that displays, enter the following information:
 - Name for the new account
 - Your username as the Account Admin
 - Your email address
 - Select a password

✓ **Note:** While account name and the username for a given user are unique across the Object Storage, the same email address can be used for multiple users. This is useful in cases the same entity needs visibility to more than a single account.

3. Click **Create Account**. This will create an account creation request that will go to the Object Storage Admin for approval. Once approved, You will automatically become the Account Admin of your new account.
4. The user initiating the request will receive an automated email response confirming the request.
5. The Object Storage Admin will receive an email informing about the pending request:
6. The Object Storage Admin should open the GUI, select **Users** in the Main Navigation Panel (Left Panel) under **Account Management**, select the pending account request, and either **Approve** or **Deny** it.
7. Upon approval, the new account will be created, the account admin will be defined with the given credentials, and receive an email notification with the following information:
 - Object Storage Account Management & Console URL
 - Object Storage API Endpoint URL
 - Account Name
 - User Name

8.2 Managing Users

8.2.1 Understanding User Roles

The Object Storage supports the following roles:

- **Object Storage Admin** - responsible for the administration of the Object Storage. This is the user that created the VPSA Object in the Zadara Provisioning Portal.
- **Object Storage Admin - Read Only** - dedicated read-only role for cross-accounts monitoring and reporting purposes. **The Read-Only role is available for the zios_admin account only.** Read-Only users will have access to the Object Storage RestAPI, however they will not have data access. The user role is designated for monitoring and reporting purposes, such as:
 - Performance monitoring
 - Capacity monitoring
 - Usage reports and billing automation
- **Account Administrators** - responsible for the administration of their accounts.
- **Account Member** - can perform Object Storage operations according to the given permissions within the limits of that account.

8.2.2 User Information

Information about the logged-in user of the current session is displayed by clicking the user name in the upper right corner of the GUI.

Some of the displayed properties have optional actions associated with them, such as viewing, copying and resetting passwords.

The following User's properties are displayed:

Property	Description
Account Information	
Username	The login ID of the User
Email	User's email address
Account	The account where the user belongs
User ID	An internally assigned unique ID
Account ID	An internally assigned unique ID
Dual Factor Authentication	Indicates if this user has dual factor authentication activated. Option to activate/deactivate dual factor authentication.
Authentication	
S3 Access Key	To be used by client using the S3 interface Option to copy the access key to the clipboard.
S3 Secret Key	To be used by client using the S3 interface Options to view the key, copy it to the clipboard, or reset it.
Region	Region name
API Token	Token to be used for authentication by the REST API The token expires in 24 hours. Good practice is for every script to start with a new token. See API guide: http://zios-api.zadarastorage.com Options to view the token, copy it to the clipboard, or reset it.
Connectivity - Front End Network	
API End-point	The effective Front End private address for REST API for all IO requests
V3 Auth Endpoint	The effective Front End private address for REST API auth requests
Account URL	The Front End private network URL that identifies this user's account. To be used by the REST API.
Connectivity - Public Network	
Public IP	Public IP of the Object Storage (see: Assigning Public IPs)
Public API endpoint	The public address for REST API for all IO requests
Public V3 Auth Endpoint	The public address for REST API auth requests
Public Account URL	The public network URL that identifies this user's account. To be used by the REST API

 **Note:** Connected users can reset their Object Storage Access/Secret keys. The existing access and secret keys will be revoked.

8.2.3 Creating a User

Scope: Object Storage Administrator Account Administrator

To create a new user in an Object Storage account:

1. In the Object Storage console, navigate to **Account Management > Users**.
2. From the top toolbar on the Users pane, click **Create**.
3. In the **Add User** dialog which opens, enter the following:

- **Username**

A Username can comprise only the following characters, or any combination of them up to a maximum of 128 characters in length:

- Uppercase and lowercase English letters (A-Z, a-z)
- Numbers (0-9)
- . period
- _ underscore
- + plus
- - dash/minus
- @ at

A Username cannot contain spaces, other special characters and other language letters.

- **Email**
- **Role**

✓ **Note:** Everything an Account admin does, is within the context of that Account. So, when an Account admin creates users, there is no need to select an Account.

✓ **Note:** Users with Object Storage Admin role can only be created in the zios_admin account.

4. Click **Add User**. The new user will receive an email with the following information:
 - Object Storage Account Management & Console URL
 - Object Storage API Endpoint URL
 - Account Name
 - User Name
 - Assigned User Role
 - Temporary Password

✓ **Note:** The new user should use the temporary password for the first login, and then change the password after logging on.

8.2.4 Viewing Users Properties

Scope: Object Storage Administrator Account Administrator

To view user properties in an Object Storage account:

1. In the Object Storage console, navigate to **Account Management > Users**. User properties are displayed in the top pane of the console.
2. To view additional properties in the lower details pane, select a single user from the displayed list in the top pane.

The following user properties are displayed:

Property	Description
Name	The login ID of the User
Email	User's email address
ID	An internally assigned unique ID
Account Name	The account where the user belongs
Account ID	An internally assigned unique ID
Role	Object Storage Admin, Account Admin, Member
Locked	Indicates if the user is locked and blocked from access
Notify on Events	Object Storage Administrator can specify for themselves whether to receive notifications for events Option to activate/deactivate
Dual Factor Authentication	Indication if this user has dual factor authentication activated
Enabled	User is active or not. A disabled user can't login and can't perform any operation.

8.2.5 Deleting users

Scope: Object Storage Administrator Account Administrator

To delete a user in an Object Storage account:

1. In the Object Storage console, navigate to **Account Management > Users**.
2. From the displayed list, select the user to be deleted and click **Delete** from the top toolbar.
3. In the **Confirm Deletion** dialog which opens, click **Yes**. Note that the deletion process may take a few minutes.

8.2.6 Disabling/Enabling users

Scope: Object Storage Administrator Account Administrator

A disabled user cannot log in to the GUI or perform any operation via the REST API. However the system remembers the user with all the properties and permissions. Once users are enabled, they can resume operations as before.

To disable a user in an Object Storage account:

1. In the Object Storage console, navigate to **Account Management > Users**.
2. From the displayed list, select the user to be disabled and click **Disable** from the top toolbar.
3. In the **Confirm Action** dialog which opens, click **Yes**. Note that the process may take a few minutes.

✓ **Note:** To enable a user who has been disabled, repeat the process above and select **Enable** from the toolbar instead of **Disable**.

8.2.7 Reset password

Scope: Object Storage Administrator Account Administrator

Object Storage Admins and Account Admins can reset users' passwords. When resetting a password, the user will receive an email with a temporary password that they will have to change at the next login.

To reset a user password in an Object Storage account:

1. In the Object Storage console, navigate to **Account Management > Users**.
2. From the displayed list, select the user whose password is to be reset and click **Reset Password** from the top toolbar.
3. In the **Confirm Password Reset** dialog which opens, click **Yes**.
4. The user will receive an email with a temporary password.

✓ **Note:** Users who have forgotten their password do not need to refer to the admin to reset their password. They can click the **Forgot Password** link on the login screen.

8.2.8 Change Role

Scope: Object Storage Administrator Account Administrator

An Account Member can be changed to an Account Admin, and vice versa. Users that are members of the system zios_admin account can be promoted to Object Storage Admin only by someone who currently has the Object Storage Admin role.

To change a user role in an Object Storage account:

1. In the Object Storage console, navigate to **Account Management > Users**.
2. From the displayed list, select the user whose role is to be changed, and click **Change Role** from the top toolbar.
3. In the **Change Role** dialog which opens, enter the new user role and click **Change Roles**.

8.3 Dual Factor Authentication

It is a common practice to protect access in cases of compromised passwords. For this purpose, the Object Storage supports Dual Factor Authentication using a mobile Authenticator application. Each user can turn Dual Factor Authentication on or off. The Object Storage Admin can force Dual Factor Authentication on all users.

To use Dual Factor Authentication, install a mobile Authenticator app (e.g. Google Authenticator) from Google Play or Apple AppStore on your mobile device.

Important: If the Object Storage administrator requires Dual Factor Authentication to be set for all Object Storage accounts, all system users must enable Dual Factor Authentication for their account in the next login. This setting cannot be disabled for a specific user.

8.3.1 Enabling Dual Factor Authentication

1. In the Object Storage console, click on user name on top, right corner of screen. Current user property details will be displayed.
2. For **Dual Factor Authentication**, click **Activate** or **Deactivate**. Close the properties dialog, and logout.
3. The next time you login, a confirmation screen will open with a QR code. Scan the code with your mobile device, and enter the token.
4. From now on, during every login, you will be asked to enter the Dual Factor Authentication token from the Authenticator app on your mobile device.

Important: The mobile device that runs the Authenticator app is needed for login. If the device was lost or replaced, the user must ask the Object Storage Admin to reset their Dual Factor Authentication settings. The Object Storage Admin must contact Zadara support to reset the Dual Factor Authentication.

8.3.2 Enforcing Dual Factor Authentication

The Object Storage Admin can force Dual Factor Authentication for all users. In **setting/Security** click **Edit** on Dual Factor Authentication, select the checkbox and **Save**. This setting change does not have immediate effect. The next time each user will login, the Dual Factor Authentication token from the mobile device's Authenticator app be required.

✔ **Note:** When MFA enforcement is removed, the users with Dual Factor Authentication configured are still required to use the temporary code when logging in. However each user can change their settings in the user properties as described above.

MANAGING PERMISSIONS

9.1 Understanding Permissions

NextGen Object Storage provides 2 levels of permissions: Account and Container.

Both permissions types are enforced on account members only, account admins always have all permissions.

Account-level permissions enforce Read (listing) and Write (creating/deleting) option for **containers** under an account.

Container-level permissions enforce Read (list/download) and Write (upload/delete) options for **objects** under a container.

Default Permissions:

An account is created with default account-level permissions that allow all account members to list/create/delete containers in the account.

After an account is created, account-level permissions can be set by a NextGen Object Storage Admin or an Account Admin.

After a container is created, container-level permissions can be set by a NextGen Object Storage Admin or an Account Admin.

9.2 Setting Account Permissions

Scope: Account Administrator

Account-level permissions are set in the **Account** view Details pane, and can be set globally for all users in an account or per-user.

The screenshot shows the ZADARA VPSA OBJECT STORAGE interface. The 'ACCOUNT' tab is selected, and the 'Details for public_OSA_58' pane is visible. The 'Permissions' tab is active, showing a table with columns for User, List Containers, Create/Delete Containers, Read Objects, and Write/Delete Objects. A red box highlights the 'ADD' button, and a red arrow points to the 'All Members' row in the table. Another red arrow points to the 'All Members' row in the 'Set Permissions' dialog box. The dialog box shows a table with columns for User, List Containers, Create/Delete Containers, Read Objects, and Write/Delete Objects, with a 'Save' button at the bottom.

- For global account permissions, check the desired permissions for **All Members** in the details pane at the bottom of the Accounts view and then click **Save**.
- For per-user account permissions, click **Add** in the toolbar on top of the Details pane. In the **Set Permissions** dialogue which opens, individually select the desired permissions per user and then click **Save**.

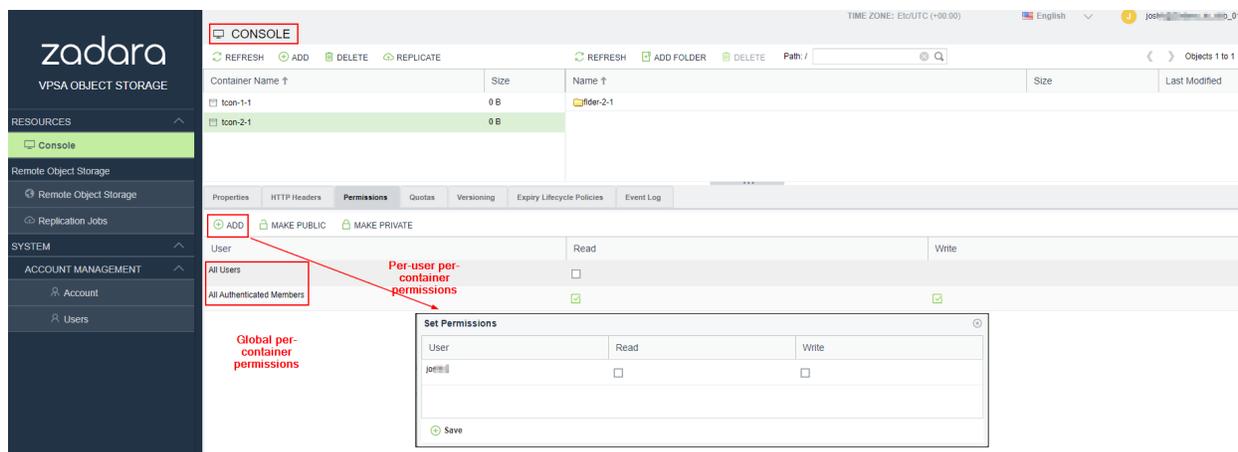
✓ Note:

- When setting permission per-user, existing global permission settings are removed.
- When setting global permissions, existing per-user permissions are removed.

9.2.1 Setting Container Permissions

Scope: Account Administrator

Container-level permissions are set in the **Console** view Details pane, and can be set globally for all users in an account or per-user.



- For global container permissions, select the desired container in the Container pane at the top of the Console view. Then select the desired permissions in the Details pane in the **All Users** row or **All Authenticated Members** row and click **Save**.
- For per-user container permissions, click **Add** in the toolbar on top of the Details pane. In the **Set Permissions** dialogue which opens, individually select the desired permissions per user and then click **Save**.

✓ Note:

- When setting permission per-user, existing global permission settings are removed.
- When setting global permissions, existing per-user permissions are removed.

✓ **Note:** By making a container public (**Make Public/Private** button) any user can list this container's objects (using "referral" API) even without permissions for this container.

9.2.2 Cross-Origin Resource Sharing (CORS)

Version: 23.09-SP1

Scope: Account Administrator

Cross-Origin Resource Sharing (CORS) is a security standard that enables servers to specify safe origins from which browsers are allowed to request resources.

CORS is an extension based on the Same-Origin Policy (SOP) that browsers use to prevent malicious applications from accessing sensitive data on domains beyond their control. SOP is a browser security feature that restricts how resources can be accessed by different web applications. It requires that a resource must be from the same origin as the web application that is attempting to access it.

SOP

For two origins to be considered the same, they must have the same domain and URI scheme. If a port is specified, then the ports of both origins must match. The URL path and query components are irrelevant for SOP purposes.

Example of SOP compliance:

- Origin 1: `https://example.com/example1/index.html`
- Origin 2: `https://support.example.com/faqs.html`

In this example, the scheme is `https` and the domain is `example.com`, and both are the same for both origins.

Example of SOP non-compliance:

- Origin 1: `https://example.com/example1/index.html`
- Origin 2: `http://example.com/example1/index.html`

Although the domain is identical to Origin 1, the `http` scheme differs from `https`.

- Origin 3: `http://example.com:1030/example1/index.html`

Although the scheme and domain are identical to Origin 2, the specified port differs from no port specified in Origin 2.

- Origin 4: `http://examples.com:1030/example1/index.html`

Although the scheme and port are identical to Origin 3, the domain `examples.com` differs from `example.com`.

CORS

CORS enables client-side web applications that are loaded in one domain to interact with resources in a different domain. Using CORS support, you can build client-side web applications and specify the permitted cross-origin access to your resources.

To configure a bucket to allow cross-origin requests, add a CORS configuration to the bucket. See [CORS configuration examples](#).

The CORS configuration is a set of rules on a bucket that:

- Identifies the origins that are allowed access to the bucket
- The HTTP methods (operations) supported for each origin, together with operation-specific information

USAGE REPORTS

Usage per account is captured in 30-minute intervals, and is stored for 3 months. The system converts these captured usage data points into a sum of the capacity used during this time period, into a GiB/month format.

A NextGen Object Storage administrator (zios_admin) can create a report with all billing metering information, and export the data into any billing system used. This report uses the pricing information that you have set, as described in the [Pricing settings](#) options and in the [Storage Policies](#) wizard.

To create a Usage Report:

1. In the left navigation menu, select the **System > Usage Reports**.
2. In the main form:

1. **Account:**

- From the dropdown list, select the **Account** for which you want to create the report, or **All** (default) to create a report for all accounts.

2. **Report:**

- Select the reporting period:

- **Monthly:** From the dropdown, select the current calendar month (default), or one of the two previous calendar months.
- **Custom:** Enter a custom period for the report, with a start date from up to 2 months ago, and the end date up to the current date.

3. Click **Generate Report**.

A high-level summary of the report will be displayed. The report can be exported to JSON or CSV format with finer granularity.

The report output is classified according to the following metering services:

1. Capacity - the protected capacity per month (or custom time frame). The protected capacity usage information is the average consumption on the Object Storage policy throughout the selected report time frame (month/custom). The report shouldn't be considered as a capacity snapshot view, as the protected capacity utilization as presented in the report is an average usage sampled on a daily basis. In order to review and generate a current capacity utilization report, the NextGen Object Storage administrator should use the "Export List" option in the **Account Management > Accounts** view.
2. Data In - summary of all ingress traffic (GiB)
3. Data Out - summary of all egress traffic (GiB)
4. Container Replication - summary of all egress traffic used by the internal Container Replication functionality.

10.1 Usage Reports - Exporting a Summary Report

The exported “Summary Report” includes a high-level report, with the same granularity as presented in the management GUI.

For the CSV option, the exported report archive includes two CSV files:

1. Report header - the general information for the Object Storage and the account such as:
 - NextGen Object Storage ID
 - NextGen Object Storage name and URL
 - NextGen Object Storage Version
 - Pricing information
 - Reporting interval
2. Usage Summary - the actual usage report information:
 - Billing units
 - Billing sub-category (incoming_bytes, outgoing_bytes and used capacity)
 - Container & Object count
 - Account information

The JSON option will include all of the above information in a single JSON object.

10.2 Usage Reports - Exporting a Detailed Report

The exported “Detailed Report” include a finer resolution report, that can assist the NextGen Object Storage administrator to break down the usage report to its building block during the requested time frame.

For the CSV option, the exported report archive include two CSV files:

1. Report header - the general information for the Object Storage and the account such as:
 - NextGen Object Storage ID
 - NextGen Object Storage name and URL
 - NextGen Object Storage Version
 - Pricing information
 - Reporting interval
2. Usage - the actual usage report information, with an hour by hour service breakdown:
 - Billing units
 - Billing sub-category (incoming_bytes, outgoing_bytes and used capacity)
 - Container & Object count
 - Account information

LOGS

Scope: Object Storage Administrator

11.1 Access Log

Access log lists all operations done by any user, either using the GUI or the REST API. Each operation is listed with all given parameters.

The list can be filtered by:

- User who took the action
- Action type (e.g. create account)
- Date and time

11.2 Events Log

The events log lists all the events reported by the system. The list can be filtered by:

- Message contains - lists only events that contain the given string
- Min severity - lists only events at the given severity level and more severe

The **Advanced Mode** provides additional options:

- Doesn't contain - excludes events in the given string
- Date - lists events in a date and time range
- Source Type - lists events from a selected source type, e.g. Disk, Storage Policy, Controller

PERFORMANCE MONITORING

12.1 Understanding Performance Monitoring

Scope: Object Storage Administrator

This section contains high-level overview to monitoring the storage performance. The Object Storage Performance Monitor allows you to check and monitor the behavior of each element that can affect the overall storage performance, from the single drive/user/account/controller to the whole Object Storage system.

Each element of the data path can impact the overall performance if not configured and operates properly. The Object Storage performance Monitor is a tool for pinpointing a storage performance bottlenecks. The following metrics are of interest to measure the performance of a storage system:

- **Bandwidth (Throughput):** This value is how much read or write throughput a certain Resource (disk, policy etc...) delivers. Usually expressed in Megabytes/Second (MB/s)
- **IOPS:** IO operations per second, which means the amount of HTTP operations done in one seconds interval. A certain amount of IO operations will also give a certain throughput of Megabytes each second, so these two are related.

Average Object Size x IOPS = Throughput

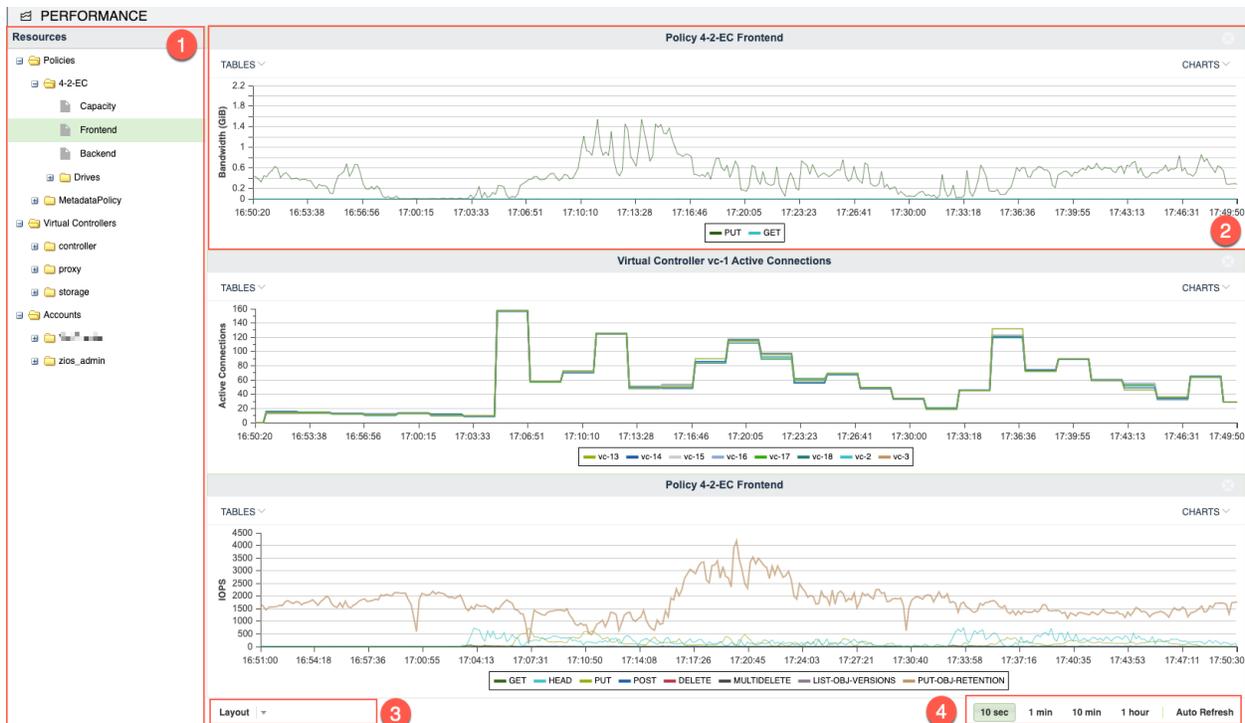
- **Response time (Latency):** is the time it takes each IO operation to complete. Latency is measured in milliseconds (ms) and should be as low as possible.

Important: The latency figures presented for the front-end traffic present the end-to-end latency as seen by the client. In some cases, high latency can be seen due to the distance of the client from the object storage service.

12.2 The Performance Monitor

To open the VPSA Performance Monitor, in the left menu navigate to **System > Performance** section.

The performance monitor view consists of the following elements:



- Resources Tree:** The Resources Tree lists all the data path objects currently exist in the NextGen Object Storage:
 - Policies (data policy and metadata policy including their respective drives)
 - Virtual Controllers (grouped by their type)
 - Accounts breakdown - Users
- Resource Tile:** The Performance Monitor has 1 to 9 resource tiles depending on the chosen layout. Each tile contain either a table or a chart.
- Layout Selector:** Toggles between number of supported layout with different number of tiles.
- Interval Selector:** Allows switching between different intervals. The interval is a sampling period. Each interval is a single point in the chart. This point represents the average value during that interval.

For examples: If 1 minute interval was selected 60 points are displayed, each one is the average value for that specific minute. In total the last 1 hour is displayed.

The interval selection affects all tiles.

12.3 Customizing the Performance Monitor

12.3.1 Customizing the Layout

- Go to **VPSA GUI > Performance** and click the **Layout** selector
- Select the layout of your choice. Note that if the selected layout has fewer tiles than the original the other tiles will be lost, and should be set again.
- Drag the object of interest from the resources tree, and drop it into a tile. Do the same for all tiles.

12.3.2 Customizing a Tile

Each tile represent a single resource, and provide number of display options related to the specific resource. The display can be either a table of the most current performance figures, or a chart over time of the recent history.

- To display a chart click the **Charts** button on the top right corner of the tile, and select the metric of interest.

SETTINGS

Scope: Object Storage Administrator

The settings view is visible to administrators of the `zios_admin` account. These collections of settings are system-wide settings:

1. General & Connectivity
2. Security
3. Pricing
4. Network

13.1 General & Connectivity settings

13.1.1 Allow Tenant Name In URL

Allow specifying the tenant name (account name) in the URL passed in the API instead of its ID. (Default: No)

Example (account ID):

```
$ wget https://vsa-00000001-mycloud-01.zadara.com/v1/AUTH_8f9388c6dfdb4352ae411e3b4e655850/my-website/cat.png
```

Example (account name):

```
$ wget https://vsa-00000001-mycloud-01.zadara.com/v1/AUTH_webhosting/my-website/cat.png
```

13.1.2 Region

For AWS v4 signature, “region” (also called `bucket_location`) must be specified for the signature mechanism to work. (Default: `us-east-1`).

✓ **Note:** The default value of the region setting was changed in Object Storage version 20.12 from `US` to `us-east-1`. Object Storages that were created prior to that version will not inherit the new region setting automatically.

The region settings in the S3 compatible object storage clients and the Object Storage should be identical.

13.1.3 API Error Alerts

The API Error Alert provides the ability to enable alerts for failed API requests (HTTP Codes 400, 403, 408, 500, 502, 503, 504) and the threshold for such alerts.

Default Status: Enabled Default Threshold: 1

13.1.4 Containers Virtual-Hosted Style Supported

While virtual-hosted style access is disabled by default, the Object Storage supports both path-style and virtual-hosted style.

In a virtual-hosted-style request, the container name is part of the domain name in the URL. Zadara's Object Storage uses the following format:

```
https://<container-name>.<object storage id>-<cloud-id>.zadara.com/<key>
```

Example of virtual-hosted style URL:

```
https://office-images.vsa-00000001-mycloud-01.zadara.com/building.png
```

In a path-style URL, the container name will be used as part of the logical path of the URL, as in the following format:

```
https://<object storage id>-<cloud-id>.zadara.com/<container-name>/<key>
```

Example of path style URL:

```
https://vsa-00000001-mycloud-01.zadara.com/office-images/building.png
```

Important: Using Virtual-Hosted style access requires a proper DNS registration and matching SSL certificates, which are handled automatically by the Object Storage engine. However, if the Object Storage uses a custom SSL certificate and API hostname, the Object Storage administrator is required to ensure the compatibility of their certificates and DNS registration.

(Default: disabled)

13.1.5 Welcome message user information

New members and account administrators are provided with connectivity details post registration to the system. The connectivity details are sent via email to the email address attached to their account.

As the object storage supports multiple network interfaces the object storage administrator can decide which network(s) information would be shared with their new users.

(Default: Front-End network)

13.1.6 Quota Alerts

Capacity usage limits can be configured as quotas per account. See [Account Quota Management](#).

The Object Storage Administrator can receive alert notifications when reaching an account quota's warning, emergency and 100% utilization thresholds.

- **Notify zios_admin with quota alerts:**

Enable or disable the quota alert notification service for the Object Storage Administrator.

Default: Yes (enabled)

✓ **Note:** The **System > Account Management > Users > Object Storage Administrator > Notify On Events** property must also be enabled for the Object Storage Administrator to receive quota notification alerts. See **Notify On Events** in [Viewing Users Properties](#).

- **Quota Warning Alert Threshold:**

The percentage of quota utilization that triggers a warning notification.

Default: 75%

- **Quota Emergency Alert Threshold:**

The percentage of quota utilization that triggers an emergency notification.

Default: 90%

- **Alert frequency:**

The options for repeated notifications on reaching a threshold:

- **Single alert** (default)
- **Once a day**

See [Account Administrator Quota Alerts](#) to configure the system to issue alert notifications to the Account Administrator when the quota's warning, emergency and 100% utilization thresholds are reached.

13.1.7 Connectivity Settings

Each consumer facing network interface is presented in this section (grouped by the network type). This section allows the admin to adjust the API hostname if a custom domain name is needed.

The Object Storage is provisioned with the Front End network interface and Public IP. Additional network interfaces can be assigned to the Object Storage.

Once additional network interfaces are assigned, their connectivity information is listed.

Front End network

- **Public IP:** (read only)

An IP address that allows access to the Object Storage system from the public internet. Assigning a Public IP is done via the Zadara Provisioning Portal, as described in [Assigning Public IPs](#).

- **API Endpoint:** (read only)

The effective API endpoint address for Object Storage REST API for all IO requests.

- **Auth (authentication) Endpoint:** (read only)

The effective address for Object Storage API for authentication requests. The authentication endpoint value is derived from the API hostname.

Starting from version 19.08 the default supported authentication for Openstack Swift client is Keystone v3 authentication.

Important: The support Keystone v2 authentication was deprecated.

- **API IP:**

The IP address of the Objects Storage host's internal API Hostname, allowing access to the Object Storage system from within the local internal network only.

- **API Hostname:**

Object Storage FQDN (fully qualified domain name), accessible within the local internal network only.

 **Note:** For the Object Storage API Hostname either static IP, or FQDN must be given.

- **Floating FE IP:** (read only)

The floating frontend IP address used by the Object Storage.

- **Proxy VC IP:** (read only)

The Object Storage Virtual Controllers IP frontend addresses.

Public network

Front End Network settings limit access to the Object Storage within the local internal network only.

Public Network settings enable access to the the Object Storage from anywhere on the public internet.

- **Public IP:** (read only)

The IP address that allows access to the Object Storage system from the public internet. Assigning a Public IP is done via the Zadara Provisioning Portal, as described in [Assigning Public IPs](#).

- **Public API Hostname:**

The public Object Storage FQDN (fully qualified domain name), accessible from anywhere.

- **Public API Endpoint:**

The public API endpoint address for Object Storage REST API for all IO requests.

- **Public Auth Endpoint:**

The IP address for Object Storage API authentication requests from the public internet. The authentication endpoint's value is derived from the Public API Hostname.

13.2 Security settings

13.2.1 Passwords Policy

The Object Storage Administrator can control the VPSA Password expiration policy and password history policy.

(Default: disabled)

13.2.2 Dual Factor Authentication

Enforce Dual Factor Authentication for all users. Once enabled, the Object Storage users will be required to set MFA.

(Default: disabled)

13.2.3 Cloud Admin Access

This sets the ability to access the cloud administrator Object Storage management interface (via Command Center).

(Default: enabled)

13.2.4 Upload SSL Certificate (Optional)

The Object Storage REST API works over HTTPS with an SSL certificate. Object Storage defaults to its built-in SSL certificate (issued for zadara.com domain). If the Object Storage administrator wants to use their own certificate, upload it in this section. The supported certificate format is "PEM". SSL "PEM" certificate format, as defined in RFCs 1421 through 1424, is a concatenated certificate container file. It is expected that the Object Storage administrator will append the private-key to the certificate prior to uploading it.

The resulting PEM should look like this:

```
-----BEGIN RSA PRIVATE KEY-----
(Your Private Key: your_domain_name.key)
-----END RSA PRIVATE KEY-----
-----BEGIN CERTIFICATE-----
(Your Primary SSL certificate: your_domain_name.crt)
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
(Your Intermediate certificate: Intermediate.crt)
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
(Your Root certificate: RootCertificate.crt)
-----END CERTIFICATE-----
```

Important: When uploading a new SSL certificate, the Object Storage endpoints, API Hostname and API IP specified in both the [Front End network](#) and [Public network](#) sections in the [General & Connectivity settings](#) tab (**System > Settings > General & Connectivity**) must be updated, so that they comply with the new certificate.

13.2.5 Encryption

This sets the encryption password for the Object Storage data-at-rest encryption.

For more information on encrypted containers see [Encrypted Containers](#) .

13.2.6 Swift Token Expiration

Swift token expiration can be set manually, default is one day (1440 minutes).

13.2.7 SSL Termination

The Object Storage defaults to HTTPS clients connectivity. The SSL termination is conducted by the internal load balancer. However, if an external load balancer is used in-front of the Object Storage, SSL termination can be set to `external` which will assume HTTP traffic between the external load balancer and the Object Storage.

(Default: internal)

13.3 Pricing settings

Currency:

Select the currency used for billing purposes. Supported currencies are:

1. USD - USA Dollar
2. GBP - Great Britain Pound
3. EUR - Euro
4. AUD - Australia Dollar
5. KRW - South Korea Won
6. JPY - Japan Yen
7. CNY - China Yuan

Data Transfer Pricing:

If you want to charge your internal/external customers for the traffic going into and from Object Storage, you can specify your currency and pricing in the Setting>Pricing tab.

<policy name> policy price:

Pricing for stored capacity depends on the storage policy used. Therefore the capacity price is set per policy as the price per GB per month. If multiple data policies exist, a different pricing can be configured for each data policy.

13.4 Network settings

13.4.1 FE MTU Size

Modify the MTU size for the Frontend interface.

Select 1500 (Default), 2048, 4096, 9000 from the dropdown, or overwrite the displayed value with a custom MTU size in the range of 1420 - 9000.

13.4.2 Public MTU Size

Modify the MTU size for the Public interface (1500 - Default, 2048, 4096, 9000)

13.4.3 Load Balancer Mode

Toggle the internal load balancer & Zadara Elastic Load Balancer mode of operation:

- **Direct Server Return (default)** - Recommended for scale. Packets from the Object Storage virtual controller bypass the load balancer, maximizing the egress throughput.
- **NAT** - The load balancer will be used as a gateway for all traffic from /to the Object Storage virtual controller.



Warning: Changing the Load Balancer mode of operation can be disruptive for existing clients workload.

13.4.4 Custom DNS Servers

A custom (private) DNS server can be set to allow proper name resolution of private domain names, this setting is useful while working with a Remote Authentication Provider.

- Custom name servers name server IP, comma separated
- DNS lookup domain (optional) - set the explicit domain name that will be searched using the custom name server

NETWORK DIAGNOSTICS

This view allows the Object Storage administrator to perform connectivity checks from within the Object Storage itself spanning its servers/networking devices.

Interface: Select the source interface of the Object Storage (Frontend, Public IP)

Diagnostics Type: Select either:

- **Tcpdump**
- **Reachability**
 - **Target Address:** IPv4 (or IPv6) of the target network device/server.
 - Select at least one of the following:
 - * **Ping:** Checkbox - Perform a ping test (count - number of echo requests to send).
 - * **Traceroute:** Checkbox - Perform a traceroute scan to the target host, and define its TTL (Time to Live: the number of times the packet can be rebroadcast by the next host encountered on the network, or hops).

LOAD BALANCING

Load balancing refers to distributing incoming traffic across a group of backend server. In order to serve a high volume of concurrent requests from users or clients it is required to have a “traffic distributor” sitting in front of the backend servers and routing requests across all workers capable of fulfilling those requests.

The load balance is responsible for the following functions:

- Distributes client requests or network load efficiently across multiple backend servers
- Ensures high availability and reliability by sending requests only to servers that are online and can handle the request in a timely manner
- Provides the flexibility to scale-up or scale-down as demand dictates

In the context of the Object Storage, a Proxy Virtual Controller is referred as a backend server.

15.1 How load is balanced in the Object Storage?

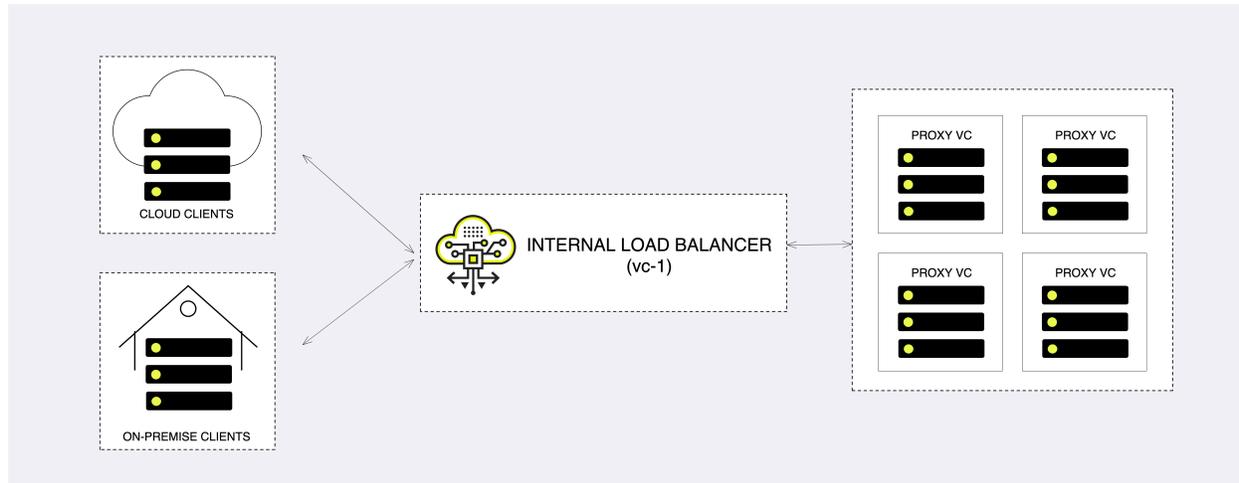
The NextGen Object Storage load-balancer has two modes of operation:

15.1.1 Internal load balancer (default)

Any NextGen Object Storage is provisioned with a built-in Load Balancer (referred as Internal Load Balancer) solution which will work seamlessly out of the box. The internal load balancer is fully configured to provide a secured, reliable and highly-available endpoint. Upon provisioning a new Object Storage instance, the system will be have a globally unique FQDN matching the Object Storage front-end network, along with a matching well-known CA certificate that will be used for a secured communication over HTTPS.

In case a public IP/additional VNI (Virtual Network Interface) is allocated to the object storage, the internal load balancer will be used to distribute the workload. The main difference between Frontend network and Public/VNI handling is the load balancer mode of operation:

- Frontend - DRS (Direct Server Return), packets from the Object Storage Virtual Controller bypass the Load Balancer, maximizing egress throughput.
- Public/VNI - The load balancer will be used as a gateway for all traffic from/to the Object Storage Virtual Controller.



Load balancing algorithm

The load balancer is configured to use weighted least connection (wlc) - new connections go to the worker with the least number of connections. Reminder - Controller VCs & Storage VCs are excluded from handling client/users operations.

High availability

As all other core services in the system, the internal load balancer is highly available. The internal load-balancer service will be hosted in a Controller VC (vc-1) and in case of a failure will failover to vc-0 to ensure service continuity.

Load balancer metering

While the actual object storage operations will be distributed to the “backend servers”, in some cases the system administrator would like to review the amount of concurrent sessions their object storage is handling.

In the NextGen Object Storage the active session count can be reviewed in the management interface:

1. Virtual Controllers view - navigate to the Virtual Controller view and select the virtual controller with `vc-1` as ID. In the south pane, select the **Frontend Metering** tab.
2. Performance section - under the system section, navigate to the **Performance** section. Expand the **Virtual Controller > Controller > vc-1** view and drag and drop the **Active Connections** pane.

The **Active Connections** connections graph will provide a breakdown of connections within the object storage (per VC). The user can toggle between the graph view and a the table view using the **TABLES** view.

15.1.2 Using an external load balancer

Zadara’s Object Storage provides an easy way to integrate with an existing or newly provisioned software/hardware load balancer.

The instructions below are example for setting up an external load balancer to terminate SSL connections and distribute the across all VCs.

There are many load balancer solutions in the market, setting them all up is quite similar procedure. This appendix gives an example of HAproxy, an open-source TCP/HTTP load-balancing proxy server that can be found in www.haproxy.org

The external load-balancer recommended configuration below will allow the following:

- SSL Termination is done on the external load balancer for both object operation API's and GUI connections. Authentication connections are always handled in the Object Storage itself.
- Custom TLS certificate located on the load balancer is used for TLS connections
- Object operation connections are redirected to NextGen Object Storage proxy VC's
- Object operation connections are distributed between VC's unevenly (proxy VC's to take more load than storage VC's, and HA VC's to take the lowest load)
- Redirected object operation connections will include the original client IP in a special header added by the load balancer (for logging in NextGen Object Storage proxy)
- HTTP-based health check is performed by the load balancer to probe all NextGen Object Storage proxy VC's
- Authentication connections are redirected to the Object Storage floating IP (SSL pass-through terminated on the NextGen Object Storage, Custom TLS certificate must be uploaded to the NextGen Object Storage as well).
- UI requests are redirected to the NextGen Object Storage floating IP (over port 8443)
- Graphical statistics interface is enabled on the load balancer

Apply the following configuration to your NextGen Object Storage Settings:

1. Set the internet-facing domain-name/IP of the external LoadBalancer as NextGen Object Storage API Hostname / IP (zadara-qa.com which resolves to the external LB IP 180.80.2.217, is set in this example as NextGen Object Storage API Hostname)
2. Upload your custom SSL certificate (will be used for authentication connections). The certificate should match the custom domain name.
3. Set SSL Termination to "External"

HAProxy Installation and configuration instructions:

- Install HA Proxy:

```
sudo add-apt-repository -y ppa:vbernat/haproxy-1.5
sudo apt-get update
sudo apt-get install -y haproxy
```

- Upload your custom SSL certificate to HAProxy server. In this example the certificate PEM file is placed under `/etc/ssl/private/zadara_custom.pem`
- Edit `/etc/haproxy/haproxy.cfg` to include the following:

```
global
    maxconn 2048
    log /dev/log local0
    log /dev/log local1 notice
    chroot /var/lib/haproxy
    stats socket /run/haproxy/admin.sock mode 660 level admin
    stats timeout 30s
    user haproxy
    group haproxy
    daemon
    tune.ssl.default-dh-param 2048
    # Default SSL material locations
    ca-base /etc/ssl/certs
    crt-base /etc/ssl/private
    # Default ciphers to use on SSL-enabled listening sockets.
    # For more information, see ciphers(1SSL). This list is from:
    # https://hynek.me/articles/hardening-your-web-servers-ssl-ciphers/ssl-default-bind-ciphers
```

(continues on next page)

(continued from previous page)

```
ECDH+AESGCM:DH+AESGCM:ECDH+AES256:DH+AES256:ECDH+AES128:DH+AES:ECDH+3DES:DH+3DES:RSA+AESGCM:
↔RSA+AES:RSA+3DES:!aNULL:!MD5:!DSS
ssl-default-bind-options no-sslv3

defaults
    log        global
    mode       http
    option     httplog
    option     dontlognull
    timeout    connect 5000
    timeout    client  50000
    timeout    server  50000
    errorfile  400 /etc/haproxy/errors/400.http
    errorfile  403 /etc/haproxy/errors/403.http
    errorfile  408 /etc/haproxy/errors/408.http
    errorfile  500 /etc/haproxy/errors/500.http
    errorfile  502 /etc/haproxy/errors/502.http
    errorfile  503 /etc/haproxy/errors/503.http
    errorfile  504 /etc/haproxy/errors/504.http
    frontend  fe-object-operations
    bind      180.80.2.217:443 ssl crt /etc/ssl/private/zadara\_custom.pem
    mode      http
    default\_backend be-zios-object-operations

frontend fe-auth
    bind      180.80.2.217:5000
    option    tcplog
    mode      tcp
    default\_backend be-floating-zios-auth

frontend fe-gui
    bind      180.80.2.217:8443 ssl crt /etc/ssl/private/zadara\_custom.pem
    mode      http
    default\_backend be-floating-zios-gui

backend be-zios-object-operations
    mode      http
    balance   roundrobin
    option    forwardfor
    option    httpclose
    option    httpchk HEAD /healthcheck HTTP/1.0
    server    ziosStorageProxy0 190.90.2.102:8080 weight 10 check
    server    ziosStorageProxy1 190.90.2.104:8080 weight 10 check
    server    ziosStorageProxy2 190.90.2.114:8080 weight 50 check
    server    ziosProxyOnly3 190.90.2.106:8080 weight 100 check
    server    ziosProxyOnly4 190.90.2.109:8080 weight 100 check

backend be-floating-zios-auth
    mode      tcp
    server    ziosFloating 190.90.2.118:5000

backend be-floating-zios-gui
    mode      http
    server    ziosFloating 190.90.2.118:80

listen stats \*:1936
    stats enable
```

(continues on next page)

(continued from previous page)

```
stats uri /
stats auth zadara:zadara
```

- Enable HAProxy logging (Optional)

- Edit rsyslog.conf

```
sudo vi /etc/rsyslog.conf
# provides UDP syslog reception
$ModLoad imudp
$UDPServerRun 514
# provides TCP syslog reception
$ModLoad imtcp
$InputTCPServerRun 514
```

- Restart the service:

```
sudo service rsyslog restart
```

- Restart HAProxy service:

```
sudo service haproxy restart
```

- Monitor statistics by browsing to `http://**<HAProxy server IP>** :1936/` Credentials: `zadara/zadara`

CHAPTER

SIXTEEN

TROUBLESHOOTING

16.1 Management interface access

16.1.1 Troubleshooting Access Issues

When opening the console for the first time after changing the default settings, you might get the following error message, as a result of wrong network configuration, or lack of SSL certification trust. Follow the instruction to fix the situation.