

Deploying Shared File Storage

Objectives

After completing this section, you should be able to provide file storage on the Ceph cluster by deploying the Ceph File System (CephFS).

The Ceph File System and MDS

The *Ceph File System (CephFS)* is a POSIX-compliant file system that is built on top of RADOS, Ceph's distributed object store. File-based storage organizes your data as a traditional file system, with a directory tree hierarchy. Implementing the Ceph File System requires a running Ceph storage cluster and at least one *Ceph Metadata Server (MDS)* to manage the CephFS metadata, separately from file data, which reduces complexity and improves reliability. Similar to RBD and RGW, the CephFS daemon is implemented as a native interface to `librados`.

File, Block, and Object Storage

File-based storage organizes your data as a traditional file system. Data is saved as files with a name and associated metadata, such as modification time stamps, an owner, and access permissions. File-based storage uses a directory tree hierarchy to organize how files are stored.

Block-based storage provides a storage volume that operates similar to a disk device, organized into equally sized chunks. Typically, block-based storage volumes are either formatted with a file system, or applications such as databases directly access and write to them.

With object-based storage, you can store arbitrary data and metadata as a unit that is labeled with a unique identifier in a flat storage pool. Rather than accessing data as blocks or in a file-system hierarchy, you use an API to store and retrieve objects. Fundamentally, the Red Hat Ceph Storage RADOS cluster is an object store.

The Metadata Server

The Metadata Server (MDS) manages metadata for CephFS clients. This daemon provides information that CephFS clients need to access RADOS objects, such as providing file locations within the file-system tree. MDS manages the directory hierarchy and stores file metadata, such as the owner, time stamps, and permission modes, in a RADOS cluster. MDS is also responsible for access caching and managing client caches to maintain cache coherence.

MDS daemons operate in two modes: active and standby. An active MDS manages the metadata on the CephFS file system. A standby MDS serves as a backup, and switches to the active mode if the active MDS becomes unresponsive. CephFS shared file systems require an active MDS service. You should deploy at least one standby MDS in your cluster to ensure high availability.

If you do not create enough MDS pools to match the number of configured standby daemons, then the Ceph cluster displays a WARN health status. The recommended solution is create more MDS pools to provide a pool for each daemon. However, a temporary solution is to set the number of standby pools to 0, which disables the Ceph MDS standby check through the `ceph fs set fs-name standby_count_wanted 0` command.