Declarative Object Storage at Scale

Integrating Rook, Ceph, and OpenStack

Joachim Kraftmayer, Artem Torubarov
FOSDEM Feb, 2025

2

Blueprint for Open Cloud

Free and Open

All parts and components are OSS. Actively maintained.

CLYSO

Blueprint for Open Cloud

Free and Open

All parts and components are OSS. Actively
maintained.

Seamlessly integrated

Tested together. Reproducible. Declarative
configuration.

Blueprint for Open Cloud

Free and Open

All parts and components are OSS. Actively maintained.

Seamlessly integrated

Tested together. Reproducible. Declarative configuration.

Maintained

Operated in production with the latest releases.

CLYSO

Case study

100 Deployments	**16** Regions	**>5Pb** Cluster size
1-2 Availability Zones	**>100** Nodes per cluster	**>1000** OSDs per cluster

Landscape

openstack®

KEYSTONE   SWIFT   BARBICAN   CINDER   MANILA

Landscape

openstack®

ceph

KEYSTONE	SWIFT	BARBICAN	CINDER	MANILA
	Rados gateway		Ceph RBD	Ceph FS

Landscape

openstack®

KEYSTONE   SWIFT   BARBICAN

ceph

Rados gateway

ROOK

Object Storage

Part 1: Ceph and Openstack

- Openstack SWIFT is eventually-consistent unlike Ceph

- Keystone V3 API was added in Ceph Jewel (2016)

- RGW SSE and Barbican in Ceph Luminous (2017)

- RGW SWIFT implementation is less mature than S3 today

- RGW is capable of replacing openstack SWIFT

Findings

- Public buckets with SWIFT & Keystone [Fixed in reef]

- RGW and Openstack SWIFT ACLs are different [TBD]

- Missing last_modified field for SWIFT container [WIP]

- SWIFT container temp_url is not supported [TBD]

- Different LastModified format in S3 HeadObject and

  ListObjectVersions [WIP]

Findings

- SWIFT role-based ACL is not supported [TBD]

- S3 Conditional writes [TBD]

- Keystone request caching issue [WIP]

- SHA256 Support in SWIFT temp_url [fixed in Quincy]

- Container GET and HEAD return different obj count [TBD]

- Cache for Barbican keys [WIP]

Findings

- Access S3 objects with SWIFT api if SSE enabled [TBD]

- Versioned S3 bucket incompatible with versioned SWIFT

  container [TBD]

- Quota and Usage is tracked only per project and per bucket

- SWIFT Delete account method [TBD]

Part 2: Declarative Ceph deployment with Rook

Rook allowed us to have fully declarative Object Storage setup:

1. Label k8s node with topology labels

2. Declare device filters for Rook OSD discovery

3. Declare OSD pools

4. Declare RGW instances

Findings

- Support RGW target placements [done in v1.15]

- Multi-instance RGW deployments [done in v1.16]

- Custom hostname topology label [WIP]

- Support external arbiter mon [WIP]

- Observability: Metrics and tracing. [Improved in Ceph Squid]

Object Storage Roadmap

- Maintaining RGW SWIFT and make it in par with RGW S3

- Start discussion about Quota and Usage improvements.

  Manage quota per user per target placement.

- Test scalability and day-two operations with Rook.

- Data migration.

- Want to hear about your use-cases, looking for case studies

Tomorrow, AW1.120 13:35

[Building Europe's Platform Mesh: Cloud-Native APIs for Multi-Provider Integration and Digital Sovereignty](#)