Part 1: Background Foundation
How to use Ceph storage via iSCSI

First, some background …

• **Ceph** makes HA storage available in several ways:
  - As a block device
  - As a RESTful (web) service
  - Others (will detail soon)

• **iSCSI** allows remote access of storage via TCP/IP
  - Storage devices or device servers are called *Targets*
    - Can export a block device as an iSCSI target, using the **LIO** package
  - Clients are called *Initiators*
    - Available in Linux via the open-iscsi package
How to use Ceph storage via iSCSI

How to combine the reliability of Ceph with the popular iSCSI (Storage over TCP/IP) protocol?

That's what this talk is about!
Background: Ceph Architecture

• Clustering technology, so failure resistant/HA
• Available in SUSE Enterprise Storage
• Makes a pool of HA storage available via different access methods, such as:
  - RGW: RADOS Gateway – for RESTful access
  - librados – roll your own application
  - CephFS – not used very much yet
  - RBD: RADOS Block Device – looks like a local storage device
    - This is the one we care about today
Background: Ceph Architecture

LIBRADOS
A library allowing apps to directly access RADOS, with support for C, C++, Java, Python, Ruby, and PHP

RADOSGW
A bucket-based REST gateway, compatible with S3 and Swift

RBD
A reliable and fully-distributed block device, with a Linux kernel client and a QEMU/KVM driver

CEPHFS
A POSIX-compliant distributed file system, with a Linux kernel client and support for FUSE

RADOS
A reliable, autonomous, distributed object store comprised of self-healing, self-managing, intelligent storage nodes.
Ceph RADOS Block Device Features

- Block device backed by RADOS objects
  - Objects replicated across Ceph **OSDs**
- Thin provisioned
- Online resizeable
- Supports snapshots and clones

- Linux kernel or librbd clients
  - Usage restricted to a subset of operating systems and applications
  - Features *can lag* behind RGW a bit (opinion)
Background: iSCSI Architecture

- Mechanism for transporting block storage traffic over a regular TCP/IP network
- iSCSI initiators (clients) communicate with iSCSI targets (servers)
- SCSI commands and responses encapsulated in iSCSI packets, inside TCP packets
- Remote storage appears on the iSCSI initiator as a local hard disk
  - Attach and format with XFS, NTFS, etc.
  - Boot from a remote target with an iSCSI capable network adapter or boot loader
Previous Method for iSCSI and RBD: “Roll your own”

• On Target system:
  - RBD converts Ceph protocol to/from Block Device
  - LIO converted block device to/from iSCSI
  - Block Device is an intermediate format: wasteful?

• On Initiator system:
  - Client access local block device
  - iSCSI initiator converts iSCSI to/from Block Device
  - This is okay, because iSCSI is designed to do this
Previous Method for iSCSI and RBD: “Roll your own”

• Problems with using the current Block Layer
  – Doesn't support *atomic compare and write*
  – Doesn't support *Persistent Group Reservations*

• Needed for Active/Active Multipath IO (*mpio*) iSCSI Gateway

• Until Block Layer supports these, we need a different approach
Updated Method for iSCSI and RBD: The iSCSI gateway for RBD

- Expose benefits of Ceph RBD to other systems
  - No requirement for Ceph-aware applications or operating systems

- Standardized iSCSI interface
  - Mature and trusted protocol (RFC 3720)

- iSCSI initiator implementations are widespread
  - Provided with most modern operating systems
  - Open-iscsi is the most common initiator on Linux

- The iSCSI target uses the LIO driver
The iSCSI LIO Target

• LIO – Linux IO Target

• In kernel SCSI target implementation
  – Support for a number of SCSI transports
  – Pluggable storage backend
  – Is the current “preferred” iSCSI Linux target

• Flexible configuration
  – Uses the targetcli utility: like a shell
Current Approach: iSCSI and RBD
Current Approach: iSCSI and RBD
Updated Approach: iSCSI and RBD
Part 2: More Detail
RBD iSCSI gateway
The Ceph View
RBD iSCSI gateway

• LIO target configured with iSCSI transport fabric

• RBD backstore module
  - Translates SCSI IO into Ceph OSD requests
  - Special handling of operations that require exclusive device access
    - Atomic COMPARE AND WRITE, WRITE SAME and reservations

• lrbd: Multi-node configuration utility
  - Applies iSCSI target configuration across multiple gateways via targetcli
RBD iSCSI gateway
Multipath Support

• Allows for initiator access via redundant paths
  – iSCSI gateway node with multiple network adapters
    – Protection from a single network adapter failure
  – Multiple iSCSI gateways exporting same RBD image
    – Protection from entire gateway failure

• Initiator responsible for utilization of redundant paths
  – Available paths advertised in iSCSI discovery exchange
  – May choose to round-robin the IO, or to failover/failback
LIO using RBD iSCSI gateway

Multipath Support

- OSD1
- OSD2
- OSD3
- OSD4

RBD Image

RBD module
- LIO iSCSI Target

RBD module
- LIO iSCSI Target

iSCSI Initiator
RBD iSCSI gateway
Optimizations

• Efficient handling of certain SCSI operations
  - Offload RBD image IO to OSDs
    - Avoid locking on iSCSI gateway nodes
  - COMPARE AND WRITE
    - New *cmpext* OSD operation to handle RBD data comparison
    - Dispatch as compound *cmpext*+*write* OSD request
  - WRITE SAME
    - New *writesame* OSD operation to expand duplicate data at the OSD
  - Reservations
    - State stored as RBD image extended attribute
    - Updated using compound *cmpxattr*+*setxattr* OSD request
Configuration with lrbd

• Apply LIO configuration across multiple iSCSI gateway nodes
  - JSON configuration format
  - Targets, portals, RBD images and authentication information

• Configuration state stored in Ceph cluster
  - iSCSI gateway nodes apply configuration on boot
Configuration with lrbd

- **Targets** section
  - iSCSI gateway hosts
  - Target iSCSI Qualified Name (IQN)
- **Portals** section
  - IP addresses to utilize for iSCSI traffic
- **Pools** section
  - RBD images to expose
- **Auth** section
  - Access restrictions based on initiator name
  - CHAP credentials
Some iSCSI Initiators

• open-iscsi
  - Default iSCSI initiator shipped with SLES10 and later
  - Multipath supported in combination with dm-multipath
  - Available on most Linux distributions

• Microsoft iSCSI initiator
  - Installed by default from Windows Server 2008 and later
    - Not available on desktops
  - Supports MPIO in recent versions

• VMware ESX
  - Concurrent clustered filesystem (VMFS) access from multiple initiators
Demonstration
Demonstration Environment

SLES12 VM

Hypervisor 10.162.186.111

Hypervisor 10.162.186.112

iSCSI Gateway 10.160.222.148

iSCSI Gateway 10.160.222.149
Demonstration Environment

IO Failover
Demonstration Environment
For More Information

• open-iscsi
  - RFC 3720: https://www.ietf.org/rfc/rfc3720.txt
  - URL: http://www.openiscsi.org
  - Discussion: openiscsi@googlegroups.com

• Ceph
  - General: http://ceph.com
  - Documentation: http://docs.ceph.com/0.80.5/

• SUSE Enterprise Storage
  - Product: https://www.suse.com/products/suse-enterprise-storage/
  - Documentation: https://www.suse.com/documentation/
Questions?

Thank you.
Unpublished Work of SUSE LLC. All Rights Reserved.
This work is an unpublished work and contains confidential, proprietary and trade secret information of SUSE LLC.
Access to this work is restricted to SUSE employees who have a need to know to perform tasks within the scope of
their assignments. No part of this work may be practiced, performed, copied, distributed, revised, modified, translated,
abridged, condensed, expanded, collected, or adapted without the prior written consent of SUSE.
Any use or exploitation of this work without authorization could subject the perpetrator to criminal and civil liability.

General Disclaimer
This document is not to be construed as a promise by any participating company to develop, deliver, or market a
product. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making
purchasing decisions. SUSE makes no representations or warranties with respect to the contents of this document,
and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. The
development, release, and timing of features or functionality described for SUSE products remains at the sole
discretion of SUSE. Further, SUSE reserves the right to revise this document and to make changes to its content, at
any time, without obligation to notify any person or entity of such revisions or changes. All SUSE marks referenced in
this presentation are trademarks or registered trademarks of Novell, Inc. in the United States and other countries. All
third-party trademarks are the property of their respective owners.