SUSE® Enterprise Storage v5.5 Implementation Guide

HPE Apollo 4510 Gen10 Series Servers

Written by:
David Byte, SUSE

Guide

www.suse.com

Implementation Guide

SUSE Enterprise Storage

Table of Contents

Introduction

The objective of this guide is to present a step-by-step process on how to implement SUSE® Enterprise Storage (v5.5) on the HPE Apollo 4510 Gen10 platform.

Before attempting the process, we recommend that you read the document in its entirety, along with the supplemental appendix information.

The platform is built and deployed to illustrate the ability to quickly deploy a robust SUSE Enterprise Storage cluster on the HPE Apollo platform. The deployment aligns with architectural best practices and will support the implementation of any of the currently supported protocols.

Upon completion of the steps in this document, a working SUSE Enterprise Storage (v5.5) will be operational, as described in the Deployment Guide.

Configuration
The recommended architecture for SUSE Enterprise Storage on HPE Apollo 4510 Gen10 leverages two models of HPE servers. The role/functionality of each SUSE Enterprise Storage component is explained in more detail in the architectural overview section.

Ceph admin, monitor and protocol gateway functions:

- *HPE Proliant DL360 Gen10 Servers*

Storage Nodes:

- *Four HPE Apollo 4510 Gen10 Servers*

Switching infrastructure:

- *Two HPE StoreFabric SN2700M Switches*

Software:

- *SUSE Enterprise Storage 5.5 (Please note: The SUSE Enterprise Storage subscription includes a limited use [for SUSE Enterprise Storage] entitlement for SUSE Linux Enterprise Server.)*

Target Audience
This reference architecture is targeted at administrators who deploy software-defined storage solutions within their data centers and make the different storage services accessible to their own customer base. By following this document as well as those referenced herein, the administrator should have a full view of the SUSE Enterprise Storage architecture, deployment and administrative tasks and a specific set of recommendations for deployment of the hardware and networking platform.

Business Problem and Business Value

SUSE Enterprise Storage delivers a highly scalable, resilient, self-healing storage system designed for large- scale environments ranging from hundreds of Terabytes to Petabytes. This software-defined storage product can reduce IT costs by leveraging industry-standard servers to present unified storage servicing block, file and object protocols. Having storage that can meet the current needs and requirements of the data center while supporting topologies and protocols demanded by new web-scale applications enables administrators to support the ever-increasing storage requirements of the enterprise with ease.

Business Problem

Customers of all sizes face a major storage challenge: While the overall cost per Terabyte of physical storage has gone down over the years, a data growth explosion has taken place, driven by the need to access and leverage new data sources (e.g., external sources such as social media) and to "manage" new data types (e.g., unstructured or object data). These ever-increasing "data lakes" need different access methods: file, block or object.

Addressing these challenges with legacy storage solutions would require a number of specialized products (usually driven by access method) with traditional protection schemes (e.g., RAID). These solutions struggle when scaling from Terabytes to Petabytes at reasonable cost and performance levels.

Business Value

This software-defined storage solution enables transformation of the enterprise infrastructure by providing a unified platform where structured and unstructured data can co-exist and be accessed as file, block or object depending on the application requirements. The combination of open source software (Ceph) and industry-standard servers reduce cost while providing the on-ramp to unlimited scalability needed to keep up with future demands.

Requirements

Enterprise storage systems require reliability, manageability and serviceability. The legacy storage players have established a high threshold for each of these areas and now expect the software-defined storage solutions to offer the same. Focusing on these areas helps SUSE make open source technology enterprise consumable. When combined with highly reliable and manageable hardware from HPE, the result is a solution that meets the customer's expectations.

Functional Requirements

A SUSE Enterprise Storage solution is:

- *Simple to setup and deploy, within the documented guidelines of system hardware, networking and environmental prerequisites.*
- *Adaptable to the physical and logical constraints needed by the business—initially and as needed over time for performance, security and scalability concerns.*
- *Resilient to changes in physical infrastructure components caused by failure or required maintenance.*
- *Capable of providing optimized object and block services to client access nodes, either directly or through gateway services.*

Architectural Overview

This section complements the SUSE Enterprise Storage Technical Overview[1] document available online, which presents the concepts behind software-defined storage and Ceph, as well as a quick start guide (non-platform specific).

1 **www.suse.com/media/white-paper/suse_enterprise_storage_technical_overview_wp.pdf**

Solution Architecture

SUSE Enterprise Storage provides unified block, file and object access based on Ceph. Ceph is a distributed storage solution designed for scalability, reliability and performance. A critical component of Ceph is the RADOS object storage. RADOS enables a number of storage nodes to function together to store and retrieve data from the cluster, using object storage techniques. The result is a storage solution that is abstracted from the hardware.

Ceph supports both native and traditional client access. The native clients are aware of the storage topology and communicate directly with the storage daemons over the public network, resulting in horizontally scaling performance. Non-native protocols, such as ISCSI, S3 and NFS require the use of gateways. While these gateways might be considered a limiting factor, the ISCSI and S3 gateways can scale horizontally using load balancing techniques.

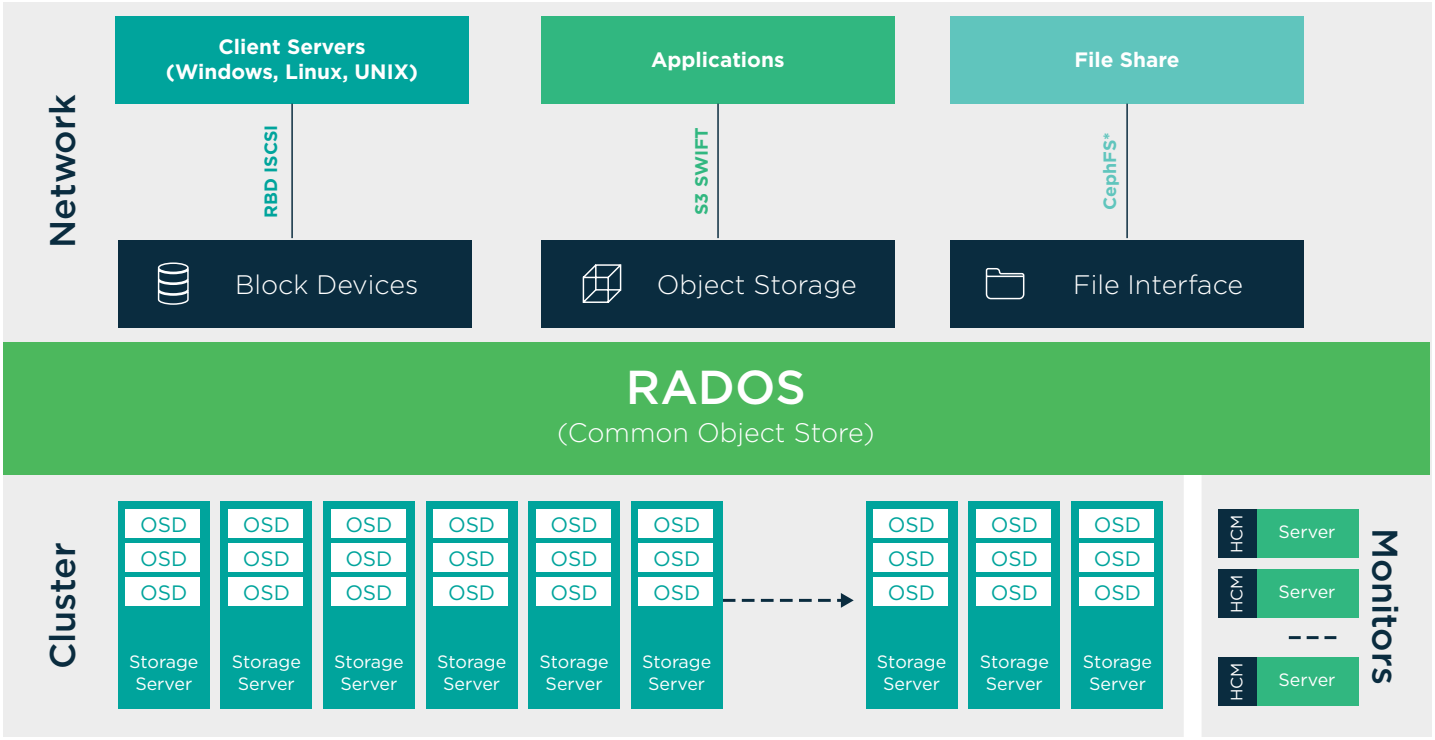

Figure 1. *Ceph architecture diagram*

In addition to the required network infrastructure, the minimum SUSE Enterprise Storage cluster is comprised of a minimum of one administration server (physical or virtual), four object storage device nodes (OSDs) and three monitor nodes (MONs). Specific to this implementation, the architecture includes the following:

- *One ProLiant DL360 system is deployed as the administrative host server. The administration host is the Salt-master and hosts the SUSE Enterprise Storage Administration Interface, openATTIC, which is the central management system that supports the cluster.*

- *Three ProLiant DL360 systems are deployed as monitor (MONs) nodes. Monitor nodes maintain information about the cluster health state, a map of the other monitor nodes and a CRUSH map. They also keep history of changes performed on the cluster.*
- *Additional ProLiant DL360 servers may be deployed as iSCSI gateway nodes. iSCSI is a storage area network (SAN) protocol that enables clients (called initiators) to send the SCSI command to SCSI storage devices (targets) on remote servers. This protocol is utilized for block-based connectivity to environments such as Microsoft Windows, VMware and traditional UNIX. These systems may be scaled horizontally through client usage of multi-path technology.*
- *The RADOS gateway may also be deployed on ProLiant DL360 nodes. The RADOS gateway provides S3 and Swift-based access methods to the cluster. These nodes are generally situated behind a load balancer infrastructure to provide redundancy and scalability. It is important to note that the load generated by the RADOS gateway can consume a significant amount of compute and memory resources, making the minimum recommended configuration contain 6-8 CPU cores and 32GB of RAM.*
- *This configuration uses HPE Apollo 4510 Gen10 series systems as storage nodes. The storage nodes contain individual storage devices that are each assigned an Object Storage Daemon (OSD). The OSD daemon assigned to the device stores data and manages the data replication and rebalancing processes. OSD daemons also communicate with the monitor (MON) nodes and provide them with the state of the other OSD daemons.*

Networking Architecture

A software-defined solution is only as reliable as its slowest and least redundant component. This makes it important to design and implement a robust, high performance storage network infrastructure. From a network perspective for Ceph, this translates into:

- *Separation of cluster (backend) and client-facing (public) network traffic. This isolates Ceph OSD daemon replication activities from Ceph clients. This can be achieved through separate physical networks or through the use of VLANs.*
- *Redundancy and capacity in the form of bonded network interfaces connected to switches.*

Figure 2 shows the logical layout of the traditional Ceph cluster implementation.

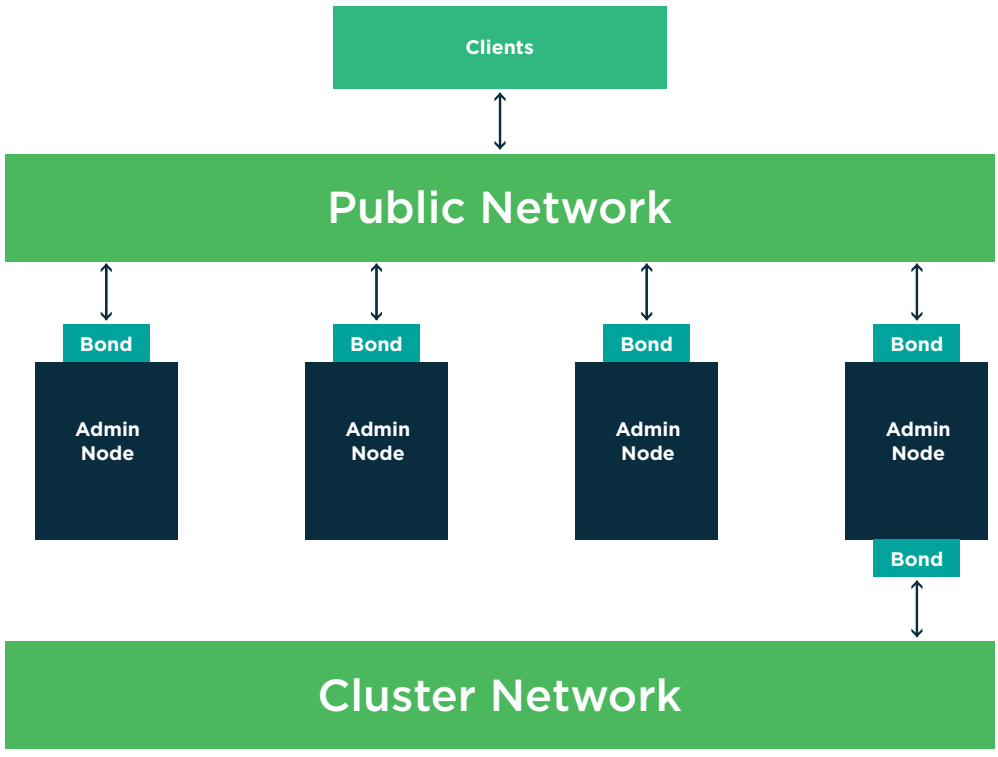

Figure 2. *Sample networking diagram for Ceph cluster*

Network/IP Address Scheme

Specific to this implementation, the following naming and addressing scheme were utilized.

Function	Hostname	Primary Network	Cluster Network
Admin (Host)	sesadmin.suse.lab	192.168.124.20	N/A
Monitor	monnode1.suse.lab	192.168.124.21	N/A
Monitor	monnode2.suse.lab	192.168.124.22	N/A
Monitor	monnode3.suse.lab	192.168.124.23	N/A
OSD Node	osdnode1.suse.lab	192.168.124.31	192.168.100.31
OSD Node	osdnode2.suse.lab	192.168.124.32	192.168.100.32
OSD Node	osdnode3.suse.lab	192.168.124.33	192.168.100.33
OSD Node	osdnode4.suse.lab	192.168.124.34	192.168.100.34

Component Model

The preceding sections provided information on both the overall HPE hardware as well as an introduction to the Ceph software architecture. In this section, the focus is on the SUSE components: SUSE Linux Enterprise Server (SLES), SUSE Enterprise Storage (SES) and the Subscription Management Tool (SMT).

Component Overview (SUSE)

- **SUSE Linux Enterprise Server**—*A world-class secure, open source server operating system, equally adept at powering physical, virtual or cloud-based mission-critical workloads. Service Pack 2 further raises the bar in helping organizations to accelerate innovation, enhance system reliability, meet tough security requirements and adapt to new technologies.*
- **Subscription Management Tool for SLES12 SP3**—*Enables enterprise customers to optimize the management of SUSE Linux Enterprise (and extensions such as SUSE Enterprise Storage) software updates and subscription entitlements. It establishes a proxy system for SUSE Customer Center with repository and registration targets.*
- **SUSE Enterprise Storage**—*Provided as an extension on top of SUSE Linux Enterprise Server, this intelligent software-defined storage solution, powered by Ceph technology with enterprise engineering and support from SUSE, enables customers to transform enterprise infrastructure to reduce costs while providing unlimited scalability.*

Deployment

This section should be considered as a supplement to the online documentation[2]. Specifically, the *SUSE Enterprise Storage 5 Deployment Guide*[3] as well as *SUSE Linux Enterprise Server Administration Guide*[4]. It is assumed that a Subscription Management Tool server exists within the environment. If not, please follow the information in Subscription Management Tool (SMT) for SLES 12 SP3[5] to make one available. The emphasis is on specific design and configuration choices.

Network Deployment Overview

The following considerations for the network configuration should be attended to:

- *Ensure that all network switches are updated with consistent firmware versions.*
- *Configure 802.3ad for system port bonding between the switches and enable jumbo frames.*

2 **www.suse.com/documentation/**
3 **www.suse.com/documentation/suse-enterprise-storage-5/book_storage_deployment/data/book_storage_deployment.html**
4 **www.suse.com/documentation/sles-12/book_sle_admin/data/book_sle_admin.html**
5 **www.suse.com/documentation/sles-12/book_smt/data/book_smt.html**

- *Specific configuration for this deployment can be found in Appendix C: Network Switch Configuration.*
- *Network IP addressing and IP ranges need proper planning. In optimal environments, a single storage subnet should be used for all SUSE Enterprise Storage nodes on the primary network, with a separate, single subnet for the cluster network. Depending on the size of the installation, ranges larger than /24 might be required. When planning the network, current as well as future growth should be taken into consideration.*
- *Set up DNS A records for all nodes. Decide on subnets and VLANs and configure the switch ports accordingly.*
- *Ensure that you have access to a valid, reliable NTP service. This is a critical requirement for all nodes. If you do not have access, it is recommended that you use the admin node.*

Hardware Deployment Configuration (Suggested)
The following considerations for the hardware platforms should be attended to:

- *Ensure Boot Mode is set to 'UEFI' for all the physical nodes that comprise the SUSE Enterprise Storage Cluster.*
- *Verify that the BIOS/uEFI level on the physical servers correspond to those on the SUSE YES certification for the HPE platforms:*
  - DL360 Gen 10
- **www.suse.com/nbswebapp/yesBulletin.jsp?bulletinNumber=145558**
  - HPE Apollo 4510 Gen10
- **www.suse.com/nbswebapp/yesBulletin.jsp?bulletinNumber=146452**
- *Configure the RAID-1 boot media as RAID-1.*
- *Configure all data and journal devices as individual RAID-0.*

Software Deployment Configuration (DeepSea and Salt)
Salt, along with DeepSea, is a stack of components that help to deploy and manage the server infrastructure. It is very scalable, fast and relatively easy to get running.

There are three key Salt imperatives that need to be followed. These are described in detail in section 4 (Deploying with DeepSea and Salt).

- *The Salt master is the host that controls the entire cluster deployment. Ceph itself should NOT be running on the master, because all resources should be dedicated to Salt master services. In our scenario, we used the Admin host as the Salt master.*
- *Salt minions are nodes controlled by Salt master. OSD, monitor and gateway nodes are all Salt minions in this installation.*
- *Salt minions need to correctly resolve the Salt master's host name over the network. This can be achieved through configuring unique host names per interface (osd1-cluster.suse.lab and osd1-public.suse.lab) in DNS and/or local /etc/hosts files.*

DeepSea consists of a series of Salt files to automate the deployment and management of a Ceph cluster. It consolidates the administrator's decision-making in a single location around cluster assignment, role assignment and profile assignment. DeepSea collects each set of tasks into a goal or stage.

The following steps, performed in order, will be used for this reference implementation:

- *Install DeepSea on the Salt master that is the Admin node:*
  - `zypper in deepsea`
- *Start the* salt-master *service and enable:*
  - `systemctl start salt-master.service`
  - `systemctl enable salt-master.service`

- *Install the* salt-minion *on all cluster nodes (including the Admin):*
  - `zypper in salt-minion`
- *Configure all minions to connect to the Salt master: modify the entry for* master *in the* `/etc/salt/minion`
  - In this case, master: sesadmin.domain.com
- *Start the* salt-minion *service and enable:*
  - `systemctl start salt-minion.service`
  - `systemctl enable salt-minion.service`
- *List and accept all Salt keys on the Salt master:* salt-key --accept-all *and verify their acceptance:*
  - `salt-key --list-all`
  - `salt-key --accept-all`
- *If the OSD nodes were used in a prior installation, zap ALL the OSD disks (ceph-disk zap <DISK>).*
- *At this point, you can deploy and configure the cluster:*
  - Prepare the cluster: `salt-run state.orch ceph.stage.prep`
  - Run the discover stage to collect data from all minions and create configuration fragments:
    - `salt-run state.orch ceph.stage.disovery`
  - A proposal for the storage layer needs to be generated at this time. For this configuration, the following commands were utilized:
    - `salt-run proposal.populate name=apollo ratio=9 wal=740-770 db=740-770 target='osd*' db-size=40g wal-size=5g data=5000-7000`

    The result of the above command is a deployment proposal for the disks that use devices with a reported size of five to six terabytes for data and establishes a 5GB write-ahead log partition and a 40GB database partition for each spinner on one of the SSDs in a ratio of 9 spinners to each SSD.
  - A `/srv/pillar/ceph/proposals/policy.cfg` file needs to be created to instruct Salt on the location and configuration files to use for the different components that make up the Ceph cluster (Salt master, admin, monitor and OSDs).
    - *See Appendix B for the* `policy.cfg` *file used in the installation.*
  - Next, proceed with the configuration stage to parse the policy.cfg file and merge the included files into the final form:
    - `salt-run state.orch ceph.stage.configure`
  - The last two steps manage the actual deployment. Deploy monitors and ODS daemons first:
    - `salt-run state.orch ceph.stage.deploy`
      (Note: The command can take some time to complete, depending on the size of the cluster.)
    - *Check for successful completion via:* `ceph –s`
    - *Finally, deploy the services (gateways [iSCSI, RADOS] and openATTIC to name a few):* `salt-run state.orch ceph.stage.services`

Post-Deployment Quick Test

The steps below can be used to validate the overall cluster health (regardless of the deployment method):

```
ceph status
ceph osd pool create test 1024
rados bench –p test 300 write --no-cleanup
rados bench –p test 300 seq
```

Once the tests are complete, you can remove the test pool via:

```
ceph tell mon.* injectargs --mon-allow-pool-delete=true
ceph osd pool delete test test --yes-i-really-really-mean-it
ceph tell mon.* injectargs --mon-allow-pool-delete=false
```

Deployment Considerations

Some final considerations before deploying your own version of a SUSE Enterprise Storage cluster, based on Ceph. As previously stated, please refer to the *Administration and Deployment Guide*.

- *With the default replication setting of 3, remember that the client-facing network will have about half or less of the traffic of the backend network. This is especially true when component failures occur or rebalancing happens on the OSD nodes. For this reason, it is important not to under provision this critical cluster and service resource.*
- *It is important to maintain the minimum number of monitor nodes at three. As the cluster increases in size, it is best to increment in pairs, keeping the total number of MON nodes as an odd number. However, only very large or very distributed clusters would likely need beyond the 3 MON nodes cited in this reference implementation. For performance reasons, it is recommended to use distinct nodes for the MON roles, so that the OSD nodes can be scaled as capacity requirements dictate.*
- *As described in this implementation guide, as well as the SUSE Enterprise Storage documentation, a minimum of four OSD nodes is recommended, with the default replication setting of 3. This will ensure that cluster operation continues, even with the loss of a complete OSD node. Generally speaking, performance of the overall cluster increases as more properly configured OSD nodes are added.*

Conclusion

The HPE Apollo 4510 Gen10 series represents a strong capacity-oriented platform. When combined with the access flexibility and reliability of SUSE Enterprise Storage and the industry-leading support from HPE, any business can feel confident in their ability to address the exponential storage growth they are currently faced with.

Appendix A: Bill of Materials

Component / System

Role	Quantity	Component	Notes
Admin/MON/Gateway servers	4	HPE ProLiant DL360 Gen10	Each node consists of: ■ 1x Xeon Silver 4110 ■ 32GB RAM (64GB for Object Gateway) ■ 2x 240GB Mixed Use M.2 SSD ■ 1x Dual Port Mellanox ConnectX-4 100Gb Ethernet adapter
OSD Hosts	4	HPE Apollo 4510 Gen10	Each node consists of: ■ 2x Xeon gold 6142 Processors ■ 256 GB RAM ■ P408i-a RAID controller with 4GB Cache w/Battery backup ■ P408i-p RAID controller with 4GB Cache w/Battery backup ■ 6x 800GB Write Intensive SATA SSDs ■ 54x 6TB 7.2k SATA Drives ■ 2x 1TB 2.5" SSDs for boot ■ 1x Dual Port Mellanox ConnectX-4 100Gb Ethernet adapter
Software	1	SUSE Enterprise 5.5 Storage Subscription Base configuration	Allows for 4 storage nodes and 6 infrastructure nodes
Switches	2	HPE StoreFabric SN2700M Switch	32 Ports of 100GbE

Appendix B: Policy.cfg

```
cluster-ceph/cluster/*.sls
role-master/cluster/salt*.sls
role-admin/cluster/salt*.sls
role-mon/cluster/mon*.sls
role-mgr/cluster/mon*.sls
role-mds/cluster/mon*.sls
role-openattic/cluster/salt*.sls
config/stack/default/global.yml
config/stack/default/ceph/cluster.yml
profile-apollo/cluster/*.sls
profile-apollo/stack/default/ceph/minions/*.yml
```

Appendix C: Network Switch Configuration

First, properly cable and configure each node on the switches. Ensuring proper switch configuration at the outset will prevent networking issues later. The key configuration items include creating the MLAG topology (stacking) and LACP groups and enabling jumbo frames. Each aggregation group needs a unique number, planned ahead of time. It is also recommended that you disable the spanning tree on the ports utilized for storage.

Configure MLAG as described in the *MLNX-OS for HPE StoreFabric M-Series Ethernet Switch User Manual*.

To configure an LACP group for switch 1 port 1 and switch 2 port 1 and to enable jumbo frame support, perform the following commands:

```
enable
configure terminal
lacp

interface port-channel 1
lacp-individual enable
mtu 9216
exit
interface ethernet 1/1
mtu 9216 force
channel-group 1 mode active
exit
interface ethernet 2/1
mtu 9216 force
channel-group 1 mode active
exit

configuration write
```

Repeat these steps for each aggregation group required.

Create at least 1 VLAN for cluster communication. In this example, we are using VLAN 3001 for the cluster traffic.

```
enable
configure terminal
vlan 3001
exit
```

Assign VLANs to ports. The configuration below assumes a port-based VLAN of 1 (default):

```
enable
configure terminal
interface port-channel 1
switchport mode hybrid
switchport access vlan 1
switchport hybrid allowed-vlan add 3001
exit
configuration write
```

Appendix D: OS Networking Configuration

Perform the network configuration during installation.

- *While the images depict ConnectX-3 cards, the actual cards used in the RA are ConnectX-4.*

Set the ConnectX-4 100Gb interfaces to No Link

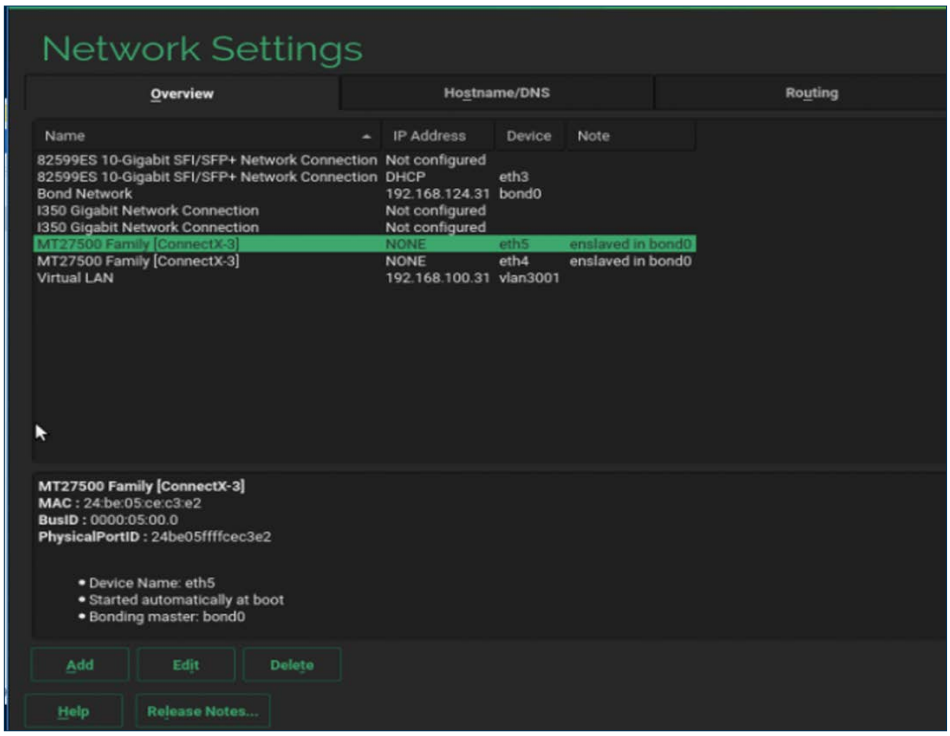

Figure 3.

Add an interface of type bond, set it to the proper IP address for the untagged VLAN, and proceed to the Bond Slaves page, where the ConnectX-4 interfaces should be selected and the mode set to 802.3ad.

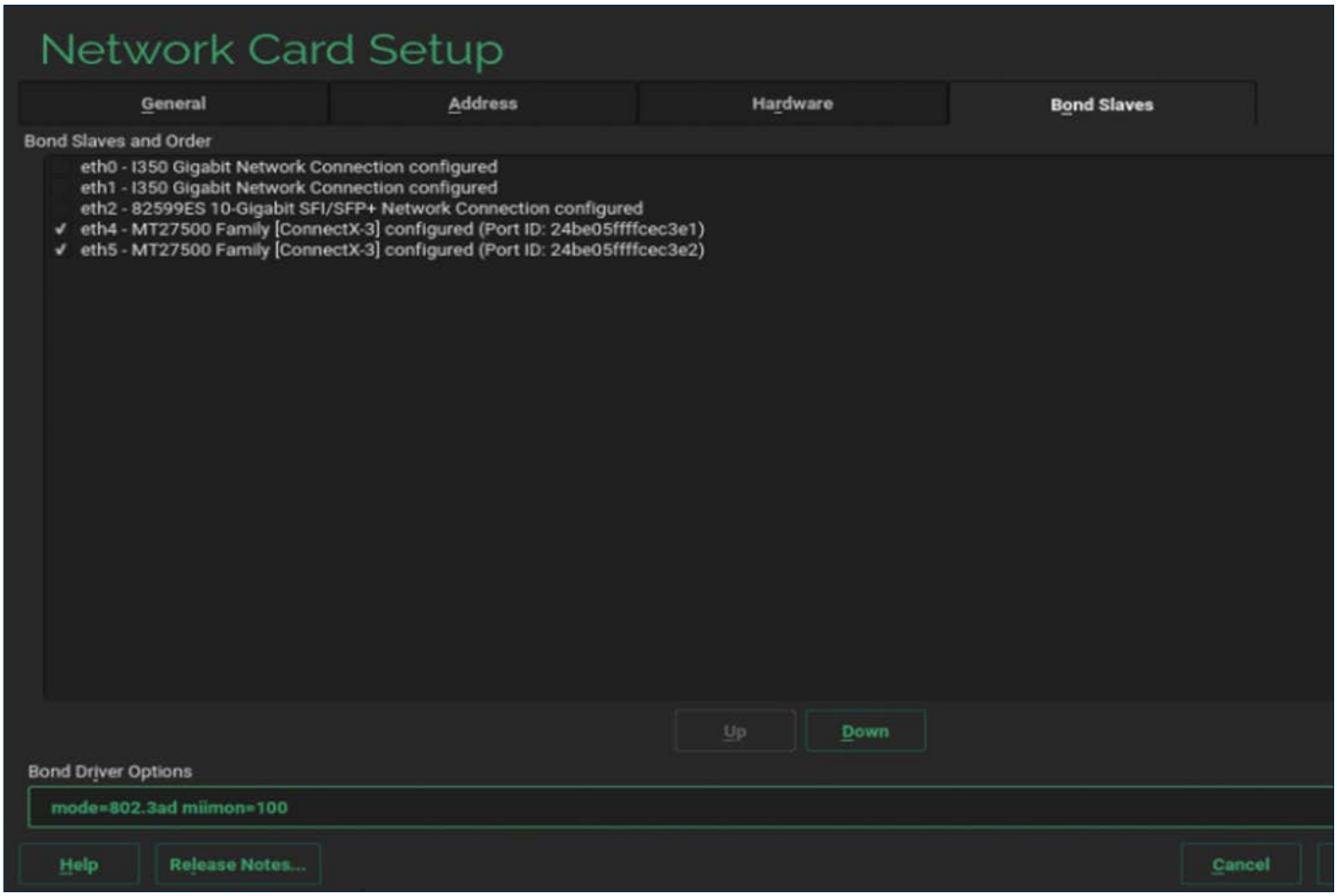

Figure 4.

Add the VLAN interfaces, making sure to select the correct VLAN ID and setting the IP and host information.

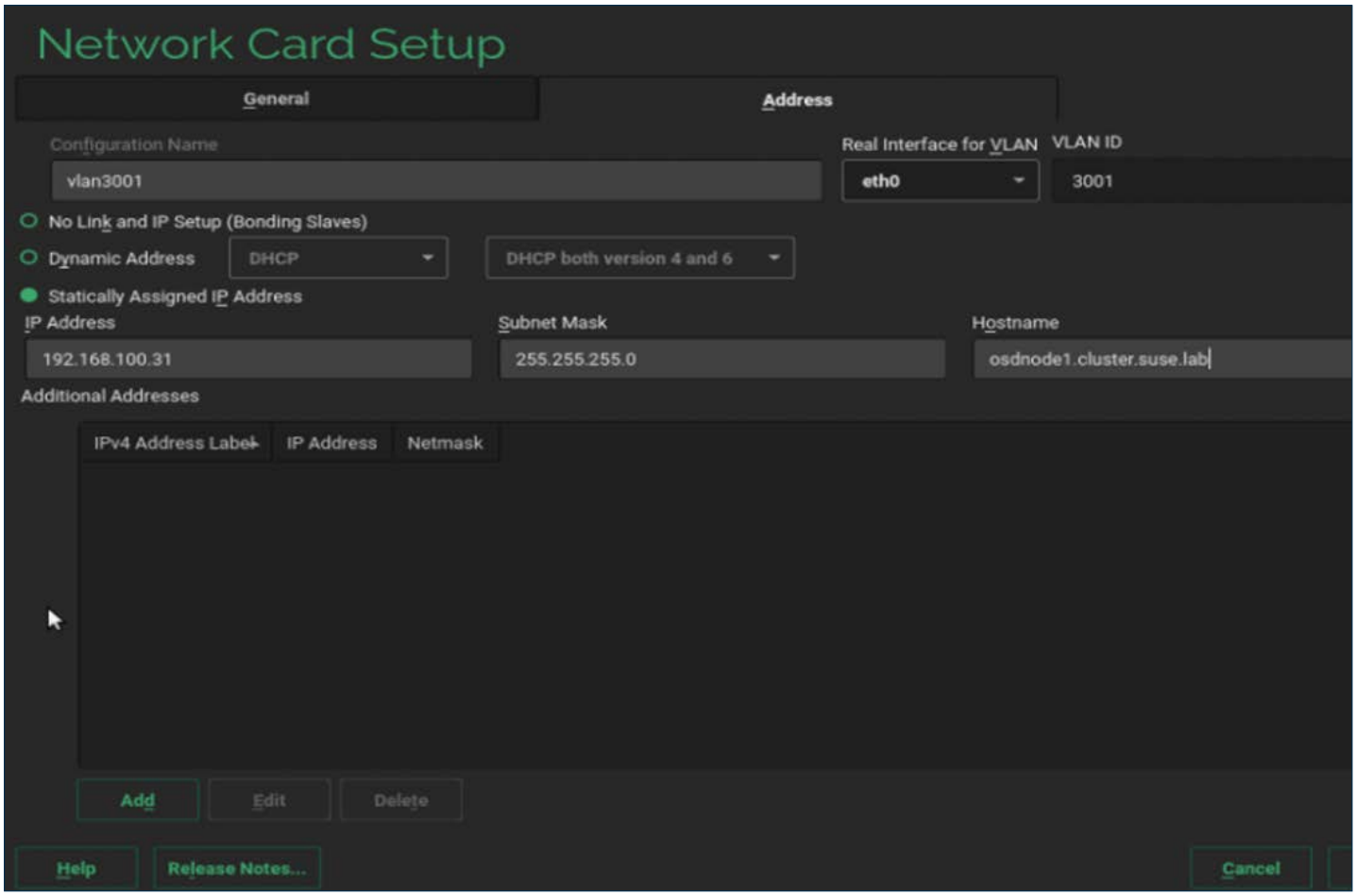

Figure 5.

This figure represents the proper network configuration for osdnode1 as configured in this guide.

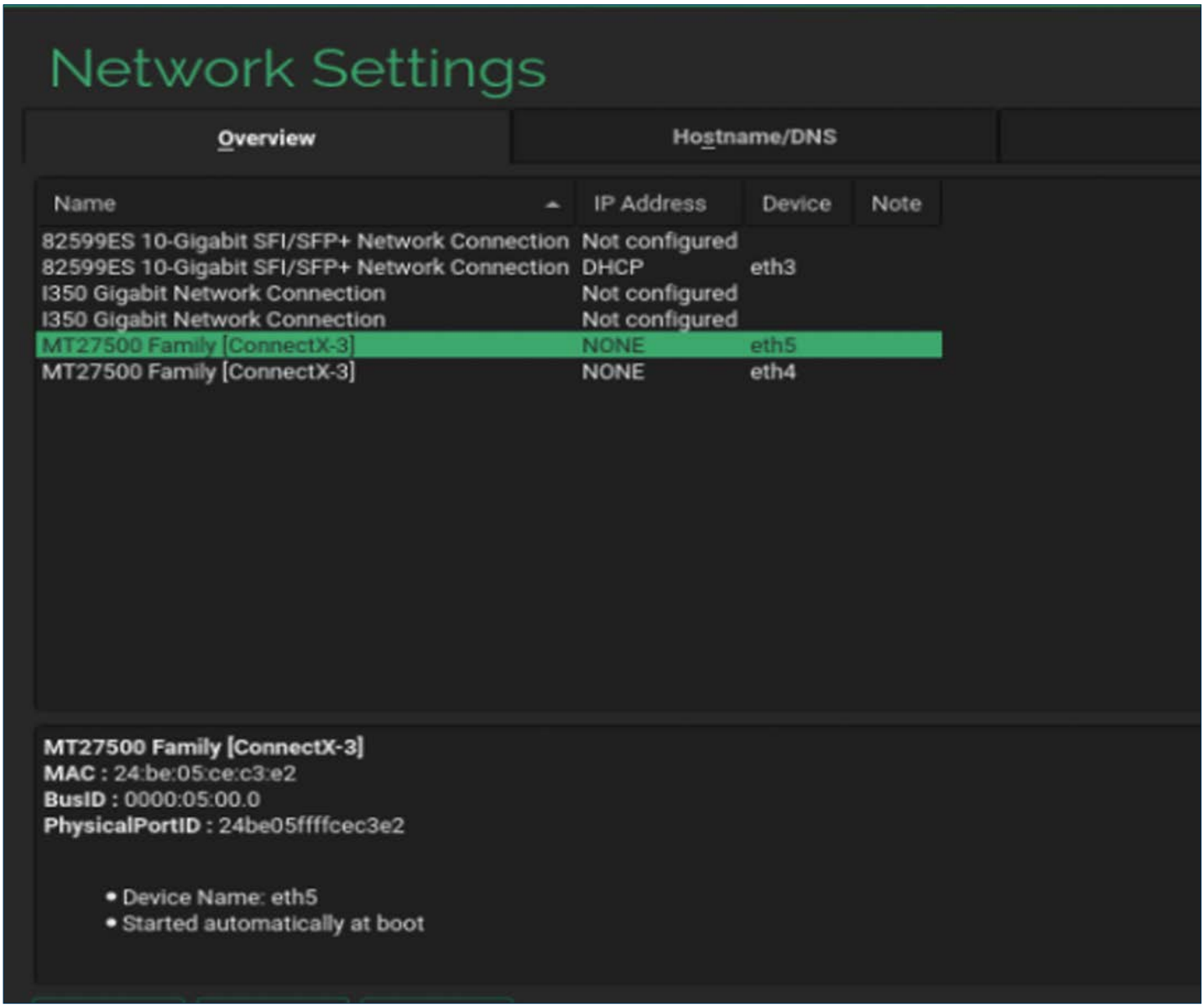

Figure 6.

Appendix E: Performance Data

Comprehensive performance baselines are run as part of a reference build activity. This activity yields a vast amount of information that can be used to approximate the size and performance of the solution. The only tuning applied is documented in the implementation portion of this document.

The tests comprise a number of Flexible I/O (fio) job files that are run against multiple worker nodes. The job files and testing scripts can be found for review at: **https://github.com/dmbyte/benchmaster**. This is a personal repository. No warranties are made in regard to the fitness and safety of the scripts found there.

The testing methodology involves two different types of long-running tests. The types and duration of the tests have very specific purposes. There are both i/o simulation jobs and single metric jobs.

The length of the test run, in combination with the ramp-up time specified in the job file, is intended to allow the system to overrun caches. This is a worst-case scenario for a system and would indicate that it is running at or near capacity. Given that few applications can tolerate significant amounts of long tail latencies, the job files have specific latency targets assigned. These targets are intended to be in line with expectations for the type of I/O operation being performed and they set realistic expectations for the application environment.

The latency target, when combined with the latency window and latency window percentage set the minimum number of I/Os that must be within the latency target during the latency window time period. For most of the tests, the latency target is 20ms of less. The latency window is five seconds and the latency target is 99.99999%. The way that fio uses this is to ramp up the queue depth at each new latency window boundary until more than .00001% of all I/O's during a five-second window are higher than 20ms. At that point, fio backs the queue depth down to where the latency target is sustainable.

In the figures below, the x-axis labels indicate the block size in KiB on the top line and the data protection scheme on the bottom line. 3xrep is indicative of the Ceph standard 3 replica configuration for data protection, while EC2+2 is Erasure Coded using the ISA plugin with k=2 and m=2. The Erasure Coding settings were selected to fit within the minimum cluster hardware size supported by SUSE.

These settings, along with block size, max queue depth, jobs per node, etc., are all visible in the job files found at the repository link above.

Load testing was provided by two additional HPE Apollo 4510 Gen10 servers on the same 100GbE network.

Sequential Writes

Sequential write I/O testing was performed across block sizes ranging from 4KiB to 4MiB.

These tests have associated latency targets: 4k is 10ms, 64k is 20ms, 1MiB is 100ms and 4MiB is 300ms.

CEPHFS SEQUENTIAL WRITES

Data Protection	I/O Size (KiB)	Write BW (MiB/s)	Write IOPS	Write Avg Latency (ms)
3xrep	4	106	27391	1.7
EC2+2	4	7	1844	24.0
3xrep	64	365	5856	8.2
EC2+2	64	231	3702	12.3
3xrep	1024	2283	2284	20.9
EC2+2	1024	1015	1016	45.4
3xrep	4096	1958	490	56.7
EC2+2	4096	2177	544	85.7

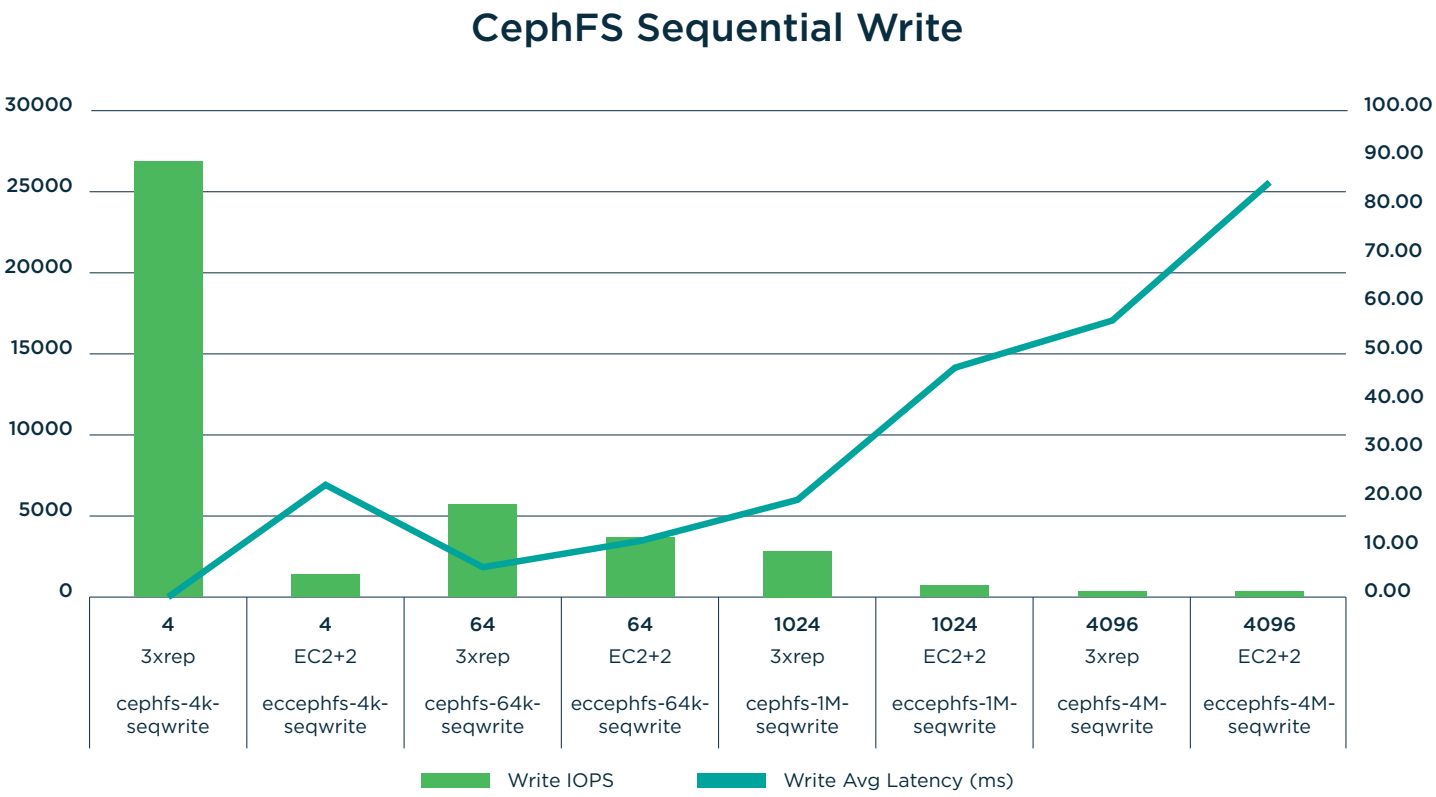

Figure 7.

RBD SEQUENTIAL WRITES

Data Protection	I/O Size (KiB)	Write BW (MiB/s)	Write IOPS	Write Avg Latency (ms)
3xrep	4	129	33048	0.5
3xrep	64	337	5395	1.4
EC2+2	64	202	3240	30.5
EC2+2	1024	836	837	400.7
3xrep	1024	2001	2001	333.3
3xrep	4096	1307	327	1135.6
EC2+2	4096	2720	680	1045.7

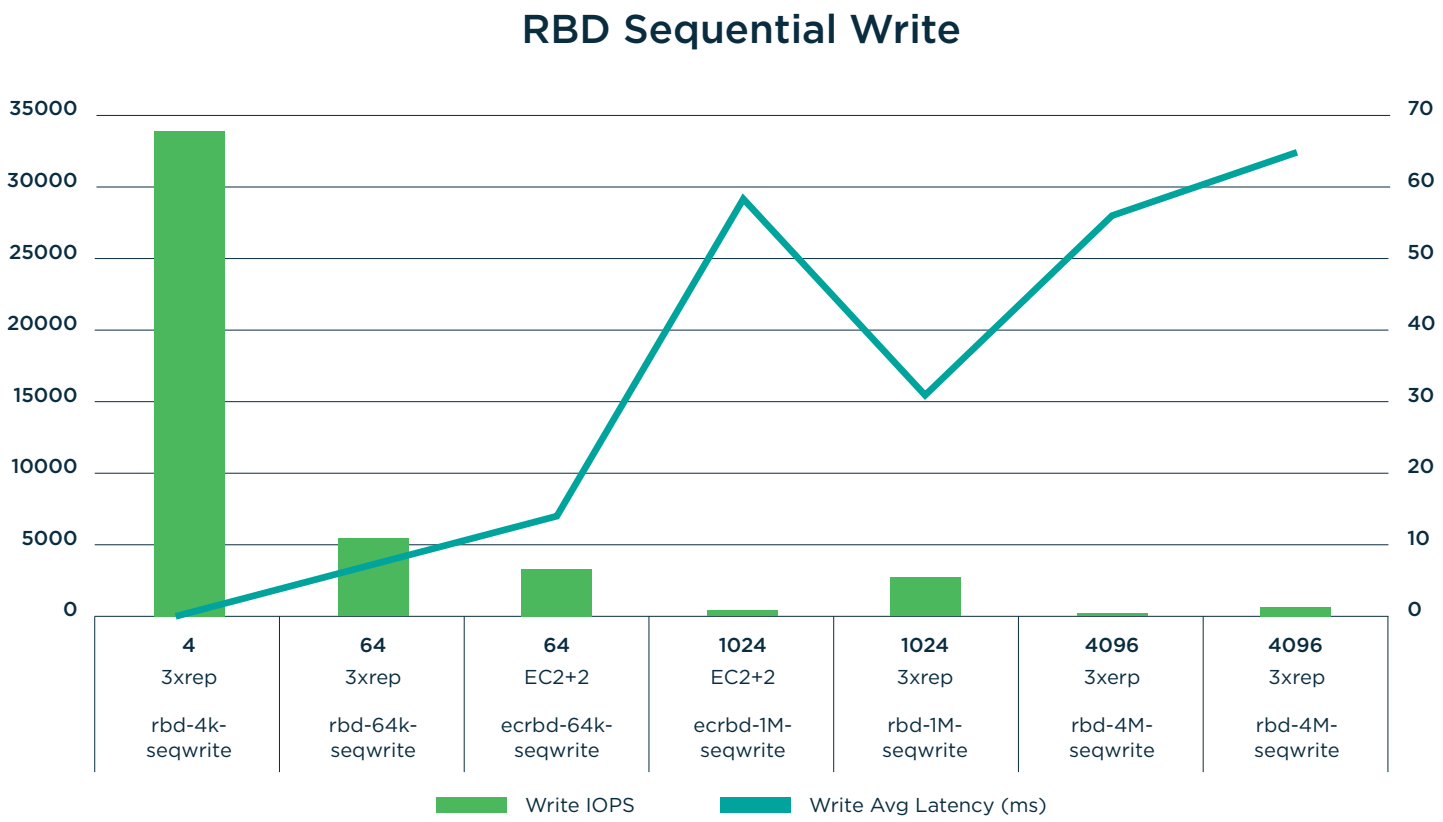

Figure 8.

Sequential Reads

The sequential read tests were conducted across the same range of block sizes as the write testing. The latency targets are only present for 4k sequential reads, where it is set to 10ms.

CEPHFS SEQUENTIAL READS

Data Protection	I/O Size (KiB)	Read BW (MiB/s)	Read IOPS	Read Avg Latency (ms)
3xrep	4	266	68309	0.5
EC2+2	4	132	34028	1.4
EC2+2	64	2192	35085	30.5
3xrep	1024	3769	3769	400.7
EC2+2	1024	4372	4372	333.3
3xrep	4096	5020	1254	1135.6
EC2+2	4096	5602	1400	1045.7

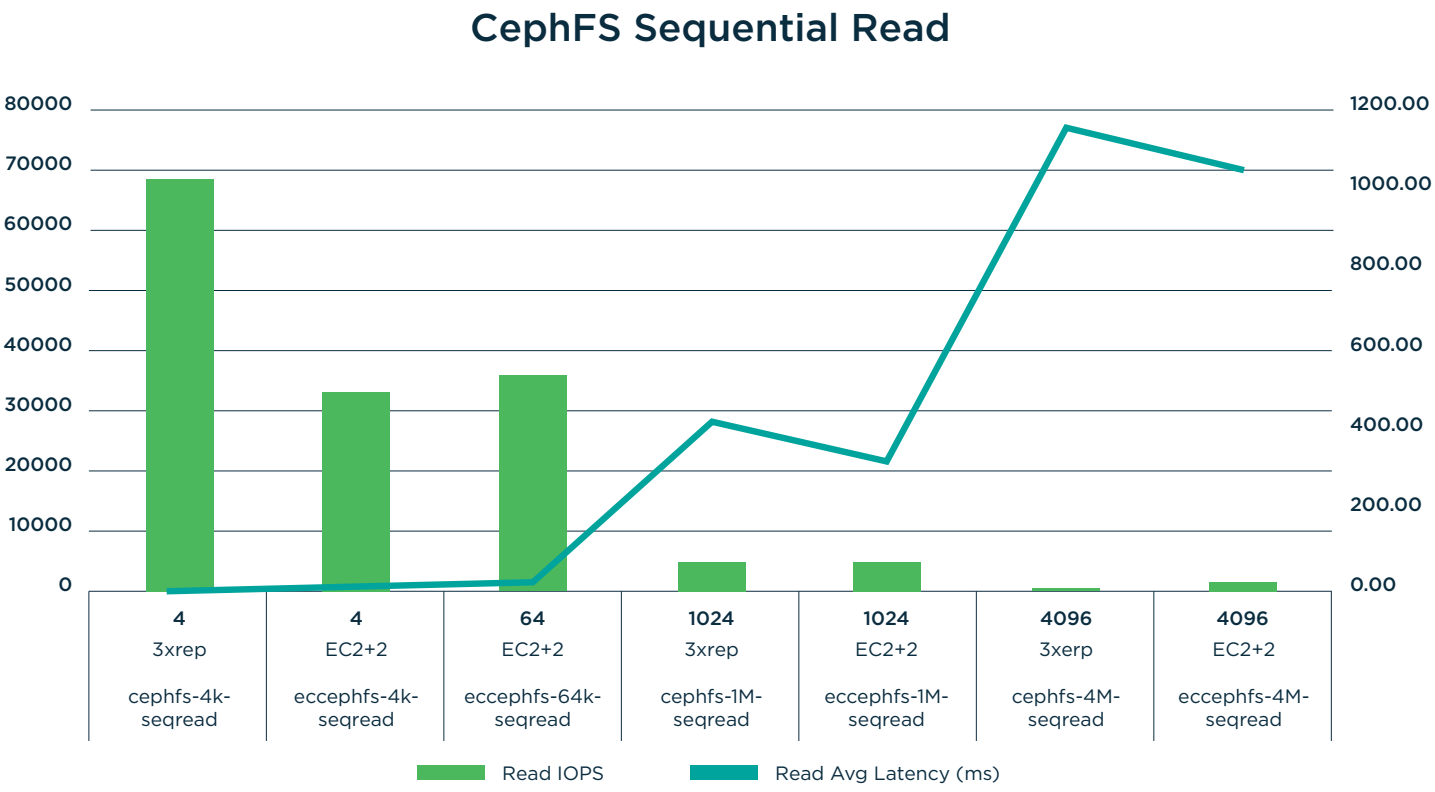

Figure 9.

RBD SEQUENTIAL READS

Data Protection	I/O Size (KiB)	Read BW (MiB/s)	Read IOPS	Read Avg Latency (ms)	Read Max Latency (ms)
3xrep	4	205	52659	0.6	25635
3xrep	64	2631	42105	12.2	19987
EC2+2	64	2960	47371	32.2	1415
EC2+2	1024	3181	3180	452.1	55320
3xrep	1024	7976	7976	192.2	123841
3xrep	4096	10180	2544	522.1	59084
EC2+2	4096	5026	1256	1247.5	195588

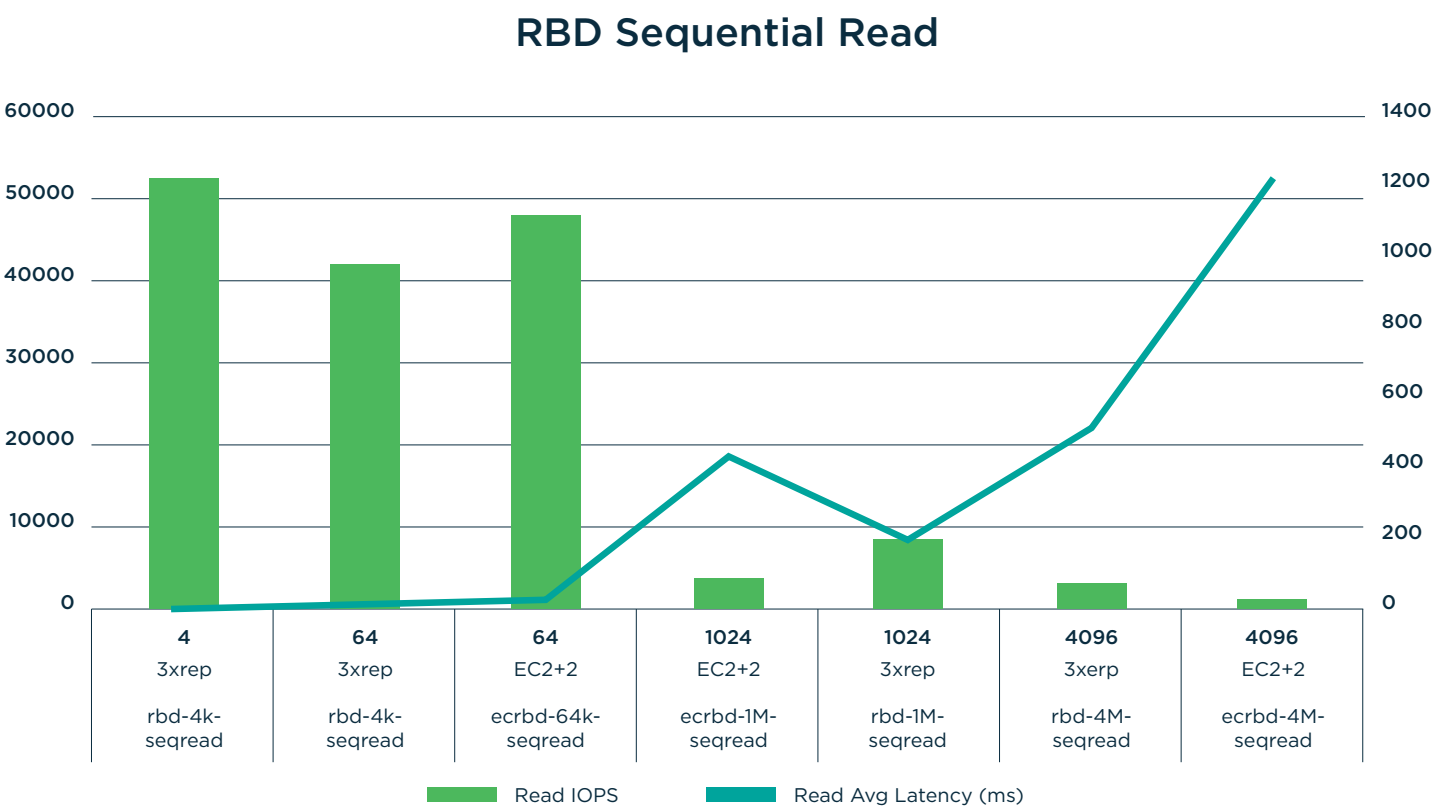

Figure 10.

Random Writes

Random write tests were performed with the smaller I/O sizes of 4k and 64k. The 4k tests have a latency target of 10ms and the 64k tests have a latency target of 20ms.

CEPHFS RANDOM WRITES

Data Protection	I/O Size (KiB)	Write BW (MiB/s)	Write IOPS	Write Avg Latency (ms)	Write Max Latency (ms)
3xrep	4	33	8664	5.5	647
EC2+2	4	11	2818	17.0	276
3xrep	64	304	4880	9.8	238
EC2+2	64	137	2204	21.7	400

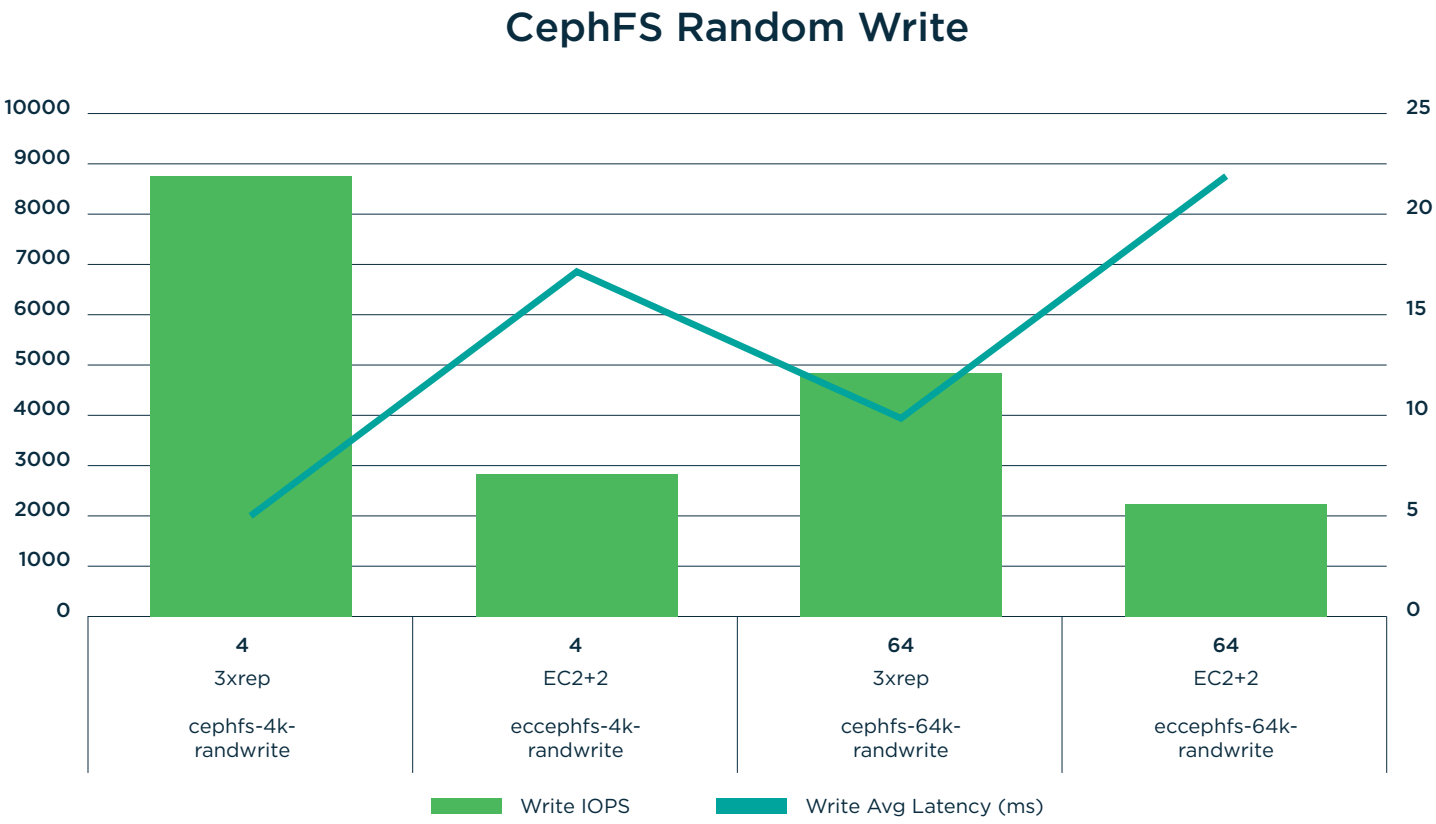

Figure 11.

RBD RANDOM WRITES

Data Protection	I/O Size (KiB)	Write BW (MiB/s)	Write IOPS	Write Avg Latency (ms)	Write Max Latency (ms)
3xrep	4	32	8225	5.8	3920
3xrep	64	265	4241	11.3	2194
EC2+2	64	127	2036	23.6	2669

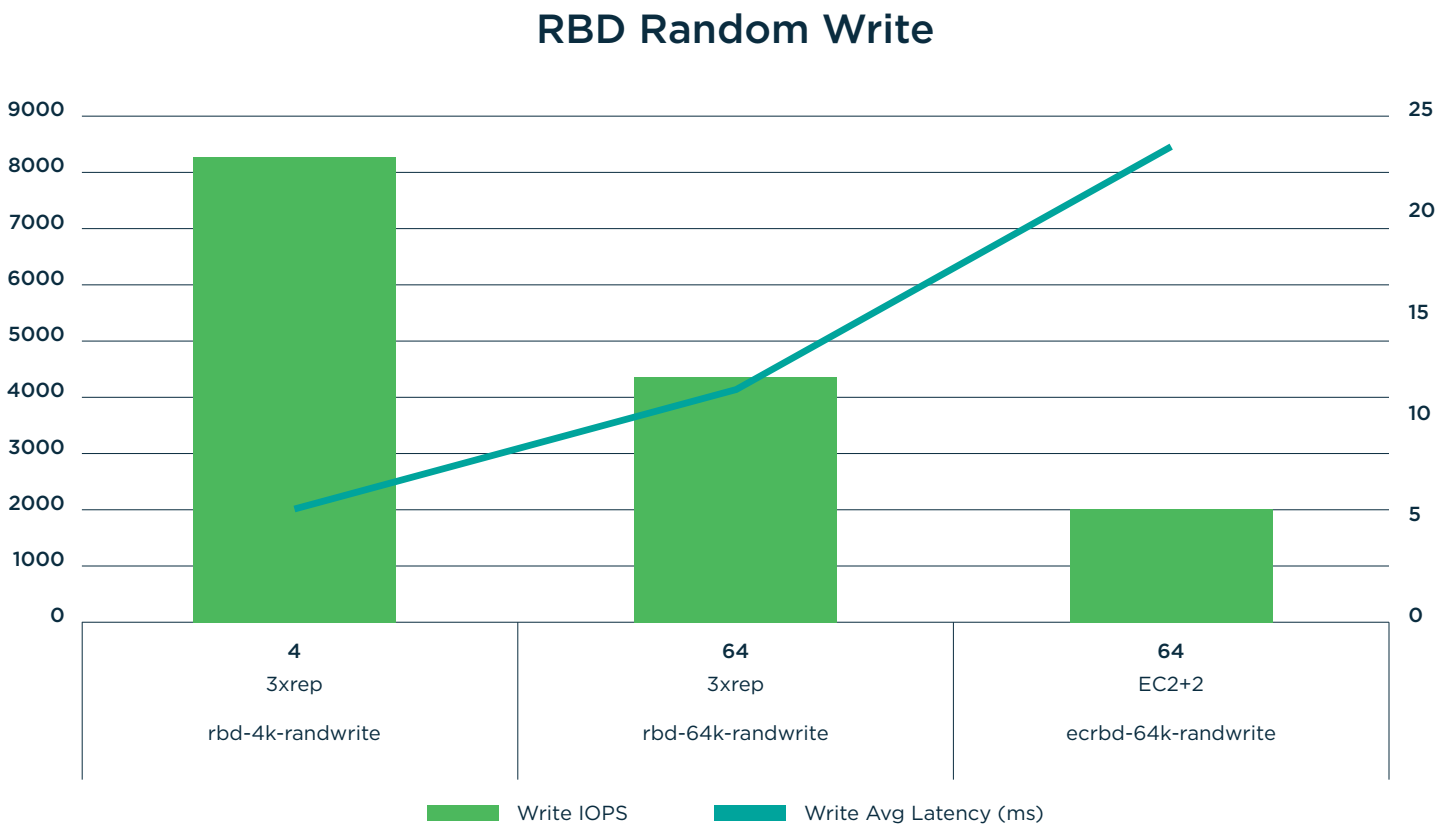

Figure 12.

Random Reads

The random read tests were conducted on both 4k and 64k I/O sizes with latency targets of 10ms and 20ms respectively.

CEPHFS RANDOM READS

Data Protection	I/O Size (KiB)	Read BW (MiB/s)	Read IOPS	Read Avg Latency (ms)	Read Max Latency (ms)
3xrep	4	23	5991	8.0	159
EC2+2	4	16	4249	11.3	208
3xrep	64	340	5450	8.8	257
EC2+2	64	244	3914	12.3	219

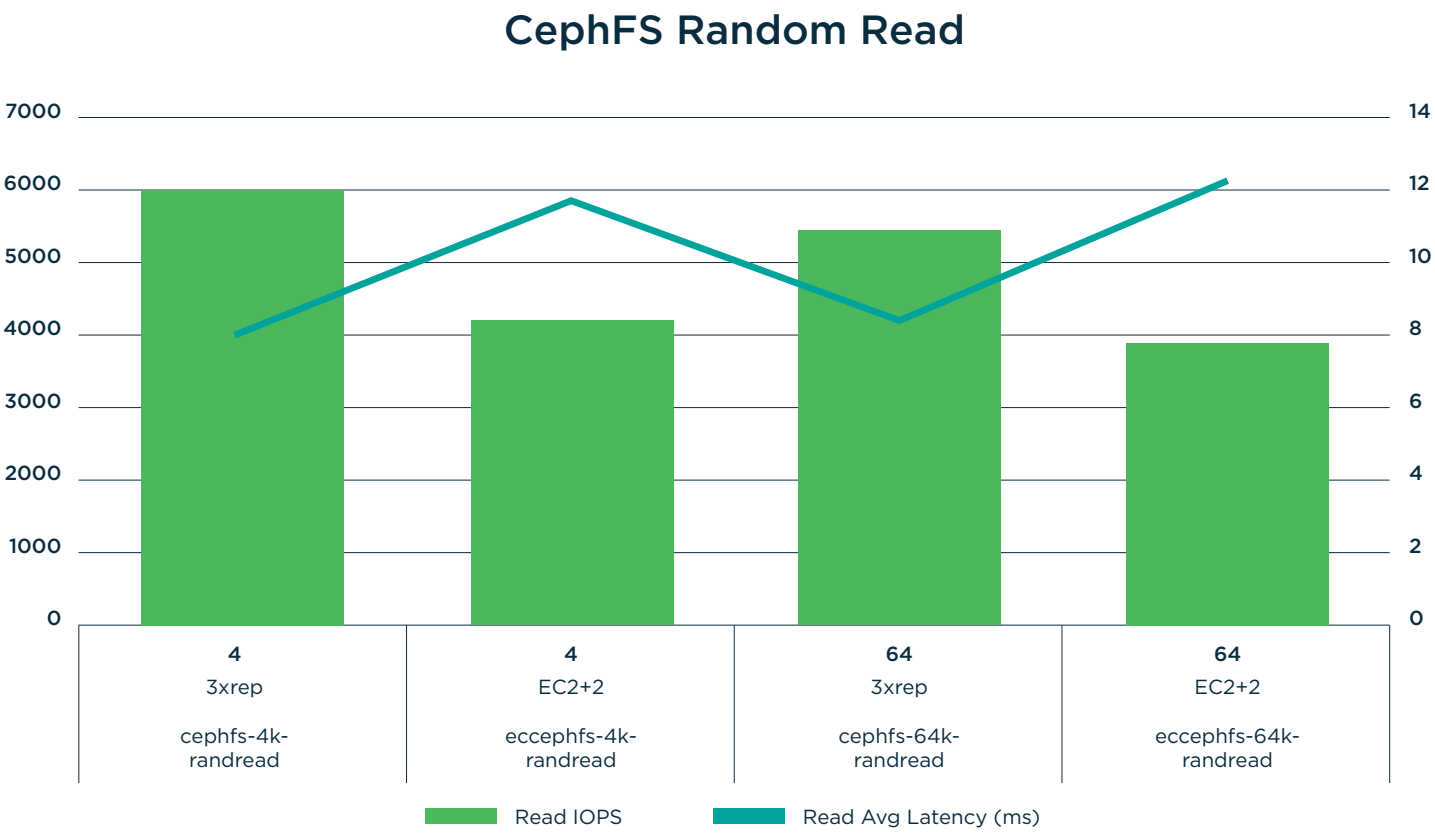

Figure 13.

RBD RANDOM READS

Data Protection	I/O Size (KiB)	Read BW (MiB/s)	Read IOPS	Read Avg Latency (ms)	Read Max Latency (ms)
3xrep	4	17	4407	10.9	14423
3xrep	64	254	4075	11.8	1930
EC2+2	64	220	3523	13.6	23781

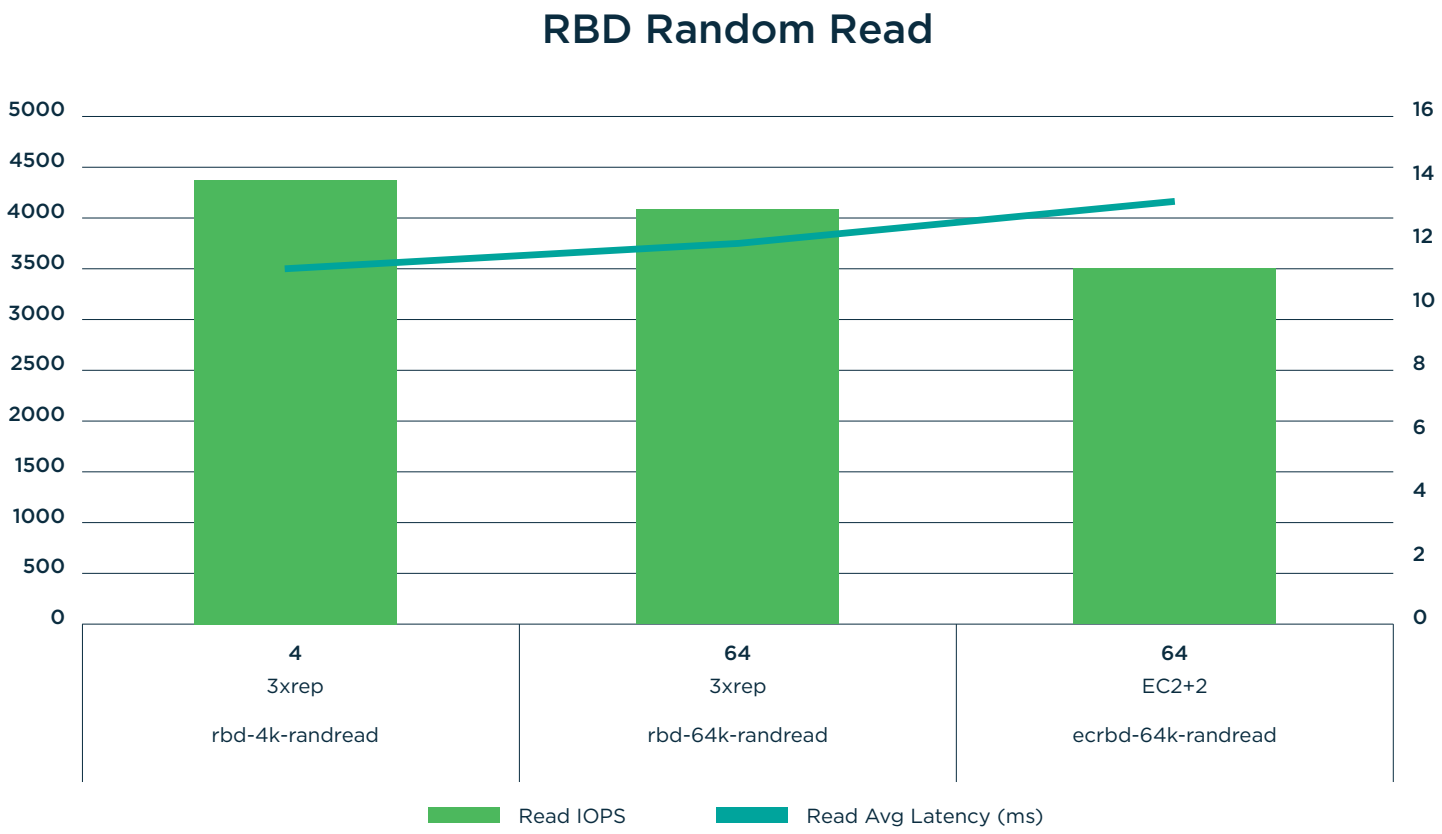

Figure 14.

Workload Simulations

The following test results are workload oriented.

BACKUP SIMULATION

The backup simulation test attempts to simulate the SUSE Enterprise Storage cluster being used as a disk-based backup target that is either hosting file systems on RBDs or is using CephFS. The test had a latency target of 200ms at the time of the test run. The latency target has since been removed.

Data Protection	Protocol	Write BW (MiB/s)	Write IOPS	Write Avg Latency (ms)
3xrep	CephFS	2559	39	35579
EC2+2	CephFS	3562	55	34318
3xrep	RBD	3058	47	30973
EC2+2	RBD	3334	51	30357

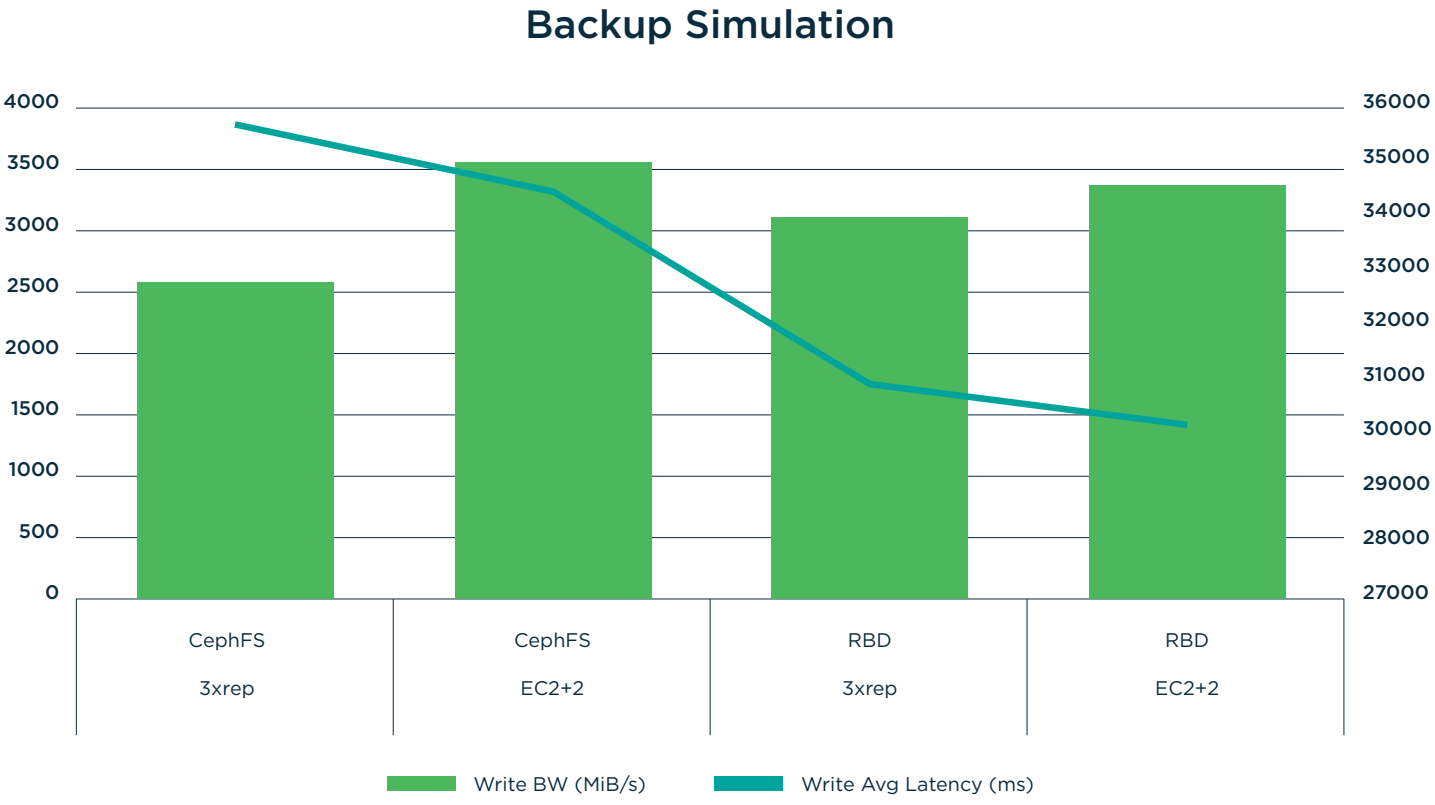

Figure 15.

RECOVERY SIMULATION

The recovery workload is intended to simulate recovery jobs being run from SUSE Enterprise Storage. It tests both RBD and CephFS.

Data Protection	Protocol	Read BW (MiB/s)	Read IOPS	Read Avg Latency (ms)
3xrep	CephFS	3428	53	27719
EC2+2	CephFS	4480	69	23554
3xrep	RBD	6983	108	14183
EC2+2	RBD	4361	67	23104

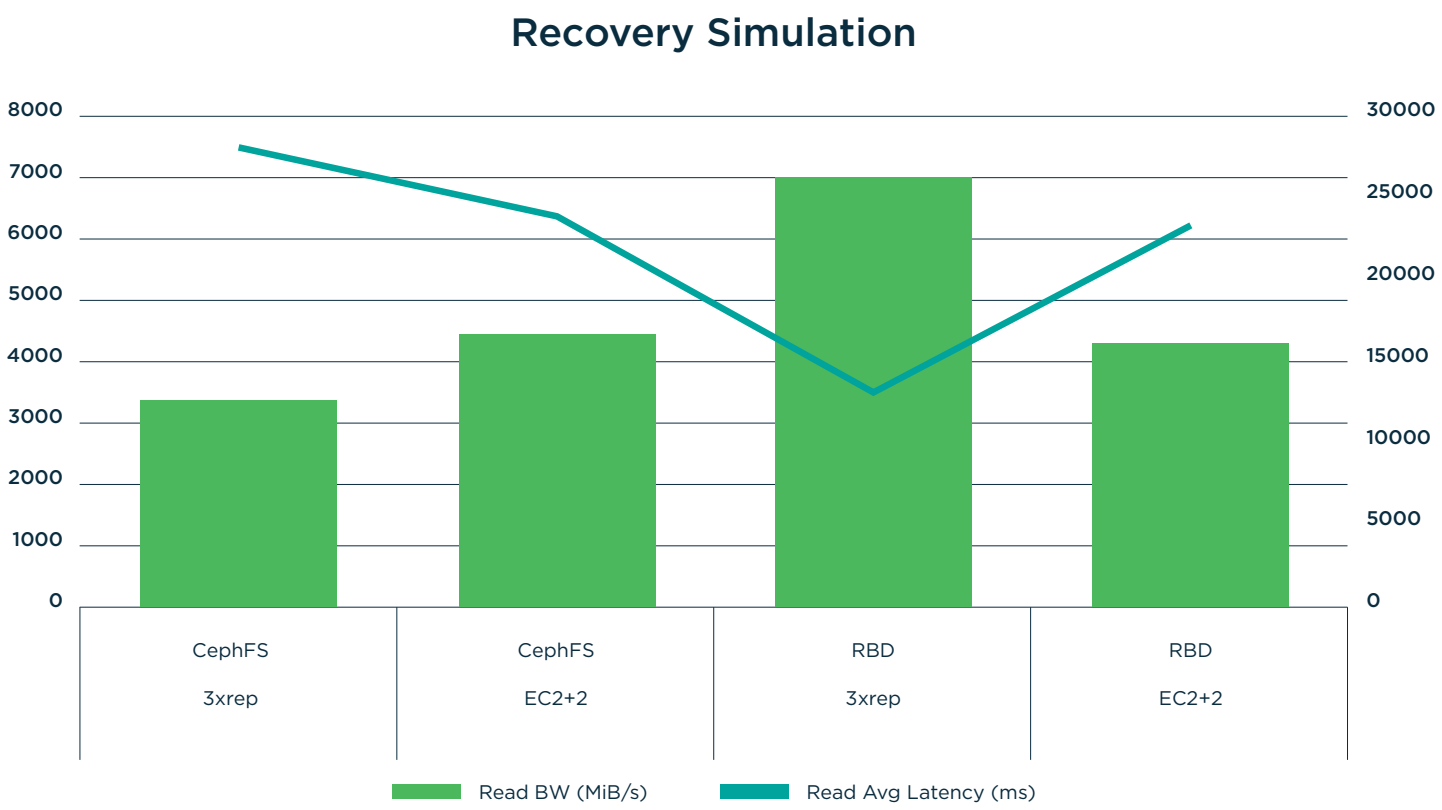

Figure 16.

KVM VIRTUAL GUEST SIMULATION

The kvm-krbd test roughly simulates virtual machines running. This test has a 20ms latency target and is 80% read with both reads and writes being random.

Write BW (MiB/s)	Write IOPS	Write Avg Latency (ms)	Data Protection	Protocol	Read Avg Latency (ms)	Read IOPS	Read BW (MiB/s)
5	1394	2.1	3xrep	CephFS	8.1	5535	21
3	778	14.6	EC2+2	CephFS	11.8	3108	12
4	1050	3.9	3xrep	RBD	10.5	4181	16

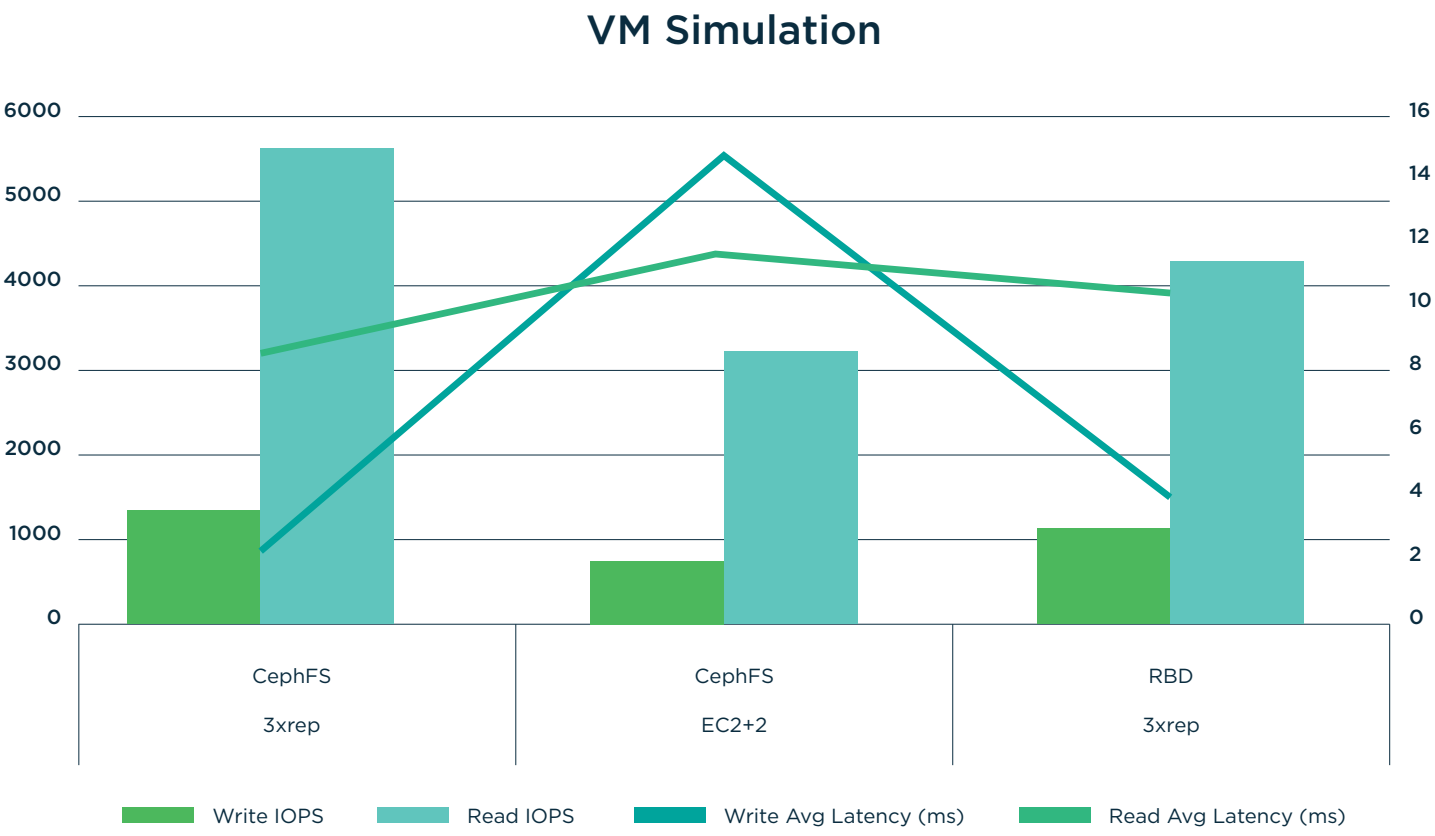

Figure 17.

Database Simulations

It is important to keep sight of the fact that Ceph is not designed for high performance database activity. These tests provide a baseline understanding of performance expectations should a database be deployed using SUSE Enterprise Storage.

OLTP DATABASE LOG

The database log simulation is based on documented I/O patterns from several major database vendors. The I/O profile is 80% sequential 8KB writes with a latency target of 1ms.

Write BW (MiB/s)	Write IOPS	Write Avg Latency (ms)	Data Protection	Protocol	Read Avg Latency (ms)	Read IOPS	Read BW (MiB/s)
215	27571	1.5	3xrep	CephFS	0.6	6888	53
20	2637	15.1	EC2+2	CephFS	11.5	655	5
202	25887	1.6	3xrep	RBD	0.6	6463	50

OLTP DATABASE DATAFILE

The OLTP Datafile simulation is set for an 80/20 mix of random reads and writes. The latency target is 10ms.

Write BW (MiB/s)	Write IOPS	Write Avg Latency (ms)	Data Protection	Protocol	Read Avg Latency (ms)	Read IOPS	Read BW (MiB/s)
10	1375	2.2	3xrep	CephFS	8.2	5458	42
5	662	16.6	EC2+2	CephFS	13.9	2659	20
9	1175	2.2	3xrep	RBD	9.8	4651	36

Resources

SUSE Enterprise Storage Technical Overview
www.suse.com/docrep/documents/1mdg7eq2kz/suse_enterprise_storage_technical_overview_wp.pdf

SUSE Enterprise Storage v5—Administration and Deployment Guide
www.suse.com/documentation/suse-enterprise-storage-5/book_storage_deployment/data/book_storage_deployment.html

SUSE Linux Enterprise Server 12 SP3—Administration Guide
www.suse.com/documentation/sles-12/book_sle_admin/data/book_sle_admin.html

Subscription Management Tool for SLES 12 SP3
www.suse.com/documentation/sles-12/book_smt/data/book_smt.html

HPE MLNX-OS for HPE StoreFabric M-Series Ethernet Switch User Manual
https://support.hpe.com/hpsc/doc/public/display?sp4ts.oid=1010292242&docLocale=en_US&docId=emr_na-a00027007en_us

Additional contact information and office locations:
www.suse.com

www.suse.com