Deploying an Object Storage Gateway

Objectives

After completing this section, you should be able to deploy a RADOS Gateway to provide clients with access to Ceph object storage.

Introducing Object Storage

Object storage stores data as discrete items, each individually called an *object*. Unlike files in a file system, objects are not organized in a tree of directories and subdirectories. Instead, objects are stored in a flat namespace. Each object is retrieved by using the object's unique *object ID*, also known as an *object key*.

Applications do not use normal file-system operations to access object data. Instead, applications access a REST API to send and receive objects. Red Hat Ceph Storage supports the two most common object APIs, Amazon S3 (Simple Storage Service) and OpenStack Swift (OpenStack Object Storage).

Amazon S3 calls the flat namespace for object storage a *bucket* while OpenStack Swift calls it a *container*. Because a namespace is flat, neither buckets nor containers can be nested. Ceph typically uses the term bucket, as does this lecture.

A single user account can be configured for access to multiple buckets on the same storage cluster. Buckets can each have different access permissions and be used to store objects for different use cases.

The advantage of object storage is that it is easy to use, expand, and scale. Because each object has a unique ID, it can be stored or retrieved without the user knowing the object's location. Without the directory hierarchy, relationships between objects are simplified.

Objects, similar to files, contain a binary data stream and can grow to arbitrarily large sizes. Objects also contain *metadata* about the object data, and natively support extended metadata information, typically in the form of key-value pairs. You can also create your own metadata keys and store custom information in the object as key values.

Introducing the RADOS Gateway

The RADOS Gateway, also called the Object Gateway (RGW), is a service that provides access to the Ceph cluster for clients using standard object storage APIs. The RADOS Gateway supports both the Amazon S3 and OpenStack Swift APIs.

The core daemon, `radosgw`, is built on top of the `librados` library. The daemon provides a web service interface, based on the Beast HTTP, WebSocket, and networking protocol library, as a front-end to handle API requests.

The `radosgw` is a client to Red Hat Ceph Storage that provides object access to other client applications. Client applications use standard APIs to communicate with the RADOS Gateway, and the RADOS Gateway uses `librados` module calls to communicate with the Ceph cluster.