

Cisco UCS S3260 M5 Server with Cloudean HyperStore Object Storage

Deployment Guide for Cloudean HyperStore Software on
Cisco UCS S3260 M5

Last Updated: June 7, 2019



About the Cisco Validated Design Program

The Cisco Validated Design (CVD) program consists of systems and solutions designed, tested, and documented to facilitate faster, more reliable, and more predictable customer deployments. For more information, go to:

<http://www.cisco.com/go/designzone>.

ALL DESIGNS, SPECIFICATIONS, STATEMENTS, INFORMATION, AND RECOMMENDATIONS (COLLECTIVELY, "DESIGNS") IN THIS MANUAL ARE PRESENTED "AS IS," WITH ALL FAULTS. CISCO AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE. IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THE DESIGNS, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

THE DESIGNS ARE SUBJECT TO CHANGE WITHOUT NOTICE. USERS ARE SOLELY RESPONSIBLE FOR THEIR APPLICATION OF THE DESIGNS. THE DESIGNS DO NOT CONSTITUTE THE TECHNICAL OR OTHER PROFESSIONAL ADVICE OF CISCO, ITS SUPPLIERS OR PARTNERS. USERS SHOULD CONSULT THEIR OWN TECHNICAL ADVISORS BEFORE IMPLEMENTING THE DESIGNS. RESULTS MAY VARY DEPENDING ON FACTORS NOT TESTED BY CISCO.

CCDE, CCENT, Cisco Eos, Cisco Lumin, Cisco Nexus, Cisco StadiumVision, Cisco TelePresence, Cisco WebEx, the Cisco logo, DCE, and Welcome to the Human Network are trademarks; Changing the Way We Work, Live, Play, and Learn and Cisco Store are service marks; and Access Registrar, Aironet, AsyncOS, Bringing the Meeting To You, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, CCVP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unified Computing System (Cisco UCS), Cisco UCS B-Series Blade Servers, Cisco UCS C-Series Rack Servers, Cisco UCS S-Series Storage Servers, Cisco UCS Manager, Cisco UCS Management Software, Cisco Unified Fabric, Cisco Application Centric Infrastructure, Cisco Nexus 9000 Series, Cisco Nexus 7000 Series, Cisco Prime Data Center Network Manager, Cisco NX-OS Software, Cisco MDS Series, Cisco Unity, Collaboration Without Limitation, EtherFast, EtherSwitch, Event Center, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, iQuick Study, LightStream, Linksys, MediaTone, MeetingPlace, MeetingPlace Chime Sound, MGX, Networkers, Networking Academy, Network Registrar, PCNow, PIX, PowerPanels, ProConnect, ScriptShare, SenderBase, SMARTnet, Spectrum Expert, StackWise, The Fastest Way to Increase Your Internet Quotient, TransPath, WebEx, and the WebEx logo are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0809R)

© 2019 Cisco Systems, Inc. All rights reserved.

Table of Contents

Executive Summary	7
Solution Overview	8
Introduction.....	8
Audience	8
Purpose of this Document.....	8
Solution Summary	8
Technology Overview	10
Cisco Unified Computing System	10
Cisco UCS Manager.....	10
Cisco UCS 6300 Fabric Interconnects.....	11
Cisco UCS C9336C-FX2 Nexus Switches	12
Cisco UCS S3260 M5 Storage Server	13
Cisco UCS C220 M5 Rack-Mount Server	14
Cisco UCS Virtual Interface Card 1387.....	15
Red Hat Enterprise Linux 7.5.....	15
Cloudian HyperStore.....	17
Cloudian Object Storage.....	17
Cloudian HyperStore Design	18
Cloudian HyperStore Architecture	18
Cloudian Management Console.....	19
S3 Compatible.....	20
Integrated Billing, Management, and Monitoring	20
Cloudian Features.....	21
Solution Design	22
Deployment Architecture	22
System Hardware and Software Specifications.....	24
Solution Overview.....	24
Software Versions	24
Hardware Requirements and Bill of Materials	25
Physical Topology and Configuration	26
Network Topology	31
High Availability	31
Deployment Hardware and Software	33
Configuration of Nexus C9336C-FX2 Switch A and B	33
Initial Setup of Nexus C9336C-FX2 Switch A and B	33
Enable Features on Nexus C9336C-FX2 Switch A and B.....	36
Configure VLANs on Nexus C9336C-FX2 Switch A and B	36
Verification Check of Nexus C9336C-FX2 Configuration for Switch A and B.....	46
Fabric Interconnect Configuration.....	49

Initial Setup of Cisco UCS 6332 Fabric Interconnects.....	49
Configure Fabric Interconnect A	49
Configure Fabric Interconnect B.....	52
Log into Cisco UCS Manager.....	53
Configure NTP Server	54
Initial Base Setup of the Environment	55
Configure Global Policies	55
Enable Fabric Interconnect Server Ports.....	56
Enable Fabric Interconnect A Ports for Uplinks	57
Label Servers for Identification	58
Create KVM IP Pool.....	59
MAC Pool.....	60
Create UUID Pool.....	61
Create VLANs	63
Enable CDP	64
QoS System Class	65
vNIC Template Setup	66
Ethernet Adapter Policy Setup	69
Boot Policy Setup.....	71
Create LAN Connectivity Policy Setup	72
Create Maintenance Policy Setup	73
Create Chassis Profile	74
Create Chassis Firmware Package	74
Create Chassis Maintenance Policy	75
Create Disk Zoning Policy	76
Create Chassis Profile Template	80
Create Chassis Profile from Template.....	82
Associate Chassis Profile	83
Create Storage Profiles	84
Set Disks for Cisco UCS S3260 M5 Servers to Unconfigured-Good.....	84
Create Storage Profiles for Cisco UCS S3260 Storage Server	84
Create Storage Profile for Cisco UCS C220 M5 Rack-Mount Servers.....	87
Create a Service Profile Template for Cisco UCS S3260 Storage Server	91
Create Service Profile Template for Cisco UCS S3260 Storage Server1 and Server2	91
Identify Service Profile Template	91
Storage Provisioning	92
Networking.....	93
vNIC/vHBA Placement	94
Server Boot Order.....	95
Maintenance Policy	96
Operational Policies.....	97

Create Service Profiles from Template	97
Associate a Service Profile for Cisco UCS S3260 M5 Server	99
Create Service Profile for Cisco UCS C220 M5 Server for HA-Proxy Node	101
Identify Service Profile	101
Storage Provisioning	102
Networking	103
vNIC/vHBA Placement	104
Server Boot Order	105
Maintenance Policy	106
Operational Policies	108
Create Port Channel for Network Uplinks	108
Create Port Channel for Fabric Interconnect A/B	108
Install Red Hat Enterprise Linux 7.5 Operating System	110
Install RHEL 7.5 on Cisco UCS C220 M5 and Cisco UCS S3260 M5 Server	110
Cloudian HyperStore Preparation	112
Software Version	112
Load-Balancer Requirements	112
Concepts of Load Balancing	113
Round-Robin DNS	113
Layer-4 Load Balancing	114
Layer-7 Load Balancing	115
Direct Routing	116
Global Server Load Balancing	117
High Availability	118
Load Balancing HyperStore	120
HyperStore Services	120
HyperStore Configuration	123
HAProxy Examples	123
HAProxy - Basic Configuration	123
DNS Requirements	129
Prepare the Master Node	130
Network Best Practices	132
Create the survey.csv File	133
Prepare Cluster Nodes	135
Cloudian HyperStore Installation	139
Software Installation	139
Generate HTTPS Certificate and Signing Request	141
Import SSL Certificate in Keystore	142
Enable HTTPS Access on s3	143
Cloudian HyperStore Configuration	147
Log into the Cloudian Management Console (CMC)	147

Create a Storage Policy.....	148
Setup Alerts and Notifications.....	151
Create a Group and User	153
Create Buckets.....	155
Verify Credentials and Service Endpoints as a User.....	157
Cloudian HyperStore Installation Verification	158
Verify HyperStore S3 Connectivity	158
Add Additional Data Center and Nodes	162
Prepare the New Nodes.....	162
Add a New DC.....	163
Create a Multi-DC Storage Policy	167
Performance	172
Erasure Code 4+2 - Read Performance.....	172
Erasure Code 4+2 - Write Performance	173
3-Way Replication - Read Performance	174
3-Way Replication - Write Performance	175
Replicated Erasure Code 4+2 - Read Performance.....	176
Replicated Erasure Code 4+2 - Write Performance	177
2DC 4-Way Replication - Read Performance	178
2DC 4-Way Replication - Write Performance	179
High Availability Tests.....	181
Fabric Interconnect Failures.....	181
Nexus 9000 Switch Failures.....	185
S3 Service Failures.....	186
Disk Failure Tests.....	190
Frequently Asked Questions	195
Troubleshooting	197
Appendix.....	198
Appendix A - Kickstart File of High Available Proxy Node for Cisco UCS C220 M5	198
Appendix B - Kickstart File of Storage Nodes for Cisco UCS S3260 M5 Server	202
Summary	208
About the Authors.....	209
Acknowledgements	209



Executive Summary

Cisco Validated Designs (CVDs) consist of systems and solutions that are designed, tested, and documented to facilitate and improve customer deployments. These designs incorporate a wide range of technologies and products into a portfolio of solutions that have been developed to address the business needs of our customers.

Most of the modern data centers are moving away from traditional file system type storage, to object storages. Object storage offers simple management, unlimited scalability and custom metadata for objects. With its low cost per gigabyte of storage, Object storage systems are suited for archive, backup, Life sciences, video surveillance, healthcare, multimedia, message and machine data, and so on.

Cisco and Cloudian are collaborating to offer customers a scalable object storage solution for unstructured data that integrates Cisco Unified Computing System (Cisco UCS) with Cloudian HyperStore. With the power of the Cisco UCS management framework, the solution is cost effective to deploy and manage and will enable the next-generation cloud deployments that drive business agility, lower operational costs and avoid vendor lock-in.

This validated design provides the framework for designing and deploying Cloudian HyperStore 7.1.2 on Cisco UCS S3260 M5 Storage Servers. Cisco Unified Computing System provides the compute, network, and storage access components for the Cloudian HyperStore, deployed as a single cohesive system. The reference architecture described in this document is a realistic use case for deploying Cloudian HyperStore object storage on Cisco UCS S3260 Storage Server.

Solution Overview

Introduction

Object storage is a highly scalable system for organizing and storing data objects. Object storage does not use a file system structure, instead it ingests data as objects with unique keys into a flat directory structure and the metadata is stored with the objects instead of hierarchical journal or tree. Search and retrieval is performed using RESTful API's, which uses HTTP verbs such as GETs and PUTs. Most of the newly generated data, about 60 to 80 percent, is unstructured today and new approaches using x86 servers are proving to be more cost effective, providing storage that can be expanded as easily as your data grows. Scale-out Object storage is the newest cost effective approach for handling large amounts of data in the Petabyte and Exabyte range.

The Clouidian HyperStore is a Software-Defined Storage software that is designed to create unbounded scale-out storage systems that accommodates Petabyte scale data from multiple applications and use-cases, including both object and file based applications

Together with Cisco UCS, Clouidian HyperStore can deliver a fully enterprise-ready solution that can manage different workloads and still remain flexible. The Cisco UCS S3260 Storage Server is an excellent platform to use with the main types of Object and File workloads, such as capacity-optimized and performance-optimized workloads. It is best suited for sequential access, as opposed to random access to unstructured data, and to whatever data size. It is essentially designed for Applications, not direct end-users.

This document describes the architecture and Deployment procedures of Clouidian HyperStore software on Cisco UCS S3260 M5. It also uses Cisco UCS C220 M5 Rack-Mount servers for load balancing.

Audience

The audience for this document includes, but is not limited to, sales engineers, field consultants, professional services, IT managers, partner engineers, IT architects, and customers who want to take advantage of an infrastructure that is built to deliver IT efficiency and enable IT innovation. The reader of this document is expected to have the necessary training and background to install and configure Red Hat Enterprise Linux, Cisco Unified Computing System, Cisco Nexus and Cisco UCS Manager, as well as a high-level understanding of Clouidian HyperStore Software and its components. External references are provided where applicable and it is recommended that the reader be familiar with these documents.

Readers are also expected to be familiar with the infrastructure, network and security policies of the customer installation.

Purpose of this Document

This document describes the steps required to deploy Clouidian HyperStore 7.1.2 on the Cisco UCS platform. It explains the deployment choices and best practices using this shared infrastructure platform.

Solution Summary

This solution is focused on Clouidian HyperStore Cluster on Red Hat Linux 7 on Cisco Unified Computing System. The advantages of Cisco UCS and Clouidian HyperStore combine to deliver an object storage solution that is

simple to install, scalable and high performance. The configuration uses the following components for the deployment:

- Cisco Unified Computing System
 - Cisco UCS 6332 Series Fabric Interconnects
 - Cisco UCS S3260 M5 Storage Servers
 - Cisco UCS S3260 System IO Controller with VIC 1380
 - Cisco UCS C220 M5 Servers with VIC 1387
- Cisco Nexus C9336C-FX2 Series Switches
- Clodian HyperStore 7.1.2
- Red Hat Enterprise Linux 7.5

Technology Overview

Cisco Unified Computing System

Cisco Unified Computing System is a state-of-the-art data center platform that unites computing, network, storage access, and virtualization into a single cohesive system.

The main components of Cisco Unified Computing System are:

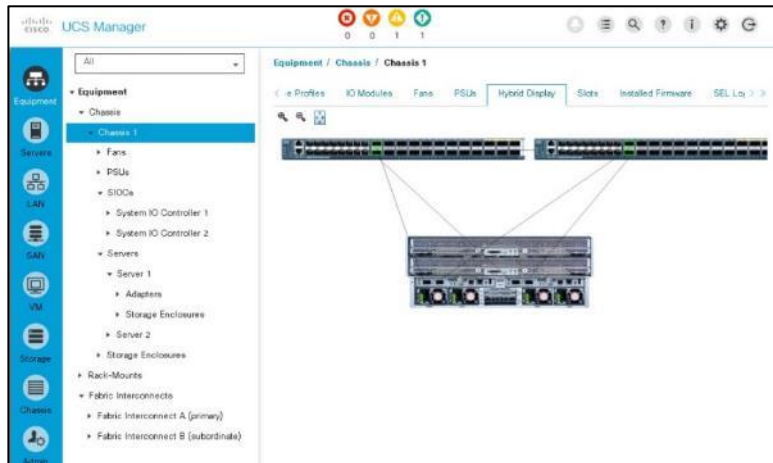
- Computing - The system is based on an entirely new class of computing system that incorporates rack-mount and blade servers based on Intel Xeon Processor scalable family. The Cisco UCS servers offer the patented Cisco Extended Memory Technology to support applications with large datasets and allow more virtual machines per server.
- Network - The system is integrated onto a low-latency, lossless, 40-Gbps unified network fabric. This network foundation consolidates LANs, SANs, and high-performance computing networks which are separate networks today. The unified fabric lowers costs by reducing the number of network adapters, switches, and cables, and by decreasing the power and cooling requirements.
- Virtualization - The system unleashes the full potential of virtualization by enhancing the scalability, performance, and operational control of virtual environments. Cisco security, policy enforcement, and diagnostic features are now extended into virtualized environments to better support changing business and IT requirements.
- Storage access - The system provides consolidated access to both SAN storage and Network Attached Storage (NAS) over the unified fabric. By unifying the storage access, Cisco Unified Computing System can access storage over Ethernet (NFS or iSCSI), Fibre Channel, and Fibre Channel over Ethernet (FCoE). This provides customers with choice for storage access and investment protection. In addition, the server administrators can pre-assign storage-access policies for system connectivity to storage resources, simplifying storage connectivity, and management for increased productivity.

Cisco Unified Computing System is designed to deliver:

- A reduced Total Cost of Ownership (TCO) and increased business agility.
- Increased IT staff productivity through just-in-time provisioning and mobility support.
- A cohesive, integrated system, which unifies the technology in the data center.
- Industry standards supported by a partner ecosystem of industry leaders.

Cisco UCS Manager

Cisco UCS Manager (UCSM) provides a unified, embedded management of all software and hardware components of the Cisco Unified Computing System across multiple chassis, rack servers, and thousands of virtual machines. It supports all Cisco UCS product models, including Cisco UCS B-Series Blade Servers, Cisco UCS C-Series Rack-Mount Servers, and Cisco UCS Mini, as well as the associated storage resources and networks. Cisco UCS Manager is embedded on a pair of Cisco UCS 6300 or 6200 Series Fabric Interconnects using a clustered, active-standby configuration for high availability. The manager participates in server provisioning, device discovery, inventory, configuration, diagnostics, monitoring, fault detection, auditing, and statistics collection.

Figure 1 Cisco UCS Manager

An instance of Cisco UCS Manager with all Cisco UCS components managed by it forms a Cisco UCS domain, which can include up to 160 servers. In addition to provisioning Cisco UCS resources, this infrastructure management software provides a model-based foundation for streamlining the day-to-day processes of updating, monitoring, and managing computing resources, local storage, storage connections, and network connections. By enabling better automation of processes, Cisco UCS Manager allows IT organizations to achieve greater agility and scale in their infrastructure operations while reducing complexity and risk. The manager provides flexible role and policy-based management using service profiles and templates.

Cisco UCS Manager manages Cisco UCS systems through an intuitive HTML 5 or Java user interface and a CLI. It can register with Cisco UCS Central Software in a multi-domain Cisco UCS environment, enabling centralized management of distributed systems scaling to thousands of servers. Cisco UCS Manager can be integrated with Cisco UCS Director to facilitate orchestration and to provide support for converged infrastructure and Infrastructure as a Service (IaaS).

The Cisco UCS XML API provides comprehensive access to all Cisco UCS Manager functions. The API provides Cisco UCS system visibility to higher-level systems management tools from independent software vendors (ISVs) such as VMware, Microsoft, and Splunk as well as tools from BMC, CA, HP, IBM, and others. ISVs and in-house developers can use the XML API to enhance the value of the Cisco UCS platform according to their unique requirements. Cisco UCS PowerTool for Cisco UCS Manager and the Python Software Development Kit (SDK) help automate and manage configurations within Cisco UCS Manager.

Cisco UCS 6300 Fabric Interconnects

The Cisco UCS 6300 Series Fabric Interconnects are a core part of Cisco UCS, providing both network connectivity and management capabilities for the system. The Cisco UCS 6300 Series offers line-rate, low-latency, lossless 10 and 40 Gigabit Ethernet, Fibre Channel over Ethernet (FCoE), and Fibre Channel functions.

Figure 2 Cisco UCS 6300 Fabric Interconnect

The Cisco UCS 6300 Series provides the management and communication backbone for the Cisco UCS B-Series Blade Servers, 5100 Series Blade Server Chassis, and C-Series Rack Servers managed by Cisco UCS. All servers attached to the fabric interconnects become part of a single, highly available management domain. In

addition, by supporting unified fabric, the Cisco UCS 6300 Series provides both LAN and SAN connectivity for all servers within its domain.

From a networking perspective, the Cisco UCS 6300 Series uses a cut-through architecture, supporting deterministic, low-latency, line-rate 10 and 40 Gigabit Ethernet ports, switching capacity of 2.56 terabits per second (Tbps), and 320 Gbps of bandwidth per chassis, independent of packet size and enabled services. The product family supports Cisco® low-latency, lossless 10 and 40 Gigabit Ethernet unified network fabric capabilities, which increase the reliability, efficiency, and scalability of Ethernet networks. The fabric interconnect supports multiple traffic classes over a lossless Ethernet fabric from the server through the fabric interconnect. Significant TCO savings can be achieved with an FCoE optimized server design in which network interface cards (NICs), host bus adapters (HBAs), cables, and switches can be consolidated.

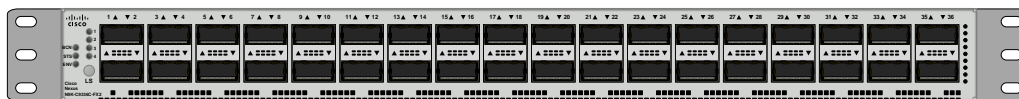
The Cisco UCS 6332 32-Port Fabric Interconnect is a 1-rack-unit (1RU) Gigabit Ethernet, and FCoE switch offering up to 2.56 Tbps throughput and up to 32 ports. The switch has 32 fixed 40-Gbps Ethernet and FCoE ports.

Both the Cisco UCS 6332UP 32-Port Fabric Interconnect and the Cisco UCS 6332 16-UP 40-Port Fabric Interconnect have ports that can be configured for the breakout feature that supports connectivity between 40 Gigabit Ethernet ports and 10 Gigabit Ethernet ports. This feature provides backward compatibility to existing hardware that supports 10 Gigabit Ethernet. A 40 Gigabit Ethernet port can be used as four 10 Gigabit Ethernet ports. Using a 40 Gigabit Ethernet SFP, these ports on a Cisco UCS 6300 Series Fabric Interconnect can connect to another fabric interconnect that has four 10 Gigabit Ethernet SFPs. The breakout feature can be configured on ports 1 to 12 and ports 15 to 26 on the Cisco UCS 6332UP fabric interconnect. Ports 17 to 34 on the Cisco UCS 6332 16-UP fabric interconnect support the breakout feature.

Cisco UCS C9336C-FX2 Nexus Switches

The Cisco Nexus 9000 Series Switches include both modular and fixed-port switches that are designed to overcome these challenges with a flexible, agile, low-cost, application-centric infrastructure.

Figure 3 Cisco Nexus C9336C-FX2 Switch



The Cisco Nexus 9300 platform consists of fixed-port switches designed for top-of-rack (ToR) and middle-of-row (MoR) deployment in data centers that support enterprise applications, service provider hosting, and cloud computing environments. They are Layer 2 and 3 nonblocking 10 and 40 Gigabit Ethernet switches with up to 2.56 terabits per second (Tbps) of internal bandwidth.

The Cisco Nexus C9336C-FX2 Switch is a 1-rack-unit (1RU) switch that supports 7.2 Tbps of bandwidth and over 2.8 billion packets per second (bps) across thirty-six 10/25/40/100 -Gbps Enhanced QSFP28 ports

All the Cisco Nexus 9300 platform switches use dual-core 2.5-GHz x86 CPUs with 64-GB solid-state disk (SSD) drives and 16 GB of memory for enhanced network performance.

With the Cisco Nexus 9000 Series, organizations can quickly and easily upgrade existing data centers to carry 40 Gigabit Ethernet to the aggregation layer or to the spine (in a leaf-and-spine configuration) through advanced and cost-effective optics that enable the use of existing 10 Gigabit Ethernet fiber (a pair of multimode fiber strands).

Cisco provides two modes of operation for the Cisco Nexus 9000 Series. Organizations can use Cisco NX-OS Software to deploy the Cisco Nexus 9000 Series in standard Cisco Nexus switch environments. Organizations also can use a hardware infrastructure that is ready to support Cisco Application Centric Infrastructure (Cisco ACI) to take full advantage of an automated, policy-based, systems management approach.

Cisco UCS S3260 M5 Storage Server

The Cisco UCS S3260 Storage Server is a modular, high-density, high availability, dual-node rack server, well suited for service providers, enterprises, and industry-specific environments. It addresses the need for dense cost effective storage for the ever-growing data needs. Designed for a new class of cloud-scale applications, it is simple to deploy and excellent for big data applications, software-defined storage environments, and other unstructured data repositories, media streaming, and content distribution.

Figure 4 Cisco UCS S3260 Storage Server



Extending the capability of the Cisco UCS C3000 portfolio, the Cisco UCS S3260 helps you achieve the highest levels of data availability. With dual-node capability that is based on the Intel Xeon scalable processors, it features up to 840 TB of local storage in a compact 4-rack-unit (4RU) form factor. All hard-disk drives can be asymmetrically split between the dual-nodes and are individually hot-swappable. The drives can be built-in in an enterprise-class Redundant Array of Independent Disks (RAID) redundancy or be in a pass-through mode.

This high-density rack server comfortably fits in a standard 32-inch depth rack, such as the Cisco R42610 Rack-Server.

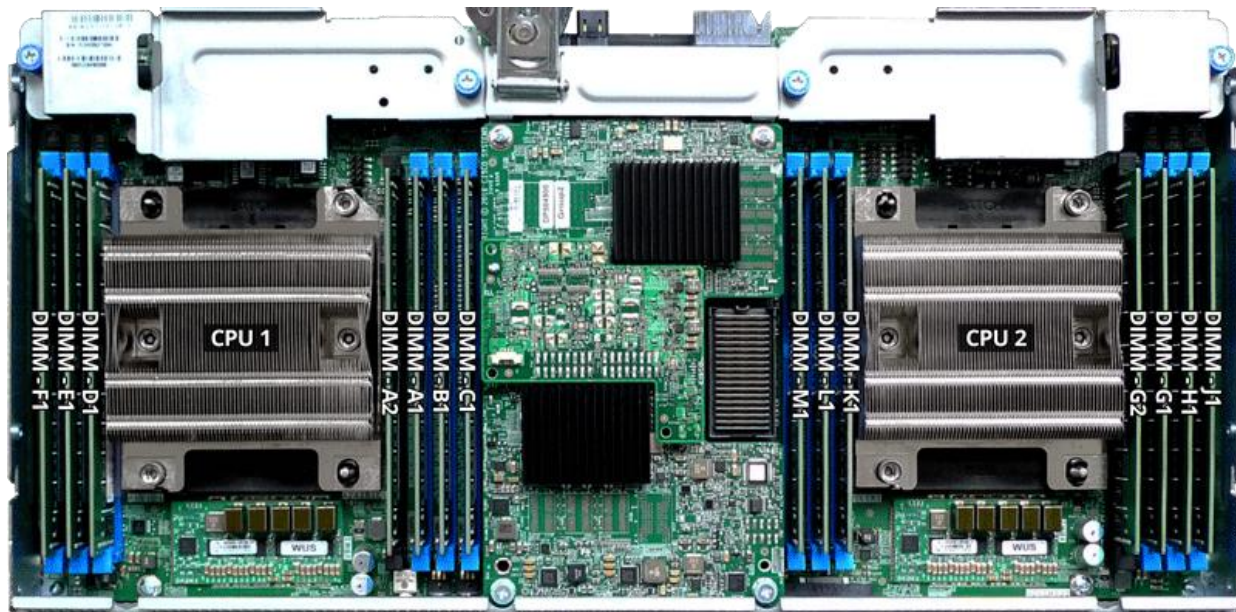
The Cisco UCS S3260 is deployed as a standalone server in both bare-metal or virtualized environments. Its modular architecture reduces TCO by allowing you to upgrade individual components over time and as use cases evolve, without having to replace the entire system.

The Cisco UCS S3260 uses a modular server architecture that, using Cisco's blade technology expertise, allows you to upgrade the computing or network nodes in the system without the need to migrate data migration from one system to another. It delivers the following:

- Dual server nodes
- Up to 48 computing cores per server node
- Up to 60 drives mixing a large form factor (LFF) with up to 28 solid-state disk (SSD) drives plus 2 SSD SATA boot drives per server node
- Up to 1.5 TB of memory per server node (3 TB Total) with 128GB DIMMs
- Support for 12-Gbps serial-attached SCSI (SAS) drives
- A system I/O Controller either with HBA Passthrough or RAID controller, with DUAL LSI 3316 Chip

- Cisco VIC 1300 Series Embedded Chip supporting Dual-port 40Gbps
- High reliability, availability, and serviceability (RAS) features with tool-free server nodes, system I/O controller, easy-to-use latching lid, and hot-swappable and hot-pluggable components
- Dual 7mm NVMe - Up to 4 TB
- 1G Host Management Port

Figure 5 Cisco UCS S3260 M5 Internals



Cisco UCS C220 M5 Rack-Mount Server

The Cisco UCS C220 M5 Rack-Mount Server is among the most versatile general-purpose enterprise infrastructure and application servers in the industry. It is a high-density 2-socket rack server that delivers industry-leading performance and efficiency for a wide range of workloads, including virtualization, collaboration, and bare-metal applications. The Cisco UCS C-Series Rack-Mount Servers can be deployed as standalone servers or as part of Cisco UCS to take advantage of Cisco's standards-based unified computing innovations that help reduce customers' TCO and increase their business agility.

The Cisco UCS C220 M5 server extends the capabilities of the Cisco UCS portfolio in a 1-Rack-Unit (1RU) form factor. It incorporates the Intel® Xeon Scalable processors, supporting up to 20 percent more cores per socket, twice the memory capacity, 20 percent greater storage density, and five times more PCIe NVMe Solid-State Disks (SSDs) compared to the previous generation of servers. These improvements deliver significant performance and efficiency gains that will improve your application performance.

Figure 6 Cisco UCS C220M5 Rack-Mount Server



The Cisco UCS C220 M5 SFF server extends the capabilities of the Cisco Unified Computing System portfolio in a 1U form factor with the addition of the Intel Xeon Processor Scalable Family, 24 DIMM slots for 2666MHz DIMMs

and capacity points up to 128GB, two 2 PCI Express (PCIe) 3.0 slots, and up to 10 SAS/SATA hard disk drives (HDDs) or solid state drives (SSDs). The Cisco UCS C220 M5 SFF server also includes one dedicated internal slot for a 12G SAS storage controller card.

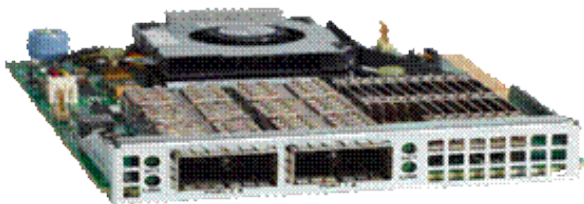
The Cisco UCS C220 M5 server included one dedicated internal modular LAN on motherboard (mLOM) slot for installation of a Cisco Virtual Interface Card (VIC) or third-party network interface card (NIC), without consuming a PCI slot, in addition to 2 x 10Gbase-T Intel x550 embedded (on the motherboard) LOM ports.

The Cisco UCS C220 M5 server can be used standalone, or as part of Cisco Unified Computing System, which unifies computing, networking, management, virtualization, and storage access into a single integrated architecture enabling end-to-end server visibility, management, and control in both bare metal and virtualized environments.

Cisco UCS Virtual Interface Card 1387

The Cisco UCS Virtual Interface Card (VIC) 1387 is a Cisco innovation. It provides a policy-based, stateless, agile server infrastructure for your data center. This dual-port Enhanced Quad Small Form-Factor Pluggable (QSFP) half-height PCI Express (PCIe) modular LAN-on-motherboard (mLOM) adapter is designed exclusively for Cisco UCS C-Series and 3260 Rack Servers. The card supports 40 Gigabit Ethernet and Fibre Channel over Ethernet (FCoE). It incorporates Cisco's next-generation converged network adapter (CNA) technology and offers a comprehensive feature set, providing investment protection for future feature software releases. The card can present more than 256 PCIe standards-compliant interfaces to the host and these can be dynamically configured as either network interface cards (NICs) or host bus adapters (HBAs). In addition, the VIC supports Cisco Data Center Virtual Machine Fabric Extender (VM-FEX) technology. This technology extends the Cisco UCS Fabric Interconnect ports to virtual machines, simplifying server virtualization deployment.

Figure 7 Cisco UCS VIC 1387



The Cisco UCS VIC 1387 provides the following features and benefits:

- Stateless and agile platform: The personality of the card is determined dynamically at boot time using the service profile associated with the server. The number, type (NIC or HBA), identity (MAC address and World Wide Name [WWN]), failover policy, bandwidth, and quality-of-service (QoS) policies of the PCIe interfaces are all determined using the service profile. The capability to define, create, and use interfaces on demand provides a stateless and agile server infrastructure.
- Network interface virtualization: Each PCIe interface created on the VIC is associated with an interface on the Cisco UCS fabric interconnect, providing complete network separation for each virtual cable between a PCIe device on the VIC and the interface on the fabric interconnect.

Red Hat Enterprise Linux 7.5

Red Hat® Enterprise Linux is a high-performing operating system that has delivered outstanding value to IT environments for more than a decade. More than 90 percent of Fortune Global 500 companies use Red Hat products and solutions including Red Hat Enterprise Linux. As the world's most trusted IT platform, Red Hat Enterprise Linux has been deployed in mission-critical applications at global stock exchanges, financial institutions,

leading telcos, and animation studios. It also powers the websites of some of the most recognizable global retail brands.

Red Hat Enterprise Linux:

- Delivers high-performance, reliability, and security
- Is certified by the leading hardware and software vendors
- Scales from workstations, to servers, to mainframes
- Provides a consistent application environment across physical, virtual, and cloud deployments

Designed to help organizations make a seamless transition to emerging datacenter models that include virtualization and cloud computing, Red Hat Enterprise Linux includes support for major hardware architectures, hypervisors, and cloud providers, making deployments across physical and different virtual environments predictable and secure. Enhanced tools and new capabilities in this release enable administrators to tailor the application environment to efficiently monitor and manage compute resources and security.

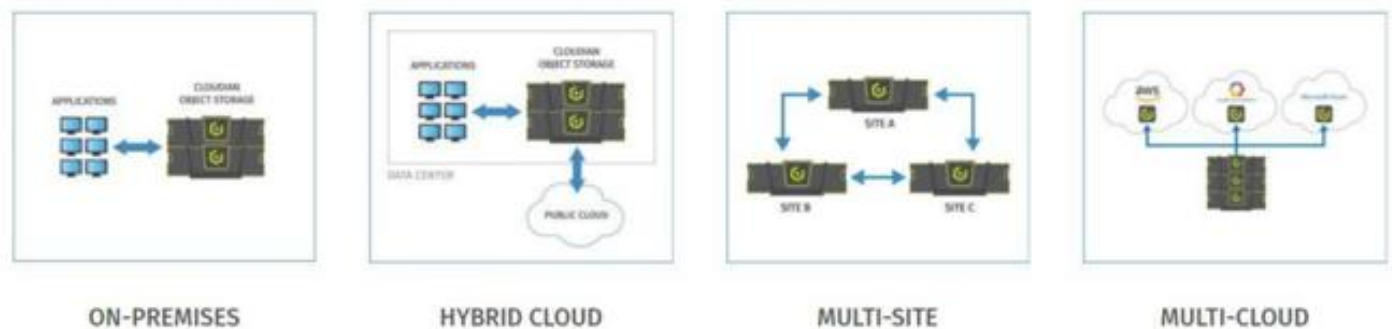
Clouidian HyperStore

Clouidian HyperStore enables data centers to provide highly cost-effective on-premise unstructured data storage repositories. Clouidian HyperStore is built on standard hardware that spans across the enterprise as well as into public cloud environments. Clouidian HyperStore is available as a stand-alone software. It easily scales to limitless capacities and offers multi-data center storage. HyperStore also has fully automated data tiering to all major public clouds, including AWS, Azure and Google Cloud Platform. It fully supports S3 applications and has flexible security options.

Clouidian HyperStore is a scale-out object storage system designed to manage massive amounts of data. It is an SDS solution that runs on the Cisco UCS platform allowing cost savings for datacenter storage while providing extreme availability and reliability.

HyperStore deployment models include on-premises storage, distributed storage, storage-as-a-service, or even other combinations (Figure 8).

Figure 8 HyperStore Deployment Models



Clouidian Object Storage

Get everything you love about cloud storage, right in your data center. Cisco and Clouidian deliver an object storage solution that provides petabyte-scalability while keeping it simple to manage. Deploy as on-premises storage or configure a hybrid cloud and automatically tier data to the public cloud. All at 70 percent less TCO than conventional storage.

View system health, manage users and groups and automate tasks with Clouidian’s web-based UI and REST API. Manage your workload with a self-service portal that lets users administer their own storage. Powerful QoS capabilities help you ensure SLAs.

Clouidian makes it easy to get started. Begin with the cluster size that fits your needs and expand on demand. In Clouidian’s modular, shared-nothing architecture, every node is identical, allowing the solution to grow from a few nodes to a few hundred without disruption. Performance scales linearly, too.

Only Clouidian HyperStore offers a 100 percent native S3 API, proven to deliver the highest interoperability in its class. Guaranteed compatible with your S3-enabled applications, Clouidian gives you investment protection and peace of mind.

Get the benefits of both on-premises and cloud storage in a single management environment. Run your S3-enabled applications in your data center with Clouidian S3 scale-out storage. Use policies you define to automatically tier data to the public cloud. It’s simple to manage and limitlessly scalable.

Get all the benefits of using the Cisco UCS platform while managing your data through a single pane of glass.

Cloudian HyperStore Design

Cloudian HyperStore is an Amazon S3-compliant multi-tenant object storage system. The system utilizes a “non-SQL” (NoSQL) storage layer for maximum flexibility and scalability. The Cloudian HyperStore system enables any service provider or enterprise to deploy an S3-compliant multi-tenant storage cloud.

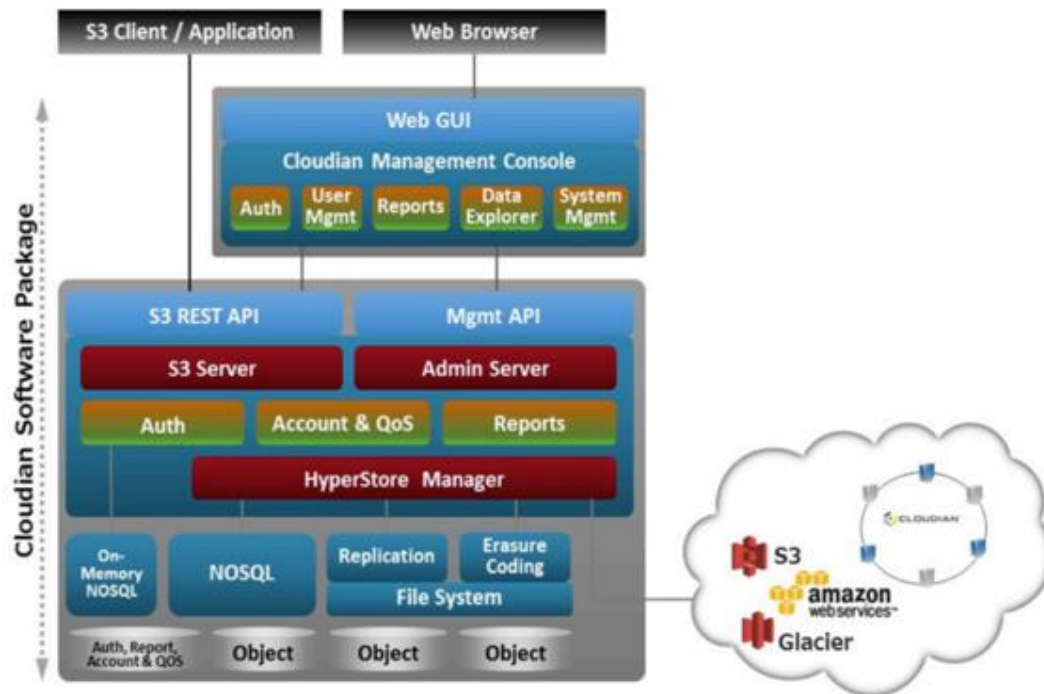
The Cloudian HyperStore system is designed specifically to meet the demands of high volume, multi-tenant data storage:

- Amazon S3 API compliance. The Cloudian HyperStore system is 100% compatible with Amazon S3’s HTTP REST API. Customer’s existing HTTP S3 applications will work with the Cloudian HyperStore service, and existing S3 development tools and libraries can be used for building Cloudian HyperStore client applications.
- Secure multi-tenancy. The Cloudian HyperStore system provides the capability to have multiple users securely reside on a single, shared infrastructure. Data for each user is logically separated from other users’ data and cannot be accessed by any other user unless access permission is explicitly granted.
- Group support. An enterprise or work group can share a single Cloudian HyperStore account. Each group member can have dedicated storage space, and the group can be managed by a designated group administrator.
- Quality of Service (QoS) controls. Cloudian HyperStore system administrators can set storage quotas and usage rate limits on a per-group and per-user basis. Group administrators can set quotas and rate controls for individual members of the group.
- Access control rights. Read and write access controls are supported at per-bucket and per-object granularity. Objects can also be exposed via public URLs for regular web access, subject to configurable expiration periods.
- Reporting and billing. The Cloudian HyperStore system supports usage reporting on a system-wide, group-wide, or individual user basis. Billing of groups or users can be based on storage quotas and usage rates (such as bytes in and bytes out).
- Horizontal scalability. Running on standard off-the-shelf hardware, a Cloudian HyperStore system can scale up to thousands of nodes across multiple datacenters, supporting millions of users and hundreds of petabytes of data. New nodes can be added without service interruption.
- High availability. The Cloudian HyperStore system has a fully distributed, peer-to-peer architecture, with no single point of failure. The system is resilient to network and node failures with no data loss due to the automatic replication and recovery processes inherent to the architecture. A Cloudian HyperStore geocluster can be deployed across multiple datacenters to provide redundancy and resilience in the event of a data center scale disaster.

Cloudian HyperStore Architecture

The Cloudian HyperStore is a fully distributed architecture that provides no single point of failure, data protection options (replication or erasure coding), data recovery upon a node failure, dynamic re-balancing on node addition, multi-data center and multi-region support. Figure 9 illustrates all of the service components that comprise a Cloudian HyperStore system.

Figure 9 Cloudian HyperStore Architecture



Cloudian Management Console

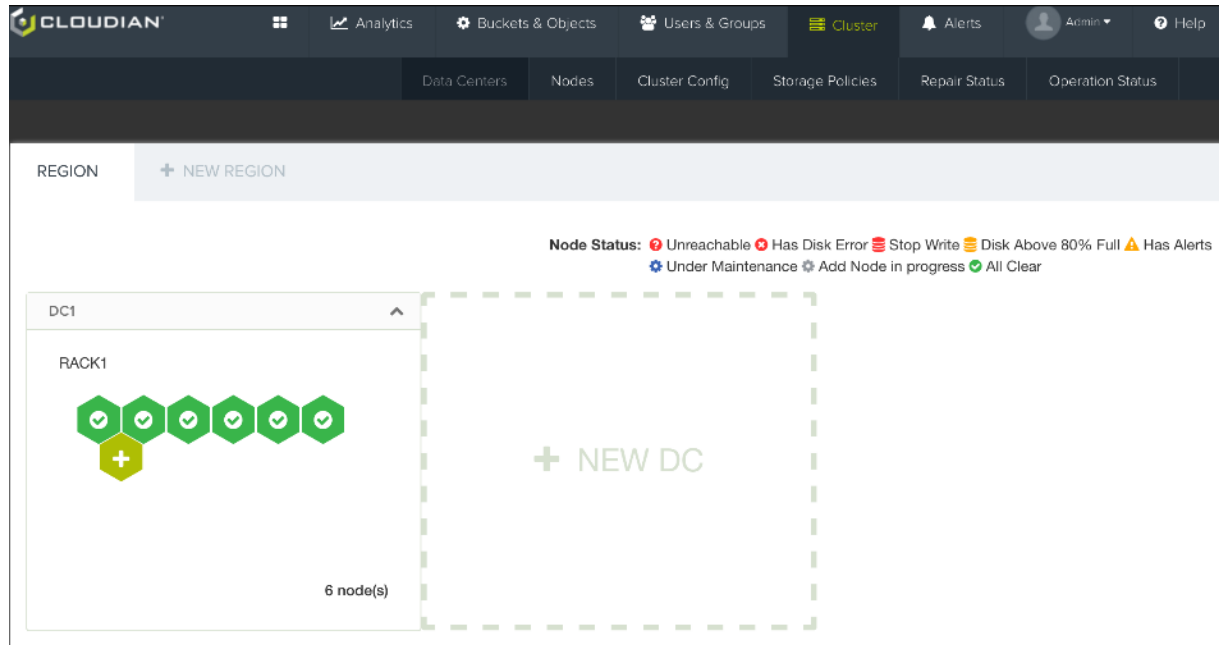
The Cloudian Management Console (CMC) is a web-based user interface for Cloudian HyperStore system administrators, group administrators, and end users. The functionality available through the CMC depends on the user type associated with a user’s login ID (system administrative, group administrative, or regular user).

As a Cloudian HyperStore system administrator, you can use the CMC to perform the following tasks:

- Provisioning groups and users
- Managing quality of service (QoS) controls
- Creating and managing rating plans
- Generating usage data reports
- Generating bills
- Viewing and managing users’ stored data objects
- Setting access control rights on users’ buckets and stored objects

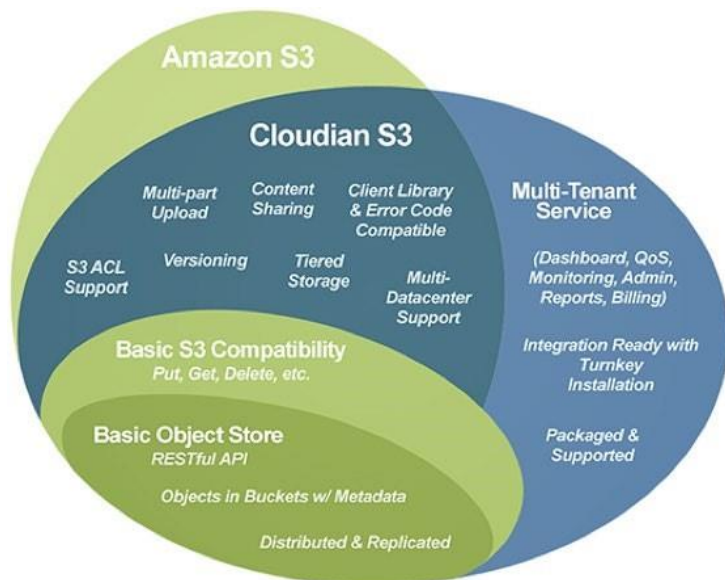
Group administrators can perform a limited range of administrative tasks pertaining to their own group. Regular users can perform S3 operations such as uploading and downloading S3 objects. The CMC acts as a client to the Administrative Service and the S3 Service.

Figure 10 Cloudian Management Console



S3 Compatible

With Amazon setting the cloud storage standard making it the largest object storage environment, and Amazon S3 API becoming the de facto standard for developers writing storage applications for cloud, it is imperative every Cloud, hybrid storage solution is S3 compliant. Cloudian HyperStore, in addition to being S3 compliant, also offers the flexibility to be on-premises object storage as well as hybrid tier to Amazon and Google clouds.



Integrated Billing, Management, and Monitoring

The HyperStore system maintains comprehensive service usage data for each group and each user in the system. This usage data, which is protected by replication, serves as the foundation for HyperStore service billing

functionality. The system allows the creation of rating plans that categorize the types of service usage for single users or groups for a selected service period. The CMC has a function to display a single user’s bill report in a browser; HyperStore Admin API can be used to generate user or group billing data that can be ingested a third-party billing application. Cloudian HyperStore also allows for the special treatment of designated source IP addresses, so that the billing mechanism does not apply any data transfer charges for data coming from or going to these “whitelisted” domains.

RATING PLANS + ADD RATING PLAN		
ID	NAME	ACTIONS
Default-RP	Default Rating Plan	Edit
Whitelist-RP	Whitelist Rating Plan	Edit

Cloudian Features

The following are the key Cloudian features:

- Auto-Tiering
- Programmable
- Multi-Tenancy
- IAM User Support
- Broad Application Support
- Cloudian HyperStore Secret Sauce
- Distributed Peer-To-Peer Architecture
- Parallel Disk IO Data Protection
- Configurable Data Consistency
- Storage Node Heterogeneity
- Compression Your Way
- Quality of Service

Solution Design

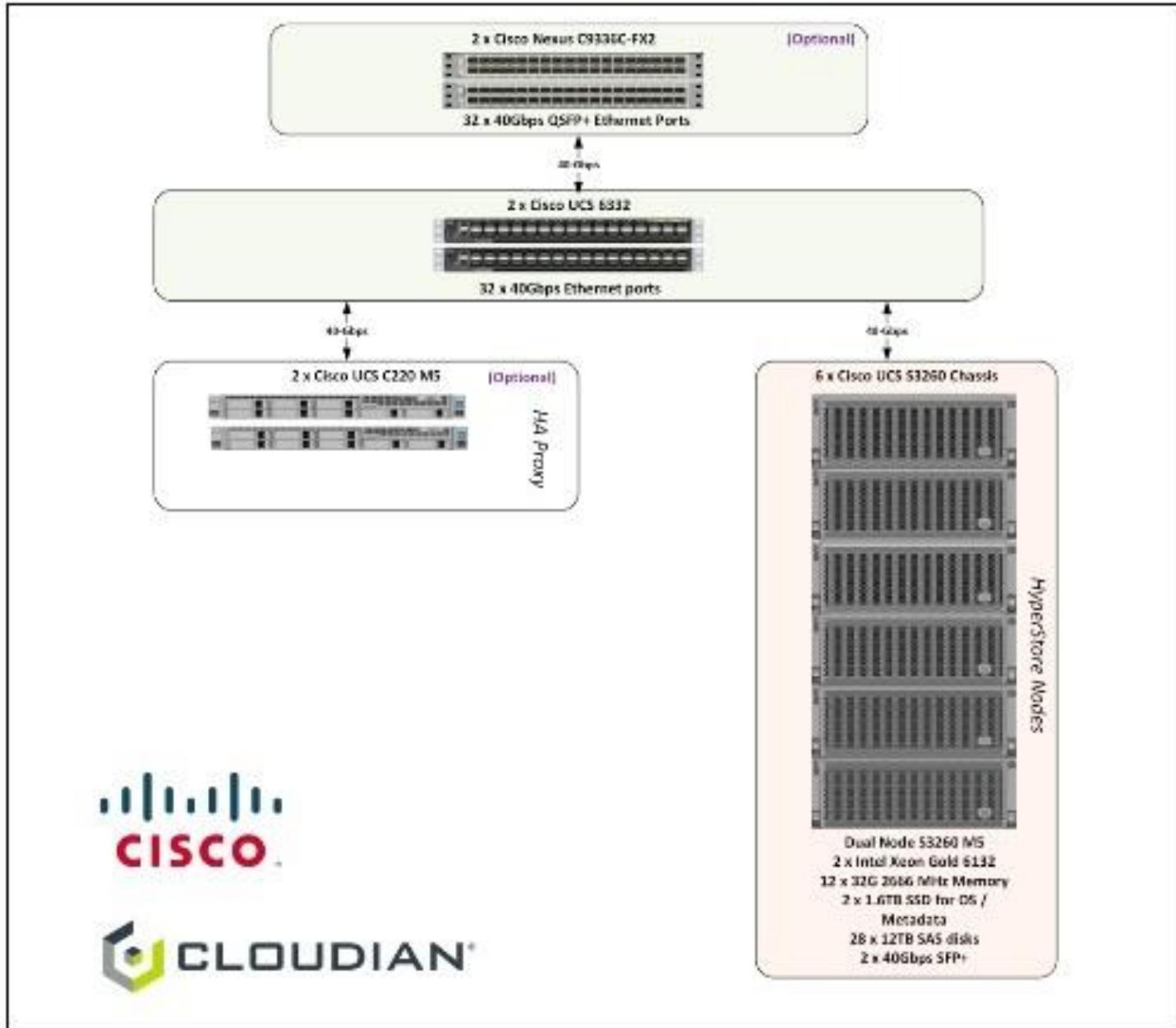
Deployment Architecture

The reference architecture use case provides a comprehensive, end-to-end example of designing and deploying Cloudian object storage on Cisco UCS S3260 as shown in Figure 11. This document describes the architecture and design of a Cloudian Scale-out object storage and file system solution on six Cisco UCS S3260 Storage Server Chassis; each with two Cisco UCS S3260 M5 nodes configured as storage servers and two optional Cisco UCS C220 M5S rack server for high availability proxy. The whole solution is connected to a pair of Cisco UCS 6332 Fabric Interconnects and a pair of upstream network Cisco Nexus 9336C-FX2 switches.

The configuration is comprised of the following:

- 2 x Cisco Nexus 9336C-FX2 Switches
- 2 x Cisco UCS 6332 Fabric Interconnects
- 6 x Cisco UCS S3260 Storage Chassis with 2 x Cisco UCS S3260 M5 server nodes each
- 2 x Cisco UCS C220 M5S Rack Servers (Optional. For HA-Proxy)

Figure 11 Cisco UCS Hardware for Cloudbian HyperStore



System Hardware and Software Specifications

Solution Overview

This solution is based on Cisco UCS and Cloudian Object and file storage.

Software Versions

Table 1 Software Versions

Layer	Component	Version or Release
Storage (Chassis) UCS S3260	Chassis Management Controller	4.0(2b)
	Shared Adapter	4.0(2b)
Compute (Server Nodes) UCS S3X60 M5	BIOS	S3260M5.4.0.1b
	CIMC Controller	4.0(1a)
Compute (Rack Server) C220 M5S	BIOS	C220M5.4.0.1c
	CIMC Controller	4.0(1a)
Network 6332 Fabric Interconnect	UCS Manager	4.0(2b)
	Kernel	5.0(3)N2(4.02b)
	System	5.0(3)N2(4.02b)
Network Nexus 9336C-FX2	BIOS	07.51
	NXOS	9.2(1)
Software	Red Hat Enterprise Linux Server	7.5 (x86_64)
	Cloudian HyperStore	7.1.2

Hardware Requirements and Bill of Materials

Table 2 lists the requirements and materials used in this CVD.

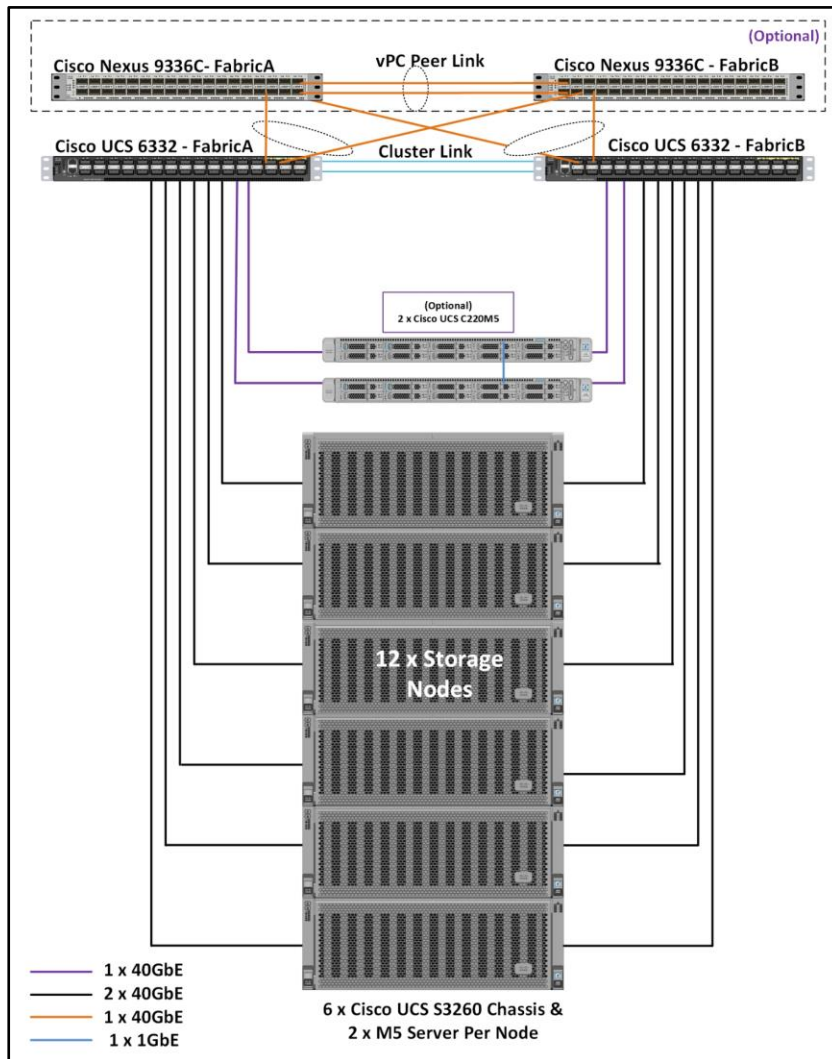
Table 2 Bill of Materials

Component	Model	Quantity	Comments
Cloudian Storage Nodes	Cisco UCS S3260 M5 Chassis	6	<ul style="list-style-type: none"> • 2 x UCS S3260 M5 Server Nodes per Chassis (Total = 12 nodes) • Per Server Node <ul style="list-style-type: none"> – 2 x Intel Xeon Gold 6132(2.6 GHz/14cores), 384 GB RAM – Cisco 12G RAID Controller – 2 x 1.6TB boot SSD for OS installation and metadata – 28 x 12TB HDDs for Data – Dual-port 40 Gbps VIC
Cloudian HA-Proxy Node (Optional)	Cisco UCS C220 M5S Rack server	2	<ul style="list-style-type: none"> • 2 x Intel Xeon Silver 4110 (2.1GHz/8 Cores) 96GB RAM • Cisco 12G SAS RAID Controller • 2 x 600GB SAS for OS installation • Dual-port 40 Gbps VIC
UCS Fabric Interconnects	Cisco UCS 6332 Fabric Interconnects	2	
Switches	Cisco Nexus C9336C-FX2 Switches	2	

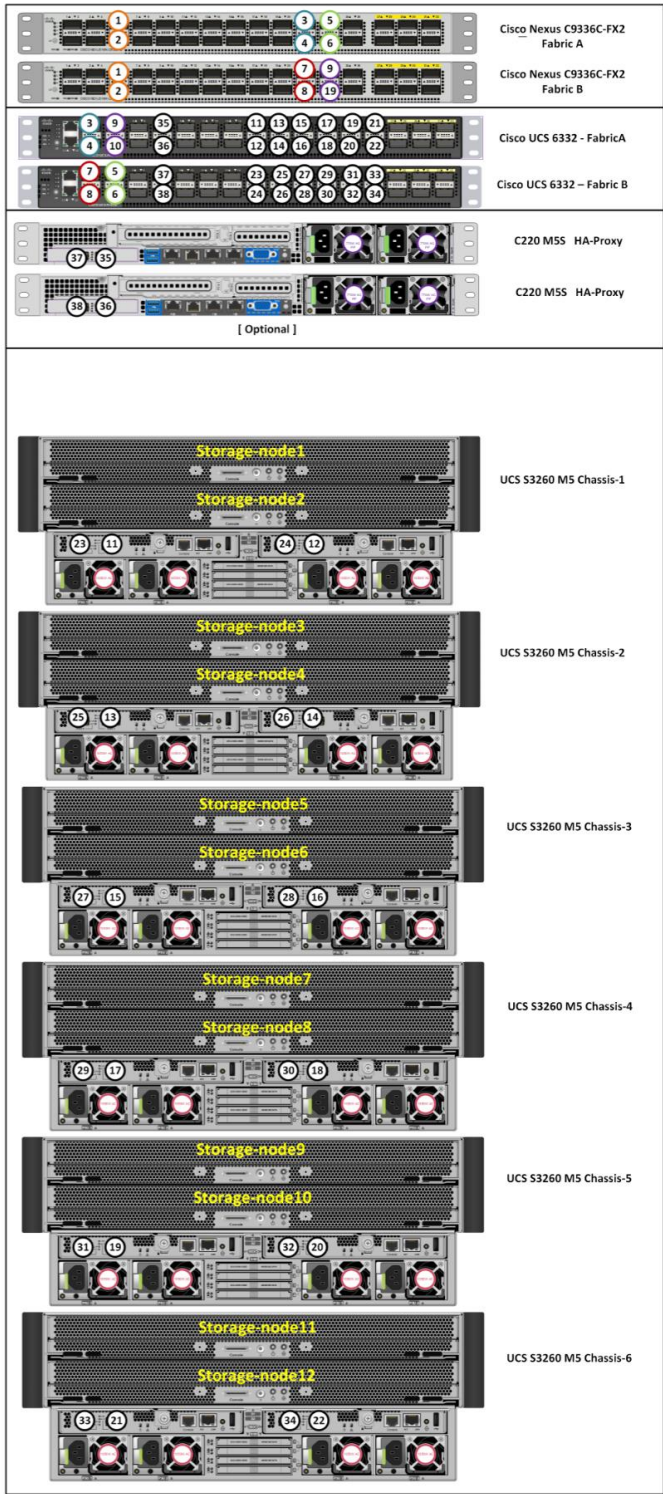
Physical Topology and Configuration

The following sections describe the physical design of the solution and the configuration of each component.

Figure 12 Physical Topography



The connectivity of the solution is based on 40 Gbit. All components are connected via 40 QSFP cables. Between both Cisco Nexus 9336-FX2 switches are 2 x 40 Gbit cabling. Each Cisco UCS 6332 Fabric Interconnect is connected with 2 x 40 Gbit to each Cisco UCS C9336C-FX2 switch, and each Cisco UCS C220 M5 is connected with 1 x 40 Gbit and each Cisco UCS S3260 M5 server is connected with 2 x 40 Gbit cable to each Fabric Interconnect. The architecture is highly redundant and system survived with little or no impact to applications under various failure test scenarios which will be covered during validation and testing.



The exact cabling for the Cisco UCS S3260 Storage Server, Cisco UCS C220 M5, and the Cisco UCS 6332 Fabric Interconnect is illustrated in Table 3 .

Table 3 Cabling Information

Local Device	Local Port	Connection	Remote Device	Remote Port	Cable
Cisco Nexus C9336C-FX2 Switch- A	Eth1/5	40GbE	Cisco Nexus C9336C-FX2 Switch- B	Eth1/5	QSFP-H40G-CU1M
	Eth1/6	40GbE	Cisco Nexus C9336C-FX2 Switch- B	Eth1/6	QSFP-H40G-CU1M
	Eth1/21	40GbE	Cisco UCS Fabric Interconnect A	Eth1/1	QSFP-H40G-CU1M
	Eth1/22	40GbE	Cisco UCS Fabric Interconnect A	Eth1/2	QSFP-H40G-CU1M
	Eth1/23	40GbE	Cisco UCS Fabric Interconnect B	Eth1/3	QSFP-H40G-CU1M
	Eth1/24	40GbE	Cisco UCS Fabric Interconnect B	Eth1/4	QSFP-H40G-CU1M
	Eth1/36	40GbE	Top of Rack (Upstream Network)	Any	QSFP+ 4SFP10G
	MGMT0	1GbE	Top of Rack (Management)	Any	1G RJ45
Cisco Nexus C9336C-FX2 Switch- B	Eth1/5	40GbE	Cisco Nexus C9336C-FX2 Switch- A	Eth1/5	QSFP-H40G-CU1M
	Eth1/6	40GbE	Cisco Nexus C9336C-FX2 Switch- A	Eth1/6	QSFP-H40G-CU1M
	Eth1/21	40GbE	Cisco UCS Fabric Interconnect B	Eth1/1	QSFP-H40G-CU1M
	Eth1/22	40GbE	Cisco UCS Fabric Interconnect B	Eth1/2	QSFP-H40G-CU1M
	Eth1/23	40GbE	Cisco UCS Fabric Interconnect A	Eth1/3	QSFP-H40G-CU1M
	Eth1/24	40GbE	Cisco UCS Fabric Interconnect A	Eth1/4	QSFP-H40G-CU1M
	Eth1/36	40GbE	Top of Rack (Upstream Network)	Any	QSFP+ 4SFP10G
	MGMT0	1GbE	Top of Rack (Management)	Any	1G RJ45
Cisco UCS 6332 Fabric Interconnect	Eth1/1	40GbE	Nexus 9332 A	Eth 1/21	QSFP-H40G-CU1M

Local Device	Local Port	Connection	Remote Device	Remote Port	Cable
A	Eth1/2	40GbE	Nexus 9332 A	Eth 1/22	QSFP-H40G-CU1M
	Eth1/3	40GbE	Nexus 9332 B	Eth 1/23	QSFP-H40G-CU1M
	Eth1/4	40GbE	Nexus 9332 B	Eth 1/24	QSFP-H40G-CU1M
	Eth1/7	40GbE	C220M5	Port1	QSFP-H40G-CU3M
	Eth1/8	40GbE	C220M5	Port2	QSFP-H40G-CU3M
	Eth1/15	40GbE	S3260 Chassis 1 - SIOC 1 (right)	Port 1	QSFP-H40G-CU3M
	Eth1/16	40GbE	S3260 Chassis 1 - SIOC 2 (left)	Port 1	QSFP-H40G-CU3M
	Eth1/17	40GbE	S3260 Chassis 2 - SIOC 1 (right)	Port 1	QSFP-H40G-CU3M
	Eth1/18	40GbE	S3260 Chassis 2 - SIOC 2 (left)	Port 1	QSFP-H40G-CU3M
	Eth1/19	40GbE	S3260 Chassis 3 - SIOC 1 (right)	Port 1	QSFP-H40G-CU3M
	Eth1/20	40GbE	S3260 Chassis 3 - SIOC 2 (left)	Port 1	QSFP-H40G-CU3M
	Eth1/21	40GbE	S3260 Chassis 4 - SIOC 1 (right)	Port 1	QSFP-H40G-CU3M
	Eth1/22	40GbE	S3260 Chassis 4 - SIOC 2 (left)	Port 1	QSFP-H40G-CU3M
	Eth1/23	40GbE	S3260 Chassis 5 - SIOC 1 (right)	Port 1	QSFP-H40G-CU3M
	Eth1/24	40GbE	S3260 Chassis 5 - SIOC 2 (left)	Port 1	QSFP-H40G-CU3M
	Eth1/25	40GbE	S3260 Chassis 6 - SIOC 1 (right)	Port 1	QSFP-H40G-CU3M
	Eth1/26	40GbE	S3260 Chassis 6 - SIOC 2 (left)	Port 1	QSFP-H40G-CU3M
	MGMT0	40GbE	Top of Rack (Management)	Any	1G RJ45
	L1	1GbE	UCS 6332 Fabric Interconnect B	L1	1G RJ45

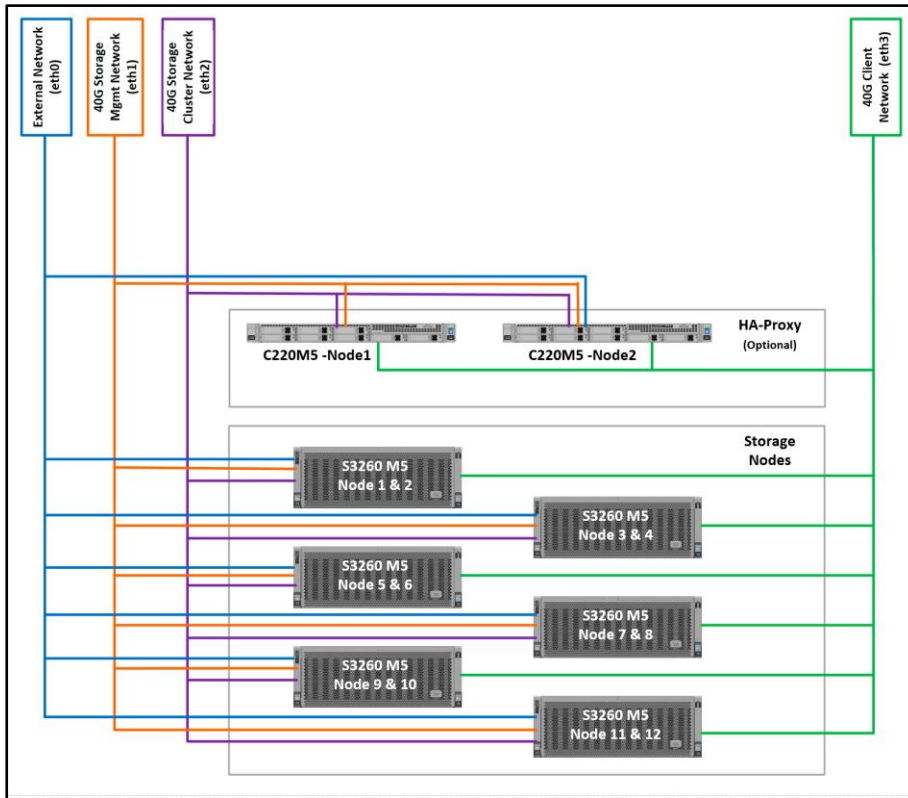
Local Device	Local Port	Connection	Remote Device	Remote Port	Cable
	L2	1GbE	UCS 6332 Fabric Interconnect B	L2	1G RJ45
Cisco UCS 6332 Fabric Interconnect B	Eth1/1	40GbE	Nexus 9332 B	Eth 1/21	QSFP-H40G-CU1M
	Eth1/2	40GbE	Nexus 9332 B	Eth 1/22	QSFP-H40G-CU1M
	Eth1/3	40GbE	Nexus 9332 A	Eth 1/23	QSFP-H40G-CU1M
	Eth1/4	40GbE	Nexus 9332 A	Eth 1/24	QSFP-H40G-CU1M
	Eth1/7	40GbE	C220M5	Port1	QSFP-H40G-CU3M
	Eth1/8	40GbE	C220M5	Port2	QSFP-H40G-CU3M
	Eth1/15	40GbE	S3260 Chassis 1 - SIOC 1 (right)	Port 1	QSFP-H40G-CU3M
	Eth1/16	40GbE	S3260 Chassis 1 - SIOC 2 (left)	Port 1	QSFP-H40G-CU3M
	Eth1/17	40GbE	S3260 Chassis 2 - SIOC 1 (right)	Port 1	QSFP-H40G-CU3M
	Eth1/18	40GbE	S3260 Chassis 2 - SIOC 2 (left)	Port 1	QSFP-H40G-CU3M
	Eth1/19	40GbE	S3260 Chassis 3 - SIOC 1 (right)	Port 1	QSFP-H40G-CU3M
	Eth1/20	40GbE	S3260 Chassis 3 - SIOC 2 (left)	Port 1	QSFP-H40G-CU3M
	Eth1/21	40GbE	S3260 Chassis 4 - SIOC 1 (right)	Port 1	QSFP-H40G-CU3M
	Eth1/22	40GbE	S3260 Chassis 4 - SIOC 2 (left)	Port 1	QSFP-H40G-CU3M
	Eth1/23	40GbE	S3260 Chassis 5 - SIOC 1 (right)	Port 1	QSFP-H40G-CU3M
	Eth1/24	40GbE	S3260 Chassis 5 - SIOC 2 (left)	Port 1	QSFP-H40G-CU3M
Eth1/25	40GbE	S3260 Chassis 6 - SIOC 1 (right)	Port 1	QSFP-H40G-CU3M	
Eth1/26	40GbE	S3260 Chassis 6 - SIOC 2 (left)	Port 1	QSFP-H40G-CU3M	

Local Device	Local Port	Connection	Remote Device	Remote Port	Cable
	MGMT0	40GbE	Top of Rack (Management)	Any	1G RJ45
	L1	1GbE	UCS 6332 Fabric Interconnect A	L1	1G RJ45
	L2	1GbE	UCS 6332 Fabric Interconnect A	L2	1G RJ45

Network Topology

Figure 13 illustrates the Network Topology used in the setup.

Figure 13 Network Layout



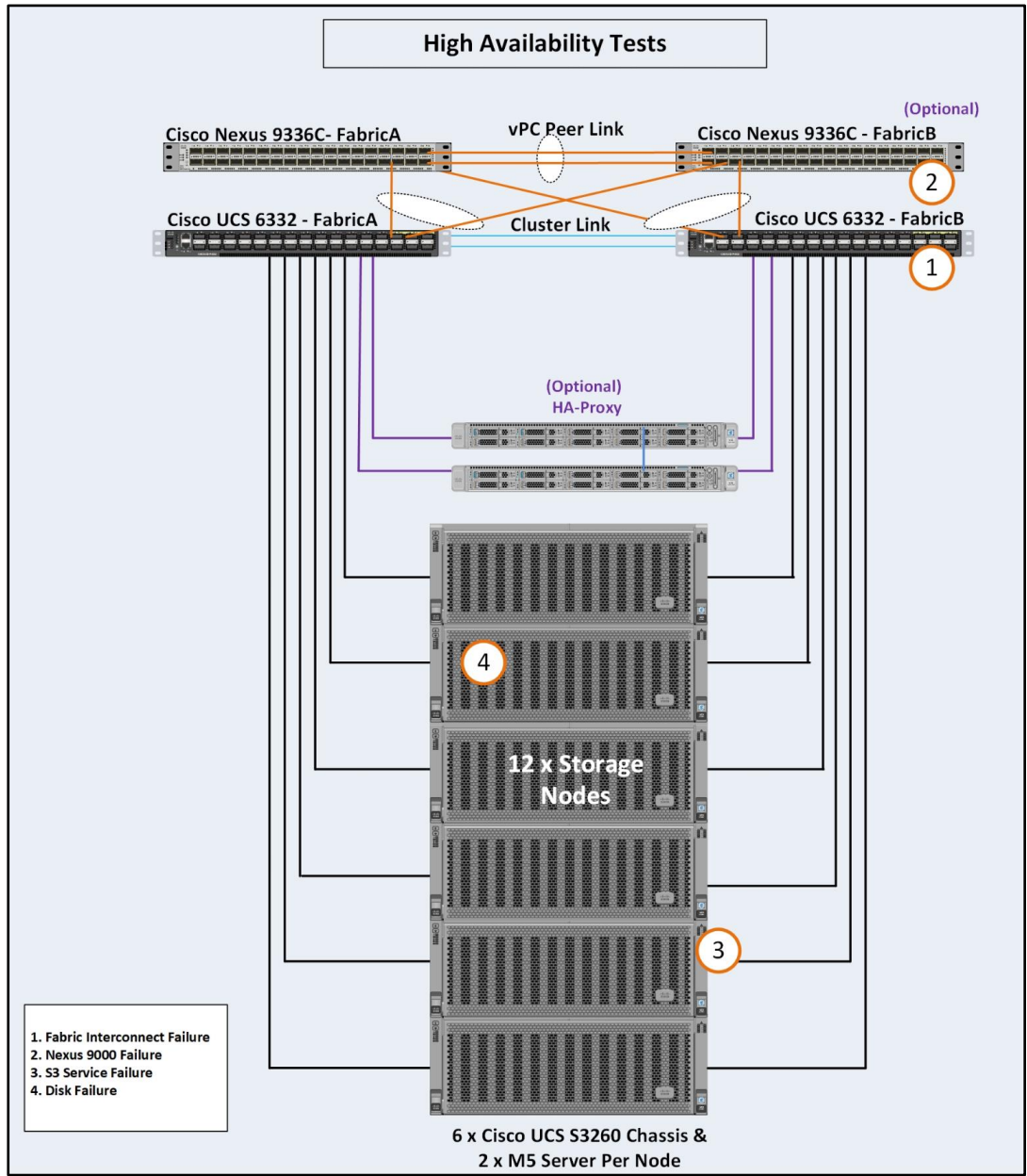
High Availability

As part of the hardware and software resiliency, random read and write load test with objects of 10MB in size will run during the failure injections. The following tests will be conducted on the test bed. The results of the tests will be included in this deployment guide.

- Fabric Interconnect failures
- Nexus 9000 failures

- S3 Service failures
- Disk failures

Figure 14 High Availability Tests



Deployment Hardware and Software

Configuration of Nexus C9336-FX2 Switch A and B

Both Cisco UCS Fabric Interconnect A and B are connected to two Cisco Nexus C9336C-FX2 switches for connectivity to Upstream Network. The following sections describe the setup of both C9336C-FX2 switches.

Initial Setup of Nexus C9336C-FX2 Switch A and B

To configure Switch A, connect a Console to the Console port of each switch, power on the switch and follow these steps:

1. Type **yes**.
2. Type **n**.
3. Type **n**.
4. Type **n**.
5. Enter the switch name.
6. Type **y**.
7. Type your IPv4 management address for Switch A.
8. Type your IPv4 management netmask for Switch A.
9. Type **y**.
10. Type your IPv4 management default gateway address for Switch A.
11. Type **n**.
12. Type **n**.
13. Type **y** for ssh service.
14. Press <Return> and then <Return>.
15. Type **y** for ntp server.
16. Type the IPv4 address of the NTP server.
17. Press <Return>, then <Return> and again <Return>.
18. Check the configuration and if correct then press <Return> and again <Return>.

The complete setup looks like the following:

---- System Admin Account Setup ----

Do you want to enforce secure password standard (yes/no) [y]: **no**

Enter the password for "admin":

Confirm the password for "admin":

---- Basic System Configuration Dialog VDC: 1 ----

This setup utility will guide you through the basic configuration of the system. Setup configures only enough connectivity for management of the system.

Please register Cisco Nexus9000 Family devices promptly with your supplier. Failure to register may affect response times for initial service calls. Nexus9000 devices must be registered to receive entitled support services.

Press Enter at anytime to skip a dialog. Use ctrl-c at anytime to skip the remaining dialogs.

Would you like to enter the basic configuration dialog (yes/no): **yes**

Create another login account (yes/no) [n]:

Configure read-only SNMP community string (yes/no) [n]: **no**

Configure read-write SNMP community string (yes/no) [n]: **no**

Enter the switch name : **N9k-Fab-A**

Continue with Out-of-band (mgmt0) management configuration? (yes/no) [y]: **yes**

Mgmt0 IPv4 address : **173.36.220.133**

Mgmt0 IPv4 netmask : **255.255.255.0**

Configure the default gateway? (yes/no) [y]: **yes**

IPv4 address of the default gateway : **173.36.220.1**

Configure advanced IP options? (yes/no) [n]: **no**

Enable the telnet service? (yes/no) [n]: **no**

```
Enable the ssh service? (yes/no) [y]: yes
  Type of ssh key you would like to generate (dsa/rsa) [rsa]: rsa
  Number of rsa key bits <1024-2048> [1024]: 1024
Configure the ntp server? (yes/no) [n]: yes
  NTP server IPv4 address : 192.168.100.220
Configure default interface layer (L3/L2) [L3]: L2
Configure default switchport interface state (shut/noshut) [shut]: shut
Configure CoPP system profile (strict/moderate/lenient/dense) [strict]:
The following configuration will be applied:
password strength-check
switchname N9k-Fab-A
vrf context management
ip route 0.0.0.0/0 173.36.220.1
exit
no feature telnet
ssh key rsa 1024 force
feature ssh
ntp server 192.168.100.220
no system default switchport
system default switchport shutdown
copp profile strict
interface mgmt0
ip address 173.36.220.133 255.255.255.0
no shutdown

Would you like to edit the configuration? (yes/no) [n]: no

Use this configuration and save it? (yes/no) [y]: yes

[#####] 100%
Copy complete.
```

```
User Access Verification
```

```
N9k-Fab-A login:
```

Repeat these steps for the Nexus C9336C-FX2 Switch B with the exception of configuring a different IPv4 management address 173.36.220.134 as described in step 7.

Enable Features on Nexus C9336C-FX2 Switch A and B

To enable the features UDLD, VLAN, HSRP, LACP, VPC, and Jumbo Frames, connect to the management interface via ssh on both switches and follow these steps on both Switch A and B:

Switch A

```
N9k-Fab-A# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
N9k-Fab-A(config)# feature udld
N9k-Fab-A(config)# feature interface-vlan
N9k-Fab-A(config)# feature hsrp
N9k-Fab-A(config)# feature lacp
N9k-Fab-A(config)# feature vpc
N9k-Fab-A(config)# system jumbomtu 9216
N9k-Fab-A(config)# exit
N9k-Fab-A(config)# copy running-config startup-config
```

Switch B

```
N9k-Fab-B# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
N9k-Fab-B(config)# feature udld
N9k-Fab-B(config)# feature interface-vlan
N9k-Fab-B(config)# feature hsrp
N9k-Fab-B(config)# feature lacp
N9k-Fab-B(config)# feature vpc
N9k-Fab-B(config)# system jumbomtu 9216
N9k-Fab-B(config)# exit
N9k-Fab-B(config)# copy running-config startup-config
```

Configure VLANs on Nexus C9336C-FX2 Switch A and B

To configure the same VLANs Storage-Management, Storage-Cluster, Client Network, and External Management as previously configured in the Cisco UCS Manager GUI, follow these steps on Switch A and Switch B:

Switch A

```
N9k-Fab-A# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
N9k-Fab-A(config)# vlan 100
N9k-Fab-A(config-vlan)# name Storage-Management
N9k-Fab-A(config-vlan)# no shut
N9k-Fab-A(config-vlan)# exit
N9k-Fab-A(config)# vlan 130
N9k-Fab-A(config-vlan)# name Storage-Cluster
N9k-Fab-A(config-vlan)# no shut
N9k-Fab-A(config-vlan)# exit
N9k-Fab-A(config)# vlan 220
N9k-Fab-A(config-vlan)# name External-Mgmt
N9k-Fab-A(config-vlan)# no shut
N9k-Fab-A(config-vlan)# exit
N9k-Fab-A(config)# vlan 120
N9k-Fab-A(config-vlan)# name Client-Network
N9k-Fab-A(config-vlan)# no shut
N9k-Fab-A(config-vlan)# exit

N9k-Fab-A(config)# interface vlan100
N9k-Fab-A(config-if)# description Storage-Mgmt
N9k-Fab-A(config-if)# no shutdown
N9k-Fab-A(config-if)# no ip redirects
N9k-Fab-A(config-if)# ip address 192.168.100.253/24
N9k-Fab-A(config-if)# no ipv6 redirects
N9k-Fab-A(config-if)# hsrp version 2
N9k-Fab-A(config-if)# hsrp 10
N9k-Fab-A(config-if-hsrp)# preempt
N9k-Fab-A(config-if-hsrp)# priority 10
N9k-Fab-A(config-if-hsrp)# ip 192.168.100 .1
N9k-Fab-A(config-if-hsrp)# exit
```

```
N9k-Fab-A(config-if) # exit

N9k-Fab-A(config) # interface vlan130
N9k-Fab-A(config-if) # description Storage-Cluster
N9k-Fab-A(config-if) # no shutdown
N9k-Fab-A(config-if) # no ip redirects
N9k-Fab-A(config-if) # ip address 192.168.130.253/24
N9k-Fab-A(config-if) # no ipv6 redirects
N9k-Fab-A(config-if) # hsrp version 2
N9k-Fab-A(config-if) # hsrp 20
N9k-Fab-A(config-if-hsrp) # preempt
N9k-Fab-A(config-if-hsrp) # priority 10
N9k-Fab-A(config-if-hsrp) # ip 192.168.130.1
N9k-Fab-A(config-if-hsrp) # exit
N9k-Fab-A(config) # interface vlan120
N9k-Fab-A(config-if) # description Client-Network
N9k-Fab-A(config-if) # no shutdown
N9k-Fab-A(config-if) # no ip redirects
N9k-Fab-A(config-if) # ip address 192.168.120.253/24
N9k-Fab-A(config-if) # no ipv6 redirects
N9k-Fab-A(config-if) # hsrp version 2
N9k-Fab-A(config-if) # hsrp 30
N9k-Fab-A(config-if-hsrp) # preempt
N9k-Fab-A(config-if-hsrp) # priority 10
N9k-Fab-A(config-if-hsrp) # ip 192.168.120.1
N9k-Fab-A(config-if-hsrp) # exit
N9k-Fab-A(config-if) # exit
```

Switch B

```
N9k-Fab-B# config terminal

Enter configuration commands, one per line. End with CNTL/Z.

N9k-Fab-B(config) # vlan 100
```

```
N9k-Fab-B(config-vlan)# name Storage-Management
N9k-Fab-B(config-vlan)# no shut
N9k-Fab-B(config-vlan)# exit
N9k-Fab-B(config)# vlan 130
N9k-Fab-B(config-vlan)# name Storage-Cluster
N9k-Fab-B(config-vlan)# no shut
N9k-Fab-B(config-vlan)# exit
N9k-Fab-B(config)# vlan 220
N9k-Fab-B(config-vlan)# name External-Mgmt
N9k-Fab-B(config-vlan)# no shut
N9k-Fab-B(config-vlan)# exit
N9k-Fab-B(config)# vlan 120
N9k-Fab-B(config-vlan)# name Client-Network
N9k-Fab-B(config-vlan)# no shut
N9k-Fab-B(config-vlan)# exit
N9k-Fab-B(config)# interface vlan100
N9k-Fab-B(config-if)# description Storage-Mgmt
N9k-Fab-B(config-if)# no ip redirects
N9k-Fab-B(config-if)# ip address 192.168.100.254/24
N9k-Fab-B(config-if)# no ipv6 redirects
N9k-Fab-B(config-if)# hsrp version 2
N9k-Fab-B(config-if)# hsrp 10
N9k-Fab-B(config-if-hsrp)# preempt
N9k-Fab-B(config-if-hsrp)# priority 5
N9k-Fab-B(config-if-hsrp)# ip 192.168.100.1
N9k-Fab-B(config-if-hsrp)# exit
N9k-Fab-B(config-if)# exit

N9k-Fab-B(config)# interface vlan130
N9k-Fab-B(config-if)# description Storage-Cluster
N9k-Fab-B(config-if)# no ip redirects
N9k-Fab-B(config-if)# ip address 192.168.130.254/24
```

```

N9k-Fab-B(config-if)# no ipv6 redirects
N9k-Fab-B(config-if)# hsrp version 2
N9k-Fab-B(config-if)# hsrp 20
N9k-Fab-B(config-if-hsrp)# preempt
N9k-Fab-B(config-if-hsrp)# priority 5
N9k-Fab-B(config-if-hsrp)# ip 192.168.130.1
N9k-Fab-B(config-if-hsrp)# exit
N9k-Fab-B(config)# interface vlan120
N9k-Fab-B(config-if)# description Storage-Cluster
N9k-Fab-B(config-if)# no ip redirects
N9k-Fab-B(config-if)# ip address 192.168.120.254/24
N9k-Fab-B(config-if)# no ipv6 redirects
N9k-Fab-B(config-if)# hsrp version 2
N9k-Fab-B(config-if)# hsrp 30
N9k-Fab-B(config-if-hsrp)# preempt
N9k-Fab-B(config-if-hsrp)# priority 5
N9k-Fab-B(config-if-hsrp)# ip 192.168.120.1
N9k-Fab-B(config-if-hsrp)# exit
N9k-Fab-B(config-if)# exit
N9k-Fab-B(config)# copy running-config startup-config

```

Configure vPC and Port Channels on Nexus C9336C-FX2 Switch A and B

To enable vPC and Port Channels on both Switch A and B, follow these steps:

vPC and Port Channels for Peerlink on Switch A

```

N9k-Fab-B# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
N9k-Fab-A(config)# vpc domain 2
N9k-Fab-A(config-vpc-domain)# peer-keepalive destination 192.168.10.104
Note:
-----:: Management VRF will be used as the default VRF ::-----
N9k-Fab-A(config-vpc-domain)# peer-gateway
N9k-Fab-A(config-vpc-domain)# exit

```

```
N9k-Fab-A(config)# interface port-channel 1
N9k-Fab-A(config-if)# description vPC peerlink for N9k-Fab-A and N9k-Fab-B
N9k-Fab-A(config-if)# switchport
N9k-Fab-A(config-if)# switchport mode trunk
N9k-Fab-A(config-if)# spanning-tree port type network
N9k-Fab-A(config-if)# speed 40000
N9k-Fab-A(config-if)# vpc peer-link

Please note that spanning tree port type is changed to "network" port type on vPC
peer-link.

This will enable spanning tree Bridge Assurance on vPC peer-link provided the STP
Bridge Assurance

(which is enabled by default) is not disabled.

N9k-Fab-A(config-if)# exit

N9k-Fab-A(config)# interface ethernet 1/1
N9k-Fab-A(config-if)# description connected to peer N9k-Fab-B port 1
N9k-Fab-A(config-if)# switchport
N9k-Fab-A(config-if)# switchport mode trunk
N9k-Fab-A(config-if)# speed 40000
N9k-Fab-A(config-if)# channel-group 1 mode active
N9k-Fab-A(config-if)# exit

N9k-Fab-A(config)# interface ethernet 1/2
N9k-Fab-A(config-if)# description connected to peer N9k-Fab-B port 2
N9k-Fab-A(config-if)# switchport
N9k-Fab-A(config-if)# switchport mode trunk
N9k-Fab-A(config-if)# speed 40000
N9k-Fab-A(config-if)# channel-group 1 mode active
N9k-Fab-A(config-if)# exit
N9k-Fab-A(config)# copy running-config startup-config
```

vPC and Port Channels for Peerlink on Switch B

```
N9k-Fab-B# config terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
N9k-Fab-B(config)# vpc domain 2
```

```
N9k-Fab-B(config-vpc-domain)# peer-keepalive destination 192.168.10.103
```

Note:

```
-----:: Management VRF will be used as the default VRF ::-----
```

```
N9k-Fab-B(config-vpc-domain)# peer-gateway
```

```
N9k-Fab-B(config-vpc-domain)# exit
```

```
N9k-Fab-B(config)# interface port-channel 1
```

```
N9k-Fab-B(config-if)# description vPC peerlink for N9k-Fab-A and N9k-Fab-B
```

```
N9k-Fab-B(config-if)# switchport
```

```
N9k-Fab-B(config-if)# switchport mode trunk
```

```
N9k-Fab-B(config-if)# spanning-tree port type network
```

```
N9k-Fab-B(config-if)# speed 40000
```

```
N9k-Fab-B(config-if)# vpc peer-link
```

Please note that spanning tree port type is changed to "network" port type on vPC peer-link.

This will enable spanning tree Bridge Assurance on vPC peer-link provided the STP Bridge Assurance

(which is enabled by default) is not disabled.

```
N9k-Fab-B(config-if)# exit
```

```
N9k-Fab-B(config)# interface ethernet 1/1
```

```
N9k-Fab-B(config-if)# description connected to peer N9k-Fab-A port 1
```

```
N9k-Fab-B(config-if)# switchport
```

```
N9k-Fab-B(config-if)# switchport mode trunk
```

```
N9k-Fab-B(config-if)# speed 40000
```

```
N9k-Fab-B(config-if)# channel-group 1 mode active
```

```
N9k-Fab-B(config-if)# exit
```

```
N9k-Fab-B(config)# interface ethernet 1/2
```

```
N9k-Fab-B(config-if)# description connected to peer N9k-Fab-A port 2
```

```
N9k-Fab-B(config-if)# switchport
```

```

N9k-Fab-B(config-if)# switchport mode trunk
N9k-Fab-B(config-if)# speed 40000
N9k-Fab-B(config-if)# channel-group 1 mode active
N9k-Fab-B(config-if)# exit
N9k-Fab-B(config)# copy running-config startup-config

```

vPC and Port Channels for Uplink from UCS Fabric A & B on Nexus Switch A

```

N9k-Fab-B# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
N9k-Fab-A(config)# interface port-channel 25
N9k-Fab-A(config-if)# description vPC for UCS FI-A ports 25 to 26
N9k-Fab-A(config-if)# vpc 25
N9k-Fab-A(config-if)# switchport
N9k-Fab-A(config-if)# switchport mode trunk
N9k-Fab-A(config-if)# switchport trunk allowed vlan 10,20,30,79
N9k-Fab-A(config-if)# spanning-tree port type edge trunk
Edge port type (portfast) should only be enabled on ports connected to a single
host. Connecting hubs, concentrators, switches, bridges, etc... to this
interface when edge port type (portfast) is enabled, can cause temporary
bridging loops.

Use with CAUTION
N9k-Fab-A(config-if)# mtu 9216
N9k-Fab-A(config-if)# exit

N9k-Fab-A(config)# interface port-channel 26
N9k-Fab-A(config-if)# description vPC for UCS FI-B ports 25 to 26
N9k-Fab-A(config-if)# vpc 26
N9k-Fab-A(config-if)# switchport
N9k-Fab-A(config-if)# switchport mode trunk
N9k-Fab-A(config-if)# switchport trunk allowed vlan 100,120,130,220
N9k-Fab-A(config-if)# spanning-tree port type edge trunk
Edge port type (portfast) should only be enabled on ports connected to a single

```

host. Connecting hubs, concentrators, switches, bridges, etc... to this interface when edge port type (portfast) is enabled, can cause temporary bridging loops.

Use with CAUTION

```
N9k-Fab-A(config-if)# mtu 9216
```

```
N9k-Fab-A(config-if)# exit
```

```
N9k-Fab-A(config)# interface ethernet 1/25
```

```
N9k-Fab-A(config-if-range)# switchport
```

```
N9k-Fab-A(config-if-range)# switchport mode trunk
```

```
N9k-Fab-A(config-if-range)# description Uplink from UCS FI-A ports 25
```

```
N9k-Fab-A(config-if-range)# channel-group 25 mode active
```

```
N9k-Fab-A(config-if)# exit
```

```
N9k-Fab-A(config)# interface ethernet 1/26
```

```
N9k-Fab-A(config-if-range)# switchport
```

```
N9k-Fab-A(config-if-range)# switchport mode trunk
```

```
N9k-Fab-A(config-if-range)# description Uplink from UCS FI-B ports 26
```

```
N9k-Fab-A(config-if-range)# channel-group 26 mode active
```

```
N9k-Fab-A(config-if)# exit
```

```
N9k-Fab-A(config)# copy running-config startup-config
```

vPC and Port Channels for Uplink from Fabric A and B on Nexus Switch B

```
N9k-Fab-B# config terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
N9k-Fab-B(config)# interface port-channel 25
```

```
N9k-Fab-B(config-if)# description vPC for UCS FI-A ports 25 to 26
```

```
N9k-Fab-B(config-if)# switchport
```

```
N9k-Fab-B(config-if)# switchport mode trunk
```

```
N9k-Fab-B(config-if)# switchport trunk allowed vlan 100,120,130,220
```

```
N9k-Fab-B(config-if)# spanning-tree port type edge trunk
```

Edge port type (portfast) should only be enabled on ports connected to a single

host. Connecting hubs, concentrators, switches, bridges, etc... to this interface when edge port type (portfast) is enabled, can cause temporary bridging loops.

Use with CAUTION

```
N9k-Fab-B(config-if)# vpc 25
```

```
N9k-Fab-B(config-if)# mtu 9216
```

```
N9k-Fab-B(config-if)# exit
```

```
N9k-Fab-B(config)# interface port-channel 26
```

```
N9k-Fab-B(config-if)# description vPC for UCS FI-B ports 25 to 26
```

```
N9k-Fab-B(config-if)# switchport
```

```
N9k-Fab-B(config-if)# switchport mode trunk
```

```
N9k-Fab-B(config-if)# switchport trunk allowed vlan 100,120,130,220
```

```
N9k-Fab-B(config-if)# spanning-tree port type edge trunk
```

Edge port type (portfast) should only be enabled on ports connected to a single host. Connecting hubs, concentrators, switches, bridges, etc... to this interface when edge port type (portfast) is enabled, can cause temporary bridging loops.

Use with CAUTION

```
N9k-Fab-B(config-if)# vpc 26
```

```
N9k-Fab-B(config-if)# mtu 9216
```

```
N9k-Fab-B(config-if)# exit
```

```
N9k-Fab-B(config)# interface ethernet 1/25
```

```
N9k-Fab-B(config-if-range)# switchport
```

```
N9k-Fab-B(config-if-range)# switchport mode trunk
```

```
N9k-Fab-B(config-if-range)# description Uplink from UCS FI-A ports 25 to 26
```

```
N9k-Fab-B(config-if-range)# channel-group 25 mode active
```

```
N9k-Fab-B(config-if)# exit
```

```
N9k-Fab-B(config)# interface ethernet 1/26
```

```
N9k-Fab-B(config-if-range)# switchport
```

```
N9k-Fab-B(config-if-range)# switchport mode trunk
```

```

N9k-Fab-B(config-if-range)# description Uplink from UCS FI-B ports 25 to 26
N9k-Fab-B(config-if-range)# channel-group 26 mode active
N9k-Fab-B(config-if)# exit
N9k-Fab-B(config)# copy running-config startup-config

```

Verification Check of Nexus C9336C-FX2 Configuration for Switch A and B

Switch A

```
N9k-Fab-B# config terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
N9k-Fab-A(config)# show vpc brief
```

Legend:

(*) - local vPC is down, forwarding via vPC peer-link

```

vPC domain id                : 2
Peer status                   : peer adjacency formed ok
vPC keep-alive status        : peer is alive
Configuration consistency status : success
Per-vlan consistency status   : success
Type-2 consistency status    : success
vPC role                      : secondary
Number of vPCs configured    : 4
Peer Gateway                  : Enabled
Dual-active excluded VLANs   : -
Graceful Consistency Check   : Enabled
Auto-recovery status         : Disabled
Delay-restore status         : Timer is off.(timeout = 30s)
Delay-restore SVI status     : Timer is off.(timeout = 10s)

```

vPC Peer-link status

```

-----
id   Port   Status Active vlans
--   ---   -
1    Po1    up     1,100,120,130,220

```

vPC status

```

-----
id   Port   Status Consistency Reason           Active vlans
--   -
25   Po25   up      success    success           100,120,130,220
26   Po26   up      success    success           100,120,130,220
    
```

N9k-Fab-A(config)#

N9k-Fab-A(config)# show port-channel summary

```

Flags:  D - Down          P - Up in port-channel (members)
        I - Individual    H - Hot-standby (LACP only)
        s - Suspended     r - Module-removed
        S - Switched      R - Routed
        U - Up (port-channel)
        p - Up in delay-lacp mode (member)
        M - Not in use. Min-links not met
    
```

```

-----
Group Port-          Type      Protocol  Member Ports
      Channel
-----
1     Po1(SU)          Eth       LACP      Eth1/1(P)  Eth1/2(P)
25    Po25(SU)         Eth       LACP      Eth1/25(P)
26    Po26(SU)         Eth       LACP      Eth1/26(P)
    
```

N9k-Fab-A(config)#

Switch B

N9k-Fab-B# config terminal

Enter configuration commands, one per line. End with CNTL/Z.

N9k-Fab-B(config)# show vpc brief

Legend:

(*) - local vPC is down, forwarding via vPC peer-link

```

vPC domain id          : 2
Peer status            : peer adjacency formed ok
vPC keep-alive status  : peer is alive
Configuration consistency status : success
Per-vlan consistency status : success
Type-2 consistency status : success
vPC role               : primary
Number of vPCs configured : 4
Peer Gateway           : Enabled
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled
Auto-recovery status   : Disabled
Delay-restore status   : Timer is off.(timeout = 30s)
Delay-restore SVI status : Timer is off.(timeout = 10s)
    
```

vPC Peer-link status

```

-----
id  Port  Status Active vlans
--  ---  -
1   Po1   up     1,100,120,130,220
    
```

vPC status

```

-----
id  Port  Status Consistency Reason           Active vlans
--  ---  -
25  Po25  up     success    success           10,20,30,79
26  Po26  up     success    success           100,120,130,220
    
```

N9k-Fab-B(config)#

N9k-Fab-B(config)# show port-channel summary

Flags: D - Down P - Up in port-channel (members)

```

I - Individual      H - Hot-standby (LACP only)
s - Suspended      r - Module-removed
S - Switched       R - Routed
U - Up (port-channel)
p - Up in delay-lACP mode (member)
M - Not in use. Min-links not met
    
```

Group	Port-Channel	Type	Protocol	Member	Ports
1	Po1 (SU)	Eth	LACP	Eth1/1 (P)	Eth1/2 (P)
25	Po25 (SU)	Eth	LACP	Eth1/25 (P)	
26	Po26 (SU)	Eth	LACP	Eth1/26 (P)	

Fabric Interconnect Configuration

This section provides the details to configure a fully redundant, highly available Cisco UCS 6332 fabric configuration:

- Initial setup of the Fabric Interconnect A and B
- Connect to Cisco UCS Manager using virtual IP address or using the web browser
- Launch Cisco UCS Manager
- Enable server and uplink ports
- Start discovery process
- Create pools and policies for service profile template
- Create chassis and storage profiles
- Create Service Profile templates and appropriate Service Profiles
- Associate Service Profiles to servers

Initial Setup of Cisco UCS 6332 Fabric Interconnects

The following section describes the initial setup of the Cisco UCS 6332 Fabric Interconnects A and B.

Configure Fabric Interconnect A

To configure Fabric A, follow these steps:

1. Connect to the console port on the first Cisco UCS 6332 Fabric Interconnect.

2. At the prompt to enter the configuration method, enter **console** to continue.
3. If asked to either perform a new setup or restore from backup, enter **setup** to continue.
4. Enter **y** to continue to set up a new Fabric Interconnect.
5. Enter **n** to enforce strong passwords.
6. Enter the password for the admin user.
7. Enter the same password again to confirm the password for the admin user.
8. When asked if this fabric interconnect is part of a cluster, answer **y** to continue.
9. Enter **A** for the switch fabric.
10. Enter the cluster name UCS-**FI-6332** for the system name.
11. Enter the Mgmt0 IPv4 address.
12. Enter the Mgmt0 IPv4 netmask.
13. Enter the IPv4 address of the default gateway.
14. Enter the cluster IPv4 address.
15. To configure DNS, answer **y**.
16. Enter the DNS IPv4 address.
17. Answer **y** to set up the default domain name.
18. Enter the default domain name.
19. Review the settings that were printed to the console, and if they are correct, answer **yes** to save the configuration.
20. Wait for the login prompt to make sure the configuration has been saved.

Example Setup for Fabric Interconnect A

```
---- Basic System Configuration Dialog ----
```

```
This setup utility will guide you through the basic configuration of
the system. Only minimal configuration including IP connectivity to
the Fabric interconnect and its clustering mode is performed through these
steps.
```

```
Type Ctrl-C at any time to abort configuration and reboot system.
```

To back track or make modifications to already entered values,
complete input till end of section and answer no when prompted
to apply configuration.

Enter the configuration method. (console/gui) ? **console**

Enter the setup mode; setup newly or restore from backup. (setup/restore) ?
setup

You have chosen to setup a new Fabric interconnect. Continue? (y/n): **y**

Enforce strong password? (y/n) [y]: **n**

Enter the password for "admin":

Confirm the password for "admin":

Is this Fabric interconnect part of a cluster(select 'no' for standalone)?
(yes/no) [n]: **yes**

Enter the switch fabric (A/B): **A**

Enter the system name: **UCS-FI-6332**

Physical Switch Mgmt0 IP address : **173.36.220.135**

Physical Switch Mgmt0 IPv4 netmask : **255.255.255.0**

IPv4 address of the default gateway : **173.36.220.1**

Cluster IPv4 address : **173.36.220.137**

Configure the DNS Server IP address? (yes/no) [n]: **no**

Configure the default domain name? (yes/no) [n]: **no**

Join centralized management environment (UCS Central)? (yes/no) [n]: **no**

Following configurations will be applied:

Switch Fabric=A

System Name= UCS-FI-6332

Enforced Strong Password=no

Physical Switch Mgmt0 IP Address=173.36.220.135

Physical Switch Mgmt0 IP Netmask=255.255.255.0

Default Gateway=173.36.220.1

Ipv6 value=0

```
Cluster Enabled=yes
```

```
Cluster IP Address=173.36.252.137
```

NOTE: Cluster IP will be configured only after both Fabric Interconnects are initialized.

UCSM will be functional only after peer FI is configured in clustering mode.

```
Apply and save the configuration (select 'no' if you want to re-enter)?  
(yes/no): yes
```

```
Applying configuration. Please wait.
```

```
Configuration file - Ok
```

```
Cisco UCS 6300 Series Fabric Interconnect
```

```
UCS-FI-6332-A login:
```

Configure Fabric Interconnect B

To configure Fabric Interconnect B, follow these steps:

1. Connect to the console port on the second Cisco UCS 6332 Fabric Interconnect.
2. When prompted to enter the configuration method, enter **console** to continue.
3. The installer detects the presence of the partner Fabric Interconnect and adds this fabric interconnect to the cluster. Enter **y** to continue the installation.
4. Enter the admin password that was configured for the first Fabric Interconnect.
5. Enter the Mgmt0 IPv4 address.
6. Answer **yes** to save the configuration.
7. Wait for the login prompt to confirm that the configuration has been saved.

Example Setup for Fabric Interconnect B

```
---- Basic System Configuration Dialog ----
```

```
This setup utility will guide you through the basic configuration of  
the system. Only minimal configuration including IP connectivity to  
the Fabric interconnect and its clustering mode is performed through these  
steps.
```


Type Ctrl-C at any time to abort configuration and reboot system.
To back track or make modifications to already entered values,
complete input till end of section and answer no when prompted
to apply configuration.

Enter the configuration method. (console/gui) ? **console**

Installer has detected the presence of a peer Fabric interconnect. This Fabric
interconnect will be added to the cluster. Continue (y/n) ? **y**

Enter the admin password of the peer Fabric interconnect:

Connecting to peer Fabric interconnect... done

Retrieving config from peer Fabric interconnect... done

Peer Fabric interconnect Mgmt0 IPv4 Address: 173.36.220.135

Peer Fabric interconnect Mgmt0 IPv4 Netmask: 255.255.255.0

Cluster IPv4 address : 173.36.220.137

Peer FI is IPv4 Cluster enabled. Please Provide Local Fabric Interconnect
Mgmt0 IPv4 Address

Physical Switch Mgmt0 IP address : **173.36.220.136**

Apply and save the configuration (select 'no' if you want to re-enter)?
(yes/no) : **yes**

Applying configuration. Please wait.

Configuration file - Ok

Cisco UCS 6300 Series Fabric Interconnect

UCS-FI-6332-B login:

Log into Cisco UCS Manager

To log into Cisco UCS Manager, follow these steps:

1. Open a Web browser and navigate to the Cisco UCS 6332 Fabric Interconnect cluster address.
2. Click the Launch link to download the Cisco UCS Manager software.

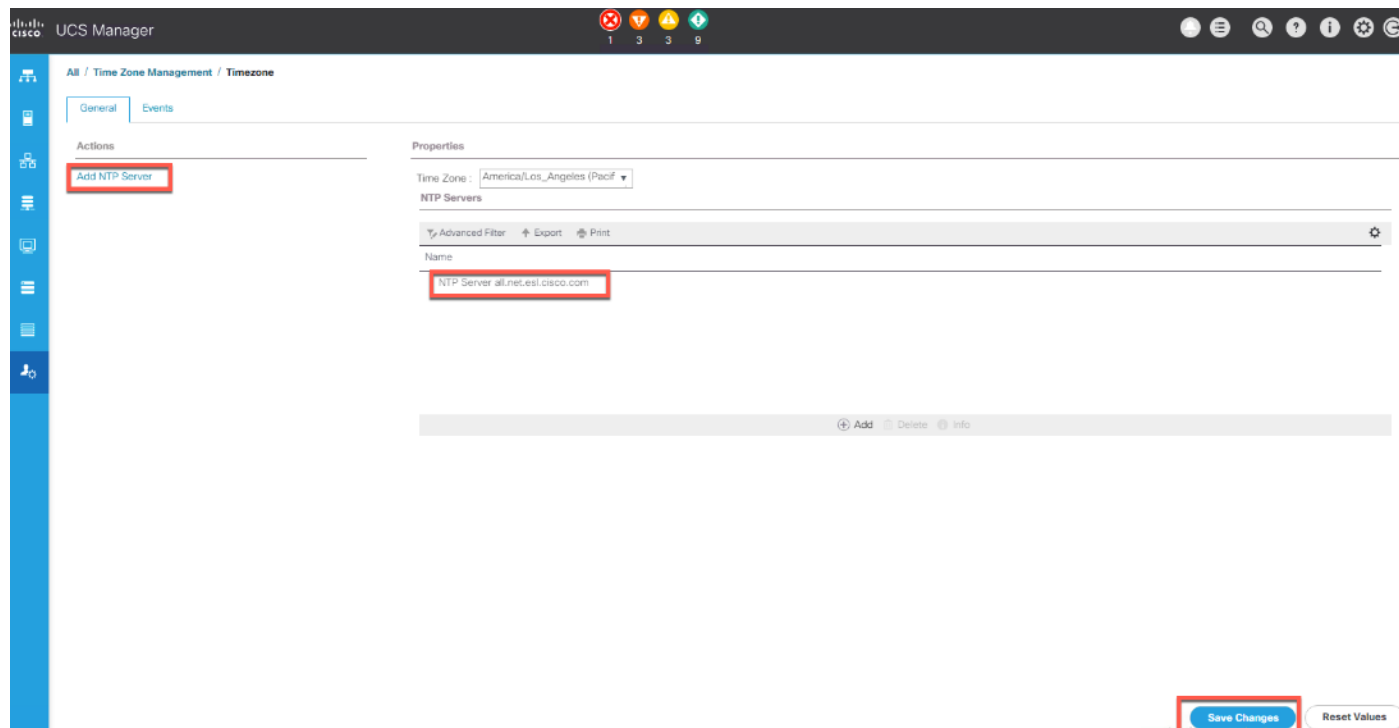
3. If prompted to accept security certificates, accept as necessary.
4. Click Launch UCS Manager HTML.
5. When prompted, enter admin for the username and enter the administrative password.
6. Click Login to log in to the Cisco UCS Manager.

Configure NTP Server

To configure the NTP server for the Cisco UCS environment, follow these steps:

1. Select Admin tab.
2. Select Time Zone Management.
3. Select Time Zone.
4. Under Properties select your time zone.
5. Select Add NTP Server.
6. Enter the IP address/DNS name of the NTP server.
7. Select OK.

Figure 15 Adding a NTP Server - Summary



Initial Base Setup of the Environment

Configure Global Policies

To configure the global policies, follow these steps:

1. Select the **Equipment** tab. of the window.
2. Select **Policies** on the right site.
3. Select Global Policies.
4. Under Chassis/FEX Discovery Policy select **Platform Max** under Action.
5. Select **40G** under Backplane Speed Preference.
6. Under Rack Server Discovery Policy select **Immediate** under Action.
7. Under Rack Management Connection Policy select **Auto Acknowledged** under Action.
8. Under Power Policy select **Redundancy N+1**.
9. Under Global Power Allocation Policy select **Policy Driven Chassis Group Cap**.
10. Select Save Changes.

Figure 16 Configuration of Global Policies

The screenshot displays the 'Equipment' configuration page, specifically the 'Policies' tab. The interface includes a navigation menu on the left and a main content area with several policy sections:

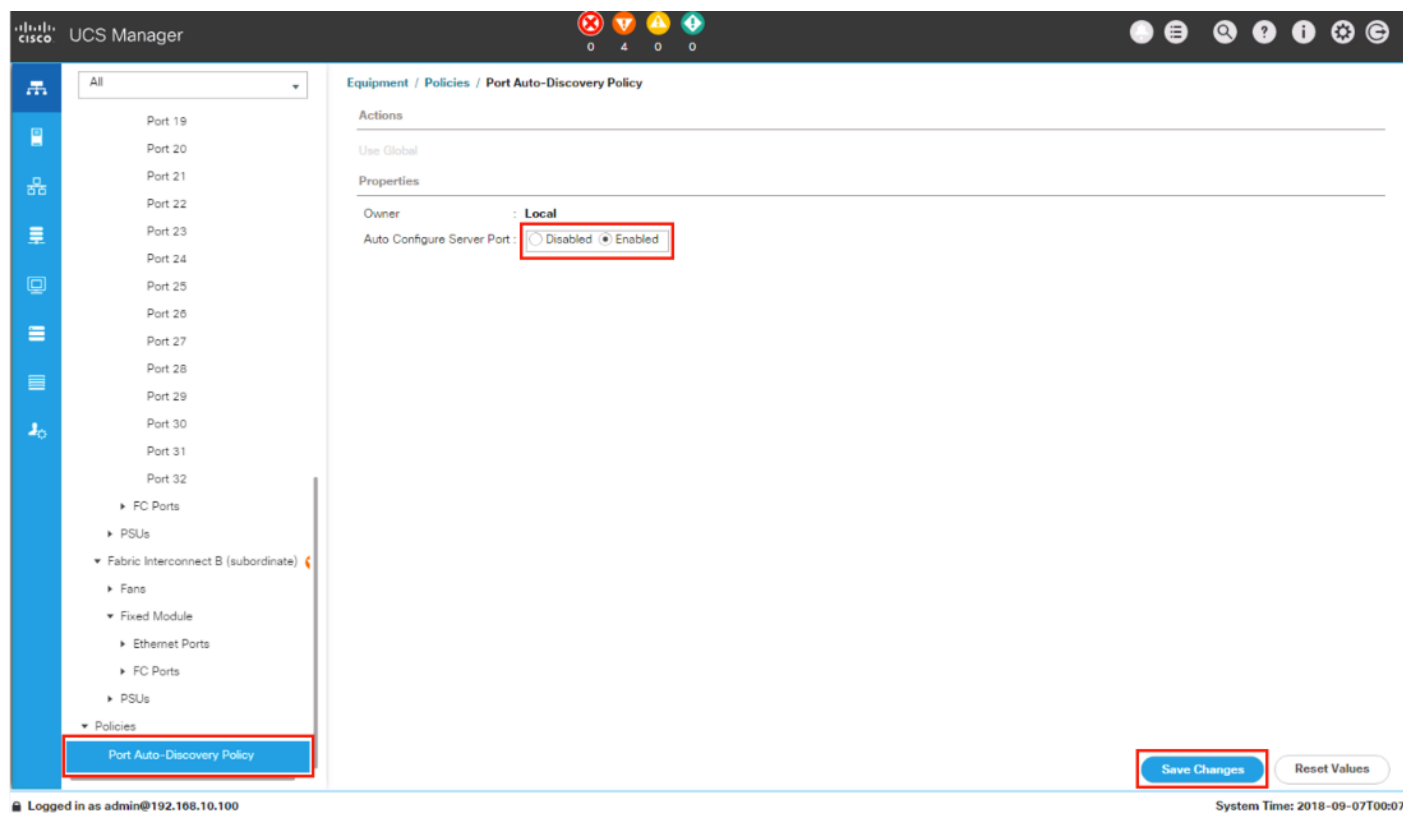
- Chassis/FEX Discovery Policy:**
 - Action:
 - Link Grouping Preference: None Port Channel
 - Backplane Speed Preference: 40G 4x10G
- Rack Server Discovery Policy:**
 - Action: Immediate User Acknowledged
 - Scrub Policy:
- Rack Management Connection Policy:**
 - Action: Auto Acknowledged User Acknowledged
- Power Policy:**
 - Redundancy: Non Redundant N+1 Grid
- MAC Address Table Aging:**
 - Aging Time: Never Mode Default other
- Global Power Allocation Policy:**
 - Allocation Method: Manual Blade Level Cap Policy Driven Chassis Group Cap

Enable Fabric Interconnect Server Ports

To enable server ports, follow these steps:

1. Select the **Equipment** tab.
2. Select Equipment > Policies > Port-Auto Discovery Policy
3. Click **Enabled** Under Properties
4. Click **Save Changes** to Configure Server Ports Automatically for FI-A and FI-B.

Figure 17 Configuration of Server Ports



5. Verify the ports Server port on Fabric Interconnect A.
6. Select Equipment > Fabric Interconnects > Fabric Interconnect A (primary) > Fixed Module.
7. Click **Ethernet Ports** section.

Figure 18 FI-A Server Ports Status

Slot	Aggr. Port ID	Port ID	MAC	If Role	If Type	Overall Status	Admin State	Peer
1	0	15	B0:8B:CF:A4:1D:36	Server	Physical	Up	Enabled	sys/chassis-4/slot-...
1	0	16	B0:8B:CF:A4:1D:3A	Server	Physical	Up	Enabled	sys/chassis-4/slot-...
1	0	17	B0:8B:CF:A4:1D:3E	Server	Physical	Up	Enabled	sys/chassis-1/slot-...
1	0	18	B0:8B:CF:A4:1D:42	Server	Physical	Up	Enabled	sys/chassis-1/slot-...
1	0	19	B0:8B:CF:A4:1D:48	Server	Physical	Up	Enabled	sys/chassis-3/slot-...
1	0	20	B0:8B:CF:A4:1D:4A	Server	Physical	Up	Enabled	sys/chassis-3/slot-...
1	0	21	B0:8B:CF:A4:1D:4E	Server	Physical	Up	Enabled	sys/chassis-6/slot-...
1	0	22	B0:8B:CF:A4:1D:52	Server	Physical	Up	Enabled	sys/chassis-6/slot-...
1	0	23	B0:8B:CF:A4:1D:56	Server	Physical	Up	Enabled	sys/chassis-5/slot-...
1	0	24	B0:8B:CF:A4:1D:5A	Server	Physical	Up	Enabled	sys/chassis-5/slot-...
1	0	25	B0:8B:CF:A4:1D:5E	Server	Physical	Up	Enabled	sys/chassis-2/slot-...
1	0	26	B0:8B:CF:A4:1D:62	Server	Physical	Up	Enabled	sys/chassis-2/slot-...

8. Verify the ports Server port on Fabric Interconnect A.

9. Select Equipment > Fabric Interconnects > Fabric Interconnect B (subordinate) > Fixed Module.

10. Click **Ethernet Ports** section.

Figure 19 FI-B Server Ports Status

Slot	Aggr. Port ID	Port ID	MAC	If Role	If Type	Overall Status	Admin State	Peer
1	0	15	A8:B4:56:D7:99:02	Server	Physical	Up	Enabled	sys/chassis-4/slot-...
1	0	16	A8:B4:56:D7:99:06	Server	Physical	Up	Enabled	sys/chassis-4/slot-...
1	0	17	A8:B4:56:D7:99:0A	Server	Physical	Up	Enabled	sys/chassis-1/slot-...
1	0	18	A8:B4:56:D7:99:0E	Server	Physical	Up	Enabled	sys/chassis-1/slot-...
1	0	19	A8:B4:56:D7:99:12	Server	Physical	Up	Enabled	sys/chassis-3/slot-...
1	0	20	A8:B4:56:D7:99:16	Server	Physical	Up	Enabled	sys/chassis-3/slot-...
1	0	21	A8:B4:56:D7:99:1A	Server	Physical	Up	Enabled	sys/chassis-6/slot-...
1	0	22	A8:B4:56:D7:99:1E	Server	Physical	Up	Enabled	sys/chassis-6/slot-...
1	0	23	A8:B4:56:D7:99:22	Server	Physical	Up	Enabled	sys/chassis-5/slot-...
1	0	24	A8:B4:56:D7:99:26	Server	Physical	Up	Enabled	sys/chassis-5/slot-...
1	0	25	A8:B4:56:D7:99:2A	Server	Physical	Up	Enabled	sys/chassis-2/slot-...
1	0	26	A8:B4:56:D7:99:2E	Server	Physical	Up	Enabled	sys/chassis-2/slot-...
1	0	27	A8:B4:56:D7:99:32	Unconfigured	Physical	Sfp Not Present	Disabled	
1	0	28	A8:B4:56:D7:99:33	Unconfigured	Physical	Sfp Not Present	Disabled	

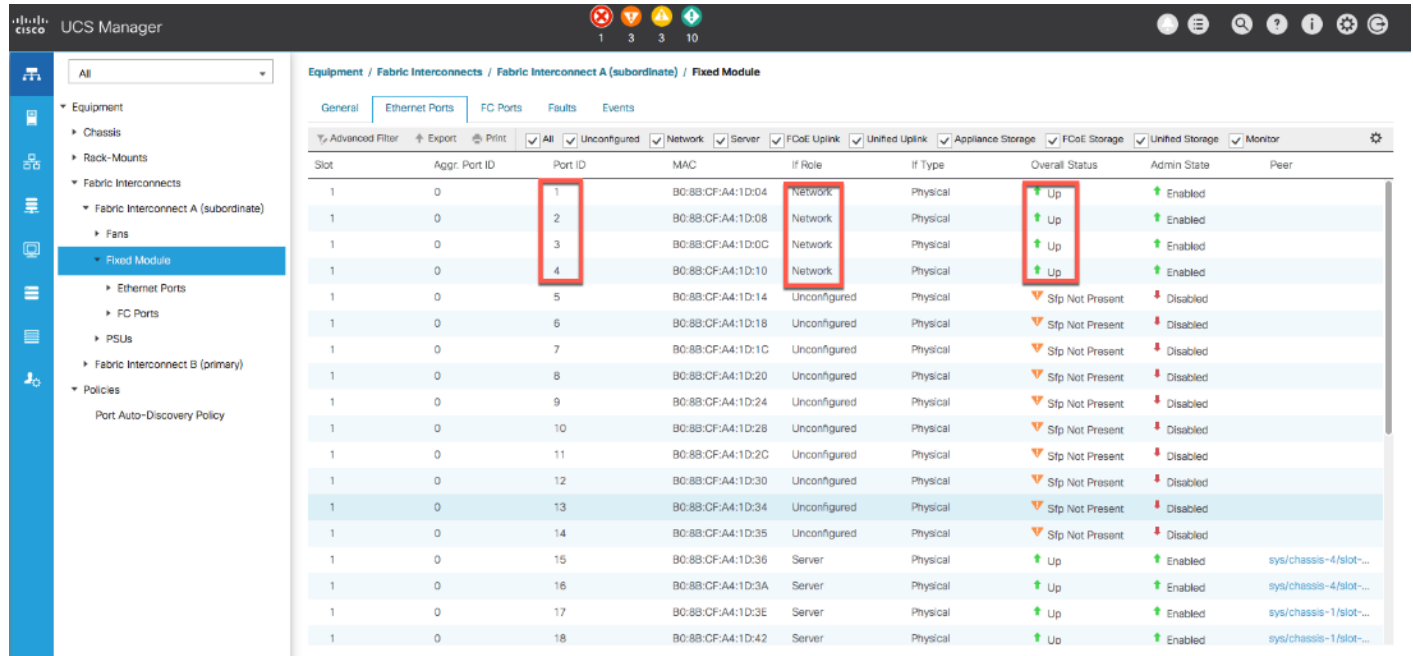
Enable Fabric Interconnect A Ports for Uplinks

To enable uplink ports, follow these steps:

1. Select the **Equipment** tab.
2. Select Equipment > Fabric Interconnects > Fabric Interconnect A (primary) > Fixed Storage Module.
3. Click **Ethernet Ports** section.

4. Select Ports 1-4, right-click and then select **Configure as Uplink Port**.
5. Click **Yes** and then **OK**.
6. Repeat steps 1-5 or Fabric Interconnect B.

Figure 20 Configuring of Network Uplink Ports



Label Servers for Identification

For better identification, label each server by following these steps:

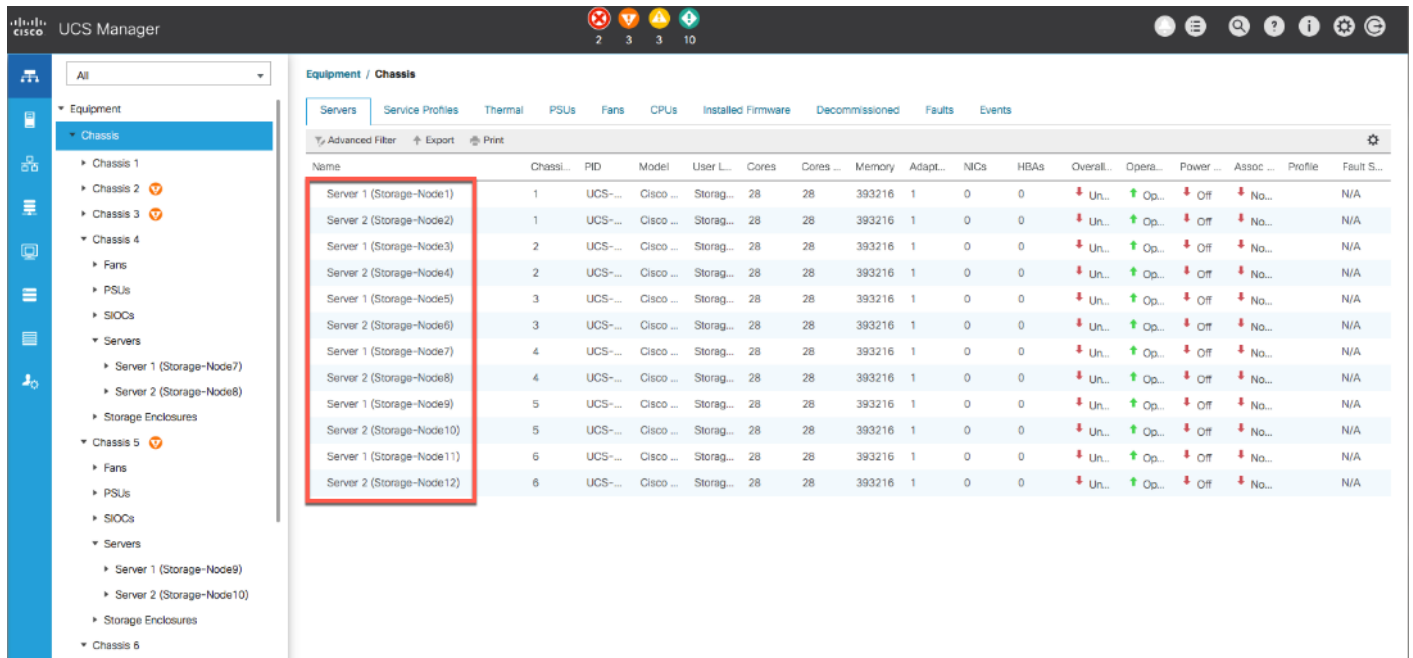
1. Select the **Equipment** tab.
2. Select Chassis > Chassis 1 > Server 1.
3. In the **Properties** section on the right go to **User Label** and add **Storage-Node1** to the field.
4. Repeat steps f1-3 or **Server 2** of **Chassis 1** and for all other servers of Chassis 2 – 6 according to Table 3.
5. Go then to **Servers > Rack-Mounts > Servers >** and repeat the step for all servers according to Table 4

Table 4 Server Label

Server	Name
Chassis 1 / Server 1	Storage-Node1
Chassis 1 / Server 2	Storage-Node2
Chassis 2 / Server 1	Storage-Node3

Server	Name
Chassis 2 / Server 2	Storage-Node4
Chassis 3 / Server 1	Storage-Node5
Chassis 3 / Server 2	Storage-Node6
Chassis 4 / Server 1	Storage-Node7
Chassis 4 / Server 2	Storage-Node8
Chassis 5 / Server 1	Storage-Node9
Chassis 5 / Server 2	Storage-Node10
Chassis 6 / Server 1	Storage-Node11
Chassis 6 / Server 2	Storage-Node12

Figure 21 Cisco UCS Server Labels



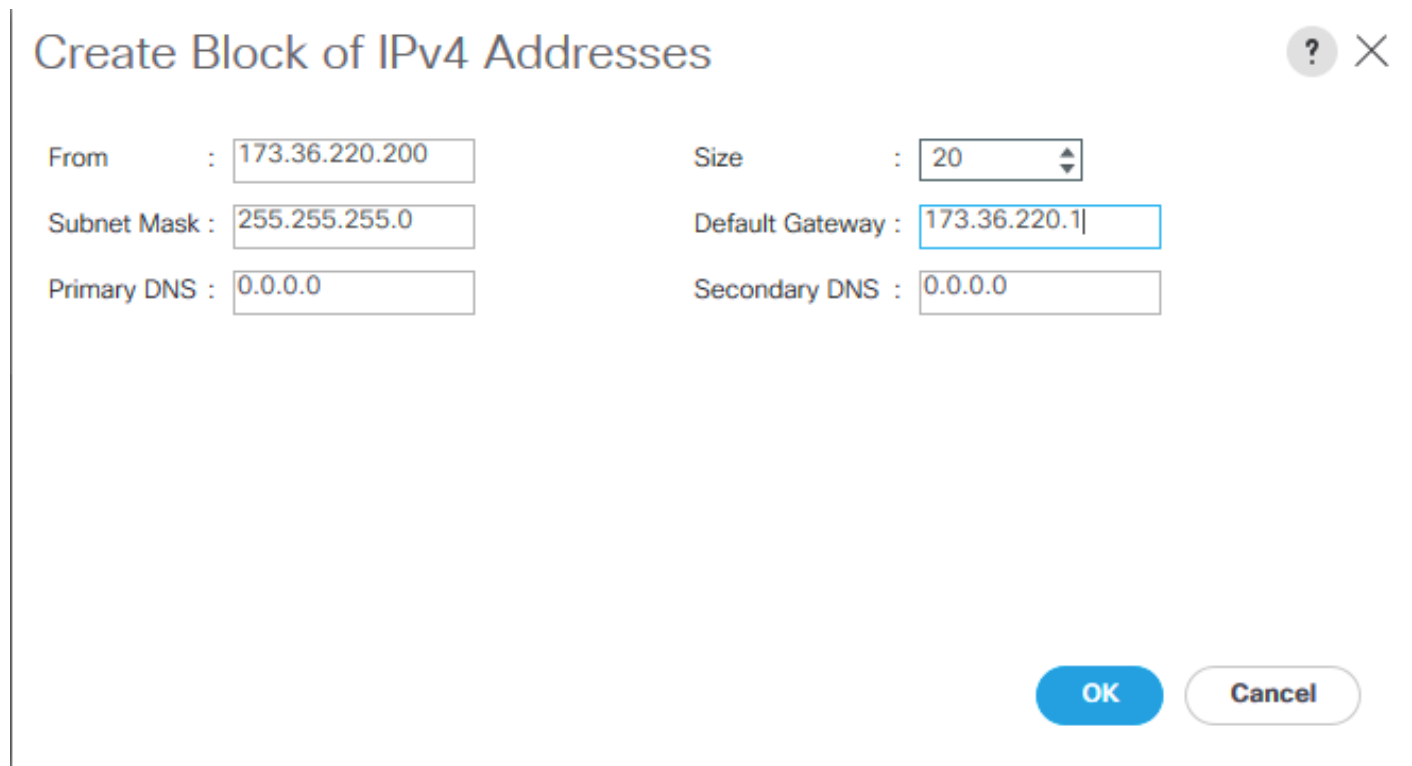
Create KVM IP Pool

To create a KVM IP Pool, follow these steps:

1. Select the **LAN** tab.
2. Go to LAN > Pools > root > IP Pools > IP Pool ext-mgmt.
3. Click Create Block of IPv4 Addresses.
4. Enter an IP Address in the **From** field.
5. Enter **Size 50**.

6. Enter your Subnet Mask.
7. Fill in your Default Gateway.
8. Enter your **Primary DNS** and **Secondary DNS** if needed.
9. Click OK.

Figure 22 Create Block of IPv4 Addresses



Create Block of IPv4 Addresses

From : 173.36.220.200 Size : 20

Subnet Mask : 255.255.255.0 Default Gateway : 173.36.220.1

Primary DNS : 0.0.0.0 Secondary DNS : 0.0.0.0

OK Cancel

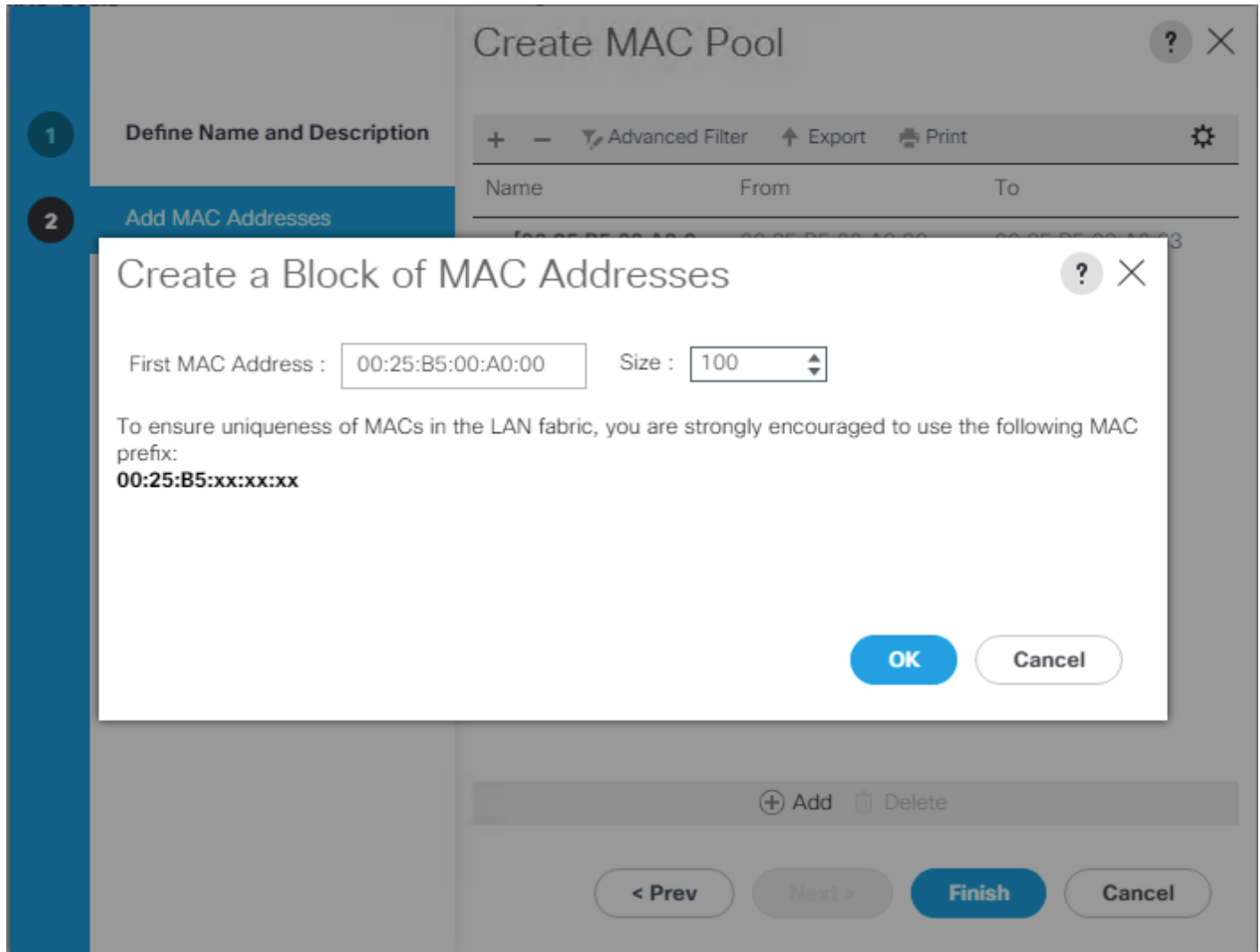
MAC Pool

To create a MAC Pool, follow these steps:

1. Select the **LAN** tab.
2. Go to LAN > Pools > root > Mac Pools and right-click Create MAC Pool.
3. Type in "Cloudian-MAC-Pools" for Name.
4. (Optional) Enter a **Description** of the MAC Pool.
5. Set Assignment Order as Sequential.
6. Click **Next**.
7. Click **Add**.

8. Specify a starting MAC address.
9. Specify a size of the MAC address pool, which is sufficient to support the available server resources, for example, 100.

Figure 23 Create a Block of MAC Addresses



10. Click **OK**.
11. Click **Finish**.

Create UUID Pool

To create a UUID Pool, follow these steps:

1. Select the **Servers** tab.
2. Go to Servers > Pools > root > UUID Suffix Pools and right-click Create UUID Suffix Pool.
3. Type in "Cloudian-UUID-Pools" for Name.

- (Optional) Enter a **Description** of the MAC Pool.
- Set Assignment Order to Sequential and click Next.
- Click **Add**.
- Specify a starting UUID Suffix.
- Specify a size of the UUID suffix pool, which is sufficient to support the available server resources, for example, 50.

Figure 24 Create a Block of UUID Suffixes

The screenshot shows the 'Create UUID Suffix Pool' interface. A modal dialog titled 'Create a Block of UUID Suffixes' is open, allowing the user to specify the starting UUID suffix and the size of the pool. The dialog has a 'From' field with the value '0000-000000000001' and a 'Size' dropdown menu set to '50'. There are 'OK' and 'Cancel' buttons at the bottom of the dialog. The background interface shows a table with columns 'Name', 'From', and 'To', and a row with the value '[0000-00000000... 0000-000000000001 0000-000000000032'. There are also 'Add' and 'Delete' buttons at the bottom of the background interface.

- Click **OK**.
- Click **Finish** and then **OK**.

Create VLANs

It is important to separate the network traffic with VLANs for Storage-Management traffic and Storage-Cluster traffic, External traffic and Client traffic (Optional). Table 5 lists the configured VLANs.



Client traffic is optional. We used Client traffic, to validate the functionality of NFS and S3 connectors.

Table 5 VLAN Configurations

VLAN	Name	Function
100	Storage-Management	Storage Management traffic for Storage Nodes
120	Client-Network (optional)	Client traffic for Storage Nodes
130	Storage-Cluster	Storage Cluster traffic and Storage Nodes
220	External-Network	External Public Network for all UCS Servers

To configure VLANs in the Cisco UCS Manager GUI, follow these steps:

1. Select **LAN** in the UCSM GUI.
2. Select LAN > LAN Cloud > VLANs and right-click Create VLANs.
3. Enter "Storage-Mgmt" for the VLAN Name.
4. Keep Multicast Policy Name as <not set>.
5. Select **Common/Global** for Public.
6. Enter 100 in the **VLAN IDs** field.
7. Click **OK** and then click **Finish**.

Figure 25 Create a VLAN

Create VLANs

VLAN Name/Prefix :

Multicast Policy Name : [Create Multicast Policy](#)

Common/Global
 Fabric A
 Fabric B
 Both Fabrics Configured Differently

You are creating global VLANs that map to the same VLAN IDs in all available fabrics. Enter the range of VLAN IDs.(e.g. "2009-2019", "29,35,40-45", "23", "23,34-45")

VLAN IDs :

Sharing Type : None Primary Isolated Community

8. Repeat steps 1–7 for the VLANs “Storage-Cluster” “External-Network and Client-Network.”

Enable CDP

To enable Network Control Policies, follow these steps:

1. Select the **LAN** tab of the Cisco UCS Manager GUI.
2. Go to LAN > Policies > root > Network Control Policies and right-click Create Network-Control Policy.
3. Type in **Enable-CDP** in the **Name** field.
4. (Optional) Enter a description in the **Description** field.
5. Click **Enabled** under **CDP**.
6. Click All Hosts VLANs under MAC Register Mode.
7. Leave everything else untouched and click **OK**.

8. Click **OK**.

Figure 26 Create a Network Control Policy

Create Network Control Policy ? X

Name :

Description :

CDP : Disabled Enabled

MAC Register Mode : Only Native Vlan All Host Vlans

Action on Uplink Fail : Link Down Warning

MAC Security

Forge : Allow Deny

LLDP

OK **Cancel**

QoS System Class

To create a Quality of Service System Class, follow these steps:

1. Select the **LAN** tab of the Cisco UCS Manager GUI.
2. Go to LAN > LAN Cloud > QoS System Class.
3. Best Effort MTU as 9216.
4. Set Fibre Channel Weight to None.
5. Click **Save Changes** and then click **OK**.

Figure 27 QoS System Class

LAN / LAN Cloud / QoS System Class

General Events FSM

Actions Properties

Use Global Owner : Local

Priority	Enabled	CoS	Packet Drop	Weight	Weight (%)	MTU	Multicast Optimized
Platinum	<input type="checkbox"/>	5	<input type="checkbox"/>	10	N/A	normal	<input type="checkbox"/>
Gold	<input type="checkbox"/>	4	<input checked="" type="checkbox"/>	9	N/A	normal	<input type="checkbox"/>
Silver	<input type="checkbox"/>	2	<input checked="" type="checkbox"/>	8	N/A	normal	<input type="checkbox"/>
Bronze	<input type="checkbox"/>	1	<input checked="" type="checkbox"/>	7	N/A	normal	<input type="checkbox"/>
Best Effort	<input checked="" type="checkbox"/>	Any	<input checked="" type="checkbox"/>	5	50	9216	<input type="checkbox"/>
Fibre Channel	<input checked="" type="checkbox"/>	3	<input type="checkbox"/>	5	50	fc	N/A

vNIC Template Setup

Based on the previous section of creating VLANs, the next step is to create the appropriate vNIC templates. For Cloudian Storage you need to create four different vNICs, depending on the role of the server. Table 6 lists items for the configuration.

Table 6 vNIC Table

vNIC Name	Fabric	Failover	VLAN Name / ID	MTU Size	MAC Pool	Network Control Policy
Storage-Mgmt	A	Yes	Storage-Mgmt 100	9000	Cloudian-MAC-Pools	Enable-CDP
Storage-Cluster	B	Yes	Storage-Cluster 130	9000	Cloudian-MAC-Pools	Enable-CDP
External-Network	A	Yes	External-Network 120	1500	Cloudian-MAC-Pools	Enable-CDP
Client-Network	B	Yes	Client-Network 120	9000	Cloudian-MAC-Pools	Enable-CDP

To create the appropriate vNICs, follow these steps:

1. Select the **LAN** tab of the Cisco UCS Manager GUI.
2. Go to LAN > Policies > root > vNIC Templates and right-click Create vNIC Template.
3. Type in **Storage-Mgmt** in the **Name** field.

4. (Optional) Enter a description in the **Description** field.
5. Click Fabric A as Fabric ID and enable failover.
6. Template Type as **Updating Template**
7. Select **default** as **VLANs** and click **Native VLAN**.
8. Select **Cloudian-MAC-Pools** as MAC Pool.
9. Select Enable-CDP as Network Control Policy.
10. Click **OK** and then click **OK** again.

Figure 28 Setup of vNIC Template for Storage-Mgmt vNIC

Create vNIC Template ? X

Name :

Description :

Fabric ID : Fabric A Fabric B Enable Failover

Redundancy

Redundancy Type : No Redundancy Primary Template Secondary Template

Target

Adapter VM

Warning

If **VM** is selected, a port profile by the same name will be created.
 If a port profile of the same name exists, and updating template is selected, it will be overwritten

Template Type : Initial Template Updating Template

VLANs | VLAN Groups

Advanced Filter | Export | Print |

Select	Name	Native VLAN	VLAN ID
<input type="checkbox"/>	Client-Network	<input type="radio"/>	120
<input type="checkbox"/>	default	<input type="radio"/>	1
<input type="checkbox"/>	External-Network	<input type="radio"/>	220
<input type="checkbox"/>	Storage-Cluster	<input type="radio"/>	130
<input checked="" type="checkbox"/>	Storage-Mgmt	<input type="radio"/>	100

Create VLAN

CDN Source : vNIC Name User Defined

MTU : 9000

MAC Pool : Cloudian-MAC_pools(100/100) ▼

QoS Policy : <not set> ▼

Network Control Policy : Enable-CDP ▼

Pin Group : <not set> ▼

Stats Threshold Policy : default ▼

Connection Policies

Dynamic vNIC usNIC VMQ

Dynamic vNIC Connection Policy : <not set> ▼

OK **Cancel**

- Repeat steps 1-10 for the vNICs “Storage-Cluster” “External-Network” and “Client-Network”. Make sure you select the correct Fabric ID, VLAN and MTU size according to Table 5 .

Ethernet Adapter Policy Setup

By default, Cisco UCS provides a set of Ethernet adapter policies. These policies include the recommended settings for each supported server operating system. Operating systems are sensitive to the settings in these policies.

Cisco UCS best practice is to enable Jumbo Frames MTU 9000 for any Storage facing Networks (Storage-Mgmt and Storage-Cluster). Enabling jumbo frames on specific interfaces and modifying Tx and Rx values guarantees 39Gb/s bandwidth on the UCS fabric.

To create a specific adapter policy for Red Hat Enterprise Linux, follow these steps:

- Select the **Server** tab of the Cisco UCS Manager GUI.
- Go to Servers > Policies > root > Adapter Policies and right-click Create Ethernet Adapter Policy.
- Type in **RHEL** in the **Name** field.
- (Optional) Enter a description in the **Description** field.
- Under **Resources** type in the following values:
 - Transmit Queues: 8

- Ring Size: 4096
 - Receive Queues: 8
 - Ring Size: 4096
 - Completion Queues: 16
 - Interrupts: 32
6. Under Options enable Receive Side Scaling (RSS).
 7. Click **OK** and then click **OK** again.

Figure 29 Adapter Policy for RHEL

Create Ethernet Adapter Policy ? X

Name :

Description :

Resources

Pooled : Disabled Enabled

Transmit Queues	<input type="text" value="8"/>	[1-1000]
Ring Size	<input type="text" value="4096"/>	[64-4096]
Receive Queues	<input type="text" value="8"/>	[1-1000]
Ring Size	<input type="text" value="4096"/>	[64-4096]
Completion Queue	<input type="text" value="16"/>	[1-2000]
Interrupts	<input type="text" value="32"/>	[1-1024]

Options

Transmit Checksum Offload : Disabled Enabled

Receive Checksum Offload : Disabled Enabled

TCP Segmentation Offload : Disabled Enabled

TCP Large Receive Offload : Disabled Enabled

Receive Side Scaling (RSS) : Disabled Enabled

Accelerated Receive Flow Steering : Disabled Enabled

Network Virtualization using Generic Routing Encapsulation : Disabled Enabled

Boot Policy Setup

To create a Boot Policy, follow these steps:

1. Select the **Servers** tab.
2. Go to Servers > Policies > root > Boot Policies and right-click Create Boot Policy.

3. Type in a **Local-OS-Boot** in the **Name** field.
4. (Optional) Enter a description in the **Description** field.

Figure 30 Create Boot Policy

Create Boot Policy ? X

Name :

Description :

Reboot on Boot Order Change :

Enforce vNIC/vHBA/iSCSI Name :

Boot Mode : Legacy Uefi

WARNINGS:
 The type (primary/secondary) does not indicate a boot order presence.
 The effective order of boot devices within the same device class (LAN/Storage/iSCSI) is determined by PCIe bus scan order.
 If **Enforce vNIC/vHBA/iSCSI Name** is selected and the vNIC/vHBA/iSCSI does not exist, a config error will be reported.
 If it is not selected, the vNICs/vHBAs are selected if they exist, otherwise the vNIC/vHBA with the lowest PCIe bus scan order is used.

Local Devices

Add Local Disk

- Add Local LUN
- Add Local JBOD
- Add SD Card
- Add Internal USB
- Add External USB
- Add Embedded Local LUN
- Add Embedded Local Disk

Add CD/DVD

- Add Local CD/DVD
- Add Remote CD/DVD

Add Floppy

- Add Local Floppy
- Add Remote Floppy

Add Remote Virtual Drive

Add NVMe

Boot Order

+ - Advanced Filter Export Print ⚙

Name	Order ▲	vNIC/v...	Type	LUN N...	WWN	Slot Nu...	Boot N...	Boot P...	Descrip...
Local LUN	1								
CD/DVD	2								

↑ Move Up ↓ Move Down 🗑 Delete

Set UEFI Boot Parameters

OK
Cancel

5. Click Add CD/DVD and click OK.
6. Click Local Disk > Add Local LUN and Set Type as “Any” and click OK.
7. Click **OK**.

Create LAN Connectivity Policy Setup

To create a LAN Connectivity Policy, follow these steps:

1. Select the **LAN** tab.

2. Go to LAN > Policies > root > LAN Connectivity Policies and right-click Create LAN Connectivity Policy for Storage Servers.
3. Type in **Storage-Node** in the **Name** field.
4. (Optional) Enter a description in the **Description** field.
5. Click **Add**.
6. Type in Storage-Mgmt in the name field.
7. Click “Use vNIC Template.”
8. Select vNIC template for “Storage-Mgmt” from drop-down list.
9. If you are using Jumbo Frame MTU 9000, select the default Adapter Policy, previously created as “RHEL” from the drop-down list.

Figure 31 LAN Connectivity Policy

Create vNIC

Name :

Use vNIC Template

Redundancy Pair :

Peer Name :

vNIC Template

[Create vNIC Template](#)

Adapter Performance Profile

Adapter Policy

[Create Ethernet Adapter Policy](#)

10. Repeat steps 1–9 for the remaining networks “Storage-Cluster”, “External-Network”, and “Client-Network” Make sure you choose Adapter Policy as “RHEL” for vNIC interface “Storage-Cluster.”

Create Maintenance Policy Setup

To setup a Maintenance Policy, follow these steps:

1. Select the **Servers** tab.
2. Go to Servers > Policies > root > Maintenance Policies and right-click Create Maintenance Policy.
3. Type in a **Server-Maint** in the Name field.
4. (Optional) Enter a description in the **Description** field.
5. Click User Ack under Reboot Policy.

6. Click **OK** and then click **OK** again.
7. Create Maintenance Policy.

Create Maintenance Policy ? X

Name :

Description :

Soft Shutdown Timer : ▼

Storage Config. Deployment Policy : Immediate User Ack

Reboot Policy : Immediate User Ack Timer Automatic

On Next Boot (Apply pending changes at next reboot.)

Create Chassis Profile

The Chassis Profile is required to assign specific disks to a particular server node in a Cisco UCS S3260 Storage Server as well as upgrading to a specific chassis firmware package.

Create Chassis Firmware Package

To create a Chassis Firmware Package, follow these steps:

1. Select the **Chassis** tab of the Cisco UCS Manager GUI.
2. Go to Chassis > Policies > root > Chassis Firmware Package and right-click Create Chassis Firmware Package.
3. Type in **S3260-FW** in the Name field.
4. (Optional) Enter a description in the **Description** field.
5. Select **4.0 (2b) C** from the drop-down list of **Chassis Package**.
6. Select **OK** and then click **OK** again.

7. Create Chassis Firmware Package.

Create Chassis Firmware Package ? X

Name :

Description :

Chassis Package :

Service Pack :

The images from Service Pack will take precedence over the images from Chassis Package

Excluded Components:

Chassis Adaptor

Chassis Board Controller

Chassis Management Controller

Local Disk

SAS Expander

Create Chassis Maintenance Policy

To create a Chassis Maintenance Policy, follow these steps:

1. Select the **Chassis** tab of the Cisco UCS Manager GUI.
2. Go to Chassis > Policies > root > Chassis Maintenance Policies and right-click Create Chassis Maintenance Policy.
3. Type in **S3260-Main** in the **Name** field.

- (Optional) Enter a description in the **Description** field.
- Click **OK** and then click **OK** again.
- Create Chassis Maintenance Policy.

Create Chassis Maintenance Policy ? ×

Name :

Description :

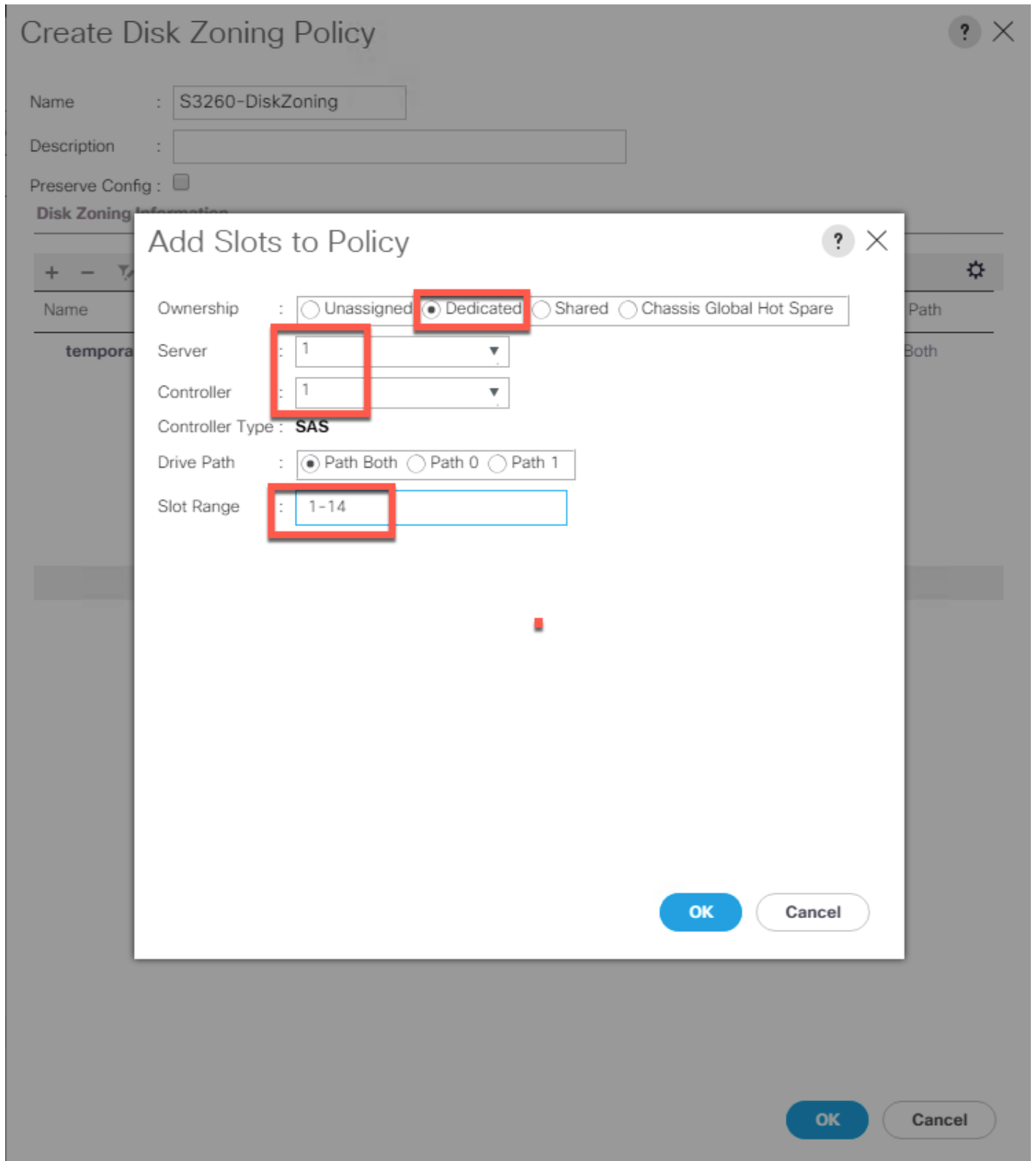
Reboot Policy : **User Ack**

Create Disk Zoning Policy

To create a Disk Zoning Policy, follow these steps:

- Select the **Chassis** tab of the Cisco UCS Manager GUI.
- Go to Chassis > Policies > root > Disk Zoning Policies and right-click Create Disk Zoning Policy.
- Type in **S3260-DiskZoning** in the Name field.
- (Optional) Enter a description in the **Description** field.
- Create Disk Zoning Policy.
- Click **Add**.
- Select Dedicated under Ownership.
- Select **Server 1** and Select **Controller 1**.

9. Add **Slot Range 1-14** for the top node of the Cisco UCS S3260 Storage Server and click **OK**.



10. Select **Server** 1 and Select **Controller** 2.

11. Add **Slot Range 15-28** for the top node of the Cisco UCS S3260 Storage Server and click **OK**.

12. Add Slots to Top Node of Cisco UCS S3260.

Add Slots to Policy ? X

Ownership : Unassigned **Dedicated** Shared Chassis Global Hot Spare

Server : 1

Controller : 2

Controller Type : **SAS**

Drive Path : Path Both Path 0 Path 1

Slot Range : 15-28

OK Cancel

13. Click **Add**.

14. Select **Dedicated** under Ownership.

15. Select **Server 2** and Select Controller 1.

16. Add **Slot Range 29-42** for the bottom node of the Cisco UCS S3260 Storage Server and click **OK**.

Add Slots to Policy ? X

Ownership : Unassigned **Dedicated** Shared Chassis Global Hot Spare

Server : ▼

Controller : ▼

Controller Type : **SAS**

Drive Path : Path Both Path 0 Path 1

Slot Range :

17. Select **Server 2** and Select Controller 2.

18. Add **Slot Range 43-56** for the bottom node of the Cisco UCS S3260 Storage Server and click OK.

Add Slots to Policy



Ownership : Unassigned **Dedicated** Shared Chassis Global Hot Spare

Server :

Controller :

Controller Type : **SAS**

Drive Path : Path Both Path 0 Path 1

Slot Range :

OK

Cancel

19. Add Slots to the Bottom Node of Cisco UCS S3260.

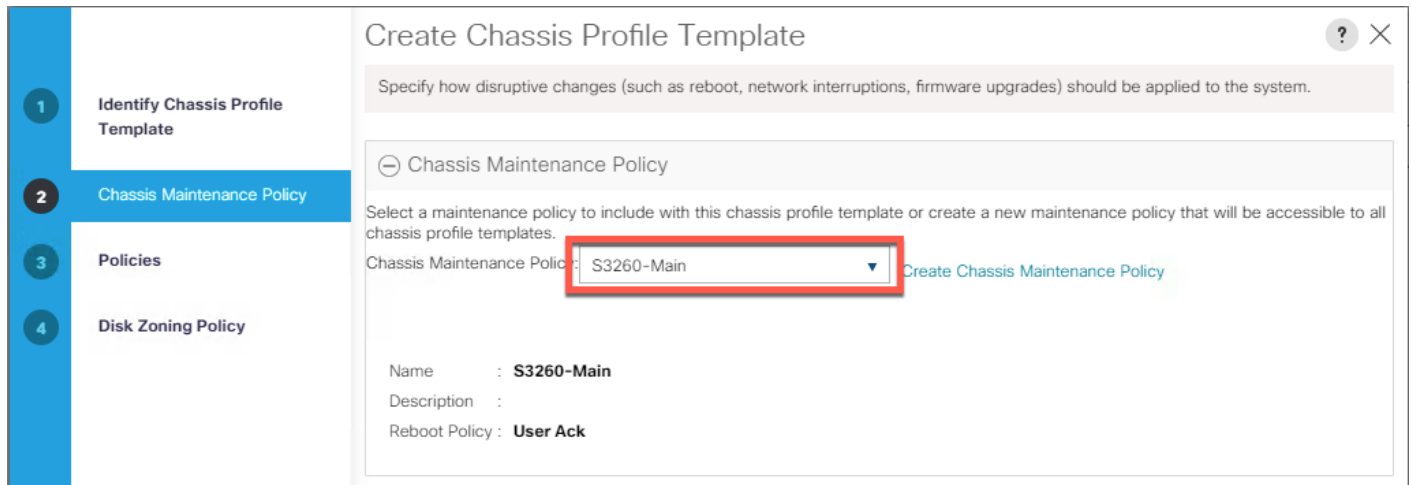
Create Chassis Profile Template

To create a Chassis Profile Template, follow these steps:

1. Select the **Chassis** tab of the Cisco UCS Manager GUI.
2. Go to Chassis > Chassis Profile Templates and right-click Create Chassis Profile Template.
3. Type in S3260-Chassis in the Name field.

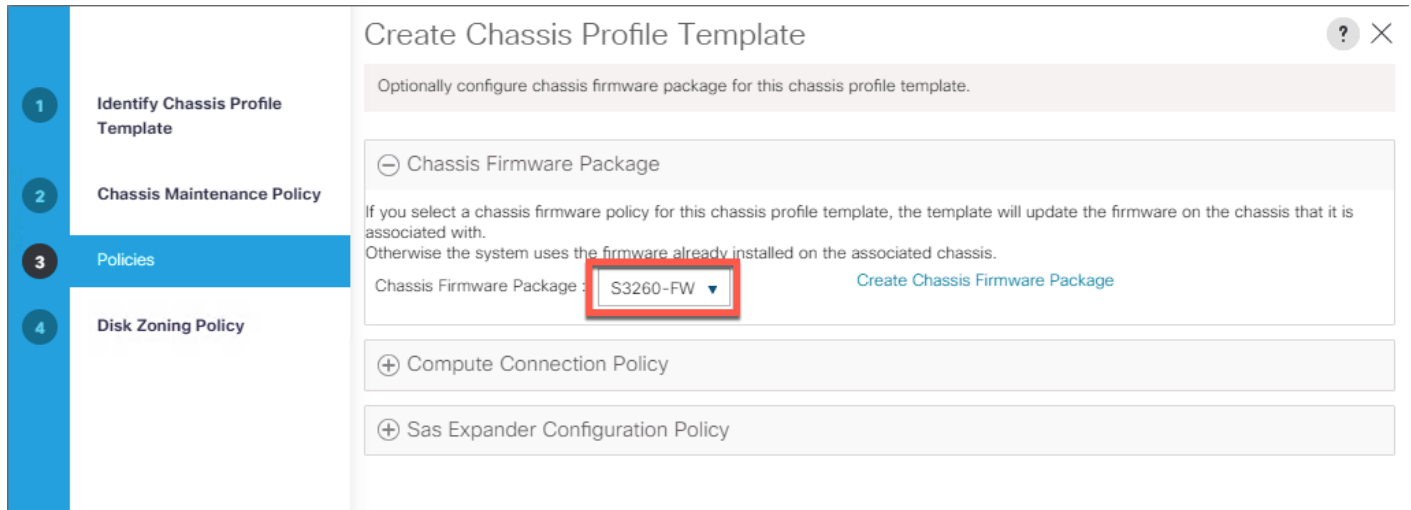
4. Under Type, select Updating Template.
5. (Optional) Enter a description in the **Description** field.
6. Create Chassis Profile Template.
7. Select **Next**.
8. Under the radio button **Chassis Maintenance Policy**, select your previously created Chassis Maintenance Policy.

Figure 32 Chassis Profile Template - Chassis Maintenance Policy



9. Select **Next**.
10. Select the + button and select under **Chassis Firmware Package** your previously created Chassis Firmware Package Policy.

Figure 33 Chassis Profile Template - Chassis Firmware Package



11. Select Next.

12. Under **Disk Zoning Policy** select your previously created Disk Zoning Policy.

Figure 34 Chassis Profile Template - Disk Zoning Policy

1 Identify Chassis Profile Template

2 Chassis Maintenance Policy

3 Policies

4 Disk Zoning Policy

Create Chassis Profile Template

Optionally specify information that affects how the system operates.
Disk Zoning policies are applicable only to UCSC-C3X60-BASE chassis

Disk Zoning Policy: **S3260-DiskZoning** ▼

[Create Disk Zoning Policy](#)

Name : **S3260-DiskZoning**

Description :

Preserve Config : **No**

Disks Zoned

Name	Slot Number	Ownership	Assigned to S...	Assigned to ...	Controller Type	Drive Path
▶ disk-slot-1	1	Dedicated				Path Both
▶ disk-slot-10	10	Dedicated				Path Both
▶ disk-slot-11	11	Dedicated				Path Both
▶ disk-slot-12	12	Dedicated				Path Both
▶ disk-slot-13	13	Dedicated				Path Both
▶ disk-slot-14	14	Dedicated				Path Both

< Prev Next > **Finish** Cancel

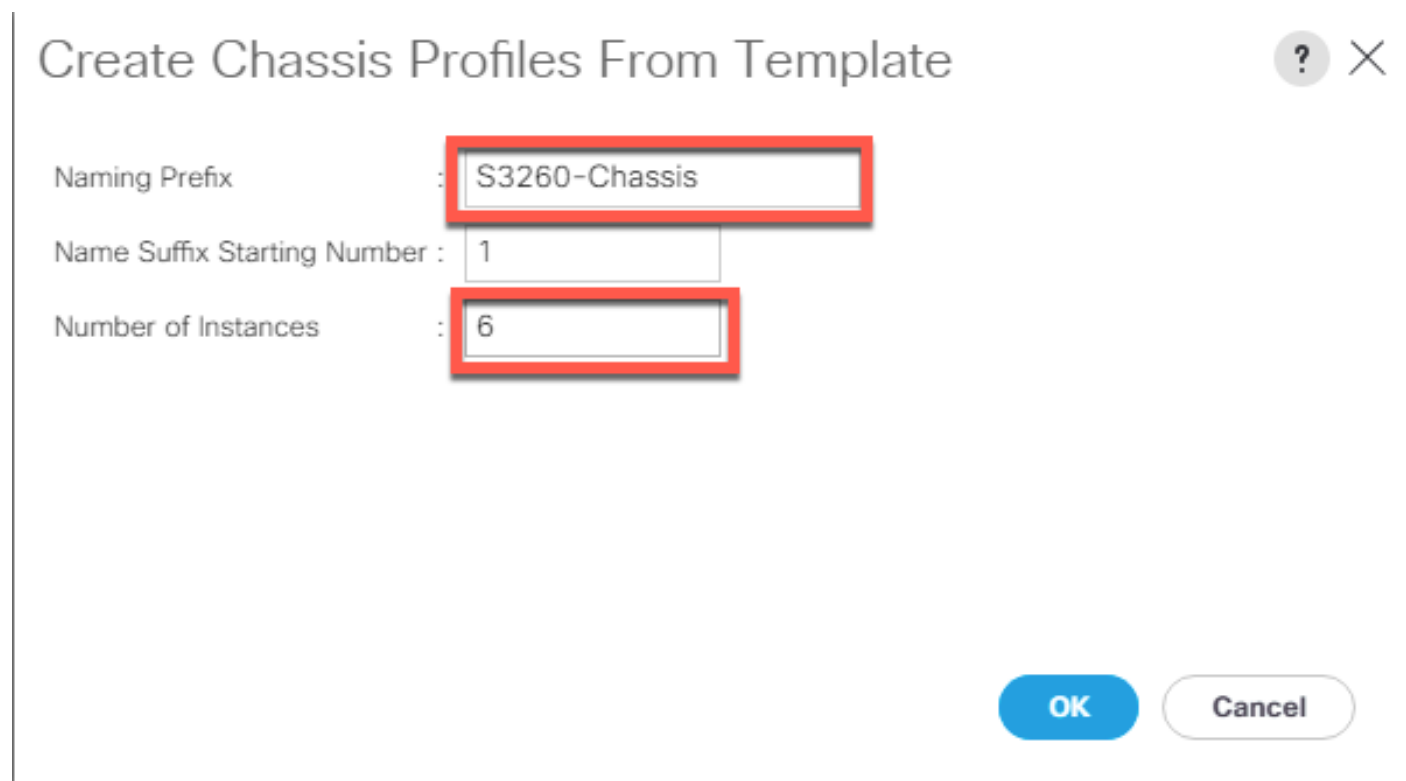
13. Click **Finish** and then click **OK** again.

Create Chassis Profile from Template

To create the Chassis Profiles from the previous created Chassis Profile Template, follow these steps:

1. Select the **Chassis** tab of the Cisco UCS Manager GUI.
2. Go to Chassis > Chassis Profile Templates and select "S3260-Chassis" you created previously.
3. Then right click to select "Create Chassis Profiles from Template."
4. Type in **S3260-Chassis** in the **Name** field.
5. Leave the Name Suffix Starting Number untouched.

6. Enter **6** for the **Number of Instances** for all connected Cisco UCS S3260 Storage Server.
7. Click **OK**.



Create Chassis Profiles From Template ? X

Naming Prefix : S3260-Chassis

Name Suffix Starting Number : 1

Number of Instances : 6

OK Cancel

Associate Chassis Profile

To associate all previous created Chassis Profile, follow these steps:

1. Select the **Chassis** tab of the Cisco UCS Manager GUI.
2. Go to Chassis > Chassis Profiles and select "S3260-Chassis1."
3. Right-click Change Chassis Profile Association.
4. Under Chassis Assignment, choose Select existing Chassis from the drop-down list.
5. Under **Available Chassis**, select ID **1**.
6. Click **OK** and then click **OK** again.
7. Repeat steps 1-6 for the other two Chassis Profiles by selecting the IDs 2 - 6.
8. A pop-up will appear on the top right side. Click Chassis Profiles and Acknowledge All Chassis profiles.
9. Click Apply.
10. Click OK.



After the association of the Chassis profile with the Disk zoning policy, the disks distribution between the nodes may get corrected in few minutes.

Create Storage Profiles

Set Disks for Cisco UCS S3260 M5 Servers to Unconfigured-Good

To prepare the OS drives reserved from the Cisco UCS S3260 M5 servers for storage profiles, make sure the disks have to be converted from “JBOD” to “Unconfigured-Good”. To convert the disks, follow these steps:

1. Select the **Equipment** tab of the Cisco UCS Manager GUI.
2. For S3260 M5 servers, Go to Equipment > Chassis > Chassis1 > Servers > Server1 > Inventory > Storage > Disks
3. Select both disks from slot “201 and 202” and right-click “**Set JBOD to Unconfigured-Good**”.

Disk 201	227928	18201C8F02D8	Operable	Unconfigured Good	Equipped	SSD	False
Disk 202	227928	18201C8EFB3A	Operable	Unconfigured Good	Equipped	SSD	False

4. Repeat steps 1-3 for the other Cisco UCS S3260 M5 Servers.

Create Storage Profiles for Cisco UCS S3260 Storage Server

To create the Storage Profile for the top node of the Cisco UCS S3260 Storage Server, follow these steps:

1. Select **Storage** of the Cisco UCS Manager GUI.
2. Go to Storage > Storage Profiles and right-click Create Storage Profile.
3. Type in **Sever1** in the **Name** field.
4. (Optional) Enter a description in the **Description** field.
5. Click **Add**.
6. Type in “**OS-Boot**” in the **Name** field.
7. Configure as follows:
 - Create Local LUN
 - Size (GB) = 1
 - Fractional Size (MB) = 0
 - Auto Deploy
 - Select Expand To Available

Create Local LUN



Create Local LUN
 Prepare Claim Local LUN

Name:

Size (GB): [0-245760]

Fractional Size (MB):

Auto Deploy: Auto Deploy No Auto Deploy

Expand To Available:

Select Disk Group Configuration: [Create Disk Group Policy](#)

OK

Cancel

8. Click "Create Disk Group Policy" to Create RAID1 LUN.
9. Type in **Server1** in the Name field.
10. (Optional) Enter a description in the **Description** field.
11. RAID Level = RAID 1 Mirrored.
12. Select Disk Group Configuration (Manual).
13. Click **Add**.
14. Type in **201** for **Slot Number**.
15. Click **OK** and then again **Add**.
16. Type in **202** for **Slot Number**.
17. Click **OK** and then click **OK** again.

Figure 35 Create Disk Group Policy

Create Disk Group Policy

Name : Server1

Description :

RAID Level : RAID 1 Mirrored

Disk Group Configuration (Automatic) Disk Group Configuration (Manual)

Disk Group Configuration (Manual)

Advanced Filter Export Print

Slot Number	Role	Span ID
201	Normal	Unspecified
202	Normal	Unspecified

+ Add - Delete Info

Virtual Drive Configuration

Strip Size (KB) : Platform Default

Access Policy : Platform Default Read Write Read Only Blocked

OK Cancel

18. Select your previously created Disk Group Policy for the Boot with the radio button under **Select Disk Group Configuration**.

19. Select Disk Group Configuration.

Create Local LUN ? X

Create Local LUN Prepare Claim Local LUN

Name :

Size (GB) : [0-245760]

Fractional Size (MB) :

Auto Deploy : Auto Deploy No Auto Deploy

Expand To Available :

Select Disk Group Configuration : Server1 ▼
Create Disk Group Policy

20. Click **OK**, click **OK** again, and then click **OK**.

21. For the storage profile for the second server, repeat steps 1-20 to create a storage profile for the second server in the chassis. The Storage Profile Server1 is for the top server and Server2 for the bottom Server of S3260

Create Storage Profile for Cisco UCS C220 M5 Rack-Mount Servers

To create a Storage Profile for the Cisco UCS C220 M5, follow these steps:

1. Select **Storage** of the UCSM GUI.
2. Go to Storage > Storage Profiles and right-click Create Storage Profile.
3. Type in **C220-OS-RAID1** in the **Name** field.
4. (Optional) Enter a description in the **Description** field.
5. Click **Add**.

Figure 36 Create Storage Profile for Cisco UCS C220 M5

Create Storage Profile

Name : C220-OS-Raid1

Description : OS Boot LUN on RAID1 for C220M5 Server

LUNs

Local LUNs | Controller Definitions | Security Policy

Advanced Filter | Export | Print

Name	Size (GB)	Order	Fractional Size (MB)
No data available			

+ Add | Delete | Info

OK Cancel

6. Type in **Boot** in the **Name** field.

7. Configure as follows:

- Create Local LUN
- Size (GB) = 1
- Fractional Size (MB) = 0
- Select Expand To Available
- Auto Deploy

Figure 37 Create Local LUN

Create Local LUN ? ✕

Create Local LUN Prepare Claim Local LUN

Name : OS-Boot

Size (GB) : 1 [0-245760]

Fractional Size (MB) : 0

Auto Deploy : Auto Deploy No Auto Deploy

Expand To Available :

Select Disk Group Configuration : <not set> Create Disk Group Policy

OK Cancel

8. Click Create Disk Group Policy to Create RAID1 LUN.
9. Type in **RAID1-C220** in the **Name** field.
10. (Optional) Enter a description in the **Description** field.
11. RAID Level = RAID 1 Mirrored.
12. Select Disk Group Configuration (Manual).
13. Click **Add**.
14. Type in **1** for **Slot Number**.
15. Click **OK** and then again **Add**.
16. Type in **2** for **Slot Number**.
17. Under “Change Virtual Drive Configuration:”
 - a. Modify Access Policy as “Read Write” and Read Policy as “Read Ahead”.
 - b. Modify Write Cache Policy as “Write Back Good BBU” and IO Policy as “Cache.”
18. Click **OK** and then click **OK** again.

Figure 38 Create Disk Group Policy for Cisco UCS C220 M5

Create Disk Group Policy ? X

Name :

Description :

RAID Level :

Disk Group Configuration (Automatic) Disk Group Configuration (Manual)

Disk Group Configuration (Manual)

Advanced Filter Export Print ⚙️

Slot Number	Role	Span ID
1	Normal	Unspecified
2	Normal	Unspecified

➕ Add 🗑️ Delete ℹ️ Info

Virtual Drive Configuration

Strip Size (KB) :

Access Policy : Platform Default Read Write Read Only Blocked

Read Policy : Platform Default Read Ahead Normal

Write Cache Policy : Platform Default Write Through Write Back Good Bbu Always Write Back

IO Policy : Platform Default Direct Cached

Drive Cache : Platform Default No Change Enable Disable

Security :

19. Select the previously created Disk Group Policy for the C220 M5 Boot Disks with the radio button under **Select Disk Group Configuration**.

Figure 39 Create Disk Group Configuration for Cisco UCS C220 M5

Create Local LUN

Create Local LUN
 Prepare Claim Local LUN

Name : OS-Boot

Size (GB) : 1 [0-245760]

Fractional Size (MB) : 0

Auto Deploy : Auto Deploy No Auto Deploy

Expand To Available :

Select Disk Group Configuration : RAID1-C220 [Create Disk Group Policy](#)

20. Click **OK** and then **OK** and again click **OK**.

Create a Service Profile Template for Cisco UCS S3260 Storage Server

Create Service Profile Template for Cisco UCS S3260 Storage Server1 and Server2

To create a Service Profile Template, follow these steps:

1. Select **Servers** of the Cisco UCS Manager GUI.
2. Go to Servers > Service Profile Templates > root and right-click Create Service Profile Template.

Identify Service Profile Template

To identify the Service Profile template, follow these steps:

1. Type in "Storage-Server1-Template" in the Name field.
2. Select Template Type "**Updating Template**"
3. In the **UUID Assignment** section, select the UUID Pool you created in the beginning.
4. (Optional) Enter a description in the **Description** field.

Figure 40 Identify Service Profile Template

Create Service Profile Template

You must enter a name for the service profile template and specify the template type. You can also specify how a UUID will be assigned to this template and enter a description.

Name :

The template will be created in the following organization. Its name must be unique within this organization.
Where : **org-root**

The template will be created in the following organization. Its name must be unique within this organization.
Type : Initial Template Updating Template

Specify how the UUID will be assigned to the server associated with the service generated by this template.
UUID

UUID Assignment:

The UUID will be assigned from the selected pool.
The available/total UUIDs are displayed after the pool name.

Optionally enter a description for the profile. The description can contain information about when and where the service profile should be used.

< Prev Next > **Finish** Cancel

5. Click **Next**.

Storage Provisioning

To provision the storage profile, follow these steps:

1. Go to the **Storage Profile Policy** tab and select the Storage Profile **Server1** for the top node of the Cisco UCS S3260 Storage Server you created before.
2. Click **Next**.

Figure 41 Storage Provisioning

Optional specify or create a Storage Profile, and select a local disk configuration policy.

Specific Storage Profile | **Storage Profile Policy** | Local Disk Configuration Policy

Storage Profile: **Server1** Create Storage Profile

Name : **Server1**
Description :

LUNs

Local LUNs | LUN Set | Controller Definitions | Security Policy

Advanced Filter | Export | Print

Name	Size (GB)	Order	Fractional Size (MB)
OS-Boot	1	Not Applicable	0

< Prev | Next > | **Finish** | Cancel

Networking

To configure networking, follow these steps:

1. Keep the Dynamic vNIC Connection Policy field at the default.
2. Select LAN connectivity to Use Connectivity Policy created before.
3. From LAN Connectivity drop-down list, select "Storage-Node" created before and click Next.

Figure 42 Summary Networking

1 Identify Service Profile Template

2 Storage Provisioning

3 **Networking**

4 SAN Connectivity

5 Zoning

6 vNIC/vHBA Placement

7 vMedia Policy

8 Server Boot Order

9 Maintenance Policy

10 Server Assignment

11 Operational Policies

Create Service Profile Template

Optionally specify LAN configuration information.

Dynamic vNIC Connection Policy: Select a Policy to use (no Dynamic vNIC Policy by default) ▼

Create Dynamic vNIC Connection Policy

How would you like to configure LAN connectivity?

Simple Expert No vNICs Use Connectivity Policy

LAN Connectivity Policy: **Storage-Node** ▼ Create LAN Connectivity Policy

Initiator Name

Initiator Name Assignment: <not set> ▼

Create IQN Suffix Pool

WARNING: The selected pool does not contain any available entities. You can select it, but it is recommended that you add entities to it.

< Prev Next > **Finish** Cancel

4. Click **Next** to continue with SAN Connectivity.
5. Select No vHBA for How would you like to configure SAN Connectivity?
6. Click **Next** to continue with Zoning.
7. Click **Next**.

vNIC/vHBA Placement

To configure the vNIC/vHBA placement, follow these steps:

1. Select **Specify Manually** from the drop-down list.
2. Under PCI order section, Sort all the vNICs.
3. Make sure the vNICs order are listed as External-Network > 1, then followed by Storage-Mgmt > 2 and Storage-Cluster > 3 Client-Network > 4.

Create Service Profile Template

Specify how vNICs and vHBAs are placed on physical network adapters

vNIC/vHBA Placement specifies how vNICs and vHBAs are placed on physical network adapters (mezzanine) in a server hardware configuration independent way.

Select Placement: **Specify Manually** [Create Placement Policy](#)

vNICs vHBAs

Name

No data available

>> assign >>
<< remove <<

Specific Virtual Network Interfaces (click on a cell to edit)

Name	O...	Adm...	Sele...	Tran...
vCon 1				
vNIC External-Network	1	ANY	All	ethe...
vNIC Storage-Mgmt	2	ANY		
vNIC Storage-Cluster	3	ANY		
vNIC Client-Network	4	ANY		
vCon 2				
			All	ethe...

↑ Move Up ↓ Move Down

< Prev **Next** > **Finish** Cancel

4. Click **Next** to continue with vMedia Policy.

5. Click **Next**.

Server Boot Order

To configure the server boot order, follow these steps:

1. Select the Boot Policy "Local-OS-Boot" you created before under Boot Policy.
2. Server Boot Order.
3. Click Next.

Create Service Profile Template

Optionally specify the boot policy for this service profile template.

Select a boot policy.

Boot Policy: **Local-OS-Boot** Create Boot Policy

Name : **Local-OS-Boot**
 Description : **OS boot policy for storage nodes**
 Reboot on Boot Order Change : **No**
 Enforce vNIC/vHBA/iSCSI Name : **Yes**
 Boot Mode : **Legacy**

WARNINGS:
 The type (primary/secondary) does not indicate a boot order presence.
 The effective order of boot devices within the same device class (LAN/Storage/iSCSI) is determined by PCIe bus scan order.
 If **Enforce vNIC/vHBA/iSCSI Name** is selected and the vNIC/vHBA/iSCSI does not exist, a config error will be reported.
 If it is not selected, the vNICs/vHBAs are selected if they exist, otherwise the vNIC/vHBA with the lowest PCIe bus scan order is used.

Boot Order

Name	Order	vNIC/vHB...	Type	LUN Name	WWN	Slot Num...	Boot Name	Boot Path	Description
Local L...	1								
CD/DVD	2								

Create ISCSI vNIC Set ISCSI Boot Parameters Set UEFI Boot Parameters

< Prev Next > Finish Cancel

Maintenance Policy

To configure the maintenance policy, follow these steps:

1. Select the Maintenance Policy you created before under Maintenance Policy.

Figure 43 Maintenance Policy

1 Identify Service Profile Template

2 Storage Provisioning

3 Networking

4 SAN Connectivity

5 Zoning

6 vNIC/vHBA Placement

7 vMedia Policy

8 Server Boot Order

9 Maintenance Policy

10 Server Assignment

11 Operational Policies

Create Service Profile Template

Specify how disruptive changes such as reboots, network interruptions, and firmware upgrades should be applied to the server associated with this service profile.

⊖ Maintenance Policy

Select a maintenance policy to include with this service profile or create a new maintenance policy that will be accessible to all service profiles.

Maintenance Policy: **Server-Maint** [Create Maintenance Policy](#)

Name : **Server-Maint**
 Description : **UCS Server Maintenance Policy**
 Soft Shutdown Timer : **150 Secs**
 Storage Config. Deployment Policy : **User Ack**
 Reboot Policy : **User Ack**

< Prev Next > **Finish** Cancel

2. Click **Next**.
3. Under Server Assignment, Leave everything else untouched.
4. Click **Next**.

Operational Policies

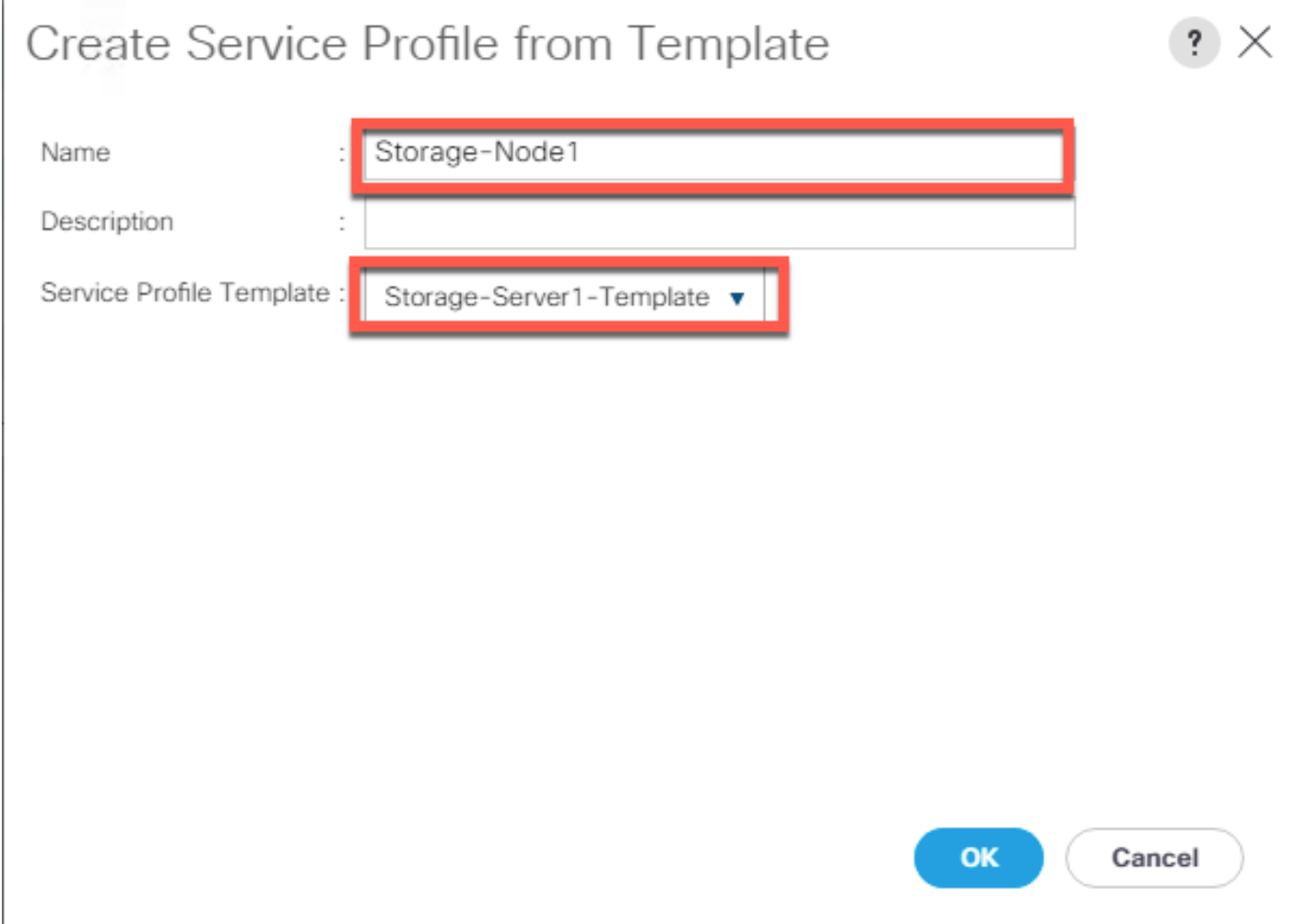
To configure the operational policies, follow these steps:

1. Click **Finish** and then click **OK**.
2. Repeat the steps for the Server2 of the Cisco UCS S3260 Storage Server by naming template as “Storage-Server2-Template.”
3. During Storage Provisioning tab, choose the Storage Profile for the Server2 “S3260-Server2-Node” you created previously.

Create Service Profiles from Template

This section details how to create the appropriate Service Profiles from the previous Service Profile Templates. To create the first profile for the Server1 of the Cisco UCS S3260 Storage Server, follow these steps:

1. Select **Servers** from the left pane of the Cisco UCS Manager GUI.
2. Go to Servers > Service Profiles and right-click Create Service Profile from Template.
3. Type in **Storage-Node1** in the Name Prefix field.
4. Choose "**Storage-Server1-Template**" as the **Service Profile Template** you created before for the top node of the Cisco UCS S3260 Storage Server.
5. Click **OK** and then click **OK** again.



Create Service Profile from Template

Name : Storage-Node1

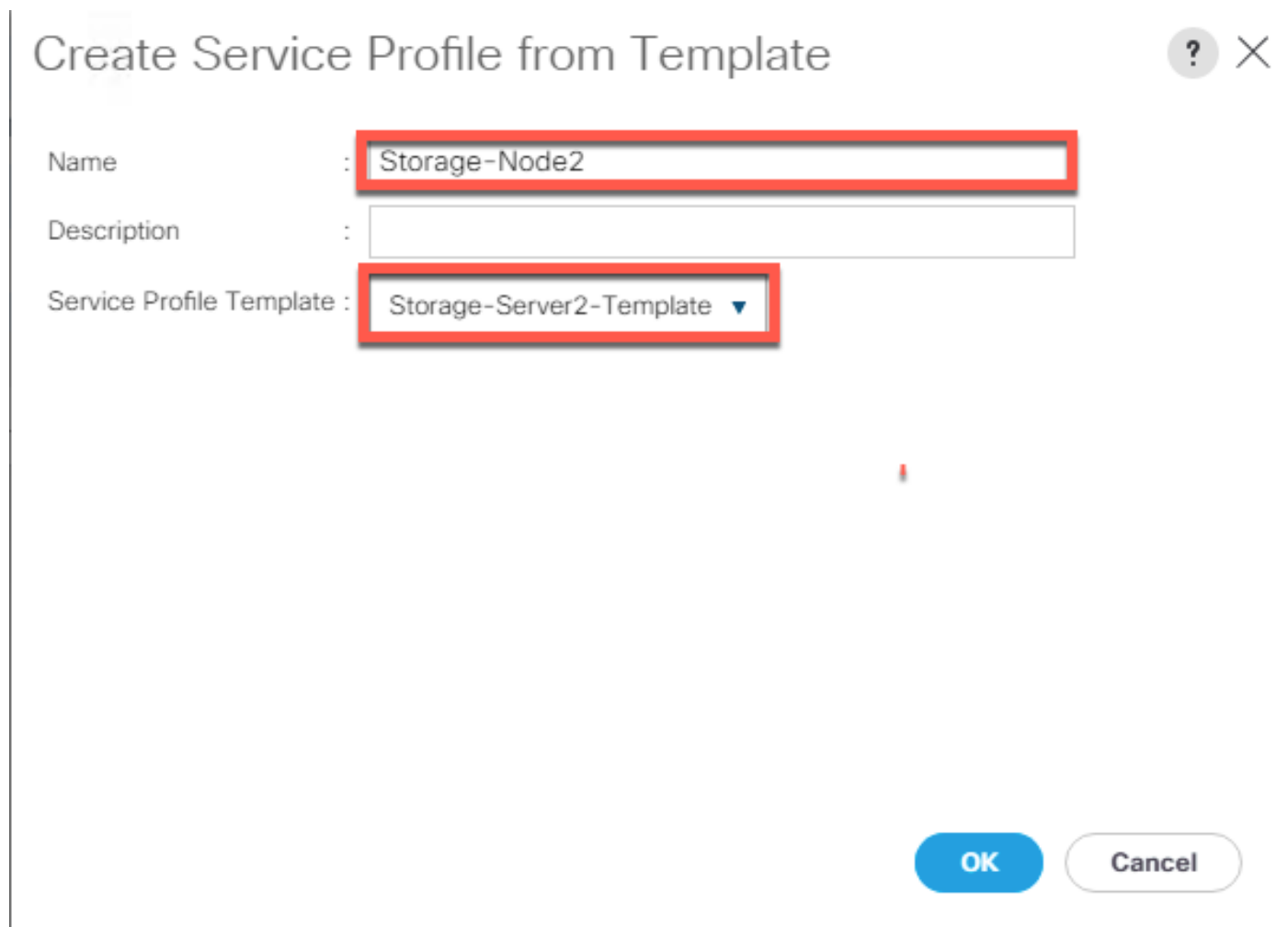
Description :

Service Profile Template : Storage-Server1-Template ▼

OK Cancel

6. Repeat steps 1-5 to create Service Profiles for the remaining S3260 M5 server1 Nodes from the Template that belongs to top Node "Storage-Server1-Template". Make sure you name it as "Storage-Node3, Storage-Node5, Storage-Node7,Storage-Node9,Storage-Node11" respectively.
7. For the remaining M5 nodes, again Navigate to Servers > Service Profiles and right-click Create Service Profile from Template.
8. Type in **Storage-Node2** in the Name Prefix field.

9. Choose "**Storage-Server2-Template**" as the **Service Profile Template** you created before for the top node of the Cisco UCS S3260 Storage Server.
10. Click **OK** and then click **OK** again.



Create Service Profile from Template

Name :

Description :

Service Profile Template :

OK **Cancel**

11. Repeat steps 1-10 to create Service Profiles for the remaining S3260 M5 server Bottom Nodes from the Template that belongs to bottom Node "Storage-Server2-Template". Make sure you name it as "Storage-Node4, Storage-Node6,Storage-Node8,Storage-Node10, Storage-Node12."

Associate a Service Profile for Cisco UCS S3260 M5 Server

To associate all the "Storage-NodeX" Service Profiles to the Cisco UCS S3260 M5 Storage Servers, follow these steps:

1. Select **Servers** from the left pane of the Cisco UCS Manager GUI.
2. Go to Servers > Service Profiles and right-click "Storage-Node1" Service profile created previously.
3. Click "Change Server Profile Association."
4. From the Server Assignment drop-down list choose "Select Existing Server."

5. Click the radio button “Available Servers.”
6. From the Chassis and Slot listed, choose Chassis1/Slot1 for Storage-Node1.
7. Click OK.

Associate Service Profile ? X

Select an existing server pool or a previously-discovered server by name, or manually specify a custom server by entering its chassis and slot ID. If no server currently exists at that location, the system waits until one is discovered.

You can select an existing server or server pool, or specify the physical location of the server you want to associate with this service profile.

Server Assignment:

Available Servers All Servers

Select	Chassis ID	Slot	Rack ID	PID	Procs	Memory	Adapters
<input checked="" type="radio"/>	1	1		UCS-S32...	2	393216	1
<input type="radio"/>	1	2		UCS-S32...	2	393216	1
<input type="radio"/>	2	1		UCS-S32...	2	393216	1
<input type="radio"/>	2	2		UCS-S32...	2	393216	1
<input type="radio"/>	3	1		UCS-S32...	2	393216	1
<input type="radio"/>	3	2		UCS-S32...	2	393216	1

Restrict Migration :

8. Repeat steps 1-7 to the Associate Remaining Service profiles “Storage-NodeX” for the Cisco UCS S3260 M5 storage server as listed in the table below.

Service Profile Template	Service Profile	S3260 Chassis	Server Slot ID
Storage-Server1-Template	Storage-Node1	1	1

Storage-Server2 -Template	Storage-Node2	1	2
Storage-Server1-Template	Storage-Node3	2	1
Storage-Server2-Template	Storage-Node4	2	2
Storage-Server1-Template	Storage-Node5	3	1
Storage-Server2-Template	Storage-Node6	3	2
Storage-Server1-Template	Storage-Node7	4	1
Storage-Server2-Template	Storage-Node8	4	2
Storage-Server1-Template	Storage-Node9	5	1
Storage-Server2-Template	Storage-Node10	5	2
Storage-Server1-Template	Storage-Node11	6	1
Storage-Server2-Template	Storage-Node12	6	2

Create Service Profile for Cisco UCS C220 M5 Server for HA-Proxy Node

To create a Service Profile, follow these steps:

1. Select **Servers** of the Cisco UCS Manager GUI.
2. Go to Servers > Service Profile > root and right-click to choose "Create Service Profile (expert)."

Identify Service Profile

To identify the service profile, follow these steps:

1. Type in in the Name field.
2. In the **UUID Assignment** section, select the UUID Pool you created in the beginning.
3. (Optional) Enter a description in the **Description** field.

Figure 44 Identify Service Profile

Create Service Profile (expert) ? X

You must enter a name for the service profile. You can also specify how a UUID will be assigned to this profile and enter a description of the profile.

Name : HA_Proxy-Node

The service profile will be created in the following organization. Its name must be unique within this organization.
Where : **org-root**

Specify how the UUID will be assigned to the server associated with this service profile.
UUID

UUID Assignment: Cloudian-UUID-Pools(26/50)

[Create UUID Suffix Pool](#)
The UUID will be assigned from the selected pool.
The available/total UUIDs are displayed after the pool name.

Optionally enter a description for the profile. The description can contain information about when and where the service profile should be used.

Service Profile creation for UCS C220M5 server used as Cloudian HA-proxy node

< Prev Next > **Finish** Cancel

4. Click **Next**.

Storage Provisioning

To configure the storage provisioning, follow these steps:

1. Go to the **Storage Profile Policy** tab and select the Storage Profile **C220-OS-Raid1** for the top node of the Cisco UCS S3260 Storage Server you created before.
2. Click **Next**.

Figure 45 Storage Provisioning

1 Identify Service Profile

2 **Storage Provisioning**

3 Networking

4 SAN Connectivity

5 Zoning

6 vNIC/vHBA Placement

7 vMedia Policy

8 Server Boot Order

9 Maintenance Policy

10 Server Assignment

11 Operational Policies

Create Service Profile (expert)

Optionally specify or create a Storage Profile, and select a local disk configuration policy.

Specific Storage Profile | **Storage Profile Policy** | Local Disk Configuration Policy

Storage Profile: **C220-OS-Raid1** Create Storage Profile

Name : **C220-OS-Raid1**
Description : **OS Boot LUN on RAID1 for C220M5 Server**

LUNs

Local LUNs | Controller Definitions | Security Policy

Advanced Filter | Export | Print

Name	Size (GB)	Order	Fractional Size (MB)
OS-Boot	1	Not Applicable	0

< Prev | Next > | **Finish** | Cancel

Networking

To configure networking, follow these steps:

1. Keep the Dynamic vNIC Connection Policy field at the default.
2. Select LAN connectivity to Use Connectivity Policy created previously.
3. From the LAN Connectivity drop-down list, select "Storage-Node" previously created.



HA-Proxy Node and Storage-Nodes use the same vNIC interfaces.

4. Click Next.

Figure 46 Summary Networking

Create Service Profile (expert)

Optionally specify LAN configuration information.

Dynamic vNIC Connection Policy:

[Create Dynamic vNIC Connection Policy](#)

How would you like to configure LAN connectivity?

Simple
 Expert
 No vNICs
 Hardware Inherited
 Use Connectivity Policy

LAN Connectivity Policy :

[Create LAN Connectivity Policy](#)

Initiator Name

Initiator Name Assignment:

[Create IQN Suffix Pool](#)

WARNING: The selected pool does not contain any available entities. You can select it, but it is recommended that you add entities to it.

< Prev Next > **Finish** Cancel

5. Click **Next** to continue with the SAN Connectivity.
6. Select No vHBA for How would you like to configure SAN Connectivity?
7. Click **Next** to continue with Zoning.
8. Click **Next**.

vNIC/vHBA Placement

To configure the vNIC/vHBA placement, follow these steps:

1. Select **Specify Manually** from the drop-down list.
2. Under PCI order section, Sort all the vNICs.
3. Make sure the vNICs order listed as External-Network > 1, then followed by Storage-Mgmt > 2 and Storage-Cluster > 3 Client-Network > 4.

Create Service Profile (expert)

Specify how vNICs and vHBAs are placed on physical network adapters

vNIC/vHBA Placement specifies how vNICs and vHBAs are placed on physical network adapters (mezzanine) in a server hardware configuration independent way.

Select Placement: **Specify Manually** [Create Placement Policy](#)

Virtual Network Interface connection provides a mechanism of placing vNICs and vHBAs on physical network adapters. vNICs and vHBAs are assigned to one of Virtual Network Interface connection specified below. This assignment can be performed explicitly by selecting which Virtual Network Interface connection is used by vNIC or vHBA or it can be done automatically by selecting "any".
vNIC/vHBA placement on physical network interface is controlled by placement preferences.

Please select one Virtual Network Interface and one or more vNICs or vHBAs

vNICs | vHBAs

Name: No data available

>> assign >>
<< remove <<

Specific Virtual Network Interfaces (click on a cell to edit)

Name	Or...	Ad...	Se...	Tr...
vCon 1			All	et...
vNIC External-Network	1	A...		
vNIC Storage-Mgmt	2	A...		
vNIC Storage-Cluster	3	A...		
vNIC Client-Network	4	A...		
vCon 2			All	et...

↑ Move Up ↓ Move Down

< Prev Next > **Finish** Cancel

4. Click Next to continue with vMedia Policy.

5. Click Next.

Server Boot Order

To configure the server boot order, follow these steps:

1. Select the Boot Policy "Local-OS-Boot" you created before under Boot Policy.
2. Server Boot Order.
3. Click Next.

Create Service Profile (expert)

Optionally specify the boot policy for this service profile.

Select a boot policy.

Boot Policy: **Local-OS-Boot** [Create Boot Policy](#)

Name : **Local-OS-Boot**
 Description : **OS boot policy for supervisor & Storage Nodes**
 Reboot on Boot Order Change : **No**
 Enforce vNIC/vHBA/iSCSI Name : **Yes**
 Boot Mode : **Legacy**

WARNINGS:
 The type (primary/secondary) does not indicate a boot order presence.
 The effective order of boot devices within the same device class (LAN/Storage/iSCSI) is determined by PCIe bus scan order.
 If **Enforce vNIC/vHBA/iSCSI Name** is selected and the vNIC/vHBA/iSCSI does not exist, a config error will be reported.
 If it is not selected, the vNICs/vHBAs are selected if they exist, otherwise the vNIC/vHBA with the lowest PCIe bus scan order is used.

Boot Order

Name	Order	vNIC/vH...	Type	LUN Name	WWN	Slot Num...	Boot Na...	Boot Path	Descripti...
CD/D...	1								
Local...	2								

[Create iSCSI vNIC](#) [Set iSCSI Boot Parameters](#) [Set UEFI Boot Parameters](#)

[< Prev](#) [Next >](#) [Finish](#) [Cancel](#)

Maintenance Policy

To configure the maintenance policy, follow these steps:

1. Select the Maintenance Policy you created before under Maintenance Policy.

Figure 47 Maintenance Policy

Create Service Profile (expert)

Specify how disruptive changes (such as reboot, network interruptions, firmware upgrades) should be applied to the system.

⊖ Maintenance Policy

Select a maintenance policy to include with this service profile or create a new maintenance policy that will be accessible to all service profiles.

Maintenance Policy: **Server-Maintenan** [Create Maintenance Policy](#)

Name : **Server-Maintenan**
 Description : **UCS Server Maintenance Policy**
 Soft Shutdown Timer : **150 Secs**
 Storage Config. Deployment Policy : **User Ack**
 Reboot Policy : **User Ack**

< Prev Next > **Finish** Cancel

2. Click **Next**.
3. From the Server Assignment drop-down list, choose “Select existing Server.”
4. Click “Available Servers” radio button.
5. From the Server list, select Rack ID “1” radio button for the Cisco UCS C220 M5 Server. This will Associate the service profile.

Create Service Profile (expert)

Optionally specify a server or server pool for this service profile.

You can select an existing server or server pool, or specify the physical location of the server you want to associate with this service profile.

Server Assignment: [Create Server Pool](#)

Select the power state to be applied when this profile is associated with the server.
 Up Down

Available Servers All Servers

Select	Chassis ...	Slot	Rack ID	PID	Procs	Memory	Adapters
<input checked="" type="radio"/>			1	UCSC-C220-M5SX	2	393216	1

Restrict Migration :

Firmware Management (BIOS, Disk Controller, Adapter)

< Prev Next > **Finish** Cancel

6. Click **Next**.

Operational Policies

To configure the operational policies, follow these steps:

1. Click **Finish** and then click **OK** and click Yes.
2. After Successful creation of "HA_Proxy-Node" Service profile, the Cisco UCS C220 M5 server will start the Service profile association.

Create Port Channel for Network Uplinks

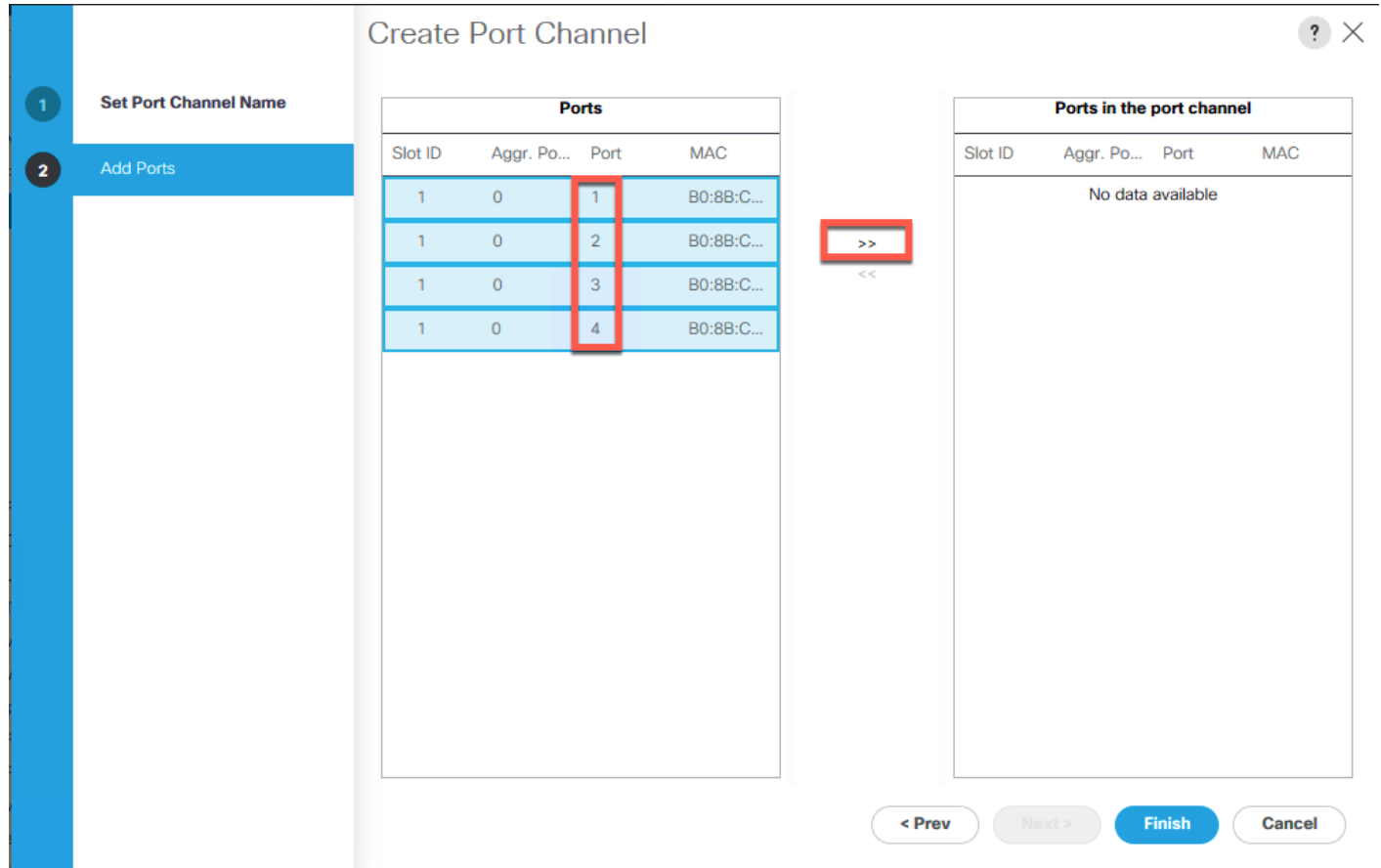
Create Port Channel for Fabric Interconnect A/B

To create Port Channels to the connected Nexus C9336C-FX2 switches, follow these steps:

1. Select the **LAN** tab of the Cisco UCS Manager GUI.
2. Go to LAN > LAN Cloud > Fabric A > Port Channels and right-click Create Port Channel.

3. Type in **ID 25**.
4. Type in **vPC25** in the Name field.
5. Click Next.
6. Select the available ports on the left **1-4** and assign them with **>>** to **Ports in the Port Channel**.
7. The “Add Ports” window will prompt you to confirm the selection, click Yes.

Figure 48 Create Port Channel



8. Click **Finish** and then click **OK**.
9. Repeat steps 1-8 for Fabric B under LAN > LAN Cloud > Fabric B > Port Channels and right-click Create Port Channel.
10. Type in **ID 26**.
11. Type in **vPC26** name in the Name field.
12. Click **Next**.
13. Select the available ports on the left **25-26** and assign them with **>>** to **Ports in the Port Channel**.

- Click **Finish** and then click **OK**.

The formal setup of the Cisco UCS Manager environment and both Cisco Nexus C9336C-FX2 switches is finished and next is the installation of the Red Hat Enterprise Linux 7.5 Operating System.

Install Red Hat Enterprise Linux 7.5 Operating System

This section provides the detailed procedures to install Red Hat Enterprise Linux 7.5 on Cisco UCS C220 M5 and Cisco UCS S3260 Storage Server. The installation uses the KVM console and virtual Media from Cisco UCS Manager.



This requires RHEL 7.5 DVD/ISO media for the installation.

Install RHEL 7.5 on Cisco UCS C220 M5 and Cisco UCS S3260 M5 Server

To install the Red Hat Linux 7.5 operating system, follow these steps:

- Log into the Cisco UCS Manager and select the **Equipment** tab from the left pane.
- Go to Equipment > Rack-Mounts > Server > Server 1 (HA-Proxy) and right-click KVM Console.
- Launch KVM Console.
- Click the **Activate Virtual Devices** in the Virtual Media tab.
- In the UCS KVM window, select the Virtual Media tab and then click **CD/DVD**.
- Click Choose File and Browse to the Red Hat Enterprise Linux 7.5 installation ISO image and select then click **Map Drive.**

Figure 49 Red Hat Enterprise Linux 7.5 ISO image



- In the KVM window, select the **Macros > Static Macros > Ctrl-Alt-Del** button in the upper left corner.
- Click **OK** and then click **OK** to reboot the system.
- In the boot screen with the Cisco Logo, press **F6** for the boot menu.
- When the Boot Menu appears, select **“Cisco vKVM-Mapped vDVD1.24”**

Figure 50 Boot Menu Selection



11. When the Red Hat Enterprise Linux 7.5 installer appears, press the Tab button for further configuration options.

12. At the prompt type:

```
inst.ks=ftp://192.168.100.220/storage-node1.cfg net.ifnames=0 biosdevname=0
ip=192.168.100.240::192.168.100.1:255.255.255.0:storage-node1:eth1:none
```



We prepared a Linux Kickstart file with all necessary options for an automatic install. The Kickstart file is located on a server in the same subnet.



The Kickstart file for the Cisco UCS C220 M5 server for HA-Proxy is in [Appendix A](#). This Kickstart file for the Cisco UCS S3260 M5 Server for Storage Nodes is in [Appendix B](#).

13. Repeat steps 1-12 to install RHEL7.5 on all the UCS S3260 M5 storage servers.

Cloudian HyperStore Preparation

Once the OS is installed login as root with the defined password in the kickstart file. The Cloudian HyperStore installation will be completed as root.

Software Version

This CVD guide is based on Cloudian HyperStore 7.1.2 but will support any version upgrades for 7.x.

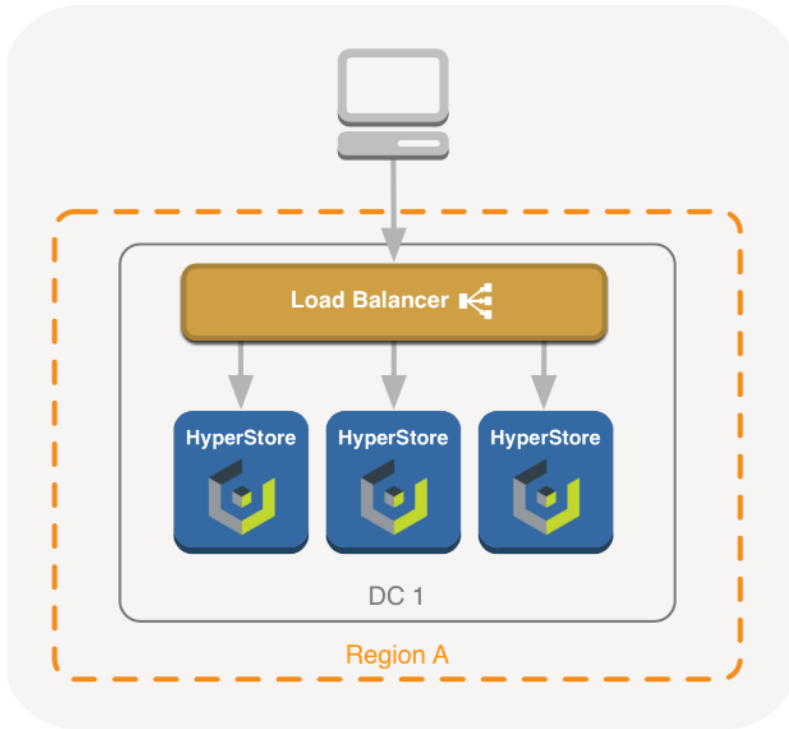
Load-Balancer Requirements

Cloudian HyperStore requires the use of a Load-Balancer or a VIP manager to ensure high-availability across the platform. Cloudian HyperStore supports working with most load-balancers and VIP managers.

In the Cloudian HyperStore High Availability architecture, the cluster is typically fronted by a Load Balancer. The purpose of the Load Balancer is to monitor the health of a node so that traffic is not routed to a node that is unhealthy or offline, as well as balance the workload evenly across cluster nodes. There must be a component that can redirect the work and there must be a mechanism to monitor for failure and transition the system if an interruption is detected. Without a Load Balancer, a node that is offline would still receive requests from clients. Those requests would then just fail. In general, Load Balancers will distribute requests to nodes that belong to a pool of available service members. A Load Balancer will also perform frequent health checks against pool member nodes to ensure they are healthy and able to support new traffic.

All Cloudian HyperStore S3, Admin-API and CMC services should be configured with a Load Balancer to ensure any kind of High Availability. There are many Load Balancing solutions that are available for use. Commercial examples are F5, A10 Networks, KEMP, Loadbalancer.org and Citrix Netscaler. Open source Load Balancer software exist as well, one popular example is HAProxy. Most of the Load Balancing technologies operate in a similar manner, some enterprise solutions and and DNS services like Amazon Route53 however also include support for GEO based load distribution (known as GI Gobar Server Load Balancing).

Figure 51 Load Balancing Example



Concepts of Load Balancing

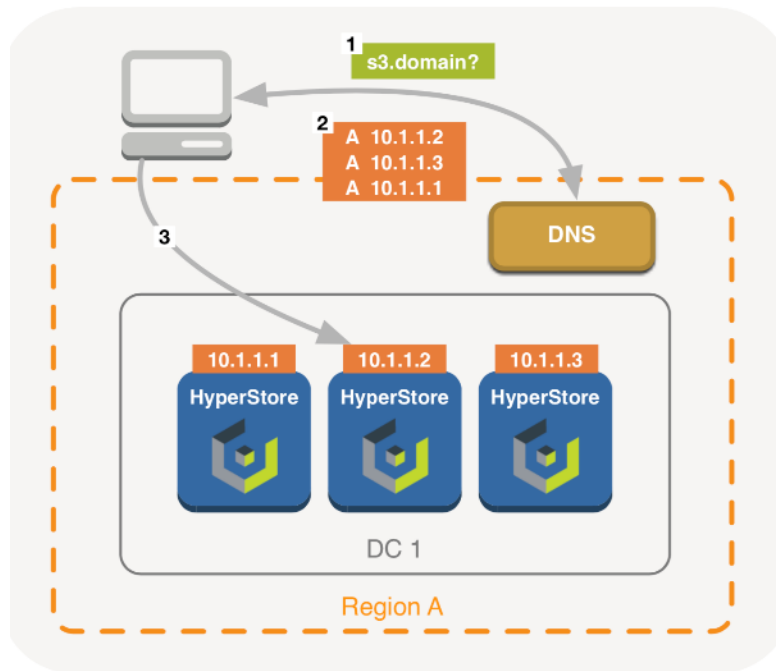
Round-Robin DNS

Round-robin DNS (RR-DNS) is a method where a series of A records is registered in DNS, each by the same name. For example, whenever a client requests “s3.domain” in DNS, the reply will contain a certain order of the above records. The next request will however be answered with a rotated list of those records. This way, traffic is automatically “balanced” across the mentioned addresses.

```
s3.domain IN A 10.1.1.1 s3.domain IN A 10.1.1.2
s3.domain IN A 10.1.1.3
```

RR-DNS is very simple to implement as DNS is already available everywhere, but without combining it with other HA solutions, it isn't very useful in itself. If any of those nodes are offline, you would still be directing requests to them and so, those requests would fail.

Figure 52 Basic Round-Robin DNS Example

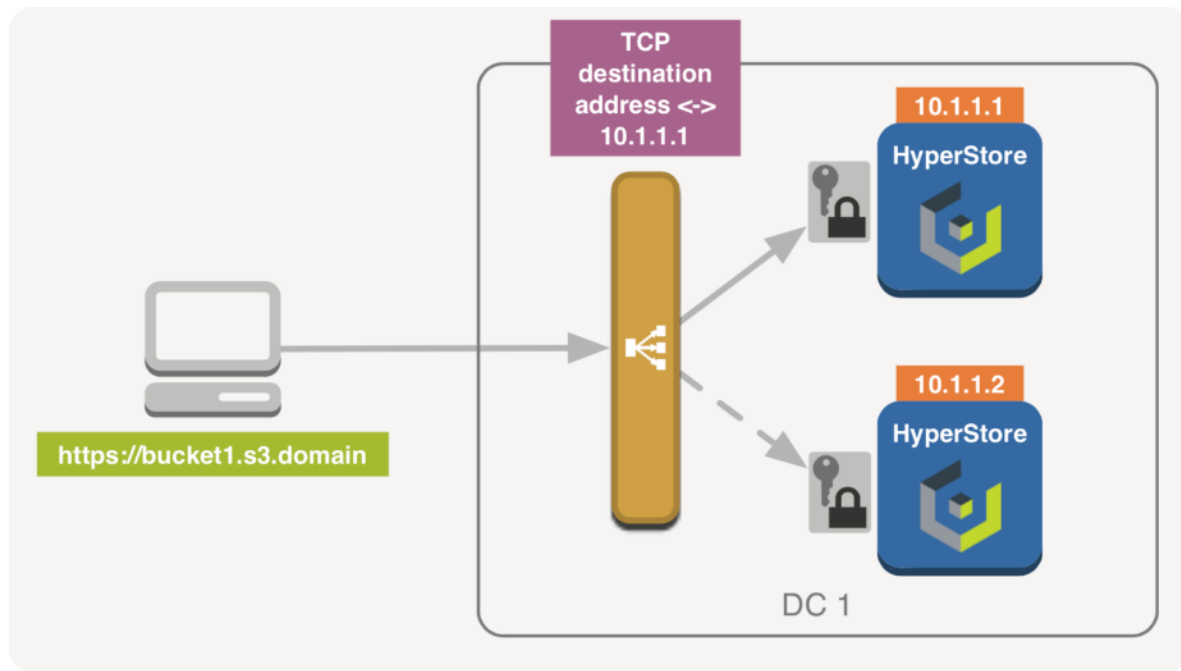


Even if you would dynamically update those DNS records according to the monitored health of the nodes, an issue might still be that you are relying on DNS and have no control over caching of those DNS records client-side, or on intermediate caching nameservers (read: low TTL values are often discarded on large DNS resolvers). In scenario's where you do have control over the nameservers our S3 clients are using, RR-DNS and dynamic DNS updates might be a proper solution when implemented correctly. If the Cloudian HyperStore services would be published externally, in most cases it will be a better solution to combine RR-DNS with other high availability and/or Load Balancing technologies.

Layer-4 Load Balancing

Layer 4 Load Balancing operates on the transport layer in the OSI model. This means that although the TCP connection is established on the Load Balancer, anything above that (like HTTP) is tunneled across both sides. The Load Balancer can therefore make a balancing decision based on anything in the TCP header, however it cannot look inside the payload or perform more advanced things like injecting a session cookie or inspect URI.

Figure 53 Layer 4 Balancing Example



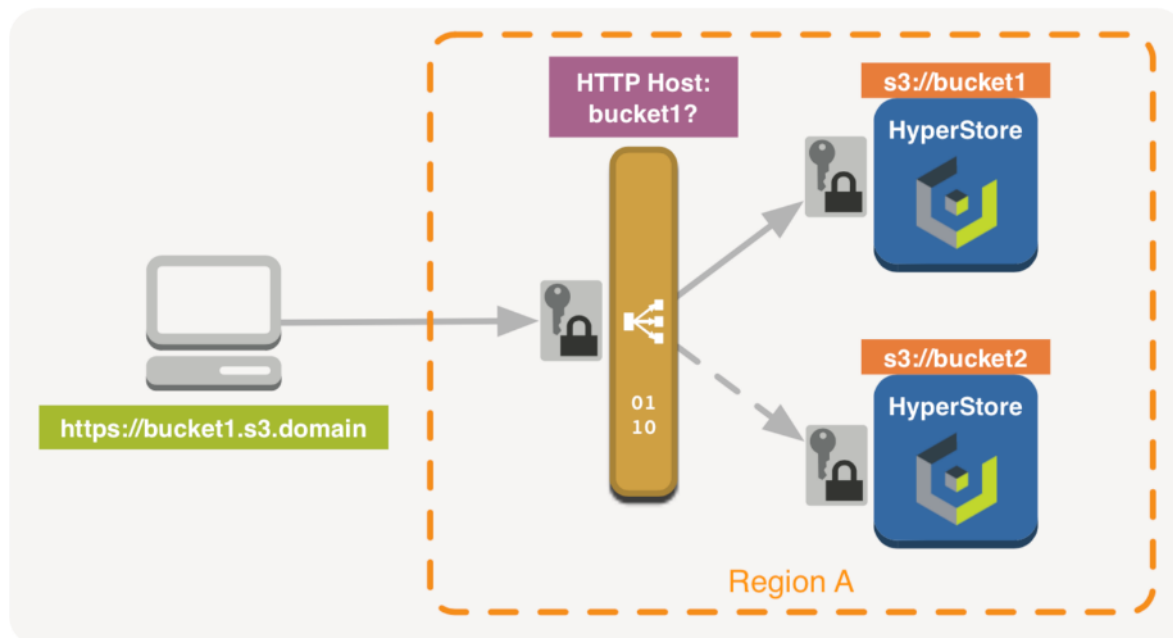
TLS termination is done on HyperStore nodes only.

The layer 4 or “TCP mode” is sufficient and has the advantage that because it operates on a much lower level, Load Balancer resources will less likely become the first bottleneck (there are limits of course, scaling the Load Balancing layer itself is explained later). In case of TLS, termination happens on the back-end nodes only. This mode fits the scale-out nature of HyperStore best, as all crypto calculation involved with TLS will be spread evenly across all HyperStore nodes as well (although the overhead involved is not nearly what it used to be, due to dedicated instructions (AES) available in many CPUs and optimizations in TLS handshake). No change to SSL certificate management is required, certificates are still only managed on the HyperStore installer node (puppet master) as described in chapter "Setting up HTTPS/SSL for the S3 Service" in the Cloudfian Documentation.

Layer-7 Load Balancing

Layer 7 Load Balancing operates on the application layer in the OSI model. In this “HTTP mode”, all incoming connections are established on the Load Balancer, the payload is inspected and new HTTP sessions are created between the Load Balancer and available back-end nodes. This mode is heavier on resources than layer 4 and HTTP mode is typically used when one needs to make balance decisions based on for example, HTTP headers or inspect cookies to maintain a session to a back-end.

Figure 54 Layer 7 Balancing



SSL Certificates need to be managed on LB.

In the case of HyperStore, this generally doesn't add too much value as the balancing algorithm for S3 and API can be more or less random, or based on e.g. least connections to a back-end node. For CMC requests, you do need to configure stickiness to a back-end node, but basing that on for example, source IP address is usually sufficiently random.

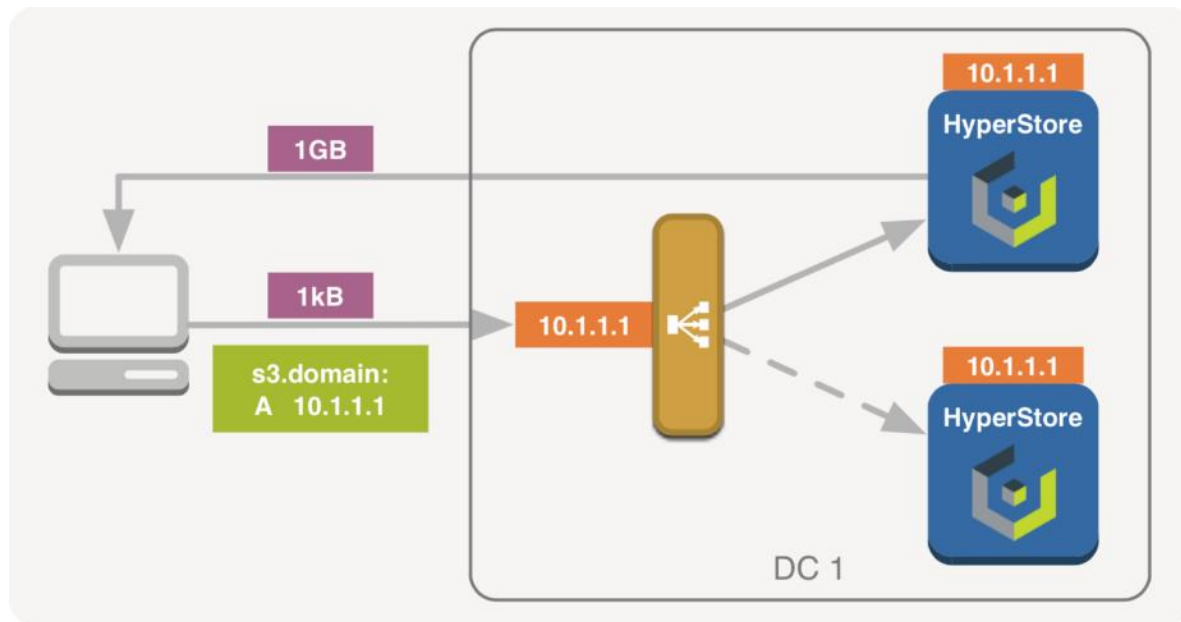
In HTTP mode, SSL certificates need to be managed both on the HyperStore installer node and on all Load Balancers involved. One exception would be, when traffic from the Load Balancers to the HyperStore nodes is not required to be encrypted. In that case SSL certificates only need to be maintained on the Load Balancers (usually referred to as SSL offloading).

Direct Routing

A typical issue to Load Balancing is that all traffic needs to pass the Load Balancer, both ways. Especially with a solution like an Object Store, where you are combining both "scale-out" and large data transfers, odds are that a Load Balancer will become the first bottleneck in the chain.

One (partial) solution to that is a concept called "Direct Routing" also known as "Direct Server Return" (DSR). With Direct Routing, a back-end node does not rely on the Load Balancer to send its reply back to the client. Instead, the back-end nodes have the "VIP" (Virtual IP) or Load Balancer address attached to their local interface (ARP replies for that address will need to be switched off), and are thus able to send a TCP reply directly back to the client with the source address (the VIP) the client is expecting. One such example is the LVS project, but some commercial Load Balancers also support Direct Routing.

Figure 55 IGB GET Object Example with Direct Routing



This would relieve the Load Balancer from (often large) GET request replies returning to the client, however all PUT requests would still need to pass the Load Balancer on their way in. Obviously, all traffic still passes other network peripherals like Routers, unless S3 clients and HyperStore or the Load Balancers are on the same layer 2 network.

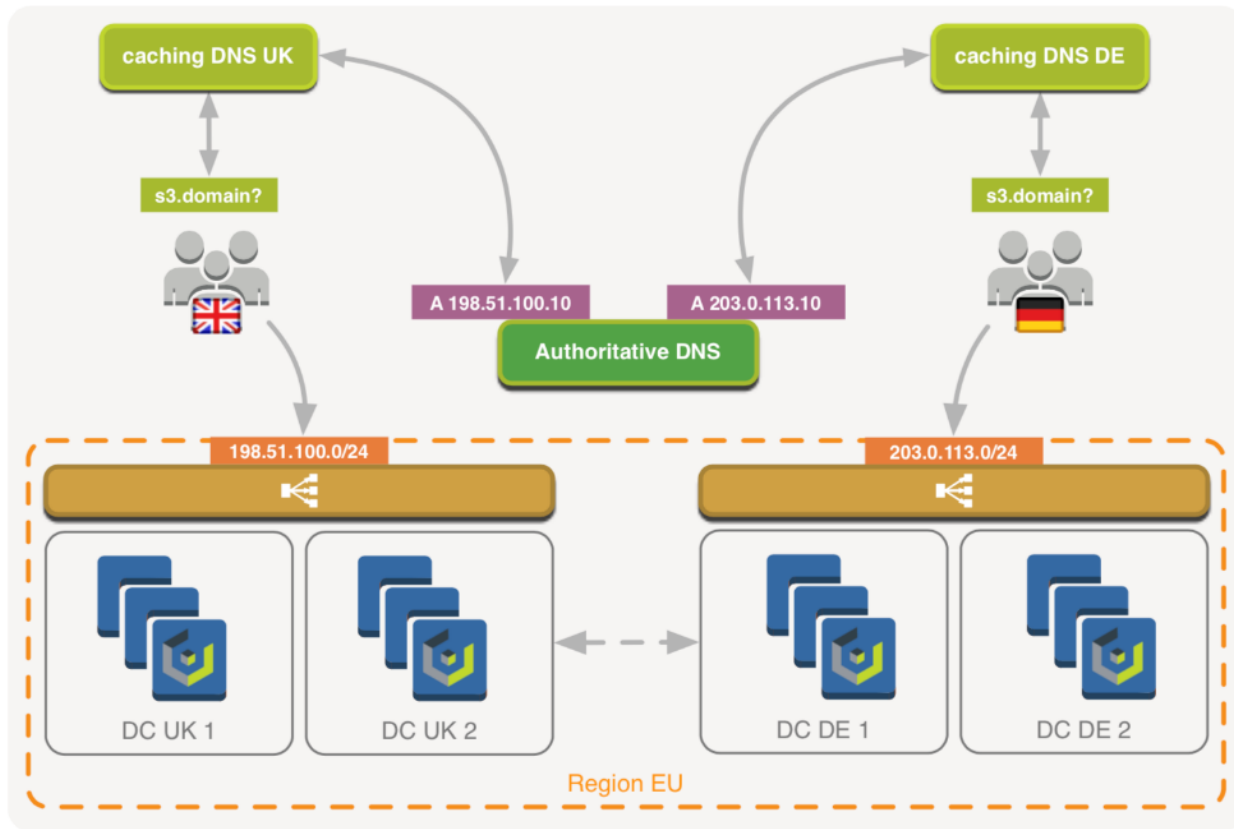
Direct Routing is not a setting one can just turn off and on, instead it's a mechanism acting on multiple levels within a network and often requires some low-level manipulation to configure correctly.

Direct Routing cannot be combined with L7 balancing or SSL offloading and on Linux only available through the Linux Virtual Server project.

Global Server Load Balancing

Global Server Load Balancing (GSLB) is a mechanism designed to provide Disaster Recovery, load distribution and/or ensure shortest path or best response between client and Datacenter. Essentially, the technology itself isn't that complicated; Based on, any number of things really but usually, geographical location and/or availability of a Datacenter, a user receives a DNS reply which routes the request accordingly. Although GSLB does provide load distribution and is present in a number of commercially available Load Balancers, GSLB is built on top of DNS and is not necessarily, or typically, a physical Load Balancer.

Figure 56 GSLB Example-Geographically Spread Balancing



In the above example, users from the UK would receive a DNS record pointing to UK-based Datacenters and likewise, German users receive an address pointing to a German DC. Just like directing users to specific geographical areas, GSLB can also be used to directly return a pool of addresses of healthy HyperStore nodes instead. GSLB solutions typically monitor health of a destination and manage the DNS records returned by an authoritative DNS nameserver, either for availability purposes, balancing, localization or a combination. A viable architecture could for example be based on Amazon’s Route53 for handling Geo- based DNS, and have traditional Load Balancers in front of HyperStore in each location.

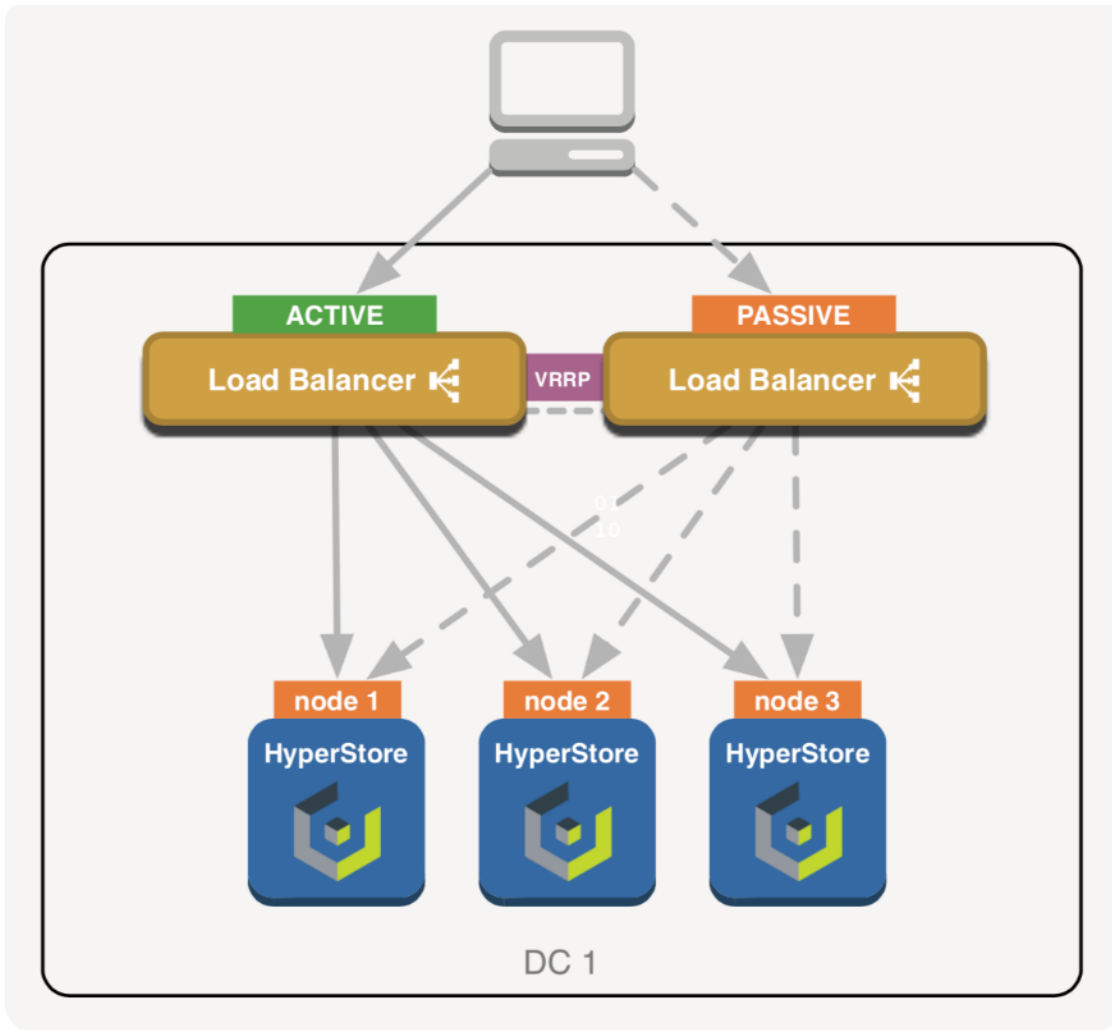
Just like with RR-DNS or anything based on DNS, aggressive caching of DNS records may become an issue depending on the overall architecture and infrastructure between client and S3 service.

High Availability

Load Balancing in itself does not necessarily equal high availability; when using multiple back-ends but just a single Load Balancer, the Load Balancer becomes the Single Point of Failure. This is usually resolved by adding another Load Balancer and enabling a failover mechanism between both Load Balancers. Most enterprise Load Balancers will support such a failover mechanism, often based on protocols like Virtual Router Redundancy Protocol (VRRP).

An open source solution often deployed in combination with LVS or HAProxy is Keepalived. Based on VRRP, Keepalived can be configured to allow a pool of “floating IP addresses” where each Virtual IP (VIP) will only be active on a single node at any given time. Whenever a node would fail, in this case a Load Balancer, any VRRP address attached to that node would be seamlessly migrated to any of the remaining nodes.

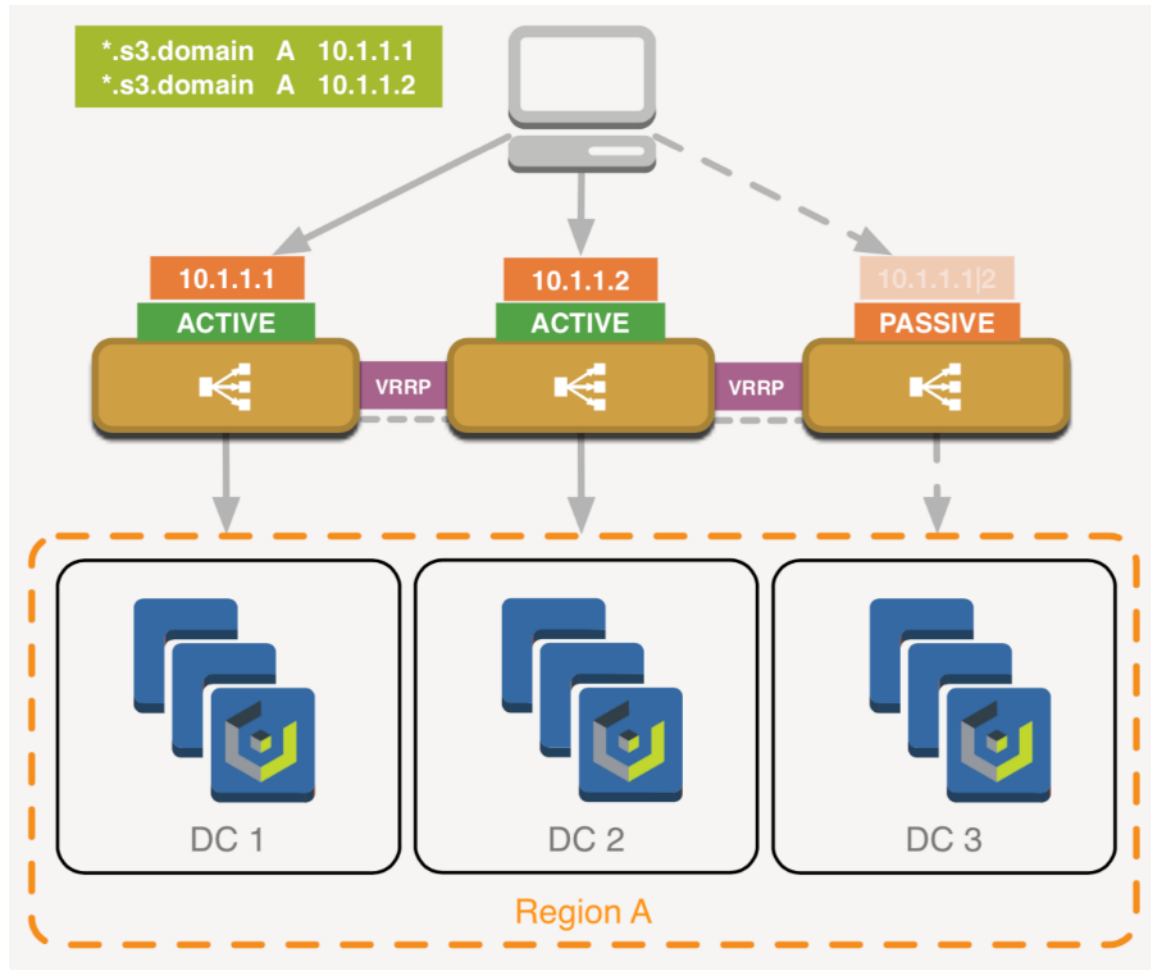
Figure 57 High Availability Load Balancing Example Based on VRRP



When working with VRRP-like protocols make sure Multicast is allowed on the relevant ports, either by disabling IGMP Snooping for those ports or joining the correct IGMP group.

When deploying a Load Balancing layer in front of HyperStore, especially when that layer is spread across multiple Datacenters, a single active Load Balancer might not be preferred or even viable. This is where Round-Robin DNS is actually very useful; When scaling the Load Balancing layer horizontally and using multiple Load Balancers in an Active-Active setup, you can now make use of RR-DNS so that multiple DNS records are spread evenly across multiple, active Load Balancers. Effectively creating an N+1 setup on Load Balancer level.

Figure 58 N+1 Load Balancing + Round-Robin DNS



Why not decide to run all Load Balancers in an Active mode and direct traffic to them? It’s usually a best practice to keep an N+1 setup, especially when N is a relatively low number, as you won’t have any real insight in how your remaining Load Balancers will handle the load, number of connections etc. until any one of the Load Balancers would fail (and it turns out they weren’t able to handle 33 percent more connections or additional throughput). As with any high availability technology, setting it up properly can get rather complex and usually involves configuration on multiple levels, often including switch port fine-tuning as well.

For that reason, this document does not provide in-depth details about how to set up high availability between your Load Balancers step-by-step, but does cover the technologies involved and some basic examples.

Load Balancing HyperStore

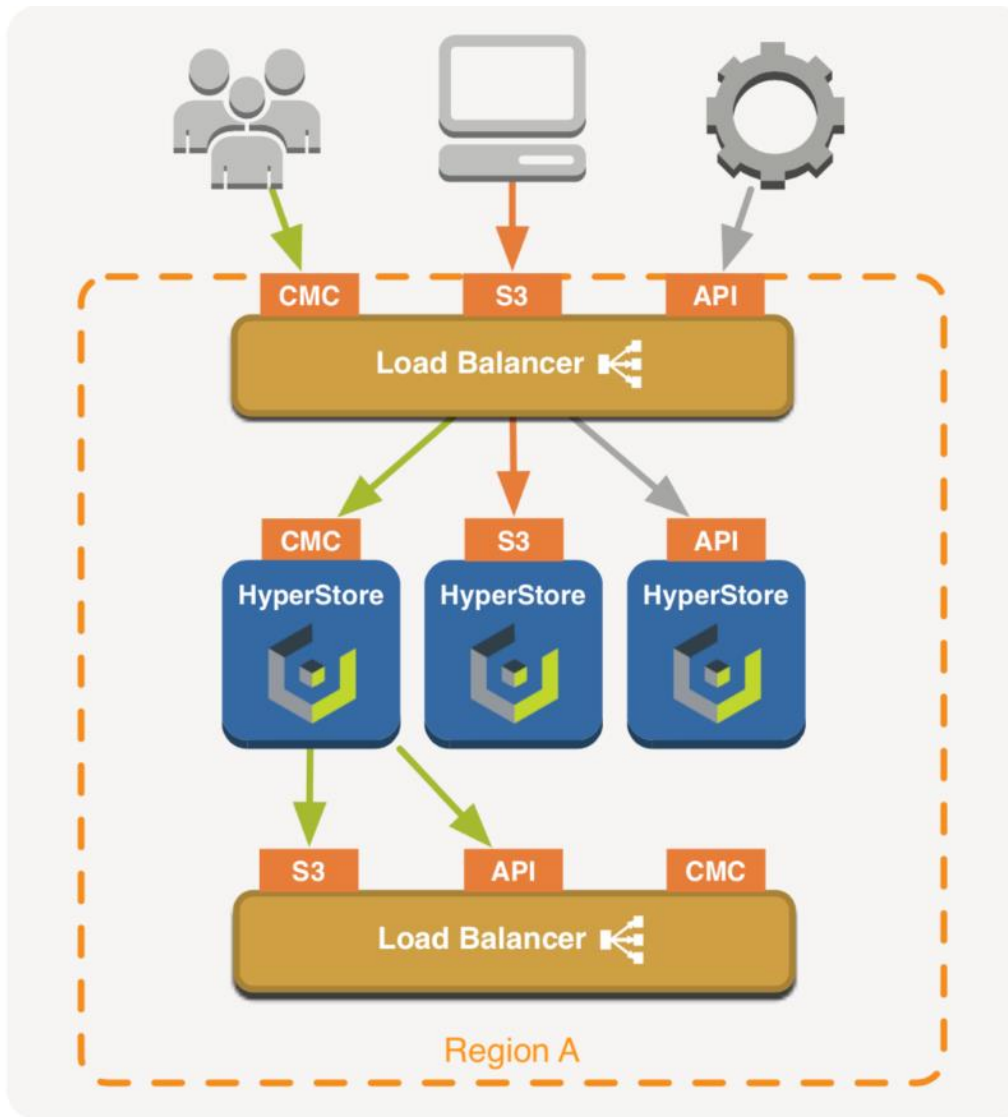
HyperStore Services

The HyperStore services that should be balanced are: S3, Cloudian Management Console (CMC) and Admin-API. Besides advertising those services to any clients, all HyperStore nodes within the cluster will also benefit and make use of S3 and Admin-API being highly available. All other, internal services like Cassandra and Redis are cluster-aware, meaning that as part of the HyperStore installation they’ve received topology information and know how to communicate directly to all other nodes. These internal services do not need to be taken into consideration when architecting Load Balancing within your network.



The only HyperStore services that need to be balanced are: S3, CMC and Admin-API.

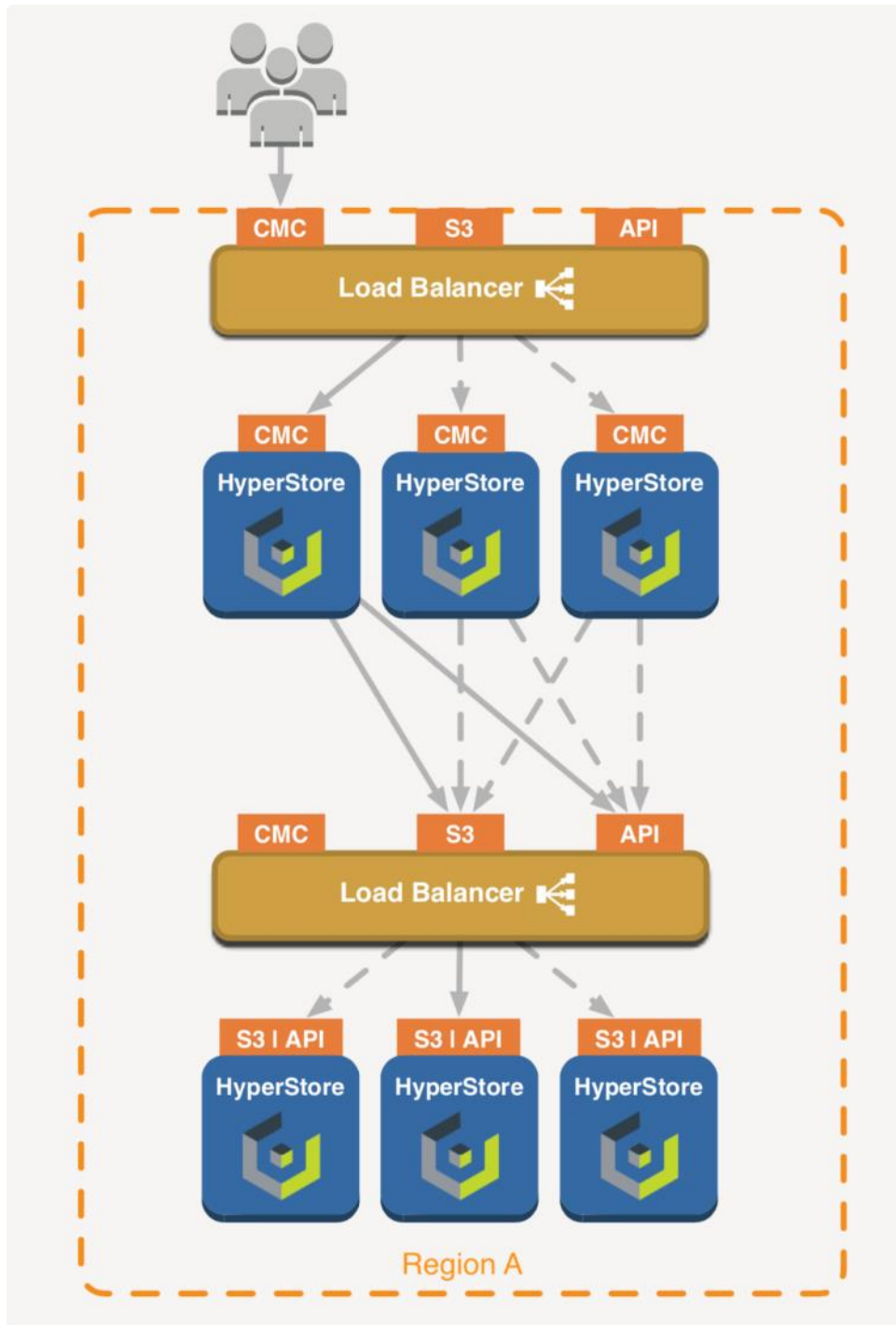
Figure 59 Balancing HyperStore Services



The above diagram is a simplified view on how different clients connect to S3, CMC and the Admin-API. As you can see, the Cloudian Management Console can be considered a client as well. The CMC connects through the Load Balancer to both S3 and Admin-API services, which should in turn be balanced across all HyperStore nodes.

Note that all S3, CMC and Admin-API services are running on each and every HyperStore node. The following diagram reflects a client connecting to the CMC.

Figure 60 CMC Connection Flow



Make sure the Admin-API is made highly available, since the CMC communicates directly with the API.

HyperStore Configuration

For detailed configuration instructions about how to prepare HyperStore and DNS for high availability, please see chapter "DNS Set-Up" in the Cloudian Documentation. You need to make sure all S3, CMC and Admin-API DNS records point to the Load Balancer.

When installing HyperStore with option `configure-dnsmasq` (`no-dnsmasq` is default), a simple resolver will be installed on each HyperStore node, and all required records will be added to `dnsmasq` automatically. Note that for production use, it is not recommended to install with `dnsmasq` enabled. Instead, you need to make sure the following records are all present in DNS before installation.

In this single Region example, 10.1.1.10 is the IP address attached to the Load Balancer, "region1" is the Region name and "domain" is the Domain name. During installation of HyperStore these hostnames can all be customized.

To install with `dnsmasq`, follow these steps:

1. Customize template: `/etc/cloudian- <version> -puppet/modules/dnsmasq/templates/dnsmasq.conf.erb` to reflect the DNS example shown above.
2. Push the update to the cluster.
3. Restart service `dnsmasq`.

HAProxy Examples

HAProxy – Basic Configuration

As mentioned earlier, HAProxy is a widely used Load Balancer solution, used by Twitter, Amazon AWS, GitHub and Netflix and is available as Open Source, Enterprise version, as an appliance (ALOHA) and also available in the Loadbalancer.org appliances. HAProxy is known to be very performant, stable and feature-rich (for in an-depth explanation of all options, features and syntax please refer to the online documentation).

Installation of HAProxy is very straightforward. When installing on a RedHat-based distribution, all that is required is to run the following commands:

```
sudo yum update
sudo yum -y install haproxy systemctl enable haproxy.service
```

When installing on a Debian-derivative the command would be:

```
sudo apt-get update
sudo apt-get install haproxy
```

And set `ENABLED=1` in `/etc/default/haproxy`

```
s3-region1.domain IN A 10.1.1.10
*.s3-region1.domain s3-website-region1.domain *.s3-website-region1.domain
IN A IN A IN A
10.1.1.10 10.1.1.10 10.1.1.10
s3-admin.domain IN A 10.1.1.10 cmc.domain IN A 10.1.1.10
```

Move the default HAProxy configuration file `/etc/haproxy/haproxy.cfg` aside and create a new one. This document assumes there are three Cloudian nodes to balance the load across. With Cloudian HyperStore all HTTP REST API services run on every node so the configuration is quite simple. The node IP's here are assumed to be 10.1.1.11, 10.1.1.12, and 10.1.1.13.

Configuration – Global Section

```
global
log /dev/log local0
log /dev/log local1 notice chroot /var/lib/haproxy user haproxy
group haproxy spread-checks 5 tune.bufsize 32768 tune.maxrewrite 1024 maxconn
16384
daemon
```

Because you're running inside a chroot environment, the local syslog server would need to create a listening socket in `/var/lib/haproxy/dev`. In rsyslog the syntax would be: `"$AddUnixListenSocket /var/lib/haproxy/dev/log"`.

Configuration – Defaults Section

```
defaults

    log global

mode tcp
maxconn 8192 timeout connect 5s timeout client 1m timeout server 1m timeout check
5s balance leastconn
```

Add "option tcplog" to the defaults section to log every connection to each front-end.

Configuration – Admin Statistics

```
# admin stats on port 8080 listen stats

bind :8080
mode http
stats enable
maxconn 128
stats uri /
stats realm Haproxy\ Statistics stats auth admin:public
```

For production use the statistics page should be reachable over TLS only and a proper password should be configured.

Configuration – Backend CMC

```
# Cloudian CMC
listen cmc.cloudian-hyperstore

bind :8888
mode http
http-request replace-value Host (.*) :8888 \1:8443 http-request redirect code 302
location

https://[%[hdr(host)]]%[capture.req.uri]

listen https.cmc.cloudian-hyperstore bind :8443

mode tcp
stick-table type ip size 100k expire 30m
stick on src
```



```

option httpchk HEAD /Cloudian/login.htm
description Cloudian HyperStore CMC - HTTPS
server cloudian-node1 10.1.1.11:8443 check check-ssl verify none

inter 5s rise 1 fall 2
server cloudian-node2 10.1.1.12:8443 check check-ssl verify none

inter 5s rise 1 fall 2
server cloudian-node3 10.1.1.13:8443 check check-ssl verify none

inter 5s rise 1 fall 2

CMC balance algorithm needs to be sticky (stick-table, stick on src)

```

Configuration - Backend S3 HTTP

```

# Cloudian S3 services
listen s3.cloudian-hyperstore

bind :80
mode tcp
option httpchk HEAD /.healthCheck
description Cloudian HyperStore S3
server cloudian-node1 10.1.1.11:80 check inter 5s rise 1 fall 2 server cloudian-
node2 10.1.1.12:80 check inter 5s rise 1 fall 2 server cloudian-node3
10.1.1.13:80 check inter 5s rise 1 fall 2

```

HyperStore S3 includes a health check page, reachable over HTTP method HEAD.

Configuration - Backend S3 HTTPS

```

# Cloudian S3 services - HTTPS listen https.s3.cloudian-hyperstore

bind :443
mode tcp
option httpchk HEAD /.healthCheck
description Cloudian HyperStore S3 - HTTPS
server cloudian-node1 10.1.1.11:443 check check-ssl verify none

inter 5s rise 1 fall 2
server cloudian-node2 10.1.1.12:443 check check-ssl verify none

inter 5s rise 1 fall 2
server cloudian-node3 10.1.1.13:443 check check-ssl verify none

inter 5s rise 1 fall 2

```

When a CA-Verified certificate is used for S3, the "verify none" should be omitted

Configuration - Backend Admin API

```

# Cloudian Admin-API
listen api.cloudian-hyperstore

bind :19443

mode tcp
option httpchk HEAD /.healthCheck HTTP/1.0\r\nAuthorization:\ Basic\
c3lzYWRtaW46cHVibGlj
description Cloudian HyperStore API

```

```
server cloudian-node1 10.1.1.11:19443 check check-ssl verify none inter 5s rise 1
fall 2

server cloudian-node2 10.1.1.12:19443 check check-ssl verify none inter 5s rise 1
fall 2

server cloudian-node3 10.1.1.13:19443 check check-ssl verify none inter 5s rise 1
fall 2
```

When you have customized the Admin-API credentials make sure to replace the base64 encoded string shown in the example (which is the base64 version of the default credentials "sysadmin:public")

The encoded credentials can be generated on the command line: `echo -n "<username>:<password>" | base64`

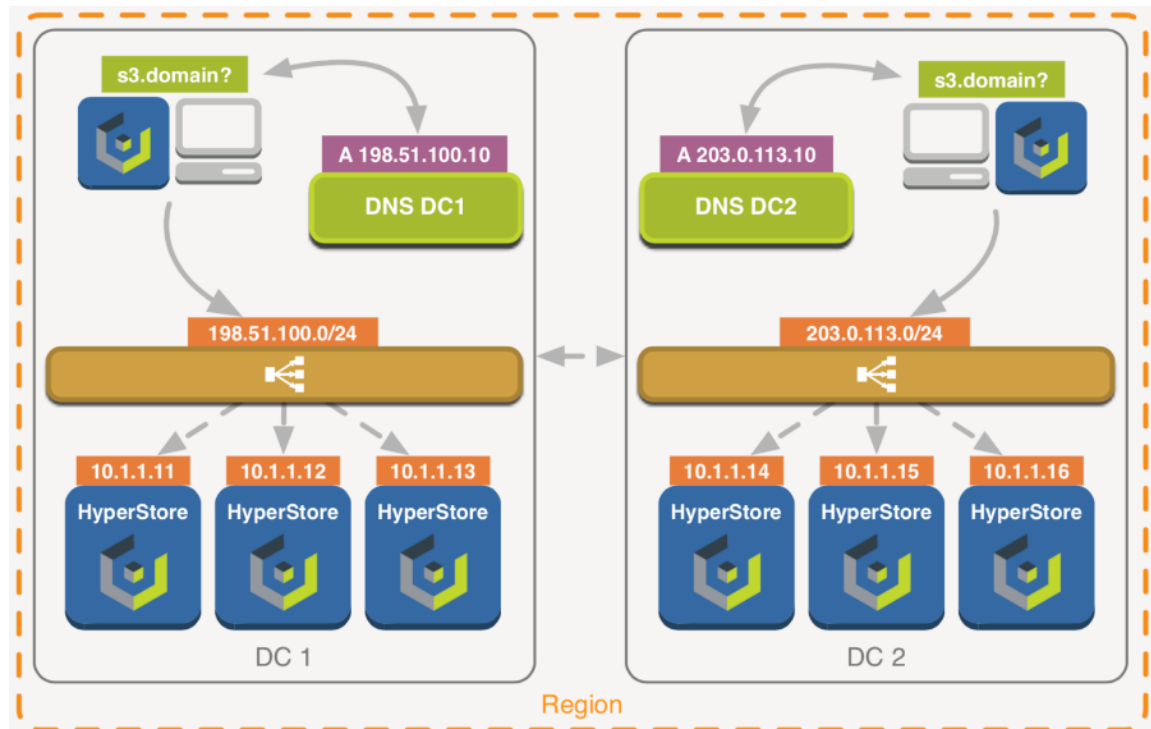
HAProxy – Location Affinity

HAProxy by itself does not provide any GSLB-like capabilities, and the following example won't actually be very useful when HyperStore is running public services, for example in the case of a publicly reachable S3 endpoint, let's say the public Storage as a Service use-case. However, imagine all your S3 clients are internal applications within a single S3 Region which spreads across multiple Datacenters, and you want to use some form of location affinity because you don't prefer an application in DC1, connecting to an S3 endpoint in DC2 half of the time.

When using a single S3 endpoint across multiple Datacenters you could create complex rules, inspect HTTP headers and base a routing decision on e.g. name of the Bucket or even the subnet the application is connecting from by defining ACL's and multiple back-ends. However, this might not always cover every scenario and may also create a configuration more complex than desired.

Instead, what you could do is simply run different DNS nameservers in both Datacenters and register all service records to point to the closest Load Balancer. This way a client connecting from within DC 1 would always be directed to the Load Balancer in DC 1. The same applies to DC 2.

Figure 61 Data Center Affinity Example



The nameservers do not necessarily need to run on dedicated servers and could for example easily be installed on the Load Balancer nodes. The following is an example setup of the S3 service in HAProxy, combined with installing and running the light-weight DNS (amongst others) server “dnsmasq”.

HAProxy Configuration – S3 DC1

```
listen s3.cloudian-hyperstore bind :80

mode tcp
option httpchk HEAD /.healthCheck
description Cloudian HyperStore S3 DC1
server cloudian-node1 10.1.1.11:80 check inter 5s rise 1 fall 2 server cloudian-
node2 10.1.1.12:80 check inter 5s rise 1 fall 2 server cloudian-node3
10.1.1.13:80 check inter 5s rise 1 fall 2 server cloudian-node4 10.1.1.14:80
check inter 5s rise 1 fall 2

backup
server cloudian-node5 10.1.1.15:80 check inter 5s rise 1 fall 2

backup
server cloudian-node6 10.1.1.16:80 check inter 5s rise 1 fall 2

backup
```

HAProxy Configuration – S3 DC2

```
listen s3.cloudian-hyperstore bind :80

mode tcp
option httpchk HEAD /.healthCheck
description Cloudian HyperStore S3 DC1
server cloudian-node4 10.1.1.14:80 check inter 5s rise 1 fall 2 server cloudian-
node5 10.1.1.15:80 check inter 5s rise 1 fall 2 server cloudian-node6
10.1.1.16:80 check inter 5s rise 1 fall 2 server cloudian-node1 10.1.1.11:80
check inter 5s rise 1 fall 2

backup
server cloudian-node2 10.1.1.12:80 check inter 5s rise 1 fall 2

backup
server cloudian-node3 10.1.1.13:80 check inter 5s rise 1 fall 2

backup
```

By adding the remote nodes as “backup” back-end servers here, then whenever local HyperStore nodes would fail, the remote HyperStore nodes would become active. Alternatively, a single “backup” entry could be used by pointing that to the VIP of the remote Load Balancer.

Install DNSMASQ

To install dnsmasq on RedHat-based distributions, run the following commands:

```
sudo yum update
sudo yum -y install dnsmasq systemctl enable dnsmasq.service
```

DNSMASQ Configuration

Use the default configuration. Make sure that you won’t be allowing recursion to requests from “the internet”. As recursion is allowed by default in dnsmasq, you can either set the “interface” option to only listen on the specified,

internal interface, or an intermediate Firewall should be configured not to allow external DNS traffic to pass in. To listen only on a specified interface, create a file `"/etc/dnsmasq.d/custom.conf"` and add the following inside (adjust Interface name, VLAN 10 in this example):

```
interface=enp0s3.10 bind-interfaces
```

Add the following configuration to that same file, one in each data center (adjust addresses, "region" and "domain" to match your environment):

DNSMASQ Configuration - DC1

```
address=/.s3-region.domain/198.51.100.10 address=/s3-region.domain/198.51.100.10
address=/.s3-website-region.domain/198.51.100.10
address=/cmc.domain/198.51.100.10 address=/s3-admin.domain/198.51.100.10
```

DNSMASQ Configuration - DC2

```
address=/.s3-region.domain/203.0.113.10 address=/s3-region.domain/203.0.113.10
address=/.s3-website-region.domain/203.0.113.10 address=/cmc.domain/203.0.113.10
address=/s3-admin.domain/203.0.113.10
```

After saving all files, restart both dnsmasq and HAProxy to apply all configurations. At this point, all HyperStore nodes can now be reconfigured to use the Load Balancers (or other nodes if you installed dnsmasq on separate servers) as resolvers. In the same way, all clients and applications within the same Datacenters that connect to HyperStore, can now be pointed to use dnsmasq as resolver(s) as well (or use DNS delegation in your existing DNS infrastructure).

Some commercial Load Balancers like Citrix Netscaler, F5 GTM and Loadbalancer.org come equipped with GSLB- or GSLB-like features (like location affinity based on subnet of incoming requests).

Proxy Protocol

By using HAProxy and most other Load Balancers or proxies, one will lose the source IP address of the actual client performing the request. For logging purposes, this could be circumvented by using the "X-Forwarded-For header" sent by the Load Balancer. However, this would only work when using HTTP level balancing but more importantly, it would still not cover more advanced S3 features such as using conditions based on IP addresses in S3 Bucket Policies, and IP addresses and/or subnets used in HyperStore Rating Plan whitelists.

HAProxy can be set up to run in full transparent mode (TPROXY) but that may require recompiling the Linux kernel with TPROXY support, recompiling HAProxy, marking packets with IPtables and adding custom routing tables. Moreover, in full transparent mode all HyperStore nodes will need to use the Load Balancers as their default gateway which is not typically preferred.

To avoid too much complexity around this issue, the HAProxy team developed the following PROXY protocol:

"The PROXY protocol provides a convenient way to safely transport connection information such as a client's address across multiple layers of NAT or TCP proxies. It is designed to require little changes to existing components and to limit the performance impact caused by the processing of the transported information."

Between HaProxy and the backend nodes an additional PROXY header is passed within a Datagram and processed by the application running on the backend nodes. Amongst others, this contains the original source IP of the actual client. So the mechanism does need to be supported by the application receiving the PROXY protocol, at the moment HyperStore S3 support the PROXY protocol but the CMC does not (yet).

If visibility of client IP addresses are a strict requirement for both S3 AND the CMC, a suggested configuration would be to run the CMC in HTTP mode plus using the X-Forwarded-For header and enabling the PROXY protocol for S3 in TCP mode.

Many other, commercial Load Balancing appliances also support the PROXY protocol. F5, ALOHA and Loadbalancer.org are known to support PROXY as well.

Enable Proxy for S3

As PROXY needs to be enabled and used on both client as server, when enabled on HyperStore it will create additional listening sockets for PROXY on dedicated ports (81 for S3 over HTTP and 4431 if S3 HTTPS is enabled). On the HAProxy level, all that is required is to add the "send-proxy" option to the S3 backend nodes and point those to the PROXY-enabled port on HyperStore.

The following is an example:

```
server cloudian-node1 10.1.1.11:81 check send-proxy inter 5s rise 1 fall 2
```

And subsequently for S3 HTTPS;

```
server cloudian-node1 10.1.1.11:4431 check send-proxy check-ssl verify none inter 5s rise 1 fall 2
```

On HyperStore you will enable the PROXY protocol by setting the following to true in

```
/etc/cloudian-7.0-puppet/manifests/extdata/common.csv:
s3_proxy_protocol_enabled,true
```

Please refer to chapter "Pushing Configuration File Edits to the Cluster and Restarting Services" of the official Cloudian HyperStore documentation about how to apply these changes to the cluster and restart the HyperStore S3 service. After this change to HyperStore, reload the HAProxy service to apply the changes on the Load Balancer(s).

To minimize downtime, make the required changes to HAProxy and HyperStore first and push those across the cluster, but only restart S3 and HAProxy services afterwards and around the same time. Both sides need to have PROXY either enabled or disabled to communicate.

DNS Requirements

Cloudian HyperStore uses Service Endpoints Names to ensure client requests are correctly resolved and handled. The following Service to be defined in DNS and accessible to clients in order to successfully connect and use the Object Store.

Service Endpoint	DNS Host A Record Example	Ports	Description
s3 Service Endpoint	s3-region1.cisco.cloudian.local	80, 443	Default s3 service endpoint that should resolve to the load balancer
s3 Wildcard Service Endpoint	*.s3-region1.cisco.cloudian.local	80, 443	Wildcard s3 service endpoint that should resolve to the load balancer

s3 Admin Service Endpoint	s3-admin.cisco.cloudian.local	19443	Admin service endpoint that should resolve to the load balancer
Cloudian Management Console	cmc.cisco.cloudian.local	8888, 8443	CMC service endpoint that should resolve to the load balancer



The Load Balancer should forward the traffic to all HyperStore nodes in the data center in a round-robin method. The traffic for the Cloudian Management Console (CMC) should be configured with sticky sessions enabled.

Prepare the Master Node

The master node will be used to push binaries and configurations to the nodes. The basic directories need to be created and the system_setup script needs to be downloaded.

To prepare the master node, follow these steps:

1. Create folders for Cloudian Installation.

```
# mkdir -p /root/CloudianTools /root/CloudianPackages/
```

2. Download and execute HyperStore system_setup script.

```
# cd /root/CloudianTools/ && yum install -y wget && wget
https://s3.cloudianhyperstore.com/downloads/Scripts/system_setup.sh && chmod +x
system_setup.sh && ./system_setup.sh
```

3. Select "D" - "Download HyperStore Files."

```
# System Setup
    1) Configure Networking
    2) Change Timezone
    5) Change root Password
    B) BMC Configuration
    D) Download HyperStore Files
        Please Download or place the HyperStore files in '/root/CloudianPackages'
    S) Script Settings
    A) About system_setup2.sh
    X) Exit
```

4. Select "EA" Version.

```
System Setup » HyperStore Downloader
Downloading HyperStore Version Information ... Done
```

```
Which HyperStore release would you like to download? (v6-GA/GA/FTP/EA) EA
Downloading HyperStore Binary v7.1.2 ... Done
Downloading HyperStore Binary v7.1.2 (md5) ... Done
Downloading HyperStore Documentation v7.1.2 ... Done
Downloading HyperStore Documentation v7.1.2 (md5) ... Done
Downloading HyperStore Release Notes ... Done
Downloading HyperStore Installation License ... Done
Press any key to continue ...
```

5. Once the HyperStore binary is downloaded exit the script and extract the binary using the by Cloudian provided License file (.lic).

```
#!/CloudianHyperStore-7.1.2.bin cloudian_289001406012.lic
Extracting package contents for installation...
Extraction completed.
*** Cloudian HyperStore(R) Cloud Storage System ***
*** Checking required packages: Oracle Java jdk-1.8.0_172, Puppetserver (JVM)
1.2.0, Puppet 3.8.7, Facter 2.4.6, Python 2.7.8, Ruby ***, bind-utils

The Cloudian HyperStore install script will now install: Java, Puppet 3.8.7,
Puppet-server 3.8.7, Python 2.7.8, facter 2.4.6, puppetserver 1.2.0, bind_utils

Self Extracting Installer

*** Running Installer for Cloudian Pre-requisite packages ***
*** Completed Installation of Cloudian Pre-requisite packages ***

Unpackaging Cloudian configuration files...

Creating Puppet configuration root directory /etc/cloudian-7.1.2-puppet ...
Successfully created Puppet configuration root directory /etc/cloudian-7.1.2-
puppet.

Default templates stored for future upgrades in
/root/CloudianPackages/orig_templates/cloudian-7.1.2-puppet.tar.gz.

Default csv's stored for future upgrades in
/root/CloudianPackages/orig_csvs/cloudian-7.1.2-puppet-csvs.tar.gz.

To install Cloudian HyperStore software:

    1. Compose a network survey file. A sample survey file, sample-survey.csv,
       is provided for your reference.

    2. Run cloudianInstall.sh

Your staging directory is /root/CloudianPackages
```

Network Best Practices

It is a best practice to create a network for client access and cluster communication. The interfaces can be bonded and be configured on separate VLANS when desired. The internal cluster communication interface should not be used as the interface for default routing.

Cloudian supports the following bonding options:

- Balanced Round Robin
- Active Backup
- Balance XOR
- Broadcast
- 802.3ad
- Balance TLB
- Balance ALB



This configuration is not necessary for this CVD



Cloudian recommends using a MTU size of 1500 if object storage is exposed to Internet. A MTU size of 9000 can be used if the cluster is not serving clients over the Internet and the entire network infrastructure including load balancers have been adjusted accordingly.

HyperStore nodes sometimes communicate with each other via JMX, and when they do, after initial connection establishment on the designated JMX a random port is used for continued communication. Therefore, there cannot be any port restrictions on communication between HyperStore nodes. Consequently, the HyperStore installation will abort if firewall, SELinux, or iptables is running on a host.



The ports listed in Table 7 (italicized) should be exposed to public traffic.

Table 7 Overview of HyperStore Network Ports

Service	Listening Port	Interface(s) Bound To	Purpose
Clouidian Management Console (CMC)	8888	All	Requests from administrators' or end users' browsers via HTTP
	8443	All	Requests from administrators' or end users' browsers via HTTPS
S3 Service	80	All	Requests from the CMC or other S3 client applications via HTTP
	443	All	Requests from the CMC or other S3 client applications via HTTPS
	81	All	Requests relayed by an HAProxy load balancer using the PROXY Protocol (if enabled by configuration; see s3_proxy_protocol_enabled in common.csv)
	4431	All	Requests relayed by an HAProxy load balancer using the PROXY Protocol with SSL (if enabled by configuration)
	19080	All	JMX access
IAM Service	16080	All	Requests from the CMC or other IAM clients via HTTP
	16443	All	Requests from the CMC or other IAM clients via HTTPS
	19084	All	JMX access
Admin Service	18081	All	Requests from the CMC or other Admin API clients via HTTP
	19443	All	Requests from the CMC or other Admin API clients via HTTPS (Note: The CMC by default)

Create the survey.csv File

The survey.csv file is used to identify the nodes that will be used for installing the HyperStore cluster. The survey.csv file includes the following information for each node:

- Region name
- Hostname
- IP that resolves to the hostname
- Datacenter name
- Rack name

- Interface name for internal cluster communication

As the interface used for internal cluster communication has to be defined in the survey.csv file the correct interface name has to be verified.

To create the survey.csv file, follow these steps:

1. Verify interface name for internal cluster network.

```
ifconfig
eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.100.240 netmask 255.255.255.0 broadcast 192.168.100.255
    inet6 fe80::225:b5ff:fe00:a001 prefixlen 64 scopeid 0x20<link>
    ether 00:25:b5:00:a0:01 txqueuelen 1000 (Ethernet)
    RX packets 1306588 bytes 105079254 (100.2 MiB)
    RX errors 0 dropped 10074 overruns 0 frame 0
    TX packets 176583 bytes 12889163 (12.2 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.130.240 netmask 255.255.255.0 broadcast 192.168.130.255
    inet6 fe80::225:b5ff:fe00:a002 prefixlen 64 scopeid 0x20<link>
    ether 00:25:b5:00:a0:02 txqueuelen 1000 (Ethernet)
    RX packets 422887 bytes 44997625 (42.9 MiB)
    RX errors 0 dropped 10074 overruns 0 frame 0
    TX packets 28 bytes 2128 (2.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

2. Once the interface name for internal cluster communication has been identified the system_setup script can be executed to create the survey.csv file.

```
System Setup
  1) Configure Networking
  2) Change Timezone
  3) Setup Disks
  4) Setup Survey.csv File
     Survey File '/root/CloudianPackages/survey.csv' Not Found
  5) Change root Password
  6) Install & Configure Prerequisites
  9) Prep New Node to Add to Cluster
  B) BMC Configuration
  S) Script Settings
  A) About system_setup.sh
  X) Exit
Choice: 4) Setup Survey.csv File
```

3. Add the correct hostname, desired region name, datacenter name, rack name and interface for internal cluster communication. Add additional entries as needed.

```
System Setup » Survey File
  Using '/root/CloudianPackages/survey.csv'

  C) Create New File

  P) Return to the Previous Menu

Choice: c
System Setup » Survey File » Create Survey File

Creating Directory '/root/CloudianPackages' ... Already Exists
```

```

Creating File '/root/CloudianPackages/survey.csv' ... Done

Would you like to add entries now? (Yes/No) Yes
System Setup » Survey File » Add Entry

No Entries Found
Region  Hostname  IP Address  Datacenter  Rack  Interface

Lines in red are commented out in the survey file.

Region Name: region1
Hostname: storage-node1
Attempting auto IP resolution for storage-node1 ... Done
IP Address: 192.168.100.240
Data Center Name: DC1
Rack name: Rack1
Internal Interface (optional): eth2

Adding entry to /root/CloudianPackages/survey.csv ... Done

Would you like to add another entry? (Yes/No) [Yes]

```

4. Once completed the survey.csv will look similar like this:

```

region1,storage-node1,192.168.100.240,DC1,Rack1,eth2
region1,storage-node2,192.168.100.241,DC1,Rack1,eth2
region1,storage-node3,192.168.100.242,DC1,Rack1,eth2
region1,storage-node4,192.168.100.243,DC1,Rack1,eth2
region1,storage-node5,192.168.100.244,DC1,Rack1,eth2
region1,storage-node6,192.168.100.245,DC1,Rack1,eth2

```



To install multiple Datacenters with the initial installation, make sure to correctly specify the “**Datacenter Name**” in the fourth tab of the survey.csv.

Prepare Cluster Nodes

Now the survey.csv file is completed and the IP communication between the nodes has been established and verified, the prerequisites can be installed on each node by running option “6” from the system_setup script.

System Setup

6) Install & Configure Prerequisites

Next the script will ask to provide the root password for each node and will setup ssh certificates.

```

Would you like to perform this on all nodes listed in your survey file? (Yes/No) Yes

```

If your root password is the same on all (or most) nodes in the cluster, you can supply it as a cluster password

If you do not want to supply a password, each server will prompt for one when connecting.

```

Cluster Password:

```

Once the prerequisites have been successfully installed it's time to format and mount the data drives on each cluster node. As part of the prerequisites installation the system_setup script has been placed under /root/CloudianTools/system_setup.sh for each node.

To prepare the cluster nodes, follow these steps:

1. Log into each node and execute /root/CloudianTools/system_setup.sh to configure the hostname, domain name and configure additional network interfaces, bonds and VLANS if you have not done so already.
2. Select "1" Configure Networking.

System Setup

1) Configure Networking

3. Select "1" to "4" to adjust network interfaces as needed, VLANS and Bondings can be created easily as well. Once finished "D" to set domain name.

System Setup » Networking

Speed	Interface	IP Address	State	Type	Mode	Master
Gb/s	1) eth0	173.36.220.240/24	Up	Ethernet	--	-- 40
Gb/s	2) eth1	fe80::225:b5ff:fe00:a000/64 192.168.100.240/24	Up	Ethernet	--	-- 40
Gb/s	3) eth2	fe80::225:b5ff:fe00:a001/64 192.168.130.240/24	Up	Ethernet	--	-- 40
Gb/s	4) eth3	fe80::225:b5ff:fe00:a002/64 192.168.120.240/24	Up	Ethernet	--	-- 40
	5) lo:1	fe80::225:b5ff:fe00:a003/64 192.168.100.100/32	Down	Ethernet	--	-- --

Select a number from the list above to edit an interface's configuration

- | | |
|---------------------------------|------------------------------------|
| D) Change Domain Name (<unset>) | H) Change Hostname (storage-node1) |
| B) Create Bond Interface | V) Create VLAN Interface |
| N) Restart Networking | R) Refresh Interface Details |

The data drives can be formatted and mounted automatically by running option "3" of the system_setup script individually on each node and selecting the drives that are going to be used to store data.

4. Select option "3"

System Setup

- 1) Configure Networking
- 2) Change Timezone
- 3) Setup Disks
- 4) Setup Survey.csv File
- 5) Change root Password
- 6) Install & Configure Prerequisites
- 7) Run Commands on each Cluster Node
- 8) Copy Local File to each Cluster Node
- 9) Prep New Node to Add to Cluster

- B) BMC Configuration
- R) Run Pre-installation Checks
- S) Script Settings
- A) About system_setup2.sh
- X) Exit

5. Select the drives that are to be used for data storage

System Setup » Setup Disks

Selected Disks: sda sdaa sdab sdb sdc sdd sde sdf sdg sdh sdi sdj sdk sdl sdm sdn sdo sdp sdq sdr sds sdt sdu sdv sdw sdx sdy sdz

Device	Size	Dependencies	Device	Size	Dependencies
1) sda	10.9T	0	2) sdaa	10.9T	0
3) sdab	10.9T	0	4) sdac	222.6G	5
5) sdb	10.9T	0	6) sdc	10.9T	0
7) sdd	10.9T	0	8) sde	10.9T	0
9) sdf	10.9T	0	10) sdg	10.9T	0
11) sdh	10.9T	0	12) sdi	10.9T	0
13) sdj	10.9T	0	14) sdk	10.9T	0
15) sdl	10.9T	0	16) sdm	10.9T	0
17) sdn	10.9T	0	18) sdo	10.9T	0
19) sdp	10.9T	0	20) sdq	10.9T	0
21) sdr	10.9T	0	22) sds	10.9T	0
23) sdt	10.9T	0	24) sdu	10.9T	0
25) sdv	10.9T	0	26) sdw	10.9T	0
27) sdx	10.9T	0	28) sdy	10.9T	0
29) sdz	10.9T	0			

- C) Configure Selected Disks
- T) Toggle Selection for all disks
- R) Refresh Disks
- P) Return to the Previous Menu

6. Alternatively, the drives can remotely formatted from the master node with the command below, please ensure to grep for the correct disk size.

```
for i in {1..6}; do ssh -t -i /root/CloudianPackages/cloudian-installation-key storage-node${i} "/root/CloudianTools/system_setup.sh --configure-disks $(lsblk -d | grep 10.9T |awk '{print $1}' |sed -r -e ':a;N;${ba;s~\n~ ~g' } <<<'y'"; done
```

7. Verify that all disks are correctly mounted on all nodes.

```
for i in {1..6}; do ssh -t -i /root/CloudianPackages/cloudian-installation-key storage-node${i} "df -h |grep -c /cloudian"; done
```

```
28
Connection to storage-node1 closed.
28
Connection to storage-node2 closed.
28
Connection to storage-node3 closed.
28
Connection to storage-node4 closed.
28
```

```
Connection to storage-node5 closed.  
28  
Connection to storage-node6 closed.
```

Cloudian HyperStore Installation

Software Installation

Once verified that all drives on all nodes have been successfully mounted, the pre-installation check should be run to ensure all requirements for the installation have been met and there are no conditions that would cause the installation to fail.

To run the pre-installation check, follow this step:

1. The pre-installation check can be run from the `/root/CloudianTools/systemsetup.sh` and selecting option "R" Run Pre-installation Checks:

```
System Setup
```

```
    R) Run Pre-installation Checks
```

```
System Setup » Pre-installation Checklist
```

```
OK   found survey file "/root/CloudianPackages/survey.csv"
OK   All 1 Datacenter(s) contain a minimum of 3 nodes
OK   entry found in hosts file for node storage-nodel
```

```
Total checks performed: 404. Warnings: 0, Errors: 0
```

```
Press any key to continue ...
```



The pre-installation check should finish without errors or warnings. If errors are encountered, the generated output should give more information about the error. Do not proceed with the installation until all the errors are resolved.

After the pre-installation check runs successfully and no errors are found, the Cloudian HyperStore installer can be executed.

To run the installer, follow these steps:

1. The installer can be executed with the "-s" option to specify the survey.csv file name, not using this option will prompt you to enter the correct survey file name.

```
./cloudianInstall.sh -s survey.csv
```

```
Cloudian HyperStore(R) 7.1.2 Installation/Configuration
```

```
-----
0 ) Run Pre-Installation checks
1 ) Install Cloudian HyperStore
2 ) Cluster Management
3 ) Upgrade From a Previous Version
4 ) Advanced Configuration Options
5 ) Uninstall Cloudian HyperStore
6 ) Help
x ) Exit
```



The Cloudian HyperStore installer provides multiple usage options that can be listed by executing help; `./cloudianInstall -h`

2. Select option “1” Install Cloudian HyperStore and answer “yes” to use the Cloudian-installation-key that was already created.

Setup Access to Hosts in Cluster

```
-----
Processing cluster host information in survey.csv file.
Connectivity check to all (6) hosts defined in survey file.
Able to ping all 6 hosts defined in survey.csv file.
Check and setup password-less SSH access to hosts.
Would you like to use key ./cloudian-installation-key? (yes/no) [yes]:
Installation key cloudian-installation-key is now being copied to all nodes ...
Installation key ./cloudian-installation-key.pub copied to all agent hosts
successfully.
Will now copy key cloudian-installation-key to this host storage-nodel.
Password-less SSH access to hosts setup.
Installation requirements check on hosts defined in survey file survey.csv.
Installing prerequisite packages on agent node 192.168.100.240. This could take
a minute.
Installing prerequisite packages on agent node 192.168.100.241. This could take
a minute.
Installing prerequisite packages on agent node 192.168.100.242. This could take
a minute.
Installing prerequisite packages on agent node 192.168.100.243. This could take
a minute.
Installing prerequisite packages on agent node 192.168.100.244. This could take
a minute.
Installing prerequisite packages on agent node 192.168.100.245. This could take
a minute.
```

3. Next, the installer asks for a default interface for the internal cluster communication, the interface that was already entered in the survey.csv file will take precedence over the default internal interface. The default internal interface is only used when no interface is defined in the survey.csv file.

Configure cluster

```
-----
Select only one from this list of known interfaces: eth0,eth1,eth2,eth3,lo:1.
Leave it blank you wish to use the default network interface.
```

```
    Please enter one of the interface names [eth0,eth1,eth2,eth3,lo:1] for
internal services: []: eth2
    Using eth2 for all internal traffic.
```

4. Provide the Top Level Domain for that will be used by the cluster.

```
Cloudian HyperStore(R) S3 service endpoints are based on your desired top
level DNS domain name. For example, yourcompany.com.
Please enter your top level domain name [cisco.cloudian.local]:
```

```
Region [region1] Cassandra cluster name: Cloudianregion1
```

5. Keep the metadata replication strategy at 3 by accepting the default value or specifying “DC1:3”

```
Please enter the service metadata replication strategy for region1 [DC1:3]:
```

6. Enter a local NTP time source or use an external NTP server.


```
NTP time server(s) for region region1:
Please enter your NTP time server(s)
[0.centos.pool.ntp.org,1.centos.pool.ntp.org,2.centos.pool.ntp.org,3.centos.pool.
ntp.org]:
```

```
NTP time server(s) for region :
0.centos.pool.ntp.org,1.centos.pool.ntp.org,2.centos.pool.ntp.org,3.centos.pool.n
tp.org
```

7. Accept default entries for the service endpoints based on the Top Level Domain or enter a custom endpoint name for s3 service, s3-website, s3-admin and CMC.

```
Service endpoints for region region1:
Region [region1] S3 service domain URLs(comma separated) [s3-
region1.cisco.cloudian.local]:
Region [region1] S3 Web site end point [s3-website-region1.cisco.cloudian.local]:
Admin endpoint [s3-admin.cisco.cloudian.local]:
S3 Admin service endpoint: s3-admin.cisco.cloudian.local
Domain name of your Cloudian Management Console service
[cmc.cisco.cloudian.local]:
Cloudian Management Console service endpoint: cmc.cisco.cloudian.local
```

Once the installation has completed successfully, it will display the predefined CMC url to manage the cluster.

<http://cmc.cisco.cloudian.local:8888>

Generate HTTPS Certificate and Signing Request

By default, Cloudian HyperStore is configured for HTTP access only, HTTPS can be setup by generating a self-signed certificate that in parallel will also create a Certificate Signing Request (CSR) in the same directory, if your Keystore file is named cloudian.jks for example, then the CSR file will be named cloudian.csr.

To Generate a Certificate and a Certificate Signing Request, follow these steps:

1. Select option "4" Advanced Configuration Options from the installation menu.

```
Cloudian HyperStore(R) 7.1.2 Installation/Configuration
-----
```

```
0 ) Run Pre-Installation checks
1 ) Install Cloudian HyperStore
2 ) Cluster Management
3 ) Upgrade From a Previous Version
4 ) Advanced Configuration Options
5 ) Uninstall Cloudian HyperStore
6 ) Help
x ) Exit
```

2. Select option "e" Generate a self-signed certificate in a JKS keystore.

```
Advanced Configuration Options
-----
```

```
a ) Change server role assignments
b ) Change S3, Admin and CMC ports
c ) Change S3, S3-Website, Admin, or CMC endpoints
d ) Configure diagnostic data collection options
e ) Generate a self-signed certificate in a JKS keystore
f ) Enable and configure HTTPS access on S3 server [OK]
```

- g) Import Java keystore to CMC
- h) Remove existing Puppet SSL certificates
- i) Start or stop Puppet daemon
- j) Remove Puppet access lock
- k) Enable or disable DNSMASQ
- l) Configure Performance Parameters on Nodes
- m) Generate a self-signed certificate for IAM in a JKS keystore
- n) Enable and configure HTTPS access for IAM
- r) Exclude host(s) from configuration push and service restarts
- x) Return to Main Menu

3. Provide the keystore name and password to use.

Generate a self-signed certificate in a JKS keystore

```
-----
Generating self-signed certificate for region region1

Please enter key store name [cloudian.jks]:

If you plan to store multiple certificates in your key store, you must provide
an alias for each certificate stored.

Please enter alias name []: cloudians3
Please enter the password for cloudian.jks [testpass]:
Please enter the key store manager password for cloudian.jks [testpass]:
```

4. Provide the wildcard domain name(s) that you want to use for the certificate and complete the organization identity fields.

```
Common name is the URL(FQDN or IP address) for SSL connection. For S3 service
bucket name is a part of the FQDN. You will need to generate and verify the
certificate as a wildcard. For example, *.s3.cloudian.com.

Please enter comma-separated domain names [*s3-region1.cisco.cloudian.local]:
Enter your organizational unit name []: cisco
Enter the name of your organization []: cisco-cloudian
Enter the name of your City or Locality []: San Jose
Enter the name of your State or Province []: CA
Enter the two-letter country code for this unit []: US
```

Certificate generated for region region1.

5. CSR location for this example.

```
/etc/cloudian-7.1.2-puppet/modules/baselayout/files/cloudian.csr
```



When intending to use an official certificate submit the generated CSR file to your preferred Certificate Authority for signing, using the instructions from the CA.

Import SSL Certificate in Keystore



If you're using a self-signed certificate, this step can be skipped.

To import SSL certificates in keystore, follow these steps:

1. Copy all the certificates that you received from the CA into the `/etc/cloudian-7.1.2-puppet/modules/baselayout/files` directory

- From the same directory, Issue the following commands to import the Root CA Certificate and Intermediate CA Certificate into your Keystore file.

Example for GoDaddy as the CA:

```
/usr/java/default/bin/keytool -import -trustcacerts -alias root -file <Root CA Certificate File> -keystore cloudian.jks
/usr/java/default/bin/keytool -import -trustcacerts -alias intermediate -file <Intermediate CA Certificate File> -keystore cloudian.jks
/usr/java/default/bin/keytool -import -trustcacerts -alias GoDaddyRoot -file gdrootg2_cross.crt -keystore cloudian.jks
/usr/java/default/bin/keytool -import -trustcacerts -alias GoDaddyCrossCA -file gd_cross_intermediate.crt -keystore cloudian.jks
/usr/java/default/bin/keytool -import -trustcacerts -alias GoDaddyG2CA -file gdig2.crt -keystore cloudian.jks
```

- Issue the following command to import your CA-signed TLS/SSL Certificate into your Keystore file

```
[files]# /usr/java/default/bin/keytool -import -trustcacerts -alias cloudians3 -file cloudianS3.crt -keystore cloudian.jks
```

Enable HTTPS Access on s3

Once the self-signed or CA-signed certificate has been created and imported to the keystore, HTTPS can be enabled for s3.

To enable HTTPS access on s3, follow these steps:

- Select option “f” Enable and configure HTTPS access on S3 server from the Advanced Configuration Options menu from within the installer.

Advanced Configuration Options

```
-----
a ) Change server role assignments
b ) Change S3, Admin and CMC ports
c ) Change S3, S3-Website, Admin, or CMC endpoints
d ) Configure diagnostic data collection options
e ) Generate a self-signed certificate in a JKS keystore
f ) Enable and configure HTTPS access on S3 server
g ) Import Java keystore to CMC
h ) Remove existing Puppet SSL certificates
i ) Start or stop Puppet daemon
j ) Remove Puppet access lock
k ) Enable or disable DNSMASQ
l ) Configure Performance Parameters on Nodes
m ) Generate a self-signed certificate for IAM in a JKS keystore
n ) Enable and configure HTTPS access for IAM
r ) Exclude host(s) from configuration push and service restarts
x ) Return to Main Menu
```

- Follow the onscreen instructions to enable HTTPS, make sure to provide the correct key store name, password and alias.

Enable and configure HTTPS access on S3 server

```
-----
```

HTTPS access to Clouidian HyperStore(R) S3 server is not enabled.
 Do you wish to enable HTTPS access on S3 server? (yes/no) [no]: yes
 HTTPS on Clouidian HyperStore(R) S3 server is enabled.

The key store is the file name of your identity store that will contain the server private key and corresponding server public certificate (self-signed or CA verified).

Please enter name of your key store [cloudian.jks]:
 Certificate alias to use []: cloudians3

The trust keystore is the file name of the identity store that will contain the client certificates for SSL mutual authentication. If not set, the system will look for client certificates in the keystore.

Please enter trust keystore [cloudian.jks]:

Passwords are obfuscated in configuration files. If the password your enter is not prefixed with 'OBF:', then password obfuscation is automatically performed using Jetty utility.

Please enter password for the keystore cloudian.jks
 [OBF:lytcl1vu9lv2ply83ly7vlv1plvv1lyta]:
 Please enter password for the trust keystore cloudian.jks
 [OBF:lytcl1vu9lv2ply83ly7vlv1plvv1lyta]:
 Please enter keystore manager password [OBF:lytcl1vu9lv2ply83ly7vlv1plvv1lyta]:

Please enter path name in which to store keystore file [/opt/cloudian/conf]:
 Please enter path name in which to store trust keystore file
 [/opt/cloudian/conf]:

Please enter connection maximum idle time in (ms) [60000]:
 Please enter connections at which system is considered to have low resource [1000]:
 Please enter low resource connection idle time in (ms) [5000]

3. To complete the HTTPS configuration the changes have to be pushed out to all cluster nodes using puppet. From the main installer menu select option "2" Cluster Management.

Clouidian HyperStore(R) 7.1.2 Installation/Configuration

- ```

0) Run Pre-Installation checks
1) Install Clouidian HyperStore
2) Cluster Management
3) Upgrade From a Previous Version
4) Advanced Configuration Options
5) Uninstall Clouidian HyperStore
6) Help
x) Exit

```

4. Select option "b" Push Configuration Settings to Cluster.

Cluster Management

- ```

-----
a ) Review Cluster Configuration
    
```

- b) Push Configuration Settings to Cluster
- c) Manage Services
- d) Run Validation Tests
- x) Return to Main Menu

5. Select default empty value to send configuration to all nodes in the cluster.

Run Puppet to configure agent nodes

region region1 contains the following hosts: storage-node2 storage-node3 storage-node4 storage-node5 storage-node6 storage-node1
Enter a comma-separated list of hosts in region1 to execute agents on? [empty for all] []:

All Puppet agent runs completed successfully in region1 region.
Puppet agent run ended for region1.

Press any key to continue ...

6. The s3 service has to be restarted on all nodes. From within the Cluster Management menu, select option "c" Manage Services.

Cluster Management

- a) Review Cluster Configuration
- b) Push Configuration Settings to Cluster [OK]
- c) Manage Services
- d) Run Validation Tests
- x) Return to Main Menu

7. From within the Service Management Menu, select option "5" S3 Service and enter "restart".

Service Management

- 0) All services
- 1) Redis Credentials
- 2) Redis QOS
- 3) Cassandra
- 4) HyperStore service
- 5) S3 service
- 6) Redis Monitor
- 7) Cloudian Agent
- 8) DNSMASQ
- 9) Cloudian Management Console (CMC)
- P) Puppet service (status only)
- X) Quit

You can execute the following list of commands:
start,stop,status,restart,version,force-stop,node-start,node-stop

Select a service to manage: 5

Enter command: (start,stop,status,restart,version) restart
Executing Cloudian S3 service command restart ...

On host storage-node2:

```
/etc/init.d/cloudian-s3 restart => Restarting cloudian-s3 (via systemctl): [ OK ]
```

On host storage-node3:

```
/etc/init.d/cloudian-s3 restart => Restarting cloudian-s3 (via systemctl): [ OK ]
```

On host storage-node4:

```
/etc/init.d/cloudian-s3 restart => Restarting cloudian-s3 (via systemctl): [ OK ]
```

On host storage-node5:

```
/etc/init.d/cloudian-s3 restart => Restarting cloudian-s3 (via systemctl): [ OK ]
```

On host storage-node6:

```
/etc/init.d/cloudian-s3 restart => Restarting cloudian-s3 (via systemctl): [ OK ]
```

On host storage-node1:

```
/etc/init.d/cloudian-s3 restart => Restarting cloudian-s3 (via systemctl): [ OK ]
```

Press any key to continue ...



For more information about how to Install Cloudian HyperStore, please refer to the Cloudian [HyperStore Installation Guide](#).

Cloudian HyperStore Configuration


Log into the Cloudian Management Console (CMC)

To log into the CMC, point a web browser to the predefined CMC URL on HTTP port 8888 or HTTPS port 8443, and follow these steps:

<http://cmc.cisco.cloudian.local:8888>

<https://cmc.cisco.cloudian.local:8443>

 When using HTTP the browser will be redirected to the HTTPS port for secure login.

 In addition to the predefined CMC URL, all IP addresses of all nodes can be used to access the CMC.

1. Log in with the default admin credentials.

Group Name: System Admin
User ID: admin
Password: public



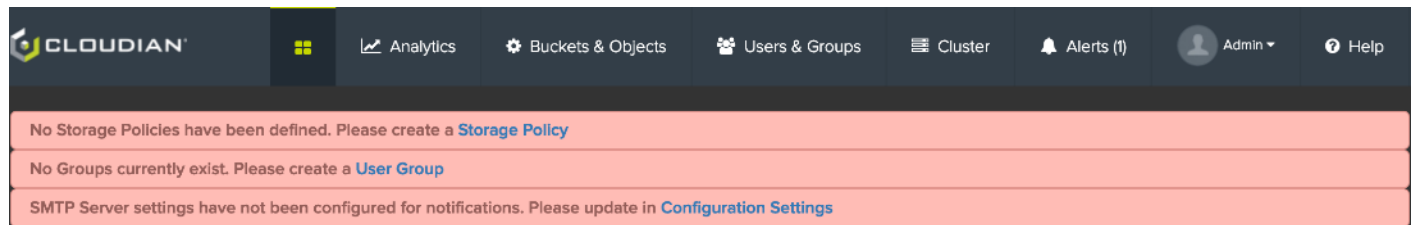
SIGN IN

Group Name:

User ID:

Password:

2. Once logged in to the CMC, the system needs to be configured with one or more Storage Policies, Groups and Users and settings to enable SMTP notifications.



Create a Storage Policy

To create a storage policy, follow these steps:

1. Click the pink bar with the reference no storage policies have been created yet or alternatively go to the “Cluster” tab followed by the “Storage Policies” tab and click “CREATE STORAGE POLICY.”
2. Provide a name for the Storage Policy in the “Policy Name” field, followed by a description when desired. In this setup the cluster has a single datacenter (DC) and exists out of 6 nodes, which provides the option to use erasure code 4+2 which will be used for this example.
3. To use erasure code 4+2 as the protection scheme for this storage policy, select “EC Within Single Datacenter” under “DATA DISTRIBUTION SCHEME”. Next select the Erasure Code value from the dropdown list under “ERASURE CODING K+M VALUE.”

The screenshot shows the Cloudian HyperStore interface for creating a new storage policy. The top navigation bar includes 'Cluster', 'Alerts (1)', 'Admin', and 'Help'. Below the navigation bar, there are tabs for 'Data Centers', 'Nodes', 'Cluster Config', 'Storage Policies', 'Repair Status', and 'Operation Status'. The 'Storage Policies' tab is active, and a '+ CREATE STORAGE POLICY' button is visible. The 'CREATE NEW POLICY' form contains the following fields and options:

- Policy Name:** ec42
- Policy Description:** Policy Description
- NUMBER OF DATACENTERS:** 1
- DATA DISTRIBUTION SCHEME:**
 - Replicas Within Single Datacenter
 - EC Within Single Datacenter
- ERASURE CODING K+M VALUE:** 4+2

4. Select the desired consistency level for the erasure code Storage Policy. The consistency level of “QUORUM” will provide strong consistency as 5 nodes out of 6 have to acknowledge the write before the operation gets acknowledged by the client. For reads 4 nodes have to respond to get sufficient chunks to process the read request.
5. By default, created storage policies are available to each group/tenant. When a group is specified the storage policy will only be visible to the defined group(s).

- Storage policies can be configured with the compression algorithms below, please be aware that CPU cycles will be wasted when the data is placed in a bucket using storage policy with compression enabled but that data is not compressible.

Supported Compression Algorithms:

- SNAPPY
- ZLIB
- LZ4

- Encryption at rest can be enabled and forced at the bucket level by setting the “Server-side Encryption box to “SSE”.

DATACENTER ASSIGNMENT

REGION	DATACENTER	REPLICA	LOCAL EC
region1	DC1	1 of 1	4+2

CONSISTENCY SETTING

CONSISTENCY LEVEL	READ	WRITE
ALL	<input type="checkbox"/>	<input type="checkbox"/>
QUORUM	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

GROUP VISIBILITY

ADD

Compression Type

Server-side Encryption



The first Storage Policy that is created will be the default Storage Policy for all Users and Groups. The default Storage Policy can be changed when multiple Storage Policies exist on the Cluster. For more information about setting up Storage Policies in HyperStore, please refer to the [HyperStore Installation Guide](#).



Storage Policies can also be created through the admin API, please see the Admin API section of the [HyperStore Installation Guide](#).

HyperStore supports multiple storage policies on the same hardware, the type of storage policies you can create depends on the number of nodes and the DC’s in the cluster.

- To create an additional storage policy using 3-way replication, use the “CREATE STORAGE POLICY” button and select “Replicas Within Single Datacenter” under “DATA DISTRIBUTION SCHEME” and enter the number of Replicas.

The screenshot shows the 'CREATE NEW POLICY' form in the Cloudian HyperStore interface. The navigation bar at the top includes 'Analytics', 'Buckets & Objects', 'Users & Groups', 'Cluster', 'Alerts (21)', 'Admin', and 'Help'. Below the navigation bar, there are tabs for 'Data Centers', 'Nodes', 'Cluster Config', 'Storage Policies', 'Repair Status', and 'Operation Status'. The main content area is titled 'STORAGE POLICIES' and includes a '+ CREATE STORAGE POLICY' button.

The 'CREATE NEW POLICY' form contains the following fields and options:

- Policy Name:** A text input field containing 'rf3'.
- Policy Description:** A text input field containing 'Policy Description'.
- NUMBER OF DATACENTERS:** A text input field containing '1'.
- DATA DISTRIBUTION SCHEME:** Two radio button options:
 - Replicas Within Single Datacenter:** Selected. The diagram shows an 'OBJECT' being replicated into three 'REPLICA' blocks within a single 'DC-1' datacenter.
 - EC Within Single Datacenter:** Unselected. The diagram shows an 'OBJECT' being broken into 'EC FRAGMENTS' within a single 'DC-1' datacenter.
- NUMBER OF REPLICAS:** A text input field containing '3'.



The minimum amount of data replicas is 3, this is to protect the data and ensure there is always a quorum.

9. Select the desired consistency setting, group restriction, compression and encryption and click Save to create the replication storage policy.

CONSISTENCY SETTING		
CONSISTENCY LEVEL	READ	WRITE
ALL	<input type="checkbox"/>	<input type="checkbox"/>
QUORUM	<input type="checkbox"/>	<input checked="" type="checkbox"/>
ONE	<input checked="" type="checkbox"/>	<input type="checkbox"/>

GROUP VISIBILITY

Please select a Group ADD

Compression Type

Server-side Encryption

SAVE

CANCEL



For increased read performance, select a Quorum of “ONE” for read. Please note that this comes at the cost of having a small window of time where potentially old data can be read.



For more information about setting up Storage Policies in HyperStore, please refer to the [HyperStore Installation Guide](#).

Setup Alerts and Notifications

Cloudian HyperStore should be configured to send out alerts and notification for events to ensure proper action can be taken in a timely fashion. Cloudian HyperStore supports alerts through SMTP and SNMP. To create alert rules the SMTP email settings and/or SNMP server details have to be completed.

To setup SMTP email details, follow these steps:

1. Go to “Cluster” then “Cluster Config” followed by the “CONFIGURATION SETTINGS” tab and complete the “SMTP/Email Settings for Alerts/Notifications.

The screenshot shows the Cloudian HyperStore configuration interface. The top navigation bar includes 'Analytics', 'Buckets & Objects', 'Users & Groups', 'Cluster', 'Alerts', 'Admin', and 'Help'. Below this, a secondary navigation bar has 'Data Centers', 'Nodes', 'Cluster Config', 'Storage Policies', 'Repair Status', and 'Operation Status'. The main content area is divided into 'CLUSTER INFORMATION' and 'CONFIGURATION SETTINGS'. Under 'CONFIGURATION SETTINGS', there is a section for 'SMTP/Email Settings for Alerts/Notifications' with the following fields:

SMTP Server FQDN	smtp.notification.cloudian.local	Cancel
SMTP Port	465	Edit
SMTP Protocol	smtps	Edit
SMTP Enable STARTTLS	No	Edit
SMTP From Address	noreply@smtp.notification.cloudian.local	Cancel
Notification Message Subject Header	Cloudian HyperStore Alert	Edit
Default Email Address to Receive Notifications	alerts@cloudian.local	Cancel
SMTP Service Requires Authorization	<input checked="" type="radio"/> Yes <input type="radio"/> No	Cancel
User Name for SMTP Server	admin	Cancel
Password for SMTP Server	•••••	Edit

At the bottom of the configuration section, there is a button labeled 'Send Test SMTP Notification'.

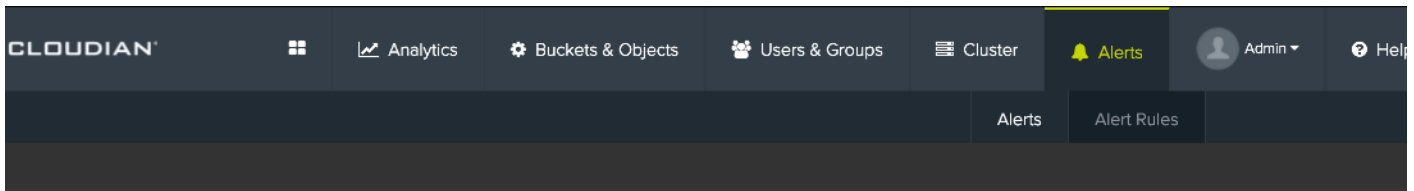
2. To setup SNMP, complete the “SNMP Trap Destination Settings.”

▼ SNMP Trap Destination Settings

Destination IP Address	10.20.142.71	Edit
Destination Port	162	Edit

Once the SMTP and/or SNMP details have been configured, alert rules can be created to trigger notification events.

- To setup an alert rule, go to “Alerts” followed by “Alert Rules”, select an “Alert Type” from the dropdown list, specify the condition, severity level and the alert destination.



ALERT RULES

CREATE ALERT RULE

Alert Type

▼ Please select an item

----Network Status----

Number of Get transactions per second

Number of Put transactions per second

Throughput for GET operations

Throughput for PUT operations

Latency for GET operations

Latency for PUT operations

Network throughput (incoming)

Network throughput (outgoing)

----General Status----

Disk space available in node

Disk space available in each device

Disk Error

Node Unreachable

Load Average (5 Min)

CPU Utilization

Repair Completion Status

----Service Status----

Admin service status

USE DEFAULT EMAIL ADDRESS

SEND SNMP TRAP

Severity Level

Medium

CREATE

	send email to	send snmp trap	severity level	actions
<input type="checkbox"/> CPU Utilization greater than 90.0 %	default		Medium	Edit Delete
<input type="checkbox"/> Disk Error	default		Critical	Edit Delete
<input type="checkbox"/> Disk space available in node less than 10.0 %	default		High	Edit Delete
<input type="checkbox"/> Disk space available in each device less than 15.0 %	default		High	Edit Delete
<input type="checkbox"/> Node Unreachable	default		Critical	Edit Delete
<input type="checkbox"/> Repair Completion Status	default		Low	Edit Delete



For more information about setting up Alerts in HyperStore, please refer to the [HyperStore Installation Guide](#).

Create a Group and User

To be able to create a user, a group/tenant has to be created first. To create a group, follow these steps:

1. Click “Users & Groups” tab in the CMC, followed by the “Manage Groups” tab. Click “+NEW GROUP”, provide a name and rating plan when required.

The screenshot shows the Cloudian CMC interface. The top navigation bar includes 'Analytics', 'Buckets & Objects', 'Users & Groups' (active), 'Cluster', 'Alerts', 'Admin', and 'Help'. Below this, a secondary navigation bar has 'Manage Users', 'Manage Groups' (active), 'Rating Plan', 'Account Activity', and 'Whitelist'. The main content area is titled 'MANAGE GROUPS' and includes '+ GROUP QOS DEFAULT' and '+ NEW GROUP' links. The 'ADD NEW GROUP' section has a checked 'Active Group' option. The form fields are: 'Group Name' (group1), 'Group Description' (empty), 'Rating Plan' (Default-RP), and 'Enable LDAP Authentication' (unchecked). 'CANCEL' and 'SAVE' buttons are at the bottom right.



LDAP authentication can be enabled to authenticate users that log into the CMC and automatically create S3 credentials. Each group/tenant can connect to its own LDAP server or Active Directory forest.

For more information about creating groups in HyperStore, please refer to the [HyperStore Installation Guide](#).

Once the group has been created, you can create a user that you can add to the group.

2. To create a user, click “Manage Users” and then “+NEW USER”. Provide a User ID, select the User Type, Group Name and provide a password.

MANAGE USERS

+ USER QOS DEFAULT

+ NEW USER

ADD NEW USER

Active User

User ID: *

User Type:
 User
 Group Admin
 System Admin

Group Name: *

Password: *

Confirm Password: *

[More](#) ▾

Search For A User By ID:

Group Name

User Type

User Status



For more information about creating users in HyperStore, please refer to the [HyperStore Installation Guide](#).

3. Click “Security Credentials” to view and copy the ACCESS and SECRET key of the newly created user.

MANAGE USERS

+ USER QOS DEFAULT

+ NEW USER

Search For A User By ID:

Group Name

User Type

User Status

USER ID	GROUP NAME	USER TYPE	STATUS	ACTIONS
user1	group1	User	Active	Edit Security Credentials Set QoS View User Data Delete

4. Copy the “ACCESS KEY ID” and “SECRET KEY”, this together with the defined s3-endpoint name is what is needed to connect any S3 enabled applications with the user that was just created.

- Alternatively, the username, password and group ID can be used to log into the CMC to retrieve this information. The s3-endpoint name(s) can be found under “CLUSTER” tab and then “ClusterConfig”

User Credentials ✕

SIGN-IN CREDENTIALS

<p>USER ID: user1</p>	<p>GROUP ID: group1</p>
<p>NEW PASSWORD: <input type="password"/></p>	<p>CONFIRM PASSWORD: <input type="password"/></p>

[CHANGE PASSWORD](#)

S3 ACCESS CREDENTIALS

CREATED	ACCESS KEY ID	ACTIONS
Mar 20 2019 11:47:29 GMT-0700	6741c65fa26f6ad8d833 *	View Secret Key Inactivate Delete

* Active Access Key

[CREATE NEW KEY](#)



Users, Groups, and credentials can also be created or retrieved through the admin API. Please see the Admin API section of the [HyperStore Installation Guide](#).

Create Buckets

Once the user has been created, that user can now create buckets and place data into those buckets using any s3 enabled application. When creating a bucket, by default the “default” Storage Policy will be used as the protection scheme for that bucket.

When multiple Storage Policies are available to the user, the user can choose to assign different Storage Policies for different buckets by creating the buckets through the CMC.

To create buckets through the CMC as a user, follow these steps:

- Log into the CMC with the associated username, password, and GroupID.



SIGN IN

Group Name:

User ID:

Password:

LOGIN

2. To add a new bucket, provide a unique bucket name and select the desired Storage Policy.

BUCKETS OBJECTS + ADD NEW BUCKET

ADD NEW BUCKET

Bucket Name: Region: Storage Policy:

Storage Policy Description



As a System or Group Admin, managed users data can be viewed by searching for the user in “Users and Groups” and clicking on the “View User Data” link for the selected user.

3. When a bucket has been created, that bucket can be configured for permissions, life cycle policies, static webhosting,CCR, versioning and logging by clicking the “Properties” for that bucket.

BUCKETS OBJECTS + ADD NEW BUCKET

NAME	REGION	POLICY		
test-bucket1	region1	ec42	Properties	Delete

BUCKET PERMISSIONS	BUCKET CANNED ACL	STORAGE POLICY	LIFECYCLE POLICY	STATIC WEBSITE HOSTING	CROSS REGION REPLICATION	VERSIONING	LOGGING
NAME	DESCRIPTION	DATA DISTRIBUTION POLICY	NO OF REPLICAS	EC K+M VALUE			
ec42		Single DC	1	4 + 2			



More information about bucket properties can be found in the [HyperStore Installation Guide](#).

Verify Credentials and Service Endpoints as a User

To verify the credentials and service endpoints, follow these steps:

1. While logged in as the user, the s3 credentials can be found under the tab “Username” and then “Security Credentials” Additional keys can be created and the CMC password can be changed.
2. The Security Credentials Page also shows the “SERVICE INFORMATION” next to the s3 credentials is required to setup a connection to the Object Store.

SIGN-IN CREDENTIALS

USER ID: user1

CURRENT PASSWORD:

NEW PASSWORD:

CONFIRM PASSWORD:

CHANGE PASSWORD

S3 ACCESS CREDENTIALS

CREATED	ACCESS KEY ID	ACTIONS
Mar-20-2019 11:47 -0700	6741c65fa26f6ad8d833 [*]	View Secret Key Inactivate Delete

^{*} Active Access Key

CREATE NEW KEY

SERVICE INFORMATION

S3 ENDPOINT (HTTP): region1: s3-region1.cisco.cloudian.local:80

S3 ENDPOINT (HTTPS): region1: s3-region1.cisco.cloudian.local:443

S3 WEBSITE ENDPOINT: region1: s3-website-region1.cisco.cloudian.local


Cloudian HyperStore Installation Verification

Verify HyperStore S3 Connectivity

To verify Cloudian HyperStore is functioning properly and the connected environment is configured correctly a couple of simple tests can be run by following these steps:

1. Verify the objects can be uploaded through the CMC. Login as a user or use the “View User Data” link to go to the bucket of the user that was created earlier and select “UPLOAD FILE”.
2. Select “Add files...” in the popup window followed by Start upload.

The screenshot displays the Cloudian HyperStore CMC interface. At the top, there are tabs for 'BUCKETS' and 'OBJECTS'. Below the tabs, the 'Bucket name' is set to 'test-bucket1'. There are three main actions: 'UPLOAD FILE', '+ CREATE FOLDER', and 'SEARCH BY PREFIX'. The region is identified as 'region1 : test-bucket1'. A table header shows columns for 'NAME', 'SIZE', and 'LAST MODIFIED'. On the right side, there are 'RESTORE' and 'DELETE' buttons. An 'Upload Files' modal window is open, showing a file named 'ucs_cloudianhyperstore_s3260m5_deployment_guide_draft_eddo1.docx' with a size of 19.15 MB. The modal includes buttons for '+ Add files...', 'Start upload', 'Cancel upload', and 'Clear finished', along with a 'Store encrypted' checkbox. A progress bar and the text 'Upload completed!' are visible at the bottom of the modal.

 The default maximum size for an Object that can be uploaded through the CMC is 5GB. More information about uploading objects through the CMC can be found in the [HyperStore Installation Guide](#).

Next, a client-side test should be run to ensure that the rest of the connecting infrastructure is correctly configured to support Cloudian HyperStore.

3. Connect to any Linux distribution client server and install s3cmd. When using Centos you will need to have epel-release installed.

```
yum install -y s3cmd
```

4. Configure s3cmd to use the s3 credentials of user1 and the s3 service endpoint used by Cloudian HyperStore. To configure s3cmd follow the instructions below.

```
[root@storage-node1 ~]# s3cmd --configure
```

Enter new values or accept defaults in brackets with Enter. Refer to user manual for detailed description of all options.

Access key and Secret key are your identifiers for Amazon S3. Leave them empty for using the env variables. Access Key: 6741c65fa26f6ad8d833

Secret Key: eeUZ+PmlKtoEyNJ0dtGdXiIREE29HVh9yndIllNt

Default Region [US]: region1

Use "s3.amazonaws.com" for S3 Endpoint and not modify it to the target Amazon S3.

S3 Endpoint [s3.amazonaws.com]: s3-region1.cisco.cloudian.local

Use "%(bucket)s.s3.amazonaws.com" to the target Amazon S3. "%(bucket)s" and "%(location)s" vars can be used

if the target S3 system supports dns based buckets.

DNS-style bucket+hostname:port template for accessing a bucket
[% (bucket) s.s3.amazonaws.com]: % (bucket) s.s3-region1.cisco.cloudian.local

Encryption password is used to protect your files from reading by unauthorized persons while in transfer to S3

Encryption password: Passw0rd!

Path to GPG program [/usr/bin/gpg]:

When using secure HTTPS protocol all communication with Amazon S3

servers is protected from 3rd party eavesdropping. This method is slower than plain HTTP, and can only be proxied with Python 2.7 or newer

Use HTTPS protocol [Yes]: No

On some networks all internet access must go through a HTTP proxy.

Try setting it here if you can't connect to S3 directly

HTTP Proxy server name:

New settings:

Access Key: 6741c65fa26f6ad8d833

Secret Key: eeUZ+PmlAtoEyNJ0dtGdXiIREE29HVh9yndIllNt

Default Region: region1

S3 Endpoint: s3-region1.cisco.cloudian.local

DNS-style bucket+hostname:port template for accessing a bucket: %(bucket)s.s3-region1.cisco.cloudian.local

Encryption password: Passw0rd!

Path to GPG program: /usr/bin/gpg

Use HTTPS protocol: False

```

HTTP Proxy server name:
HTTP Proxy server port: 0
Test access with supplied credentials? [Y/n] y
Please wait, attempting to list all buckets...
Success. Your access key and secret key worked fine :-)
Now verifying that encryption works...
Success. Encryption and decryption worked fine :-)
Save settings? [y/N] y
Configuration saved to '/root/.s3cfg'

```



When DNS is NOT available, the client should have the service endpoints defined in /etc/hosts. As the usage of wildcards is not allowed in /etc/hosts, the buckets that will be used by the client should also be defined in /etc/hosts.

5. When s3cmd has been successfully configured with S3 credentials and the S3 service endpoint name, you can now list the buckets that have been created with the following command:

```

s3cmd ls
2019-03-21 18:52  s3://test-bucket1

```

6. The objects in the bucket can be listed with the following command:

```

s3cmd la
2019-03-21 19:04  20075381  s3://test-
bucket1/ucs_clouidianhyperstore_s3260m5_deployment_guide.docx

```

7. To download an object from Clouidian HyperStore to the local home directory, run the following command:

```

s3cmd get s3://test-bucket1/ucs_clouidianhyperstore_s3260m5_deployment_guide.docx
~/home/ucs_clouidianhyperstore_s3260m5_deployment_guide_.docx

download: 's3://test-
bucket1/ucs_clouidianhyperstore_s3260m5_deployment_guide_draft.docx' ->
'/root/home/ucs_clouidianhyperstore_s3260m5_deployment_guide_.docx' [1 of 1]

20075381 of 20075381  100% in    0s   227.67 MB/s  done

```

8. To upload a file to Clouidian HyperStore, run the following:

```

s3cmd put /root/s3cmd-2.0.2-1.el7.noarch.rpm s3://test-bucket1/

upload: '/root/s3cmd-2.0.2-1.el7.noarch.rpm' -> 's3://test-bucket1/s3cmd-2.0.2-
1.el7.noarch.rpm' [1 of 1]

194693 of 194693  100% in    0s    7.59 MB/s  done

```

9. To verify the file was successfully uploaded, run the following:

```

s3cmd la

```

```
2019-03-21 21:39    194693    s3://test-bucket1/s3cmd-2.0.2-1.el7.noarch.rpm
```

```
2019-03-21 19:04  20075381  s3://test-  
bucket1/ucs_cloudianhyperstore_s3260m5_deployment_guide.docx
```

When all tests have been successful, your environment has been setup correctly and is ready for client access.

Add Additional Data Center and Nodes

Adding nodes and data centers to Clouidian HyperStore can be done through the CMC. However, the candidate cluster nodes need to be properly prepared, similar to the nodes that were used for the initial installation.

Ensure the OS is installed, and the basic network configuration has been setup for the additional nodes that will be added to Clouidian HyperStore.



Multiple data centers can also be installed during the initial installation; when doing so make sure to correctly specify the “Datacenter Name” in the fourth tab of the survey.csv.

Prepare the New Nodes

To prepare the new nodes, follow these steps:

1. From the master node run the `system_setup.sh` script and select option 9 “Prep New Node to add to Cluster”. This will remotely connect to the new candidate server and copy over the required binaries to prepare the system.

```

System Setup

9) Prep New Node to Add to Cluster

Choice: 9
System Setup » Run On Cluster

1) Prep New Node to Add to Cluster

P) Return to the Previous Menu

Choice: 1
System Setup » Prep New Node to Add to Cluster

IP Address of new node: 192.168.100.246

Attempting to install SSH key on 192.168.100.246
If your root password is the same on all (or most) nodes in the cluster, you can
supply it as a cluster password
If you do not want to supply a password, each server will prompt for one when
connecting.
Cluster Password:

=> On Server: 192.168.100.246 ... Done
Adding SSH Key to '/root/.ssh/authorized_keys'

Attempting to copy self extract installer to 192.168.100.246
Checking and creating remote directory path before transferring
'/root/ClouidianPackages/selfextract_prereq_el7.bin'

=> On Server: 192.168.100.246 ... Done
=> Transferring to Server: 192.168.100.246 ... Done
    Attempting to copy System Setup script to 192.168.100.246
Checking and creating remote directory path before transferring
'/root/ClouidianPackages/system_setup.sh'

```

```

=> On Server: 192.168.100.246 ... Done
=> Transferring to Server: 192.168.100.246 ... Done
=> On Server: 192.168.100.246 ... <==

```



The script is now remotely running on the new candidate cluster node and should be properly prepared as described in section “Prepare Cluster Nodes”. Complete this step for all nodes that will be added to the cluster.

System Setup (storage-node7)

- 1) Configure Networking
- 2) Change Timezone
- 3) Install & Configure Prerequisites
- 4) Setup Disks
- 5) Script Settings

X) Return to Master Node

Choice:

2. After running system_setup.sh on all new nodes and completing preparations, run the following command to verify all data drives have been successfully mounted.

```
for i in {6..12}; do ssh -t -i /root/CloudianPackages/cloudian-installation-key
storage-node${i} df -h |grep -c cloudian ; done
```

```

Connection to storage-node6 closed.
28
Connection to storage-node7 closed.
28
Connection to storage-node8 closed.
28
Connection to storage-node9 closed.
28
Connection to storage-node10 closed.
28
Connection to storage-node11 closed.
28
Connection to storage-node12 closed.
28

```

Add a New DC

To add a new DC, follow these steps:

1. When it's confirmed all drives are mounted correctly, connect to the CMC, go to Cluster, Data Centers and click the “+ NEW DC” next to the initial installed data center.

The screenshot shows the Cloudian management console. At the top, there is a navigation bar with the Cloudian logo and menu items: Analytics, Buckets & Objects, Users & Groups, and Cluster. Below this is a secondary navigation bar with Data Centers, Nodes, Cluster Config, and Storage Policies. The main content area shows a region labeled 'REGION1' with a '+ NEW REGION' button. A 'Node Status' legend is located at the top right, defining icons for Unreachable (red question mark), Has Disk Error (red X), Under Maintenance (blue gear), and Add Node (grey gear). The central part of the interface displays a data center 'DC1' containing a rack 'Rack1' with six green nodes, each with a white checkmark, and a yellow node with a white plus sign. Below the rack, it says '6 node(s)'. To the right of the DC1 view is a large dashed green box containing a '+ NEW DC' label, indicating the option to add a new data center.

2. When adding a DC, set the “System Metadata Replication Factor” to 3 and complete the required fields to add a host and click “ADD MORE NODES” to add details for all nodes that will be added to the additional Datacenter. Once the information for all nodes has been provided and verified click execute to start the add DC progress.



Make sure to provide the correct Interface name for the internal cluster communication network as it might be different from the initial DC when using VLAN interfaces.

Add DC ?



System Metadata Replication Factor

Data Center Name

Hostname <input type="text" value="storage-node7"/>	Region Name region1
IP Address <input type="text" value="192.168.100.246"/>	Data Center Name
Internal Network Interface Name (optional) <input type="text" value="eth2"/>	Rack Name <input type="text" value="rack1"/>
Installation User's Password <input type="password" value="....."/>	
<input type="checkbox"/> Private Key Authentication	

Hostname <input type="text" value="storage-node8"/>	Region Name region1 ✕
IP Address <input type="text" value="192.168.100.247"/>	Data Center Name
Internal Network Interface Name (optional) <input type="text" value="eth2"/>	Rack Name <input type="text" value="rack1"/>
Installation User's Password <input type="password" value="....."/>	
<input type="checkbox"/> Private Key Authentication	



Do not use more than one "Rack Name" unless you are using replication and fully understand the concept.

- To follow the progress of the DC add operation, go to the Operation Status Page by clicking the "Operation Status Page" link or by going to "Cluster" followed by "Operation Status".

Add DC Status: **Started** ✔

Add DC has been successfully started. Go to the Operation Status page to check the progress.

[Operation Status page](#)

- Click "View" to see more details of the "addDC" operation.

OPERATION LIST

Show 10 entries

Search:

OPERATION NAME	TARGET	STATUS	PROGRESS	START TIME	LAST UPDATE	
addDC	DC2	In progress	20%	Mar-26-2019 10:29	Mar-26-2019 10:35	View

Showing 1 to 1 of 1 entries

Previous Next



The “addDC” operation creates the SSH keys, updates the survey.csv, and runs all of the pre-installation checks before adding the nodes to the additional data center.

Operation Status



OPERATION NAME	TARGET	STATUS	START TIME	LAST UPDATE
addDC	DC2	In progress	Mar-26-2019 10:29	Mar-26-2019 10:36
		20%		

```

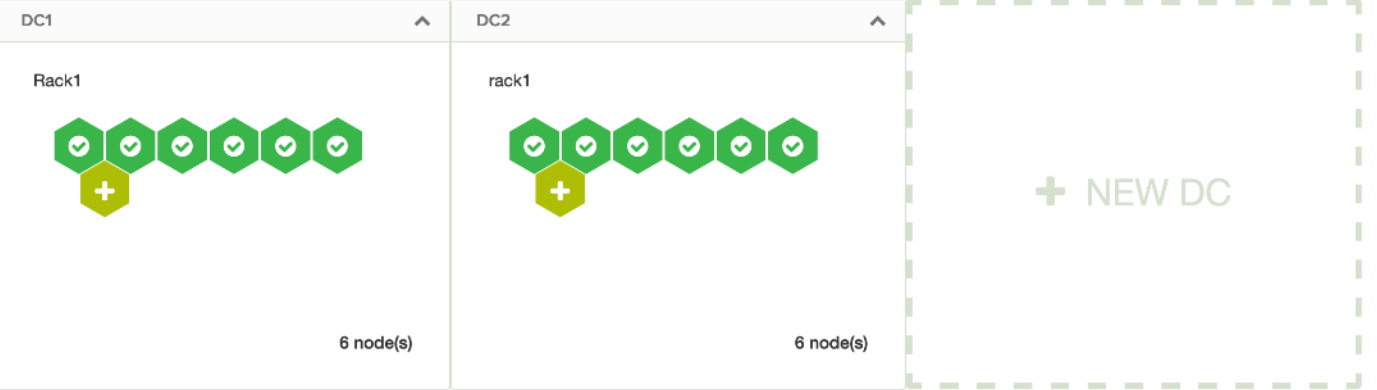
The operation is initialized and the session will be hold by the node: storage-node1
addDC is starting
Copied /export/home/cloudian/cloudian-installation-key.pub to /root/cloudian-installation-key.pub on remote host.
-----
Ensured SSH directory exists on node: 192.168.100.246
-----

Successfully added SSH pub key to authorized keys on node: 192.168.100.246
-----
Successfully set permissions on authorized keys on node: 192.168.100.246
Copied /export/home/cloudian/cloudian-installation-key.pub to /root/cloudian-installation-key.pub on remote host.
-----
Ensured SSH directory exists on node: 192.168.100.247
-----
    
```

Once the “addDC” operation has completed successfully, the new data center and nodes should now be listed under “Data Centers” and are ready to be configured with a storage policy.

REGION1 + NEW REGION

Node Status: Unreachable Has Disk Error Stop Write Disk Above 80% Full Has Alerts
 Under Maintenance Add Node in progress All Clear



DC1 ^ DC2 ^

Rack1 rack1

6 node(s) 6 node(s)

+ NEW DC



To add a new node to an existing data center, follow the same node preparation steps and simply click the “+” symbol in the DC you want to add the node to be added and complete the node information. For more information about adding Datacenters please refer to the [HyperStore Installation Guide](#).

Create a Multi-DC Storage Policy

Once the additional Datacenter and nodes have been successfully added, a storage policy has to be created to utilize the new DC.

To create a multi-DC storage policy, follow these steps:

1. Within the CMC, go to “Cluster”, “Storage Policies” and click “+ CREATE STORAGE POLICY”.
2. Specify a “Policy Name” and select the desired data distribution scheme which could be “Replication Across Datacenters” or ‘Replicated EC” with different EC scheme options.
3. For “Replication Across Datacenters” select the total number of copies that need to be replicated.

STORAGE POLICIES

[+ CREATE STORAGE POLICY](#)

CREATE NEW POLICY

Policy Name
multi-dc-rf

Policy Description
Replication across two datacenters

NUMBER OF DATACENTERS
2

DATA DISTRIBUTION SCHEME

Replication Across Datacenters

Replicated EC

NUMBER OF REPLICAS
4

- Select the data center assignment for each replica and set the desired consistency level. Since there are multiple DC's there are a lot more consistency level options. To ensure strong consistency in the local DC but eventual consistency for replication to the remote DC, make sure to use "LOCAL QUORUM".



For more information about consistency levels, please refer to the [HyperStore Installation Guide](#).

DATACENTER ASSIGNMENT

REGION	DATACENTER	REPLICA	LOCAL EC
region1	DC1	1 of 4	
	DC1	2 of 4	disable
	DC2	3 of 4	
	DC2	4 of 4	

CONSISTENCY SETTING

CONSISTENCY LEVEL	READ	WRITE
ALL	<input type="checkbox"/>	<input type="checkbox"/>
EACH QUORUM		<input type="checkbox"/>
QUORUM	<input type="checkbox"/>	<input type="checkbox"/>
LOCAL QUORUM	<input type="checkbox"/>	<input checked="" type="checkbox"/>
ONE	<input checked="" type="checkbox"/>	

GROUP VISIBILITY

ADD

Compression Type

Server-side Encryption

SAVE
CANCEL

5. For “Replicated EC” select the desired Erasure Coding scheme.

169

STORAGE POLICIES

[+ CREATE STORAGE POLICY](#)

CREATE NEW POLICY

Policy Name
repl-ec42

Policy Description
Replicated erasure code

NUMBER OF DATACENTERS
2

DATA DISTRIBUTION SCHEME

Replication Across Datacenters
 Replicated EC

ERASURE CODING K+M VALUE

- 2+1
- ✓ 4+2
- 6+2
- 8+2
- 9+3
- 12+4

6. Select the target DC's and set the desired consistency level and click Save to exit.

DATACENTER ASSIGNMENT

REGION	DATACENTER	SELECTED
region1	DC1	<input checked="" type="checkbox"/>
	DC2	<input checked="" type="checkbox"/>

CONSISTENCY SETTING

CONSISTENCY LEVEL	READ	WRITE
ALL	<input type="checkbox"/>	<input type="checkbox"/>
EACH QUORUM		<input type="checkbox"/>
LOCAL QUORUM	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
ANY QUORUM	<input type="checkbox"/>	<input type="checkbox"/>

GROUP VISIBILITY

Please select a Group [ADD](#)

Compression Type: NONE

Server-side Encryption: NONE

[SAVE](#) [CANCEL](#)

- 7. A local storage policy for the new DC can be created as well. When more than one storage policy exists, the default policy can be set by editing a storage policy and configure the “set default” setting.

STORAGE POLICIES + CREATE STORAGE POLICY

<input type="checkbox"/>	REGION	STATUS	NAME	DATA DISTRIBUTION POLICY	NO OF REPLICAS	EC K+M VALUE	DEFAULT	
<input type="checkbox"/>	region1	ACTIVE	ec42	Single DC	1	4 + 2 each dc	<input checked="" type="checkbox"/>	View/Edit
<input type="checkbox"/>	region1	ACTIVE	ec42-dc2	Single DC	1	4 + 2 each dc	<input type="checkbox"/>	View/Edit
<input type="checkbox"/>	region1	ACTIVE	multi-dc-rf	Multi DC	4	N/A	<input type="checkbox"/>	View/Edit
<input type="checkbox"/>	region1	ACTIVE	repl-ec42	Multi DC	2	4 + 2 each dc	<input type="checkbox"/>	View/Edit
<input type="checkbox"/>	region1	ACTIVE	rf3	Single DC	3	N/A	<input type="checkbox"/>	View/Edit

ENABLE DISABLE DELETE



For more information about installing and configuring Cloudfan HyperStore, please refer to the [Hyper-Store Installation Guide](#).

Performance

S3 Performance was evaluated on the Cloudian HyperStore system running on Cisco S3260 M5 UCS hardware. The goal of the performance testing was to evaluate peak object performance under ideal conditions.

The performance tests were done using Intel's Cosbench with the following range of object sizes and worker threads:

Object sizes: 4KB, 16KB, 512KB, 1MB, 4MB, 10MB
Threads: 1, 10, 100, 200, 400, 600, 1000

To run the Cosbench workload, 12 client nodes with 40Gb ethernet were used as Cosbench drivers. The same Cosbench workload was used to run load on buckets with different storage policies. The following storage policies were tested:

- Erasure Coding 4+2
- 3-way Replication
- Replicated Erasure Coding 4+2
- 4-way Replication across 2 DC's

Erasure Code 4+2 - Read Performance

Figure 62 Erasure Code 4+2 - Read Performance



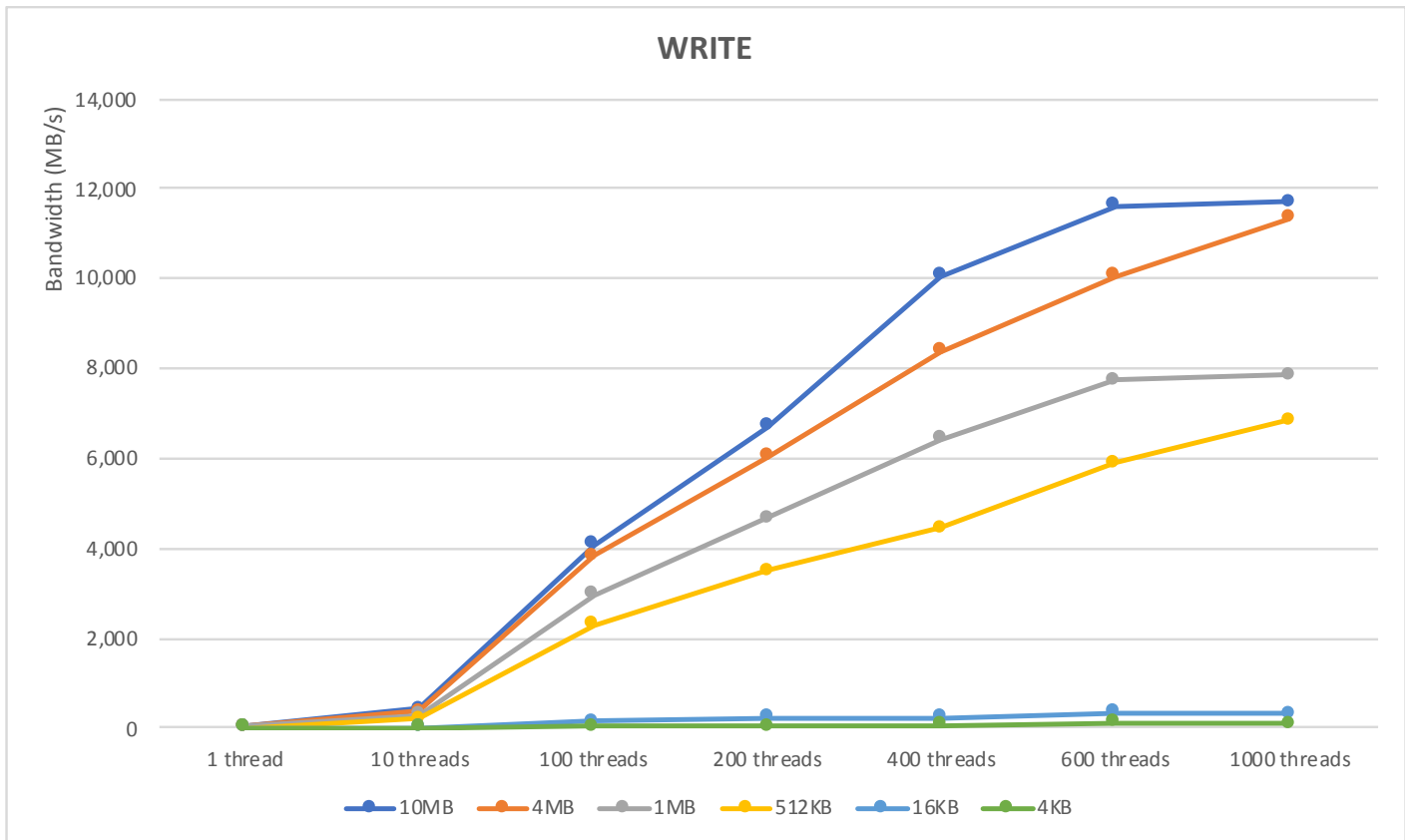
	1 thread	10 threads	100 threads	200 threads	400 threads	600 threads	1000 threads
10MB	314690853	2075814777	13981645774	17076272803	19034818946	20082315950	20469254168
4MB	182316712	1201472143	9820837606	15022419543	17842684687	18992401219	19871436187
1MB	52735563.6	386396877	3302688201	6811909282	12365487140	13401737691	13481466469
512KB	26975669.5	209056578	1852340023	4009007357	7857440915	5794635711	8265813799
16KB	1214297.02	9129689.03	72951811.97	156005793.8	293738299.9	220664889.4	348380608.5
4KB	288495.52	2146777.48	19537973.43	40760695.5	68719877.68	57965600.16	90688237.71

Test Description	
Policy	EC4+2
Cluster Nodes	12
Bandwidth	2x 40G
DC's	2
Target buckets	2 (1 DC1, 1 DC2)
Client nodes	12
Bandwidth	1x 40G

Observe that Read bandwidth peaks at 20.47 GB/s at an object size of 10MB with 1000 threads.

Erasure Code 4+2 - Write Performance

Figure 63 Erasure Code 4+2 - Write Performance



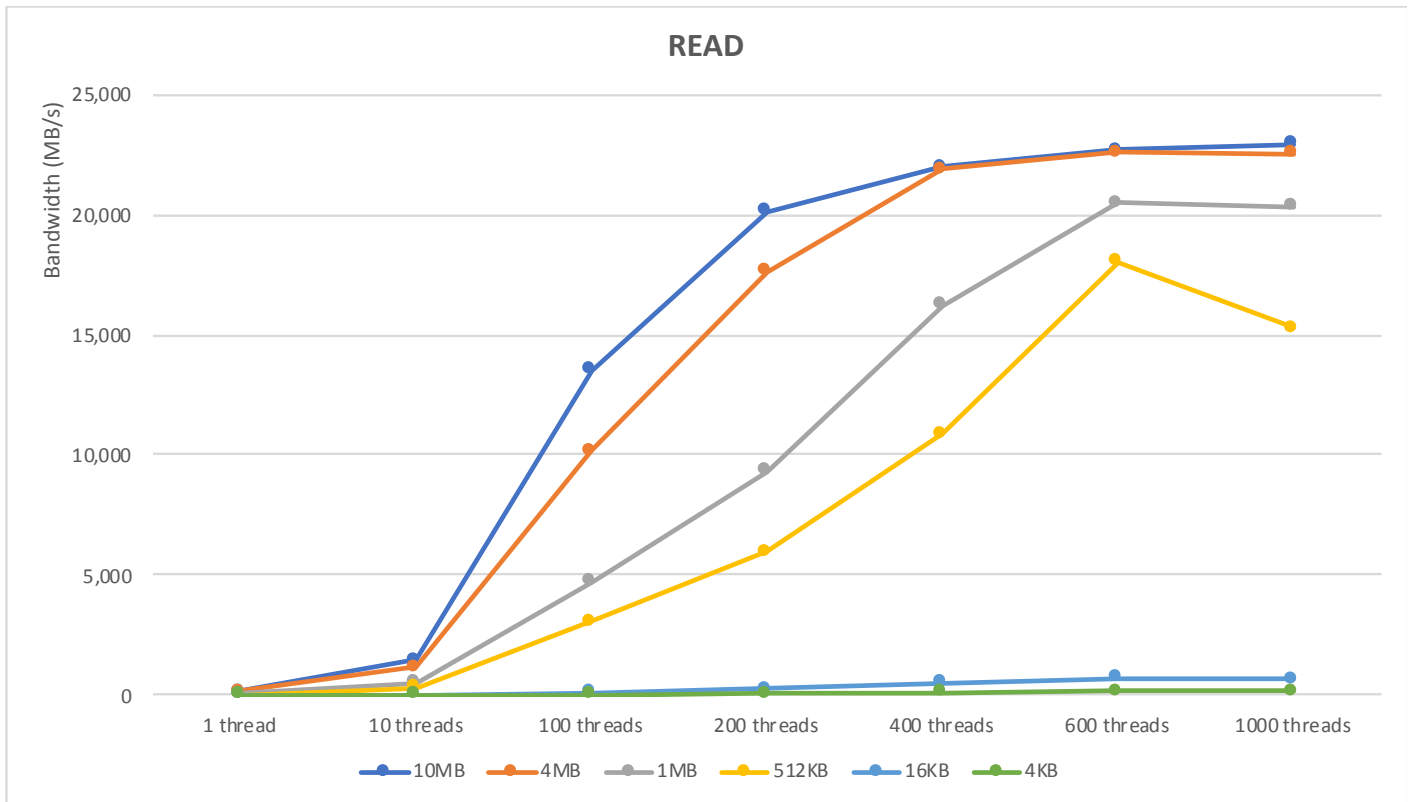
	1 thread	10 threads	100 threads	200 threads	400 threads	600 threads	1000 threads
10MB	45284340.6	430531923	4096461936	6717584583	10068022853	11617716990	11707638583
4MB	40435043.7	385156296	3829394497	6023877111	8390574163	10064543799	11351932200
1MB	27350783.6	278605792	2972762176	4678899277	6419207383	7746335501	7848727102
512KB	20262185.8	206587589	2293417387	3488891500	4458914762	5899076329	6846273159
16KB	1130094.17	14511465.6	155728356	240168151	224213733.8	337812377.6	325007836.4
4KB	324732.47	3667265.08	42034917.1	40110373.5	63456384.16	113805871.6	85322887.86

Test Description	
Policy	EC4+2
Cluster Nodes	12
Bandwidth	2x 40G
DC's	2
Target buckets	2 (1 DC1, 1 DC2)
Client nodes	12
Bandwidth	1x 40G

Observe that Write bandwidth peaks at 11.71 GB/s at an object size of 10MB with 1000 threads.

3-Way Replication - Read Performance

Figure 64 3-Way Replication - Read Performance



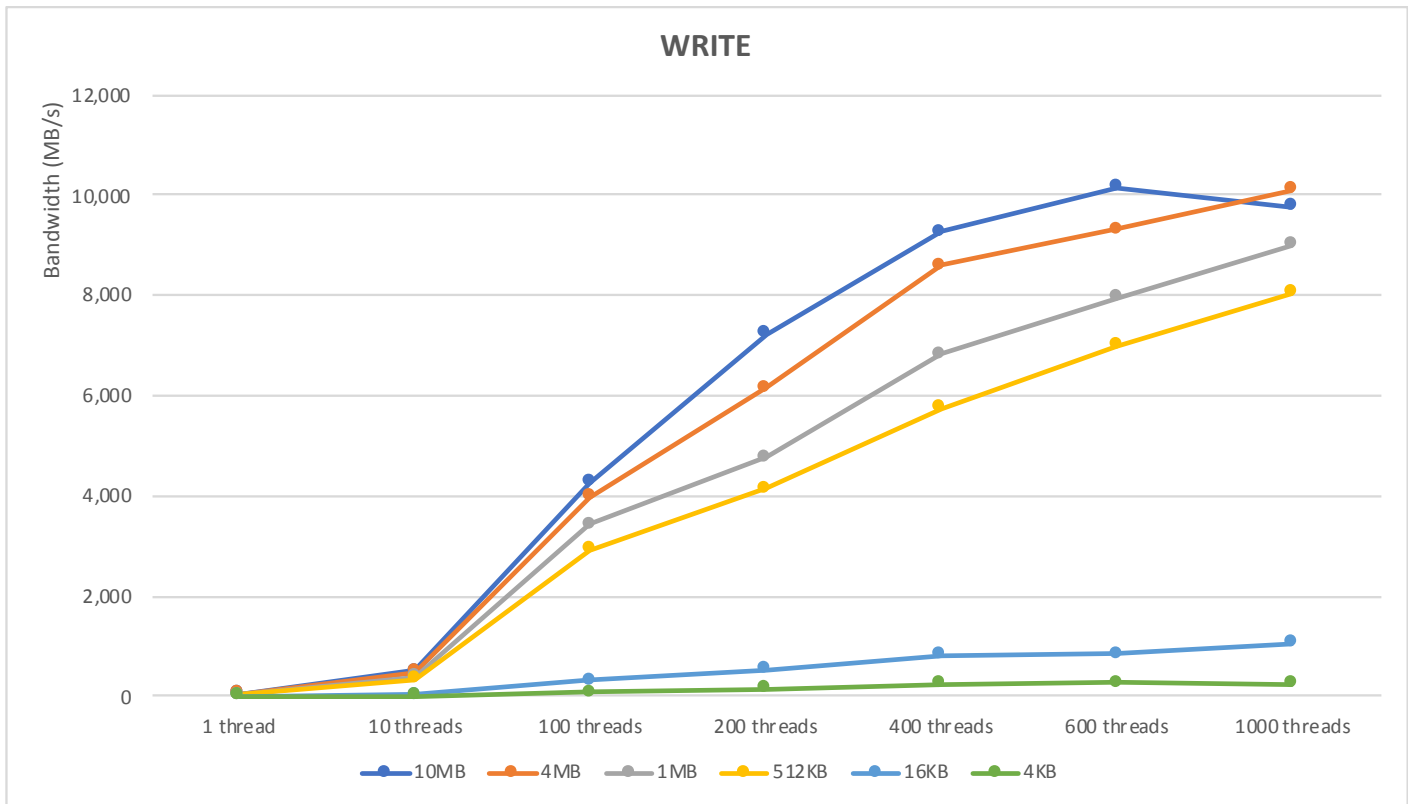
	1 thread	10 threads	100 threads	200 threads	400 threads	600 threads	1000 threads
10MB	171187472	1478569262	13524139719	20138217241	21995485364	22702992484	22929969129
4MB	153929521	1155144860	10163035595	17605357232	21883896359	22597141643	22520807389
1MB	71681600.3	501782263	4724423541	9320489227	16226755383	20504465332	20352797902
512KB	40153683.3	306879834	3054309616	5996017652	10866027234	18008500490	15297082678
16KB	1729567.56	11829361.9	128492208.7	281567210.2	508363302.4	710547354.5	677321976.9
4KB	475153.13	3274431.34	36169666.25	77405274.75	139199844	169218134.1	181462996.7

Test Description	
Policy	RF3
Cluster Nodes	12
Bandwidth	2x 40G
DC's	2
Target buckets	2 (1 DC1, 1 DC2)
Client nodes	12
Bandwidth	1x 40G

Observe that Read bandwidth peaks at 22.93 GB/s at an object size of 10MB with 1000 threads.

3-Way Replication - Write Performance

Figure 65 3-Way Replication - Write Performance



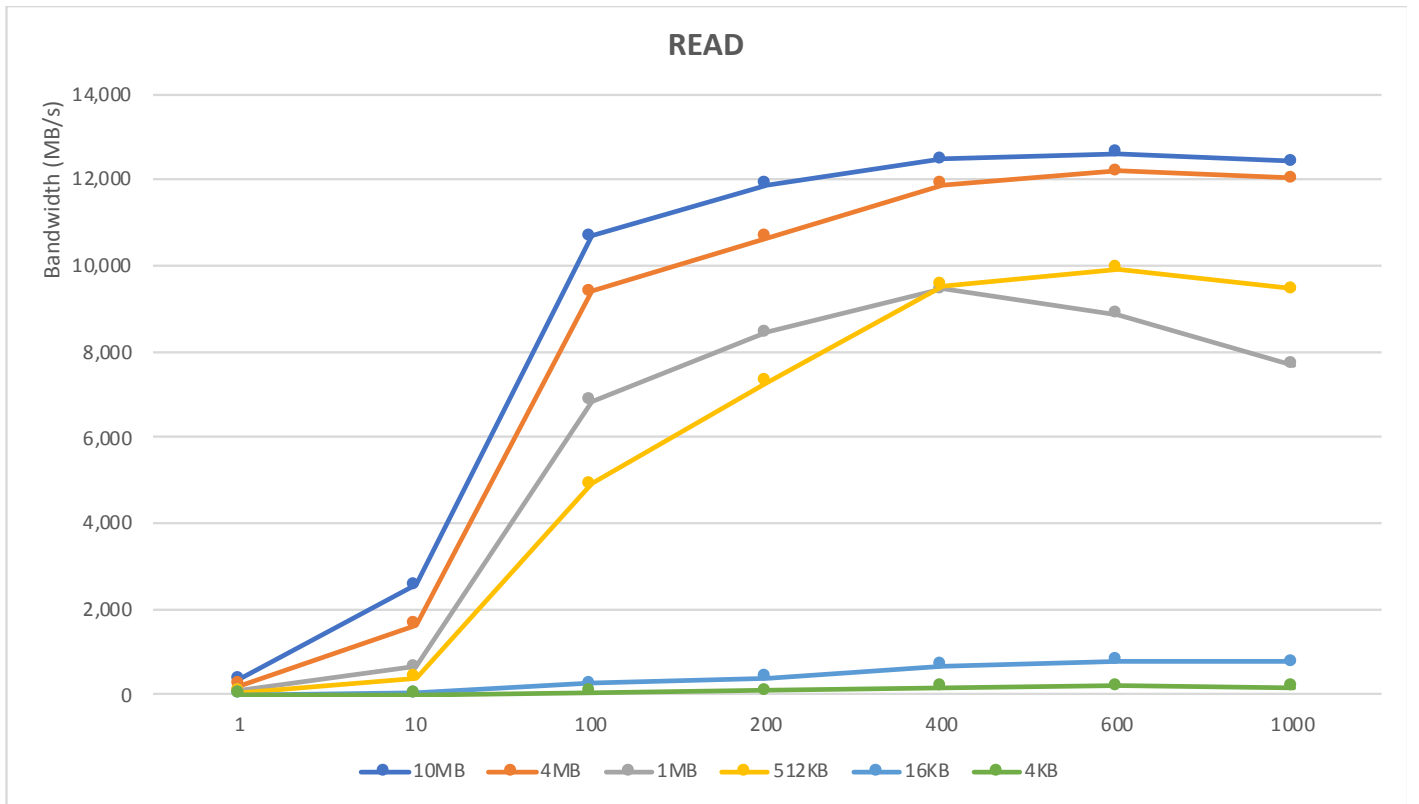
	1 thread	10 threads	100 threads	200 threads	400 threads	600 threads	1000 threads
10MB	49742524	502926374	4275857109	7231526664	9267149199	10156941392	9756789527
4MB	47058038.8	484230319	3978638472	6157699427	8602875336	9302532868	10092738841
1MB	38776684	402463920	3431521356	4767894234	6834960686	7938505525	9012637404
512KB	31072456	334765967	2932061355	4151493022	5745536518	6999835278	8047870987
16KB	2741691.39	38445904	319555143	543673084	831960130	853795175.7	1062033127
4KB	780646.34	10713614.3	89997447.66	154284492	237246243	266625566.7	256462241

Test Description	
Policy	RF3
Cluster Nodes	12
Bandwidth	2x 40G
DC's	2
Target buckets	2 (1 DC1, 1 DC2)
Client nodes	12
Bandwidth	1x 40G

Observe that Write bandwidth peaks at 10.2 GB/s at an object size of 10MB with 600 threads.

Replicated Erasure Code 4+2 - Read Performance

Figure 66 Replicated Erasure Code 4+2 - Read Performance



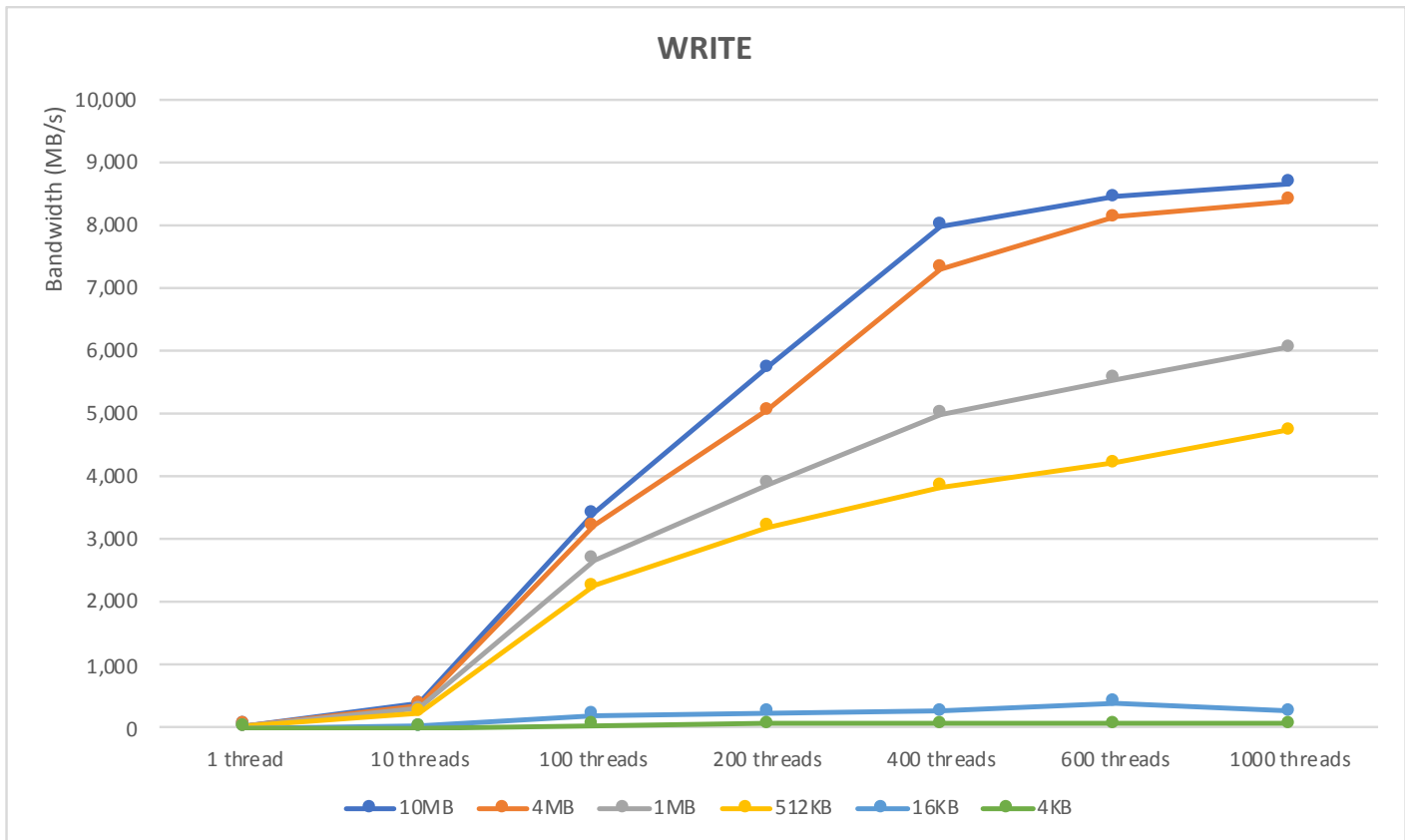
	1 thread	10 threads	100 threads	200 threads	400 threads	600 threads	1000 threads
10MB	371182681	2569290553	10682317536	11895164852	12469190621	12628291583	12425810687
4MB	239165222	1636202244	9408788799	10657991759	11881146494	12194232860	12032597768
1MB	98415714.5	650189271	6837566367	8450103146	9439612497	8858169063	7673379423
512KB	62242016.1	415943726	4912389789	7284565961	9522279602	9933510740	9444721774
16KB	3795288.06	28098308.5	268450475.9	395478684.6	675367221	791054604.2	767811862.9
4KB	925423.14	6376323.18	65236694.67	92180872.76	167763532.6	193490715.2	185492578

Test Description	
Policy	REP-EC42
Cluster Nodes	12
Bandwidth	2x 40G
DC's	2
Target buckets	1
Client nodes	12
Bandwidth	1x 40G

Observe that Read bandwidth peaks at 12.63 GB/s at an object size of 10MB with 600 threads.

Replicated Erasure Code 4+2 - Write Performance

Figure 67 Replicated Erasure Code 4+2 - Write Performance



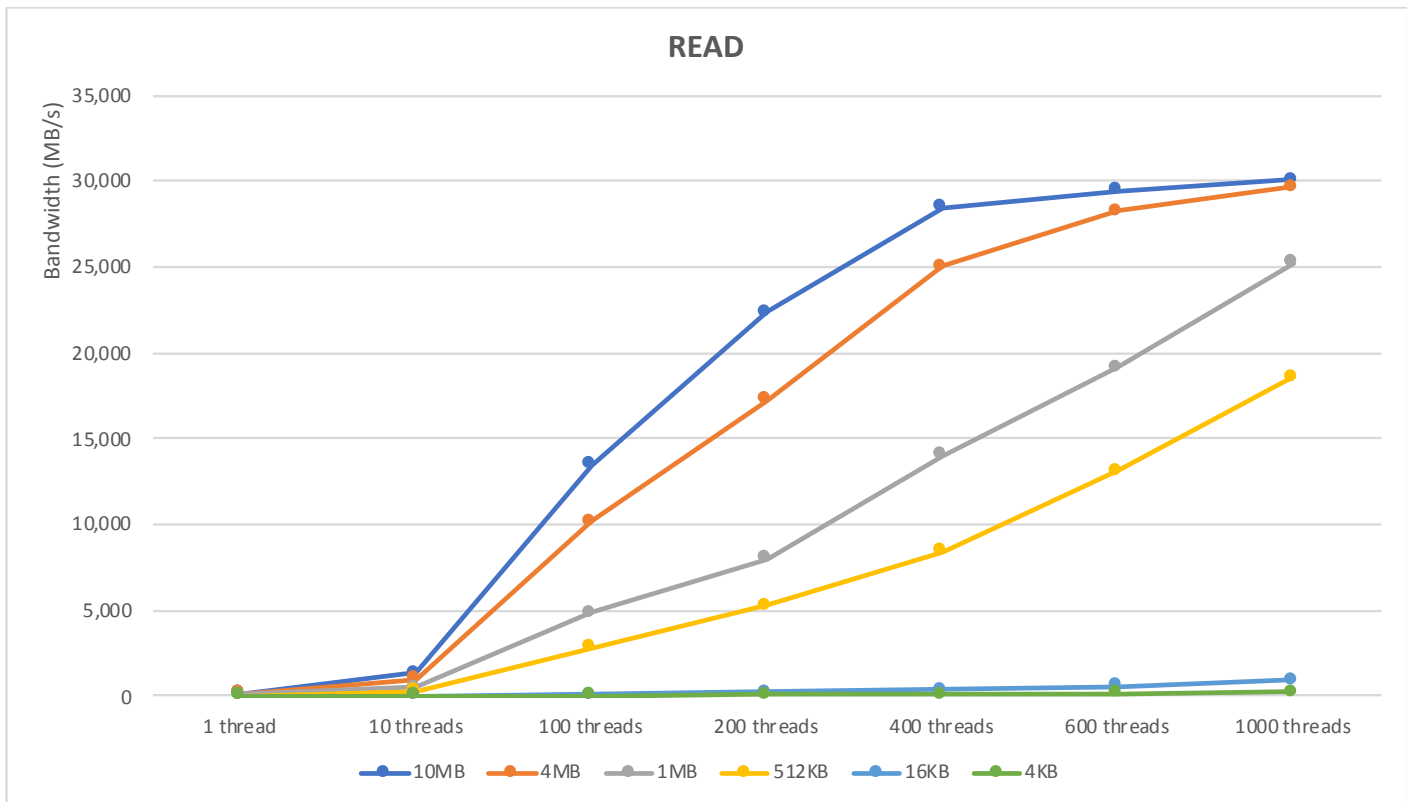
	1 thread	10 threads	100 threads	200 threads	400 threads	600 threads	1000 threads
10MB	39017232.6	394670071	3414104214	5726109826	7972990770	8437130640	8653919818
4MB	37419053.3	375587239	3220497418	5049839299	7305969603	8114628407	8384232563
1MB	30038840.5	315292182	2675020817	3873142098	4984495758	5542323023	6042872829
512KB	24175300.3	249613296	2255146767	3197766322	3835970563	4205816263	4728219692
16KB	2285238.06	28816258.2	201338834	251727053	263092698	411310380	272436507.2
4KB	680873.05	7829368.33	52875753.5	64921639.5	65988663.3	71640773.9	72312431.36

Test Description	
Policy	REP-EC42
Cluster Nodes	12
Bandwidth	2x 40G
DC's	2
Target buckets	1
Client nodes	12
Bandwidth	1x 40G

Observe that Write bandwidth peaks at 8.65 GB/s at an object size of 4MB with 1000 threads.

2DC 4-Way Replication - Read Performance

Figure 68 2DC 4-Way Replication - Read Performance



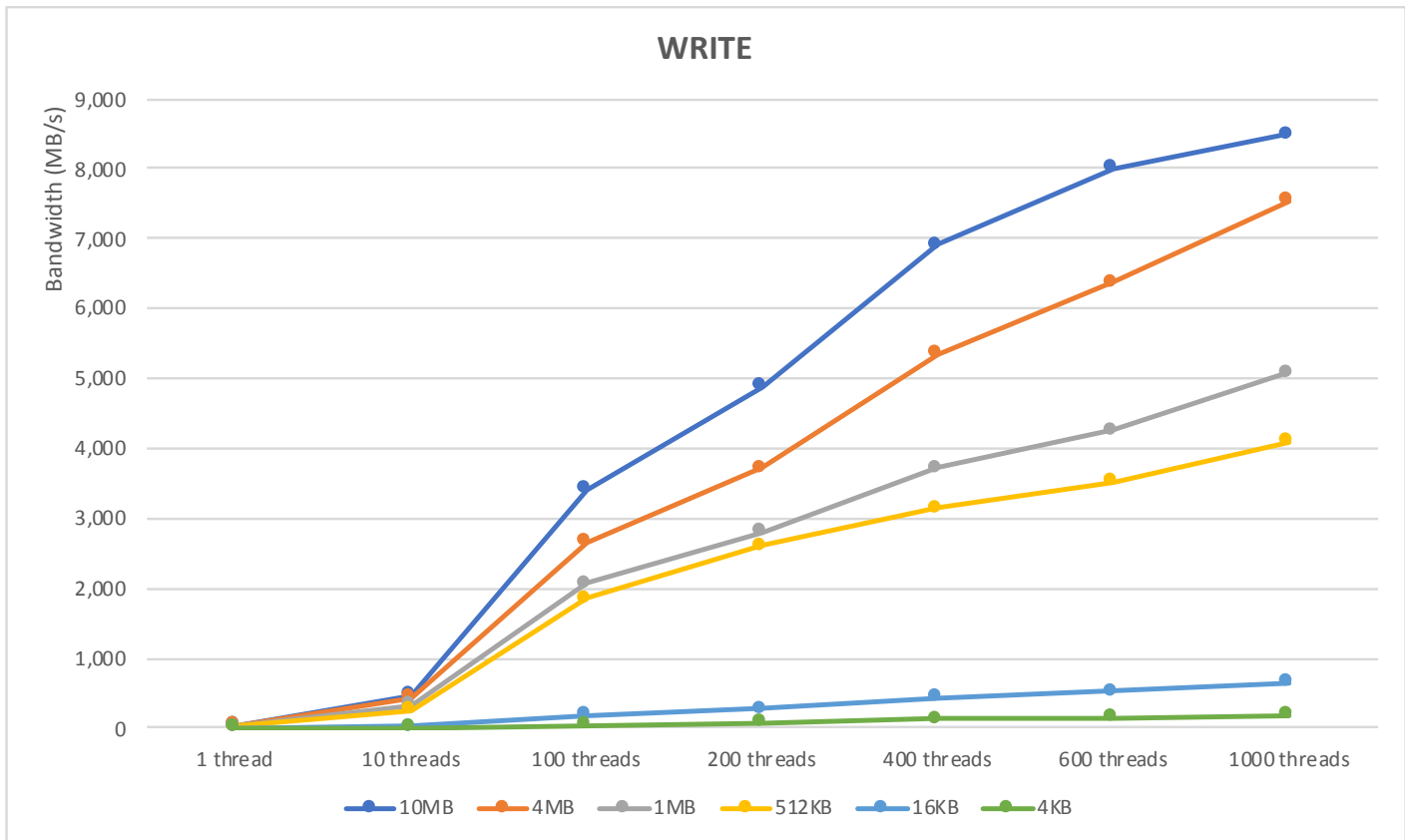
	1 thread	10 threads	100 threads	200 threads	400 threads	600 threads	1000 threads
10MB	157035752	1376200566	13507717406	22402055895	28484445877	29459919818	30058075257
4MB	145304204	1037783688	10192748377	17246946003	25011535137	28250699861	29646664815
1MB	69225639.7	495284718	4842926549	8029578063	14043731889	19164800273	25208439413
512KB	39901590.2	294354865	2843334756	5243297860	8430351552	13149827476	18604815572
16KB	1634762.03	12503660.8	124209906.2	231010016.9	403891134.6	609919062.5	922496248.6
4KB	536787.01	3618757.12	33848075.16	68967413.84	113277664.5	175060895.2	255413892.5

Test Description	
Policy	REP-EC42
Cluster Nodes	12
Bandwidth	2x 40G
DC's	2
Target buckets	1
Client nodes	12
Bandwidth	1x 40G

Observe that Read bandwidth peaks at 30.1 GB/s at an object size of 10MB with 1000 threads.

2DC 4-Way Replication - Write Performance

Figure 69 2DC 4-Way Replication - Write Performance



	1 thread	10 threads	100 threads	200 threads	400 threads	600 threads	1000 threads
10MB	46084873	481296961	3418289539	4893606354	6919503781	8011024451	8496868638
4MB	43749947.9	441191420	2661900884	3711885883	5351291535	6368434119	7546567448
1MB	29635309	334246715	2078384707	2807567921	3725117841	4259729386	5076535313
512KB	22187776.1	265669720	1857913407	2608573729	3141560887	3518437928	4099612054
16KB	1680540.05	23749356.3	180058203	276845921	442159176	519777365	654557972.6
4KB	459670.09	6574642.93	49563106.2	81148013.6	124514645	142335439	184170149.9

Test Description	
Policy	REP-EC42
Cluster Nodes	12
Bandwidth	2x 40G
DC's	2
Target buckets	1
Client nodes	12
Bandwidth	1x 40G

Observe that Write bandwidth peaks at 8.5 GB/s at an object size of 10MB with 1000 threads.

High Availability Tests

The high availability of this solution was validated by failing-out one of the components of the infrastructure.

The purpose of the high availability tests is to ensure Business Continuity when the underlying hardware components fail and study the behavior of the system during fault injections. The following points were considered while doing the high availability tests:

- As part of the high availability testing, a random read and write load test with objects of 10MB in size was run during the failure injections. The outputs like bandwidth and operations was collected before and after the failure events.
- Only one fault is injected at any point of time. No double failures are considered.
- Performance degradation is acceptable but there should not be any business interruption. The underlying infrastructure components should continue to operate with the remaining components.

The following are a few of the high availability tests conducted for this solution:

- Fabric Interconnect Failures
- Nexus 9000 Failures
- S3 Service failure
- Disk Failures

Fabric Interconnect Failures

To check the business continuity of the system during fabric interconnect failures, one of the fabric interconnects was rebooted after ramping up load through COSBench. The sequence of events for fault injection and checking the health of the cluster is provided below:

1. Log into one of the Fabric Interconnects.
2. Check the cluster status.

```
UCS-FI-6332-B# show cluster extended-state
Cluster Id: 0xe8594a622bd711e9-0x869bb08bcfa41dfd
```

```
Start time: Wed May 29 01:29:49 2019
Last election time: Wed May 29 01:40:04 2019
```

```
B: UP, PRIMARY
```

```
A: UP, SUBORDINATE
```

```
B: memb state UP, lead state PRIMARY, mgmt services state: UP
```

```
A: memb state UP, lead state SUBORDINATE, mgmt services state: UP
  heartbeat state PRIMARY_OK
```

```
INTERNAL NETWORK INTERFACES:
```

```
eth1, UP
```

```
eth2, UP
```

```
HA READY
```

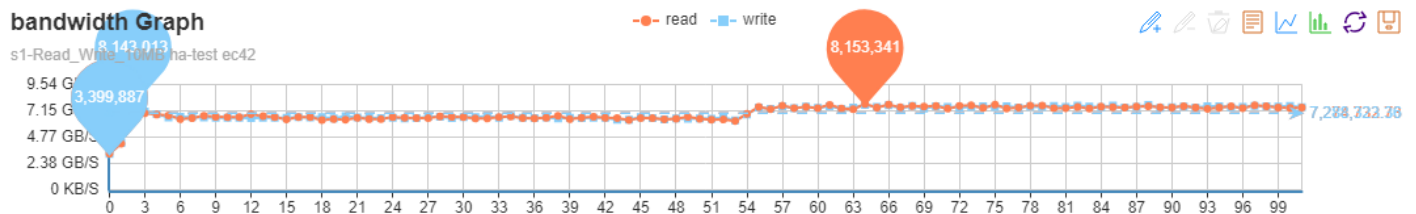
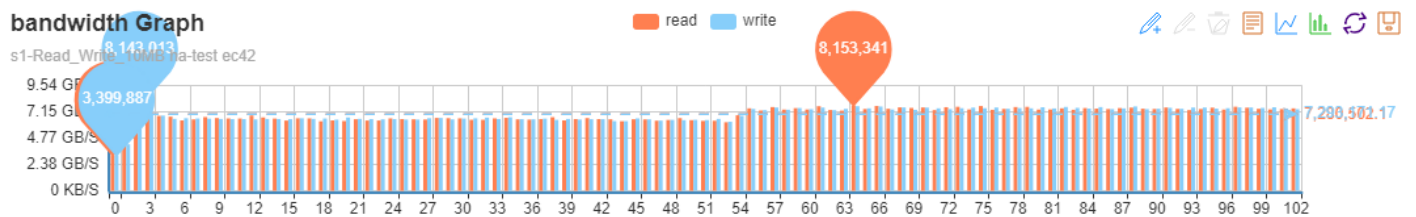
```
Detailed state of the device selected for HA storage:
Chassis 2, serial: FOX2232P39J, state: active
Chassis 3, serial: FOX2235P4CV, state: active
Chassis 5, serial: FOX2235P4DW, state: active
UCS-FI-6332-B#
```

The S3 COSBench test started for 10MB object size and with mix of read and write.

The following data was gathered after ramping up the load before fault injection:

Op-Type	Op-Count	Byte-Count	Avg-ResTime	Avg-ProcTime	Throughput	Bandwidth	Succ-Ratio
op1:read	7.12 kops	71.2 GB	96.76 ms	56.08 ms	710.17 op/s	7.1 GB/S	100%
op2:write	6.91 kops	69.06 GB	477.73 ms	319.07 ms	690.37 op/s	6.9 GB/S	100%

Bandwidth observed is as below:



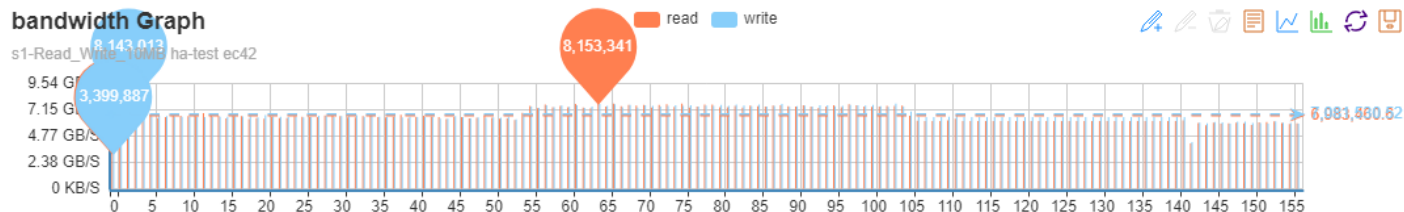
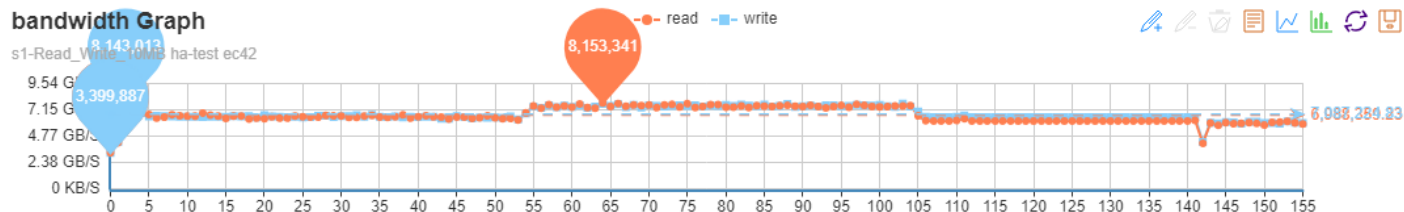
Reboot the fabric interconnect which carries the cluster traffic:

```
UCS-FI-6332-B# connect local-mgmt
Cisco Nexus Operating System (NX-OS) Software
TAC support: http://www.cisco.com/tac
Copyright (c) 2009, Cisco Systems, Inc. All rights reserved.
The copyrights to certain works contained in this software are
owned by other third parties and used and distributed under
license. Certain components of this software are licensed under
the GNU General Public License (GPL) version 2.0 or the GNU
Lesser General Public License (LGPL) Version 2.1. A copy of each
such license is available at
http://www.opensource.org/licenses/gpl-2.0.php and
http://www.opensource.org/licenses/lgpl-2.1.php

UCS-FI-6332-B(local-mgmt)# reboot
Before rebooting, please take a configuration backup.
Do you still want to reboot? (yes/no):yes
```

FI was reboot between 140 to 145.

When one of the FI's is down, COSBench continues to send the requests. However, the bandwidth comes down to 4.3 GB/sec for read and 4.4 GB/sec for write operations.



The output (below) confirms that FI is down now and the cluster is running on single FI in a degraded mode:

```
UCS-FI-6332-A# show cluster extended-state
Cluster Id: 0xe8594a622bd711e9-0x869bb08bcfa41dfd
```

```
Start time: Wed May 29 01:38:20 2019
Last election time: Wed May 29 02:08:57 2019
```

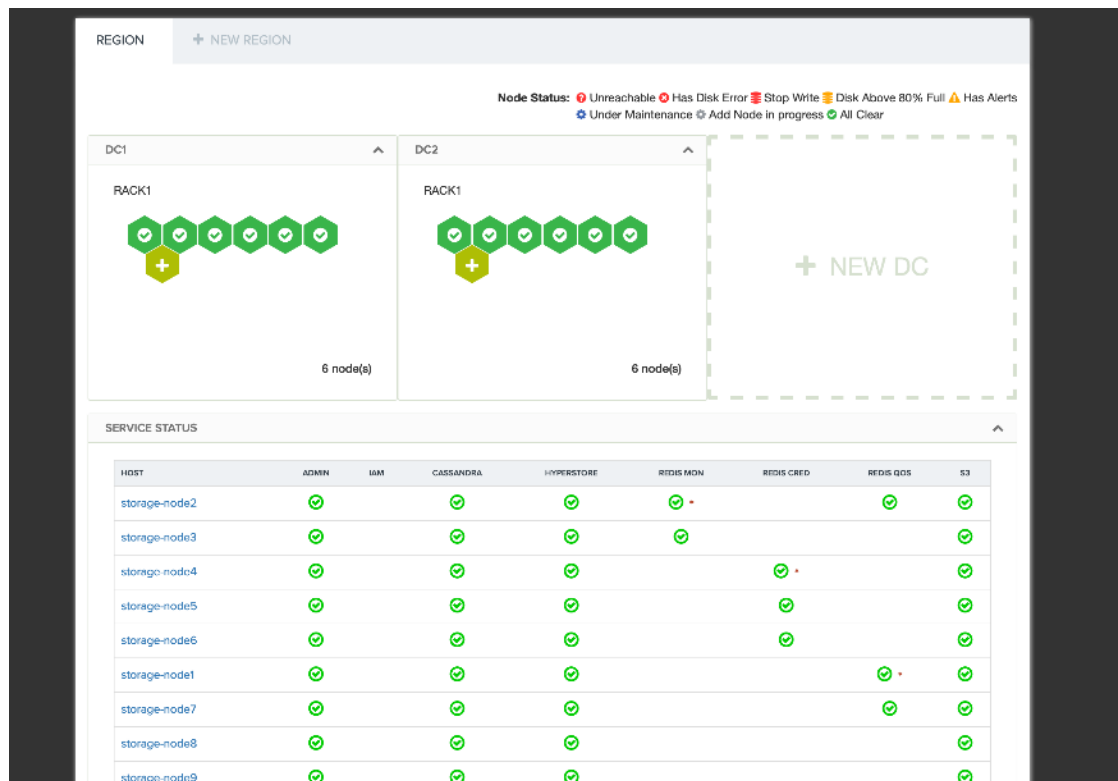
```
A: UP, PRIMARY
B: DOWN, INAPPLICABLE
```

```
A: memb state UP, lead state PRIMARY, mgmt services state: UP
B: memb state DOWN, lead state INAPPLICABLE, mgmt services state: DOWN
   heartbeat state SECONDARY_FAILED
```

```
INTERNAL NETWORK INTERFACES:
eth1, DOWN
eth2, DOWN
```

```
HA NOT READY
Peer Fabric Interconnect is down
Detailed state of the device selected for HA storage:
Chassis 2, serial: FOX2232P39J, state: active
Chassis 3, serial: FOX2235P4CV, state: active
Chassis 5, serial: FOX2235P4DW, state: active
UCS-FI-6332-A#
```

CMC dash board does not show any faults because of FI failure.



The system recovers after the Fabric joins the cluster and when HA READY. The dip in the graphs show the activity when the FI was rebooted.

```
UCS-FI-6332-A# show cluster extended-state
Cluster Id: 0xe8594a622bd711e9-0x869bb08bcfa41dfd
```

```
Start time: Wed May 29 01:38:20 2019
Last election time: Wed May 29 02:14:59 2019
```

```
A: UP, PRIMARY
B: UP, SUBORDINATE
```

```
A: memb state UP, lead state PRIMARY, mgmt services state: UP
B: memb state UP, lead state SUBORDINATE, mgmt services state: UP
  heartbeat state PRIMARY_OK
```

```
INTERNAL NETWORK INTERFACES:
eth1, UP
eth2, UP
```

HA READY

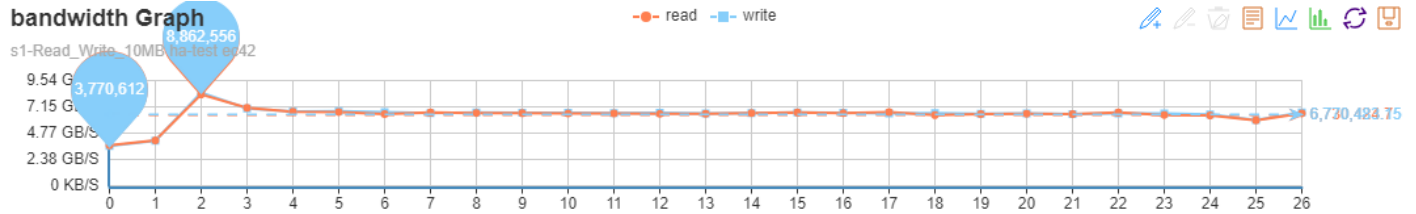
```
Detailed state of the device selected for HA storage:
Chassis 2, serial: FOX2232P39J, state: active
Chassis 3, serial: FOX2235P4CV, state: active
Chassis 5, serial: FOX2235P4DW, state: active
UCS-FI-6332-A#
```

Nexus 9000 Switch Failures

Similar to FI failures, one of the upstream Nexus switches was reloaded to make sure that there is business continuity. As both the FI's are connected to either of the switches and with VPC, the requests from the Nexus will still be forwarded to the FI's.

Reloaded the switch to check VPC status and impact on the application.

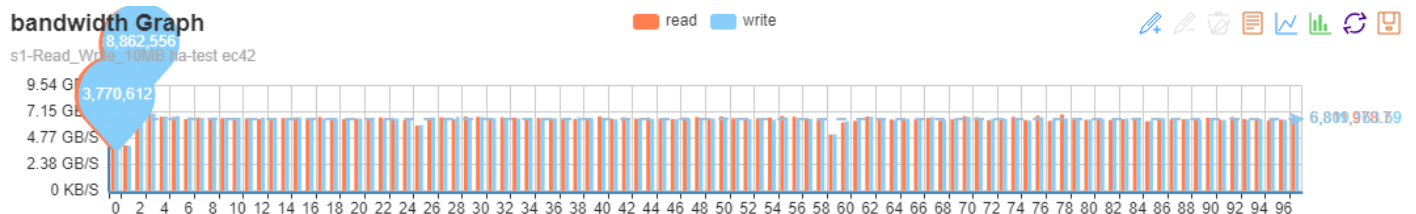
Similar workload as FI failures above was started on the system:



The N9K switch was reloaded.

```
N9K-Fab-B# show version | grep uptime  
Kernel uptime is 124 day(s), 15 hour(s), 19 minute(s), 17 second(s)  
N9K-Fab-B# configure terminal  
N9K-Fab-B(config)# reload  
This command will reboot the system. (y/n)? [n] y
```

```
N9K-Fab-B# show version | grep uptime  
Kernel uptime is 0 day(s), 0 hour(s), 7 minute(s), 20 second(s)
```



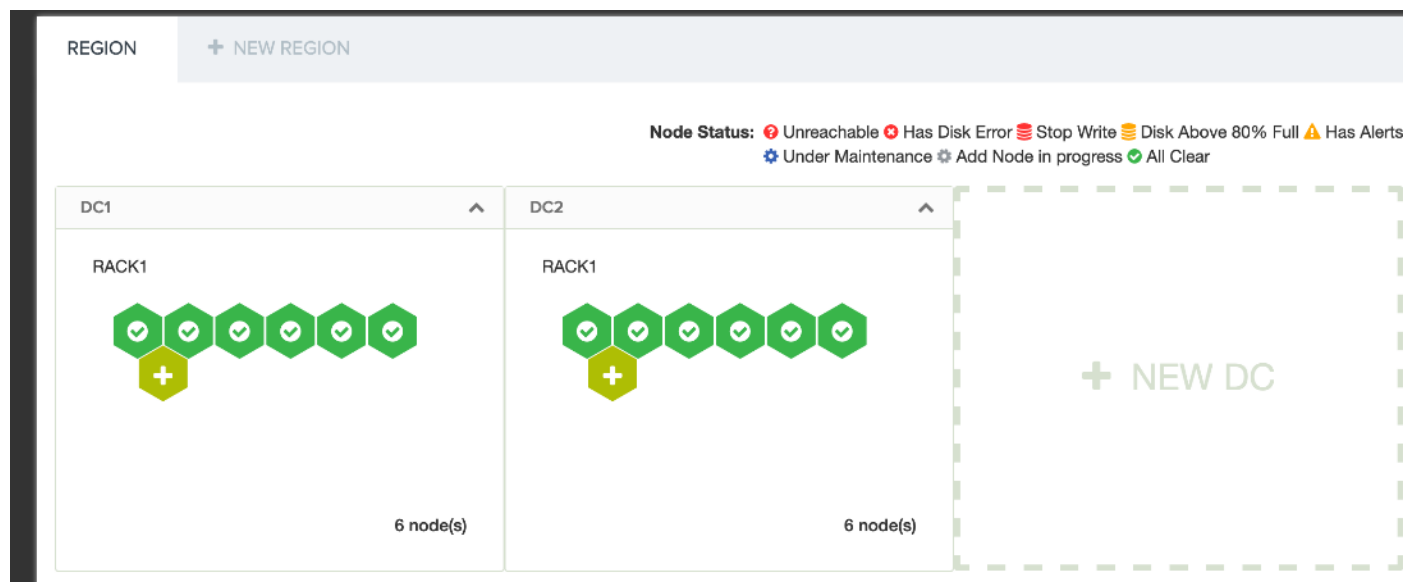
System was doing read/write mix of around 5.4 GB/s when the Nexus switch was reloaded [Between 58 and 60].

System continues to operate without any interruption.

S3 Service Failures

Client load was generated using COSBench and storage node 10 was rebooted.

Status of the storage nodes before fault injection:

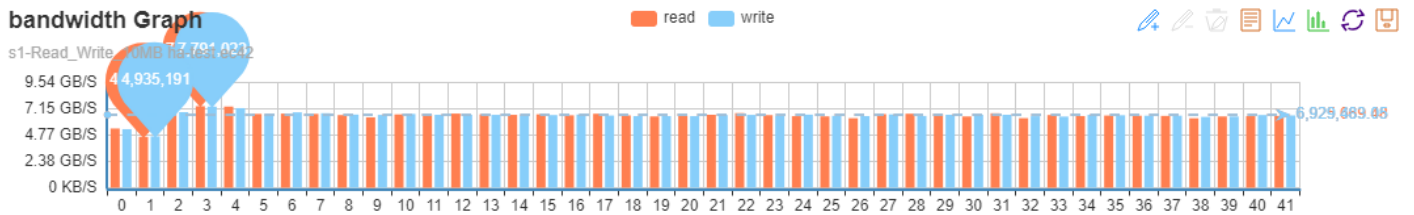


All of the services were running fine in each node.

SERVICE STATUS								
HOST	ADMIN	IAM	CASSANDRA	HYPERSTORE	REDIS MON	REDIS CRED	REDIS QOS	S3
storage-node2	✔		✔	✔	✔	✔	✔	✔
storage-node3	✔		✔	✔	✔			✔
storage-node4	✔		✔	✔		✔		✔
storage-node5	✔		✔	✔		✔		✔
storage-node6	✔		✔	✔		✔		✔
storage-node1	✔		✔	✔			✔	✔
storage-node7	✔		✔	✔			✔	✔
storage-node8	✔		✔	✔				✔
storage-node9	✔		✔	✔				✔
storage-node10	✔		✔	✔				✔
storage-node11	✔		✔	✔		✔		✔
storage-node12	✔		✔	✔		✔		✔

* Master/Primary

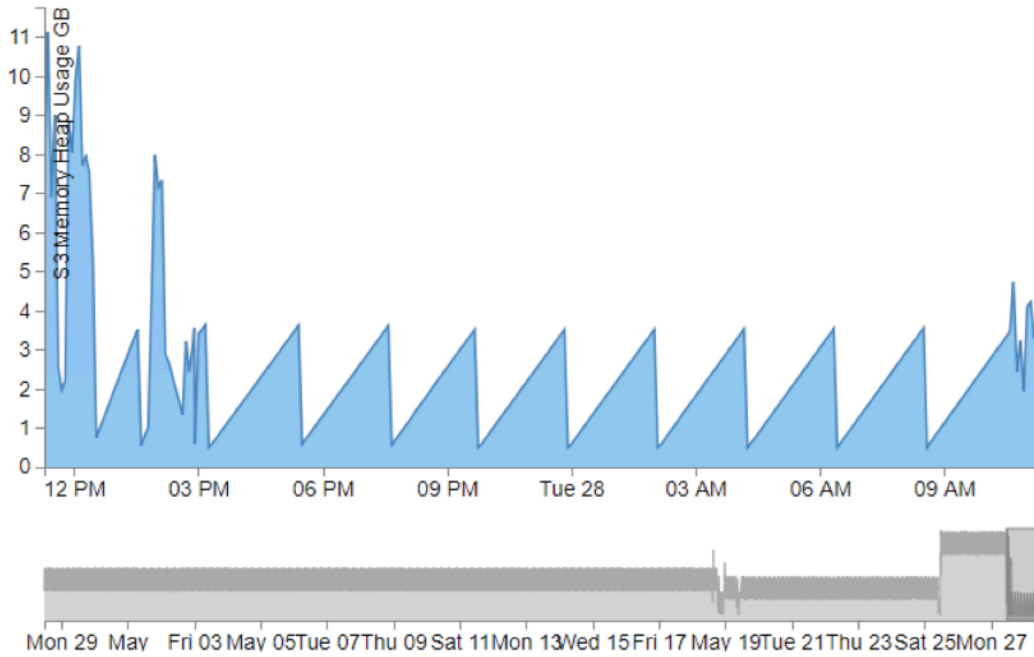
Bandwidth graph before the reboot:



Success ratio was 100 percent for both read and write operations.

Op-Type	Op-Count	Byte-Count	Avg-ResTime	Avg-ProcTime	Throughput	Bandwidth	Succ-Ratio
op1:read	7.06 kops	70.62 GB	89.91 ms	49.51 ms	693.53 op/s	6.94 GB/S	100%
op2:write	7.13 kops	71.33 GB	472.77 ms	308.53 ms	700.24 op/s	7 GB/S	100%

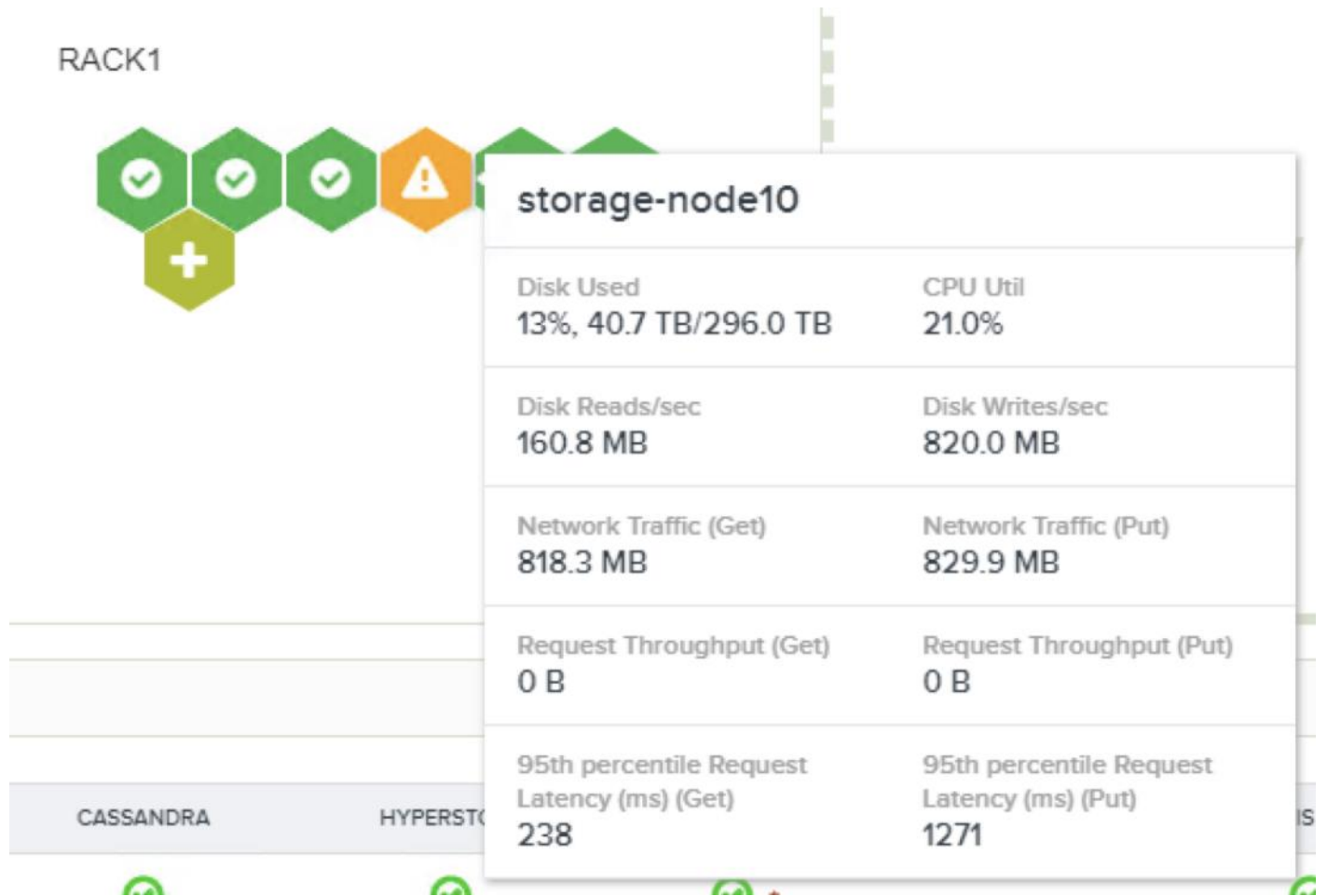
Memory heap usage report:



Server uptime report:

```
[root@storage-node10 ~]# uptime
11:22:32 up 9 days, 3:43, 1 user, load average: 31.45, 28.62, 24.92
[root@storage-node10 ~]#
```

The server was brought down. Node reported error in CMC.



Alerts were seen for storage node.

ALERT LIST + SHOW ACKNOWLEDGED

<input type="checkbox"/>	SEVERITY	ALERT TYPE	ALERT TEXT	LAST UPDATE	COUNT
<input type="checkbox"/>	High	Cassandra	[Service Down or Unreachable]	May-28-2019 11:27	2
<input type="checkbox"/>	High	HyperStore	[Service Down or Unreachable]	May-28-2019 11:27	2
<input type="checkbox"/>	High	Admin	[Service Down or Unreachable]	May-28-2019 11:27	5
<input type="checkbox"/>	High	S3	[Service Down or Unreachable]	May-28-2019 11:27	5
<input type="checkbox"/>	Critical	Node Unreachable	Node is unreachable from monitoring host (data collector)	May-28-2019 11:26	3

There were few write miss because of node failure:

Op-Type	Op-Count	Byte-Count	Avg-ResTime	Avg-ProcTime	Throughput	Bandwidth	Succ-Ratio
op1:read	7.18 kops	71.76 GB	95.56 ms	54.78 ms	721.2 op/s	7.21 GB/S	100%
op2:write	7.04 kops	70.44 GB	471.29 ms	309.98 ms	707.97 op/s	7.08 GB/S	99.87%

Read bandwidth was temporarily down to 6.1 abd write to 6.2 [Between 76 and 78]



Everything was back to normal when the storage node came back up . [After 99 server was up]



Disk Failure Tests

Disk was removed to understand the impact on HyperStore.

The figure below shows the healthy disks on node 3 as reported CMC.

DISK DETAIL INFO				
STATUS	DEVICE	MOUNT POINT	USE TYPE	DISK USAGE
	/dev/adc5	/var	CASSANDRA, LOG	17.3 GB of 107.0 GB used
	/dev/adaa	/loudian27	HS	2.4 TB of 10.6 TB used
	/dev/adab	/loudian28	HS	2.4 TB of 10.6 TB used
	/dev/ada	/loudian1	HS	2.3 TB of 10.6 TB used
	/dev/adb	/loudian2	HS	2.2 TB of 10.6 TB used
	/dev/adc	/loudian3	HS	2.2 TB of 10.6 TB used
	/dev/add	/loudian4	HS	2.3 TB of 10.6 TB used
	/dev/adc	/loudian5	HS	2.3 TB of 10.6 TB used
	/dev/adf	/loudian6	HS	2.3 TB of 10.6 TB used
	/dev/adg	/loudian7	HS	2.3 TB of 10.6 TB used
	/dev/adh	/loudian8	HS	2.3 TB of 10.6 TB used
	/dev/adi	/loudian9	HS	2.2 TB of 10.6 TB used
	/dev/adj	/loudian10	HS	2.4 TB of 10.6 TB used
	/dev/adk	/loudian11	HS	2.4 TB of 10.6 TB used
	/dev/adl	/loudian12	HS	2.4 TB of 10.6 TB used
	/dev/adm	/loudian13	HS	2.4 TB of 10.6 TB used
	/dev/adn	/loudian14	HS	2.4 TB of 10.6 TB used
	/dev/ado	/loudian15	HS	2.4 TB of 10.6 TB used
	/dev/adp	/loudian16	HS	2.4 TB of 10.6 TB used
	/dev/adq	/loudian17	HS	2.4 TB of 10.6 TB used
	/dev/adr	/loudian18	HS	2.5 TB of 10.6 TB used
	/dev/ada	/loudian19	HS	2.4 TB of 10.6 TB used
	/dev/adt	/loudian20	HS	2.3 TB of 10.6 TB used
	/dev/adu	/loudian21	HS	2.3 TB of 10.6 TB used
	/dev/adv	/loudian22	HS	2.4 TB of 10.6 TB used
	/dev/adw	/loudian23	HS	2.4 TB of 10.6 TB used
	/dev/adx	/loudian24	HS	2.4 TB of 10.6 TB used
	/dev/ady	/loudian25	HS	2.4 TB of 10.6 TB used
	/dev/adz	/loudian26	HS	2.4 TB of 10.6 TB used

UCSM also showed all the disks are healthy.

Name	Size (MB)	Serial	Operability	Drive State	Presence	Technology	Bootable
Storage Controller PC...							
Storage Controller SA...							
Disk 1	11443108	AAGA563H	Operable	Jboot	Equipped	HDD	False
Disk 2	11443108	AAG9H88H	Operable	Jboot	Equipped	HDD	False
Disk 3	11443108	BHKEJUNH	Operable	Jboot	Equipped	HDD	False
Disk 4	11443108	BHKBYRNH	Operable	Jboot	Equipped	HDD	False
Disk 5	11443108	AAG0YHSH	Operable	Jboot	Equipped	HDD	False
Disk 6	11443108	AAG9NTTH	Operable	Jboot	Equipped	HDD	False
Disk 7	11443108	AAGA0RJH	Operable	Jboot	Equipped	HDD	False
Disk 8	11443108	AAG9NXTH	Operable	Jboot	Equipped	HDD	False
Disk 9	11443108	AAG9LEMH	Operable	Jboot	Equipped	HDD	False
Disk 10	11443108	AAG87ZKH	Operable	Jboot	Equipped	HDD	False
Disk 11	11443108	AAG93AWH	Operable	Jboot	Equipped	HDD	False
Disk 12	11443108	AAG9YWAH	Operable	Jboot	Equipped	HDD	False
Disk 13	11443108	AAG9VWWH	Operable	Jboot	Equipped	HDD	False
Disk 14	11443108	AAGA235H	Operable	Jboot	Equipped	HDD	False

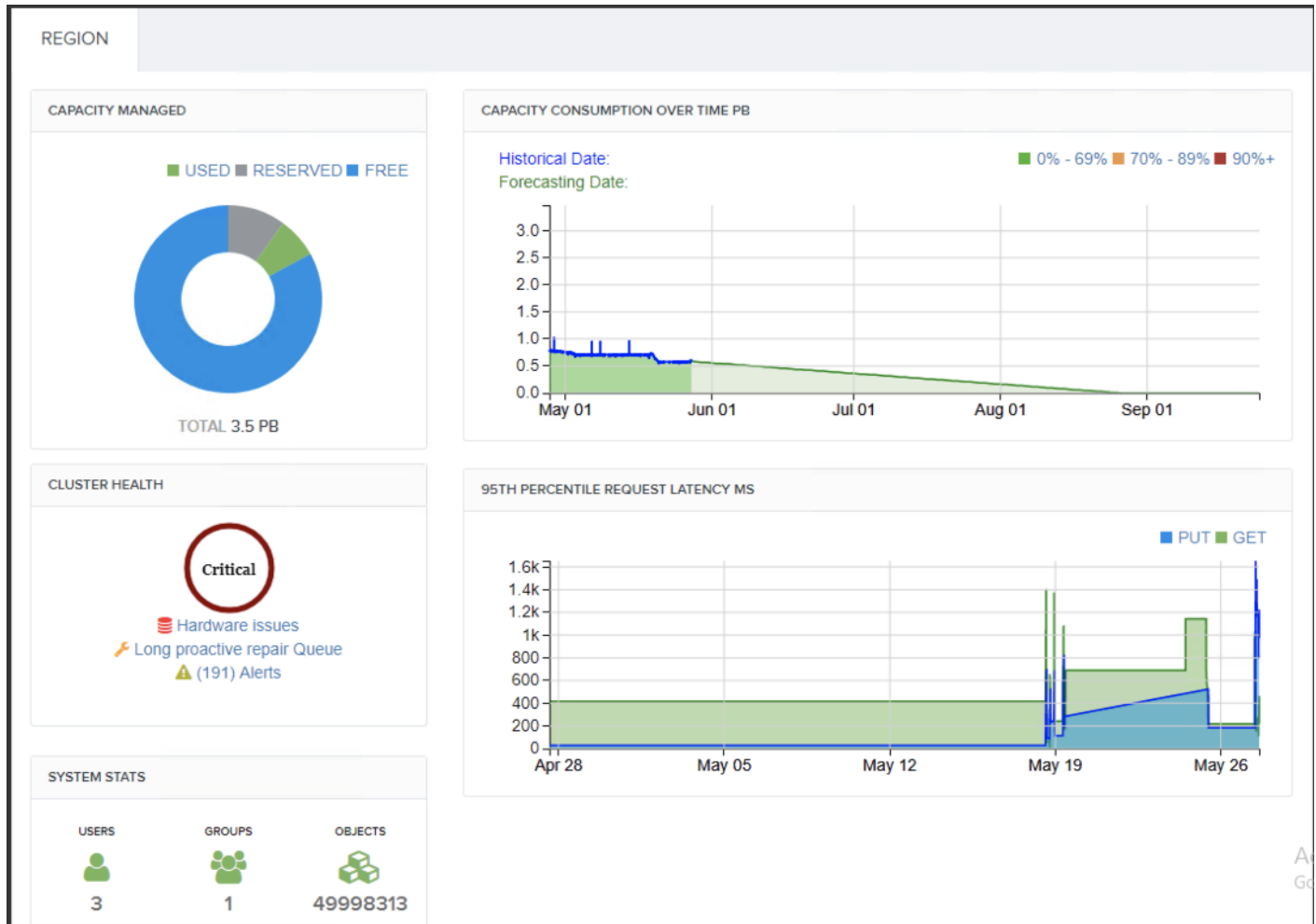
Success ratio was 100 percent for both read and write.

Op-Type	Op-Count	Byte-Count	Avg-ResTime	Avg-ProcTime	Throughput	Bandwidth	Succ-Ratio
op1:read	13.95 kops	139.51 GB	72.99 ms	41.58 ms	827.62 op/s	8.28 GB/S	100%
op2:write	13.99 kops	139.91 GB	484.76 ms	339.3 ms	832.7 op/s	8.33 GB/S	100%

The disk-3 was removed. After the disk is removed, some of the write operation were missed.

Op-Type	Op-Count	Byte-Count	Avg-ResTime	Avg-ProcTime	Throughput	Bandwidth	Succ-Ratio
op1:read	14.73 kops	147.31 GB	75.35 ms	43.07 ms	727.59 op/s	7.28 GB/S	100%
op2:write	14.78 kops	147.84 GB	465.66 ms	321.31 ms	730.21 op/s	7.3 GB/S	100%

Alert was reported in CMC:



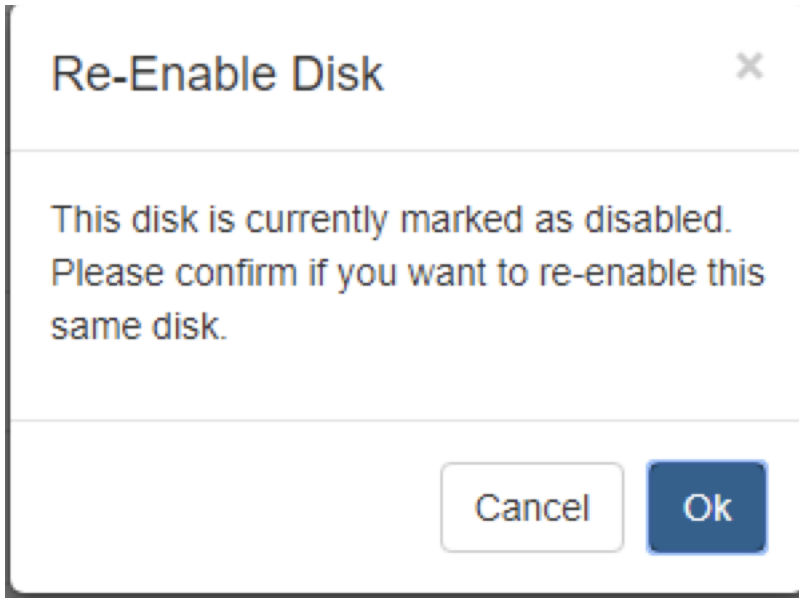
Disk is reported as unavailable.

	/dev/sdz	/cloudian26	NOTAVAIL	Disk usage info is unavailable
--	----------	-------------	----------	--------------------------------

CMC reported below alert:

<input type="checkbox"/>	SEVERITY	ALERT TYPE	ALERT TEXT
<input type="checkbox"/>	Critical	Disk Error	/cloudian26
<input type="checkbox"/>	High	Hyp erSt ore	HS180032 2019-05-27 14:41:26,606 ERROR[73af27ba-b178-12d7-8767-0025b500a01b][qtp1027319653-224] StorageHandler:HSn26/hsfs/16BYpG5XsrmEHr1CuP06O8/788fb6129270dbcd97aea04b33c581e2/142/069/4773967058080674946743558993286268...

After re-inserting the disk, the disk was re-enabled in CMC:



Mount points were re-enabled.

Interface for disk management with tabs: NODE STATUS, NODE ACTIVITY, ADVANCED.

Command Type:

Command: Target Node:

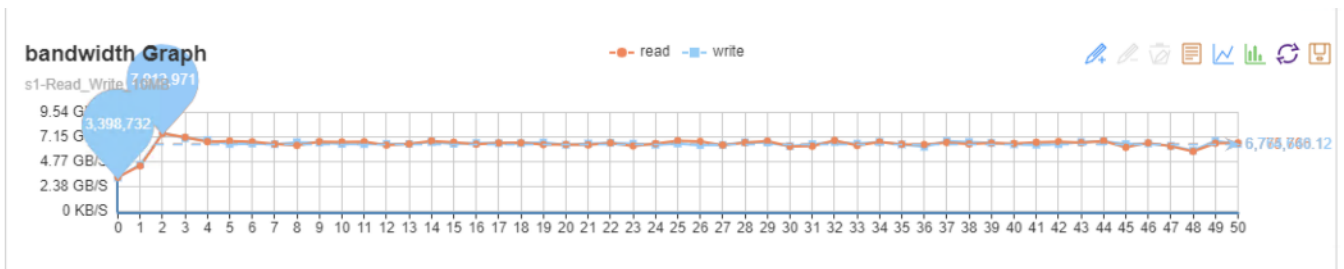
Mount Point:

Description: Re-enable mount point for an existing disk which is currently disabled. For more information about this operation, please see the online Help.

The disk was back online.

Storage status bar showing: /dev/sdz /cloudian26 HS 2.4 TB of 10.6 TB used

There was small dip in performance [At point 48] After that it was back to normal.



Frequently Asked Questions

1. What Cisco UCS Manager version is supported with Clouidian HyperStore on Cisco UCS?

Clouidian HyperSTORE has been validated with UCSM version 4.0(2B) and with 40Gb Fabric Interconnects and Switches. It is strongly recommended to run the infrastructure on versions higher than the validation done in this CVD.

2. What are the minimum number of nodes required?

The minimum number of storage nodes are 3. Thi can be scaled-up as your storage requiremts increases.

3. Can Clouidian HyperStore work without a load balancer?

Yes. Clouidian can run without a load balancer but high availability will be compromised unless using a virtual IP manager like CTDB.

4. Can Clouidian HyperStore work with MTU 4000?

Yes. Clouidian can run on a MTU size of 4000 but please be aware this id not an option when the ObjectStore is exposed to external clients that connect over the Internet as that will require and MTU size of 1500.

5. Can Clouidian HyperStore support multiple storage policies?

Yes. Clouidian supports 20 storage policies by default but this number can be increased, it can be a combination of erasure code and replication.

6. Does Clouidian support both eventual and strong consistency?

Yes. Clouidian support both eventual, strong and dynamic consistency to be able accommodate every requirement.

7. How many DC's can Clouidian HyperStore support?

Clouidian HyperStore supports up to 200 DC's within 20 regions.

8. Does Clouidian HyperStore support a single global namespace?

Yes. Clouidian supports one ore more single global namespace, and supports one or more individual name spaces as well.

9. Does Clouidian HyperStore support compression?

Yes. Clouidian supports snappy, lib and lz4 compression.

10. Can Clouidian HyperStore tier to another object store?

Yes. Clouidian HyperStore can tier to any other S3 compatible objectstore including AWS, s3, glacier, Google GCP, and Azure.

11. Does Clouidian HyperStore support heterogeneous nodes?

Yes. Cloudian HyperStore supports nodes from different vendors with different hardware components , performance characteristics and capacities.

12. Can I use Cloudian HyperStore as a backup target?

Yes. when using any of the major backup vendors that has an s3 connector, Cloudian HyperStore will be an excellent choice to store your backups.

13. Can I use Cloudian HyperStore as a media target for MAM?

Yes. Any MAM software that support s3 as a repository target will be a good use case for Cloudian.

Troubleshooting

1. Where to find the HyperStore the log files?

The log files can be found under `/var/log/cloudian`. The CMC also provides a log page that can be reviewed before acknowledging the errors

2. How can generate a log bundle to send to support?

To create log bundles to send to support for further analyses, run the following:

```
/opt/cloudian/tools/smartsup_systeminfo.sh on each node.
```

3. Read requests are failing specifically with larger objects

Please ensure NTP is properly configured and time is in sync on all nodes and clients.

4. A service has failed on my cluster

The failed service can be restarted from the CMC under that node or from the installer menu under the maintenance option.

5. S3 service does not successfully restart

When the S3 fails after restarting the service, stop the S3 service first and then start the S3 service again.

Appendix

Appendix A – Kickstart File of High Available Proxy Node for Cisco UCS C220 M5

```
#version=DEVEL

#from the linux installation menu, hit tab and append this:

#biosdevname=0 net.ifnames=0 ip=eth1:dhcp

#ks=ftp://192.168.10.2/{hostname}.cfg

# System authorization information
auth --enablesshadow --passalgo=sha512

# Use CDROM installation media
cdrom

# Use text install
text

# Run the Setup Agent on first boot
firstboot --disable

selinux --disable

firewall --disable

# Keyboard layouts
keyboard --vckeymap=us --xlayouts='us'

# System language
lang en_US.UTF-8

# Network information

network --bootproto=static --device=eth0 --ip=128.107.79.211 --netmask=255.255.255.0 --onboot=on --
gateway=128.107.79.1 --nameserver=171.70.168.183 --ipv6=auto --activate

network --bootproto=static --device=eth1 --ip=192.168.10.191 --netmask=255.255.255.0 --onboot=on --
ipv6=auto --activate

network --bootproto=static --device=eth2 --ip=192.168.20.191 --netmask=255.255.255.0 --onboot=on --
ipv6=auto --activate

network --bootproto=static --device=eth3 --ip=192.168.30.191 --netmask=255.255.255.0 --onboot=on --
ipv6=auto --activate
```

```
network --hostname=ha-proxy

# Root password

rootpw --iscrypted
$6$yfE2jHtdy.OSmO8g$InneiVXQI9Kc9m4w2cEiS8/og6BKUlu5HSR0eCYgh5dVaeCV54Q6piS7k10lalXignLCBvAZ
Pqmw4dvYgy66V1

# System services

services --disabled=" chronyd"

# System timezone

timezone America/Los_Angeles --isUtc --nontp

# System bootloader configuration

bootloader --append=" crashkernel=auto" --location=mbr --boot-drive=sda

# Partition clearing information

clearpart --drives=sda --all --initlabel

# Disk partitioning information

part /boot --fstype=" ext4" --ondisk=sda --size=8192
part swap --fstype=" swap" --ondisk=sda --size=32767
part /var --fstype=" ext4" --ondisk=sda --grow
part / --fstype=" ext4" --ondisk=sda --size=40960

reboot --eject

%packages
@^minimal
@core
kexec-tools

%end

%addon com_redhat_kdump --enable --reserve-mb='auto'
```

```
%end
```

```
%anaconda
```

```
pwpolicy root --minlen=6 --minquality=50 --notstrict --nochanges --notempty
```

```
pwpolicy user --minlen=6 --minquality=50 --notstrict --nochanges --notempty
```

```
pwpolicy luks --minlen=6 --minquality=50 --notstrict --nochanges --notempty
```

```
%end
```

```
#####
```

```
#POST SCRIPT
```

```
#####
```

```
%post --log=/root/ks-post.log
```

```
#####
```

```
#GPT Labels for HDDs
```

```
#####
```

```
for i in a b {d..z} aa ab ac; do parted -s /dev/sd$i mklabel gpt; done;
```

```
#####
```

```
#Turn off Transparent Hugepages and ensure that hyperthreading
```

```
#is turned off.
```

```
#####
```

```
grubby --update-kernel=ALL --args=" transparent_hugepage=never numa=off" ;
```

```
tuned-adm profile latency-performance;
```

```
systemctl enable ntpd;
```

```
#####
```

```
#Preconfigure /etc/hosts
```

```
#####
```

```
cat >> /etc/hosts <<EOF4
```

```
192.168.10.191    ha-proxy
```

```
192.168.10.185    storage-node1
```

```
192.168.10.186    storage-node2
```

```
192.168.10.187 storage-node3
192.168.10.188 storage-node4
192.168.10.189 storage-node5
192.168.10.190 storage-node6
```

EOF4

#####

#Setup ssh keys

#####

mkdir /root/.ssh;

cat > /root/.ssh/id_rsa <<EOF5

-----BEGIN RSA PRIVATE KEY-----

```
MIIEpAlBAAKCAQEAsYGqxWxQdGUsiUzafYLuX6MVD3mj3r6KaL0QcNSuZ8F3Xfw
7WJWJmhuu/rurLV0A90fjZDQY6aEAdHSH+o27mH6hfkMVqyunwQ6u3MtUqqkWRK2
NtEJqJBihZw9+bmgofyFYI5wBSWPGlig0kb8m+cBm0uRoE5SFFuAGc7usHkflFIO
QQd9vz9h6OX8ba3c6yUAZDzWSnt2udyLOTqV4SPpQY4O2NvYgm1VpblHvUvmP7Yu
5yl8hxn0in+RmferTq8WwyZihMV0EyN4q5HfT+gdbSY6xPMM9UHF89+IYNNxdZ4/
VuBcbBskey3UbQ332KqA7wS+Sra2DXmnfysWbwIDAQABAolBAQCbeRFUXiyR5IP9
5lyw9k9HYRX/OfGLLumSMnJyb1wzzP9cHcPeh/V8QihLadxHVZTHXZRxcHG19pFE
7rx2y7RVU2gUIDCkchd4nEG9EYKvF1u66GLE3I7zH5Nwj/sQkfAKMZ26rTC8sUsG
mBUUWKzE+k7Fklj6ud7WidZHxKH32ok1IEcFOsH/nK1BXR29XmQ/O/Kg2h0V/KiM
1Y9CJngpgghnybcDzlvP6LS8bEiRieHJGT5RTyDk+ad0uSv+f2YtlpvSUly7NAft
e1feAq3RWT82ZGyKTHWGTFNbfItcUjzPI/dcyS8AurYf+oQjJVAKhAl+yln7IUrL
V6xKsdYBAoGBANwNb96gJHZUeSoOP/JCnTps+MeOhT1vyrhRRZf1laFnEmX7hXmE
RKXaQUvGcOSPumZMkKYyqRN22B2PLM7n1D0ypKshRmk1eq6tZ/W9gkYfldn0+QAx
AAVfUA8vJm9XLgkCAE4o2BHvtQ1w63CfygoF4V3OAsQv677F6ltROeiBAoGBAM6A
9quEOrPiRDif25HnXXFUeRUXM4H77QB6WRV3AKggJjVIBXkhNt34g8Jr6/MfW4WO
SebQEwwBYH6NN7IG1Q0PeDRzrcv2voqzM7bV711rpc2E2BQhplcSyGr/aA6IW0OA
LI/HZldqb6OXXR8ImcP0rfxuqUJ8e6SHskG6qAbvAoGAlrw4QXMT7I3NNndDXtFn
EjbrWkzD+XuxC0FA9Aisw1aKz/BRFGptj6SRFA4B+gl6ETXay3FJwRnMaXYVQ5/S
n8pjteOtwqO/dt1GgMLmUn1NkaMavw39C9wMvijaL08apC9drvjBiqE8Bc4AvIm
```

```

KUJeVzIStHdABkAIQgCTXIECgYEAur6BU4YWmAnsa7kRyRZ7uDsN7Ha4y7mJED+U
RAcD/wZjxZF+C5ZvybgtXyq9i3U2DMcqKaLNNrQgERGF5kyrak4tBDIAX0zZ7xAz
mgplrw7kN8EErt/nTyLbP3eNIIGE0LwgM9IbHeKw5p3BRok+IKi2lmt0gX2VSqq0
FyC3Rt0CgYADqOJ53sV7NEXfd/NG5D9bzS5yCKW+KNH4fzxAoAYhMBo3nAkppa/1
rdjPH4f5bAMX6dKZCh5Sy9BFxgqbl0tdjVGZBUPK8tb0xabcnJ2F3+aLq02fCfyr+
TfYW1tZ7g7gZJ+To42h4Tv9wj8iWGe+pnR4Moh3WqM1TttuaCJf1nQ==
-----END RSA PRIVATE KEY-----

```

```
EOF5
```

```
cat > /root/.ssh/id_rsa.pub <<EOF6
```

```
ssh-rsa
```

```

AAAAB3NzaC1yc2EAAAADAQABAAQACxgarFbFB0ZSyJTNp9gu5foxUPeaOrevopovRBw1K5nwXdd/DtYlaOaG
67+u6stWgD3R+NkNBjpoQB0dlf6jbuYfqF+QxWrK6fBDq7cy1SqqTBERY20QmokGldnD35uaCh/IViXnAFJY8YiKDSR
vyb5wGbS5GgTIIUW4AZzu6weR8gWU5BB32/P2Ho5fxtrdzrJQBkPNZKe3a53ls5OpXhl+IBjg7Y29iCbVWluUe9S+Y
/ti7nKXyHGfSKf5GZ96tOrxbDJmKExXQTI3irkd9P6B1tJrE8wz1QcXz36Vg03F1nj9W4FxsGyR7LdRtDffYqoDvBL5Kt
rYNead/KxZv root@storage-node7

```

```
EOF6
```

```
cat > /root/.ssh/authorized_keys <<EOF7
```

```
ssh-rsa
```

```

AAAAB3NzaC1yc2EAAAADAQABAAQACxgarFbFB0ZSyJTNp9gu5foxUPeaOrevopovRBw1K5nwXdd/DtYlaOaG
67+u6stWgD3R+NkNBjpoQB0dlf6jbuYfqF+QxWrK6fBDq7cy1SqqTBERY20QmokGldnD35uaCh/IViXnAFJY8YiKDSR
vyb5wGbS5GgTIIUW4AZzu6weR8gWU5BB32/P2Ho5fxtrdzrJQBkPNZKe3a53ls5OpXhl+IBjg7Y29iCbVWluUe9S+Y
/ti7nKXyHGfSKf5GZ96tOrxbDJmKExXQTI3irkd9P6B1tJrE8wz1QcXz36Vg03F1nj9W4FxsGyR7LdRtDffYqoDvBL5Kt
rYNead/KxZv root@storage-node7

```

```
EOF7
```

```
chmod 700 /root/.ssh;
```

```
chmod 600 /root/.ssh/authorized_keys;
```

```
chmod 600 /root/.ssh/id_rsa;
```

```
chmod 644 /root/.ssh/id_rsa.pub;
```

```
#####
```

```
# Remove NetworkManager, a core package which is not needed.
```

```
yum -y remove NetworkManager;
```

```
%end
```

Appendix B – Kickstart File of Storage Nodes for Cisco UCS S3260 M5 Server

```
#For storage-node-1
```

```

#version=DEVEL

#from the linux installation menu, hit tab and append this:

#biosdevname=0 net.ifnames=0 ip=eth1:dhcp

#ks=ftp://192.168.10.2/{hostname}.cfg

# System authorization information

auth --enablshadow --passalgo=sha512

# Use CDROM installation media

cdrom

# Use text install

text

# Run the Setup Agent on first boot

firstboot --disable

selinux --disable

firewall --disable

# Keyboard layouts

keyboard --vckeymap=us --xlayouts='us'

# System language

lang en_US.UTF-8

# Network information

network --bootproto=static --device=eth0 --ip=173.36.220.240 --netmask=255.255.255.0 --onboot=on --
gateway=173.36.220.1 --nameserver=171.70.168.183 --ipv6=auto --activate

network --bootproto=static --device=eth1 --ip=192.168.100.240 --netmask=255.255.255.0 --onboot=on --
ipv6=auto --activate

network --bootproto=static --device=eth2 --ip=192.168.130.240 --netmask=255.255.255.0 --onboot=on --
ipv6=auto --activate

network --bootproto=static --device=eth3 --ip=192.168.120.240 --netmask=255.255.255.0 --onboot=on --
ipv6=auto --activate

network --hostname=storage-node1

# Root password

rootpw --iscrypted
$6$yE2jHtdy.OSmO8g$InneiVXQI9Kc9m4w2cEiS8/og6BKUlu5HSR0eCYgh5dVaeCV54Q6piS7k10lalXignLCBvAZ
Pqmw4dvYgy66V1

```

```
# System services
services --disabled=" chronyd"

# System timezone
timezone America/Los_Angeles --isUtc --nontp

# System bootloader configuration
bootloader --append=" crashkernel=auto" --location=mbr --boot-drive=sda

# Partition clearing information
clearpart --drives=sda --all --initlabel

# Disk partitioning information
part /boot --fstype=" ext4" --ondisk=sda --size=1024
part swap --fstype=" swap" --ondisk=sda --size=4096
part /var --fstype=" ext4" --ondisk=sda --grow
part / --fstype=" ext4" --ondisk=sda --grow

reboot --eject

%packages
@^minimal
@core
kexec-tools

%end

%addon com_redhat_kdump --enable --reserve-mb='auto'

%end

%anaconda
pwpolicy root --minlen=6 --minquality=50 --notstrict --nochanges --notempty
pwpolicy user --minlen=6 --minquality=50 --notstrict --nochanges --notempty
```



```
pwpolicy luks --minlen=6 --minquality=50 --notstrict --nochanges --notempty
```

```
%end
```

```
#####
```

```
#POST SCRIPT
```

```
#####
```

```
%post --log=/root/ks-post.log
```

```
#####
```

```
#GPT Labels for HDDs
```

```
#####
```

```
for i in a b {d..z} aa ab ac; do parted -s /dev/sd$i mklabel gpt; done;
```

```
#####
```

```
#Turn off Transparent Hugepages and ensure that hyperthreading
```

```
#is turned off.
```

```
#####
```

```
grubby --update-kernel=ALL --args=" transparent_hugepage=never numa=off nr_cpus=24" ;
```

```
tuned-adm profile latency-performance;
```

```
systemctl enable ntpd;
```

```
#####
```

```
#Preconfigure /etc/hosts
```

```
#####
```

```
cat >> /etc/hosts <<EOF4
```

```
192.168.100.240 storage-node1
```

```
192.168.100.241 storage-node2
```

```
192.168.100.242 storage-node3
```

```
192.168.100.243 storage-node4
```

```
192.168.100.244 storage-node5
```

```
192.168.100.245 storage-node6
```

```
192.168.100.246 storage-node7
```

```
192.168.100.247 storage-node8
```

```
192.168.100.248 storage-node9
192.168.100.249 storage-node10
192.168.100.250 storage-node11
192.168.100.251 storage-node12
```

EOF4

#####

#Setup ssh keys

#####

mkdir /root/.ssh;

cat > /root/.ssh/id_rsa <<EOF5

-----BEGIN RSA PRIVATE KEY-----

```
MIIEpAlBAAKCAQEAsYGqxWxQdGUsiUzafYLuX6MVD3mjq3r6KaL0QcNSuZ8F3Xfw
7WJWjmhUU/rurLVoa90fjZDQY6aEAdHSH+o27mH6hfkMVqyunwQ6u3MtUqqkWRK2
NtEJqJBihZw9+bmgofyFYI5wBSWPGlig0kb8m+cBm0uRoE5SFFuAGc7usHkflFIO
QQd9vz9h6OX8ba3c6yUAZDzWSnt2udyLOTqV4SPpQY4O2NvYgm1VpblHvUvmP7Yu
5yl8hxn0in+RmferTq8WwyZihMV0EyN4q5HfT+gdbSY6xPMM9UHF89+IYNNxdZ4/
VuBcbBskey3UbQ332KqA7wS+Sra2DXmnfysWbwIDAQABAolBAQCbeRFUXiyR5IP9
5lyw9k9HYRX/OfGLLumSMnJyb1wzzP9cHcPeh/V8QihLadxHVZTHXZRxcHG19pFE
7rx2y7RVU2gUIDCkchd4nEG9EYKvF1u66GLE3I7zH5Nwj/sQkfAKMZ26rTC8sUsG
mBUUWKzE+k7Fklj6ud7WidZHxKH32ok1IEcFOsH/nK1BXR29Xmq/O/Kg2h0V/KiM
1Y9CJngpgghnybcDzlvV6LS8bEiRieHJGT5RTyDk+ad0uSv+f2YtlpvSUly7NAft
e1feAq3RWT82ZGyKTHWGTFNbfItcUjzPI/dcyS8AurYf+oQjJVAKhAl+yln7IUrL
V6xKsdYBAoGBANwNb96gJHZUeSoOP/JCnTps+MeOhT1vyrhRRZf1laFnEmX7hXmE
RKXaQUvGcOSPumZMkKYyqRN22B2PLM7n1D0ypKshRmk1eq6tZ/W9gkYfldn0+QAx
AAVfUA8vJm9XLgkCAE4o2BHvtQ1w63CfygoF4V3OAsQv677F6ltROeiBAoGBAM6A
9quEOrPiRDif25HnXXFUeRUXM4H77QB6WRV3AKggJjVIBXkhNt34g8Jr6/MfW4WO
SebQEwwBYH6NN7IG1Q0PeDRzrcv2voqzM7bV711rpc2E2BQhplcSyGr/aA6IW0OA
LI/HZldqb6OXXR8ImcP0rfxuqUJ8e6SHskG6qAbvAoGAlrw4QXMT7I3NNndDXtFn
EjbrWkzD+XuxC0FA9Aisw1aKz/BRFGptj6SRFA4B+gl6ETXay3FJwRnMaXYVQ5/S
n8pjteOtwqO/dt1GgMLmUn1NkaMavw39C9wMvijaL08apC9drvjBiqE8Bc4AvIm
```

```

KUJeVzIStHdABkAIQgCTXIECgYEAur6BU4YWmAnsa7kRyRZ7uDsN7Ha4y7mJED+U
RAcD/wZjxZF+C5ZvybgtXyq9i3U2DMcqKaLNNrQgERGF5kyrak4tBDIAX0zZ7xAz
mgplrw7kN8EErt/nTyLbP3eNIIGE0LwgM9IbHeKw5p3BRok+IKi2lmt0gX2VSqq0
FyC3Rt0CgYADqOJ53sV7NEXfd/NG5D9bzS5yCKW+KNH4fzxAoAYhMBo3nAkppa/1
rdjPH4f5bAMX6dKZCh5Sy9BFxgqbl0tdjVGZBUPK8tb0xbcnJ2F3+aLq02fCfyr+
TfYW1tZ7g7gZJ+To42h4Tv9wj8iWGe+pnR4Moh3WqM1TttuaCJf1nQ==
-----END RSA PRIVATE KEY-----

```

EOF5

```
cat > /root/.ssh/id_rsa.pub <<EOF6
```

ssh-rsa

```

AAAAB3NzaC1yc2EAAAADAQABAAQACxgarFbFB0ZSyJTNp9gu5foxUPeaOrevopovRBw1K5nwXdd/DtYlaOaG
67+u6stWgD3R+NkNBjpoQB0dlf6jbuYfqF+QxWrK6fBDq7cy1SqqtBERY20QmokGldnD35uaCh/IViXnAFJY8YiKDSR
vyb5wGbS5GgTIIUW4AZzu6weR8gWU5BB32/P2Ho5fxtrdzrJQBkPNZKe3a53ls5OpXhl+IBjg7Y29iCbVWluUe9S+Y
/ti7nKXyHGfSKf5GZ96tOrxbDJmKExXQTi3irkd9P6B1tJrE8wz1QcXz36Vg03F1nj9W4FxsGyR7LdRtDffYqoDvBL5Kt
rYNead/KxZv root@storage-node7

```

EOF6

```
cat > /root/.ssh/authorized_keys <<EOF7
```

ssh-rsa

```

AAAAB3NzaC1yc2EAAAADAQABAAQACxgarFbFB0ZSyJTNp9gu5foxUPeaOrevopovRBw1K5nwXdd/DtYlaOaG
67+u6stWgD3R+NkNBjpoQB0dlf6jbuYfqF+QxWrK6fBDq7cy1SqqtBERY20QmokGldnD35uaCh/IViXnAFJY8YiKDSR
vyb5wGbS5GgTIIUW4AZzu6weR8gWU5BB32/P2Ho5fxtrdzrJQBkPNZKe3a53ls5OpXhl+IBjg7Y29iCbVWluUe9S+Y
/ti7nKXyHGfSKf5GZ96tOrxbDJmKExXQTi3irkd9P6B1tJrE8wz1QcXz36Vg03F1nj9W4FxsGyR7LdRtDffYqoDvBL5Kt
rYNead/KxZv root@storage-node7

```

EOF7

```
chmod 700 /root/.ssh;
```

```
chmod 600 /root/.ssh/authorized_keys;
```

```
chmod 600 /root/.ssh/id_rsa;
```

```
chmod 644 /root/.ssh/id_rsa.pub;
```

```
#####
```

```
# Remove NetworkManger, a core package which is not needed.
```

```
yum -y remove NetworkManager;
```

```
%end
```

Summary

Cisco UCS Infrastructure for Clouidian Software Defined Storage is an integrated solution to deploy Clouidian HyperStore and combines the value of Intel Xeon architecture, Cisco data center hardware and software, along with Red Hat Linux. This solution increases the speed of deployment and reduces the risk of scaling from proof-of-concept to full-enterprise production, and is validated and supported by Cisco and Clouidian.

Cisco UCS hardware with Cisco UCS Manager Software brings an integrated, scalable, multi-chassis platform in which all resources participate in a unified management domain. Creating and cloning service profiles from its templates and maintaining the hardware from a single pane of glass not only provides rapid provisioning of hardware but also makes management and firmware upgrades simpler.

Clouidian HyperStore software makes it easy to build fully featured, Amazon S3-compliant cloud storage, on-premise. Clouidian HyperStore software ensures unlimited scale, multi-data center storage, fully automated data tiering, and support for all S3 applications—all behind your firewall.

Clouidian HyperStore software deployed on UCS S-Series servers, combines robust availability with system management control, monitoring capabilities and reporting. A host of features, including hybrid cloud streaming, virtual nodes, configurable erasure coding, and data compression and encryption sets Clouidian apart with highly efficient storage and seamless data management that lets you store and access your data where you want it, when you want it. Built on a robust object storage platform for effortless data sharing, cloud service providers around the world use Clouidian HyperStore to deploy and manage both public and private clouds, while enterprises rely on it to maintain their private and hybrid clouds.

This Cisco Validated Design is a partnership of Cisco Systems and Clouidian. Combining these technologies, expertise and experience in the field, you are able to provide an enterprise-ready hardware and software solution.

About the Authors

Paniraja Koppa, Cisco Systems, Inc.

Paniraja Koppa is a Technical Marketing Engineer for UCS Solutions. In his current role at Cisco Systems, he works on best practices, optimization, automation and performance tuning of software defined storage solutions on Cisco UCS platforms. He has more than 13 years of experience with a primary focus on data center technologies such as Servers, Storage, Operating systems, Automation, Virtualization and Cloud. Prior to this, he has led QA efforts for 4 new virtual adapter card's firmware and software features for Cisco UCS. He also worked as customer advocate in the Data Center Virtualization space.

Muhammad Ashfaq, Cisco Systems, Inc.

Muhammad Ashfaq is a Technical Marketing Engineer in Cisco UCS and Data Center Solutions Group. He has over 10 years of experience in IT Infrastructure, Server Virtualization, and Cloud Computing. His current role includes building Cloud Computing, Software defined Storage, Automation and Management, Converged and Hyper-Converged Solutions on Cisco UCS platforms. Muhammad also holds Cisco Internetwork Expert Data Center Certification (CCIE-DC).

Eddo Jansen, Cloudian, Inc

Eddo Jansen is Principal Architect at Cloudian. He has over 15 years of experience in IT Infrastructure, Storage, Virtualization and automation. His current role is building performant, scalable, highly available, and durable object store solutions with specialties in Performance testing, analyzing, troubleshooting and tuning.

Acknowledgements

For their support and contribution to the design, validation, and creation of this Cisco Validated Design, the authors would like to thank:

- Chris O'Brien, Cisco Systems, Inc.
- Samuel Nagalingam, Cisco Systems, Inc.
- Jawwad Memon, Cisco Systems, Inc.
- Sanjay Jagad, Cloudian