

Cisco UCS C240 M5 with Red Hat Ceph Storage 4

Design and Deployment Guide for Red Hat Ceph Storage 4 on Cisco UCS C240 M5 with Cisco Intersight and Terraform

Published: April 2021



In partnership with:



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 Inter-network 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, Giga-Drive, 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. LDW2.

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)

© 2021 Cisco Systems, Inc. All rights reserved.

Executive Summary

Cisco Validated Designs 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.

The purpose of this document is to describe the design and deployment of Red Hat Ceph Storage on the latest generation of Cisco UCS C240 Rack Servers. This validated design provides the framework of designing and deploying Red Hat Ceph SDS software on Cisco UCS C240 Rack Servers together with Cisco Intersight. The Cisco Unified Computing System provides the storage, network, and storage access components for Red Hat Ceph Storage, deployed as a single cohesive system.

The Cisco Validated Design describes how the Cisco Unified Computing System can be used in conjunction with the latest release of Red Hat Ceph Storage. With the continuous evolution of Software Defined Storage (SDS), there has been increased demand to have Red Hat Ceph Storage solutions validated on Cisco UCS servers. The Cisco UCS C240 Rack Server, originally designed for the data center, together with Red Hat Ceph Storage is optimized for such object storage solutions, making it an excellent fit for unstructured data workloads such as active archive, backup, and cloud data. The Cisco UCS C240 Rack Server delivers a complete infrastructure with exceptional scalability for computing and storage resources together with 25 Gigabit Ethernet networking.

Cisco and Red Hat are collaborating to offer customers a scalable object storage solution for unstructured data that is integrated with Red Hat Ceph Storage. With the power of the Cisco Intersight 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.

Solution Overview

Introduction

Traditional storage systems are limited in their ability to scale easily and cost-effectively to support large amounts of unstructured data. With about 80 percent of data being unstructured, new approaches using x86 servers are proving to be more cost effective, providing storage that can be expanded as easily as your data grows. Software Defined Storage is a scalable and cost-effective approach for handling large amounts of data.

But more and more there are requirements to store unstructured data even in smaller quantities as object storage. The advantage of identifying the data by metadata and not taking over management of the location is very attractive even for smaller quantities. As a result, new technologies need to be developed to provide similar levels of availability and reliability as large scale-out object storage solutions.

Organizations are starting to understand the insights and opportunities that effective data management can present to their businesses. More than just accommodating the growing need for storage, data now offers an opportunity to disrupt existing competitive business models by facilitating continuous innovation.

Cisco and Red Hat offer a solution, which solves the problem of connecting storage and managing data effectively. Cisco with Cisco UCS provides an enterprise-grade compute, network, and storage infrastructure, building the foundation for Red Hat Ceph Storage. To offer a more intelligent level of management that enables IT organizations to analyze, simplify, and automate their environments, Cisco Intersight as a Cisco's systems management platform plays a major role in building and managing the infrastructure. With Terraform for building, changing, and versioning infrastructure safely and efficiently, Terraform is an ideal tool for building and managing these hybrid cloud storage infrastructures.

Red Hat Ceph Storage provides a robust and compelling data storage solution that can support customer data, no matter the format or origin. As a self-healing, self-managing platform with no single point of failure, Red Hat Ceph Storage significantly lowers the cost of storing enterprise data and helps companies manage exponential data growth in an automated fashion. Red Hat Ceph Storage is optimized for large installations—efficiently scaling to multiple petabytes or greater. Unlike traditional network-attached storage (NAS) and storage area network (SAN) approaches, it does not become dramatically more expensive as a cluster grows. Red Hat Ceph Storage also supports increasingly popular containerized environments such as Red Hat OpenShift Container Platform.

This document describes the architecture, design, and deployment procedures of Red Hat Ceph Storage on Cisco UCS C240 M5 servers together Cisco Intersight and Terraform.

Audience

The intended audience for this document includes, but is not limited to, sales engineers, field consultants, professional services, IT managers, partner engineering, and customers who want to deploy Red Hat Ceph Storage on Cisco UCS C240 M5 Servers with Cisco Intersight and Terraform.

Purpose of this Document

This document describes how to deploy Red Hat Ceph Storage with Cisco Intersight and Terraform on Cisco UCS C240 M5 Servers.

It presents a tested and validated solution and provides insight into operational best practices.

What's New in this Release?

This is a new document and contains the following:

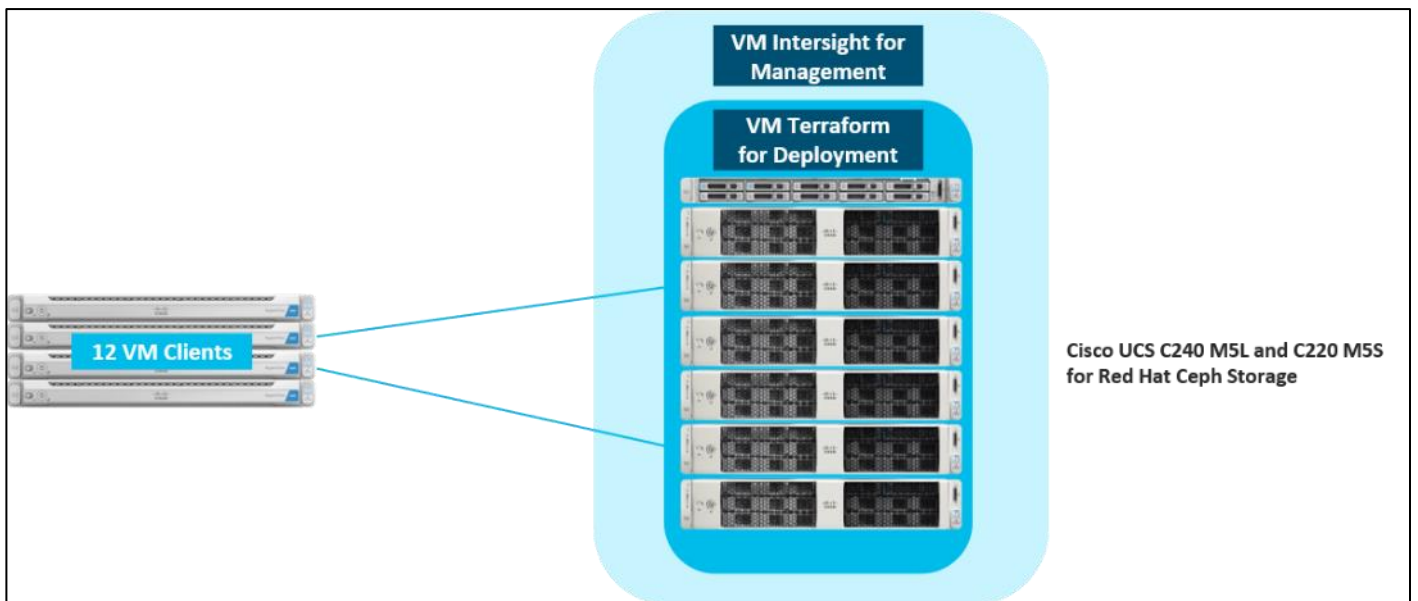
- Cisco Intersight Virtual Appliance
- Terraform Provider for Cisco Intersight
- Red Hat Ceph Storage 4.2

This revision of the CVD focuses on the latest release of Red Hat Ceph Storage 4.2, a massively scalable, open, software-defined storage platform that combines the most stable version of the Ceph storage system with a Ceph management platform, deployment utilities, and support services.

Solution Summary

In this architecture we have deployed Red Hat Ceph Storage on Cisco UCS C240 M5 with Cisco Intersight. The deployment of Cisco UCS C240 M5 on Cisco Intersight was done through the Terraform provider for Cisco Intersight. We used 12 virtual clients to test the performance of the whole cluster.

Figure 1. High-level Overview



The configuration uses the following architecture for the deployment:

- 6 x Cisco UCS C240 M5L
- 1 x Cisco UCS C220 M5S
- 2 x Cisco Nexus 93180YC-FX

In addition, a 4-node Cisco HyperFlex Edge cluster was used for the virtual machines Cisco Intersight and Terraform as well as 12 Linux clients for performance benchmarking.

Technology Overview

Cisco Unified Computing System

Cisco Unified Computing System (Cisco UCS) 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 Scalable processors. Cisco UCS servers offer the patented Cisco Extended Memory Technology to support applications with large datasets and allow more virtual machines (VM) per server.
- **Network** - The system is integrated onto a low-latency, lossless, 10/25/40/100-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, the 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.

The 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 C240 Rack Server

The Cisco UCS C240 Rack Server is a 2-socket, 2-Rack-Unit (2RU) rack server offering industry-leading performance and expandability. It supports a wide range of storage and I/O-intensive infrastructure workloads, from big data and analytics to collaboration. Cisco UCS C-Series Rack Servers can be deployed as standalone servers or as part of a Cisco UCS managed environment to take advantage of Cisco's standards-based unified computing innovations that help reduce customers' TCO and increase their business agility.

Figure 2. Cisco UCS C240 Rack Server



In response to ever-increasing computing and data-intensive real-time workloads, the enterprise-class Cisco UCS C240 server extends the capabilities of the Cisco UCS portfolio in a 2RU form factor. It incorporates the Intel® Xeon® Scalable processors, supporting up to 20 percent more cores per socket, twice the memory capacity, and five times more Non-Volatile Memory Express (NVMe) PCI Express (PCIe) 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. The Cisco UCS C240 M5 delivers outstanding levels of storage expandability with exceptional performance, comprised of the following:

- The latest second-generation Intel Xeon Scalable CPUs, with up to 28 cores per socket
- Supports the first-generation Intel Xeon Scalable CPU, with up to 28 cores per socket
- Support for the Intel Optane DC Persistent Memory (128G, 256G, 512G)
- Up to 24 DDR4 DIMMs for improved performance including higher density DDR4 DIMMs
- Up to 26 hot-swappable Small-Form-Factor (SFF) 2.5-inch drives, including 2 rear hot-swappable SFF drives (up to 10 support NVMe PCIe SSDs on the NVMe-optimized chassis version), or 12 Large-Form-Factor (LFF) 3.5-inch drives plus 2 rear hot-swappable SFF drives
- Support for 12-Gbps SAS modular RAID controller in a dedicated slot, leaving the remaining PCIe Generation 3.0 slots available for other expansion cards
- Modular LAN-On-Motherboard (mLOM) slot that can be used to install a Cisco UCS Virtual Interface Card (VIC) without consuming a PCIe slot, supporting dual 10-, 25- or 40-Gbps network connectivity
- Dual embedded Intel x550 10GBASE-T LAN-On-Motherboard (LOM) ports
- Modular M.2 or Secure Digital (SD) cards that can be used for boot

The Cisco UCS C240 rack server is well suited for a wide range of enterprise workloads, including:

- Object Storage
- Big Data and analytics
- Collaboration
- Small and medium-sized business databases
- Virtualization and consolidation
- Storage servers
- High-performance appliances

Cisco UCS C240 rack servers can be deployed as standalone servers or in a Cisco UCS managed environment. When used in combination with Cisco UCS Manager, the Cisco UCS C240 brings the power and automation of

unified computing to enterprise applications, including Cisco SingleConnect technology, drastically reducing switching and cabling requirements.

Cisco UCS Manager uses service profiles, templates, and policy-based management to enable rapid deployment and help ensure deployment consistency. It also enables end-to-end server visibility, management, and control in both virtualized and bare-metal environments.

The Cisco Integrated Management Controller (IMC) delivers comprehensive out-of-band server management with support for many industry standards, including:

- Redfish Version 1.01 (v1.01)
- Intelligent Platform Management Interface (IPMI) v2.0
- Simple Network Management Protocol (SNMP) v2 and v3
- Syslog
- Simple Mail Transfer Protocol (SMTP)
- Key Management Interoperability Protocol (KMIP)
- HTML5 GUI
- HTML5 virtual Keyboard, Video, and Mouse (vKVM)
- Command-Line Interface (CLI)
- XML API

Management Software Development Kits (SDKs) and DevOps integrations exist for Python, Microsoft PowerShell, Ansible, Puppet, Chef, and more. For more information about integrations, see Cisco DevNet (<https://developer.cisco.com/site/ucs-dev-center/>).

The Cisco UCS C240 is Cisco Intersight™ ready. Cisco Intersight is a new cloud-based management platform that uses analytics to deliver proactive automation and support. By combining intelligence with automated actions, you can reduce costs dramatically and resolve issues more quickly.

Cisco UCS C220 Rack Server

The Cisco UCS C220 M5 Rack 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 Servers can be deployed as standalone servers or as part of the Cisco Unified Computing System (Cisco UCS) to take advantage of Cisco's standards-based unified computing innovations that help reduce customers' Total Cost of Ownership (TCO) and increase their business agility.

Figure 3. Cisco UCS C220 Rack Server



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. The C220 M5 delivers outstanding levels of expandability and performance in a compact package, with:

- Latest (second generation) Intel Xeon Scalable CPUs with up to 28 cores per socket
- Supports first-generation Intel Xeon Scalable CPUs with up to 28 cores per socket
- Up to 24 DDR4 DIMMs for improved performance
- Support for the Intel Optane DC Persistent Memory (128G, 256G, 512G)
- Up to 10 Small-Form-Factor (SFF) 2.5-inch drives or 4 Large-Form-Factor (LFF) 3.5-inch drives (77 TB storage capacity with all NVMe PCIe SSDs)
- Support for 12-Gbps SAS modular RAID controller in a dedicated slot, leaving the remaining PCIe Generation 3.0 slots available for other expansion cards
- Modular LAN-On-Motherboard (mLOM) slot that can be used to install a Cisco UCS Virtual Interface Card (VIC) without consuming a PCIe slot
- Dual embedded Intel x550 10GBASE-T LAN-On-Motherboard (LOM) ports

The Cisco UCS C220 M5 Rack Server is well-suited for a wide range of workloads, including:

- IT and web infrastructure
- High-performance virtual desktops
- Medium-sized and distributed databases
- Middleware
- Collaboration
- Public cloud

Cisco UCS C220 M5 servers can be deployed as standalone servers or in a Cisco UCS managed environment. When used in combination with Cisco UCS Manager, the Cisco UCS C220 M5 brings the power and automation of unified computing to enterprise applications, including Cisco SingleConnect technology, drastically reducing switching and cabling requirements.

Cisco UCS Manager uses service profiles, templates, and policy-based management to enable rapid deployment and help ensure deployment consistency. It also enables end-to-end server visibility, management, and control in both virtualized and bare-metal environments.

The Cisco Integrated Management Controller (IMC) delivers comprehensive out-of-band server management with support for many industry standards, including:

- Redfish Version 1.01 (v1.01)
- Intelligent Platform Management Interface (IPMI) v2.0
- Simple Network Management Protocol (SNMP) v2 and v3
- Syslog

- Simple Mail Transfer Protocol (SMTP)
- Key Management Interoperability Protocol (KMIP)
- HTML5 GUI
- HTML5 virtual Keyboard, Video, and Mouse (vKVM)
- Command-Line Interface (CLI)
- XML API

Management Software Development Kits (SDKs) and DevOps integrations exist for Python, Microsoft PowerShell, Ansible, Puppet, Chef, and more. For more information about integrations, see Cisco DevNet (<https://developer.cisco.com/site/ucs-dev-center/>).

The Cisco UCS C220 M5 is Cisco Intersight ready. Cisco Intersight is a new cloud-based management platform that uses analytics to deliver proactive automation and support. By combining intelligence with automated actions, you can reduce costs dramatically and resolve issues more quickly.

Cisco UCS Virtual Interface Card 1455

The Cisco UCS VIC 1455 is a quad-port Small Form-Factor Pluggable (SFP28) half-height PCIe card designed for the M5 generation of Cisco UCS C-Series Rack Servers. The card supports 10/25-Gbps Ethernet or FCoE. The card can present PCIe standards-compliant interfaces to the host, and these can be dynamically configured as either NICs or HBAs.

Figure 4. Cisco UCS Virtual Interface Card 1455



The Cisco UCS VIC 1400 series 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 Worldwide 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.

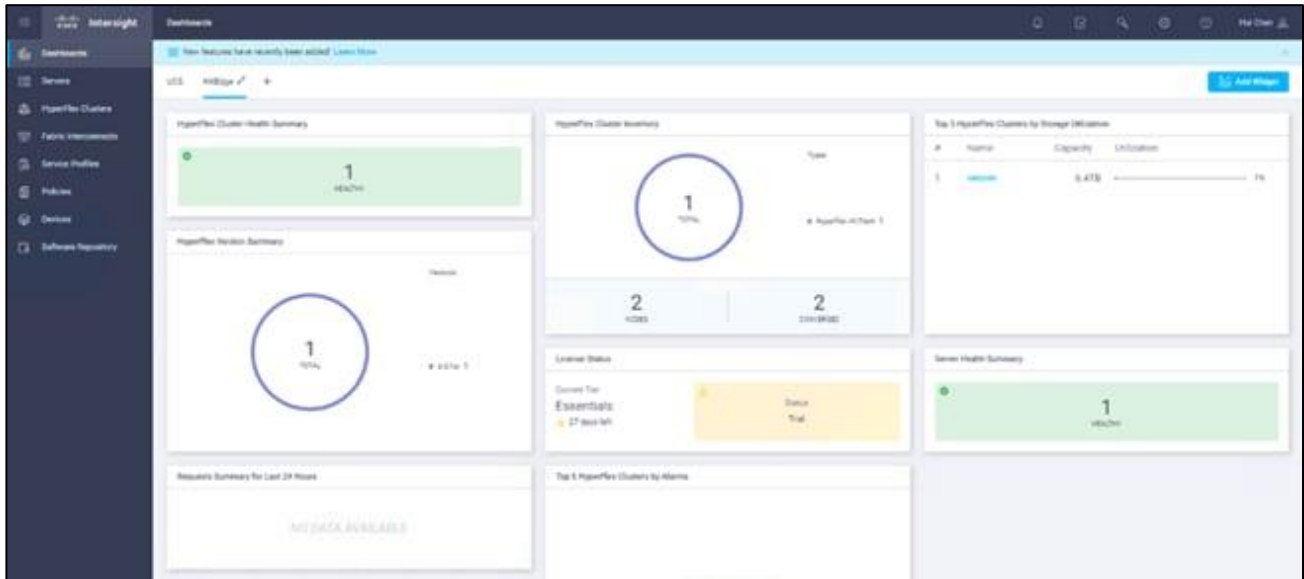
-
- Network interface virtualization: Each PCIe interface created on the VIC is associated with an interface on the Cisco UCS fabric interconnect, providing complete network separation for each virtual cable between a PCIe device on the VIC and the interface on the Fabric Interconnect.

Cisco Intersight

Cisco Intersight (<https://intersight.com>) is an API driven, cloud-based system management platform. It is designed to help organizations to achieve their IT management and operations with a higher level of automation, simplicity, and operational efficiency. It is a new generation of global management tool for the Cisco Unified Computing System (Cisco UCS) and Cisco HyperFlex systems and provides a holistic and unified approach to managing the customers' distributed and virtualized environments. Cisco Intersight simplifies the installation, monitoring, troubleshooting, upgrade, and support for your infrastructure with the following benefits:

- **Cloud Based Management:** The ability to manage Cisco UCS and HyperFlex from the cloud provides the customers the speed, simplicity, and easy scaling in the management of their infrastructure whether in the datacenters or remote and branch office locations.
- **Automation:** Unified API in Cisco UCS and Cisco HyperFlex systems enables policy driven configuration and management of the infrastructure and it makes Intersight itself and the devices connected to it fully programmable and DevOps friendly.
- **Analytics and Telemetry:** Intersight monitors the health and relationships of all the physical and virtual infrastructure components. It also collects telemetry and configuration information for developing the intelligence of the platform in the way in accordance with Cisco information security requirements.
- **Connected TAC:** Solid integration with Cisco TAC enables more efficient and proactive technical support. Intersight provides enhanced operations automation by expediting sending files to speed troubleshooting.
- **Recommendation Engine:** Driven by analytics and machine learning, Intersight recommendation engine provides actionable intelligence for IT operations management from daily increasing knowledge base and practical insights learned in the entire system.
- **Management as A Service:** Cisco Intersight provides management as a service and is designed to be infinitely scale and easy to implement. It relieves users of the burden of maintaining systems management software and hardware.

Figure 5. Cisco Intersight

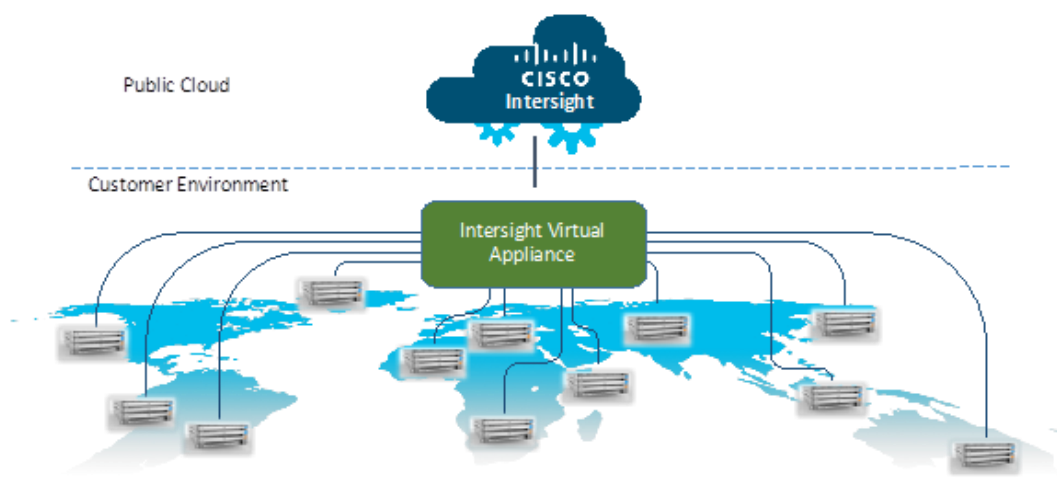


Intersight Virtual Appliance

The Cisco Intersight Virtual Appliance delivers the management features of Intersight for Cisco UCS and HyperFlex into the on-premise environment. It is deployed from a VMware OVA that enables the additional control to specify what data is sent back to Cisco with a single point of egress within the enterprises network. The virtual appliance form factor enables additional data locality, security, or compliance needs that are not completely met by connecting directly to intersight.com in the cloud. However, The Cisco Intersight Virtual Appliance is not intended for an environment with no external connectivity, the Cisco Intersight virtual appliance requires an internet connection back to Cisco and the cloud-based Intersight services for updates and to deliver some of the product features. Communication back to Cisco can be redirected via a proxy server if direct connectivity is not available or allowed by policy. Updates to the virtual appliance are automated and applied during a user specified recurring maintenance window. This connection also facilitates the streamlining of Cisco TAC services for Cisco UCS and HyperFlex systems, with features like automated support log collection.

Cisco Intersight Virtual Appliance OVA can be downloaded from Cisco website and can be deployed as a virtual machine in your existing environment. Cisco Intersight Virtual Appliance uses a subscription-based license delivered via Cisco Smart Licensing. After the installation of the appliance OVA is completed, you must connect the appliance to Cisco Intersight, and register the license as part of the initial setup process.

Figure 6. Cisco Intersight Virtual Appliance



Cisco Nexus 93180YC-FX

The Cisco Nexus® 9300-FX Series switches belongs to the fixed Cisco Nexus 9000 platform based on [Cisco Cloud Scale technology](#). The platform supports cost-effective cloud-scale deployments, an increased number of endpoints, and cloud services. The platform is built on modern system architecture designed to provide high performance and meet the evolving needs of highly scalable data centers and growing enterprises.

Cisco Nexus 9300-FX series switches offer a variety of interface options to transparently migrate existing data centers from 100-Mbps, 1-Gbps, and 10-Gbps speeds to 25-Gbps at the server, and from 10- and 40-Gbps speeds to 50- and 100-Gbps at the aggregation layer. The platforms provide investment protection for customers, delivering large buffers, highly flexible Layer 2 and Layer 3 scalability, and performance to meet the changing needs of virtualized data centers and automated cloud environments.

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

The Cisco Nexus 93180YC-FX Switch is a 1-Rack-Unit (1RU) switch with latency of less than 1 microsecond that supports 3.6 Terabits per second (Tbps) of bandwidth and over 1.2 billion packets per second (bps). The 48 downlink ports on the 93180YC-FX are capable of supporting 1-, 10-, or 25-Gbps Ethernet or as 16-, 32-Gbps Fibre Channel ports, creating a point of convergence for primary storage, compute servers, and back-end storage resources at the top of rack. The uplink can support up to six 40- and 100-Gbps ports, or a combination of 1-, 10-, 25-, 40, 50-, and 100-Gbps connectivity, offering flexible migration options. The switch has IEEE compliant, FC-FEC and RS-FEC enabled for 25-Gbps support. All ports support wire-rate MACsec encryption. Please see the Licensing guide section to enable features on the platform.

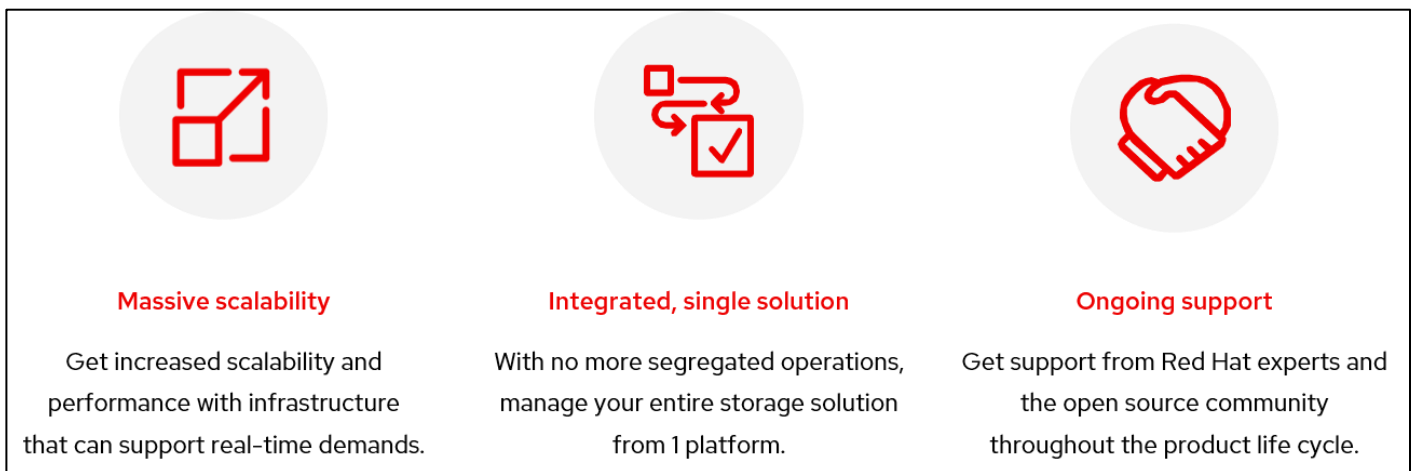
Figure 7. Cisco Nexus 93180 YC-FX



Red Hat Ceph Storage 4

Organizations are starting to understand the insights and opportunities that effective data management can present to their businesses. More than just accommodating the growing need for storage, data now offers an opportunity to disrupt existing competitive business models by facilitating continuous innovation.

Figure 8. Benefits of Red Hat Ceph Storage



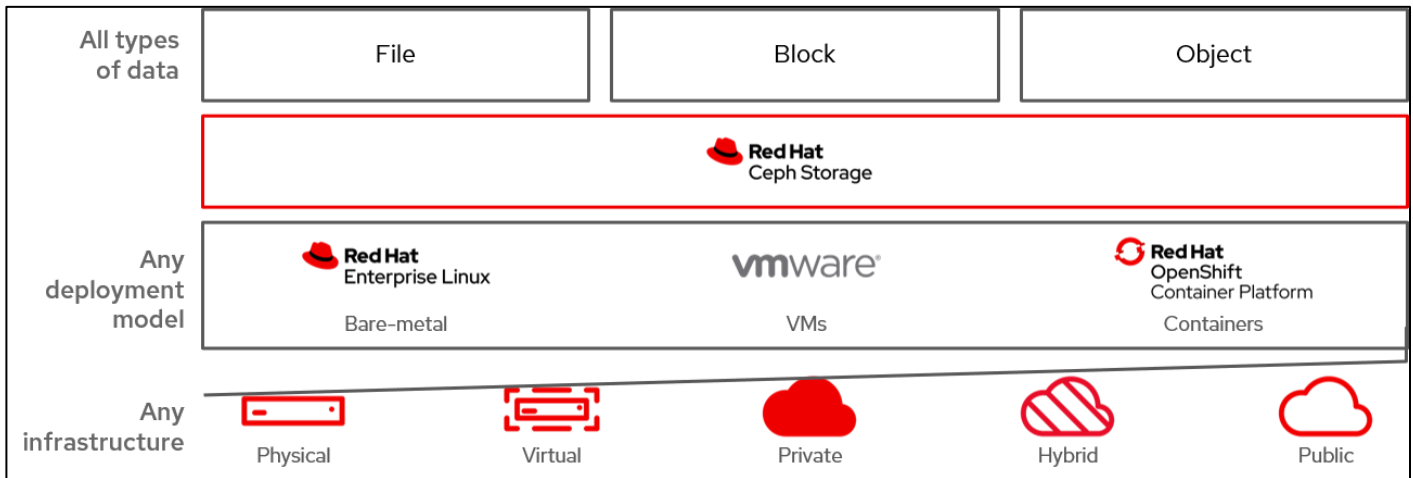
Red Hat Ceph Storage provides a robust and compelling data storage solution that can support your data, no matter the format or origin. As a self-healing, self-managing platform with no single point of failure, Red Hat Ceph Storage significantly lowers the cost of storing enterprise data and helps companies manage exponential data growth in an automated fashion. Red Hat Ceph Storage is optimized for large installations—efficiently scaling to multiple petabytes or greater. Unlike traditional network-attached storage (NAS) and storage area network (SAN) approaches, it does not become dramatically more expensive as a cluster grows. Red Hat Ceph Storage also supports increasingly popular containerized environments such as Red Hat OpenShift Container Platform.

Red Hat Ceph Storage is suitable for a wide range of storage workloads, including:

- **Data analytics and artificial intelligence/machine learning (AI/ML).** As a data lake, Red Hat Ceph Storage uses object storage to deliver massive scalability and high availability to support demanding multi-tenant analytics and AI/ML workloads.
- **Object storage-as-a-service.** Red Hat Ceph Storage is ideal for implementing an object storage service, with proven scalability and performance for both small and large object storage.

- **Hybrid cloud applications.** With support for the Amazon Web Services (AWS) Simple Storage Service (S3) interface, applications can access their storage with the same application programming interface (API)—in public, private, or hybrid clouds.
- **OpenStack applications.** Red Hat Ceph Storage offers scalability for OpenStack deployments, including Red Hat OpenStack Platform.
- **Backups.** A growing list of software vendors have certified their backup applications with Red Hat Ceph Storage, making it easy to use a single storage technology to serve a wide variety of performance-optimized workloads.

Figure 9. Red Hat Ceph Storage Portfolio



Red Hat Ceph Storage cluster is a distributed data object store designed to provide excellent performance, reliability, and scalability. Distributed object stores are the future of storage because they accommodate unstructured data, and because clients can use modern object interfaces and legacy interfaces simultaneously.

For example:

- APIs in many languages (C/C++, Java, Python)
- RESTful interfaces (S3/Swift)
- Block device interface
- Filesystem interface

The power of Red Hat Ceph Storage cluster can transform your organization's IT infrastructure and your ability to manage vast amounts of data, especially for cloud computing platforms like RHEL OSP. Red Hat Ceph Storage cluster delivers extraordinary scalability—thousands of clients accessing petabytes to exabytes of data and beyond.

At the center of every Ceph deployment is the Red Hat Ceph Storage cluster. It consists of three types of daemons:

-
- **Ceph OSD Daemon:** Ceph OSDs store data on behalf of Ceph clients. Additionally, Ceph OSDs utilize the CPU, memory, and networking of Ceph nodes to perform data replication, erasure coding, rebalancing, recovery, monitoring and reporting functions.
 - **Ceph Monitor:** A Ceph Monitor maintains a master copy of the Red Hat Ceph Storage cluster map with the current state of the Red Hat Ceph Storage cluster. Monitors require high consistency and use Paxos to ensure agreement about the state of the Red Hat Ceph Storage cluster.
 - **Ceph Manager:** The Ceph Manager maintains detailed information about placement groups, process metadata and host metadata in lieu of the Ceph Monitor—significantly improving performance at scale. The Ceph Manager handles execution of many of the read-only Ceph CLI queries, such as placement group statistics. The Ceph Manager also provides the RESTful monitoring APIs.

Ceph client interfaces read data from and write data to the Red Hat Ceph Storage cluster. Clients need the following data to communicate with the Red Hat Ceph Storage cluster:

- The Ceph configuration file, or the cluster name (usually ceph) and the monitor address
- The pool name
- The username and the path to the secret key

Ceph clients maintain object IDs and the pool names where they store the objects. However, they do not need to maintain an object-to-OSD index or communicate with a centralized object index to look up object locations. To store and retrieve data, Ceph clients access a Ceph Monitor and retrieve the latest copy of the Red Hat Ceph Storage cluster map. Then, Ceph clients provide an object name and pool name to librados, which computes an object's placement group and the primary OSD for storing and retrieving data using the CRUSH (Controlled Replication Under Scalable Hashing) algorithm. The Ceph client connects to the primary OSD where it may perform read and write operations. There is no intermediary server, broker, or bus between the client and the OSD.

When an OSD stores data, it receives data from a Ceph client—whether the client is a Ceph Block Device, a Ceph Object Gateway, a Ceph Filesystem, or another interface—and it stores the data as an object.

Ceph OSDs store all data as objects in a flat namespace. There are no hierarchies of directories. An object has a cluster-wide unique identifier, binary data, and metadata consisting of a set of name/value pairs.

Ceph clients define the semantics for the client's data format. For example, the Ceph block device maps a block device image to a series of objects stored across the cluster.

General Principles for selecting Hardware

As a storage administrator, you must select the appropriate hardware for running a production Red Hat Ceph Storage cluster. When selecting hardware for Red Hat Ceph Storage, review these following general principles. These principles will help save time, avoid common mistakes, save money, and achieve a more effective solution.

Figure 10. Ceph Design Principles






One of the most important steps in a successful Ceph deployment is identifying a price-to-performance profile suitable for the cluster's use case and workload. It is important to choose the right hardware for the use case. For example, choosing IOPS-optimized hardware for a cold storage application increases hardware costs unnecessarily. Whereas, choosing capacity-optimized hardware for its more attractive price point in an IOPS-intensive workload will likely lead to unhappy users complaining about slow performance.

The primary use cases for Ceph are:

- **IOPS optimized:** IOPS optimized deployments are suitable for cloud computing operations, such as running MySQL or MariaDB instances as virtual machines on OpenStack. IOPS optimized deployments require higher performance storage such as 15k RPM SAS drives and separate SSD journals to handle frequent write operations. Some high IOPS scenarios use all flash storage to improve IOPS and total throughput.
- **Throughput optimized:** Throughput-optimized deployments are suitable for serving up significant amounts of data, such as graphic, audio and video content. Throughput-optimized deployments require networking hardware, controllers, and hard disk drives with acceptable total throughput characteristics. In cases where write performance is a requirement, SSD journals will substantially improve write performance.
- **Capacity optimized:** Capacity-optimized deployments are suitable for storing significant amounts of data as inexpensively as possible. Capacity-optimized deployments typically trade performance for a more attractive price point. For example, capacity-optimized deployments often use slower and less expensive SATA drives and co-locate journals rather than using SSDs for journaling.

Figure 11. Ceph Use Cases

IOPS OPTIMIZED NVMe SSD in SLED chassis	THROUGHPUT OPTIMIZED SSD, HDD in standard / dense chassis	COST/ CAPACITY OPTIMIZED HDD in dense / ultra-dense chassis
High IOPS / GB Smaller, random IO Read / write mix	High MB/s throughput Large, sequential IO Read / write mix	Low cost / GB Sequential IO Write mostly
Use Case: MySQL 	Use Case: Rich Media 	Use Case: Active Archives 

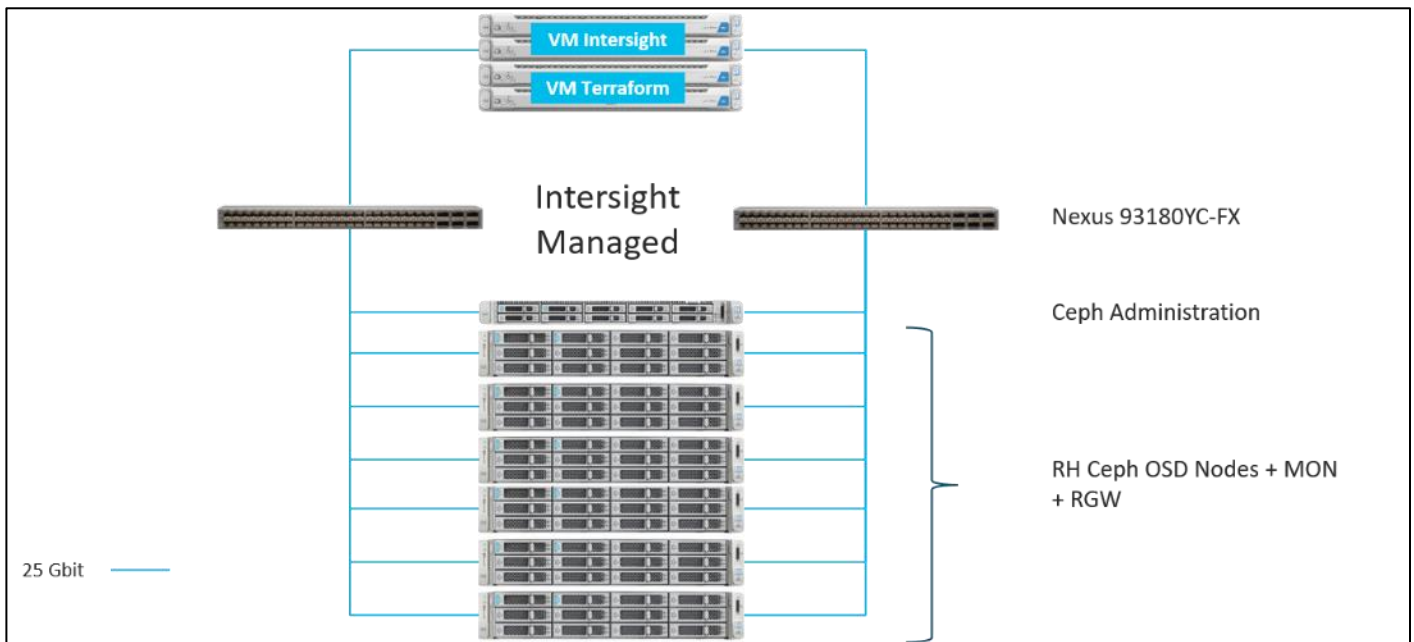
Solution Design

Solution Overview

In this architecture, we have Red Hat Ceph Storage deployed on Cisco UCS with Cisco Intersight and Terraform provider for Cisco Intersight. We automatically have setup six Cisco UCS C240 M5L server and one Cisco UCS C220 M5S server with Terraform provider for Cisco Intersight, simplifying the process of orchestrating a scale-out storage environment. All seven servers were installed with the latest Red Hat Enterprise Linux 8 operating system.

We deployed manually Cisco Intersight virtual Appliance and Terraform as virtual machines. Both virtual machines were deployed on a Cisco UCS HyperFlex Edge cluster, connected to a pair of Cisco Nexus switches. The HyperFlex Edge cluster is not part of the overall deployment but fits well into the overall solution because of the ability to host multiple virtual machines by a simple deployment and management.

Figure 12. Solution Overview



This Cisco Validated Design provides a comprehensive, end-to-end guide for deploying Red Hat Ceph Storage on Cisco UCS C240 M5 and Cisco UCS C220 M5S with Cisco Intersight and Terraform provider for Cisco Intersight.

The 6-node Red Hat Ceph Storage solution has various options to scale capacity. The tested configuration uses Erasure Coding 4+2 and RF=3 replication, configured per Ceph pool. A base capacity summary for the tested solution is listed in [Table 1](#). The usable capacity assumes that 100% of your data is stored either with RF=3 or EC 4+2. The real usable capacity is between both values because some data will be stored with EC 4+2 and some with RF=3.

Table 1. Storage Capacity

HDD Type	Number of Disks	Usable Capacity RF=3	Usable Capacity EC 4+2
4 TB 7200-rpm LFF NL-SAS	72	96 TB	190 TB
6 TB 7200-rpm LFF NL-SAS		144 TB	285 TB
8 TB 7200-rpm LFF NL-SAS		192 TB	380 TB
10 TB 7200-rpm LFF NL-SAS		240 TB	475 TB
12 TB 7200-rpm LFF NL-SAS		288 TB	570 TB
14 TB 7200-rpm LFF NL-SAS		336 TB	665 TB
16 TB 7200-rpm LFF NL-SAS		384 TB	760 TB
18 TB 7200-rpm LFF NL-SAS		432 TB	855 TB

Solution Flow

The solution setup consists of multiple parts. It covers basic setup of the network components, policies and profiles setup, installations of various parts as well as functional tests and high availability testing. The high-level flow of the solution setup is as follows:

1. Install and configure Cisco Nexus 93180YC-FX.
2. Deploy Cisco Intersight virtual Appliance.
3. Deploy Terraform virtual machine.
4. Install and configure Cisco UCS C240 M5 with Cisco Intersight and Terraform provider for Cisco Intersight.
5. Deploy Red Hat Enterprise Linux and Red Hat Ceph Storage.
6. Configure and install Red Hat Ceph Storage.
7. Functional tests of the whole solution.
8. Performance tests for S3 and Block device.
9. High Availability testing of the solution.

Requirements

The following sections detail the physical hardware, software revisions, and firmware versions required to install a single Red Hat Ceph Storage cluster on Cisco UCS. This is specific to the solution built in this CVD.

Physical Components

Table 2. Hardware Components used in this CVD

Component	Model	Quantity	Comments
Switches	Cisco Nexus 93180YC-FX	2	
Cisco UCS	Cisco UCS C240 M5L	6	<p>Each Node:</p> <p>2 x Intel Xeon Silver 4214R (2.4 GHz, 12 Cores)</p> <p>384 GB Memory</p> <p>Cisco 12G Modular Raid Controller with 2GB cache</p> <p>1 x 3.2 TB + 1 x 1.6 TB NVMe HGST SN260 NVMe</p> <p>Extreme Perf. High Endurance for Bluestore WAL DB</p> <p>2 x 960 GB 6 Gbps SATA SSD for System</p> <p>12 x 10 TB 12 Gbps NL-SAS HDD for Data</p> <p>1 x VIC 1455</p>
Cisco UCS	Cisco UCS C220 M5S	1	<p>2 x Intel Xeon Platinum 5218R (2.1 GHz, 20 Cores)</p> <p>384 GB Memory</p> <p>Cisco 12G Modular Raid Controller with 2GB cache</p> <p>2 x 1.9 TB 6 Gbps SATA SSD for System</p> <p>1 x VIC 1455</p>
Cisco Intersight Virtual Appliance	Virtual Machine	1	<p>16 vCPU</p> <p>32 GB Memory</p> <p>500 GB Disk</p> <p>1 x Network</p>
Terraform	Virtual Machine	1	<p>2 vCPU</p> <p>16 GB Memory</p> <p>100 GB Disk</p> <p>1 x Network</p>

Software Components

The required software distribution versions are listed in [Table 3](#).

Table 3. Software Versions

Layer	Component	Version or Release
Cisco UCS C240 M5L	Firmware Version	4.1(3b)
Cisco UCS C220 M5SX	Firmware Version	4.1(3b)
Network Nexus 93180YC-FX	BIOS	07.67
	NXOS	9.3(4)
Cisco Intersight Virtual Appliance	Version	1.0.9-214
Software	Terraform	0.13.5
Software	Terraform Provider for Intersight	0.1.3
Software	Red Hat Enterprise Linux	8.3
Software	Red Hat Ceph Storage	4.2 / 14.2.11-95.el8cp

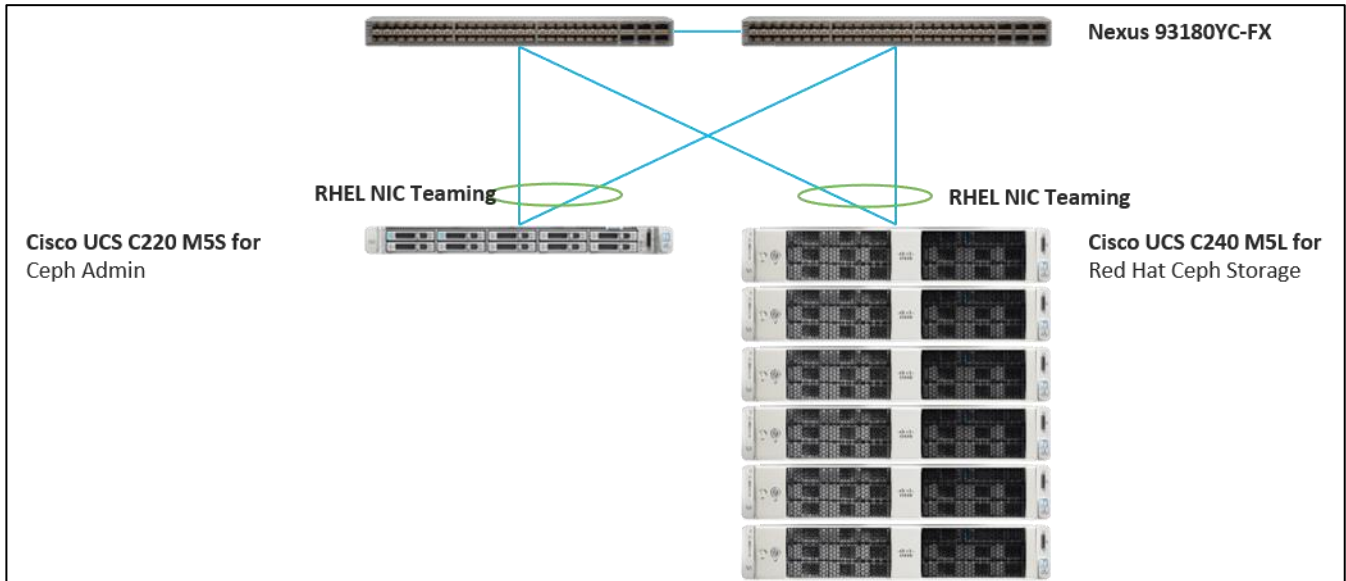
Physical Topology

Topology Overview

The solution contains one topology configuration. There are six Cisco UCS C240 M5 and one Cisco UCS C220 M5 connected to a pair of Cisco Nexus 93180YC-FX switches. Each Cisco UCS C240 M5 and C220 M5 server relates to one 25-Gbps cable to each Cisco Nexus 93180YC-FX. All six Cisco UCS C240 M5 server and the one Cisco CUS C220 M5S server use RHEL NIC teaming with “load-balance” runner to achieve high availability and high performance.

The following diagram illustrates the details of the configuration.

Figure 13. Datacenter Topology



Network Design

VLANS and Subnets

For the base configuration multiple VLANs need to be carried to the Cisco UCS domain from the upstream LAN, and these VLANs are also defined in the Cisco UCS configuration. Table 4 lists the VLANs created by the Cisco Intersight used in this CVD and their functions:

Table 4. VLANs and Subnets

VLAN Name	VLAN ID	Subnet	Purpose
Management	300	172.16.32.0/24	Cisco UCS CIMC management interfaces
		GW 172.16.32.1	Cisco Intersight
			Terraform
Client	301	172.16.33.0/24	Client network for Red Hat Ceph Storage
		GW 172.16.33.1	COSBench clients for performance testing
Storage	302	172.16.34.0/24	Storage network for Red Hat Ceph Storage
		GW 172.16.34.1	

Jumbo Frames

All traffic traversing the Client and Storage VLAN and subnet is configured by default to use jumbo frames, or to be precise, all communication is configured to send IP packets with a Maximum Transmission Unit (MTU) size of 9000 bytes. Using a larger MTU value means that each IP packet sent carries a larger payload, therefore transmitting more data per packet, and consequently sending and receiving data faster.

Naming Scheme and DNS

DNS servers are highly recommended to be configured for querying Fully Qualified Domain Names (FQDN). DNS records need to be created prior to beginning the installation. At a minimum, it is highly recommended to create A records and reverse PTR records.

Use [Table 5](#) to gather the required DNS information for the installation and list the information required for this CVD:

Table 5. DNS Server Information

Item	Name
DNS Server	192.168.10.51
DNS Domain	sjc02dmz.net
vCenter Server Name	sjc02dmz-vcsa
Cisco Nexus 93180YC-FX #1	sjc02dmz-i14-n93180ycfx-a
Cisco Nexus 93180YC-FX #2	sjc02dmz-i14-n93180ycfx-b
Cisco Intersight virtual Appliance	sjc02dmz-intersight
Cisco UCS C240 M5 #1	cephosd1
Cisco UCS C240 M5 #2	cephosd2
Cisco UCS C240 M5 #3	cephosd3
Cisco UCS C240 M5 #4	cephosd4
Cisco UCS C240 M5 #5	cephosd5
Cisco UCS C240 M5 #6	cephosd6
Cisco UCS C220 M5	cephadm
Terraform	sjc02dmz-i14-terraform

Cabling

The physical layout of the solution was previously described in section [Topology Overview](#). The Cisco Nexus switches, and the Cisco UCS server need to be cabled properly before beginning the installation activities. [Table 6](#) provides the cabling map for installation of a Red Hat Ceph Storage solution on Cisco UCS.

Table 6. Cabling Map Cisco Nexus 93180YC-FX

Device	Port	Connected To	Port	Note
sjc02dmz-i14-n93180ycfx-a	1	cephosd1	Port 0	
sjc02dmz-i14-n93180ycfx-a	2	cephosd2	Port 0	

Device	Port	Connected To	Port	Note
sjc02dmz-i14-n93180ycfx-a	3	cephosd3	Port 0	
sjc02dmz-i14-n93180ycfx-a	4	cephosd4	Port 0	
sjc02dmz-i14-n93180ycfx-a	5	cephosd5	Port 0	
sjc02dmz-i14-n93180ycfx-a	6	cephosd6	Port 0	
sjc02dmz-i14-n93180ycfx-a	7	cephadm	Port 0	
sjc02dmz-i14-n93180ycfx-a	8	sjc02dmz-i14-hxe220c1	Port 0	HX Edge
sjc02dmz-i14-n93180ycfx-a	9	sjc02dmz-i14-hxe220c2	Port 0	HX Edge
sjc02dmz-i14-n93180ycfx-a	10	sjc02dmz-i14-hxe220c3	Port 0	HX Edge
sjc02dmz-i14-n93180ycfx-a	11	sjc02dmz-i14-hxe220c4	Port 0	HX Edge
sjc02dmz-i14-n93180ycfx-a	49	sjc02dmz-i14-n93180ycfx-b	Eth1/49	vPC Peer Link
sjc02dmz-i14-n93180ycfx-a	50	sjc02dmz-i14-n93180ycfx-b	Eth1/50	vPC Peer Link
sjc02dmz-i14-n93180ycfx-b	1	cephosd1	Port 2	
sjc02dmz-i14-n93180ycfx-b	2	cephosd2	Port 2	
sjc02dmz-i14-n93180ycfx-b	3	cephosd3	Port 2	
sjc02dmz-i14-n93180ycfx-b	4	cephosd4	Port 2	
sjc02dmz-i14-n93180ycfx-b	5	cephosd5	Port 2	
sjc02dmz-i14-n93180ycfx-b	6	cephosd6	Port 2	
sjc02dmz-i14-n93180ycfx-b	7	cephadm	Port 2	
sjc02dmz-i14-n93180ycfx-b	8	sjc02dmz-i14-hxe220c1	Port 2	HX Edge
sjc02dmz-i14-n93180ycfx-b	9	sjc02dmz-i14-hxe220c2	Port 2	HX Edge
sjc02dmz-i14-n93180ycfx-b	10	sjc02dmz-i14-hxe220c3	Port 2	HX Edge
sjc02dmz-i14-n93180ycfx-b	11	sjc02dmz-i14-hxe220c4	Port 2	HX Edge

Device	Port	Connected To	Port	Note
sjc02dmz-i14-n93180ycfx-b	49	sjc02dmz-i14-n93180ycfx-a	Eth1/49	vPC Peer Link
sjc02dmz-i14-n93180ycfx-b	50	sjc02dmz-i14-n93180ycfx-a	Eth1/50	vPC Peer Link

Rack Layout

The core solution with the Cisco UCS C220 M5S and C240 M5L takes 13 RU space in a standard rack. The additional Cisco HyperFlex Edge solution for the virtual machines takes another 4 RU space on top of the Ceph solution. The below figure shows the rack layout.

Figure 14. Rack Layout



Deployment Hardware and Software

Fabric Configuration

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

Configure Cisco Nexus 93180YC-FX Switch A and B

Both Cisco UCS Fabric Interconnect A and B are connected to two Cisco Nexus 93180YC-FX switches for connectivity to applications and clients. The following sections describe the setup of both Cisco Nexus 93180YC-FX switches.

Initial Setup of Cisco Nexus 93180YC-FX 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 `y`.
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. Type in L2 for interface layer.

18. Press <Return> and again <Return>.

19. 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]:

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 any time 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]:
Configure read-write SNMP community string (yes/no) [n]:
Enter the switch name : sjc02dmz-i14-n93180ycfx-a
Continue with Out-of-band (mgmt0) management configuration? (yes/no) [y]:
  Mgmt0 IPv4 address : 192.168.10.20
  Mgmt0 IPv4 netmask : 255.255.255.0
Configure the default gateway? (yes/no) [y]:
  IPv4 address of the default gateway : 192.168.10.3
Configure advanced IP options? (yes/no) [n]:
Enable the telnet service? (yes/no) [n]:
Enable the ssh service? (yes/no) [y]:
```

```
Type of ssh key you would like to generate (dsa/rsa) [rsa]:
Number of rsa key bits <1024-2048> [1024]:
Configure the ntp server? (yes/no) [n]: y
NTP server IPv4 address : 173.38.201.115
Configure default interface layer (L3/L2) [L3]: L2
Configure default switchport interface state (shut/noshut) [shut]:
Configure CoPP system profile (strict/moderate/lenient/dense) [strict]:
The following configuration will be applied:
password strength-check
switchname sjc02dmz-f9-n93180ycfx-a
vrf context management
ip route 0.0.0.0/0 192.168.10.3
exit
no feature telnet
ssh key rsa 1024 force
feature ssh
ntp server 173.38.201.115
no system default switchport
system default switchport shutdown
copp profile strict
interface mgmt0
ip address 192.168.10.20 255.255.255.0
no shutdown

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

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

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

User Access Verification
sjc02dmz-i14-n93180ycfx-a login:
```



Repeat steps 1-19 for the Cisco Nexus 93180YC-FX Switch B with the exception of configuring a different IPv4 management address in step 7.

Enable Features on Cisco Nexus 93180YC-FX Switch A and B

To enable the features UDLD, VLAN, LACP, HSRP, 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

```
sjc02dmz-i14-n93180ycfx-a # configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
sjc02dmz-i14-n93180ycfx-a (config)# feature udld
sjc02dmz-i14-n93180ycfx-a (config)# feature interface-vlan
sjc02dmz-i14-n93180ycfx-a(config)# feature lacp
sjc02dmz-i14-n93180ycfx-a(config)# feature vpc
sjc02dmz-i14-n93180ycfx-a(config)# feature hsrp
sjc02dmz-i14-n93180ycfx-a(config)# system jumbomtu 9216
sjc02dmz-i14-n93180ycfx-a(config)# spanning-tree port type edge bpduguard default
sjc02dmz-i14-n93180ycfx-a(config)# spanning-tree port type edge bpdufilter default
sjc02dmz-i14-n93180ycfx-a(config)# port-channel load-balance src-dst ip-l4port
sjc02dmz-i14-n93180ycfx-a(config)# exit
sjc02dmz-i14-n93180ycfx-a#
```

Switch B

```
sjc02dmz-i14-n93180ycfx-b# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
sjc02dmz-i14-n93180ycfx-b(config)# feature udld
sjc02dmz-i14-n93180ycfx-b(config)# feature interface-vlan
sjc02dmz-i14-n93180ycfx-b(config)# feature lacp
sjc02dmz-i14-n93180ycfx-b(config)# feature vpc
sjc02dmz-i14-n93180ycfx-b(config)# feature hsrp
sjc02dmz-i14-n93180ycfx-b(config)# system jumbomtu 9216
sjc02dmz-i14-n93180ycfx-b(config)# spanning-tree port type edge bpduguard default
sjc02dmz-i14-n93180ycfx-b(config)# spanning-tree port type edge bpdufilter default
sjc02dmz-i14-n93180ycfx-b(config)# port-channel load-balance src-dst ip-l4port
sjc02dmz-i14-n93180ycfx-b(config)# exit
sjc02dmz-i14-n93180ycfx-b#
```

Configure VLANs on Nexus 93180YC-FX Switch A and B

To configure VLAN Client and Storage, follow these steps on Switch A and Switch B:

Switch A

```
sjc02dmz-i14-n93180ycfx-a# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
sjc02dmz-i14-n93180ycfx-a(config)# vlan 300
sjc02dmz-i14-n93180ycfx-a(config-vlan)# name Management
sjc02dmz-i14-n93180ycfx-a(config-vlan)# exit
sjc02dmz-i14-n93180ycfx-a(config)# vlan 301
```

```
sjc02dmz-i14-n93180ycfx-a(config-vlan)# name Client
sjc02dmz-i14-n93180ycfx-a(config-vlan)# exit
sjc02dmz-i14-n93180ycfx-a(config)# vlan 302
sjc02dmz-i14-n93180ycfx-a(config-vlan)# name Storage
sjc02dmz-i14-n93180ycfx-a(config-vlan)# exit
sjc02dmz-i14-n93180ycfx-a(config)# copy run start
```

Switch B

```
sjc02dmz-i14-n93180ycfx-b# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
sjc02dmz-i14-n93180ycfx-b(config)# vlan 300
sjc02dmz-i14-n93180ycfx-b(config-vlan)# name Management
sjc02dmz-i14-n93180ycfx-b(config-vlan)# exit
sjc02dmz-i14-n93180ycfx-b(config)# vlan 301
sjc02dmz-i14-n93180ycfx-b(config-vlan)# name Client
sjc02dmz-i14-n93180ycfx-b(config-vlan)# exit
sjc02dmz-i14-n93180ycfx-b(config)# vlan 302
sjc02dmz-i14-n93180ycfx-b(config-vlan)# name Storage
sjc02dmz-i14-n93180ycfx-b(config-vlan)# exit
sjc02dmz-i14-n93180ycfx-b(config)# copy run start
```

Configure vPC Domain on Nexus 93180YC-FX Switch A and B

To configure the vPC Domain, follow these steps on Switch A and Switch B:

Switch A

```
sjc02dmz-i14-n93180ycfx-a# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
sjc02dmz-i14-n93180ycfx-a(config)# vpc domain 2
sjc02dmz-i14-n93180ycfx-a(config-vpc-domain)# role priority 10
sjc02dmz-i14-n93180ycfx-a(config-vpc-domain)# peer-keepalive destination 192.168.10.21
source 192.168.10.20
sjc02dmz-i14-n93180ycfx-a(config-vpc-domain)# peer-switch
sjc02dmz-i14-n93180ycfx-a(config-vpc-domain)# peer-gateway
sjc02dmz-i14-n93180ycfx-a(config-vpc-domain)# ip arp synchronize
sjc02dmz-i14-n93180ycfx-a(config-vpc-domain)# auto-recovery
sjc02dmz-i14-n93180ycfx-a(config-vpc-domain)# copy run start
sjc02dmz-i14-n93180ycfx-a(config-vpc-domain)# exit
```

Switch B

```
sjc02dmz-i14-n93180ycfx-b# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
```

```
sjc02dmz-i14-n93180ycfx-b(config)# vpc domain 1
sjc02dmz-i14-n93180ycfx-b(config-vpc-domain)# role priority 20
sjc02dmz-i14-n93180ycfx-b(config-vpc-domain)# peer-keepalive destination 192.168.10.20
source 192.168.10.21
sjc02dmz-i14-n93180ycfx-b(config-vpc-domain)# peer-switch
sjc02dmz-i14-n93180ycfx-b(config-vpc-domain)# peer-gateway
sjc02dmz-i14-n93180ycfx-b(config-vpc-domain)# ip arp synchronize
sjc02dmz-i14-n93180ycfx-b(config-vpc-domain)# auto-recovery
sjc02dmz-i14-n93180ycfx-b(config-vpc-domain)# copy run start
sjc02dmz-i14-n93180ycfx-b(config-vpc-domain)# exit
```

Configure Network Interfaces for vPC Peer Links on Nexus 93180YC-FX Switch A and B

To configure the network interfaces for vPC Peer Links, follow these steps on Switch A and Switch B:

Switch A

```
sjc02dmz-i14-n93180ycfx-a# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
sjc02dmz-i14-n93180ycfx-a(config)# interface Eth 1/49
sjc02dmz-i14-n93180ycfx-a(config-if)# description VPC Peer Nexus B Port 1/49
sjc02dmz-i14-n93180ycfx-a(config-if)# interface Eth 1/50
sjc02dmz-i14-n93180ycfx-a(config-if)# description VPC Peer Nexus B Port 1/50
sjc02dmz-i14-n93180ycfx-a(config-if)# interface Eth1/49,Eth1/50
sjc02dmz-i14-n93180ycfx-a(config-if)# channel-group 2 mode active
sjc02dmz-i14-n93180ycfx-a(config-if)# no shutdown
sjc02dmz-i14-n93180ycfx-a(config-if)# udld enable
sjc02dmz-i14-n93180ycfx-a(config-if)# interface port-channel 2
sjc02dmz-i14-n93180ycfx-a(config-if)# description vPC peer-link
sjc02dmz-i14-n93180ycfx-a(config-if)# switchport
sjc02dmz-i14-n93180ycfx-a(config-if)# switchport mode trunk
sjc02dmz-i14-n93180ycfx-a(config-if)# switchport trunk allowed vlan 300-302
sjc02dmz-i14-n93180ycfx-a(config-if)# spanning-tree port type network
sjc02dmz-i14-n93180ycfx-a(config-if)# vpc peer-link
sjc02dmz-i14-n93180ycfx-a(config-if)# no shutdown
sjc02dmz-i14-n93180ycfx-a(config-if)# copy run start
```

Switch B

```
sjc02dmz-i14-n93180ycfx-b# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
sjc02dmz-i14-n93180ycfx-b(config)# interface Eth 1/49
sjc02dmz-i14-n93180ycfx-b(config-if)# description VPC Peer Nexus A Port 1/49
```

```

sjc02dmz-i14-n93180ycfx-b(config-if)# interface Eth 1/50
sjc02dmz-i14-n93180ycfx-b(config-if)# description VPC Peer Nexus A Port 1/50
sjc02dmz-i14-n93180ycfx-b(config-if)# interface Eth1/49,Eth1/50
sjc02dmz-i14-n93180ycfx-b(config-if)# channel-group 2 mode active
sjc02dmz-i14-n93180ycfx-b(config-if)# no shutdown
sjc02dmz-i14-n93180ycfx-b(config-if)# udld enable
sjc02dmz-i14-n93180ycfx-b(config-if)# interface port-channel 2
sjc02dmz-i14-n93180ycfx-b(config-if)# description vPC peer-link
sjc02dmz-i14-n93180ycfx-b(config-if)# switchport
sjc02dmz-i14-n93180ycfx-b(config-if)# switchport mode trunk
sjc02dmz-i14-n93180ycfx-b(config-if)# switchport trunk allowed vlan 300-302
sjc02dmz-i14-n93180ycfx-b(config-if)# spanning-tree port type network
sjc02dmz-i14-n93180ycfx-b(config-if)# vpc peer-link
sjc02dmz-i14-n93180ycfx-b(config-if)# no shutdown
sjc02dmz-i14-n93180ycfx-b(config-if)# copy run start

```

Verification Check of Cisco Nexus 93180YC-FX Configuration for Switch A and B

Switch A

```

sjc02dmz-i14-n93180ycfx-a# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
sjc02dmz-i14-n93180ycfx-a# 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     : 2
Peer Gateway                  : Enabled
Dual-active excluded VLANs    : -
Graceful Consistency Check    : Enabled
Auto-recovery status         : Enabled, timer is off.(timeout = 240s)
Delay-restore status         : Timer is off.(timeout = 30s)
Delay-restore SVI status      : Timer is off.(timeout = 10s)
Operational Layer3 Peer-router : Disabled

```

```
vPC Peer-link status
```

```
-----  
id      Port      Status Active vlans  
--      -  
1       Po2       up      300-302  
-----
```

Please check "show vpc consistency-parameters vpc <vpc-num>" for the consistency reason of down vpc and for type-2 consistency reasons for any vpc.

```
sjc02dmz-i14-n93180ycfx-a# 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  
        b - BFD Session Wait  
        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  
-----  
2      Po2(SU)     Eth       LACP      Eth1/49(P)  Eth1/50(P)
```

Switch B

```
sjc02dmz-i14-n93180ycfx-b# config terminal
```

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

```
sjc02dmz-i14-n93180ycfx-b# 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      : 2
Peer Gateway                   : Enabled
Dual-active excluded VLANs    : -
Graceful Consistency Check    : Enabled
Auto-recovery status          : Enabled, timer is off.(timeout = 240s)
Delay-restore status          : Timer is off.(timeout = 30s)
Delay-restore SVI status      : Timer is off.(timeout = 10s)
Operational Layer3 Peer-router : Disabled

```

vPC Peer-link status

```

-----
id      Port      Status Active vlans
--      ----      -
1       Po2       up      300-302

```

Please check "show vpc consistency-parameters vpc <vpc-num>" for the consistency reason of down vpc and for type-2 consistency reasons for any vpc.

```

sjc02dmz-i14-n93180ycfx-b# 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
         b - BFD Session Wait
         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
-----
2       Po2(SU)    Eth       LACP      Eth1/49(P)  Eth1/50(P)

```

Implement Intelligent Buffer Management for Cisco Nexus 93180YC-FX

Cisco Nexus 9000 Series Switches with Cisco cloud-scale ASICs are built with a moderate amount of on-chip buffer space to achieve 100 percent throughput on high-speed 10/25/40/50/100-Gbps links and with intelligent buffer management functions to efficiently serve mixed mice flows and elephant flows. The critical concept in Cisco's innovative intelligent buffer management is the capability to distinguish mice and elephant flows and apply different queue management schemes to them based on their network forwarding requirements in the event

of link congestion. This capability allows both elephant and mice flows to achieve their best performance, which improves overall application performance.

Cisco intelligent buffer management includes approximate fair dropping (AFD) with elephant trap (ETRAP), and dynamic packet prioritization (DPP) functions. It uses an algorithm-based architectural approach to address the buffer requirements in modern data centers. It offers a cost-effective and sustainable solution to support the ever-increasing network speed and data traffic load.

The intelligent buffer management capabilities are built in to Cisco cloud-scale ASICs for hardware-accelerated performance. The main functions include approximate fair dropping (AFD) with elephant trap (ETRAP) and dynamic packet prioritization (DPP). AFD focuses on preserving buffer space to absorb mice flows, particularly microbursts, which are aggregated mice flows, by limiting the buffer use of aggressive elephant flows. It also aims to enforce bandwidth allocation fairness among elephant flows. DPP provides the capability of separating mice flows and elephant flows into two different queues so that buffer space can be allocated to them independently, and different queue scheduling can be applied to them. For example, mice flows can be mapped to a low-latency queue (LLQ), and elephant flows can be sent to a weighted fair queue. AFD and DPP can be deployed separately or jointly.

Configure Queuing Policy with AFD

AFD itself is configured in queuing policies and applied to the egress class-based queues. The only parameter in a queuing policy map that needs to be configured for AFD is the desired queue depth for a given class-based queue. This parameter controls when AFD starts to apply algorithm-based drop or ECN marking to elephant flows within this class. AFD can be defined in any class-based queues.

The desired queue depth should be set differently for different link speeds of the egress port because it needs to be sufficient to achieve 100 percent throughput. It also should be a balance of the buffer headroom that needs to be reserved for mice flows, the number of packet retransmissions, and queue latency. [Table 7](#) lists the recommended values for some typical link speeds, but users can choose different values in their particular data center environments.

Table 7. Recommended Desired Queue Depth for Typical Link Speeds

Port Speed	Value of Desired Queue Depth
10 Gbps	150 KB
25 Gbps	375 KB
40 Gbps	600 KB
100 Gbps	1500 KB

To configure the queue depth for switch A, run the following:

```
sjc02dmz-i14-n93180ycfx-a# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
sjc02dmz-i14-n93180ycfx-a(config)# policy-map type queuing afd_8q-out
sjc02dmz-i14-n93180ycfx-a(config-pmap-que)# class type queuing c-out-8q-q7
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# priority level 1
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# class type queuing c-out-8q-q6
```

```
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# bandwidth remaining percent 0
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# class type queuing c-out-8q-q5
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# bandwidth remaining percent 0
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# class type queuing c-out-8q-q4
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# bandwidth remaining percent 0
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# class type queuing c-out-8q-q3
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# bandwidth remaining percent 0
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# class type queuing c-out-8q-q2
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# bandwidth remaining percent 0
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# class type queuing c-out-8q-q1
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# bandwidth remaining percent 0
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# class type queuing c-out-8q-q-default
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# afd queue-desired 375 kbytes
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# bandwidth remaining percent 100
sjc02dmz-i14-n93180ycfx-a(config-pmap-c-que)# exit
sjc02dmz-i14-n93180ycfx-a(config-pmap-que)# exit
sjc02dmz-i14-n93180ycfx-a(config)# system qos
sjc02dmz-i14-n93180ycfx-a(config-sys-qos)# service-policy type queuing output afd_8q-
out
sjc02dmz-i14-n93180ycfx-a(config-sys-qos)# exit
sjc02dmz-i14-n93180ycfx-a(config)# copy run start
[#####] 100%
Copy complete, now saving to disk (please wait)...
Copy complete.
sjc02dmz-i14-n93180ycfx-a(config)# sh policy-map type queuing afd_8q-out
```

Type queuing policy-maps

=====

```
policy-map type queuing afd_8q-out
  class type queuing c-out-8q-q7
    priority level 1
  class type queuing c-out-8q-q6
    bandwidth remaining percent 0
  class type queuing c-out-8q-q5
    bandwidth remaining percent 0
  class type queuing c-out-8q-q4
    bandwidth remaining percent 0
  class type queuing c-out-8q-q3
```

```

    bandwidth remaining percent 0
class type queuing c-out-8q-q2
    bandwidth remaining percent 0
class type queuing c-out-8q-q1
    bandwidth remaining percent 0
class type queuing c-out-8q-q-default
    afd queue-desired 375 kbytes
    bandwidth remaining percent 100

```

The line in yellow shows the configured queue depth for 25 Gbps connectivity. Please repeat this step for switch B.

Configure Network-QoS Policy with DPP

To configure the network-QoS policy for switch A, follow these steps:

```

sjc02dmz-i14-n93180ycfx-a# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
sjc02dmz-i14-n93180ycfx-a(config)# policy-map type network-qos dpp
sjc02dmz-i14-n93180ycfx-a(config-pmap-nqos)# class type network-qos c-8q-nq-default
sjc02dmz-i14-n93180ycfx-a(config-pmap-nqos-c)# dpp set-qos-group 7
sjc02dmz-i14-n93180ycfx-a(config-pmap-nqos-c)# mtu 9216
sjc02dmz-i14-n93180ycfx-a(config-pmap-nqos-c)# system qos
sjc02dmz-i14-n93180ycfx-a(config-sys-qos)# service-policy type network-qos dpp
sjc02dmz-i14-n93180ycfx-a(config-sys-qos)# exit
sjc02dmz-i14-n93180ycfx-a(config)# copy run start
[#####] 100%
Copy complete, now saving to disk (please wait)...
Copy complete.

```

Repeat this step for switch B.

Configure Switch Ports for Ceph Nodes

To configure the switch ports for all nodes in our solution, run the following:

```

sjc02dmz-i14-n93180ycfx-a(config)# int eth 1/1-7
sjc02dmz-i14-n93180ycfx-a(config-if-range)# switchport
sjc02dmz-i14-n93180ycfx-a(config-if-range)# switchport mode trunk
sjc02dmz-i14-n93180ycfx-a(config-if-range)# switchport trunk allowed vlan 300-302
sjc02dmz-i14-n93180ycfx-a(config-if-range)# 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

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

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

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

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

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

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

```
sjc02dmz-i14-n93180ycfx-a(config-if-range)# mtu 9216
sjc02dmz-i14-n93180ycfx-a(config-if-range)# fec fc-fec
sjc02dmz-i14-n93180ycfx-a(config-if-range)# copy run start
[#####] 100%
Copy complete, now saving to disk (please wait)...
Copy complete.
sjc02dmz-i14-n93180ycfx-a(config-if-range)# exit
sjc02dmz-i14-n93180ycfx-a(config)# exit
```

Repeat this step for switch B. The formal setup for the Cisco Nexus 93180YC-FX switches is now finished.

Installation of Cisco Intersight

Cisco Intersight provides infrastructure management for Cisco Unified Compute System (Cisco UCS) and Cisco HyperFlex platforms. This platform offers an intelligent level of management that enables IT organizations to analyze, simplify, and automate their environments in more advanced ways than previous generations of tools.

Cisco Intersight Virtual Appliance delivers the management features of Intersight for Cisco UCS and HyperFlex in an easy to deploy VMware OVA that allows you to control what system details leave your premises. The Virtual Appliance form factor enables additional data locality, security, or compliance needs that are not completely met by intersight.com. Cisco Intersight Virtual Appliance requires a connection back to Cisco and Intersight services for updates and access required services for full functionality of intersight.com. Cisco Intersight Virtual Appliance is not intended for an environment where you operate data centers with no external connectivity.

You can deploy Cisco Intersight Virtual Appliance as a virtual machine in your existing environment quickly in a few easy steps, which will be shown in the next couple of steps. This guide provides an overview of how to install and set up Cisco Intersight Virtual Appliance in your environment.

Licensing Requirements

Cisco Intersight Virtual Appliance uses a subscription-based license that is required to use the features of the appliance. Intersight Essentials is a subscription license delivered via Cisco Smart Licensing. Enabled platforms are those Cisco UCS and Cisco HyperFlex systems with a Cisco Intersight device connector, including eligible Cisco UCS Manager, Cisco IMC, Cisco HyperFlex software.

You must register the license as part of the initial setup of Cisco Intersight Virtual Appliance. After you complete the installation of the appliance OVA, launch the UI, and set up a password, connect the appliance to Intersight, and register the license.

You can obtain an Intersight evaluation license for Cisco Intersight Virtual Appliance from your Cisco sales representative, channel partner, or reseller. If you already have a Cisco Smart Account, the evaluation license will be added to your Cisco Smart Account. You can then generate a token for the virtual account in the Smart account and proceed with registering Cisco Intersight Virtual Appliance. In our validated design we obtained an evaluation license for 90 days.

VM Configuration Requirements

The Cisco Intersight Virtual Appliance OVA can be deployed on VMware ESXi 6.0 and higher. The following sections describe the various system requirements to install and deploy Cisco Intersight Virtual Appliance:

You can deploy Intersight Virtual Appliance in the Small or Medium options. For more information on the resource requirements and supported maximum configuration limits for Intersight Virtual Appliance Sizing Options, see Intersight Virtual Sizing Options.

Table 8. Resource Requirements for the Intersight Virtual Appliance

Resource Requirements	System Requirements	
	Small	Medium
vCPU	16	24

Resource Requirements	System Requirements	
	Small	Medium
RAM (GiB)	32	64
Storage (Disk)(GiB)	500 Cisco recommends that you use thick provisioning	500 Cisco recommends that you use thick provisioning
Number of servers	2000	5000
Supported Hypervisors	VMware ESXi 6.0 and higher VMware vSphere Web Client 6.5 and higher	

IP Address and Hostname Requirements

Setting up Intersight Appliance requires an IP address and 2 hostnames for that IP address. The hostnames must be in the following formats:

- **myhost.mydomain.com**—A hostname in this format is used to access the GUI. This must be defined as an A record and PTR record in DNS. The PTR record is required for reverse lookup of the IP address. For details about Regular Expression for a valid hostname, see RFC 1123. If an IP address resolves to multiple hostnames, the first resolved hostname is used.
- **dc-myhost.mydomain.com**—The dc- must be prepended to your hostname. This hostname must be defined as the CNAME of myhost.mydomain.com. Hostnames in this format are used internally by the appliance to manage device connections.



Ensure that the appropriate entries of type **A**, **CNAME**, and **PTR** records exist in the DNS, as described above.

Port Requirements

The following table lists the ports required to be open for Intersight Appliance communication.

Table 9. Port requirements for Cisco Intersight

Port	Protocol	Description
443	TCP/UDP	This port is required for communication between: <ul style="list-style-type: none"> • Intersight Virtual Appliance and the users' web browser. • Intersight Virtual Appliance to and from the endpoint devices.
80	TCP	This port is optional for normal operation but is required for initial monitoring of the appliance setup and when using the one-time device connector upgrade. For more information, see Device Connector Upgrade. This port is used for communication between: <ul style="list-style-type: none"> • Intersight Virtual Appliance and the user's web browser for initial

Port	Protocol	Description
		<p>monitoring of the appliance setup and when using the one-time device connector up-grade.</p> <ul style="list-style-type: none"> Appliance and the endpoint device for upgrade of the device connector. Port 80 is required when the device connector version is lower than the minimum supported version. For more information, see Device Connector Requirements. <p>Port 80 is not used if the device connector is at the minimum supported version.</p>

Connectivity Requirements

Ensure that Cisco Intersight Virtual Appliance has access to the following sites directly or through a proxy. For more information about setting up a proxy, see [Cloud Connection](#). All the following URLs are accessed through HTTPS.

- Access to Cisco services (*.cisco.com)
- tools.cisco.com:443—for access to Cisco Smart Licensing Manager
- api.cisco.com:443— for access to Cisco Software download site
- Access to Intersight Cloud services.

Intersight Virtual Appliance connects to Intersight by resolving one of the following URLs:

- svc.intersight.com—(Preferred)
- svc.ucs-connect.com—(Will be deprecated in the future)
- IP address for any given URL could change. In case you need to specify firewall configurations for URLs with fixed IPs, use one of the following:
 - svc-static1.intersight.com—(Preferred)
 - svc-static1.ucs-connect.com—(Will be deprecated in the future)

Both these URLs resolve to the following IP addresses:

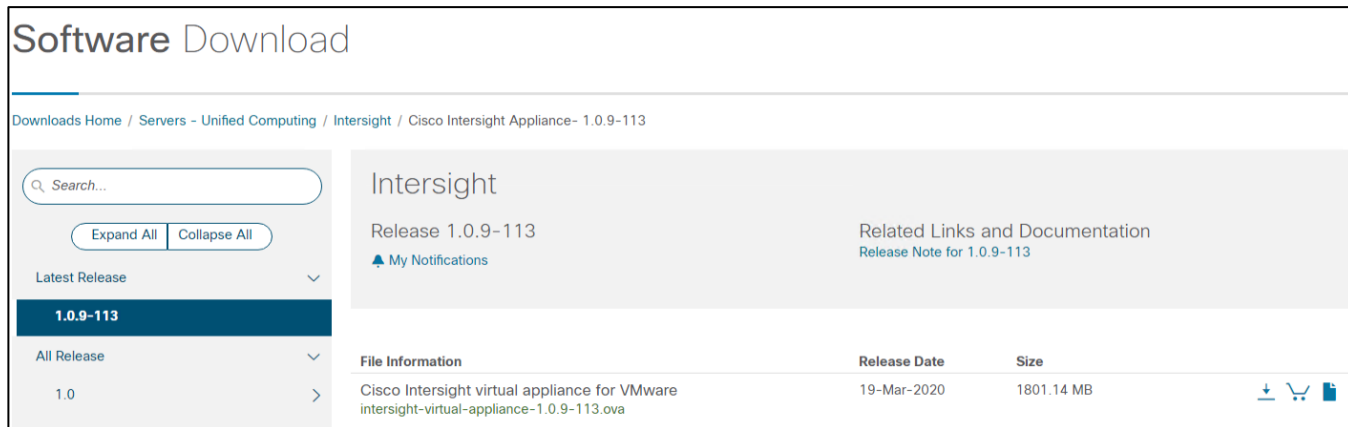
- 3.208.204.228
- 54.165.240.89
- 3.92.151.78

Install Cisco Intersight Virtual Appliance Using VMware vSphere Web Client

Cisco Intersight Virtual Appliance is distributed as a deployable virtual machine contained in an Open Virtual Appliance (OVA) file format. You can install the appliance on an ESXi server. Cisco Intersight Virtual Appliance supports VMware High Availability (VMHA) to ensure non-disruptive operation of the virtual appliance. Use the following procedure to install and deploy the appliance using a VMware vSphere Web Client.

Ensure that you have downloaded the Cisco Intersight Virtual Appliance package from the URL provided by your Cisco representative or a location accessible from your setup, such as a local hard drive, a network share, or a CD/DVD drive.

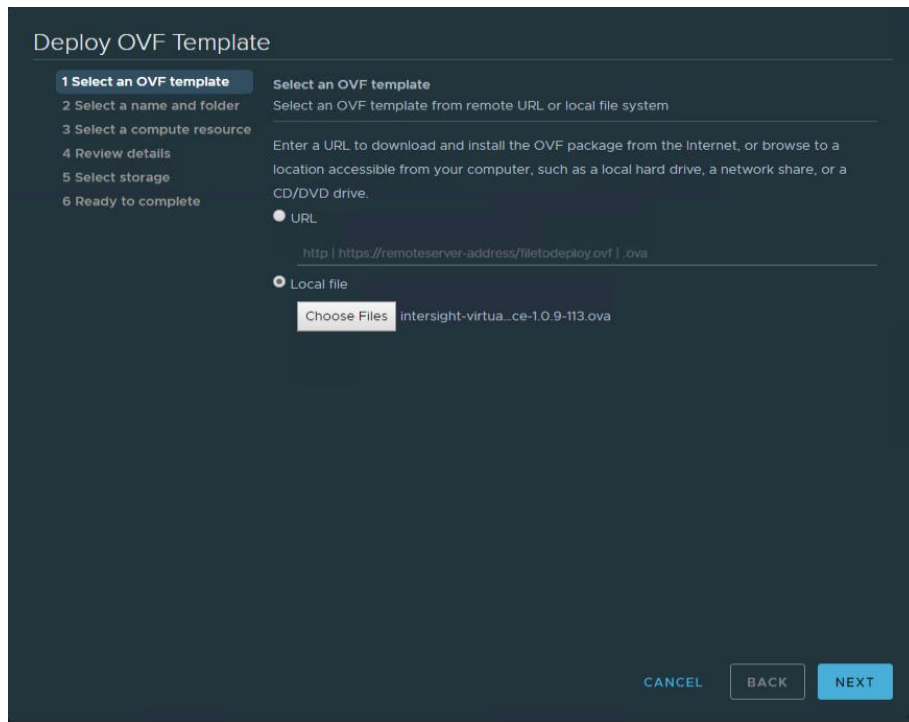
Figure 15. Download of Cisco Intersight from cisco.com



To install Cisco Intersight Virtual Appliance, follow these steps:

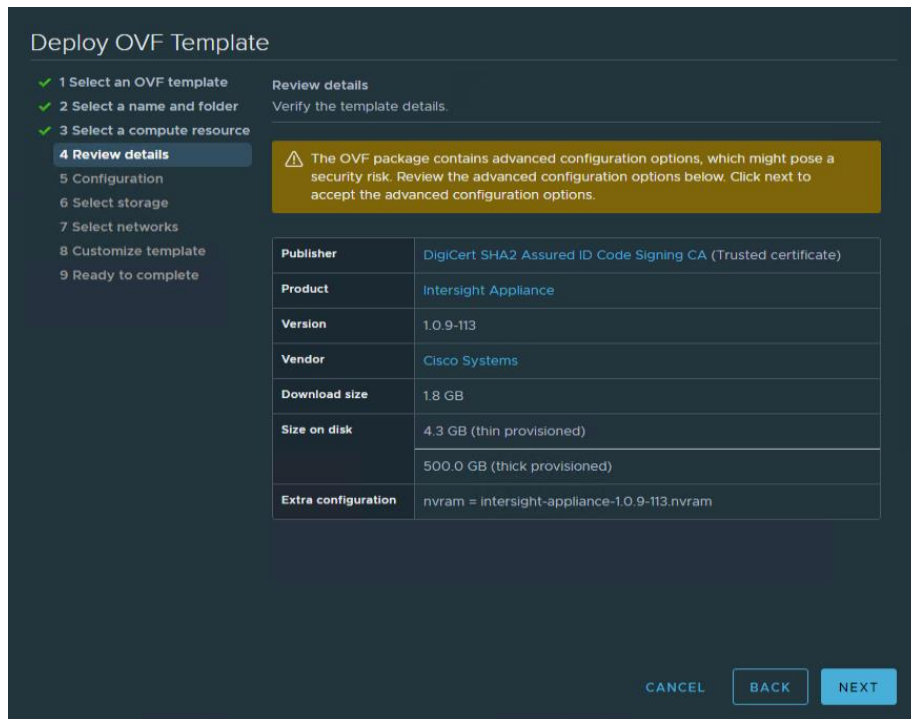
1. Log into VMware vSphere Web Client with administrator credentials.
2. Right-click the host and select Deploy OVF Template.
3. On the Deploy OVF Template wizard Select template page, specify the source location, and click Next. You can specify a URL or browse to location accessible from your local hard drive, a network share, or a DVD/CD drive.

Figure 16. Deploy OVF Template



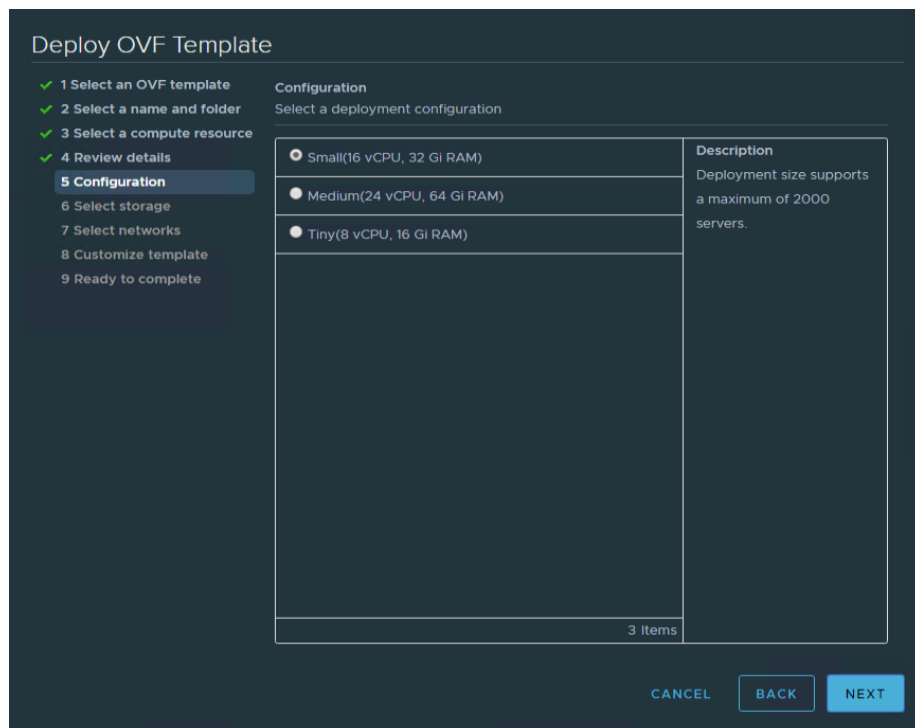
4. On the OVF Template Details page, verify the OVF template details and click Next. No input is necessary.
5. On the Select a name and location page, add/edit the Name and Location for the Virtual appliance, and click Next.
6. On the Select a resource page, select the specific Host (ESX station), Cluster, Resource Pool, or virtual appliance you want to deploy and click Next.
7. Each VM must be assigned to a specific host on clusters that are configured with vSphere HA or Manual mode vSphere DRS.
8. On the Review details page, verify the OVA template details and click Next.

Figure 17. Review Details



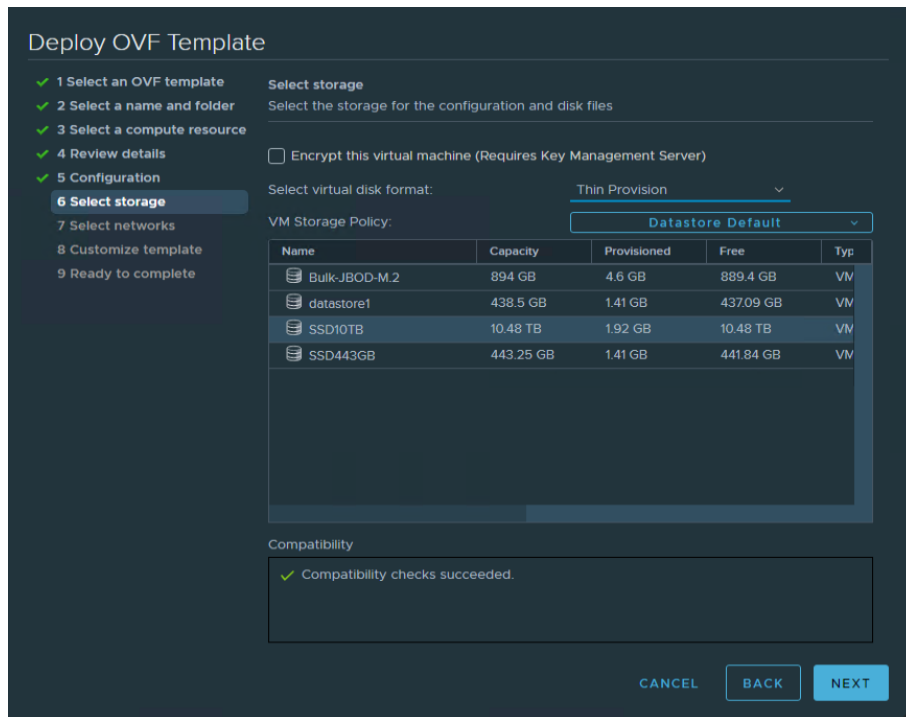
9. On the Configuration page, select a deployment configuration and click Next. You can select Small or Medium deployment configuration based on your requirement for Intersight Virtual Appliance. A brief description of the selected size displays. You can select Tiny (8 vCPU, 16 Gi RAM) deployment configuration for Intersight Assist only.

Figure 18. Select Configuration



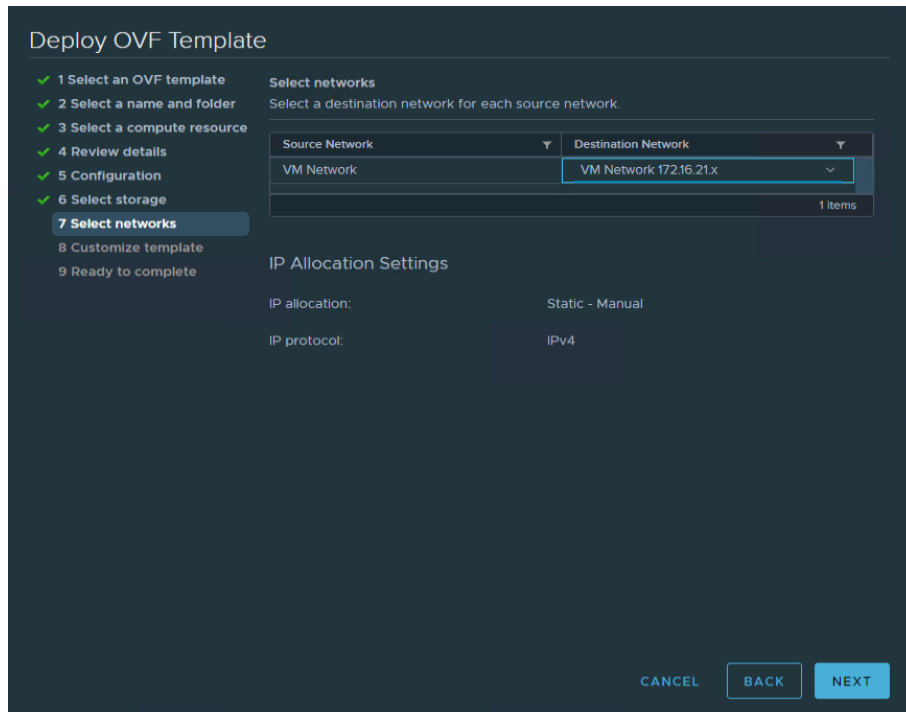
10. On the Select storage page, select a destination storage (hard drives) for the VM files in the selected host (ESX station) and click Next. Select the Disk Format for the virtual machine virtual disks. Select Thin Provision to optimize disk usage.

Figure 19. Select Storage



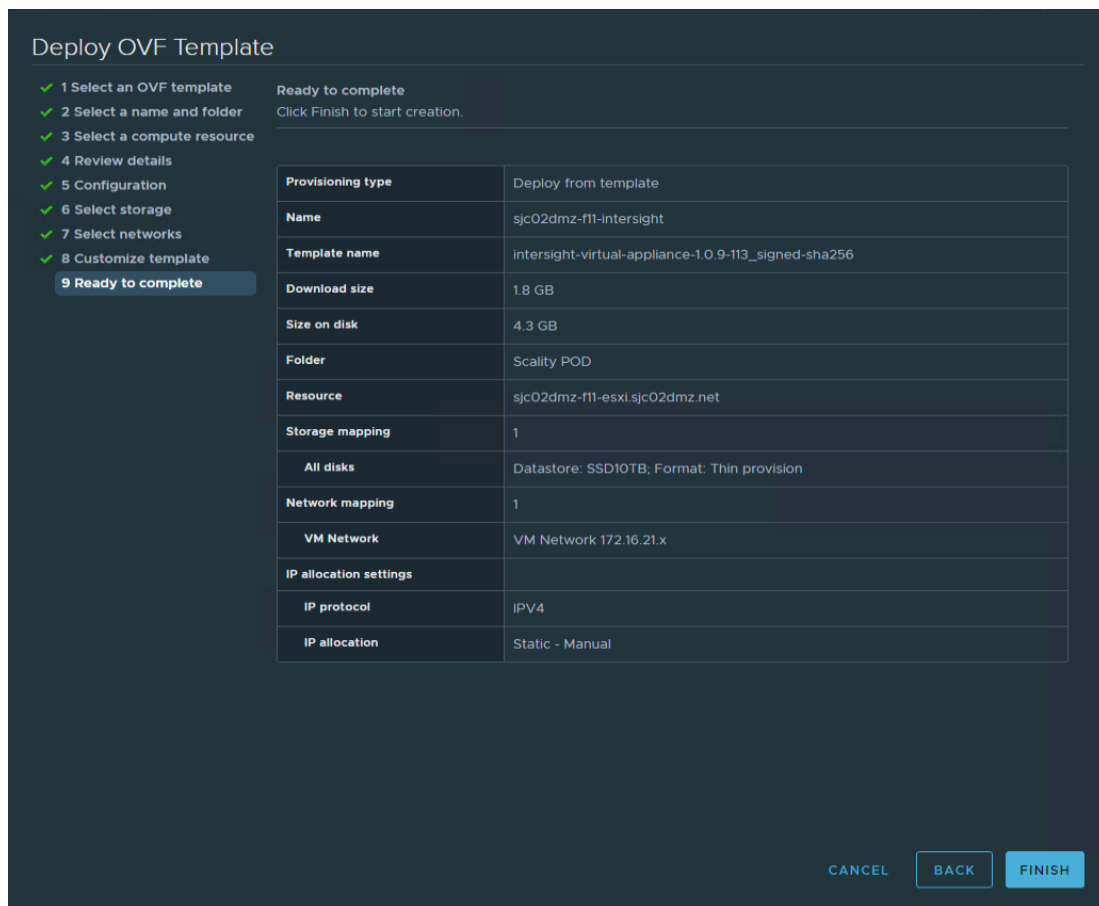
11. On the Select networks page, for each network that is specified in the OVF template, select a source network, and map it to a destination network and click Next.

Figure 20. Select Network



12. On the Customize Template page, customize the deployment properties of the OVF template and click Next.

Figure 21. OVF Template Summary



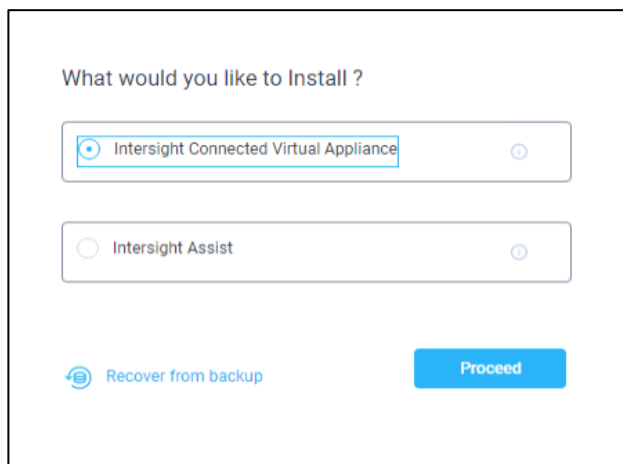
13. After finishing the deployment, power on the virtual machine.

Log into Intersight Virtual Appliance

After installing the Intersight Virtual Appliance OVA, you can connect to the configured IP address or DNS name. To log into the Intersight Virtual Appliance, follow these steps:

1. Select the installation “Intersight Connected Virtual Appliance.”

Figure 22. Select Installation



What would you like to Install ?

Intersight Connected Virtual Appliance

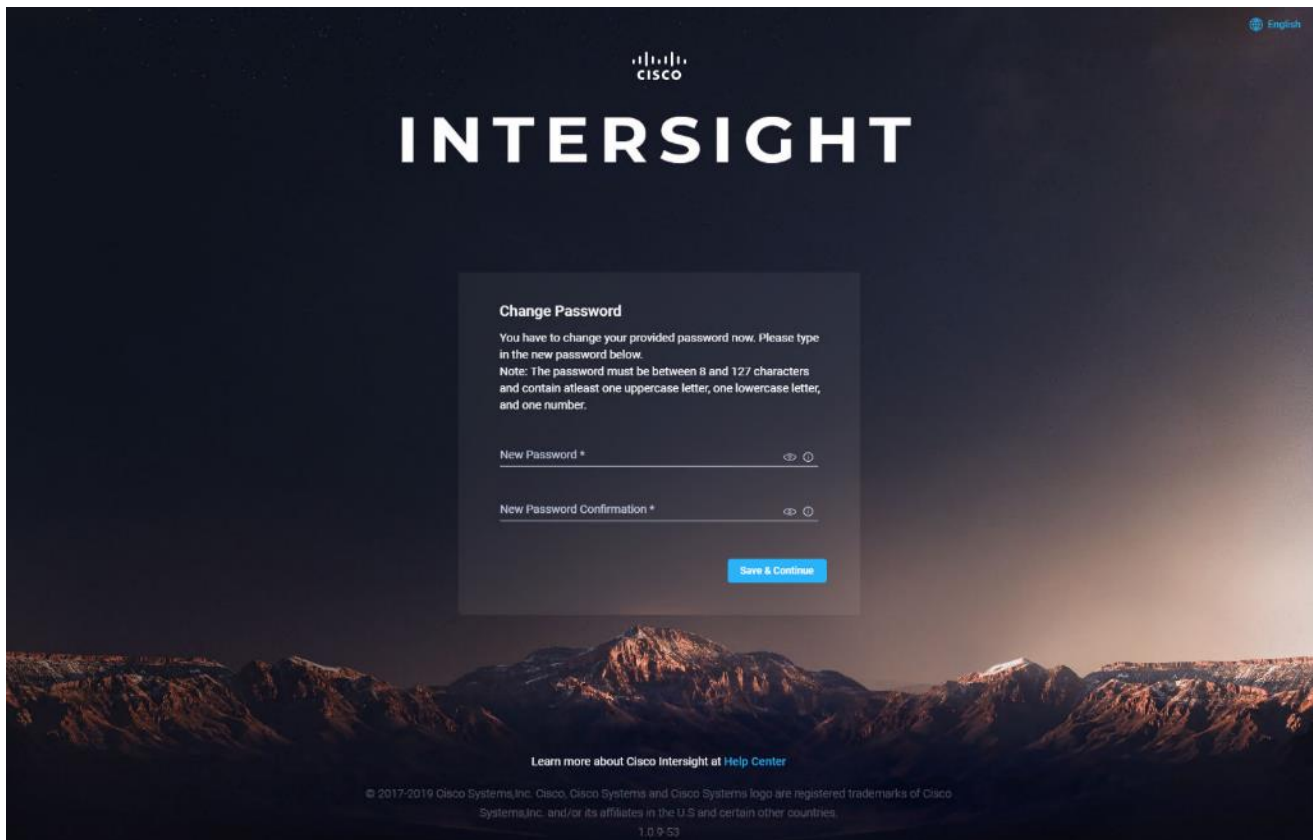
Intersight Assist

[Recover from backup](#) Proceed

2. After you install the Cisco Intersight Virtual Appliance OVA, go to `<<http://your fqdn.com>>`. The Initial Setup Wizard appears and allows you to complete the setup for one of the following:
 - Intersight Connected Virtual Appliance
 - Intersight Assist—For more information, see the [Cisco Intersight Assist documentation](#).
3. Select Intersight Connected Virtual Appliance and click Proceed.

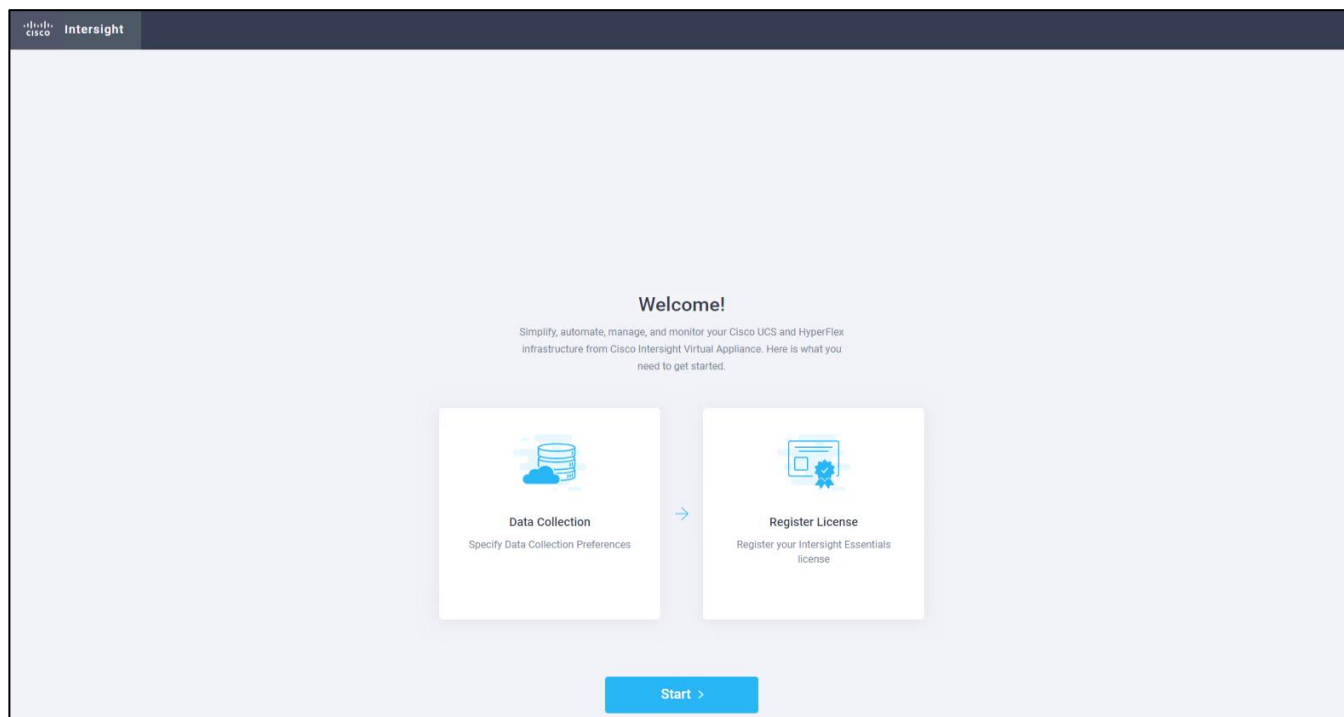
The wizard runs through a series of steps to download and install software packages. You can view the progress of the installation. You can expect this process to complete in about an hour's time. After the formal setup is finished, you're getting redirected to the login page where you have to change the password.

Figure 23. Initial Connection to Intersight



The initial Setup Wizard displays. The wizard enables you to complete the setup of the Intersight appliance.

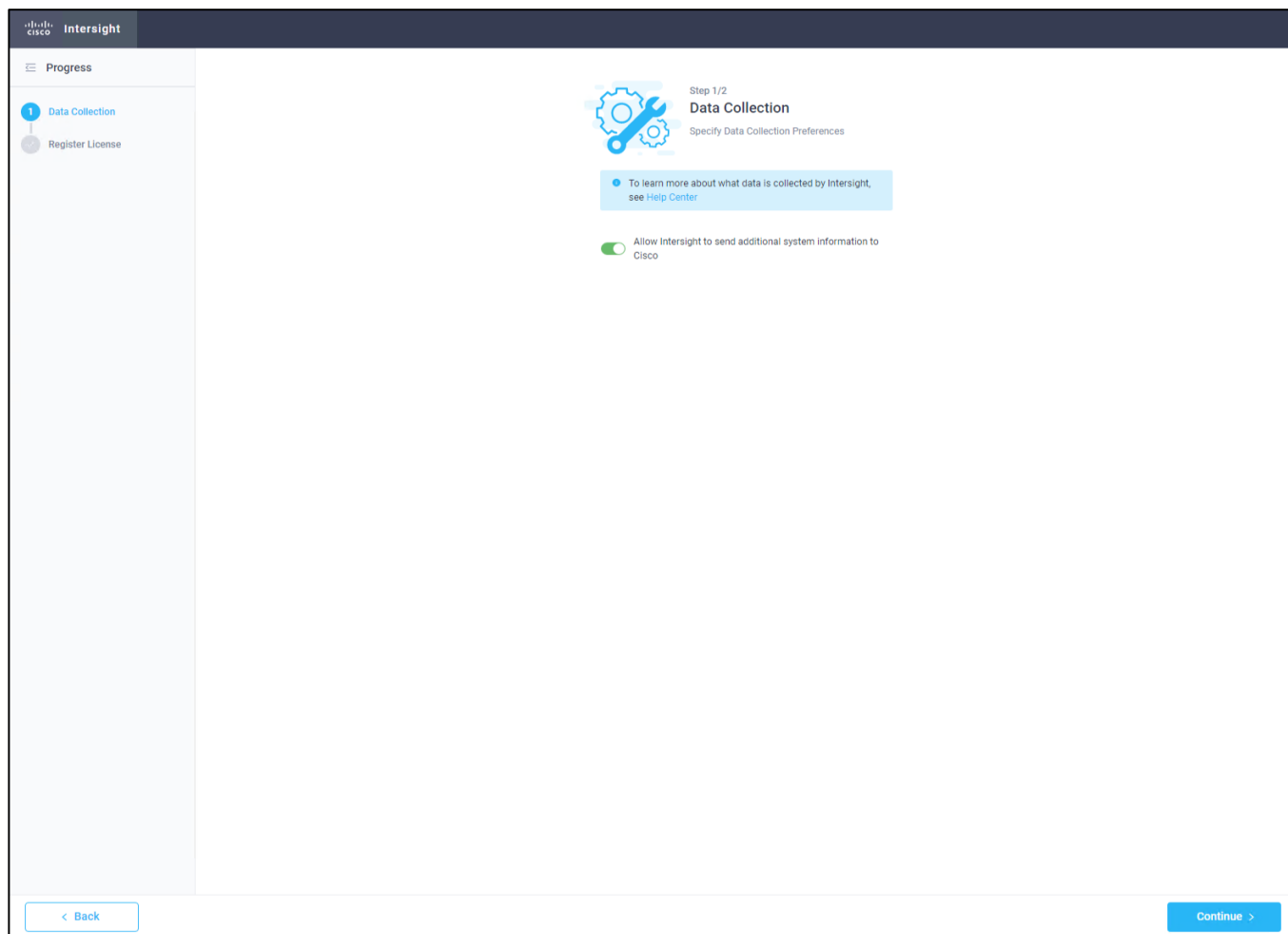
Figure 24. Intersight Setup Wizard



To complete the setup, follow these steps:

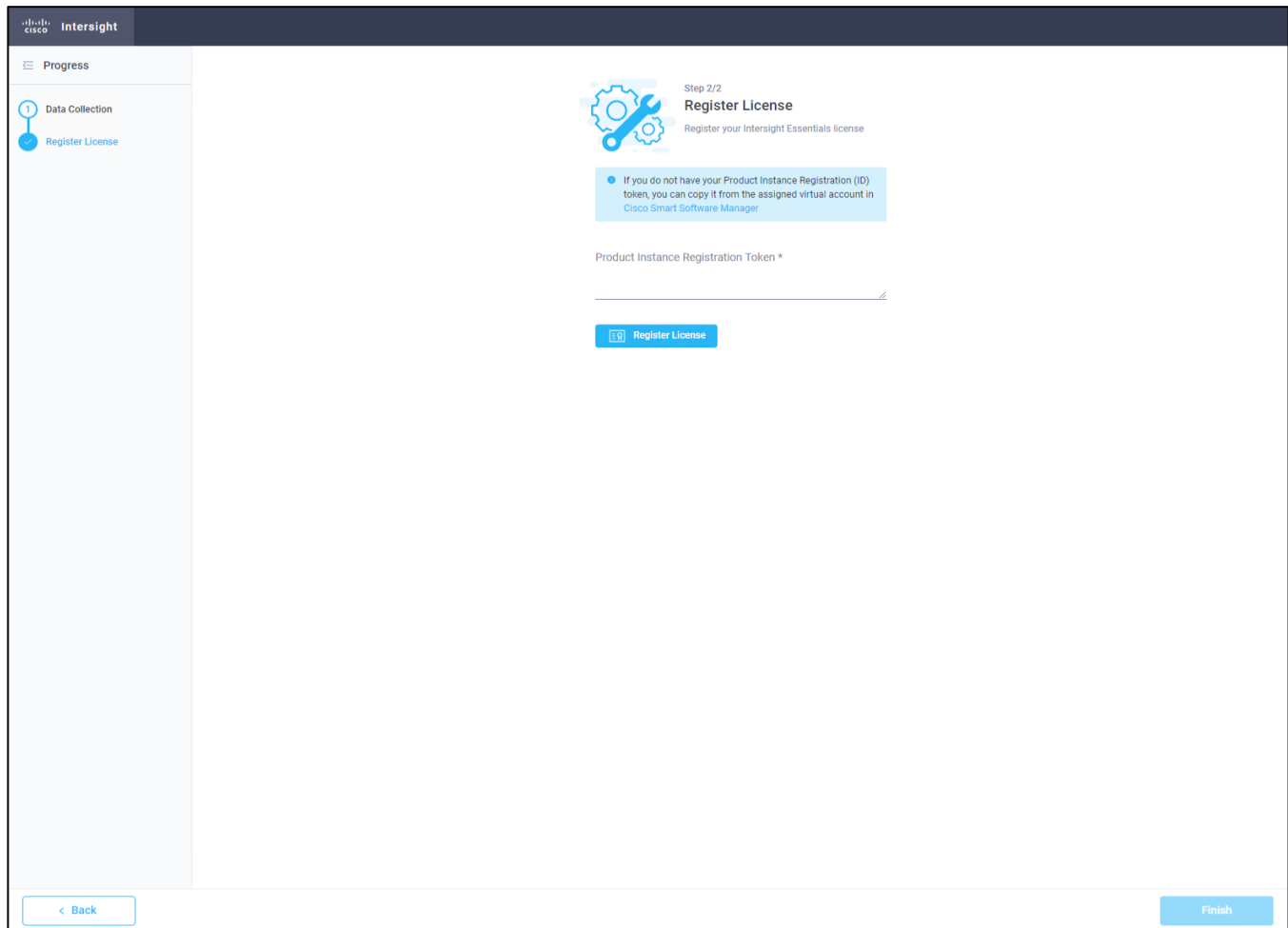
1. **Data Collection**—Specify your preference to allow Intersight to send additional system information to Cisco. This option is enabled by default. For more information about what data is collected by Intersight, see Data Collected from Intersight Virtual Appliance.

Figure 25. Intersight Setup Wizard - Data Collection



2. **Register License**—Click Register License. Obtain a license registration token from Cisco Smart License Manager and apply add the token to activate your license. The license registration process could take a few minutes to complete. For more information about registering your Intersight license, watch [Activating Intersight License](#).

Figure 26. Intersight Register License



3. Click Finish. The Cisco Intersight Virtual Appliance dashboard displays.

Cisco Intersight Virtual Appliance Settings

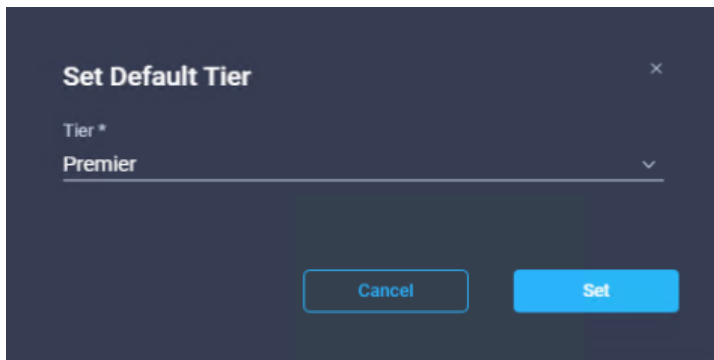
Before you start building the solution, you need configure the virtual appliance for using the correct license and for performing backups.

Change License Tier

You need to use the right license to automatically install an operating system. You need at a minimum, the Advantage license. In our solution we used the Premier license. To change to Premier license, follow these steps:

1. Click Settings/Licensing and then click Set Default Tier.
2. Select Premier and click Set.

Figure 27. Set Intersight License



Back Up Data

Backing up the Cisco Intersight Virtual Appliance regularly is essential. Without regular backups, there is no automatic way to reconstruct the configuration settings and recreating the profiles and policies. You can perform a regular backup once a day using a scheduled backup or create backup on demand if there is a data loss or corruption event. Cisco Intersight Virtual Appliance enables you to take a full state backup of the data in the appliance and store it in a remote server. If there is a total site failure or other disaster recovery scenarios, the restore capability enables you to do a full state system restore from the backed-up system data.

Schedule Backup enables you to schedule a periodic backup of the data in the Intersight Appliance. The Appliance can store three copies of the backup locally on the appliance.

To schedule a backup, follow these steps:

1. Log into Cisco Intersight Virtual Appliance as a user with account administrator role.
2. From the Appliance UI, navigate to Settings icon > Settings > General > Backup, click Schedule Backup.
3. On the Schedule Backup window, toggle ON Use Backup Schedule.



If you disable this option, you must enable the Use Backup Schedule option to schedule a backup.

4. Provide the following details to complete creating the Backup Schedule:
 - a. Backup Schedule
 - b. Day of Week—Specify the day in the week when you want to schedule a data backup.
 - c. Time of Day—Specify the time in the selected day when you want to schedule a data backup. The Time of Day follows the browser time of your session and displays your local time of the day.
 - d. Backup Destination
 - e. Protocol—Communication protocol (SCP/ STFP) used in the backup process.
 - f. Remote Port—Remote TCP port on the backup server.
 - g. Remote Host—The remote host for saving the backup files.
 - h. Remote Path—Directory location where the backup files are saved.

- i. Filename—Name of the backup file to restore
- j. Username—Username for authenticating the backup client to the backup server.
- k. Password—Password for authenticating the backup client to the backup server.
- l. Password Confirmation—Reenter the password and click Schedule Backup to complete the process.

Figure 28. Schedule Backup Configuration

Schedule Backup

You can schedule a full system backup to save on a remote server

Use Backup Schedule

Backup Schedule

Day Of Week
Daily

Time Of Day
Midnight

Backup Destination

Protocol *
scp

Remote Port *
0

Remote Host *
sjc02dmz-centos.sjc02dmz.net

Remote Path
/mnt/nfs/backup/intersight

Filename
intersight-scality

Username *
root

Password *

Password Confirmation *

Cancel Schedule Backup

Claim a Device

Device Connector Requirements

You can claim a device in Cisco Intersight Virtual Appliance through the embedded device connector. Before you claim a device, ensure that the device connector requirements are met. The following table lists the software compatibility and the supported device connector versions for Intersight Virtual Appliance:

Table 10. Device Connector Requirements

Component	Minimum Software Version	Supported Device Connector Version	Versions which include Supported Device Connectors
Cisco UCS Manager	3.2(1)	1.0.9-2290	4.0(2a) or later
Cisco IMC Software	For M5 Servers: 3.1(3a) For M4 Servers: 3.0(4)	1.0.9-335	4.0(2d) or later
HyperFlex Connect and Data Platform	2.6	1.0.9-1335	3.5(2a) or later
Cisco UCS Director	6.7.2.0	1.0.9-911	6.7.2.0

To claim a device, follow these steps:

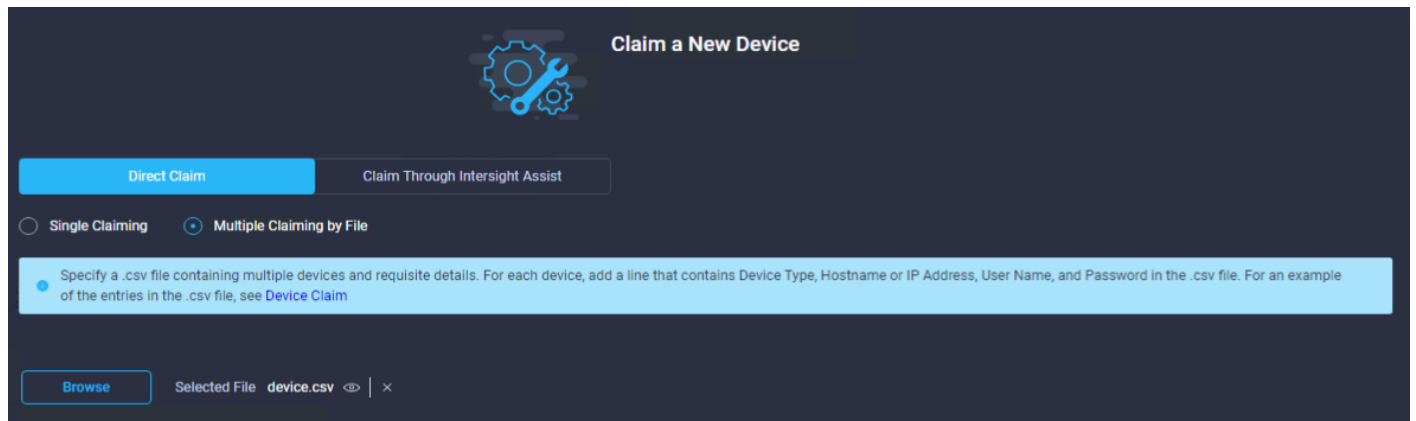
1. Log into the appliance as a user with account administrator privileges.
2. Ensure that you have completed the Cisco Intersight Virtual Appliance OVA installation and set up the appliance.
3. You have an account on the device being claimed that has administrative privileges.
4. You can claim a device or multiple devices in bulk.
5. From Intersight Dashboard > Devices, click Claim a New Device.
6. Select Multiple by File to claim multiple devices using a file.
7. Create a .csv file with the following configuration:

```

IMC,172.16.32.11,admin,<your_password>
IMC,172.16.32.12,admin,<your_password>
IMC,172.16.32.13,admin,<your_password>
IMC,172.16.32.14,admin,<your_password>
IMC,172.16.32.15,admin,<your_password>
IMC,172.16.32.16,admin,<your_password>
IMC,172.16.32.17,admin,<your_password>

```


Figure 29. Claim multiple Devices by File



8. Click Claim and wait for a couple of minutes to get the devices connected with Cisco Intersight.
9. Click Servers to view the discovered UCS servers.

Figure 30. Claimed Devices under Servers

Name	Health	Management IP	Model	C. (CPU)	Memory Capa...	Server Profile	CPUs	CPU Cores	Firmware Ver...	Memory Spee...
sjc02dmz-i14-ceph1	Critical	172.16.32.11	UCSC-C240-M5L	57.6	384.0			2	24 4.1(2b)	2400
sjc02dmz-i14-ceph2	Critical	172.16.32.12	UCSC-C240-M5L	57.6	384.0			2	24 4.1(2b)	2400
sjc02dmz-i14-ceph3	Critical	172.16.32.13	UCSC-C240-M5L	57.6	384.0			2	24 4.1(2b)	2400
sjc02dmz-i14-ceph4	Critical	172.16.32.14	UCSC-C240-M5L	57.6	384.0			2	24 4.1(2b)	2400
sjc02dmz-i14-ceph5	Critical	172.16.32.15	UCSC-C240-M5L	57.6	384.0			2	24 4.1(2b)	2400
sjc02dmz-i14-ceph6	Critical	172.16.32.16	UCSC-C240-M5L	57.6	384.0			2	24 4.1(2b)	2400
sjc02dmz-i14-cephadm	Critical	172.16.32.17	UCSC-C220-M5SX	84.0	384.0			2	40 4.1(1d)	2666

 The critical status comes from having only one PSU connected.

Create a New Organization

An Organization is a logical entity which enables multi-tenancy through separation of devices. Organization allows you to group the devices into logical groups, enabling you to apply the configuration settings on a subset of devices. To create a new organization for all six Ceph OSD servers, follow these steps:

1. Log into your Cisco Intersight virtual Appliance.
2. Go to Settings, Organization and click on Create Organization.
3. Type in a Name and a Description.
4. Search for Type: Standalone M5 Server and select all seven server and click Create.

Figure 31. Create Organization

Edit Organization

Edit an organization to manage access to your logical and physical resources.

General

Name * Description

Memberships

Custom All

Select devices to create a Custom Organization. Profiles and Policies that are created within a Custom Organization are applicable only to devices in the same Organization.

Search: Name ceph x 7 items found 25 per page 1 of 1

Name	Status	Type	Device IP	Device ID
sjc02dmz-i14-ceph3	Connected	Standalone Server	172.16.32.13	WZP21450ZRQ
sjc02dmz-i14-ceph1	Connected	Standalone Server	172.16.32.11	WZP21460GQA
sjc02dmz-i14-ceph2	Connected	Standalone Server	172.16.32.12	WZP21450ZS8
sjc02dmz-i14-ceph4	Connected	Standalone Server	172.16.32.14	WZP21450ZRG
sjc02dmz-i14-ceph5	Connected	Standalone Server	172.16.32.15	WZP21460GQ9
sjc02dmz-i14-ceph6	Connected	Standalone Server	172.16.32.16	WZP21450ZRA
sjc02dmz-i14-cephadm	Connected	Standalone Server	172.16.32.17	WMP242300PR

Selected 7 of 7 Show Selected Unselect All 1 of 1

Create a Terraform Configuration Environment for Cisco Intersight

You need to prepare the environment prior to starting the automated configuration:

- Install Terraform
- Clone Repository
- Copy Terraform provider binary file
- Generate API keys
- Define Cisco Intersight Provider
- Configure Variables

Install Terraform

You will install Terraform on an administration host; in our solution we used a virtual Linux RHEL machine. Hash-iCorp distributes Terraform as a binary package. You can also install Terraform using popular package managers.

To install Terraform, follow these steps:

1. Obtain the [appropriate package](#) for your system and download it as a zip archive.
2. After downloading Terraform, unzip the package. Terraform runs as a single binary named terraform. Any other files in the package can be safely removed and Terraform will still function. (You can also [compile the Terraform binary](#) from source.)
3. Make sure that the terraform binary is available on your PATH. This process will differ depending on your operating system.

```
[root@sjc02dmz-i14-terraform ~]# ll
total 16448
-rw-----. 1 root root      1812 Jun 15 10:49 anaconda-ks.cfg
-rw-r--r--. 1 root root 34869112 Jun 16 06:23 terraform_0.13.5_linux_amd64.zip
[root@sjc02dmz-i14-terraform ~]# unzip terraform_0.13.5_linux_amd64.zip
Archive:  terraform_0.13.5_linux_amd64.zip
  inflating: terraform
[root@sjc02dmz-i14-terraform ~]# ll
total 68080
-rw-----. 1 root root      1812 Jun 15 10:49 anaconda-ks.cfg
-rwxr-xr-x. 1 root root 85545348 May 27 09:38 terraform
-rw-r--r--. 1 root root 34869112 Jun 16 06:23 terraform_0.13.5_linux_amd64.zip
[root@sjc02dmz-i14-terraform ~]# mkdir terraform-intersight
```

4. Move the terraform binary to one of the listed locations. The following command assumes that the binary is currently in your downloads folder and that your PATH includes /usr/local/bin, but you can customize it if your locations are different.

```
[root@sjc02dmz-i14-terraform ~]# echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin
[root@sjc02dmz-i14-terraform ~]# mv ~/terraform /usr/local/bin/terraform
```

5. Verify the installation:

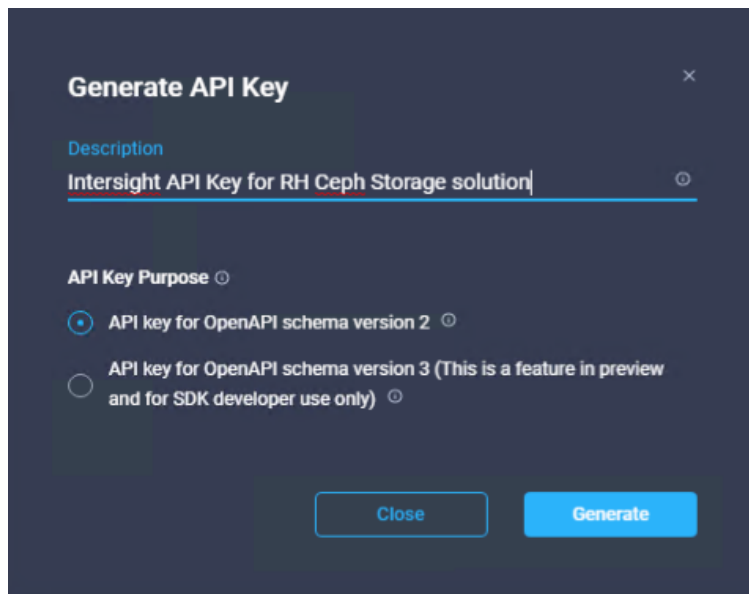
```
[root@sjc02dmz-i14-terraform ~]# terraform -version
Terraform v0.13.5
```

Generate Intersight API Keys

To start using the provider the API Key, Secret Key, and Intersight endpoint URL are required. To generate the API Keys, follow these steps:

1. Log into your Cisco Intersight virtual Appliance.
2. Go to Settings, API Keys and click on Generate API Keys.
3. Enter a description and click Generate.

Figure 32. Generate API Key



4. Copy the API Key.
5. Save the secret key into a .pem file on your Terraform administration host.

Configure Variables

To provision the infrastructure, you'll need to define variables for various workflows. These variables are:

- VLANs
- Remote server hosting images

- Remote server share
- Remote server OS image
- Remote server HUU image
- Remote server protocol
- Manages object ID for all nodes that needs to be provisioned
- Managed object ID for organization

Before you download and store all necessary images, you need to install and configure a http server on the Terraform administration host. You will need to download the specific RHEL ISO images for each Ceph OSD node from the Terraform host:

```
[root@sjc02dmz-i14-terraform]# dnf -y install httpd genisoimage
[root@sjc02dmz-i14-terraform]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service →
/usr/lib/systemd/system/httpd.service.
[root@sjc02dmz-i14-terraform]# systemctl start httpd
[root@sjc02dmz-i14-terraform]# firewall-cmd --zone=public --permanent --add-
service=http
success
[root@sjc02dmz-i14-terraform]# firewall-cmd --reload
Success
[root@sjc02dmz-i14-terraform]# mkdir -p /var/www/html/images
```

In the next step, create a Terraform module, which helps to get the MOID for each server, the organization, and the catalog for the OS installation. A module is a container for multiple resources that are used together. Every Terraform configuration has at least one module, known as its *root module*, which consists of the resources defined in the .tf files in the main working directory. A module can call other modules, which lets you include the child module's resources into the configuration in a concise way. Modules can also be called multiple times, either within the same configuration or in separate configurations, allowing resource configurations to be packaged and re-used.

Create a subdirectory where you can configure a [main.tf](#) and [variables.tf](#) file for the module. This ensures that the MOIDs are automatically retrieved for the main configuration file:

```
[root@sjc02dmz-i14-terraform]# mkdir terraform-intersight-moids
[root@sjc02dmz-i14-terraform ~]# cd terraform-intersight-moids
[root@sjc02dmz-i14-terraform terraform-intersight-moids]# vi variables.tf
# Server and Organization names
variable "server_names" {
    type = list
}

variable "organization_name" {}
```

```
variable "catalog_name" {}

[root@sjc02dmz-il14-terraform terraform-inter sight-moids]# vi main.tf
# Intersight Provider Information
terraform {
  required_providers {
    intersight = {
      source = "CiscoDevNet/intersight"
      version = ">=0.1.3"
    }
  }
}

data "intersight_compute_physical_summary" "server_moid" {
  name = var.server_names[count.index]
  count = length(var.server_names)
}

output "server_moids" {
  value = data.intersight_compute_physical_summary.server_moid
}

data "intersight_organization_organization" "organization_moid" {
  name = var.organization_name
}

output "organization_moid" {
  value = data.intersight_organization_organization.organization_moid.moid
}

data "intersight_softwarerepository_catalog" "catalog_moid" {
  name = var.catalog_name
}

output "catalog_moid" {
  value = data.intersight_softwarerepository_catalog.catalog_moid.moid
}
```

Create another subdirectory for the configuration files for the Ceph OSD nodes and configure the main [variables file](#). Split the whole configuration into several tasks to have a better overview and with that having different sub-directories:

```
[root@sjc02dmz-i14-terraform]# mkdir terraform-inter sight-sds
[root@sjc02dmz-i14-terraform ~]# cd terraform-inter sight-sds
[root@sjc02dmz-i14-terraform]# mkdir create_policy_profile deploy_profile install_os
[root@sjc02dmz-i14-terraform terraform-inter sight-sds]# cd create_policy_profile
[root@sjc02dmz-i14-terraform create_policy_profile]# vi variables.tf
//Define all the basic variables here

variable "api_private_key" {
    default = "/root/terraform-inter sight-sds/inter sight.pem"
}

variable "api_key_id" {
    default =
"5e5fb2b17564612d3028b5b4/5e5fbd137564612d3028bcc4/5fa1a9107564612d3007f934"
}

variable "api_endpoint" {
    default = "https://sjc02dmz-inter sight.sjc02dmz.net"
}

variable "management_vlan" {
    default = 300
}

variable "client_vlan" {
    default = 301
}

variable "storage_vlan" {
    default = 302
}

variable "remote-server" {
    default = "sjc02dmz-i14-terraform.sjc02dmz.net"
}
```

```
variable "remote-share" {
  default = "/images"
}

variable "remote-os-image-cephosd" {
  type = list(string)
  default = ["rhel8.2-cephosd1.iso", "rhel8.2-cephosd2.iso", "rhel8.2-cephosd3.iso",
"rhel8.2-cephosd4.iso", "rhel8.2-cephosd5.iso", "rhel8.2-cephosd6.iso", "rhel8.2-
cephadm.iso"]
}

variable "remote-os-image-link" {
  type = list(string)
  default = ["http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephosd1.iso",
"http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephosd2.iso",
"http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephosd3.iso",
"http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephosd4.iso",
"http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephosd5.iso",
"http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephosd6.iso",
"http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephadm.iso"]
}

variable "remote-protocol" {
  default = "softwarerepository.HttpServer"
}

variable "server_names" {
  default = ["sjc02dmz-i14-ceph1", "sjc02dmz-i14-ceph2", "sjc02dmz-i14-ceph3",
"sjc02dmz-i14-ceph4", "sjc02dmz-i14-ceph5", "sjc02dmz-i14-ceph6", "sjc02dmz-i14-
cephadm"]
}

variable "organization_name" {
  default = "Ceph"
}

variable "server_profile_action" {
  default = "No-op"
}

variable "catalog_name" {
  default = "appliance-system-catalog"
```

```
}
```

Define Cisco Intersight Provider

The Cisco Intersight Provider can be found at

<https://registry.terraform.io/providers/CiscoDevNet/intersight/latest> To enable and define the Cisco Intersight Provider for the RH Ceph Storage solution, follow these steps, also documented at the right site of the web page under “Use Provider:”

1. On the Terraform administration host, go into one of the subdirectories under terraform-intersight-sds and create a main.tf file:

```
terraform {
  required_providers {
    intersight = {
      source = "CiscoDevNet/intersight"
      version = ">=0.1.3"
    }
  }
}

provider "intersight" {
  apikey      = var.api_key_id
  secretkeyfile = var.api_private_key
  endpoint    = var.api_endpoint
}
```



This step will be used for all tasks in each main.tf file.

Understand Cisco Intersight Provider and Terraform Configuration

The following is an example code snippet for creating a specific vNIC from the infrastructure file (understand and create the main configuration file):

```
resource "intersight_vnic_eth_if" "eth0" { -> Define the resource
  name = "eth0"
  order = 0
  placement { -> Define the placement of the vNIC
    id = "MLOM"
    pci_link = 0
    uplink = 0
  }
  cdn {
    nr_source = "vnic"
  }
}
```



```

vmq_settings {
    enabled = false
    num_interrupts = 1
    num_vmps = 1
}
lan_connectivity_policy { -> Define LAN Connectivity Policy to use
    moid      = intersight_vnic_lan_connectivity_policy.ceph-lan-connectivity-
policy.id
    object_type = "vnic.LanConnectivityPolicy"
}
eth_network_policy { -> Define the Network Policy to use
    moid = intersight_vnic_eth_network_policy.ceph-mgt-network.id
}
eth_adapter_policy { -> Define the Adapter Policy to use
    moid = intersight_vnic_eth_adapter_policy.ceph-ethernet-adapter-policy.id
}
eth_qos_policy { -> Define the QoS Policy to use
    moid = intersight_vnic_eth_qos_policy.ceph-ethernet-qos-policy.id
}
}

```

Each resource is assigned a name, which can later be used for tracking and referencing. This name will not be reflected anywhere in the Cisco Intersight platform. It is only for reference among the .tf files. A resource can point to or reference another resource using the format <resource>.<resource_name>.<property_name>

Documentation about provider resources and configuration options can be found at <https://github.com/CiscoDevNet/terraform-provider-intersight/tree/master/website/docs>.

The [Appendix](#) contains the configuration file for reference.

Implement Terraform Configuration – Init, Plan, Apply

After creating all the configuration files and the main infrastructure files, the next step is to validate and deploy the configuration.

```
terraform plan
```

The terraform plan command is used to create an execution plan. Terraform performs a refresh, unless explicitly disabled, and then determines what actions are necessary to achieve the desired state specified in the configuration files.

This command is a convenient way to check whether the execution plan for a set of changes matches the expectations without making any changes to real resources or to the state. For example, terraform plan might be run before committing a change to version control, to create confidence that it will behave as expected.

In the output, the symbols show you the following:

-
- Resources with a plus sign (+) will be created.
 - Resources with a minus sign (-) will be deleted.
 - Resources with a tilde (~) will be modified in place.

The terraform apply command is used to apply the changes required to reach the desired state of the configuration, or the pre-determined set of actions generated by a terraform plan execution plan.

Configure Red Hat Ceph Storage Infrastructure with Terraform

The configuration to automatically prepare the environment for the following Red Hat Ceph Storage installation consists of three steps. All these steps were run in sub-directories for a better overview. They can also run in just one configuration file and one directory.

1. Create Policies and Profiles for 6 x Cisco UCS C240 M5 and 1 x Cisco UCS C220 M5.
2. Deploy Profiles.
3. Install custom RHEL 8 ISO images on all nodes.

Create Cisco Intersight Policies and Profiles with Terraform

We can now start creating the policies we need for the Red Hat Ceph Storage solution and build the server profiles out of the policies. The full configuration file is in the [Appendix](#). The following policies will be built by Terraform:

Table 11. Terraform Provider Policies and Resource Objects

Policy	Terraform Resource Object	Comments
Adapter Configuration	intersight_adapter_config_policy	Specify the PCI slot ID where the Cisco VIC adapter is placed and set FEC mode for 25G connectivity. Configured: <ul style="list-style-type: none">• Slot MLOM• FEC mode cl74
Ethernet Adapter	intersight_vnic_eth_adapter_policy	Specify the adapter properties to improve the throughput over network. Configured: <ul style="list-style-type: none">• Interrupt 32• Completion 16• Rx count 8 and ring size 4096• Tx count 8 and ring size 4096• RSS true
Ethernet Network	intersight_vnic_eth_network_policy	Specify the network and VLANs used for Ceph, in our cases three networks with different VLANs. Configured: <ul style="list-style-type: none">• Management network VLAN 300

Policy	Terraform Resource Object	Comments
		<ul style="list-style-type: none"> Client network VLAN 301 Storage network VLAN 302
Ethernet QoS	intersight_vnic_eth_qos_policy	<p>Specify the Quality of Service with MTU size 9000.</p> <p>Configured:</p> <ul style="list-style-type: none"> MTU 9000
LAN Connectivity	inter-sight_vnic_lan_connectivity_policy intersight_vnic_eth_if	<p>Specify the LAN connectivity with the vNICs.</p> <p>Configured:</p> <ul style="list-style-type: none"> eth0 (uplink port 0) for Management network eth1 (uplink port 0), eth2 (uplink port 1) for Client network eth3 (uplink port 0), eth4 (uplink port 1) for Storage network
NTP	intersight_ntp_policy	<p>Specify the NTP servers to be used</p> <p>Configured:</p> <ul style="list-style-type: none"> NTP IP 173.38.201.115
Disk Group	intersight_storage_disk_group_policy	<p>Specify the disk group policies for Boot disks (RAID 1).</p> <p>Configured:</p> <ul style="list-style-type: none"> Slot 13 and 14 RAID 1 for Boot (C240 M5) Slot 1 and 2 RAID 1 for Boot (C220 M5)
Storage	intersight_storage_storage_policy	<p>Specify the Storage Policies with the previous created disk group policies.</p> <p>Configured:</p> <ul style="list-style-type: none"> Virtual disk for Boot All in common: ReadWrite, ReadAhead, WriteBackGood-BBU
Boot Order	intersight_boot_precision_policy	Specify the boot order.

Policy	Terraform Resource Object	Comments
		Configured: <ul style="list-style-type: none"> • Local disk from MRAID • Virtual media from CIMC mapped DVD

After creating the policies, the same task creates the server profiles for all six Ceph OSD nodes. To start building the policies and profiles, log into the Terraform administration host and follow these steps:

```
[root@sjc02dmz-114-terraform ~]# cd terraform-inter sight-sds/create_policy_profile/
[root@sjc02dmz-114-terraform create_policy_profile]# terraform init
Initializing modules...
```

```
Initializing the backend...
```

```
Initializing provider plugins...
```

```
- Finding cisco devnet/inter sight versions matching "0.1.3, 0.1.3"...
```

```
- Installing cisco devnet/inter sight v0.1.3...
```

```
- Installed cisco devnet/inter sight v0.1.3 (signed by a HashiCorp partner, key ID
7FA19DB0A5A44572)
```

```
Partner and community providers are signed by their developers.
```

```
If you'd like to know more about provider signing, you can read about it here:
```

```
https://www.terraform.io/docs/plugins/signing.html
```

```
Terraform has been successfully initialized!
```

```
You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.
```

```
If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
```

```
[root@sjc02dmz-114-terraform create_policy_profile]# terraform plan
```

```
... -> We skipped the full output since it is very lengthy.
```

```
Plan: 27 to add, 0 to change, 0 to destroy.
```

```
[root@sjc02dmz-114-terraform create_policy_profile]# terraform apply
```

```
...
```

```
Plan: 27 to add, 0 to change, 0 to destroy.
```

Do you want to perform these actions?

Terraform will perform the actions described above.

Only 'yes' will be accepted to approve.

Enter a value: yes

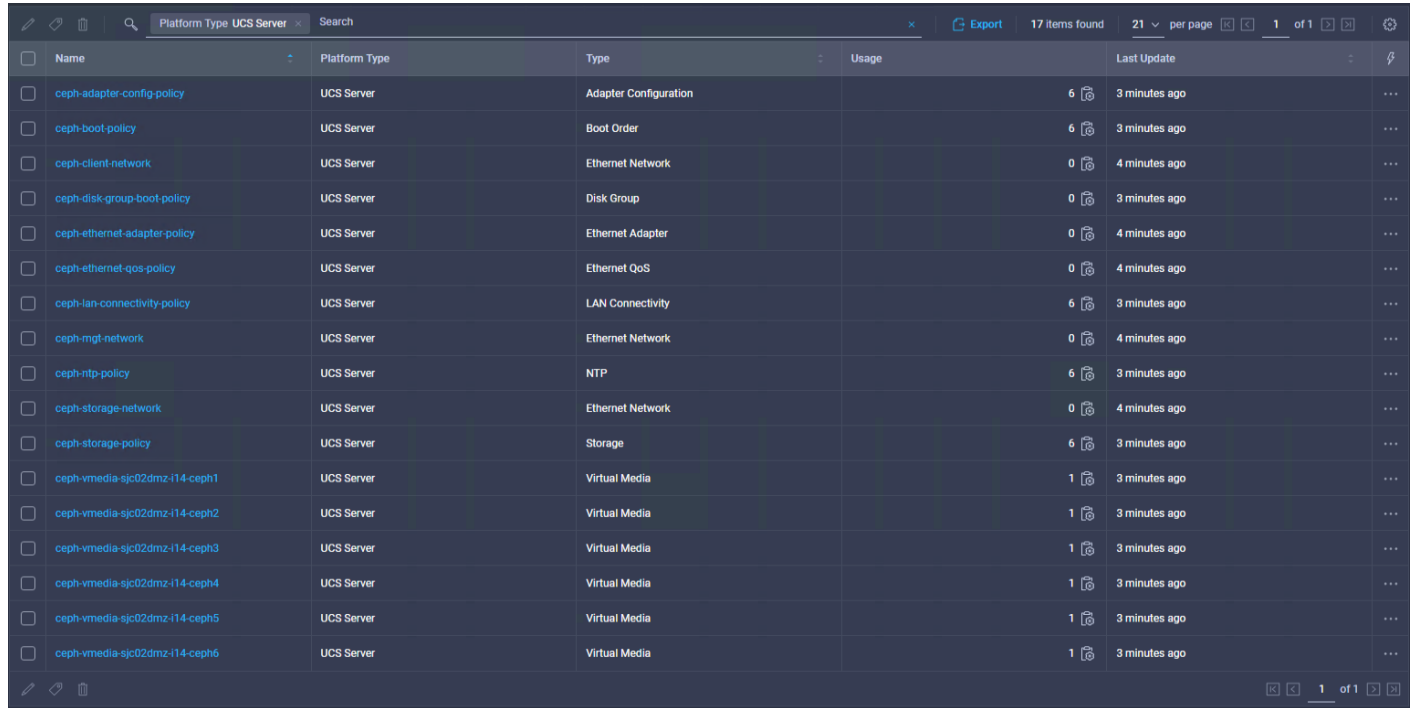
```
intersight_server_profile.cephosd[1]: Creating...
intersight_vnic_eth_qos_policy.ceph-ethernet-qos-1500-policy: Creating...
intersight_vnic_eth_network_policy.ceph-client-network: Creating...
intersight_server_profile.cephosd[0]: Creating...
intersight_vnic_eth_network_policy.ceph-storage-network: Creating...
intersight_storage_disk_group_policy.ceph-disk-group-boot-policy-c240: Creating...
intersight_storage_disk_group_policy.ceph-disk-group-boot-policy-c220: Creating...
intersight_vnic_eth_qos_policy.ceph-ethernet-qos-9000-policy: Creating...
intersight_server_profile.cephosd[3]: Creating...
intersight_vnic_eth_network_policy.ceph-mgt-network: Creating...
intersight_server_profile.cephosd[1]: Creation complete after 1s
[id=60337a7277696e2d30a35434]
intersight_server_profile.cephosd[2]: Creating...
intersight_vnic_eth_network_policy.ceph-storage-network: Creation complete after 1s
[id=60337a73f7c9bab50596fda7]
intersight_vnic_eth_adapter_policy.ceph-ethernet-adapter-policy: Creating...
intersight_storage_disk_group_policy.ceph-disk-group-boot-policy-c220: Creation complete after 1s
[id=60337a73656f6e2d300c639b]
intersight_server_profile.cephosd[4]: Creating...
intersight_vnic_eth_network_policy.ceph-mgt-network: Creation complete after 1s
[id=60337a73f7c9bab50596fdad]
intersight_server_profile.cephosd[6]: Creating...
intersight_vnic_eth_network_policy.ceph-client-network: Creation complete after 1s
[id=60337a73f7c9bab50596fdb3]
intersight_server_profile.cephosd[5]: Creating...
intersight_vnic_eth_qos_policy.ceph-ethernet-qos-9000-policy: Creation complete after 1s
[id=60337a73f7c9bab50596fdb9]
intersight_storage_disk_group_policy.ceph-disk-group-boot-policy-c240: Creation complete after 1s
[id=60337a73656f6e2d300c63a1]
intersight_vnic_eth_qos_policy.ceph-ethernet-qos-1500-policy: Creation complete after 1s
[id=60337a73f7c9bab50596fdbf]
intersight_server_profile.cephosd[0]: Creation complete after 1s
[id=60337a7377696e2d30a35440]
intersight_server_profile.cephosd[3]: Creation complete after 1s
[id=60337a7377696e2d30a3544c]
```

```
intersight_server_profile.cephosd[2]: Creation complete after 0s
[id=60337a7477696e2d30a35458]
intersight_vnic_eth_adapter_policy.ceph-ethernet-adapter-policy: Creation complete af-
ter 0s [id=60337a74f7c9bab50596fdce]
intersight_server_profile.cephosd[4]: Creation complete after 0s
[id=60337a7477696e2d30a35464]
intersight_server_profile.cephosd[6]: Creation complete after 0s
[id=60337a7477696e2d30a35470]
intersight_storage_storage_policy.ceph-storage-policy-admin: Creating...
intersight_server_profile.cephosd[5]: Creation complete after 1s
[id=60337a7477696e2d30a3547c]
intersight_networkconfig_policy.ceph-network-policy: Creating...
intersight_vnic_lan_connectivity_policy.ceph-lan-connectivity-policy: Creating...
intersight_adapter_config_policy.ceph-adapter-config-policy: Creating...
intersight_boot_precision_policy.ceph-boot-policy: Creating...
intersight_ntp_policy.ceph-ntp-policy: Creating...
intersight_storage_storage_policy.ceph-storage-policy-osd: Creating...
intersight_storage_storage_policy.ceph-storage-policy-admin: Creation complete after 1s
[id=60337a74656f6e2d300c63ad]
intersight_networkconfig_policy.ceph-network-policy: Creation complete after 0s
[id=60337a746275722d308cdea4]
intersight_adapter_config_policy.ceph-adapter-config-policy: Creation complete after 0s
[id=60337a74f7c9bab50596fde3]
intersight_boot_precision_policy.ceph-boot-policy: Creation complete after 0s
[id=60337a746275722d308cdec5]
intersight_vnic_lan_connectivity_policy.ceph-lan-connectivity-policy: Creation complete
after 0s [id=60337a74f7c9bab50596fe0e]
intersight_storage_storage_policy.ceph-storage-policy-osd: Creation complete after 0s
[id=60337a75656f6e2d300c63c3]
intersight_vnic_eth_if.eth2: Creating...
intersight_vnic_eth_if.eth1: Creating...
intersight_vnic_eth_if.eth0: Creating...
intersight_vnic_eth_if.eth3: Creating...
intersight_vnic_eth_if.eth4: Creating...
intersight_ntp_policy.ceph-ntp-policy: Creation complete after 0s
[id=60337a756275722d308cdef8]
intersight_vnic_eth_if.eth2: Creation complete after 1s [id=60337a75f7c9bab50596feab]
intersight_vnic_eth_if.eth3: Creation complete after 1s [id=60337a75f7c9bab50596febd]
intersight_vnic_eth_if.eth1: Creation complete after 1s [id=60337a75f7c9bab50596fec3]
intersight_vnic_eth_if.eth4: Creation complete after 1s [id=60337a75f7c9bab50596fed7]
intersight_vnic_eth_if.eth0: Creation complete after 1s [id=60337a75f7c9bab50596fedd]
```

Apply complete! Resources: 27 added, 0 changed, 0 destroyed.

You can now view in Intersight under Policies the newly created policies.

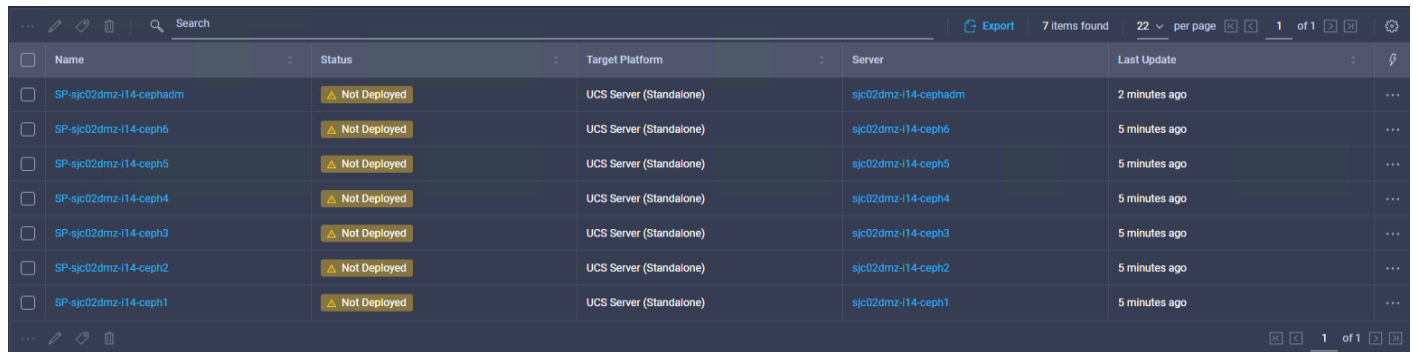
Figure 33. Cisco Intersight Server Policies after Terraform task



Name	Platform Type	Type	Usage	Last Update
ceph-adapter-config-policy	UCS Server	Adapter Configuration	6	3 minutes ago
ceph-boot-policy	UCS Server	Boot Order	6	3 minutes ago
ceph-client-network	UCS Server	Ethernet Network	0	4 minutes ago
ceph-disk-group-boot-policy	UCS Server	Disk Group	0	3 minutes ago
ceph-ethernet-adapter-policy	UCS Server	Ethernet Adapter	0	4 minutes ago
ceph-ethernet-qos-policy	UCS Server	Ethernet QoS	0	4 minutes ago
ceph-lan-connectivity-policy	UCS Server	LAN Connectivity	6	3 minutes ago
ceph-mgt-network	UCS Server	Ethernet Network	0	4 minutes ago
ceph-ntp-policy	UCS Server	NTP	6	3 minutes ago
ceph-storage-network	UCS Server	Ethernet Network	0	4 minutes ago
ceph-storage-policy	UCS Server	Storage	6	3 minutes ago
ceph-vmedia-sjc02dmz-114-ceph1	UCS Server	Virtual Media	1	3 minutes ago
ceph-vmedia-sjc02dmz-114-ceph2	UCS Server	Virtual Media	1	3 minutes ago
ceph-vmedia-sjc02dmz-114-ceph3	UCS Server	Virtual Media	1	3 minutes ago
ceph-vmedia-sjc02dmz-114-ceph4	UCS Server	Virtual Media	1	3 minutes ago
ceph-vmedia-sjc02dmz-114-ceph5	UCS Server	Virtual Media	1	3 minutes ago
ceph-vmedia-sjc02dmz-114-ceph6	UCS Server	Virtual Media	1	3 minutes ago

Under Profiles you can view the seven new server profiles for the Ceph OSD nodes.

Figure 34. Cisco Intersight Server Profiles after Terraform task



Name	Status	Target Platform	Server	Last Update
SP-sjc02dmz-114-cephadm	Not Deployed	UCS Server (Standalone)	sjc02dmz-114-cephadm	2 minutes ago
SP-sjc02dmz-114-ceph6	Not Deployed	UCS Server (Standalone)	sjc02dmz-114-ceph6	5 minutes ago
SP-sjc02dmz-114-ceph5	Not Deployed	UCS Server (Standalone)	sjc02dmz-114-ceph5	5 minutes ago
SP-sjc02dmz-114-ceph4	Not Deployed	UCS Server (Standalone)	sjc02dmz-114-ceph4	5 minutes ago
SP-sjc02dmz-114-ceph3	Not Deployed	UCS Server (Standalone)	sjc02dmz-114-ceph3	5 minutes ago
SP-sjc02dmz-114-ceph2	Not Deployed	UCS Server (Standalone)	sjc02dmz-114-ceph2	5 minutes ago
SP-sjc02dmz-114-ceph1	Not Deployed	UCS Server (Standalone)	sjc02dmz-114-ceph1	5 minutes ago

The formal task of creating policies and profiles is now finished and in the next step the server profiles get associated with the selected servers.

Associate Cisco Intersight Profiles with Terraform

In the next step, associate the former created server profiles with Terraform to the physical servers. Use the same variables.tf file from the task before. The configuration file is located in the deploy profile subdirectory and can be seen in the [Appendix](#) section. Log into the Terraform administration server and run the following commands:


```
[root@sjc02dmz-i14-terraform ~]# cd terraform-inter sight-sds/deploy_profile/
[root@sjc02dmz-i14-terraform deploy_profile]# cp ../create_policy_profile/variables.tf
.
[root@sjc02dmz-i14-terraform deploy_profile]# terraform apply
... -> We skip the full output as it is very lengthy.
Plan: 7 to add, 0 to change, 0 to destroy.
-----
Plan: 7 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

intersight_server_profile.cephosd[2]: Creating...
intersight_server_profile.cephosd[5]: Creating...
intersight_server_profile.cephosd[1]: Creating...
intersight_server_profile.cephosd[3]: Creating...
intersight_server_profile.cephosd[0]: Creating...
intersight_server_profile.cephosd[4]: Creating...
intersight_server_profile.cephosd[6]: Creating...
intersight_server_profile.cephosd[2]: Creation complete after 1s
[id=5fbf4a3777696e2d30481278]
intersight_server_profile.cephosd[5]: Creation complete after 1s
[id=5fbf4a3677696e2d3048126c]
intersight_server_profile.cephosd[1]: Creation complete after 1s
[id=5fbf4a3777696e2d30481290]
intersight_server_profile.cephosd[0]: Creation complete after 1s
[id=5fbf4a3777696e2d30481284]
intersight_server_profile.cephosd[4]: Creation complete after 1s
[id=5fbf4a3677696e2d30481260]
intersight_server_profile.cephosd[3]: Creation complete after 1s
[id=5fbf4a3777696e2d3048129c]
intersight_server_profile.cephosd[6]: Creation complete after 1s
[id=5fbf4a3777696e2d304812a8]

Apply complete! Resources: 7 added, 0 changed, 0 destroyed.
```

Verify the deployment by checking the status in Cisco Intersight under Profiles.

Figure 35. Server Profile Deployment Status



Name	Status	Target Platform	Server	Last Update
SP-sjc02dmz-114-ceph1	OK	UCS Server (Standalone)	sjc02dmz-114-ceph1	Nov 13, 2020 3:54 AM
SP-sjc02dmz-114-ceph6	OK	UCS Server (Standalone)	sjc02dmz-114-ceph6	Nov 13, 2020 3:27 AM
SP-sjc02dmz-114-ceph5	OK	UCS Server (Standalone)	sjc02dmz-114-ceph5	Nov 13, 2020 3:27 AM
SP-sjc02dmz-114-ceph3	OK	UCS Server (Standalone)	sjc02dmz-114-ceph3	Nov 13, 2020 3:26 AM
SP-sjc02dmz-114-ceph4	OK	UCS Server (Standalone)	sjc02dmz-114-ceph4	Nov 13, 2020 3:26 AM
SP-sjc02dmz-114-ceph2	OK	UCS Server (Standalone)	sjc02dmz-114-ceph2	Nov 13, 2020 3:26 AM

Prepare Custom RHEL ISO Images for Automated Installation

Before installing the OS, you need to create the custom images for each Ceph OSD node. For a goal, you want to have seven individual installations, each for every single OSD node and one for the Admin node. For that, we created custom ISO files, which include a kickstart file with all the specific details for the OSD node. An example kickstart file for cephosd1 is provided below:

```
lang en_US.UTF-8
keyboard --vckeymap=us --xlayouts='us'
timezone --isUtc America/Los_Angeles --ntpserver=173.38.201.115
# System services
services --enabled="chronyd"
rootpw
$6$dA8apVZJJhnc1jrS$IuVqcdAuHQVijluX6S6vw88FYteyog12ZZczrFDRhIROitEIWdI41SjPSsgNgIoVGb3
YanQGm.lyWsK7v48P81 --iscrypted
#platform x86, AMD64, or Intel EM64T
cdrom
reboot
#Network Information
network --bootproto=static --device=eth0 --ip=172.16.32.101 --netmask=255.255.255.0 --
gateway=172.16.32.1 --hostname=cephosd1.sjc02dmz.net --nameserver=192.168.10.51 --
noipv6 --mtu=1500 --onboot=on --activate
network --bootproto=static --device=team1 --ip=172.16.33.101 --netmask=255.255.255.0 -
-noipv6 --mtu=9000 --onboot=on --activate --teamslaves="eth1,eth2" --
teamconfig="{\"runner\": {\"name\": \"loadbalance\"}}"
network --bootproto=static --device=team2 --ip=172.16.34.101 --netmask=255.255.255.0 -
-noipv6 --mtu=9000 --onboot=on --activate --teamslaves="eth3,eth4" --
teamconfig="{\"runner\": {\"name\": \"loadbalance\"}}"
bootloader --location=mbr --append="rhgb quiet crashkernel=auto" --boot-
drive=/dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0
clearpart --all --initlabel
zerombr
# Disk partitioning information
```

```

part pv.1 --fstype="lvmpv" --ondisk=/dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0 --
size=890000
part /boot --fstype="xfs" --ondisk=/dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0 --
size=1024
volgroup ceph --pesize=4096 pv.1
logvol /home --fstype="xfs" --size=10240 --name=home --vgname=ceph
logvol swap --fstype="swap" --size=4096 --name=swap --vgname=ceph
logvol / --fstype="xfs" --size=204800 --name=root --vgname=ceph
logvol /var --fstype="xfs" --grow --size=1 --name=var --vgname=ceph
logvol /tmp --fstype="xfs" --size=40960 --name=tmp --vgname=ceph
auth --passalgo=sha512 --useshadow
selinux --disabled
firewall --disabled
firstboot --disable
ignoredisk --only-use=/dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0

%packages
@^minimal-environment
chrony
kexec-tools
%end

%addon com_redhat_kdump --enable --reserve-mb='auto'

%end

```

To create a custom ISO for RHEL 8.2, follow these steps:

1. Mount the DVD ISO:

```

[root@sjc02dmz-i14-terraform ~]# mount -o loop /tmp/rhel-8.2-x86_64-dvd.iso /mnt
mount: /mnt: WARNING: device write-protected, mounted read-only.

```

2. Create a directory and copy all the content of the ISO:

```

[root@sjc02dmz-i14-terraform ~]# shopt -s dotglob
[root@sjc02dmz-i14-terraform ~]# mkdir /tmp/rhel8
[root@sjc02dmz-i14-terraform ~]# cp -avRf /mnt/* /tmp/rhel8

```

3. Verify that all hidden files, such as .treeinfo, are there in /tmp/rhel8:

```

[root@sjc02dmz-i14-terraform ~]# cd /tmp/rhel8
[root@sjc02dmz-i14-terraform rhel8]# ls -a
.      addons      EFI      extra_files.json  images      LiveOS      Packages  RPM-GPG-KEY-
redhat-beta  TRANS.TBL

```

```
.. .discinfo EULA GPL isolinux media.repo repodata RPM-GPG-KEY-
redhat-release .treeinfo
```

4. Get the kickstart file and copy it to /tmp/rhel8:

```
[root@sjc02dmz-i14-terraform rhel8]# cp /root/ks-cephosd1.cfg /tmp/rhel8/
```

5. Confirm the LABEL of the DVD iso. This provides the LABEL information.

```
[root@sjc02dmz-i14-terraform rhel8]# blkid /tmp/rhel-8.2-x86_64-dvd.iso
/tmp/rhel-8.2-x86_64-dvd.iso: BLOCK_SIZE="2048" UUID="2020-04-04-08-21-15-00" LA-
BEL="RHEL-8-2-0-BaseOS-x86_64" TYPE="iso9660" PTUUID="47055c33" PTTYPE="dos"
```

6. Add the following part in /tmp/rhel8/isolinux/isolinux.cfg file as follows. Make sure that the part has inst.stage2 and the correct label. Remove “menu default” from “label check” and change timeout to 100.

```
label kickstart
    menu label ^Kickstart Installation of RHEL8.2
    kernel vmlinuz
    menu default
    append initrd=initrd.img inst.stage2=hd:LABEL=RHEL-8.2\x20Server.x86_64
    inst.ks=cdrom:/ks-cephosd1.cfg net.ifnames=0 biosdevname=0
```

7. Save the file and create the ISO as follows. Make sure that -V has the correct LABEL, a mistake will cause the DVD not to work.

```
[root@sjc02dmz-i14-terraform rhel8]# mkisofs -o /tmp/rhel8.2-cephosd1.iso -b
isolinux/isolinux.bin -J -R -l -c isolinux/boot.cat -no-emul-boot -boot-load-size 4 -
boot-info-table -eltorito-alt-boot -e images/efiboot.img -no-emul-boot -graft-points -V
"RHEL-8.2 Server.x86_64" .
[root@sjc02dmz-i14-terraform rhel8]# mv ../rhel8.2-cephosd1.iso /var/www/html/images/
```

8. Repeat steps 1-7 for cephosd2-6 and cephadm with the different kickstart file and move them to /var/www/html/images.

Automated Install of RHEL OS with Terraform

After creating policies and profiles, assigning, and deploying profiles, in the last steps we do an automated install of RHEL 8.2 on all Ceph OSD and Ceph Admin nodes. The process contains two actions:

- Create the Software Repository with the OS image for each node for the environment.
- Trigger all nodes to reboot and boot via vMedia from the OS image.

For that you need to specify one configuration files, main.tf. It creates the software repositories in Cisco Intersight and reboots the nodes and installs the OS based on the software repository.

To do an automated OS install with Terraform under Cisco Intersight, follow these steps:

1. Log into the Terraform server and go to subdirectory for OS install:

```
[root@sjc02dmz-i14-terraform ~]# cd terraform-intersight-sds/install_os/
[root@sjc02dmz-i14-terraform install_os]# cp ../create_policy_profile/variables.tf .
```

2. Run terraform plan to see whether your configuration runs through:

```
[root@sjc02dmz-i14-terraform install_os]# terraform plan
... -> We skip the full output as it is very lengthy.
Plan: 14 to add, 0 to change, 0 to destroy.
```

Note: You didn't specify an "-out" parameter to save this plan, so Terraform can't guarantee that exactly these actions will be performed if "terraform apply" is subsequently run.

3. If your configuration is good, run the apply command to deploy the OS:

```
[root@sjc02dmz-i14-terraform install_os]# terraform apply
... -> We skip the full output as it is very lengthy.
Plan: 14 to add, 0 to change, 0 to destroy.
```

Do you want to perform these actions?

Terraform will perform the actions described above.

Only 'yes' will be accepted to approve.

Enter a value: yes

```
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[0]: Creating...
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[3]: Creating...
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[5]: Creating...
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[1]: Creating...
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[4]: Creating...
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[2]: Creating...
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[6]: Creating...
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[5]: Creation complete after 0s [id=5fbf50586567612d308738cc]
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[0]: Creation complete after 1s [id=5fbf50596567612d308738d4]
```

```
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[3]: Creation complete after 1s [id=5fbf50596567612d308738db]
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[4]: Creation complete after 1s [id=5fbf50596567612d308738e2]
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[1]: Creation complete after 2s [id=5fbf505a6567612d308738e9]
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[6]: Creation complete after 2s [id=5fbf505a6567612d308738f0]
intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-
cephosd[2]: Creation complete after 2s [id=5fbf505a6567612d308738f7]
intersight_os_install.cephosd[2]: Creating...
intersight_os_install.cephosd[3]: Creating...
intersight_os_install.cephosd[1]: Creating...
intersight_os_install.cephosd[4]: Creating...
intersight_os_install.cephosd[5]: Creating...
intersight_os_install.cephosd[0]: Creating...
intersight_os_install.cephosd[6]: Creating...
intersight_os_install.cephosd[2]: Creation complete after 2s
[id=5fbf505adc8ad2fa31b57ff7]
intersight_os_install.cephosd[1]: Creation complete after 2s
[id=5fbf505cdc8ad2fa31b58012]
intersight_os_install.cephosd[0]: Creation complete after 3s
[id=5fbf505ddc8ad2fa31b5802b]
intersight_os_install.cephosd[5]: Creation complete after 3s
[id=5fbf505ddc8ad2fa31b58044]
intersight_os_install.cephosd[3]: Creation complete after 4s
[id=5fbf505edc8ad2fa31b5805c]
intersight_os_install.cephosd[6]: Creation complete after 4s
[id=5fbf505edc8ad2fa31b58074]
intersight_os_install.cephosd[4]: Creation complete after 5s
[id=5fbf505fdc8ad2fa31b5808c]
```

Apply complete! Resources: 14 added, 0 changed, 0 destroyed.

The setup of the Cisco UCS environment for the Ceph nodes is now finished and in the next step we're going to prepare the Ceph administration host as well as the Ceph OSD nodes and deploy RH Ceph Storage.

Red Hat Ceph Storage 4 Installation and Configuration

The following procedures document the installation and configuration of Red Hat Ceph Storage 4. The installation process is comprised of six unique stages listed below:

1. Prepare the Ceph administration host and the Ceph OSD nodes.
2. Run base performance tests to set the baseline for the environment.
3. Run the Ansible installer to deploy Red Hat Ceph Storage.
4. Create pools with replication factor 3 and erasure coding 4+2.
5. Run performance tests for block and object configurations.
6. Run high availability tests.

Prerequisites

After setting up the Cisco UCS environment with Cisco Intersight, Terraform and Red Hat Enterprise Linux 8.2, you need to have an active subscription to update all nodes with the latest patches and to install Red Hat Ceph Storage.



Contact your Red Hat sales representative for subscription to Red Hat Ceph Storage.

Further prerequisites for the current installation are:

- Verifying the operating system
- Registering Ceph nodes
- Enabling Ceph software repositories
- Creating an Ansible user
- Enabling password-less SSH

Preparation of Ceph Nodes

To prepare all nodes, follow these steps:

1. Log into the Ceph administration node, register the node and attach the subscription.

```
[root@cephadm ~]# subscription-manager register
Registering to: subscription.rhsm.redhat.com:443/subscription
Username: XXX
Password: YYY
The system has been registered with ID: 7c32c16e-9055-442c-b7e9-bd24f21c88e7
The registered system name is: cephadm.sjc02dmz.net
[root@cephadm ~]# subscription-manager refresh
```

```
All local data refreshed
[root@cephadm ~]# subscription-manager list -available
[root@cephadm ~]# subscription-manager attach --pool=8a85f99b71a877770171bb0e40f66b7a
Successfully attached a subscription for: Red Hat Ceph Storage, Self-Support (8 Nodes,
NFR, Partner Only)
[root@cephadm ~]# subscription-manager repos --disable=*
[root@cephadm ~]# subscription-manager repos --enable=rhel-8-for-x86_64-baseos-rpms --
enable=rhel-8-for-x86_64-appstream-rpms
Repository 'rhel-8-for-x86_64-baseos-rpms' is enabled for this system.
Repository 'rhel-8-for-x86_64-appstream-rpms' is enabled for this system.
[root@cephadm ~]# dnf -y update
```

2. Repeat step 1 for all Ceph nodes.

3. On the Ceph administration node enable the Red Hat Ceph Storage 4 Tools repository and Ansible repository. Because of the containerized deployment there is no need to install repositories like Monitor, OSD, or Ceph Object Gateways.

```
[root@cephadm ~]# subscription-manager repos --enable=rhceph-4-tools-for-rhel-8-x86_64-
rpms --enable=ansible-2.9-for-rhel-8-x86_64-rpms
Repository 'rhceph-4-tools-for-rhel-8-x86_64-rpms' is enabled for this system.
Repository 'ansible-2.9-for-rhel-8-x86_64-rpms' is enabled for this system.
```

4. Enable password-less ssh for root user.

```
[root@cephadm ~]# ssh-keygen
[root@cephadm ~]# for i in {1..6}; do ssh-copy-id root@cephosd$i; done
```

5. Create an Ansible user with sudo access.

```
[root@cephadm ~]# adduser admin
[root@cephadm ~]# passwd admin
Changing password for user admin.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@cephadm ~]# cat << EOF >/etc/sudoers.d/admin
> admin ALL = (root) NOPASSWD:ALL
> EOF
[root@cephadm ~]# chmod 0440 /etc/sudoers.d/admin
[root@cephadm ~]# for i in {1..6}; do ssh -t root@cephosd$i "adduser admin"; done
[root@cephadm ~]# for i in {1..6}; do ssh -t root@cephosd$i "passwd admin"; done
[root@cephadm ~]# for i in {1..6}; do scp /etc/sudoers.d/admin
root@cephosd$i:/etc/sudoers.d/; done
```



```
[root@cephadm ~]# for i in {1..6}; do ssh -t root@cephosd$i "chmod 0440
/etc/sudoers.d/admin"; done
```

6. Enable password-less ssh for user admin.

```
[root@cephadm ~]# su - admin
[admin@cephadm ~]$ ssh-keygen
[admin@cephadm ~]$ for i in {1..6}; do ssh-copy-id admin@cephosd$i; done
[admin@cephadm ~]$ ssh-copy-id admin@cephadm
[admin@cephadm ~]$ touch ~/.ssh/config
[admin@cephadm ~]$ vi ~/.ssh/config
Host node1
    Hostname cephosd1
    User admin
Host node2
    Hostname cephosd2
    User admin
Host node3
    Hostname cephosd3
    User admin
Host node4
    Hostname cephosd4
    User admin
Host node5
    Hostname cephosd5
    User admin
Host node6
    Hostname cephosd6
    User admin
Host node7
    Hostname cephadm
    User admin
[admin@cephadm ~]$ chmod 600 ~/.ssh/config
```

The preparation of all nodes is completed.

Base Performance Testing of the Hardware

Before installing Red Hat Ceph Storage, it is useful to verify the current install in terms of network and base disk performance. Run the two main test procedures to verify each:

- Network performance test with iperf3 to verify the current line speed and the MTU size on both networks' client and storage.

- Disk performance test with fio to verify the maximum performance for a single HDD and both NVMe and an entire node with 12 disks. Block size will be 4KB and 4MB with tests running random read/write and sequential read/write.

Network Verification

To verify that jumbo frames are correctly implemented in the environment and the link speed is equivalent to 25 Gbps, we install iperf3 on all nodes. Before moving on to more complex activities, it can be beneficial to verify the network as operating as intended. This test will also determine if SSH was configured correctly. To test if MTU 9000 is correctly configured, follow these steps.

1. Install iperf3 on all Ceph OSD nodes.

```
[root@cephadm ~]# for i in {1..6}; do ssh -t root@cephosd$i "dnf -y install iperf3
fio"; done
```

2. Start iperf3 in server mode on one Ceph OSD node.

```
[root@cephosd2 ~]# iperf3 -s -p 5002
```

3. The following command `iperf3 -c 172.16.34.101 -P 4 -p 5002 -t 5 -V` will run iperf3 on the storage interface. There are two things this will test: total throughput and MSS size in bytes. The frame size is marked in yellow and the total throughput is marked in red. Repeat this same test on the client interface.

```
[root@cephosd6 ~]# iperf3 -c 172.16.34.101 -P 4 -p 5002 -t 5 -V
iperf 3.5
Linux cephosd6 4.18.0-193.el8.x86_64 #1 SMP Fri Mar 27 14:35:58 UTC 2020 x86_64
Control connection MSS 8948
Time: Fri, 20 Nov 2020 13:32:41 GMT
Connecting to host 172.16.34.101, port 5002
Cookie: eolxtgulcctg55nwekpgqkmmnsegaaokrjceo
TCP MSS: 8948 (default)
[ 5] local 172.16.34.106 port 49742 connected to 172.16.34.101 port 5002
[ 7] local 172.16.34.106 port 49744 connected to 172.16.34.101 port 5002
[ 9] local 172.16.34.106 port 49746 connected to 172.16.34.101 port 5002
[11] local 172.16.34.106 port 49748 connected to 172.16.34.101 port 5002
Starting Test: protocol: TCP, 4 streams, 131072 byte blocks, omitting 0 seconds, 5 second test, tos 0
[ ID] Interval          Transfer          Bitrate          Retr  Cwnd
[ 5]  0.00-1.00    sec    985 MBytes    8.26 Gbits/sec     0   577 KBytes
[ 7]  0.00-1.00    sec    984 MBytes    8.26 Gbits/sec     0   577 KBytes
[ 9]  0.00-1.00    sec    493 MBytes    4.13 Gbits/sec     0   498 KBytes
[11]  0.00-1.00    sec    493 MBytes    4.14 Gbits/sec     0   498 KBytes
[SUM]  0.00-1.00    sec    2.89 GBytes    24.8 Gbits/sec     0
- - - - -
[ 5]  1.00-2.00    sec    983 MBytes    8.25 Gbits/sec     0   603 KBytes
```

```

[ 7] 1.00-2.00 sec 983 MBytes 8.25 Gbits/sec 0 603 KBytes
[ 9] 1.00-2.00 sec 492 MBytes 4.13 Gbits/sec 0 542 KBytes
[11] 1.00-2.00 sec 492 MBytes 4.12 Gbits/sec 0 498 KBytes
[SUM] 1.00-2.00 sec 2.88 GBytes 24.7 Gbits/sec 0

```

```

-----
[ 5] 2.00-3.00 sec 983 MBytes 8.25 Gbits/sec 0 603 KBytes
[ 7] 2.00-3.00 sec 983 MBytes 8.25 Gbits/sec 0 603 KBytes
[ 9] 2.00-3.00 sec 491 MBytes 4.12 Gbits/sec 0 542 KBytes
[11] 2.00-3.00 sec 491 MBytes 4.12 Gbits/sec 0 524 KBytes
[SUM] 2.00-3.00 sec 2.88 GBytes 24.7 Gbits/sec 0

```

```

-----
[ 5] 3.00-4.00 sec 983 MBytes 8.25 Gbits/sec 0 603 KBytes
[ 7] 3.00-4.00 sec 983 MBytes 8.25 Gbits/sec 0 603 KBytes
[ 9] 3.00-4.00 sec 492 MBytes 4.13 Gbits/sec 0 542 KBytes
[11] 3.00-4.00 sec 492 MBytes 4.13 Gbits/sec 0 524 KBytes
[SUM] 3.00-4.00 sec 2.88 GBytes 24.7 Gbits/sec 0

```

```

-----
[ 5] 4.00-5.00 sec 983 MBytes 8.24 Gbits/sec 0 603 KBytes
[ 7] 4.00-5.00 sec 983 MBytes 8.25 Gbits/sec 0 603 KBytes
[ 9] 4.00-5.00 sec 492 MBytes 4.13 Gbits/sec 0 542 KBytes
[11] 4.00-5.00 sec 492 MBytes 4.12 Gbits/sec 0 524 KBytes
[SUM] 4.00-5.00 sec 2.88 GBytes 24.7 Gbits/sec 0

```

Test Complete. Summary Results:

[ID]	Interval		Transfer	Bitrate	Retr	
[5]	0.00-5.00	sec	4.80 GBytes	8.25 Gbits/sec	0	sender
[5]	0.00-5.04	sec	4.80 GBytes	8.19 Gbits/sec		receiver
[7]	0.00-5.00	sec	4.80 GBytes	8.25 Gbits/sec	0	sender
[7]	0.00-5.04	sec	4.80 GBytes	8.19 Gbits/sec		receiver
[9]	0.00-5.00	sec	2.40 GBytes	4.13 Gbits/sec	0	sender
[9]	0.00-5.04	sec	2.40 GBytes	4.09 Gbits/sec		receiver
[11]	0.00-5.00	sec	2.40 GBytes	4.13 Gbits/sec	0	sender
[11]	0.00-5.04	sec	2.40 GBytes	4.10 Gbits/sec		receiver
[SUM]	0.00-5.00	sec	14.4 GBytes	24.7 Gbits/sec	0	sender
[SUM]	0.00-5.04	sec	14.4 GBytes	24.6 Gbits/sec		receiver

CPU Utilization: local/sender 52.1% (2.9%u/49.2% s), remote/receiver 10.4% (0.7%u/9.7% s)

snd_tcp_congestion cubic

rcv_tcp_congestion cubic

```

iperf Done.
[root@cephosd6 ~]# iperf3 -c 172.16.33.101 -P 4 -p 5002 -t 5 -V
iperf 3.5
Linux cephosd6 4.18.0-193.el8.x86_64 #1 SMP Fri Mar 27 14:35:58 UTC 2020 x86_64
Control connection MSS 8948
Time: Fri, 20 Nov 2020 13:34:43 GMT
Connecting to host 172.16.33.101, port 5002
    Cookie: tuexit
6isc5ni45cx6gfr2y6xfjyqdsjdf223c4i
    TCP MSS: 8948 (default)
[ 5] local 172.16.33.106 port 55078 connected to 172.16.33.101 port 5002
[ 7] local 172.16.33.106 port 55080 connected to 172.16.33.101 port 5002
[ 9] local 172.16.33.106 port 55082 connected to 172.16.33.101 port 5002
[11] local 172.16.33.106 port 55084 connected to 172.16.33.101 port 5002
Starting Test: protocol: TCP, 4 streams, 131072 byte blocks, omitting 0 seconds, 5 second test, tos 0
[ ID] Interval          Transfer      Bitrate      Retr  Cwnd
[ 5]  0.00-1.00    sec   985 MBytes  8.26 Gbits/sec    0   612 KBytes
[ 7]  0.00-1.00    sec   495 MBytes  4.15 Gbits/sec    0   594 KBytes
[ 9]  0.00-1.00    sec   985 MBytes  8.26 Gbits/sec    0   664 KBytes
[11]  0.00-1.00    sec   492 MBytes  4.13 Gbits/sec    0   594 KBytes
[SUM]  0.00-1.00    sec   2.89 GBytes 24.8 Gbits/sec    0
- - - - -
[ 5]  1.00-2.00    sec   983 MBytes  8.25 Gbits/sec    0   612 KBytes
[ 7]  1.00-2.00    sec   491 MBytes  4.12 Gbits/sec    0   594 KBytes
[ 9]  1.00-2.00    sec   983 MBytes  8.24 Gbits/sec    0   699 KBytes
[11]  1.00-2.00    sec   491 MBytes  4.12 Gbits/sec    0   594 KBytes
[SUM]  1.00-2.00    sec   2.88 GBytes 24.7 Gbits/sec    0
- - - - -
[ 5]  2.00-3.00    sec   983 MBytes  8.24 Gbits/sec    0   638 KBytes
[ 7]  2.00-3.00    sec   492 MBytes  4.13 Gbits/sec    0   594 KBytes
[ 9]  2.00-3.00    sec   983 MBytes  8.25 Gbits/sec    0   699 KBytes
[11]  2.00-3.00    sec   491 MBytes  4.12 Gbits/sec    0   594 KBytes
[SUM]  2.00-3.00    sec   2.88 GBytes 24.7 Gbits/sec    0
- - - - -
[ 5]  3.00-4.00    sec   984 MBytes  8.25 Gbits/sec    0   638 KBytes
[ 7]  3.00-4.00    sec   492 MBytes  4.12 Gbits/sec    0   594 KBytes
[ 9]  3.00-4.00    sec   983 MBytes  8.25 Gbits/sec    0   699 KBytes
[11]  3.00-4.00    sec   491 MBytes  4.12 Gbits/sec    0   594 KBytes
[SUM]  3.00-4.00    sec   2.88 GBytes 24.7 Gbits/sec    0

```

```

- - - - -
[ 5]  4.00-5.00  sec   982 MBytes  8.24 Gbits/sec   0    638 KBytes
[ 7]  4.00-5.00  sec   491 MBytes  4.12 Gbits/sec   0    594 KBytes
[ 9]  4.00-5.00  sec   982 MBytes  8.24 Gbits/sec   0    699 KBytes
[11]  4.00-5.00  sec   491 MBytes  4.12 Gbits/sec   0    594 KBytes
[SUM] 4.00-5.00  sec   2.88 GBytes 24.7 Gbits/sec   0
- - - - -

```

Test Complete. Summary Results:

[ID]	Interval		Transfer	Bitrate	Retr	
[5]	0.00-5.00	sec	4.80 GBytes	8.25 Gbits/sec	0	sender
[5]	0.00-5.04	sec	4.80 GBytes	8.19 Gbits/sec		receiver
[7]	0.00-5.00	sec	2.40 GBytes	4.13 Gbits/sec	0	sender
[7]	0.00-5.04	sec	2.40 GBytes	4.10 Gbits/sec		receiver
[9]	0.00-5.00	sec	4.80 GBytes	8.25 Gbits/sec	0	sender
[9]	0.00-5.04	sec	4.80 GBytes	8.19 Gbits/sec		receiver
[11]	0.00-5.00	sec	2.40 GBytes	4.12 Gbits/sec	0	sender
[11]	0.00-5.04	sec	2.40 GBytes	4.09 Gbits/sec		receiver
[SUM]	0.00-5.00	sec	14.4 GBytes	24.7 Gbits/sec	0	sender
[SUM]	0.00-5.04	sec	14.4 GBytes	24.7 Gbits/sec		receiver

CPU Utilization: local/sender 52.6% (2.5%/50.1%), remote/receiver 9.0% (0.5%/8.5%)

snd_tcp_congestion cubic

rcv_tcp_congestion cubic

iperf Done.

4. Perform this test on an all Ceph OSD nodes as desired to confirm that jumbo frames are enabled on all servers and bandwidth is ~25 Gb/s

Base HDD/NVMe Performance

The following test with fio gives a base understanding what the maximum performance of each device is. It helps to get a better understanding later for Ceph performance tests with block or object devices.

We ran the tests with fio for a single HDD, both single NVMe and all 12 HDDs. Before testing the base performance, make sure the drive cache is enabled under RHEL by running the following on each Ceph OSD node:

```

[root@cephosd6 ~]# for i in {a..l}; do sdparm --get=WCE /dev/sd$i;done
/dev/sda: HGST          HUH721010AL42C0   A3Z4
WCE          1 [cha: y, def:  0, sav:  1]
/dev/sdb: HGST          HUH721010AL42C0   A3Z4
WCE          1 [cha: y, def:  0, sav:  1]
/dev/sdc: HGST          HUH721010AL42C0   A3Z4
WCE          1 [cha: y, def:  0, sav:  1]

```

```

/dev/sdd: HGST      HUH721010AL42C0  A3Z4
WCE          1 [cha: y, def: 0, sav: 1]
/dev/sde: HGST      HUH721010AL42C0  A3Z4
WCE          1 [cha: y, def: 0, sav: 1]
/dev/sdf: HGST      HUH721010AL42C0  A3Z4
WCE          1 [cha: y, def: 0, sav: 1]
/dev/sdg: HGST      HUH721010AL42C0  A3Z4
WCE          1 [cha: y, def: 0, sav: 1]
/dev/sdh: HGST      HUH721010AL42C0  A3Z4
WCE          1 [cha: y, def: 0, sav: 1]
/dev/sdi: HGST      HUH721010AL42C0  A3Z4
WCE          1 [cha: y, def: 0, sav: 1]
/dev/sdj: HGST      HUH721010AL42C0  A3Z4
WCE          1 [cha: y, def: 0, sav: 1]
/dev/sdk: HGST      HUH721010AL42C0  A3Z4
WCE          1 [cha: y, def: 0, sav: 1]
/dev/sdl: HGST      HUH721010AL42C0  A3Z4
WCE          1 [cha: y, def: 0, sav: 1]
```



sdparm can be installed via epel-release. Make sure you remove the epel repository after installing sdparm.

1. WCE 1 informs you that the drive cache is enabled. If WCE is 0 please run the following command:

```
[root@cephosd6 ~]# for i in {a..l}; do sdparm --set=WCE --save /dev/sd$i;done
/dev/sda: HGST      HUH721010AL42C0  A3Z4
/dev/sdb: HGST      HUH721010AL42C0  A3Z4
/dev/sdc: HGST      HUH721010AL42C0  A3Z4
/dev/sdd: HGST      HUH721010AL42C0  A3Z4
/dev/sde: HGST      HUH721010AL42C0  A3Z4
/dev/sdf: HGST      HUH721010AL42C0  A3Z4
/dev/sdg: HGST      HUH721010AL42C0  A3Z4
/dev/sdh: HGST      HUH721010AL42C0  A3Z4
/dev/sdi: HGST      HUH721010AL42C0  A3Z4
/dev/sdj: HGST      HUH721010AL42C0  A3Z4
/dev/sdk: HGST      HUH721010AL42C0  A3Z4
/dev/sdl: HGST      HUH721010AL42C0  A3Z4
```

2. To run a fio test on a single disk, run the following:

```
root@cephosd6:~# fio --filename=/dev/sda --name=read-4k --rw=read --ioengine=libaio --
bs=4k --numjobs=1 --direct=1 --randrepeat=0 --iodepth=16 --runtime=300 --ramp_time=5 -
-size=100G --group_reporting
```

3. Delete the cache of the system by running after each fio test:

```
root@cephosd6:~# sync; echo 3 > /proc/sys/vm/drop_caches
```

4. Run the tests with 4k and 4m block sizes and random read/write and sequential read/write. After that run the same tests with all 12 disks:

```
root@cephosd6:~#
disk_list=/dev/sda,/dev/sdb,/dev/sdc,/dev/sdd,/dev/sde,/dev/sdf,/dev/sdg,/dev/sdh,/dev/
sdi,/dev/sdj,/dev/sdk,/dev/sdl
root@cephosd6:~# genfio -d $disk_list -b 4k -D 16 -r 300 -p -m randread -x read4k.fio
root@cephosd6:~# fio read4k.fiocephosd4-4k-parallel-randread-
sda,sdb,sdc,sdd,sde,sdf,sdg,sdh,sdi,sdj,sdk,sdl.fio
```

A summary of the base performance in MB/s is listed in [Table 12](#).

Table 12. Base Performance Values in MB/s for HDD and NVMe for a Single Ceph OSD Node

Workload	Single Disk		12 Disks		1.6 TB NVMe		3.2 TB NVMe	
	4 KB	4 MB	4 KB	4 MB	4 KB	4 MB	4 KB	4 MB
Random Read MB/s	1.6	214	7.9	1790	744	6417	1275	6172
Random Write MB/s	2.6	205	21.1	1657	1320	2333	1327	2407
Sequential Read MB/s	253	259	3044	3045	745	6425	927	6444
Sequential Write MB/s	259	258	3045	3043	1380	2340	1419	2456

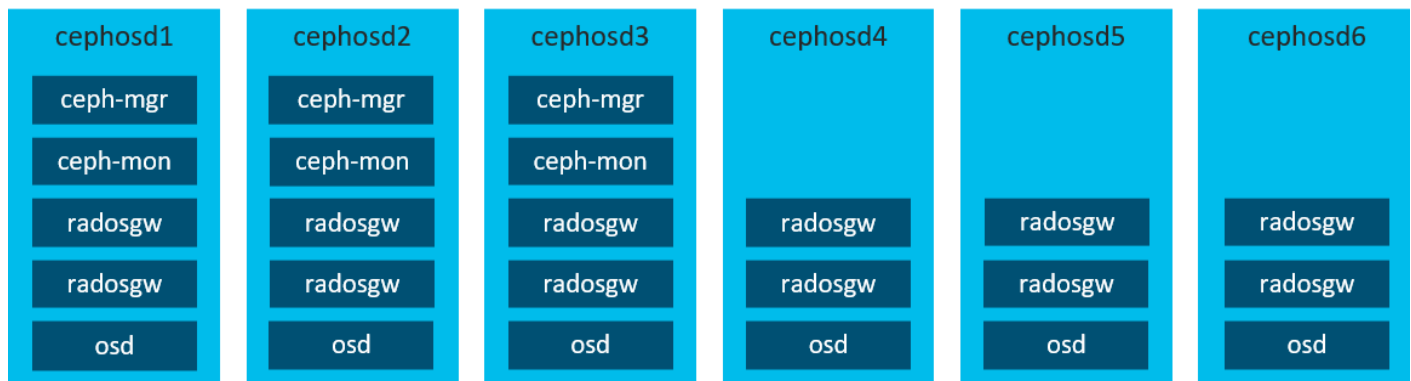
The base performance testing is done now, and we can move over to start building the Red Hat Ceph Storage cluster.

Red Hat Ceph Storage Installation and Configuration

This chapter describes the usage of Ansible to deploy a Red Hat Ceph Storage cluster and other components such as Ceph Monitor and Ceph Object Gateway. We deploy the cluster with collocation of containerized Ceph daemons. Collocation helps improving the total cost of ownership, having easier upgrades and a better resource isolation.

Together with the OSD daemons, deploy the Ceph Monitor, Ceph Manager, and Ceph Object Gateway on the Ceph OSD nodes. [Figure 36](#) shows the collocation of the tested environment.

Figure 36. Collocation of Containerized Ceph Daemons



Red Hat recommends collocating the Ceph Object Gateway with OSD containers to increase performance as well as reducing the CAPEX. To achieve the highest performance without incurring an additional cost, use two gateways by setting `radosgw_num_instances: 2` in `group_vars/all.yml`.

The Grafana and Prometheus daemons will run on the Ceph administration node.

To configure and install Red Hat Ceph Storage 4 with Ansible, follow these steps:

1. Login as root to the Ceph administration node and install the `ceph-ansible` package.

```
[root@cephadm ~]# dnf -y install ceph-ansible
```

2. Go to `/usr/share/ceph-ansible` and create new yml files.

```
[root@cephadm ~]# cd /usr/share/ceph-ansible/
[root@cephadm ceph-ansible]# cp group_vars/all.yml.sample group_vars/all.yml
[root@cephadm ceph-ansible]# cp group_vars/osds.yml.sample group_vars/osds.yml
[root@cephadm ceph-ansible]# cp site-container.yml.sample site-container.yml
```

3. Edit the `all.yml` per your needs. We changed the following variables:

```
dummy:
fetch_directory: ~/ceph-ansible-keys
configure_firewall: false
ceph_repository_type: cdn
```



```
ceph_origin: repository
ceph_repository: rhcs
ceph_rhcs_version: 4
ceph_iscsi_config_dev: false
monitor_interface: team1
public_network: 172.16.33.0/24
cluster_network: 172.16.34.0/24
osd_objectstore: bluestore
radosgw_civetweb_port: 8080
radosgw_civetweb_num_threads: 1024
radosgw_thread_pool_size: 1024
radosgw_interface: team1
radosgw_num_instances: 2
ceph_conf_overrides:
  global:
    mon_allow_pool_delete: true
    mon_max_pg_per_osd: 1000
    ms_dispatch_throttle_bytes: 1048576000
    objecter_inflight_op_bytes: 1048576000
    objecter_inflight_ops: 5120
    osd_enable_op_tracker: False
    max_open_files: 500000
    bluefs_buffered_io: false
    rgw_list_buckets_max_chunk: 999999
    osd_op_thread_timeout: 900
    osd_op_thread_suicide_timeout: 2000
    osd_max_write_size: 500
ceph_docker_image: "rhceph/rhceph-4-rhel8"
ceph_docker_image_tag: "latest"
ceph_docker_registry: "registry.redhat.io"
ceph_docker_registry_auth: true
ceph_docker_registry_username: "XXX"
ceph_docker_registry_password: "YYY"
containerized_deployment: true
dashboard_enabled: True
dashboard_protocol: http
dashboard_port: 8443
dashboard_admin_user: admin
dashboard_admin_password: XXX
```

```
node_exporter_container_image: registry.redhat.io/openshift4/ose-prometheus-node-exporter:v4.1
node_exporter_port: 9100
grafana_admin_user: admin
grafana_admin_password: XXX
grafana_container_image: registry.redhat.io/rhceph/rhceph-4-dashboard-rhel8:4
grafana_dashboard_files:
  - ceph-cluster.json
  - host-details.json
  - hosts-overview.json
  - osd-device-details.json
  - osds-overview.json
  - pool-detail.json
  - pool-overview.json
  - radosgw-detail.json
  - radosgw-overview.json
  - rbd-overview.json
grafana_plugins:
  - vonage-status-panel
  - grafana-piechart-panel
grafana_allow_embedding: True
grafana_port: 3000
prometheus_container_image: registry.redhat.io/openshift4/ose-prometheus:4.1
prometheus_port: 9092
alertmanager_container_image: registry.redhat.io/openshift4/ose-prometheus-alertmanager:4.1
alertmanager_port: 9093
```



If you do not have a Red Hat Registry Service Account, create one using the [Registry Service Account webpage](#).

4. Change osds.yml per your needs. We changed the following lines:

```
copy_admin_key: true
devices:
  - /dev/sda
  - /dev/sdb
  - /dev/sdc
  - /dev/sdd
  - /dev/sde
  - /dev/sdf
```

```
- /dev/sdg
- /dev/sdh
- /dev/sdi
- /dev/sdj
- /dev/sdk
- /dev/sdl
bluestore_wal_devices:
- /dev/nvme0n1
- /dev/nvme1n1
```

5. Create an Ansible inventory file.

```
[root@cephadm ~]# cd /usr/share/ceph-ansible/
[root@cephadm ceph-ansible]# touch hosts
[root@cephadm ceph-ansible]# vi hosts
[grafana-server]
cephadm

[mons]
cephosd[1:3]

[mgrs]
cephosd[1:3]

[osds]
cephosd[1:6]

[rgws]
cephosd[1:6]
```

6. Switch to the Ansible user, create a directory where Ansible stores temporary values generated by the ceph-ansible playbook.

```
[root@cephadm ceph-ansible]# su - admin
Last login: Fri Nov 20 05:05:40 PST 2020 on pts/1
[admin@cephadm ~]$ mkdir ~/ceph-ansible-keys
[admin@cephadm ~]$ cd /usr/share/ceph-ansible/
```

7. Verify that Ansible can reach the Ceph nodes.

```
[admin@cephadm ceph-ansible]$ ansible all -m ping -i hosts
[WARNING]: Invalid characters were found in group names but not replaced, use -vvvv to see details
```

```

cephosd3 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
cephosd2 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
cephosd4 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
cephosd1 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
cephosd6 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
cephosd5 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
cephadm | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": false,
    "ping": "pong"
}

```

8. Run the ceph-ansible playbook to do a container deployment.

```
[admin@cephadm ceph-ansible]$ ansible-playbook site-container.yml -i hosts
```

9. After ~20 minutes the cluster is deployed and the final Ansible messages will look like the following.

```

TASK [show ceph status for cluster ceph]
*****
Tuesday 24 November 2020  01:06:37 -0800 (0:00:01.063)          0:19:39.997 *****
ok: [cephosd1 -> cephosd1] =>

```

```

msg:
- ' cluster:'
- '   id:      961f5cdb-3a8f-4711-897e-c6809f21000a'
- '   health: HEALTH_WARN'
- '           too few PGs per OSD (5 < min 30)'
- ' '
- ' services:'
- '   mon: 3 daemons, quorum cephosd1,cephosd2,cephosd3 (age 14m)'
- '   mgr: cephosd1(active, since 41s), standbys: cephosd3, cephosd2'
- '   osd: 72 osds: 72 up (since 5m), 72 in (since 5m)'
- '   rgw: 12 daemons active (cephosd1.rgw0, cephosd1.rgw1, cephosd2.rgw0, ceph-
cephosd2.rgw1, cephosd3.rgw0, cephosd3.rgw1, cephosd4.rgw0, cephosd4.rgw1, cephosd5.rgw0,
cephosd5.rgw1, cephosd6.rgw0, cephosd6.rgw1)'
- ' '
- ' task status:'
- ' '
- ' data:'
- '   pools:  4 pools, 128 pgs'
- '   objects: 240 objects, 17 KiB'
- '   usage:   73 GiB used, 655 TiB / 655 TiB avail'
- '   pgs:     128 active+clean'
- ' '
- ' io:'
- '   client:  2.2 KiB/s rd, 170 B/s wr, 2 op/s rd, 0 op/s wr'
- ' '

```

PLAY RECAP

```

*****
*****

```

cephadm		: ok=145	changed=23	unreachable=0	failed=0
skipped=242	rescued=0	ignored=0			
cephosd1		: ok=530	changed=40	unreachable=0	failed=0
skipped=470	rescued=0	ignored=0			
cephosd2		: ok=444	changed=32	unreachable=0	failed=0
skipped=416	rescued=0	ignored=0			
cephosd3		: ok=448	changed=28	unreachable=0	failed=0
skipped=415	rescued=0	ignored=0			
cephosd4		: ok=240	changed=18	unreachable=0	failed=0
skipped=297	rescued=0	ignored=0			
cephosd5		: ok=240	changed=18	unreachable=0	failed=0
skipped=297	rescued=0	ignored=0			

```
cephosd6          : ok=243  changed=18  unreachable=0  failed=0
skipped=297  rescued=0  ignored=0
```

INSTALLER STATUS

```
*****
*****
```

```
Install Ceph Monitor      : Complete (0:02:43)
Install Ceph Manager      : Complete (0:01:18)
Install Ceph OSD          : Complete (0:06:47)
Install Ceph RGW          : Complete (0:01:48)
Install Ceph Dashboard    : Complete (0:01:34)
Install Ceph Grafana      : Complete (0:00:52)
Install Ceph Node Exporter : Complete (0:01:17)
```

```
Tuesday 24 November 2020 01:06:37 -0800 (0:00:00.045) 0:19:40.043 *****
```

```
=====
```

```
ceph-osd : use ceph-volume lvm batch to create bluestore osds -----
----- 249.46s
```

```
ceph-handler : restart ceph mon daemon(s) -----
----- 70.49s
```

```
ceph-osd : systemd start osd -----
----- 27.12s
```

```
gather and delegate facts -----
----- 20.20s
```

```
ceph-dashboard : get radosgw system user -----
----- 18.09s
```

```
ceph-config : look up for ceph-volume rejected devices -----
----- 15.60s
```

```
ceph-prometheus : start prometheus services -----
----- 15.30s
```

```
ceph-config : look up for ceph-volume rejected devices -----
----- 14.60s
```

```
ceph-config : look up for ceph-volume rejected devices -----
----- 14.25s
```

```
ceph-config : look up for ceph-volume rejected devices -----
----- 13.94s
```

```
ceph-osd : wait for all osd to be up -----
----- 12.60s
```

```
ceph-config : run 'ceph-volume lvm batch --report' to see how many osds are to be cre-
ated ----- 11.18s
```

```
ceph-node-exporter : start the node_exporter service -----
----- 10.75s
```

```

ceph-dashboard : set or update dashboard admin username and password -----
----- 8.82s

ceph-config : run 'ceph-volume lvm batch --report' to see how many osds are to be cre-
ated ----- 8.59s

ceph-config : run 'ceph-volume lvm batch --report' to see how many osds are to be cre-
ated ----- 8.39s

ceph-config : run 'ceph-volume lvm batch --report' to see how many osds are to be cre-
ated ----- 7.82s

ceph-handler : restart ceph mgr daemon(s) -----
----- 7.09s

ceph-handler : restart ceph mgr daemon(s) -----
----- 6.91s

ceph-config : run 'ceph-volume lvm list' to see how many osds have already been created
----- 6.69s

```

10. Run the following command to check the status of the cluster.

```

[root@cephadm ~]# ssh -t root@cephosd1 "podman exec ceph-mon-cephosd1 ceph status"
cluster:
  id:          961f5cdb-3a8f-4711-897e-c6809f21000a
  health: HEALTH_WARN
             too few PGs per OSD (5 < min 30)

services:
  mon: 3 daemons, quorum cephosd1,cephosd2,cephosd3 (age 18m)
  mgr: cephosd1(active, since 3m), standbys: cephosd2, cephosd3
  osd: 72 osds: 72 up (since 9m), 72 in (since 9m)
  rgw: 12 daemons active (cephosd1.rgw0, cephosd1.rgw1, cephosd2.rgw0, cephosd2.rgw1,
cephosd3.rgw0, cephosd3.rgw1, cephosd4.rgw0, cephosd4.rgw1, cephosd5.rgw0, ceph-
osd5.rgw1, cephosd6.rgw0, cephosd6.rgw1)

task status:

data:
  pools:   4 pools, 128 pgs
  objects: 240 objects, 17 KiB
  usage:   73 GiB used, 655 TiB / 655 TiB avail
  pgs:    128 active+clean

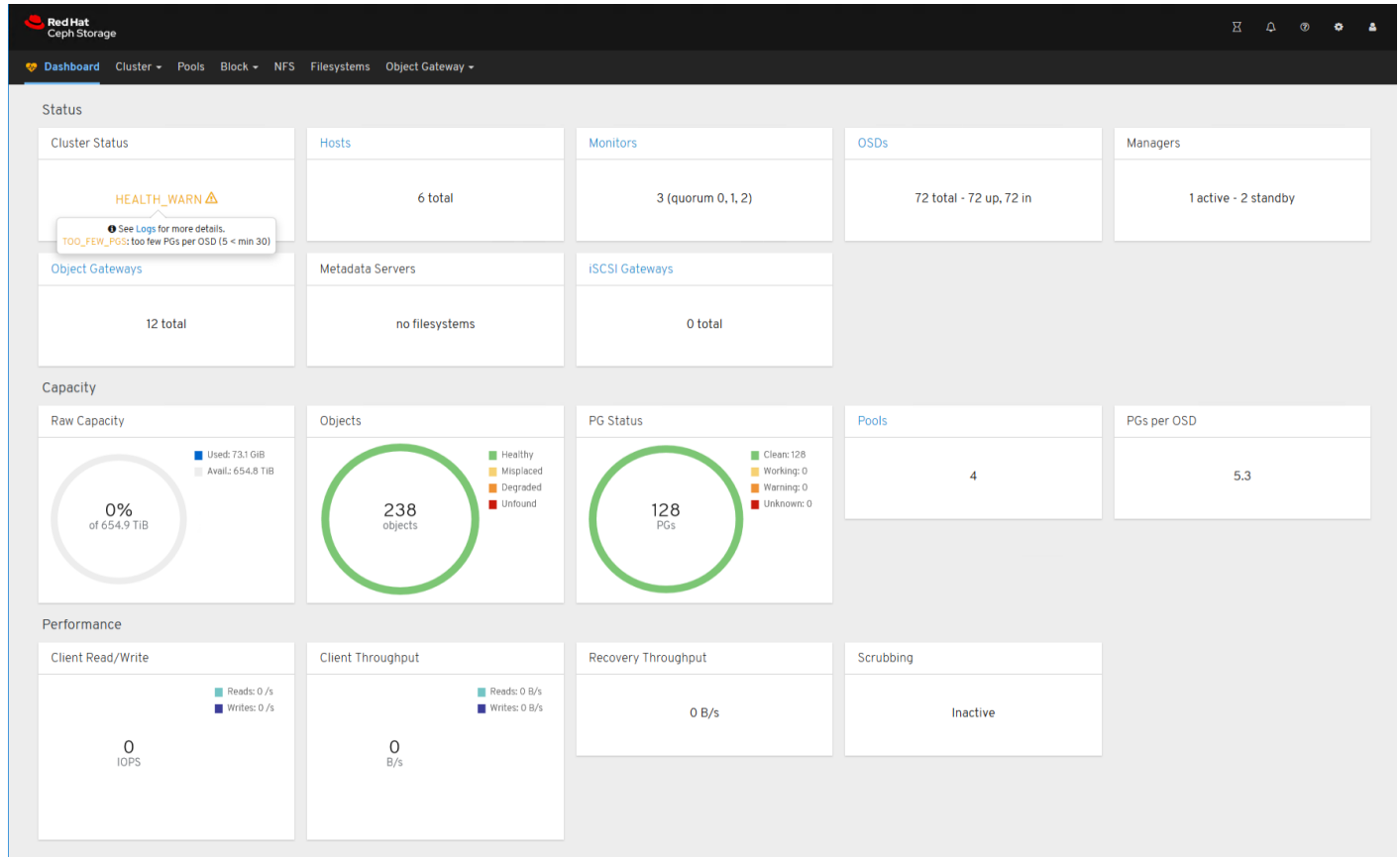
Connection to cephosd1 closed.

```

11. Connect to the Ceph Dashboard. You'll find the information at the very end of the Ansible output when creating the cluster.

```
Tuesday 24 November 2020 01:06:36 -0800 (0:00:01.063)      0:15:05.290 *****
ok: [cephosd1] =>
msg: The dashboard has been deployed! You can access your dashboard web UI at
http://cephosd1.sjc02dmz.net:8443/ as an 'admin' user with 'XXX' password.
```

Figure 37. Ceph Dashboard



The HEATH_WARN message for the Cluster Status results from the low Placement Group (PG) count immediately after deploying the cluster. In the following section explains a way of calculating the correct number of PG count per pool and the warning message will disappear.

Verification of Installation

Block Pool Access

To block pool access, follow these steps:

1. Create a pool for the block data. To determine the correct number of placement groups per OSD, please go to <https://access.redhat.com/labs/cephpgc/> and enter the needed information:
 - a. Ceph version RHCS 3.0+
 - b. Rados Block device

- c. Default configuration
- d. Pool Type Replicated
- e. Size 3
- f. OSD 72
- g. Target PGs per OSD 100

Figure 38. Ceph Placement Groups per Pool Calculator

Ceph Placement Groups (PGs) per Pool Calculator

Ceph Placement Groups (PGs) per Pool Calculator

- Select the Ceph version

Luminous and later -- RHCS 3.0+
- Select a Ceph use case

Rados Block device
- Select special conditions (optional)

Support Erasure Coding (EC)
Supported only for the RGW and Rados Block Device storage.
- Adjust values for pools and PGs

Assuming default configuration

Pool Type * ▼
Replicated

Size *

OSD # *

Target PGs per OSD *

Pool Name ?	Pool Type ?	Size ?	OSD # ?	%Data ?	Target PGs per OSD ?	Suggested PG Count
rbid-cisco	Replicate	3	72	100.00	100	2048

[+ Add Backing Storage Pool](#)
- Download pool creation commands

[Download Commands](#)

- You can either configure the pool manually via CLI by pushing the “Download Commands” button and enter the commands on one of the Ceph Monitor nodes or configure it through the Ceph Dashboard. We configured it through the Dashboard. Go to Dashboard > Pools and click Create. Enter the following information and click CreatePool.
 - Name = <your pool name>
 - Pool type = replicated

- c. Placement groups = 2048
- d. Replicated size = 3
- e. Applications = rbd

Figure 39. Create Pool under Ceph Dashboard

CreatePool

Name *

Pool type *

Placement groups *

[Calculation help](#)

The current PGs settings were calculated for you, you should make sure the values suit your needs before submit.

Crush ruleset

Replicated size *

Applications rbd

Compression

Mode

RBD Configuration

Quality of Service +

CreatePool
Cancel

3. Map a block device to the Ceph administration node to verify the access to the cluster and create a file.

```
[root@cephadm ~]# rbd create disk01 --size 102400 -p rbd-cisco
[root@cephadm ~]# rbd ls -l -p rbd-cisco
NAME    SIZE    PARENT  FMT  PROT  LOCK
disk01  100 GiB          2
[root@cephadm ~]# modprobe rbd
[root@cephadm ~]# rbd feature disable disk01 exclusive-lock object-map fast-diff deep-
flatten -p rbd-cisco
[root@cephadm ~]# rbd map disk01 -p rbd-cisco
/dev/rbd0
[root@cephadm ~]# rbd showmapped
```

```

id pool      namespace image snap device
0 rbd-cisco  disk01 - /dev/rbd0
[root@cephadm ~]# mkfs.xfs /dev/rbd0
meta-data=/dev/rbd0          isize=512    agcount=16, agsize=1638400 blks
           =                  sectsz=512   attr=2, projid32bit=1
           =                  crc=1       finobt=1, sparse=1, rmapbt=0
           =                  reflink=1
data       =                  bsize=4096 blocks=26214400, imaxpct=25
           =                  sunit=16     swidth=16 blks
naming     =version 2         bsize=4096 ascii-ci=0, ftype=1
log        =internal log     bsize=4096 blocks=12800, version=2
           =                  sectsz=512   sunit=16 blks, lazy-count=1
realtime   =none             extsz=4096  blocks=0, rtextents=0
Discarding blocks...Done.
[root@cephadm ~]# mkdir -p /mnt/mydisk
[root@cephadm ~]# mount /dev/rbd0 /mnt/mydisk
[root@cephadm ~]# cd /mnt/mydisk/
[root@cephadm mydisk]# truncate -s 5G cisco.txt
[root@cephadm mydisk]# ll
total 0
-rw-r--r-- 1 root root 5368709120 Dec 16 04:07 cisco.txt

```

Object Pool Access

To create object pool access, follow these steps:

1. Create another pool for erasure coded storing of data via CLI, which is going to be used for the Rados Gateway configuration. In the first step, define the Erasure Code profile 4+2 before creating a pool.

```

[root@cephadm ~]# ceph osd erasure-code-profile set ec-profile-4-2 crush-failure-
domain=host k=4 m=2
[root@cephadm ~]# ceph osd erasure-code-profile ls
default
ec-profile-4-2
[root@cephadm ~]# ceph osd erasure-code-profile get ec-profile-4-2
crush-device-class=
crush-failure-domain=host
crush-root=default
jerasure-per-chunk-alignment=false
k=4
m=2
plugin=jerasure

```

```

technique=reed_sol_van
w=8
[root@cephadm ~]# ceph osd pool create cosbench 1024 1024 erasure ec-profile-4-2
pool 'cosbench' created
[root@cephadm ~]# ceph df
RAW STORAGE:
      CLASS      SIZE      AVAIL      USED      RAW USED      %RAW USED
      TOTAL      655 TiB    655 TiB    6.9 GiB     79 GiB         0.01

POOLS:
      POOL              ID  STORED  OBJECTS  USED      %USED  MAX
AVAIL
      .rgw.root          1    12 KiB    22      4.1 MiB    0      207
TiB
      default.rgw.control 2         0 B      8         0 B      0      207
TiB
      default.rgw.meta    3    393 B     2       384 KiB    0      207
TiB
      default.rgw.log     4    3.4 KiB  206      6 MiB     0      207
TiB
      cosbench           85         0 B     0         0 B      0      415
TiB

```

2. Create a S3 user for the Rados Gateway. Take note of the access and secret key (marked yellow).

```

[root@cephadm ~]# radosgw-admin user create --uid=rgw-user --display-name=RGW-user --
system
{
  "user_id": "rgw-user",
  "display_name": "RGW-user",
  "email": "",
  "suspended": 0,
  "max_buckets": 1000,
  "subusers": [],
  "keys": [
    {
      "user": "rgw-user",
      "access_key": "CSH45H9Y15FU1HF7MHZS",
      "secret_key": "G4y0pWi13LKR1tHJzi9hxcvCoi0du71jSuMiR6Ac"
    }
  ],
  "swift_keys": [],

```

```

    "caps": [],
    "op_mask": "read, write, delete",
    "system": "true",
    "default_placement": "",
    "default_storage_class": "",
    "placement_tags": [],
    "bucket_quota": {
        "enabled": false,
        "check_on_raw": false,
        "max_size": -1,
        "max_size_kb": 0,
        "max_objects": -1
    },
    "user_quota": {
        "enabled": false,
        "check_on_raw": false,
        "max_size": -1,
        "max_size_kb": 0,
        "max_objects": -1
    },
    "temp_url_keys": [],
    "type": "rgw",
    "mfa_ids": []
}

```

3. Create a new storage policy for the new created pool to make sure that data from the above user gets stored in the right pool. Simplify that task by using the pool “cosbench” as the data target pool for the default placement storage policy.

```
[root@cephadm ~]# radosgw-admin zone get > zone.json
```

4. Edit the file zone.json by changing data_pool to “cosbench”:

```

[root@cephadm ~]# vi zone.json
{
    "id": "f6103943-dfd8-4725-b796-4be6f3757cbb",
    "name": "default",
    "domain_root": "default.rgw.meta:root",
    "control_pool": "default.rgw.control",
    "gc_pool": "default.rgw.log:gc",
    "lc_pool": "default.rgw.log:lc",

```

```

"log_pool": "default.rgw.log",
"intent_log_pool": "default.rgw.log:intent",
"usage_log_pool": "default.rgw.log:usage",
"reshard_pool": "default.rgw.log:reshard",
"user_keys_pool": "default.rgw.meta:users.keys",
"user_email_pool": "default.rgw.meta:users.email",
"user_swift_pool": "default.rgw.meta:users.swift",
"user_uid_pool": "default.rgw.meta:users.uid",
"otp_pool": "default.rgw.otp",
"system_key": {
    "access_key": "",
    "secret_key": ""
},
"placement_pools": [
    {
        "key": "default-placement",
        "val": {
            "index_pool": "default.rgw.buckets.index",
            "storage_classes": {
                "STANDARD": {
                    "data_pool": "cosbench"
                }
            },
            "data_extra_pool": "default.rgw.buckets.non-ec",
            "index_type": 0
        }
    }
],
"metadata_heap": "",
"realm_id": ""
}

```

5. Set the new zone configuration and update the zone group map.

```

[root@cephadm ~]# radosgw-admin zone set < zone.json
zone id f6103943-dfd8-4725-b796-4be6f3757cbb{
  "id": "f6103943-dfd8-4725-b796-4be6f3757cbb",
  "name": "default",
  "domain_root": "default.rgw.meta:root",
  "control_pool": "default.rgw.control",
  "gc_pool": "default.rgw.log:gc",

```

```

"lc_pool": "default.rgw.log:lc",
"log_pool": "default.rgw.log",
"intent_log_pool": "default.rgw.log:intent",
"usage_log_pool": "default.rgw.log:usage",
"reshard_pool": "default.rgw.log:reshard",
"user_keys_pool": "default.rgw.meta:users.keys",
"user_email_pool": "default.rgw.meta:users.email",
"user_swift_pool": "default.rgw.meta:users.swift",
"user_uid_pool": "default.rgw.meta:users.uid",
"otp_pool": "default.rgw.otp",
"system_key": {
  "access_key": "",
  "secret_key": ""
},
"placement_pools": [
  {
    "key": "default-placement",
    "val": {
      "index_pool": "default.rgw.buckets.index",
      "storage_classes": {
        "STANDARD": {
          "data_pool": "cosbench"
        }
      },
      "data_extra_pool": "default.rgw.buckets.non-ec",
      "index_type": 0
    }
  }
],
"metadata_heap": "",
"realm_id": ""
}

```

6. Restart all RGW container on the OSD nodes and verify the uptime. It should take just a few seconds.

```

[root@cephadm ~]# for i in {0,1}; do for j in {1..6}; do ssh -t root@cephosd$j "podman
restart ceph-rgw-cephosd$j-rgw$i"; done; done
0af7189016e4933016d0c955d6f1b7e64022a0706785bb88e505b1616d1c7429
Connection to cephosd1 closed.
28ecd698b4c0f214056a6b8d4b3f0884d7f0b046d7d3019c8335ebe8ed50463f
Connection to cephosd2 closed.

```

```
eca4573df85dcd0f04f84d87272e0a884106e5376e086d9f41f9979ea42bf66b
Connection to cephosd3 closed.
b90dd52a75d07dc9f83ccd528093ad5b5e5dc8415e59c6a98e5ab80082170801
Connection to cephosd4 closed.
075dea27a0cdeb280f9421f72b9bf36caa5d7d10b840233ede5eaf0172597866
Connection to cephosd5 closed.
20cc7e0995bd4e5ed416b4f7393e51dc4b6ff23feaa7c432a67db85890c9415
Connection to cephosd6 closed.
434924be533ae12f9042b46baa42e463e5521c69b97d661c4c72a197b66993af
Connection to cephosd1 closed.
2f9f57bcc1bae79fa6858f67f7fd71dc18fe1a236bb268511c55737e90c0410f
Connection to cephosd2 closed.
39c23000166d9877b83d6dcfa2819847c58ab08cd0108aa1cc3c6a17cd78af10
Connection to cephosd3 closed.
d00d6e5c06ad6ba997b7688f3559c7572d859719219ee0da20a90c642536c9be
Connection to cephosd4 closed.
07077f190bc9278136ca19c563869c114003050920eb4c48b2e3c150bf61eec6
Connection to cephosd5 closed.
f66ca630db705dcea036f7e3898241d8c524f56bf9566148d6c025039d9b471b
Connection to cephosd6 closed.
[root@cephadm ~]# for i in {1..6}; do ssh -t root@cephosd$i "podman ps -a | grep rgw";
done
0af7189016e4 registry.redhat.io/rhceph/rhceph-4-rhel8:latest
6 days ago Up 52 seconds ago ceph-rgw-cephosd1-rgw0
434924be533a registry.redhat.io/rhceph/rhceph-4-rhel8:latest
6 days ago Up 45 seconds ago ceph-rgw-cephosd1-rgw1
Connection to cephosd1 closed.
28ecd698b4c0 registry.redhat.io/rhceph/rhceph-4-rhel8:latest
6 days ago Up 51 seconds ago ceph-rgw-cephosd2-rgw0
2f9f57bcc1ba registry.redhat.io/rhceph/rhceph-4-rhel8:latest
6 days ago Up 44 seconds ago ceph-rgw-cephosd2-rgw1
Connection to cephosd2 closed.
39c23000166d registry.redhat.io/rhceph/rhceph-4-rhel8:latest
6 days ago Up 44 seconds ago ceph-rgw-cephosd3-rgw1
eca4573df85d registry.redhat.io/rhceph/rhceph-4-rhel8:latest
6 days ago Up 51 seconds ago ceph-rgw-cephosd3-rgw0
Connection to cephosd3 closed.
b90dd52a75d0 registry.redhat.io/rhceph/rhceph-4-rhel8:latest
6 days ago Up 50 seconds ago ceph-rgw-cephosd4-rgw0
d00d6e5c06ad registry.redhat.io/rhceph/rhceph-4-rhel8:latest
6 days ago Up 43 seconds ago ceph-rgw-cephosd4-rgw1
Connection to cephosd4 closed.
```



```
07077f190bc9 registry.redhat.io/rhceph/rhceph-4-rhel8:latest
6 days ago Up 42 seconds ago ceph-rgw-cephosd5-rgw1
075dea27a0cd registry.redhat.io/rhceph/rhceph-4-rhel8:latest
6 days ago Up 48 seconds ago ceph-rgw-cephosd5-rgw0
Connection to cephosd5 closed.
20cc7e0995bd registry.redhat.io/rhceph/rhceph-4-rhel8:latest
6 days ago Up 48 seconds ago ceph-rgw-cephosd6-rgw0
f66ca630db70 registry.redhat.io/rhceph/rhceph-4-rhel8:latest
6 days ago Up 42 seconds ago ceph-rgw-cephosd6-rgw1
Connection to cephosd6 closed.
```

7. Install the AWS CLI and to verify the S3 connection to the cluster and to store a file in a bucket.

```
[root@cephadm ~]# dnf -y install python3-pip
[root@cephadm ~]# pip3 install awscli --upgrade --user
[root@cephadm ~]# aws --version
aws-cli/1.18.199 Python/3.6.8 Linux/4.18.0-240.1.1.el8_3.x86_64 botocore/1.19.39
[root@cephadm ~]# aws configure
AWS Access Key ID [None]: CSH45H9Y15FU1HF7MHZS
AWS Secret Access Key [None]: G4y0pWi13LKR1tHJzi9hxcvCoi0du71jSuMir6Ac
Default region name [None]:
Default output format [None]:
[root@cephadm ~]# aws --endpoint-url=http://172.16.33.101:8080 s3 ls
```

8. Create a bucket for the previously created user and store a file into that bucket.

```
[root@cephadm ~]# aws --endpoint-url=http://172.16.33.101:8080 s3api create-bucket --
bucket cisco
[root@cephadm ~]# aws --endpoint-url=http://172.16.33.101:8080 s3 ls
2020-12-18 01:22:34 cisco
[root@cephadm ~]# aws --endpoint-url=http://172.16.33.101:8080 s3 cp /root/rhel8.2-
cephadm.iso s3://cisco/
upload: ./rhel8.2-cephadm.iso to s3://cisco/rhel8.2-cephadm.iso
[root@cephadm ~]# aws --endpoint-url=http://172.16.33.101:8080 s3 ls cisco
2020-12-18 01:27:25 8511741952 rhel8.2-cephadm.iso
```

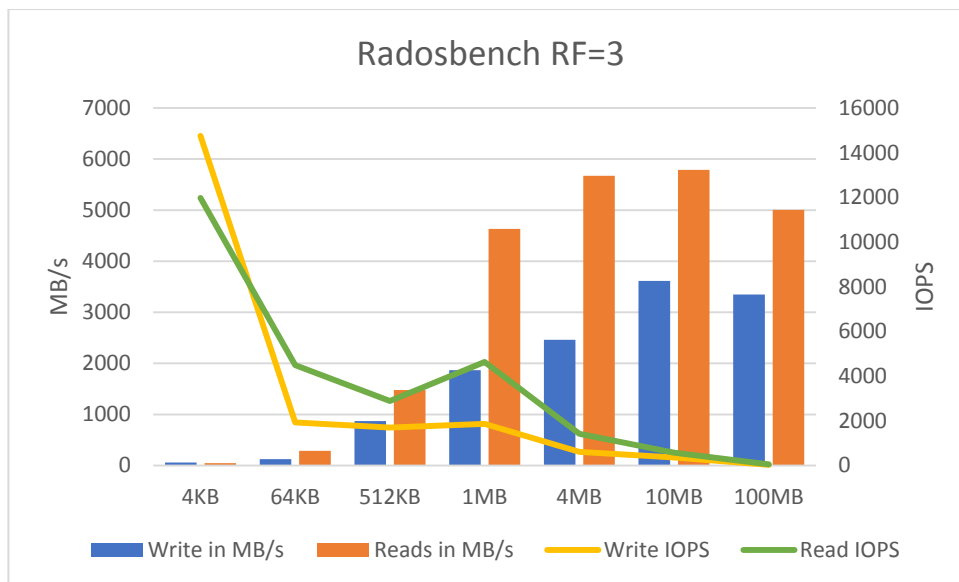
The installation is now finished, and data can be stored in either replicated or erasure-codes pools.

Red Hat Ceph Storage Performance Testing

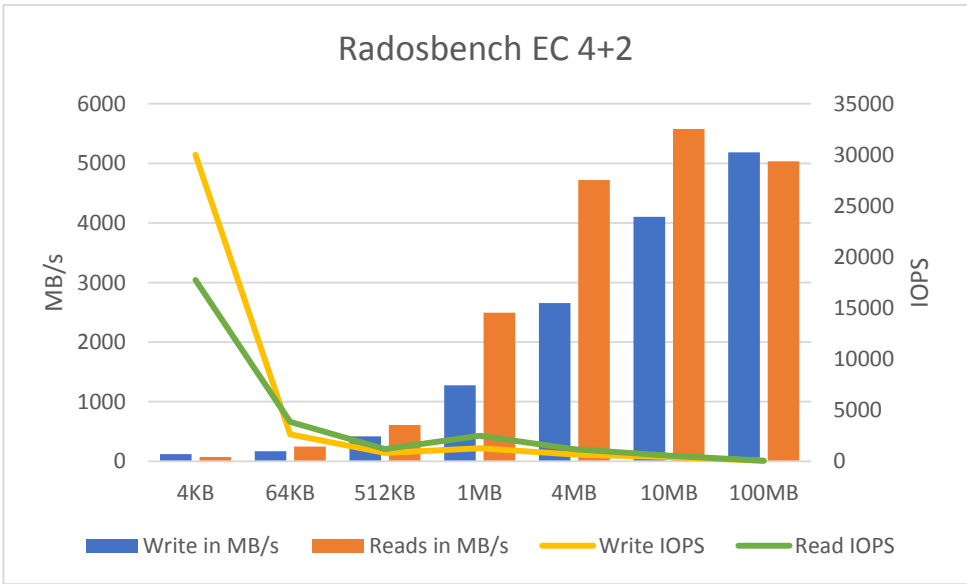
Performance was evaluated on Red Hat Ceph Storage 4 running on Cisco UCS C240 M5 hardware. The goal of the performance testing was to evaluate peak block and object performance under ideal conditions.

Radosbench Performance Tests

Radosbench performance testing was conducted with the Ceph Benchmarking Tool (CBT, <https://github.com/ceph/cbt>). Twelve virtual machines were used as Ceph clients to generate the Radosbench workload. Performance tests were conducted on replication pools with RF=3 and erasure coded pools with EC 4+2.



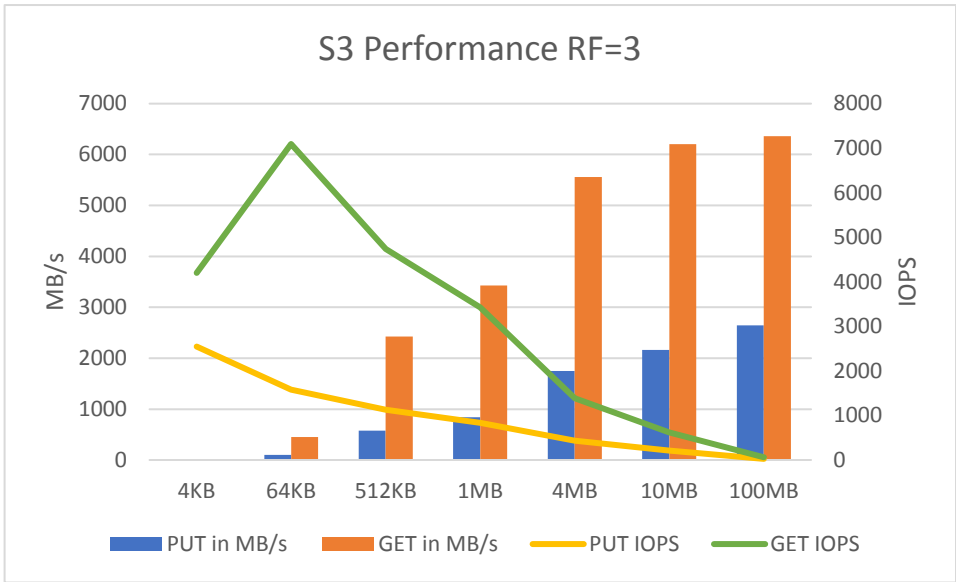
- Read bandwidth peaks at 5789 GB/s at an object size of 10MB. This translates to a disk performance of 80 MB/s/disk.
- Write bandwidth peaks at 3615 GB/s at an object size of 10MB. This translates to a disk performance of 50 MB/s/disk.



- Read bandwidth peaks at 5575 GB/s at an object size of 10MB. This translates to a disk performance of 77 MB/s/disk.
- Write bandwidth peaks at 5183 GB/s at an object size of 100MB. This translates to a disk performance of 72 MB/s/disk.

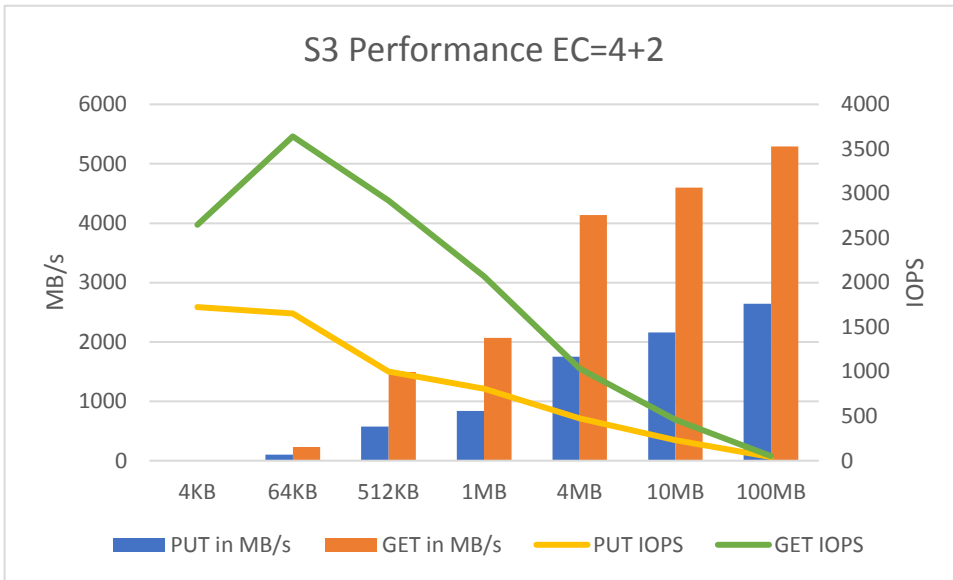
S3 Performance Tests

S3 performance testing was conducted with COSBench the standard cloud object storage benchmark. Twelve virtual machines were used as COSBench drivers to generate the object workload. Performance tests were conducted on replication pools with RF=3 and erasure coded pools with EC 4+2.



- GET bandwidth peaks at 6359 GB/s at an object size of 100MB. This translates to a disk performance of 88 MB/s/disk.

- PUT bandwidth peaks at 2642 GB/s at an object size of 100MB. This translates to a disk performance of 37 MB/s/disk.



- GET bandwidth peaks at 5294 GB/s at an object size of 100MB. This translates to a disk performance of 74 MB/s/disk.
- PUT bandwidth peaks at 3994 GB/s at an object size of 100MB. This translates to a disk performance of 55 MB/s/disk.

The Red Hat Ceph Storage configuration used for this CVD was sized for functional testing only. There was a limited number of clients that could be used to saturate the cluster. There are several architectural and design considerations that needs to carefully plan when sizing Ceph for high performance use cases, but we would expect to see slightly better numbers with more clients in this CVD. Please contact Red Hat for more information.

Red Hat Ceph Storage High Availability Testing

It is important for business continuity to help ensure high availability of the hardware and software stack. Some of these features are built into the Cisco UCS Infrastructure and enabled by the software stack and some of these features are possible from the Red Hat Ceph Storage software itself. To properly test for high availability, the following considerations were given priority:

- The Red Hat Ceph Storage deployment will process a reasonable amount of load when the fault is triggered. Total throughput will be recorded for S3 from the COSBench interface.
- Only a single fault will be triggered at any given time. Double failure is not a part of this consideration.
- Performance degradation is acceptable and even expected, but there should be no business interruption tolerated. The underlying infrastructure components should continue to operate within the remaining environment.

The following High Availability tests were performed:

- Cisco Nexus 93180YC-FX Switch A failure
- Cisco UCS C240 M5 - Red Hat Ceph Storage node disk failure
- Cisco UCS C240 M5 - Red Hat Ceph Storage node failure

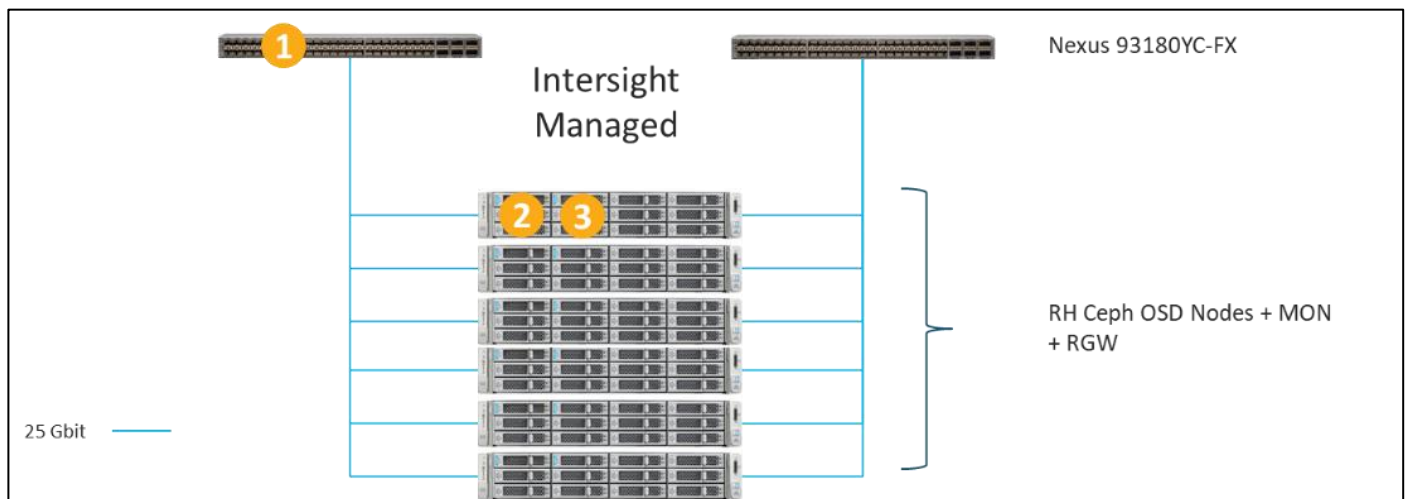
These tests will be performed for S3 protocol.

S3 Failover Testing

As indicated previously, a reasonable amount of load will be defined as follows:

- The COSBench application will be configured to send a steady stream of data to the Red Hat Ceph Storage cluster.

Figure 40. High Availability Testing



Cisco Nexus 93180YC-FX High Availability Testing

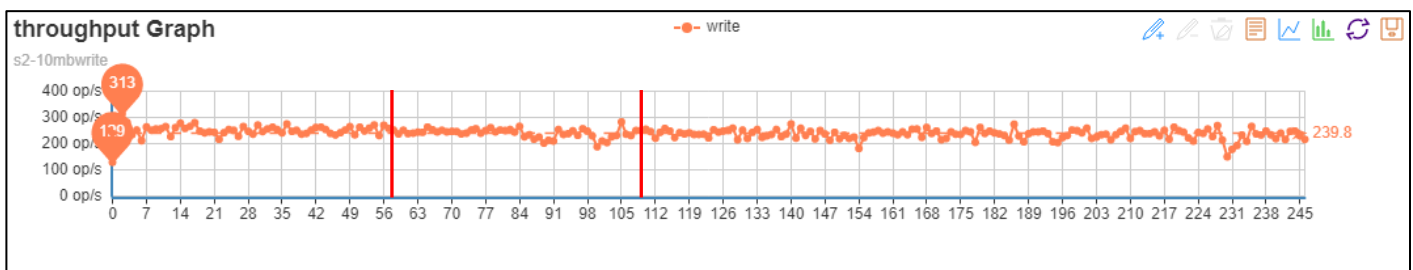
Sequence of Events

1. Connect to Cisco Nexus 93180YC-FX Switch A and make sure running-config is copied to startup-config to make certain no configuration changes are lost during power cycle.

```
sjc02dmz-i14-n93180ycfx-a# copy run start
[#####] 100%
Copy complete, now saving to disk (please wait)...
Copy complete.
```

2. Initiate load to the cluster by utilizing COSBench.
3. Initiate a reload command to reboot the switch.

```
sjc02dmz-i14-n93180ycfx-a(config)# reload
This command will reboot the system. (y/n)? [n] y
```



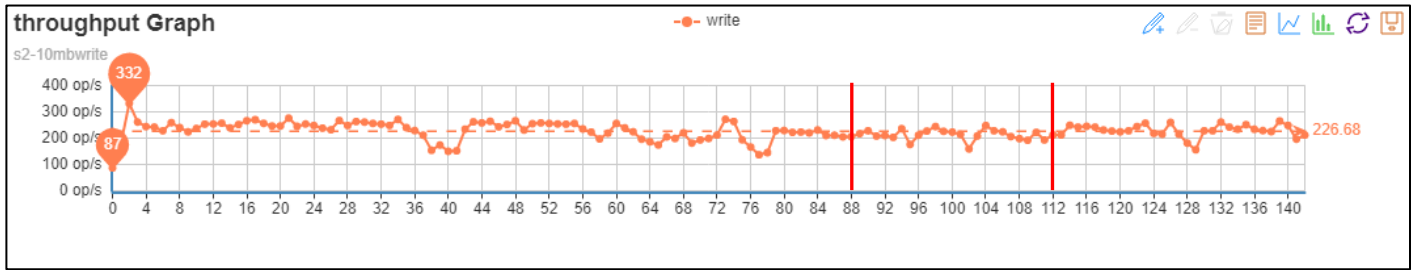
Aside from a loss of response from the Cisco Nexus 93180YC-FX switch, Red Hat Ceph Storage environment remained functional, load continued with almost no interruption, and redundancy was reestablished upon Switch A completing the reboot process. There was only a small loss in bandwidth between both red lines, but traffic continued.

Cisco UCS C240 M5L Disk Failure Testing

Sequence of Events

1. Connect to one of the storage nodes.
2. Initiate load to the cluster by utilizing COSBench.
3. Mark one of the Ceph OSD disks as out.

The graph below is a snapshot from COSBench.



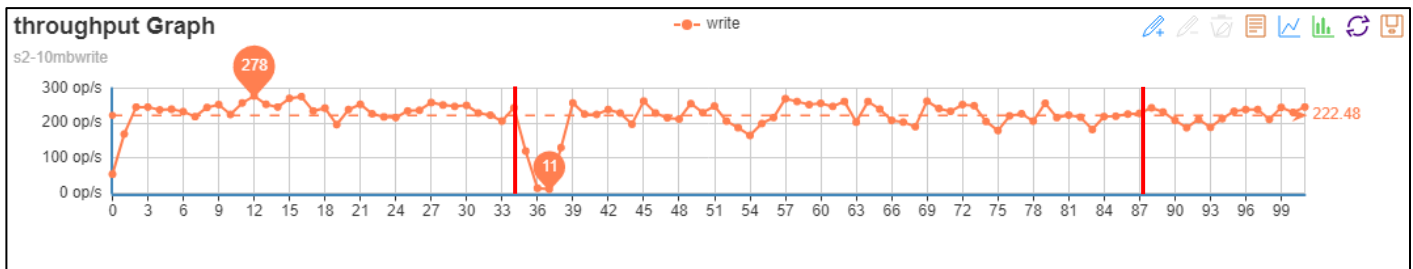
There was almost no loss in bandwidth between both red lines when the OSD was marked out.

Cisco UCS C240 M5L Node Failure Testing

Sequence of Events

1. Connect to Cisco Intersight and verify that the node is in a known good working condition.
2. Initiate load to the cluster by utilizing COSBench.
3. Power off one of the storage nodes.

The graph below is a snapshot from COSBench. At the vertical red line is where cephosd2 was powered off. An impact in throughput of about 30 seconds was observed. The cluster started the self-healing process, but no traffic interruption was observed. The overall throughput remained the same during recovery.



Appendix

Infrastructure Terraform Configuration File for MOIDs

```
# Intersight Provider Information
terraform {
  required_providers {
    intersight = {
      source = "CiscoDevNet/intersight"
      version = "0.1.3"
    }
  }
}

data "intersight_compute_physical_summary" "server_moid" {
  name = var.server_names[count.index]
  count = length(var.server_names)
}

output "server_moids" {
  value = data.intersight_compute_physical_summary.server_moid
}

data "intersight_organization_organization" "organization_moid" {
  name = var.organization_name
}

output "organization_moid" {
  value = data.intersight_organization_organization.organization_moid.moid
}

data "intersight_softwarerepository_catalog" "catalog_moid" {
  name = var.catalog_name
}

output "catalog_moid" {
  value = data.intersight_softwarerepository_catalog.catalog_moid.moid
}
```

Infrastructure Terraform Variable File for MOIDs

```
# Server and Organization names
variable "server_names" {
    type = list
}

variable "organization_name" {}

variable "catalog_name" {}
```

Infrastructure Terraform Variable File for Terraform Deployment

```
//Define all the basic variables here

variable "api_private_key" {
    default = "/root/terraform-intersight-sds/intersight.pem"
}

variable "api_key_id" {
    default =
"5e5fb2b17564612d3028b5b4/5e5fbd137564612d3028bcc4/5fa1a9107564612d3007f934"
}

variable "api_endpoint" {
    default = "https://sjc02dmz-intersight.sjc02dmz.net"
}

variable "management_vlan" {
    default = 300
}

variable "client_vlan" {
    default = 301
}

variable "storage_vlan" {
    default = 302
}

variable "remote-server" {
    default = "sjc02dmz-i14-terraform.sjc02dmz.net"
```

```
}

variable "remote-share" {
  default = "/images"
}

variable "remote-os-image-cephosd" {
  type = list(string)
  default = ["rhel8.2-cephosd1.iso", "rhel8.2-cephosd2.iso", "rhel8.2-cephosd3.iso",
"rhel8.2-cephosd4.iso", "rhel8.2-cephosd5.iso", "rhel8.2-cephosd6.iso", "rhel8.2-
cephadm.iso"]
}

variable "remote-os-image-link" {
  type = list(string)
  default = ["http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephosd1.iso",
"http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephosd2.iso",
"http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephosd3.iso",
"http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephosd4.iso",
"http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephosd5.iso",
"http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephosd6.iso",
"http://sjc02dmz-i14-terraform.sjc02dmz.net/images/rhel8.2-cephadm.iso"]
}

variable "remote-protocol" {
  default = "softwarerepository.HttpServer"
}

variable "server_names" {
  default = ["sjc02dmz-i14-ceph1", "sjc02dmz-i14-ceph2", "sjc02dmz-i14-ceph3",
"sjc02dmz-i14-ceph4", "sjc02dmz-i14-ceph5", "sjc02dmz-i14-ceph6", "sjc02dmz-i14-
cephadm"]
}

variable "organization_name" {
  default = "Ceph"
}

variable "server_profile_action" {
  default = "No-op"
}
```

```
variable "catalog_name" {
  default = "appliance-system-catalog"
}
```

Infrastructure Terraform Configuration File for Policies and Profiles

```
# Intersight Provider Information
terraform {
  required_providers {
    intersight = {
      source = "CiscoDevNet/intersight"
      version = "0.1.3"
    }
  }
}

provider "intersight" {
  apikey      = var.api_key_id
  secretkeyfile = var.api_private_key
  endpoint    = var.api_endpoint
}

module "intersight-moids" {
  source          = "../..//terraform-intersight-moids"
  server_names   = var.server_names
  organization_name = var.organization_name
  catalog_name   = var.catalog_name
}

resource "intersight_server_profile" "cephosd" {
  name = "SP-${var.server_names[count.index]}"
  organization {
    object_type = "organization.Organization"
    moid        = module.intersight-moids.organization_moid
  }

  assigned_server {
    moid        = module.intersight-moids.server_moids[count.index].moid
    object_type = "compute.RackUnit"
  }

  action = var.server_profile_action
}
```

```
count = length(var.server_names)
}

resource "intersight_networkconfig_policy" "ceph-network-policy" {
  name          = "ceph-network-policy"
  description   = "DNS Configuration Policy for CIMC"
  organization {
    object_type = "organization.Organization"
    moid       = module.intersight-moids.organization_moid
  }
  preferred_ipv4dns_server = "192.168.10.51"
  alternate_ipv4dns_server = ""
  dynamic "profiles" {
    for_each = intersight_server_profile.cephosd
    content {
      moid = profiles.value["moid"]
      object_type = "server.Profile"
    }
  }
}

resource "intersight_adapter_config_policy" "ceph-adapter-config-policy" {
  name          = "ceph-adapter-config-policy"
  description   = "Adapter Configuration Policy for Ceph"
  organization {
    object_type = "organization.Organization"
    moid       = module.intersight-moids.organization_moid
  }
  settings {
    slot_id = "MLOM"
    dce_interface_settings {
      fec_mode = "cl74"
      interface_id = "0"
    }
    dce_interface_settings {
      fec_mode = "cl74"
      interface_id = "1"
    }
    dce_interface_settings {
```

```
        fec_mode = "cl74"
        interface_id = "2"
    }
    dce_interface_settings {
        fec_mode = "cl74"
        interface_id = "3"
    }
    eth_settings {
        lldp_enabled = true
    }
    fc_settings {
        fip_enabled = false
    }
    port_channel_settings {
        enabled = "true"
    }
}
dynamic "profiles" {
    for_each = intersight_server_profile.cephosd
    content {
        moid = profiles.value["moid"]
        object_type = "server.Profile"
    }
}
}

resource "intersight_vnic_eth_adapter_policy" "ceph-ethernet-adapter-policy" {
    name = "ceph-ethernet-adapter-policy"
    description = "Ethernet Adapter Policy for Ceph"
    rss_settings = true
    organization {
        object_type = "organization.Organization"
        moid = module.intersight-moids.organization_moid
    }
    vxlan_settings {
        object_type = "vnic.VxlanSettings"
        enabled = false
    }
    nvgre_settings {
```

```
enabled = false
object_type = "vnic.NvgreSettings"
}
arfs_settings {
    object_type = "vnic.ArfsSettings"
    enabled = true
}
roce_settings {
    object_type = "vnic.RoceSettings"
    enabled = false
}
interrupt_settings {
    coalescing_time = 125
    coalescing_type = "MIN"
    nr_count        = 32
    mode            = "MSIX"
    object_type     = "vnic.EthInterruptSettings"
}
completion_queue_settings {
    object_type = "vnic.CompletionQueueSettings"
    nr_count    = 16
    ring_size   = 1
}
rx_queue_settings {
    object_type = "vnic.EthRxQueueSettings"
    nr_count    = 8
    ring_size   = 4096
}
tx_queue_settings {
    object_type = "vnic.EthTxQueueSettings"
    nr_count    = 8
    ring_size   = 4096
}
tcp_offload_settings {
    object_type = "vnic.TcpOffloadSettings"
    large_receive = true
    large_send    = true
    rx_checksum   = true
    tx_checksum   = true
}
```

```
    }
}

resource "intersight_vnic_eth_network_policy" "ceph-mgt-network" {
  name = "ceph-mgt-network"
  description = "Mgt Network for Ceph"
  organization {
    object_type = "organization.Organization"
    moid = module.intersight-moids.organization_moid
  }
  vlan_settings {
    object_type = "vnic.VlanSettings"
    default_vlan = var.management_vlan
    mode          = "TRUNK"
  }
}

resource "intersight_vnic_eth_network_policy" "ceph-client-network" {
  name = "ceph-client-network"
  description = "Client Network for Ceph"
  organization {
    object_type = "organization.Organization"
    moid = module.intersight-moids.organization_moid
  }
  vlan_settings {
    object_type = "vnic.VlanSettings"
    default_vlan = var.client_vlan
    mode          = "TRUNK"
  }
}

resource "intersight_vnic_eth_network_policy" "ceph-storage-network" {
  name = "ceph-storage-network"
  description = "Storage Network for Ceph"
  organization {
    object_type = "organization.Organization"
    moid = module.intersight-moids.organization_moid
  }
  vlan_settings {
```

```
        object_type = "vnic.VlanSettings"
        default_vlan = var.storage_vlan
        mode         = "TRUNK"
    }
}

resource "intersight_vnic_eth_qos_policy" "ceph-ethernet-qos-9000-policy" {
    name          = "ceph-ethernet-qos-9000-policy"
    description   = "Ethernet quality of service for Ceph"
    mtu           = 9000
    rate_limit    = 0
    cos           = 0
    trust_host_cos = false
    organization {
        object_type = "organization.Organization"
        moid        = module.intersight-moids.organization_moid
    }
}

resource "intersight_vnic_eth_qos_policy" "ceph-ethernet-qos-1500-policy" {
    name          = "ceph-ethernet-qos-1500-policy"
    description   = "Ethernet quality of service for Ceph"
    mtu           = 1500
    rate_limit    = 0
    cos           = 0
    trust_host_cos = false
    organization {
        object_type = "organization.Organization"
        moid        = module.intersight-moids.organization_moid
    }
}

resource "intersight_vnic_lan_connectivity_policy" "ceph-lan-connectivity-policy" {
    name = "ceph-lan-connectivity-policy"
    description = "LAN Connectivity Policy for Ceph"
    organization {
        object_type = "organization.Organization"
        moid        = module.intersight-moids.organization_moid
    }
}
```



```
dynamic "profiles" {
  for_each = intersight_server_profile.cephosd
  content {
    moid = profiles.value["moid"]
    object_type = "server.Profile"
  }
}

resource "intersight_vnic_eth_if" "eth0" {
  name = "eth0"
  order = 0
  placement {
    object_type = "vnic.PlacementSettings"
    id = "MLOM"
    pci_link = 0
    uplink = 0
  }
  cdn {
    nr_source = "vnic"
  }
  vmq_settings {
    enabled = false
    num_interrupts = 1
    num_vmq_s = 1
  }
  lan_connectivity_policy {
    moid = intersight_vnic_lan_connectivity_policy.ceph-lan-connectivity-policy.id
    object_type = "vnic.LanConnectivityPolicy"
  }
  eth_network_policy {
    moid = intersight_vnic_eth_network_policy.ceph-mgt-network.id
  }
  eth_adapter_policy {
    moid = intersight_vnic_eth_adapter_policy.ceph-ethernet-adapter-policy.id
  }
  eth_qos_policy {
    moid = intersight_vnic_eth_qos_policy.ceph-ethernet-qos-1500-policy.id
  }
}
```

```
}

resource "intersight_vnic_eth_if" "eth1" {
  name = "eth1"
  order = 1
  placement {
    object_type = "vnic.PlacementSettings"
    id = "MLOM"
    pci_link = 0
    uplink = 0
  }
  cdn {
    nr_source = "vnic"
  }
  vmq_settings {
    enabled = false
    num_interrupts = 1
    num_vmqs = 1
  }
  lan_connectivity_policy {
    moid = intersight_vnic_lan_connectivity_policy.ceph-lan-connectivity-policy.id
    object_type = "vnic.LanConnectivityPolicy"
  }
  eth_network_policy {
    moid = intersight_vnic_eth_network_policy.ceph-client-network.id
  }
  eth_adapter_policy {
    moid = intersight_vnic_eth_adapter_policy.ceph-ethernet-adapter-policy.id
  }
  eth_qos_policy {
    moid = intersight_vnic_eth_qos_policy.ceph-ethernet-qos-9000-policy.id
  }
}

resource "intersight_vnic_eth_if" "eth2" {
  name = "eth2"
  order = 2
  placement {
    object_type = "vnic.PlacementSettings"
```

```
    id      = "MLOM"
    pci_link = 0
    uplink  = 1
  }
  cdn {
    nr_source = "vnic"
  }
  vmq_settings {
    enabled = false
    num_interrupts = 1
    num_vmqs = 1
  }
  lan_connectivity_policy {
    moid      = intersight_vnic_lan_connectivity_policy.ceph-lan-connectivity-
policy.id
    object_type = "vnic.LanConnectivityPolicy"
  }
  eth_network_policy {
    moid = intersight_vnic_eth_network_policy.ceph-client-network.id
  }
  eth_adapter_policy {
    moid = intersight_vnic_eth_adapter_policy.ceph-ethernet-adapter-policy.id
  }
  eth_qos_policy {
    moid = intersight_vnic_eth_qos_policy.ceph-ethernet-qos-9000-policy.id
  }
}

resource "intersight_vnic_eth_if" "eth3" {
  name = "eth3"
  order = 3
  placement {
    object_type = "vnic.PlacementSettings"
    id      = "MLOM"
    pci_link = 0
    uplink = 0
  }
  cdn {
    nr_source = "vnic"
  }
}
```

```
vmq_settings {
    enabled = false
    num_interrupts = 1
    num_vmq_s = 1
}
lan_connectivity_policy {
    moid      = intersight_vnic_lan_connectivity_policy.ceph-lan-connectivity-
policy.id
    object_type = "vnic.LanConnectivityPolicy"
}
eth_network_policy {
    moid = intersight_vnic_eth_network_policy.ceph-storage-network.id
}
eth_adapter_policy {
    moid = intersight_vnic_eth_adapter_policy.ceph-ethernet-adapter-policy.id
}
eth_qos_policy {
    moid = intersight_vnic_eth_qos_policy.ceph-ethernet-qos-9000-policy.id
}
}

resource "intersight_vnic_eth_if" "eth4" {
    name = "eth4"
    order = 4
    placement {
        object_type = "vnic.PlacementSettings"
        id      = "MLOM"
        pci_link = 0
        uplink = 1
    }
    cdn {
        nr_source = "vnic"
    }
    vmq_settings {
        enabled = false
        num_interrupts = 1
        num_vmq_s = 1
    }
    lan_connectivity_policy {
```

```
    moid          = intersight_vnic_lan_connectivity_policy.ceph-lan-connectivity-
policy.id
    object_type = "vnic.LanConnectivityPolicy"
}
eth_network_policy {
    moid = intersight_vnic_eth_network_policy.ceph-storage-network.id
}
eth_adapter_policy {
    moid = intersight_vnic_eth_adapter_policy.ceph-ethernet-adapter-policy.id
}
eth_qos_policy {
    moid = intersight_vnic_eth_qos_policy.ceph-ethernet-qos-9000-policy.id
}
}

resource "intersight_ntp_policy" "ceph-ntp-policy" {
    name      = "ceph-ntp-policy"
    description = "NTP Policy for Ceph"
    enabled = true
    ntp_servers = [
        "173.38.201.115"
    ]
    organization {
        object_type = "organization.Organization"
        moid = module.intersight-moids.organization_moid
    }
    dynamic "profiles" {
        for_each = intersight_server_profile.cephosd
        content {
            moid = profiles.value["moid"]
            object_type = "server.Profile"
        }
    }
}

resource "intersight_storage_disk_group_policy" "ceph-disk-group-boot-policy-c220" {
    name          = "ceph-disk-group-boot-policy-c220"
    description = "Disk Group Boot Policy for Ceph Admin"
    raid_level   = "Raid1"
    use_jbods    = true
}
```

```
span_groups {
  disks {
    slot_number = 1
  }
  disks {
    slot_number = 2
  }
}
organization {
  object_type = "organization.Organization"
  moid = module.intersight-moids.organization_moid
}
}

resource "intersight_storage_disk_group_policy" "ceph-disk-group-boot-policy-c240" {
  name          = "ceph-disk-group-boot-policy-c240"
  description   = "Disk Group Boot Policy for Ceph OSD"
  raid_level    = "Raid1"
  use_jbods     = true
  span_groups {
    disks {
      slot_number = 13
    }
    disks {
      slot_number = 14
    }
  }
  organization {
    object_type = "organization.Organization"
    moid = module.intersight-moids.organization_moid
  }
}

resource "intersight_storage_storage_policy" "ceph-storage-policy-osd" {
  name          = "ceph-storage-policy-osd"
  description   = "Storage Policy for Ceph OSD"
  retain_policy_virtual_drives = false
  unused_disks_state          = "Jbod"
  virtual_drives {
```

```
object_type = "storage.VirtualDriveConfig"
boot_drive = true
drive_cache = "Default"
expand_to_available = true
io_policy = "Default"
name = "ceph-os-boot"
access_policy = "ReadWrite"
disk_group_policy = intersight_storage_disk_group_policy.ceph-disk-group-boot-
policy-c240.id
read_policy = "ReadAhead"
write_policy = "WriteBackGoodBbu"
}
organization {
  object_type = "organization.Organization"
  moid = module.intersight-moids.organization_moid
}
profiles {
  moid = intersight_server_profile.cephosd[0].id
  object_type = "server.Profile"
}
profiles {
  moid = intersight_server_profile.cephosd[1].id
  object_type = "server.Profile"
}
profiles {
  moid = intersight_server_profile.cephosd[2].id
  object_type = "server.Profile"
}
profiles {
  moid = intersight_server_profile.cephosd[3].id
  object_type = "server.Profile"
}
profiles {
  moid = intersight_server_profile.cephosd[4].id
  object_type = "server.Profile"
}
profiles {
  moid = intersight_server_profile.cephosd[5].id
  object_type = "server.Profile"
}
```

```
}

resource "intersight_storage_storage_policy" "ceph-storage-policy-admin" {
  name                = "ceph-storage-policy-admin"
  description         = "Storage Policy for Ceph Admin"
  retain_policy_virtual_drives = false
  unused_disks_state = "Jbod"
  virtual_drives {
    object_type = "storage.VirtualDriveConfig"
    boot_drive = true
    drive_cache = "Default"
    expand_to_available = true
    io_policy = "Default"
    name = "ceph-os-boot"
    access_policy = "ReadWrite"
    disk_group_policy = intersight_storage_disk_group_policy.ceph-disk-group-boot-policy-c220.id
    read_policy = "ReadAhead"
    write_policy = "WriteBackGoodBbu"
  }
  organization {
    object_type = "organization.Organization"
    moid = module.intersight-moids.organization_moid
  }
  profiles {
    moid      = intersight_server_profile.cephosd[6].id
    object_type = "server.Profile"
  }
}

resource "intersight_boot_precision_policy" "ceph-boot-policy" {
  name                = "ceph-boot-policy"
  description         = "Boot Policy for Ceph"
  configured_boot_mode = "Legacy"
  enforce_uefi_secure_boot = false
  organization {
    object_type = "organization.Organization"
    moid = module.intersight-moids.organization_moid
  }
  boot_devices {
```



```

    enabled      = true
    name         = "disk"
    object_type  = "boot.LocalDisk"
    additional_properties = jsonencode({
      Slot = "MRAID"
    })
  }
  boot_devices {
    enabled      = true
    name         = "vmedia"
    object_type  = "boot.VirtualMedia"
    additional_properties = jsonencode({
      Subtype = "cimc-mapped-dvd"
    })
  }
  dynamic "profiles" {
    for_each = intersight_server_profile.cephosd
    content {
      moid = profiles.value["moid"]
      object_type = "server.Profile"
    }
  }
}

```

Infrastructure Terraform Configuration File for Deploying Profiles

```

# Intersight Provider Information
terraform {
  required_providers {
    intersight = {
      source = "CiscoDevNet/intersight"
      version = ">= 0.1.3"
    }
  }
}

provider "intersight" {
  apikey      = var.api_key_id
  secretkeyfile = var.api_private_key
  endpoint    = var.api_endpoint
}

```

```

module "intersight-moids" {
  source          = "../../terraform-intersight-moids"
  server_names    = var.server_names
  organization_name = var.organization_name
  catalog_name    = var.catalog_name
}

resource "intersight_server_profile" "cephosd" {
  count = length(var.server_names)
  name = "SP-${var.server_names[count.index]}"
  organization {
    object_type = "organization.Organization"
    moid = module.intersight-moids.organization_moid
  }
  assigned_server {
    moid          = module.intersight-moids.server_moids[count.index].moid
    object_type = "compute.RackUnit"
  }
  action = "Deploy"
}

```

Infrastructure Terraform Configuration File for Installing OS

```

# Intersight Provider Information
terraform {
  required_providers {
    intersight = {
      source = "CiscoDevNet/intersight"
      version = ">= 0.1.3"
    }
  }
}

provider "intersight" {
  apikey          = var.api_key_id
  secretkeyfile = var.api_private_key
  endpoint        = var.api_endpoint
}

module "intersight-moids" {

```

```
source          = "../../terraform-intersight-moids"
server_names    = var.server_names
organization_name = var.organization_name
catalog_name    = var.catalog_name
}

resource "intersight_softwarerepository_operating_system_file" "rhel-custom-iso-with-kickstart-cephosd" {
  count = length(var.server_names)
  nr_version = "Red Hat Enterprise Linux 8.2"
  description = "RHEL 8.2 installer ISO with embedded kickstart cephosd"
  name = "ISO-${var.server_names[count.index]}"
  nr_source {
    additional_properties = jsonencode({
      LocationLink = var.remote-os-image-link[count.index]
    })
    object_type = var.remote-protocol
  }
  vendor = "Red Hat"
  catalog {
    moid = module.intersight-moids.catalog_moid
  }
}

resource "intersight_os_install" "cephosd" {
  count = length(var.server_names)
  name = "ceph-os-${var.server_names[count.index]}"
  server {
    object_type = "compute.RackUnit"
    moid = module.intersight-moids.server_moids[count.index].moid
  }
  image {
    object_type = "softwarerepository.OperatingSystemFile"
    moid = intersight_softwarerepository_operating_system_file.rhel-custom-iso-with-kickstart-cephosd[count.index].moid
  }
  answers {
    nr_source = "Embedded"
  }
  description = "OS install"
```

```

install_method = "vMedia"
organization {
    object_type = "organization.Organization"
    moid = module.intersight-moids.organization_moid
}
}

```

Kickstart File for Red Hat Ceph Storage

```

lang en_US.UTF-8
keyboard --vckeymap=us --xlayouts='us'
timezone --isUtc America/Los_Angeles --ntpserver=173.38.201.115
# System services
services --enabled="chronyd"
rootpw
$6$dA8apVZJJhnc1jrS$IuVqcdAuHQVijluX6S6vw88FYteyog12ZZczrFDRhIROitEIWdI41SjPSsgNgIoVGb3
YanQGm.lyWsK7v48P81 --iscrypted
#platform x86, AMD64, or Intel EM64T
cdrom
reboot
#Network Information
network --bootproto=static --device=eth0 --ip=172.16.32.101 --netmask=255.255.255.0 --
gateway=172.16.32.1 --hostname=cephosd1.sjc02dmz.net --nameserver=192.168.10.51 --
noipv6 --mtu=1500 --onboot=on --activate
network --bootproto=static --device=team1 --ip=172.16.33.101 --netmask=255.255.255.0 -
-noipv6 --mtu=9000 --onboot=on --activate --teamslaves="eth1,eth2" --
teamconfig="{\"runner\": {\"name\": \"loadbalance\"}}"
network --bootproto=static --device=team2 --ip=172.16.34.101 --netmask=255.255.255.0 -
-noipv6 --mtu=9000 --onboot=on --activate --teamslaves="eth3,eth4" --
teamconfig="{\"runner\": {\"name\": \"loadbalance\"}}"

bootloader --location=mbr --append="rhgb quiet crashkernel=auto" --boot-
drive=/dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0
clearpart --all --initlabel
zerombr
# Disk partitioning information
part pv.1 --fstype="lvm" --ondisk=/dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0 --
size=890000
part /boot --fstype="xfs" --ondisk=/dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0 --
size=1024
volgroup ceph --pesize=4096 pv.1
logvol /home --fstype="xfs" --size=10240 --name=home --vgname=ceph
logvol swap --fstype="swap" --size=4096 --name=swap --vgname=ceph
logvol / --fstype="xfs" --size=204800 --name=root --vgname=ceph

```

```
logvol /var --fstype="xfs" --grow --size=1 --name=var --vgname=ceph
logvol /tmp --fstype="xfs" --size=40960 --name=tmp --vgname=ceph
auth --passalgo=sha512 --useshadow
selinux --disabled
firewall --disabled
firstboot --disable
ignoredisk --only-use=/dev/disk/by-path/pci-0000:18:00.0-scsi-0:2:0:0

%packages
@^minimal-environment
chrony
kexec-tools
%end

%addon com_redhat_kdump --enable --reserve-mb='auto'

%end
```

Summary

Object storage is an increasingly popular form of distributing data in a scale-out system. Cisco UCS with Red Hat Ceph Storage is one of the most valuable and performant scale-out storage solutions on the market. The solution in this design guide provides customers and partners with everything necessary to store object data easily and securely. Cisco's leading technology of centralized management and advanced networking technology helps to easily deploy, manage, and operate the Red Hat Ceph Storage solution.

The way infrastructure is managed by Cisco Intersight together with Terraform Orchestration enables customers to easily install and configure a multi-node scale-out storage infrastructure without any boundaries.

About the Authors

Oliver Walsdorf, Technical Marketing Engineer for Software Defined Storage, Computer Systems Product Group, Cisco Systems, Inc.

Oliver has more than 20 years of storage experience, working in different roles at different storage vendors, and is now an expert for software-defined storage at Cisco. For the past four years Oliver was focused on developing storage solutions at Cisco. He now works on Red Hat Ceph Storage, develops Co-Solutions with Red Hat for the overall storage market and published several Cisco documents. With his focus on SDS he drives the overall attention in the market for new technologies. In his leisure time, Oliver enjoys hiking with his dogs and motorcycling.

Acknowledgements

For their support and contribution to the design, validation, and creation of this Cisco Validated Design, we would like to acknowledge the following for their significant contribution and expertise that resulted in developing this document:

- Chris O'Brien, Cisco Systems, Inc.
- Jawwad Memon, Cisco Systems, Inc.
- Karan Singh, Red Hat, Inc.

Feedback

For comments and suggestions about this guide and related guides, join the discussion on [Cisco Community](https://cs.co/en-cvds) at <https://cs.co/en-cvds>.

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at <https://www.cisco.com/go/offices>.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/go/trademarks>. Third-party trademarks mentioned 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. (1110R)