

Cisco Data Intelligence Platform on Cisco UCS S3260 with Cloudera Data Platform

Deployment Guide for Cisco Data Intelligence Platform with Cisco UCS S3260 and Cloudera Data Platform Data Center 7.1.1

Published: August 2020



In partnership with: **CLOUDERA**

About the Cisco Validated Design Program

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

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

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

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

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

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

© 2020 Cisco Systems, Inc. All rights reserved.

Table of Contents

Executive Summary	6
Solution Overview	7
Introduction.....	7
Audience	7
Purpose of this Document.....	7
What's New in this Release?	8
What's Next?	8
Solution Summary	8
Cisco Data Intelligence Platform	8
CDP on CDIP.....	10
Reference Architecture	11
Data Lake Reference Architecture	11
Scaling the Solution.....	13
Technology Overview	16
Cisco UCS Integrated Infrastructure for Big Data and Analytics	16
Cisco UCS	16
Cisco Intersight	17
Cisco UCS Manager.....	18
Cisco UCS 6300 Series Fabric Interconnects	19
Cisco UCS 6400 Series Fabric Interconnect	19
Cisco UCS C-Series Rack-Mount Servers	20
Cisco UCS S3260 Storage Servers.....	20
Cisco UCS Virtual Interface Cards (VICs).....	22
Cloudera Data Platform (CDP) Data Center.....	24
Apache Ozone (Tech preview)	24
Red Hat Ansible Automation.....	25
Solution Design	26
Requirements	26
Physical Topology	26
Port Configuration on Fabric Interconnect.....	27
Server Configuration and Cabling for Cisco UCS S3260.....	27
Software Distributions and Firmware Versions	28
Cisco Intersight.....	29
Connected TAC	29
Cisco Intersight Integration for HCL	31

Advisories (PSIRTs).....	32
Deployment Hardware and Software	34
Cisco Unified Computing System Configuration	34
Configure Cisco UCS Fabric Interconnect	34
Configure Fabric Interconnects for a Cluster Setup	34
Register UCSM with Intersight.....	36
Configure Cisco UCS Manager through Intersight	39
Create Service Profile Template	78
Install Red Hat Enterprise Linux 7.7.....	90
Post OS Install Configuration	115
Configure /etc/hosts	115
Set Up Passwordless Login	116
Create a Red Hat Enterprise Linux (RHEL) 7.7 Local Repository	117
Create the Red Hat Repository Database.....	118
Set Up Ansible	119
Install httpd.....	121
Disable the Linux Firewall	121
Set Up All Nodes to use the RHEL Repository	121
Disable SELinux.....	122
Upgrade the Cisco Network Driver for VIC1387	123
Set Up JAVA	123
Enable Syslog.....	125
Set the ulimit	125
Set TCP Retries.....	126
Disable IPv6 Defaults.....	127
Disable Swapping.....	127
Disable Memory Overcommit	127
Disable Transparent Huge Pages	128
NTP Configuration	128
Install Megaraid StorCLI	130
Configure the Filesystem for NameNodes and DataNodes	130
Delete Partitions	132
Cluster Verification	132
Install Cloudera Data Platform.....	134
Prerequisites for CDP DC Installation	135
Set Up the Local Parcels for CDP DC 7.1.1	136
Download Parcels	136

Install and Configure Database for Cloudera Manager	137
Databases for CDP.....	138
Cloudera Manager Installation.....	139
Install Cloudera Manager.....	140
Set Up the Cloudera Manager Server Database.....	140
Install Cloudera Data Platform Data Center (CDP DC 7)	141
Scale the Cluster.....	157
Rack-Aware Replica Placement.....	157
Enable High Availability	161
HDFS High Availability.....	161
Configure Hive Metastore to Use HDFS High Availability.....	165
Configure Hue to Work with HDFS High Availability.....	167
YARN High Availability	170
Configure Yarn (MR2 Included) and HDFS Services	171
Configure Spark.....	172
Tune Resource Allocation for Spark	173
Submit a Job	174
Shuffle Performance Improvement	175
Improve Serialization Performance	175
Spark SQL Tuning	175
Compression for Hive.....	176
Change the Log Directory for All Applications	176
Summary	177
For More Information	177
Bill of Materials	178
Appendix.....	180
Configure Data Drives on Name Node and Other Management Nodes	180
Configure Data Drives on Data Nodes.....	182
About the Author.....	183
Acknowledgements	183

Executive Summary

Digital transformation is generating vast amount of data hence the burgeoning demand to store and process relevant data quickly from big data analytics. Enterprises are exploring new ways to collect data and decipher critical business insight on trends and customer preferences, offer new products and services and explore revenue opportunities. As organizations adopt modern data platforms at larger scale, aspects such as scale, compute, storage network, monitoring, and performance becomes essential as well.

Today, many enterprises are facing the challenge of growing data such as capacity constraint, increased energy cost, infrastructure complexity, and inefficient resource utilization. The rapid increase in the amount of structured, semi-structured and unstructured data being created requires enterprises to find cost-effective ways to store and manage their data. Organizations are embarking on IT modernization to incorporate data driven decision-making architecture to glean intelligence from data science, machine learning, and advanced analytics.

The rapid advancement with artificial intelligence and machine learning has brought new set of challenges for business and IT organization's data strategy when it comes to implementing high performance, scalable and agile fashion cloud scale architecture. Pliable data management architecture focused on data life cycle to store, process, manage, and provide data through Big data technology such as Hadoop and Object storage (Apache Ozone) enabling scalability and cost efficiency.

This CVD implements the [Cisco Data Intelligence Platform](#) on Cisco UCS S3260 storage server with Cloudera Data Platform Data Center (CDP DC) 7.1.1. Modular design of Cisco UCS S3260 Storage Server delivers a highly dense, cost-optimized, on-premises storage with broader infrastructure flexibility for object storage, Hadoop, and Big Data analytics solutions. The computing, storage, and network components can be upgraded independently as technology advances. There is no need to replace the entire server; simply upgrade an individual component. It offers multi-petabyte scalability with low total cost of ownership (TCO) and future proof architecture for big data analytics.

This solution offers cohesive platform for both IT and data scientists by providing a scalable infrastructure for IT while also providing application platform for data scientists.

Solution Overview

Introduction

Both Big Data and machine learning technology have progressed to the point where they are being implemented in production systems running 24x7. There exists a very clear need for a proven, dependable, high-performance platform for the ingestion, processing, storage, and analysis of the data, as well as the seamless dissemination of the output, results, and insights of the analysis.

This solution implements Cloudera Data Platform Data Center (CDP DC) on Cisco UCS Integrated Infrastructure for Big Data and Analytics based on Cisco Data Intelligence Platform (CDIP) architecture, a world-class platform specifically designed for demanding workloads that is both easy to scale and easy to manage, even as the requirements grow to thousands of servers and petabytes of storage.

Many companies, recognizing the immense potential of big data and machine learning technology, are gearing up to leverage these new capabilities, building out departments and increasing hiring. However, these efforts face a new set of challenges:

- Making the data available to the diverse set of people who need it
- Enabling access to high-performance computing resources, GPUs, that also scale with the data growth
- Allowing people to work with the data using the environments in which they are familiar
- Publishing their results so the organization can make use of it
- Enabling the automated production of those results
- Managing the data for compliance and governance
- Scaling the system as the data grows
- Managing and administering the system in an efficient, cost-effective way

This solution is based on the Cisco UCS Integrated Infrastructure for Big Data and Analytics and includes computing, storage, connectivity, and unified management capabilities to help companies manage the immense amount of data being collected. It is built on Cisco Unified Computing System (Cisco UCS) infrastructure, using Cisco UCS 6332 Series Fabric Interconnects, and Cisco UCS C-Series Rack Servers. This architecture is specifically designed for performance and linear scalability for big data and machine learning workload.

Audience

The intended audience of this document includes sales engineers, field consultants, professional services, IT managers, partner engineering and customers who want to deploy the Cloudera Data Platform Data Center on the Cisco UCS Integrated Infrastructure for Big Data and Analytics.

Purpose of this Document

This document describes the architecture, design choices, and deployment procedures for Cisco Data Intelligence Platform using Cloudera Data Platform DC on Cisco UCS S3260 Storage chassis with Cisco UCS S3260 M5 Server Node.

This document also serves as a step-by-step guide on how to deploy CDP DC 7.1.1 on 16 node cluster of Cisco UCS S3260 Storage Server.

What's New in this Release?

This solution extends the portfolio of Cisco Data Intelligence Platform (CDIP) architecture with Cloudera Data Platform Data Center, a state-of-the-art platform, providing a data cloud for demanding workloads that is easy to deploy, scale and manage. Furthermore, as the enterprise's requirements and needs changes overtime, the platform can grow to thousands of servers, hence providing peta bytes of storage.

The following design consideration will be implemented in this validated design:

- Data Lake with Cloudera Data Platform Datacenter on Cisco UCS Integrated Infrastructure for Big Data and Analytics
- Cisco Intersight

What's Next?

This CVD showcases Cisco UCS Manager (UCSM). This solution can also be deployed using Cisco Intersight. Additional Cisco UCS features will be added to the Appendix in the following months. Some of these include the following:

- Cloudera Data Platform Private Cloud
- Apache Ozone – Object Storage
- A fully integrated CDP on CDIP with
 - Data lake enabled through CDP DC
 - AI/ML enabled through CDP Private Cloud
 - Exabyte storage enabled through Apache Ozone

Solution Summary

This CVD details the process of installing Cloudera Data Platform Data Center and the configuration details of the cluster. The current version of Cisco UCS Integrated Infrastructure for Big Data and Analytics offers the following configurations depending on the compute and storage requirements.

Cisco Data Intelligence Platform

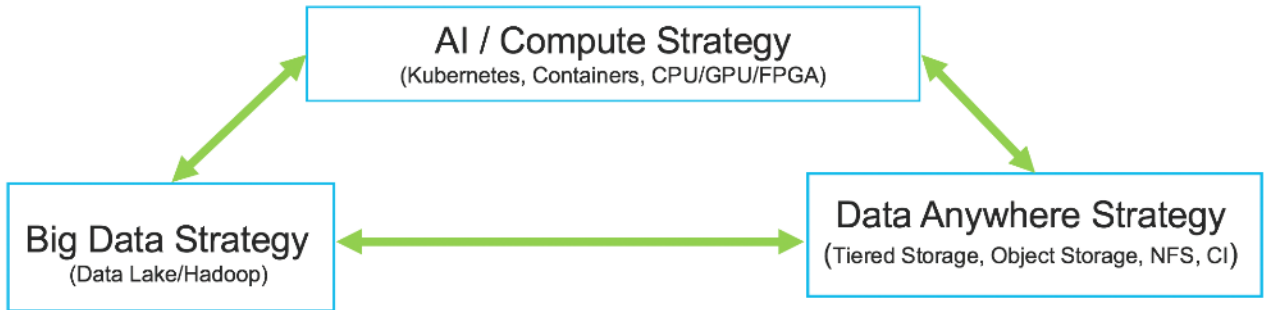
Cisco Data Intelligence Platform (CDIP) is a cloud scale architecture which brings together big data, AI/compute farm, and storage tiers to work together as a single entity while also being able to scale independently to address the IT issues in the modern data center. This architecture allows for:

- Extremely fast data ingest, and data engineering done at the data lake
- AI compute farm allowing for different types of AI frameworks and compute types (GPU, CPU, FPGA) to work on this data for further analytics
- A storage tier, allowing to gradually retire data which has been worked on to a storage dense system with a lower \$/TB providing a better TCO

- Seamlessly scale the architecture to thousands of nodes with a single pane of glass management using Cisco Application Centric Infrastructure (ACI)

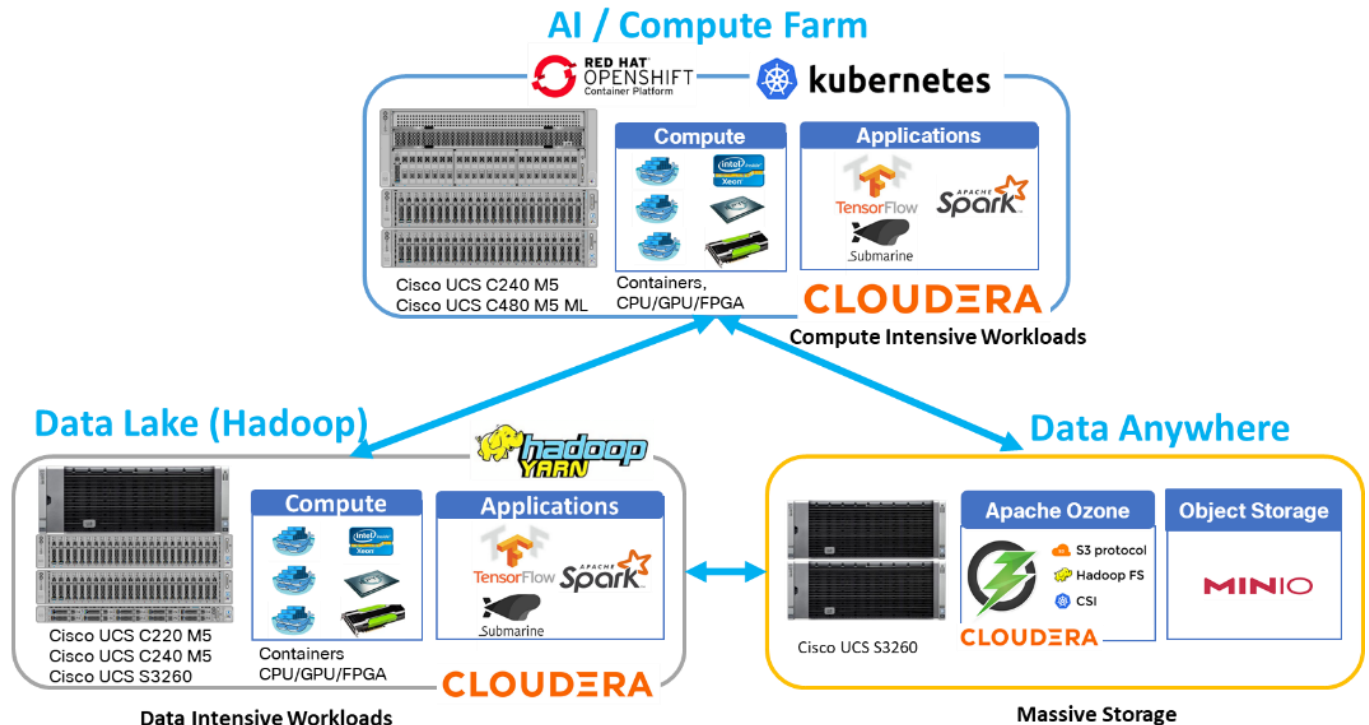
Cisco Data Intelligence Platform caters to the evolving architecture bringing together a fully scalable infrastructure with centralized management and fully supported software stack (in partnership with industry leaders in the space) to each of these three independently scalable components of the architecture including data lake, AI/ML and Object stores.

Figure 1 Cisco Data Intelligent Platform



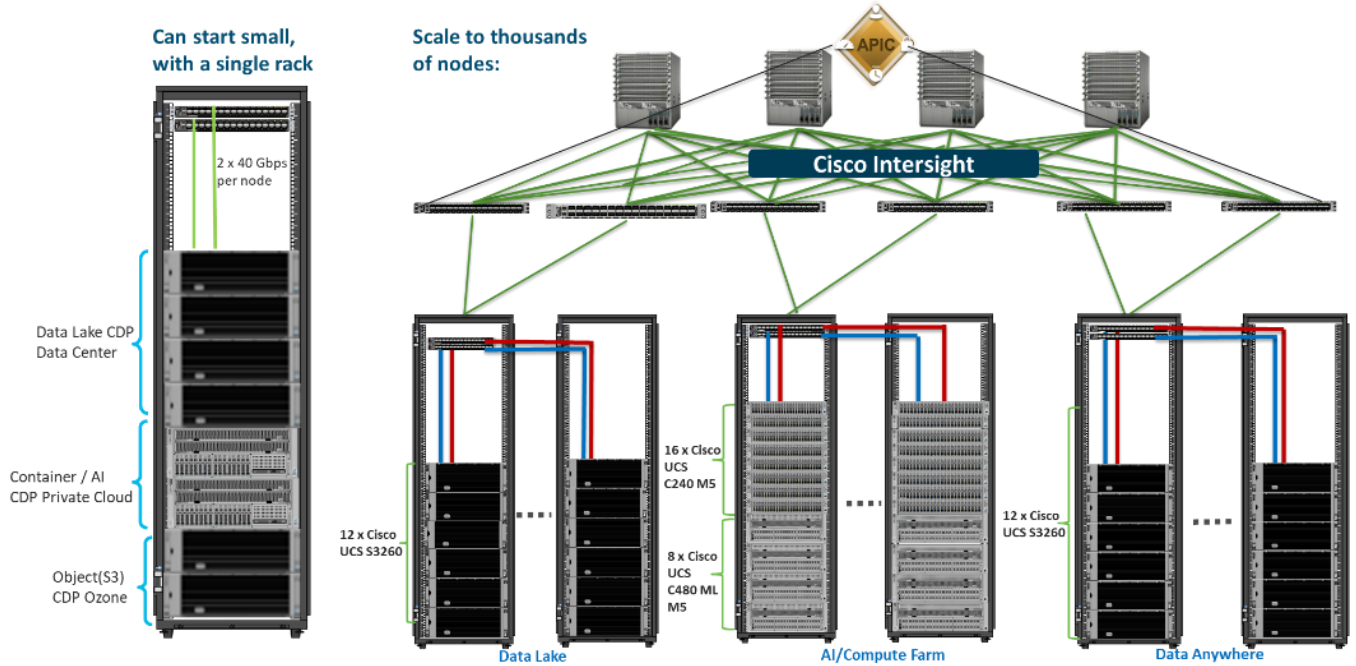
Cisco has developed numerous industry leading Cisco Validated Designs (reference architectures) in the area of Big Data (CVDs with Cloudera, Hortonworks and MapR), compute farm with Kubernetes (CVD with RedHat OpenShift) and Object store (Scality, SwiftStack, Clouidian, and others).

Figure 2 Cisco Data Intelligence Platform with Hadoop, Kubernetes, and Object Store



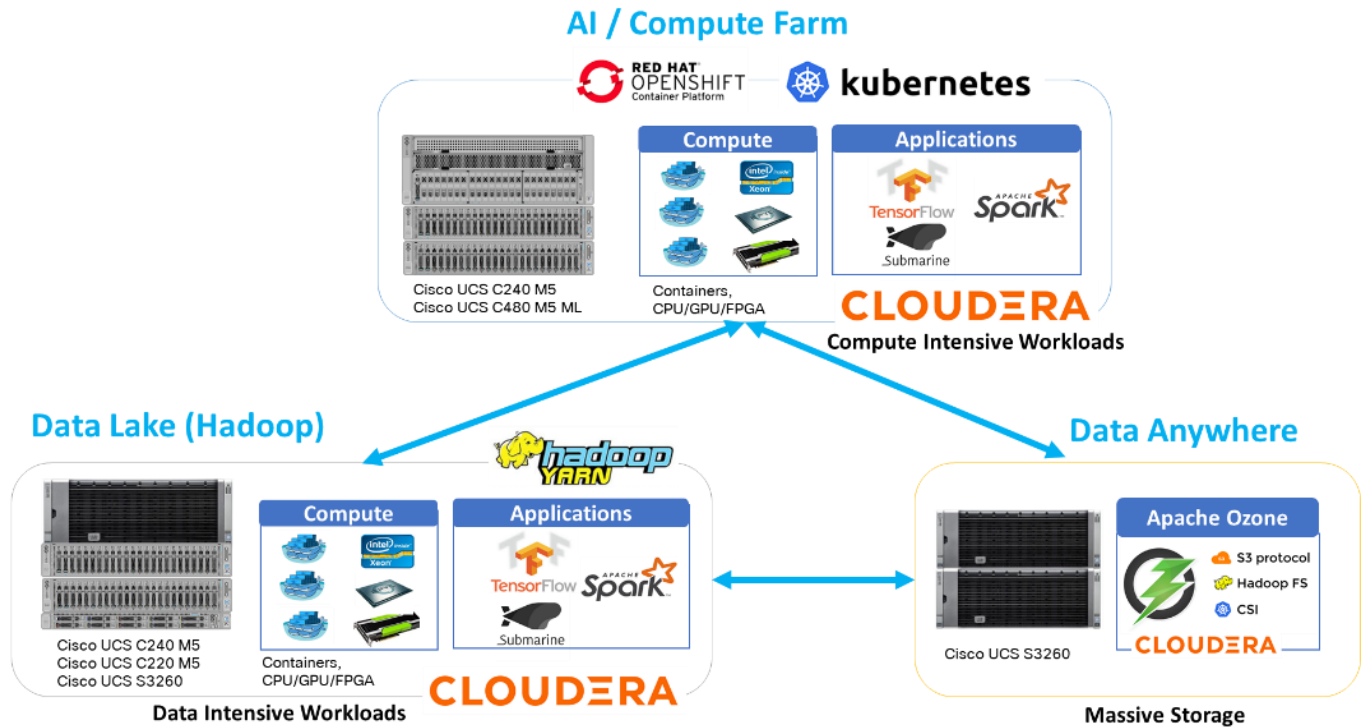
This architecture can start from a single rack and scale to thousands of nodes with a single pane of glass management with Cisco Application Centric Infrastructure (ACI).

Figure 3 Cisco Data Intelligent Platform at Scale



CDP on CDIP

Figure 4 Cloudera Data Platform on Cisco Data Intelligent Platform



A CDIP architecture can fully be enabled by Cloudera Data Platform with the following components:

- Data lake enabled through CDP DC

- AI/ML enabled through CDP Private Cloud and
- Exabyte storage enabled through Apache Ozone

Reference Architecture

Data Lake Reference Architecture

[Table 1](#) lists the data lake reference architecture configuration details for Cisco UCS Integrated Infrastructure for Big Data and Analytics.

Table 1 Cisco UCS Integrated Infrastructure for Big Data and Analytics Configuration Options

	High Performance	Performance	Capacity	High Capacity
Servers	16 x Cisco UCS C220 M5 Rack Servers with small-form-factor (SFF) drives	16 x Cisco UCS C240 M5 Rack Servers with small-form-factor (SFF) drives	16 x Cisco UCS C240 M5 Rack Servers with large-form-factor (LFF) drives	8 x Cisco UCS S3260 Storage Servers
CPU	2 x 2 nd Gen Intel® Xeon® Scalable Processors 6230R (2 x 26 cores, at 2.1 GHz)	2 x 2 nd Gen Intel® Xeon® Scalable Processors 5218R processors (2 x 20 cores, at 2.1 GHz)	2 x 2 nd Gen Intel Xeon Scalable Processors 5218R (2 x 20 cores, at 2.1 GHz)	2 x 2 nd Gen Intel Xeon Scalable Processors 6230R (2 x 26 cores, 2.1 GHz)
Memory	12 x 32GB DDR4 (384 GB)	12 x 32GB DDR4 (384 GB)	12 x 32GB DDR4 (384 GB)	12 x 32GB DDR4 (384 GB)
Boot	M.2 with 2 x 240-GB SSDs	M.2 with 2 x 240-GB SSDs	M.2 with 2 x 240-GB SSDs	2 x 240-GB SATA SSDs
Storage	10 x 8TB 2.5in U.2 Intel P4510 NVMe High Perf. Value Endurance	26 x 2.4TB 10K rpm SFF SAS HDDs or 12 x 1.6-TB Enterprise Value SATA SSDs	12 x 8-TB 7.2K rpm LFF SAS HDDs	28 x 4 TB 7.2K rpm LFF SAS HDDs per server node
Virtual interface card (VIC)	25 Gigabit Ethernet (Cisco UCS VIC 1457) or 40/100 Gigabit Ethernet (Cisco UCS VIC 1497)	25 Gigabit Ethernet (Cisco UCS VIC 1455) or 40/100 Gigabit Ethernet (Cisco UCS VIC 1497)	25 Gigabit Ethernet (Cisco UCS VIC 1455) or 40/100 Gigabit Ethernet (Cisco UCS VIC 1497)	40 Gigabit Ethernet (Cisco UCS VIC 1387) or 25 Gigabit Ethernet (Cisco UCS VIC 1455) or 40/100 Gigabit Ethernet (Cisco UCS VIC 1495)
Storage controller	NVMe Switch	Cisco 12-Gbps SAS modular RAID controller with 4-GB flash-based write cache (FBWC) or Cisco 12-Gbps modular SAS host bus adapter (HBA)	Cisco 12-Gbps SAS modular RAID controller with 2-GB FBWC or Cisco 12-Gbps modular SAS host bus adapter (HBA)	Cisco 12-Gbps SAS Modular RAID Controller with 4-GB flash-based write cache (FBWC)
Network connectivity	Cisco UCS 6332 Fabric Interconnect or Cisco UCS 6454/64108 Fabric Interconnect	Cisco UCS 6332 Fabric Interconnect or Cisco UCS 6454/64108 Fabric Interconnect	Cisco UCS 6332 Fabric Interconnect or Cisco UCS 6454/64108 Fabric Interconnect	Cisco UCS 6332 Fabric Interconnect or Cisco UCS 6454/64108 Fabric Interconnect

	High Performance	Performance	Capacity	High Capacity
GPU (optional)	Up to 2 x NVIDIA Tesla T4 with 16 GB memory each	Up to 2 x NVIDIA Tesla V100 with 32 GB memory each Or Up to 6 x NVIDIA Tesla T4 with 16 GB memory each	2 x NVIDIA Tesla V100 with 32 GB memory each Or Up to 6 x NVIDIA Tesla T4 with 16 GB memory each	



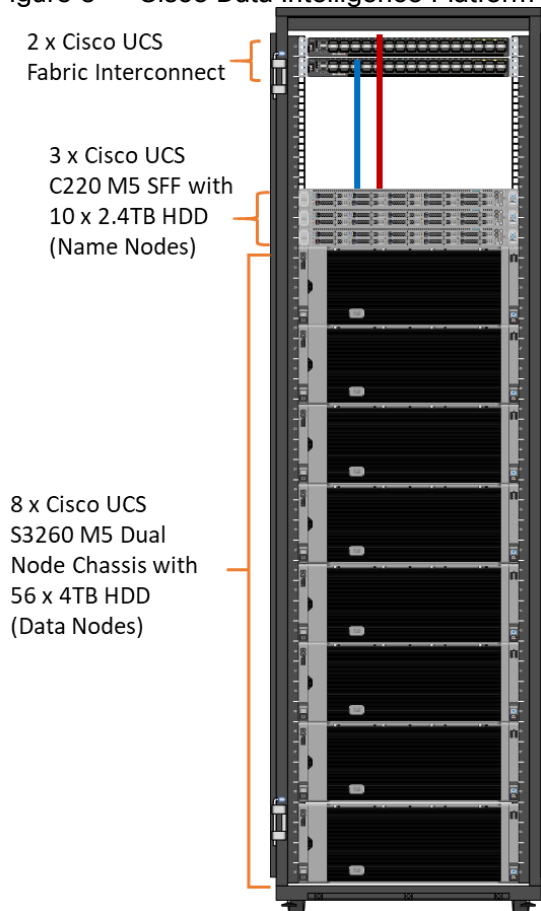
This configuration can also be deployed with the 4th Generation Cisco UCS 6454 Fabric Interconnect with PCIe slot based with choice of Cisco UCS VIC 1455 Quad Port 10/25G for Cisco UCS S3260 M5. However, this could lead to a performance slow down compared to a 40G VIC and Fabric Interconnect 6332.



Cisco UCS VIC 1387 provides 40Gbps, VIC 1457 provides 10/25Gbps and VIC 1497 provides 40/100Gbps connectivity for Cisco UCS C-series rack server. For more information see, [Cisco UCS C-Series Servers Managing Network Adapters](#)

As illustrated in [Figure 5](#), a 16-node cluster with Rack#1 hosting 8 Cisco UCS S3260 Storage Server chassis with dual S3260 M5 Server Node per chassis. Each link in the figure represents a 40 Gigabit Ethernet link from each of the sixteen servers directly connected to a Fabric Interconnect. Every server is connected to both Fabric Interconnects.

Figure 5 Cisco Data Intelligence Platform with Cloudera Data Platform Data Center – Data Lake



Scaling the Solution

[Figure 6](#) illustrates how to scale the solution. Each pair of Cisco UCS 6332 Fabric Interconnects has 24 Cisco UCS C240 M5 servers connected to it. This allows for eight uplinks from each Fabric Interconnect to the Cisco Nexus 9332 switch. Six pairs of Cisco UCS 6332 FI's can connect to a single switch with four uplink ports each. With 24 servers per FI, a total of 144 servers can be supported. Additionally, this solution can scale to thousands of nodes with the Cisco Nexus 9500 series family of switches.

In the reference architectures discussed here, each of the components is scaled separately, and for the purposes of this example, scaling is uniform. Two scale scenarios are as follows:

- Scaled architecture with 3:1 oversubscription with Cisco fabric interconnects and Cisco ACI
- Scaled architecture with 2:1 oversubscription with Cisco ACI

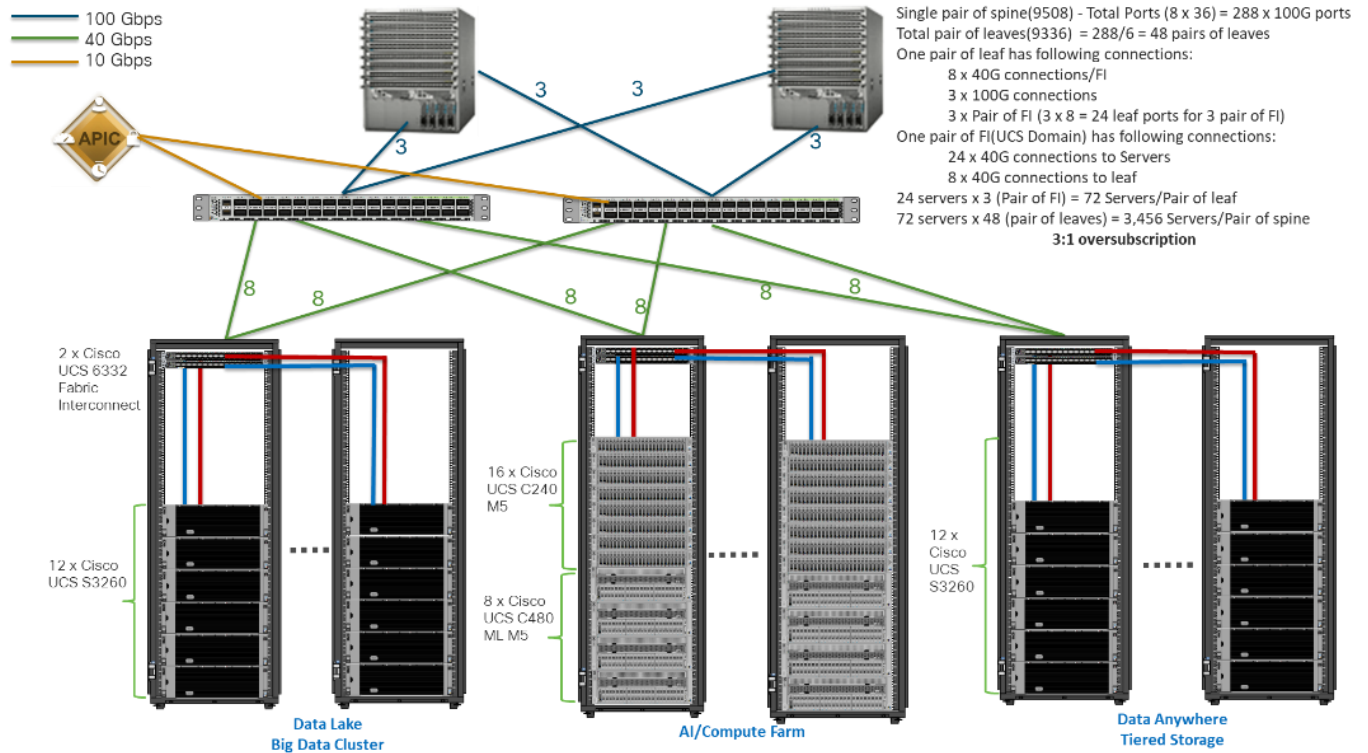
In the following scenarios, the goal is to populate up to a maximum of 200 leaf nodes in a Cisco ACI domain. Not all cases reach that number because they use the Cisco Nexus® 9508 Switch for this sizing and not the Cisco Nexus 9516 Switch.

Scaled Architecture with 3:1 Oversubscription with Cisco Fabric Interconnects and Cisco ACI

The architecture discussed here and shown in [Figure 6](#) supports 3:1 network oversubscription from every node to every other node across a multidomain cluster (nodes in a single domain within a pair of Cisco fabric interconnects are locally switched and not oversubscribed).

From the viewpoint of the data lake, 24 Cisco UCS C240 M5 Rack Servers are connected to a pair of Cisco UCS 6332 Fabric Interconnects (with 24 x 40-Gbps throughput). From each fabric interconnect, 8 x 40-Gbps links connect to a pair of Cisco Nexus 9336 Switches. Three pairs of fabric interconnects can connect to a single pair of Cisco Nexus 9336 Switches (8 x 40-Gbps links per Fabric Interconnect to a pair of Nexus switch). Each of these Cisco Nexus 9336 Switches connects to a pair of Cisco Nexus 9508 Cisco ACI switches with 6 x 100-Gbps uplinks (connecting to a Cisco N9K-X9736C-FX line card). the Cisco Nexus 9508 Switch with the Cisco N9K-X9736C-FX line card can support up to 36 x 100-Gbps ports, each and 8 such line cards.

Figure 6 Scaled Architecture with 3:1 Oversubscription with Cisco Fabric Interconnects and Cisco ACI



Scaled Architecture with 2:1 Oversubscription with Cisco ACI

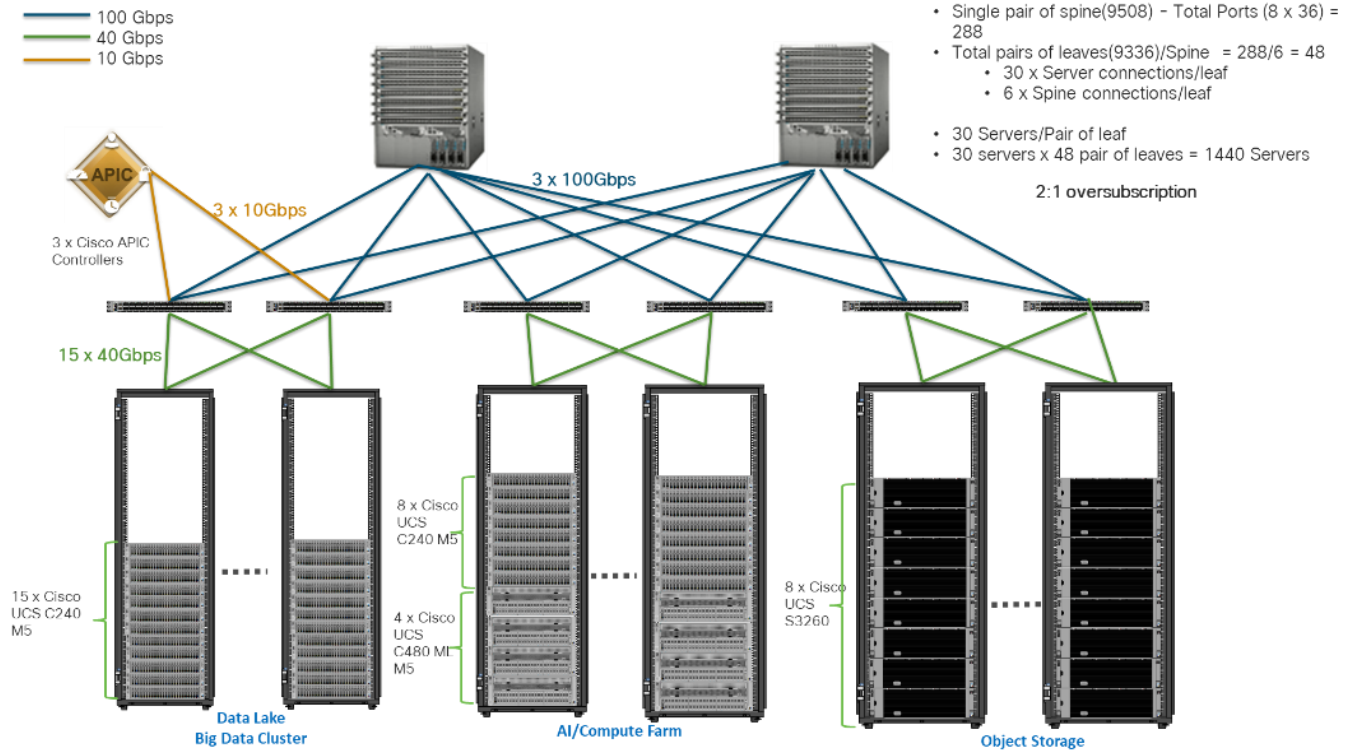
In the scenario discussed here and shown in [Figure 7](#), the Cisco Nexus 9508 Switch with the Cisco N9K-X9736C-FX line card can support up to 36 x 100-Gbps ports, each and 8 such line cards.

Here, for the 2:1 oversubscription, 30 Cisco UCS C240 M5 Rack Servers are connected to a pair of Cisco Nexus 9336 Switches, and each Cisco Nexus 9336 connects to a pair of Cisco Nexus 9508 Switches with three uplinks each. A pair of Cisco Nexus 9336 Switches can support 30 servers and connect to a spine with 6 x 100-Gbps links on each spine. This single pod (pair of Cisco Nexus 9336 Switches connecting to 30 Cisco UCS C240 M5 servers and 6 uplinks to each spine) can be repeated 48 times (288/6) for a given Cisco Nexus 9508 Switch and can support up to 1440 servers.

To reduce the oversubscription ratio (to get 1:1 network subscription from any node to any node), you can use just 15 servers under a pair of Cisco Nexus 9336 Switches and then move to Cisco Nexus 9516 Switches (the number of leaf nodes would double).

To scale beyond this number, multiple spines can be aggregated.

Figure 7 Scaled Architecture with 2:1 Oversubscription with Cisco ACI



Technology Overview

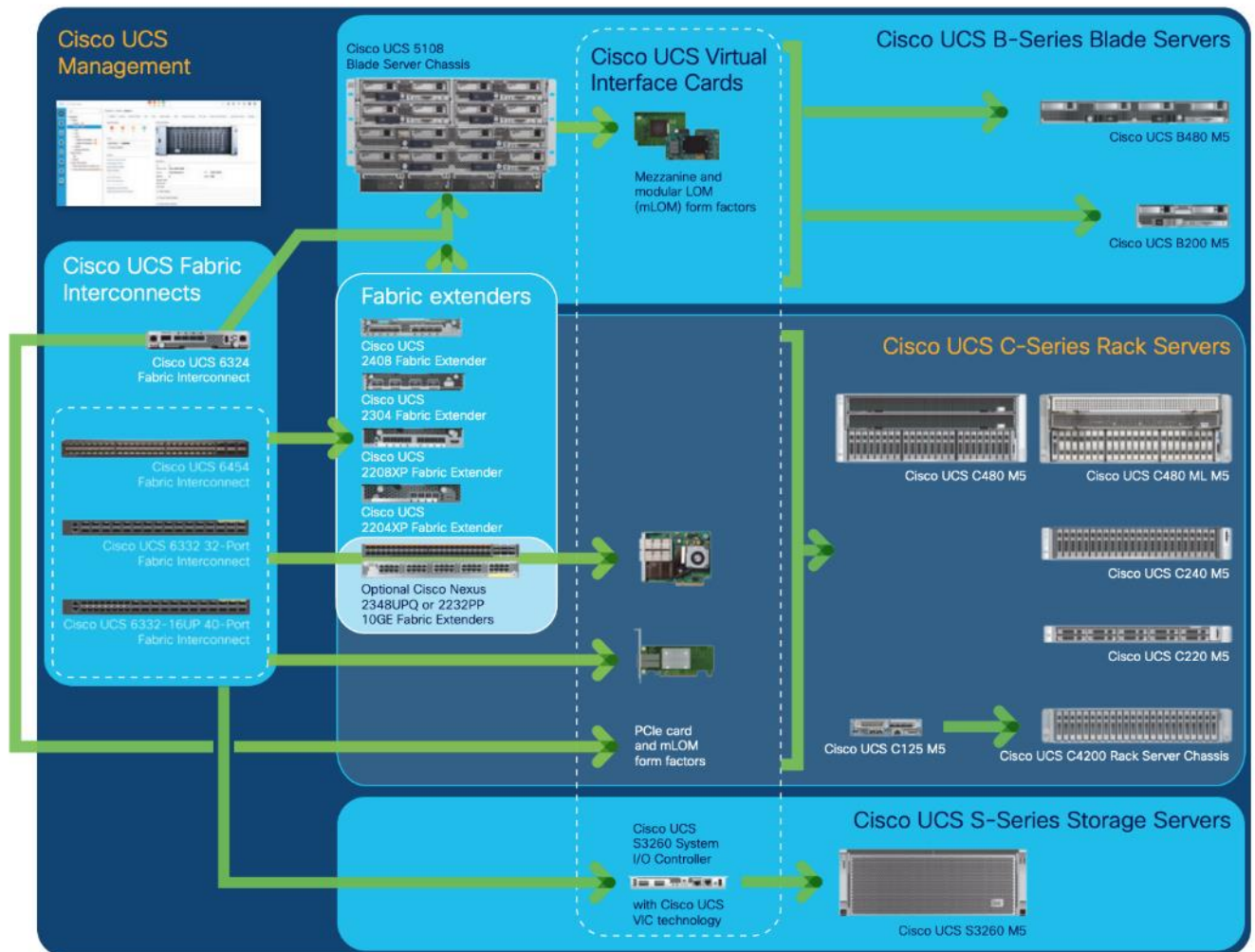
Cisco UCS Integrated Infrastructure for Big Data and Analytics

The Cisco UCS Integrated Infrastructure for Big Data and Analytics solution for Cloudera is based on [Cisco UCS Integrated Infrastructure for Big Data and Analytics](#), a highly scalable architecture designed to meet a variety of scale-out application demands with seamless data integration and management integration capabilities built using the components described in this section.

Cisco UCS

Cisco Unified Computing System™ (Cisco UCS®) is a next-generation data center platform that unites computing, networking, storage access, and virtualization resources into a cohesive system designed to reduce Total Cost of Ownership (TCO) and increase business agility. The system integrates a low-latency, lossless 10/25/40/100 Gigabit Ethernet unified network fabric with enterprise-class, x86-architecture servers. The system is an integrated, scalable, multi-chassis platform in which all resources participate in a unified management domain ([Figure 8](#)).

Figure 8 Cisco UCS Component Hierarchy



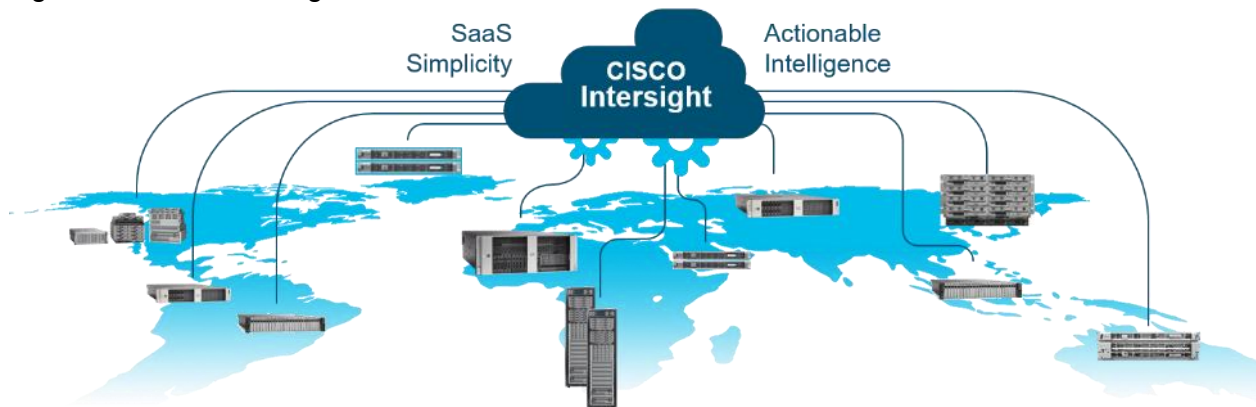
Cisco Intersight

Cisco Intersight is Cisco's systems management platform that delivers intuitive computing through cloud-powered intelligence. This platform offers a more intelligent level of management that enables IT organizations to analyze, simplify, and automate their environments in ways that were not possible with prior generations of tools. This capability empowers organizations to achieve significant savings in Total Cost of Ownership (TCO) and to deliver applications faster, so they can support new business initiatives.

Cisco Intersight is a Software as a Service (SaaS) infrastructure management which provides a single pane of glass management of CDIP infrastructure in the data center. Cisco Intersight scales easily, and frequent updates are implemented without impact to operations. Cisco Intersight Essentials enables customers to centralize configuration management through a unified policy engine, determine compliance with the Cisco UCS Hardware Compatibility List (HCL), and initiate firmware updates. Enhanced capabilities and tight integration with Cisco TAC enables more efficient support. Cisco Intersight automates uploading files to speed troubleshooting. The Intersight recommendation engine provides actionable intelligence for IT operations management. The insights are driven by expert systems and best practices from Cisco.

Cisco Intersight offers flexible deployment either as Software as a Service (SaaS) on Intersight.com or running on your premises with the Cisco Intersight virtual appliance. The virtual appliance provides users with the benefits of Cisco Intersight while allowing more flexibility for those with additional data locality and security requirements.

Figure 9 Cisco Intersight



Cisco Intersight has the following:

- Connected TAC
- Security Advisories
- Hardware Compatibility List (HCL) and much more

To learn more about all the features of Intersight go to: <https://www.cisco.com/c/en/us/products/servers-unified-computing/intersight/index.html>

Cisco UCS Manager

Cisco UCS Manager (UCSM) resides within the Cisco UCS Fabric Interconnect. It makes the system self-aware and self-integrating, managing all the system components as a single logical entity. Cisco UCS Manager can be accessed through an intuitive graphical user interface (GUI), a command-line interface (CLI), or an XML application-programming interface (API). Cisco UCS Manager uses service profiles to define the personality, configuration, and connectivity of all resources within Cisco UCS, radically simplifying provisioning of resources so that the process takes minutes instead of days. This simplification allows IT departments to shift their focus from constant maintenance to strategic business initiatives.

Key Features

- Supports Cisco UCS B-Series Blade and Cisco UCS C-Series Rack Servers, the Cisco UCS C3260 storage server, Cisco UCS Mini, and the Cisco HyperFlex hyperconverged infrastructure.
- Programmatically controls server, network, and storage resources, with a unified, policy-driven management, so they can be efficiently managed at scale through software.
- Works with HTML 5, Java, or CLI graphical user interfaces.
- Can automatically detect, inventory, manage, and provision system components that are added or changed.
- Facilitates integration with third-party systems management tools.
- Builds on existing skills and supports collaboration across disciplines through role-based administration

Cisco UCS 6300 Series Fabric Interconnects

Cisco UCS 6300 Series Fabric Interconnects provide high-bandwidth, low-latency connectivity for servers, with integrated, unified management provided for all connected devices by Cisco UCS Manager. Deployed in redundant pairs, Cisco fabric interconnects offer the full active-active redundancy, performance, and exceptional scalability needed to support the large number of nodes that are typical in clusters serving big data applications. Cisco UCS Manager enables rapid and consistent server configuration using service profiles, automating ongoing system maintenance activities such as firmware updates across the entire cluster as a single operation. Cisco UCS Manager also offers advanced monitoring with options to raise alarms and send notifications about the health of the entire cluster.

The Cisco UCS 6300 Series Fabric Interconnects are a core part of Cisco UCS, providing low-latency, lossless 10 and 40 Gigabit Ethernet, Fiber Channel over Ethernet (FCoE), and Fiber Channel functions with management capabilities for the entire system. All servers attached to Fabric interconnects become part of a single, highly available management domain.

Figure 10 Cisco UCS 6332UP 32 -Port Fabric Interconnect



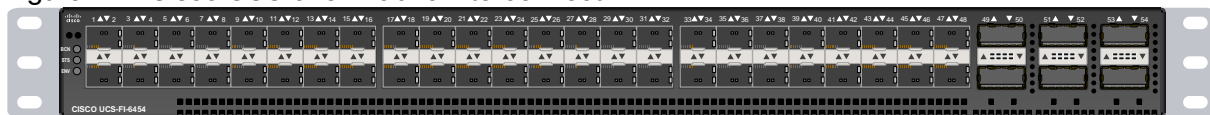
For more information, go to: <https://www.cisco.com/c/en/us/products/collateral/servers-unified-computing/ucs-6300-series-fabric-interconnects/datasheet-c78-736682.html?cachemode=refresh>

Cisco UCS 6400 Series Fabric Interconnect

The Cisco UCS 6400 Series Fabric Interconnects are a core part of the Cisco Unified Computing System, providing both network connectivity and management capabilities for the system (Figure 2). The Cisco UCS 6400 Series offer line-rate, low-latency, lossless 10/25/40/100 Gigabit Ethernet, Fibre Channel over Ethernet (FCoE), and Fibre Channel functions. (Figure 11 and Figure 12).

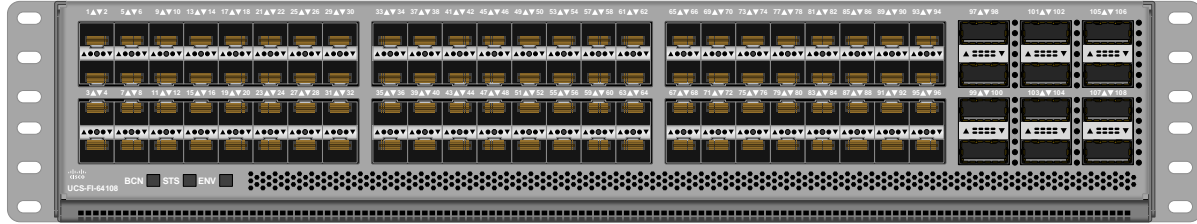
The Cisco UCS 6454 54-Port Fabric Interconnect (Figure 11) is a One-Rack-Unit (1RU) 10/25/40/100 Gigabit Ethernet, FCoE, and Fibre Channel switch offering up to 3.82 Tbps throughput and up to 54 ports. The switch has 28 10/25-Gbps Ethernet ports, 4 1/10/25-Gbps Ethernet ports, 6 40/100-Gbps Ethernet uplink ports, and 16 unified ports that can support 10/25-Gbps Ethernet ports or 8/16/32-Gbps Fibre Channel ports. All Ethernet ports are capable of supporting FCoE.

Figure 11 Cisco UCS 6454 Fabric Interconnect



The Cisco UCS 64108 Fabric Interconnect (Figure 12) is a 2-RU top-of-rack switch that mounts in a standard 19-inch rack such as the Cisco R Series rack. The 64108 is a 10/25/40/100 Gigabit Ethernet, FCoE and Fiber Channel switch offering up to 7.42 Tbps throughput and up to 108 ports. The switch has 16 unified ports (port numbers 1-16) that can support 10/25-Gbps SFP28 Ethernet ports or 8/16/32-Gbps Fibre Channel ports, 72 10/25-Gbps Ethernet SFP28 ports (port numbers 17-88), 8 1/10/25-Gbps Ethernet SFP28 ports (port numbers 89-96), and 12 40/100-Gbps Ethernet QSFP28 uplink ports (port numbers 97-108). All Ethernet ports are capable of supporting FCoE.

Figure 12 Cisco UCS 64108 Fabric Interconnect



Cisco UCS C-Series Rack-Mount Servers

Cisco UCS C-Series Rack-Mount Servers keep pace with Intel Xeon processor innovation by offering the latest processors with increased processor frequency and improved security and availability features. With the increased performance provided by the Intel Xeon Scalable Family Processors, Cisco UCS C-Series servers offer an improved price-to-performance ratio. They also extend Cisco UCS innovations to an industry-standard rack-mount form factor, including a standards-based unified network fabric, Cisco VN-Link virtualization support, and Cisco Extended Memory Technology.

It is designed to operate both in standalone environments and as part of Cisco UCS managed configuration, these servers enable organizations to deploy systems incrementally—using as many or as few servers as needed—on a schedule that best meets the organization’s timing and budget. Cisco UCS C-Series servers offer investment protection through the capability to deploy them either as standalone servers or as part of Cisco UCS. One compelling reason that many organizations prefer rack-mount servers is the wide range of I/O options available in the form of PCIe adapters. Cisco UCS C-Series servers support a broad range of I/O options, including interfaces supported by Cisco and adapters from third parties.

Cisco UCS S3260 Storage Servers

The Cisco UCS S3260 Storage Server is a modular storage server with dual M5 server nodes and is optimized to deliver efficient, industry-leading storage for data-intensive workloads. The Cisco UCS S3260 server with dual-node capability that is based on the 2nd Gen Intel® Xeon® Scalable and Intel® Xeon® Scalable processors, the server features up to 840 TB of local storage in a compact 4-Rack-Unit (4RU) form factor. The drives can be configured with enterprise-class Redundant Array of Independent Disks (RAID) redundancy or with a pass-through Host Bus Adapter (HBA) controller. Network connectivity is provided with dual-port 40-Gbps nodes in each server, with expanded unified I/O capabilities for data migration between Network-Attached Storage (NAS) and SAN environments. This storage-optimized server comfortably fits in a standard 32-inch-depth rack, such as the Cisco® R 42610 Rack.

Figure 13 Cisco UCS S3260 Storage Server



The Cisco UCS S3260 Storage Server chassis has 56 top-load LFF HDDs option as shown above with a maximum capacity of 4 TB per HDD and can be mixed with up to 28 SSDs.

The modular Cisco UCS S3260 Storage Server chassis offers flexibility with more computing, storage, and PCIe expansion on the second slot in the chassis. This second slot can be used for:

- An additional server node
- Four additional LFF HDDs with up to 10 TB capacity per HDD
- New PCIe expansion tray with up to two x8 half-height, half-width PCIe slots that can use any industry-standard PCIe card including Fibre Channel and Ethernet cards.

The Cisco UCS S3260 Storage Server Chassis includes a Cisco UCS Virtual Interface Card (VIC) 1300 platform chip onboard the system I/O controller, offering high-performance bandwidth with dual-port 40 Gigabit Ethernet and FCoE interfaces per system I/O controller.

Figure 14 Cisco UCS S3260 Storage Server: Rear View



Cisco UCS Virtual Interface Cards (VICs)

Cisco UCS VIC 1387

Cisco UCS Virtual Interface Cards (VIC) are unique to Cisco. Cisco UCS Virtual Interface Cards incorporate next-generation converged network adapter (CNA) technology from Cisco and offer dual 10- and 40-Gbps ports designed for use with Cisco UCS servers. Optimized for virtualized networking, these cards deliver high performance and bandwidth utilization, and support up to 256 virtual devices.

The Cisco UCS Virtual Interface Card 1387 ([Figure 15](#)) offers dual-port Enhanced Quad Small Form-Factor Pluggable (QSFP+) 40 Gigabit Ethernet and Fiber Channel over Ethernet (FCoE) in a modular-LAN-on-motherboard (mLOM) form factor. The mLOM slot can be used to install a Cisco VIC without consuming a PCIe slot providing greater I/O expandability.

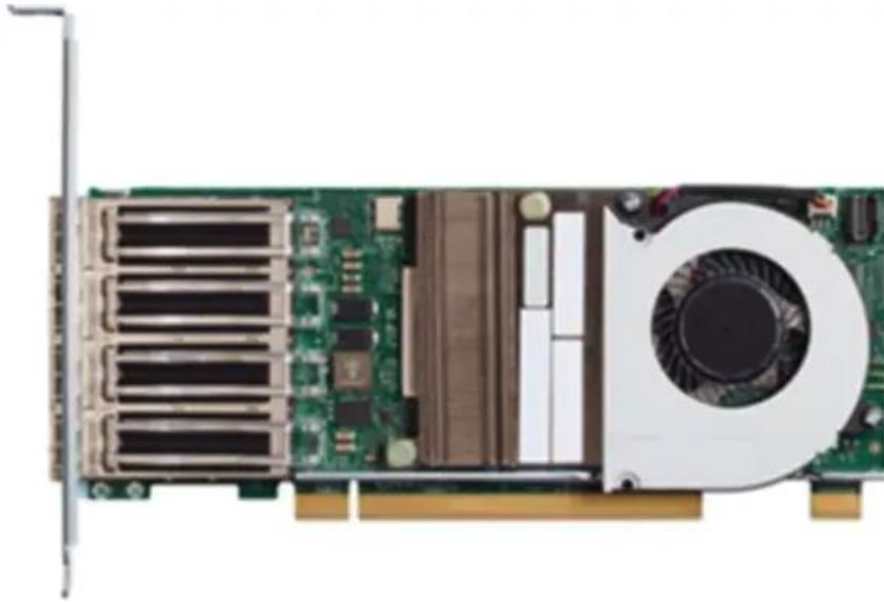
Figure 15 Cisco UCS VIC 1387



Cisco UCS VIC 1455

The Cisco UCS VIC 1455 ([Figure 16](#)) Cisco 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.

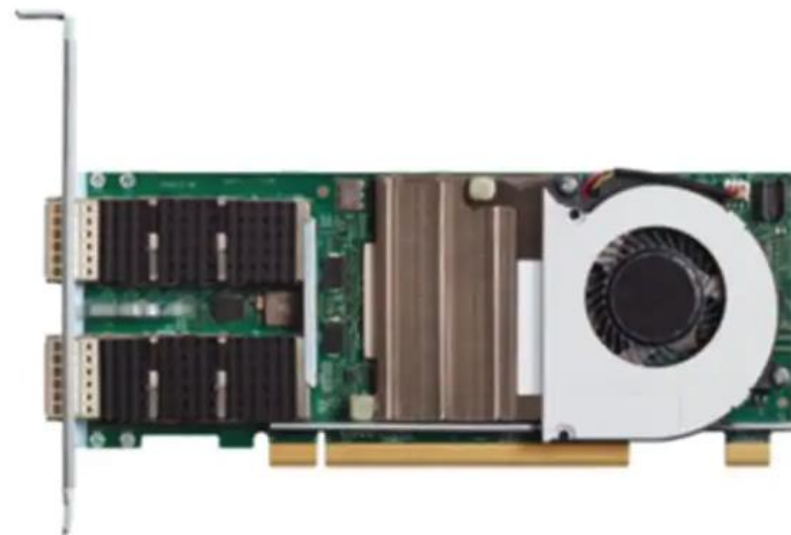
Figure 16 Cisco UCS VIC 1455



Cisco UCS VIC 1495

The Cisco VIC 1495 is a dual-port Small Form-Factor (QSFP28) half-height PCIe card designed for the M5 generation of Cisco UCS C-Series Rack Servers. The card supports 40/100-Gbps Ethernet and FCoE. The card can present PCIe standards-compliant interfaces to the host, and these can be dynamically configured as NICs and HBAs.

Figure 17 Cisco UCS VIC 1495



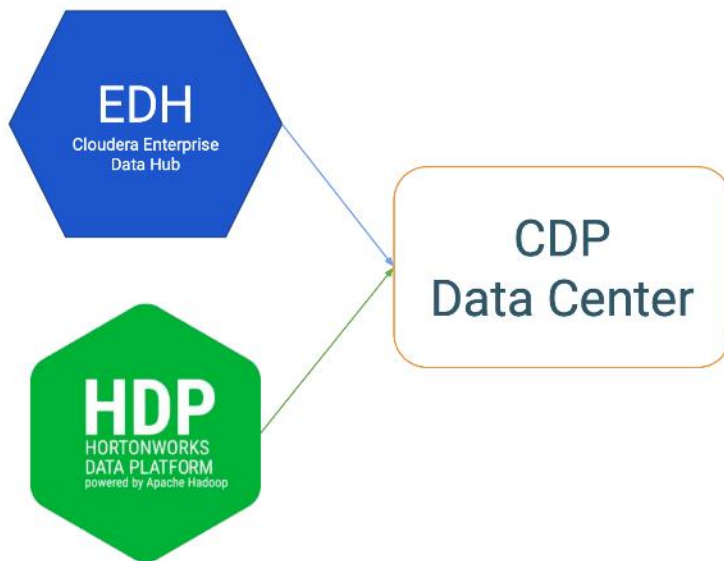
For more information, go to the Cisco UCS S3260 Storage Server datasheet and select the System I/O Controller section: <https://www.cisco.com/c/dam/en/us/products/collateral/servers-unified-computing/ucs-s-series-storage-servers/s3260-specsheet.pdf>

Cloudera Data Platform (CDP) Data Center

CDP is an integrated data platform that is easy to deploy, manage, and use. By simplifying operations, CDP reduces the time to onboard new use cases across the organization. It uses machine learning to intelligently auto scale workloads up and down for more cost-effective use of cloud infrastructure.

Cloudera Data Platform (CDP) Data Center is the on-premises version of Cloudera Data Platform. This new product combines the best of both worlds i.e. Cloudera Enterprise Data Hub and Hortonworks Data Platform Enterprise along with new features and enhancements across the stack. This unified distribution is a scalable and customizable platform where you can securely run many types of workloads.

Figure 18 Cloudera Data Platform – Unity Release



Cloudera Data Platform provides:

- **Unified Distribution:** Whether you are coming from CDH or HDP, CDP caters both. It offers richer feature sets and bug fixes with concentrated development and higher velocity.
- **Hybrid & On-prem:** Hybrid and multi-cloud experience, on-prem it offers best performance, cost, and security. It is designed for data centers with optimal infrastructure.
- **Management:** It provides consistent management and control points for deployments.
- **Consistency:** Security and governance policies can be configured once and applied across all data and workloads.
- **Portability:** Policies stickiness with data, even if it moves across all supported infrastructure.

Apache Ozone (Tech preview)

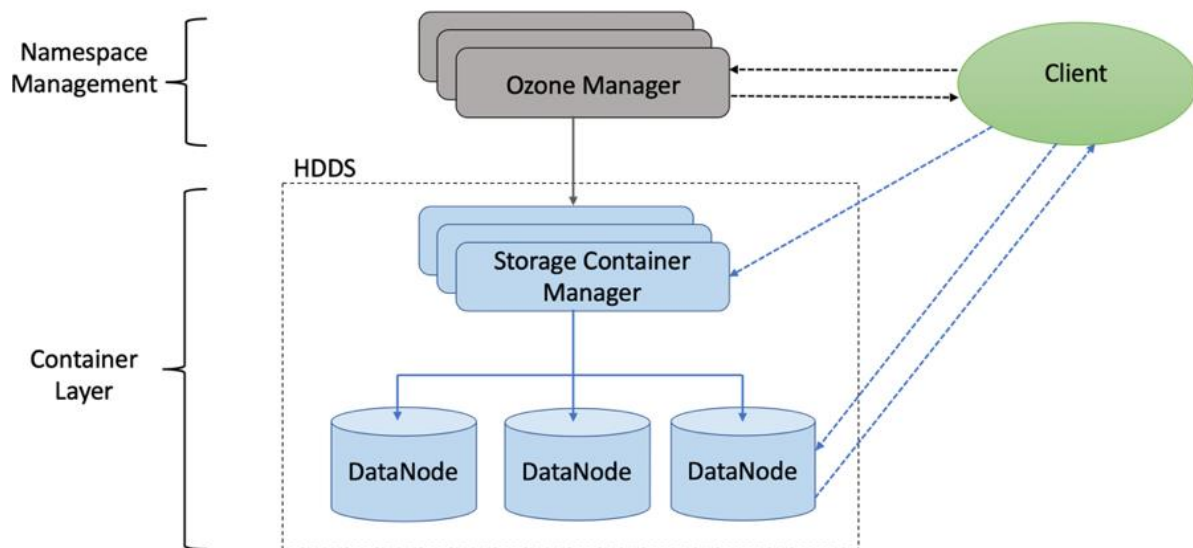
Ozone is a scalable, redundant, and distributed object store optimized for big data workloads. Apart from scaling to billions of objects of varying sizes, Ozone can function effectively in containerized environments such as Kubernetes and YARN.

Ozone consists of three important storage elements: volumes, buckets, and keys. Each key is part of a bucket, which, in turn, belongs to a volume. Only an administrator can create volumes. Depending on their requirements, users can create buckets in volumes. Ozone stores data as keys inside these buckets.

When a key is written to Ozone, the associated data is stored on the DataNodes in chunks called blocks. Therefore, each key is associated with one or more blocks. Within the DataNodes, a series of unrelated blocks is stored in a container, allowing many blocks to be managed as a single entity.

Ozone separates management of namespaces and storage, helping it to scale effectively. Ozone Manager manages the namespaces while Storage Container Manager handles the containers.

Figure 19 Basic Architecture for Ozone



Ozone is available for technical preview and considered to be under development. Do not use this component in your production systems.

Red Hat Ansible Automation

This solution uses Red Hat Ansible Automation for all pre and post deployment steps for automating repeatable tasks to maintain consistency.

Red Hat Ansible Automation is a powerful IT automation tool. It is capable of provisioning numerous types of resources and deploying applications. It can configure and manage devices and operating system components. Due to its simplicity, extensibility, and portability, this solution extensively utilizes Ansible for performing repetitive deployment steps across the nodes.



For more information about Ansible, go to:

<https://www.redhat.com/en/technologies/management/ansible>

Solution Design

Requirements

This CVD describes architecture and deployment procedures for Cloudera Data Platform Data Center on a 16-node cluster based on Cisco UCS Integrated Infrastructure for Big Data and Analytics. The solution provides the details to configure CDP DC on the infrastructure.

The cluster configuration consists of the following:

- 2 Cisco UCS 6332UP Fabric Interconnects
- 8 Cisco UCS S3260 Storage servers with dual Cisco UCS S3260 Server Node per chassis
- 2 Cisco R42610 standard racks
- 4 Vertical Power distribution units (PDUs) (Country Specific)

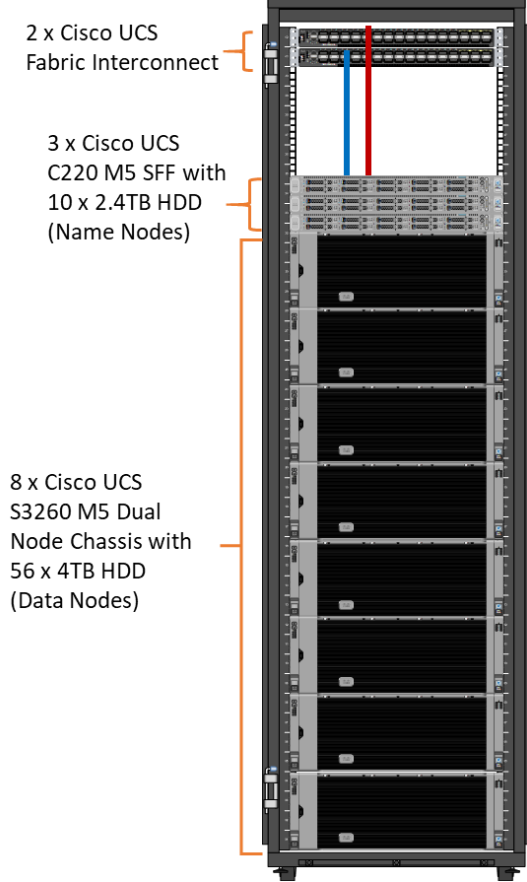
Physical Topology

Each rack consists of two vertical PDUs. The first rack consists of two Cisco UCS 6332UP Fabric Interconnects, sixteen Cisco UCS C240 M5 Rack Servers connected to each of the vertical PDUs for redundancy; thereby, ensuring availability during power source failure. The second rack consists of twelve Cisco UCS C240 M5 Servers connected to each of the vertical PDUs for redundancy; thereby, ensuring availability during power source failure, like the first rack. [Figure 20](#) represents a 40 Gigabit Ethernet link from each server is connected to both Fabric Interconnects



Please contact your Cisco representative for country-specific information.

Figure 20 Cisco Data Intelligence Platform - 16 Node Configuration with CDP DC



Port Configuration on Fabric Interconnect

[Table 2](#) lists the port configuration on Cisco UCS FI 6332 Fabric Interconnect.

Table 2 Port Configuration on Fabric Interconnect

Port Type	Port Number
Server	1-16
Network	29-32

Server Configuration and Cabling for Cisco UCS S3260

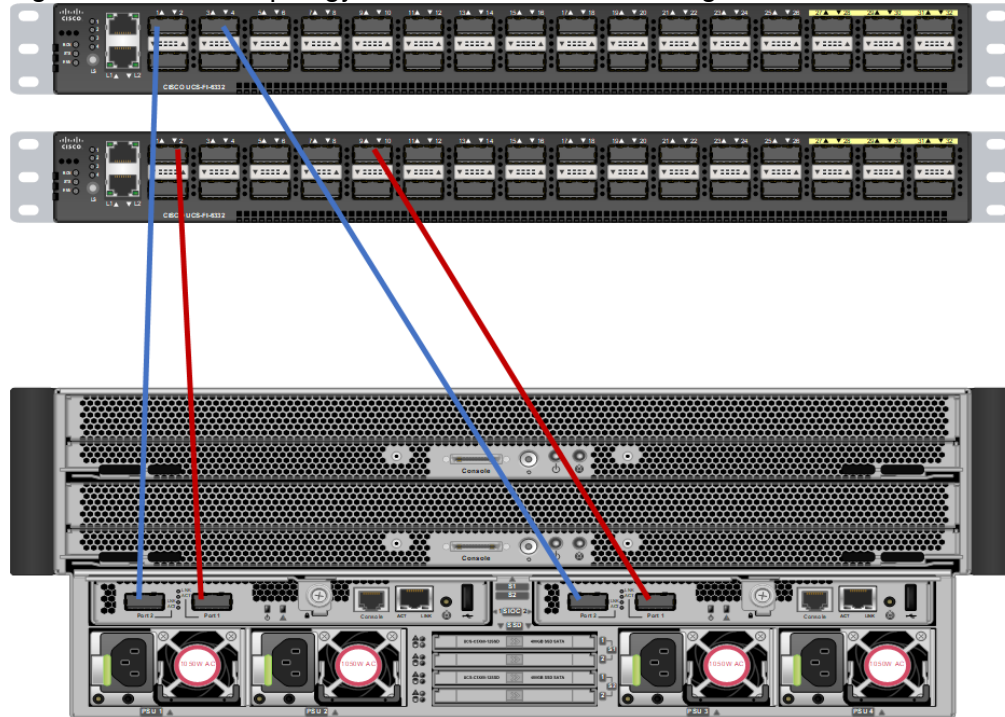
The Cisco UCS S3260 M5 Server Node is equipped with 2 x 2nd Gen Intel Xeon Scalable Family Processor 6230R (2 x 26 cores, 2.1 GHz), 384 GB of memory (12 x 32GB @ 2933MHz), SIOC with VIC 1300 included, Cisco 12-Gbps SAS Modular Raid Controller with 4-GB FBWC, 28 x 4 TB 10K rpm SFF SAS HDDs or 12 x 1.6 TB Enterprise Value SATA SSDs, M.2 with 2 x 240-GB SSDs for Boot.

[Figure 21](#) illustrates the port connectivity between the Cisco UCS FI 6332 and Cisco UCS C240 M5 Rack Server. 8 Cisco UCS S3260 Storage Servers are installed in this configuration.

For information on physical connectivity and single-wire management, go to:

https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/c-series_integration/ucsm4-0/b_C-Series-Integration_UCSM4-0/b_C-Series-Integration_UCSM4-0_chapter_01.html

Figure 21 Fabric Topology for Cisco UCS S3260 Storage Server



Software Distributions and Firmware Versions

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

Table 3 Software Distribution and Version

Layer	Component	Version or Release
Compute	Cisco UCS S3260	S3X60M5.4.1.1b
Network	Cisco UCS 6332	4.1(1a)
	System IO Controller with VIC 1380 included	4.1(1a)C
Storage	Cisco UCS S3260 Dual Raid controller	29.00.1-0358
	LSI MegaRAID SAS Driver	07.708.03.00
	Red Hat Enterprise Linux Server	7.7
Software	Cisco UCS Manager	4.1(1a)

Layer	Component	Version or Release
	Cloudera CDP DC	7.1.1
	Hadoop	3.1.2
	Spark	2.4.0



The latest drivers can be downloaded from the link below [https://software.cisco.com/download/home/283862063/type/283853158/release/4.1\(1a\)](https://software.cisco.com/download/home/283862063/type/283853158/release/4.1(1a))



Support for Intel second generation scalable family processor added in UCSM version 4.0.4a.

Cisco Intersight

Cisco Intersight provides following features for ease of operations and administrator to the IT staff.

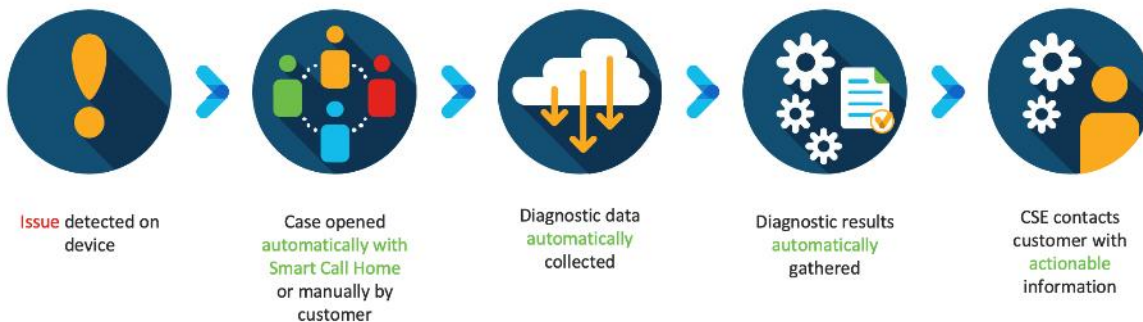
Connected TAC

Connected TAC is an automated transmission of technical support files to the Cisco® Technical Assistance Center (TAC) for accelerated troubleshooting.

Cisco Intersight enables Cisco TAC to automatically generate and upload Tech Support Diagnostic files when a Service Request is opened. If you have devices that are connected to Intersight but not claimed, Cisco TAC can only check the connection status and will not be permitted to generate Tech Support files. When enabled, this feature works in conjunction with the Smart Call Home service and with an appropriate service contract. Devices that are configured with Smart Call Home and claimed in Intersight can use Smart Call Home to open a Service Request and have Intersight collect Tech Support diagnostic files.

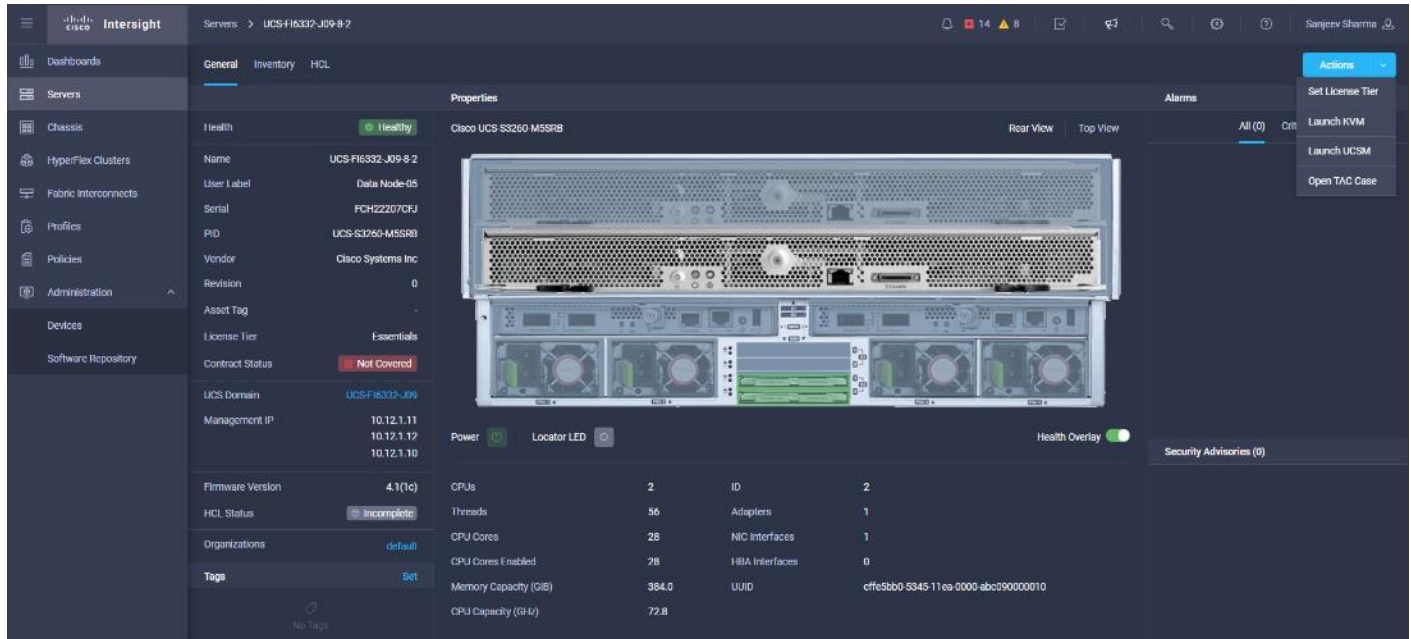
Figure 22 Cisco Intersight: Connected TAC

Cisco Intersight + Cisco TAC + Smart Call Home = Proactive resolution

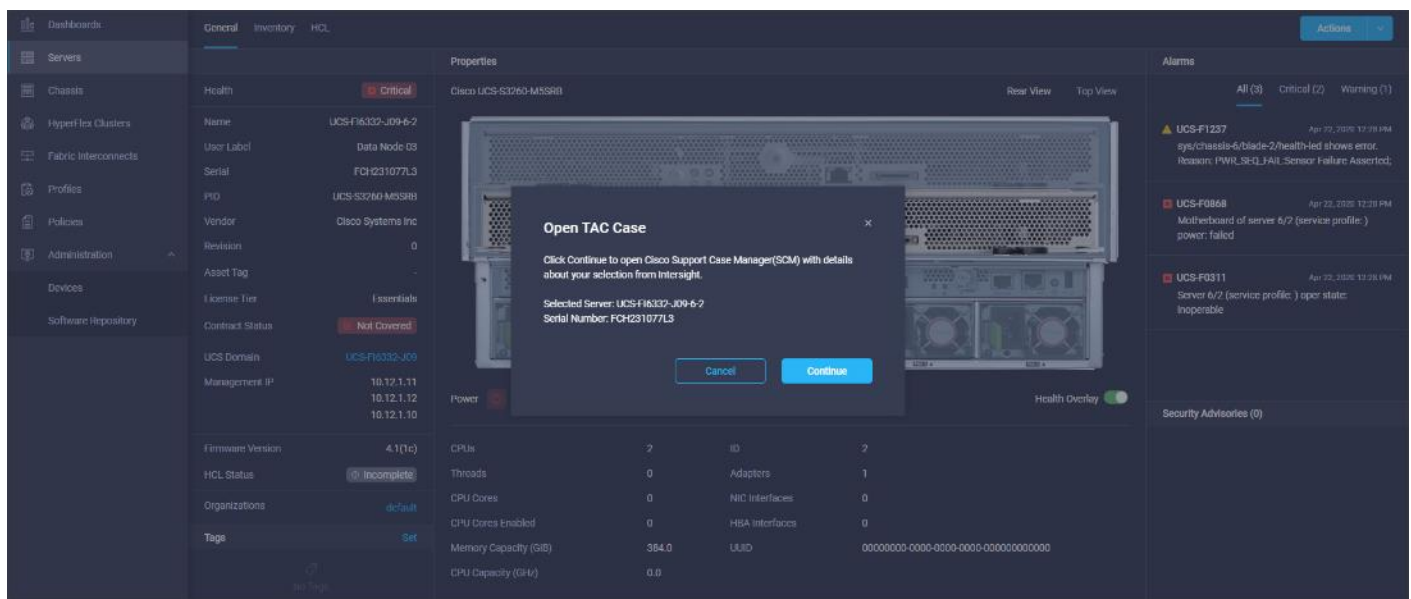


To enable Connected TAC, follow these steps:

1. Log into intersight.com.
2. Click the Servers tab. Select Server > Actions tab. From the drop-down list, select Open TAC Case.
3. Clicking “Open TAC Case” launches Cisco URL for Support case manager where associated service contracts for Server or Fabric Interconnect is displayed.



4. Click Continue.



5. Follow the procedure to Open TAC Case.

The screenshot displays the Cisco Support Case Manager interface. At the top, there is a navigation bar with the Cisco logo and links for Products & Services, Support, How to Buy, Training & Events, and Partners. The user's name, Sarjeev Sharma, is visible in the top right corner. The main heading is 'Support Case Manager' with a sub-heading 'Open a new support case for Sarjeev Sharma (sarjeev@...)'.

A progress bar shows three steps: 1. Check Entitlement (active), 2. Describe Problem, and 3. Review & Submit. Below the progress bar, the 'Request Type' section has three radio buttons: 'Diagnose and Fix' (selected), 'Request RMA', and 'Ask a Question'.

Under 'Find Product by Serial Number', there is a search input field containing 'PC23107A3' and a 'SEARCH' button. A checkbox option 'Search for other Open cases for the Serial Number' is present. Below this, there is a section for 'Find Product by Service Agreement'.

The 'Bypass Entitlement' section features a dropdown menu labeled 'Select one'. At the bottom left, there is a 'NEXT' button and a 'Save draft and exit' link.

Cisco Intersight Integration for HCL

Cisco Intersight evaluates the compatibility of your Cisco UCS and Cisco HyperFlex systems to check if the hardware and software have been tested and validated by Cisco or Cisco partners. Cisco Intersight reports validation issues after checking the compatibility of the server model, processor, firmware, adapters, operating system, and drivers, and displays the compliance status with the Hardware Compatibility List (HCL).

You can use Cisco UCS Tools, a host utility vSphere Installation Bundle (VIB), or OS Discovery Tool, an open source script to collect OS and driver information to evaluate HCL compliance.

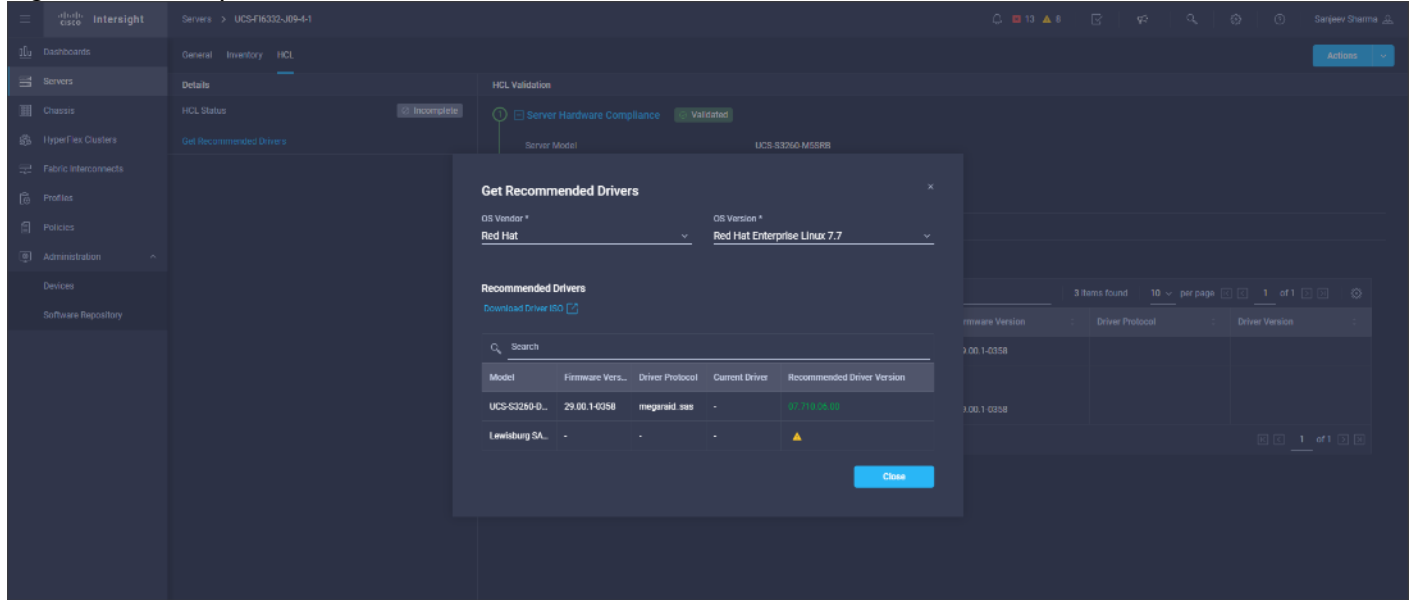
In Intersight, you can view the HCL compliance status in the dashboard (as a widget), the Servers table view, and the Server details page. Below is the server details page.



For more information, go to:

[https://www.intersight.com/help/features#compliance_with_hardware_compatibility_list_\(hcl\)](https://www.intersight.com/help/features#compliance_with_hardware_compatibility_list_(hcl))

Figure 23 Example of HCL Status and Driver Recommendation for RHEL 7.6



Advisories (PSIRTs)

Cisco Intersight sources critical security advisories from the Cisco Security Advisory service to alert users about the endpoint devices that are impacted by the advisories and deferrals. These alerts are displayed as Advisories in Intersight. The Cisco Security Advisory service identifies and monitors and updates the status of the advisories to provide the latest information on the impacted devices, the severity of the advisory, the impacted products, and any available workarounds. If there are no known workarounds, you can open a support case with Cisco TAC for further assistance. A select list of the security advisories is shown in Intersight under Advisories.

Figure 24 Intersight Dashboard

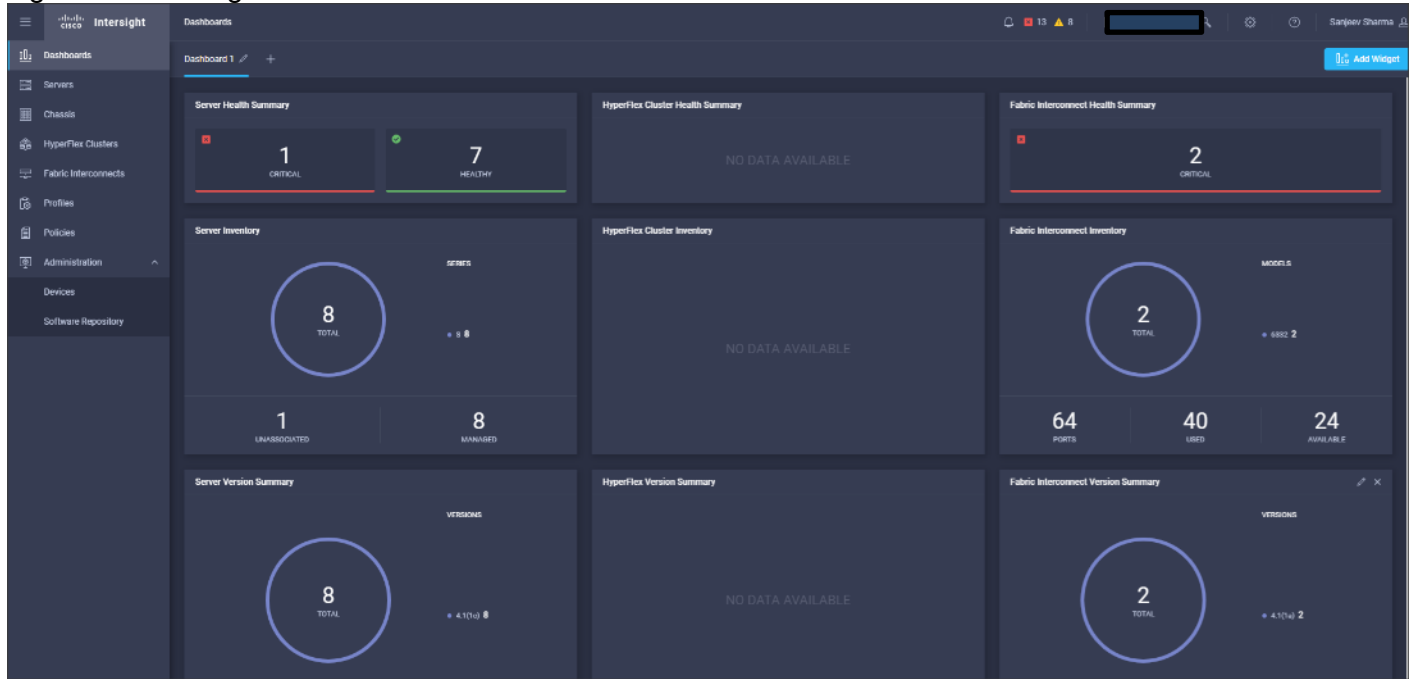


Figure 25 Example: List of PSIRTs Associated with Sample Intersight Account

The screenshot shows the Intersight 'Chassis' page. A notification banner at the top states 'New features have recently been added! Learn More'. Below the navigation sidebar, there is a search bar and an 'Export' button. The main content is a table with the following columns: Name, Health, Chassis ID, UCS Domain, Model, and Serial. There are 4 items found, displayed on page 1 of 1.

Name	Health	Chassis ID	UCS Domain	Model	Serial
UCS-F16332-J09-0	Healthy		8 UCS-F16332-J09	UCSC-C3X66-BASE	FOX212627A
UCS-F16332-J09-4	Healthy		4 UCS-F16332-J09	UCSC-C3X66-BASE	FOX2308P3GT
UCS-F16332-J09-5	Healthy		5 UCS-F16332-J09	UCSC-C3X66-BASE	FOX2307P4AJ
UCS-F16332-J09-6	Warning		6 UCS-F16332-J09	UCSC-C3X66-BASE	FOX2251PHDQ

The screenshot shows the Intersight 'Fabric Interconnects' page for 'UCS-F16332-J09 F1-A (Subordinate)'. The left sidebar shows details for the fabric interconnect, including its name, peer switch, management IP, model, serial, contract status (Not Covered), expansion modules, and firmware version. The main area displays a list of servers associated with this fabric interconnect. The servers table has columns for Name, Health, Model, Serial, and Management IP. There are 8 items found, displayed on page 1 of 1.

Name	Health	Model	Serial	Management IP
UCS-F16332-J09-0-2	Healthy	UCS-S3260-M5SRB	FCH22207CFJ	10.12.1.10
UCS-F16332-J09-4-1	Healthy	UCS-S3260-M5SRB	FCH231077M8	10.12.1.10
UCS-F16332-J09-0-1	Healthy	UCS-S3260-M5SRB	FCH2218725M	10.12.1.10
UCS-F16332-J09-4-2	Healthy	UCS-S3260-M5SRB	FCH231077M8	10.12.1.10
UCS-F16332-J09-0-1	Healthy	UCS-S3260-M5SRB	FCH231077L2	10.12.1.10
UCS-F16332-J09-0-2	Healthy	UCS-S3260-M5SRB	FCH231077KQ	10.12.1.10
UCS-F16332-J09-4-1	Healthy	UCS-S3260-M5SRB	FCH231077LK	10.12.1.10
UCS-F16332-J09-4-2	Critical	UCS-S3260-M5SRB	FCH231077L3	10.12.1.10

On the right side, there is an 'Alarms' section showing several critical alerts for UCS-F0276, indicating link-down issues on fabric interconnect A.

Deployment Hardware and Software

Cisco Unified Computing System Configuration

This section details the Cisco Unified Computing System (Cisco UCS) configuration that was done as part of the infrastructure build out. The racking, power, and installation of the Cisco UCS Rack Server is described in the physical topology section earlier in this document. Please refer to the [Cisco UCS Manager Getting Started Guide](#). For more information about each step, see the [Cisco UCS Manager – Configuration Guides](#).

Configure Cisco UCS Fabric Interconnect

This document assumes you are using Cisco UCS Manager Software version 4.1(1a). To upgrade the Cisco UCS Manager software and the Cisco UCS 6332 Fabric Interconnect software to a higher version of the firmware, see the [Cisco UCS Manager Install and Upgrade Guides](#).

Alternatively, if you intend to clear the existing Cisco UCS Manager configuration, follow these steps:

1. Connect a console cable to the console port on what will become the primary fabric interconnect.
2. If the fabric interconnects were previously deployed and you want to erase it to redeploy, follow these steps:
 - a. Login with the existing username and password.

```
#connect local-mgmt  
  
#erase config  
  
#yes (to confirm)
```
3. After the fabric interconnect restarts, the out-of-the-box first time installation prompt appears, type “console” and press Enter.
4. Follow the Initial Configuration steps as outlined in Cisco UCS Manager Getting Started Guide. When configured, log into UCSM IP Address via the web interface to perform the base Cisco UCS configuration.

Configure Fabric Interconnects for a Cluster Setup

To configure the Cisco UCS Fabric Interconnects, follow these steps:

1. Verify the following physical connections on the fabric interconnect:
 - The management Ethernet port (mgmt0) is connected to an external hub, switch, or router.
 - The L1 ports on both fabric interconnects are directly connected to each other.
 - The L2 ports on both fabric interconnects are directly connected to each other

Configure Fabric Interconnect A

To configure Fabric Interconnect A, follow these steps:

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

At the prompt to enter the configuration method, enter `console` to continue.
If asked to either perform a new setup or restore from backup, enter `setup` to continue.
Enter `y` to continue to set up a new Fabric Interconnect.
Enter `y` to enforce strong passwords.

2. Enter the password for the admin user.
3. Enter the same password again to confirm the password for the admin user.

When asked if this fabric interconnect is part of a cluster, answer `y` to continue.
Enter `A` for the switch fabric.

4. Enter the cluster name for the system name.
5. Enter the Mgmt0 IPv4 address.
6. Enter the Mgmt0 IPv4 netmask.
7. Enter the IPv4 address of the default gateway.
8. Enter the cluster IPv4 address.

To configure DNS, answer `y`.

9. Enter the DNS IPv4 address.

Answer `y` to set up the default domain name.

10. Enter the default domain name.

Review the settings that were printed to the console, and if they are correct, answer `yes` to save the configuration.

11. Wait for the login prompt to make sure the configuration has been saved.

Configure Fabric Interconnect B

To configure Fabric Interconnect B, follow these steps:

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

When prompted to enter the configuration method, enter `console` to continue.
The installer detects the presence of the partner Fabric Interconnect and adds this fabric interconnect to the cluster. Enter `y` to continue the installation.

2. Enter the admin password that was configured for the first Fabric Interconnect.
3. Enter the Mgmt0 IPv4 address.
4. Answer `yes` to save the configuration.
5. Wait for the login prompt to confirm that the configuration has been saved.

For more information about configuring Cisco UCS 6332 Series Fabric Interconnect, go to:
https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/ucs-manager/GUI-User-Guides/Getting-Started/4-1/b_UCSM_Getting_Started_Guide_4_1.html

Log into Cisco UCS Manager

To log into Cisco UCS Manager, follow these steps:

1. Open a Web browser and navigate to the Cisco UCS 6332 Fabric Interconnect cluster address.
2. Click the Launch link to download the Cisco UCS Manager software.
3. If prompted to accept security certificates, accept as necessary.
4. When prompted, enter admin for the username and enter the administrative password.
5. Click Login to log into the Cisco UCS Manager.

Upgrade Cisco UCS Manager Software to Version 4.1(1a)

This document assumes you're using Cisco UCS 4.1(1a). Refer to the [Cisco UCS 4.1 Release](#) (upgrade Cisco UCS Manager software and Cisco UCS 6332 Fabric Interconnect software to version 4.1(1a)). Also, make sure the Cisco UCS C-Series version 4.1(1a) software bundles are installed on the Fabric Interconnects.

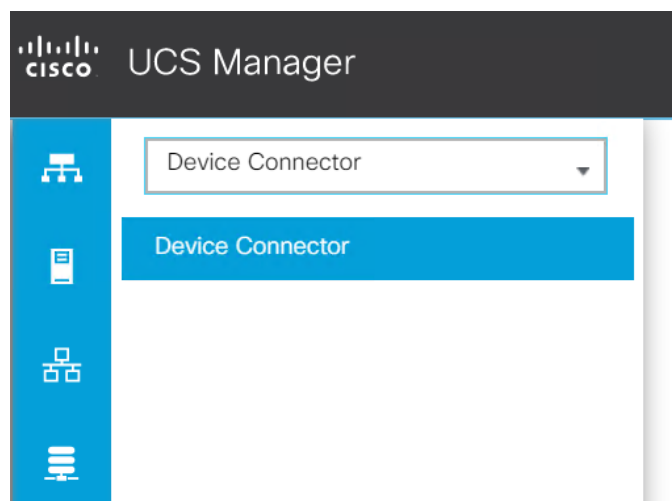


Upgrading Cisco UCS firmware is beyond the scope of this document. However for complete Cisco UCS Install and Upgrade Guides, go to: <https://www.cisco.com/c/en/us/support/servers-unified-computing/ucs-manager/products-installation-guides-list.html>

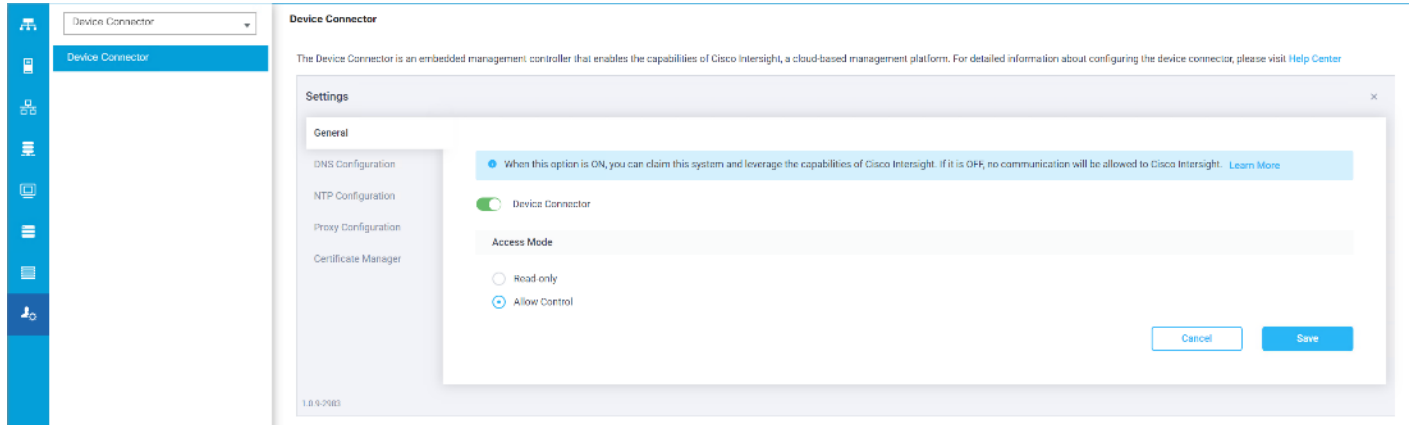
Register UCSM with Intersight

To register UCSM with Intersight, follow these steps:

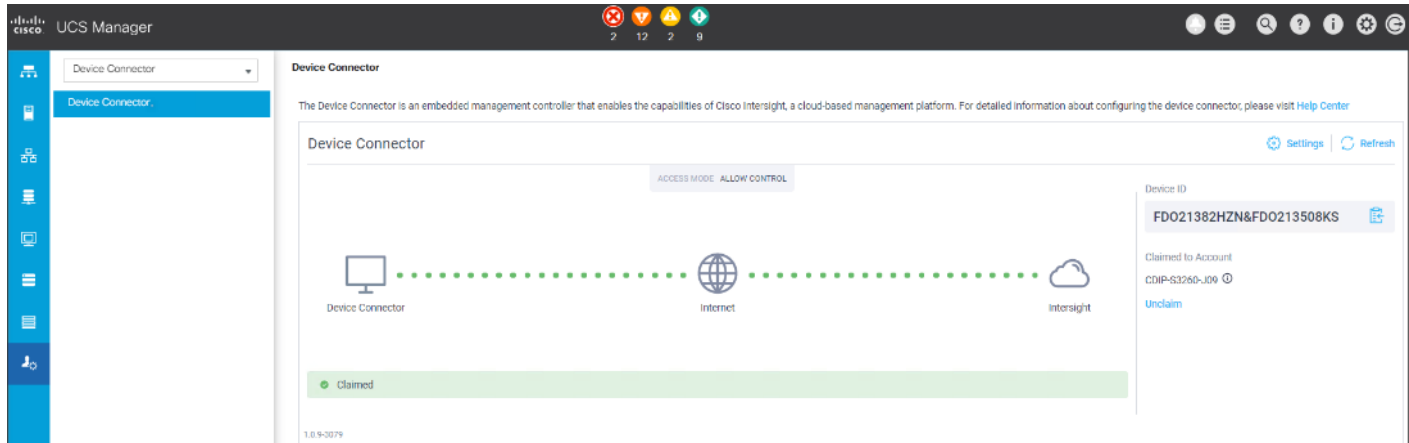
1. Log into WebUI for Cisco UCS Manager, go to the Admin tab. From the drop-down list select Device Connector. Click Settings.



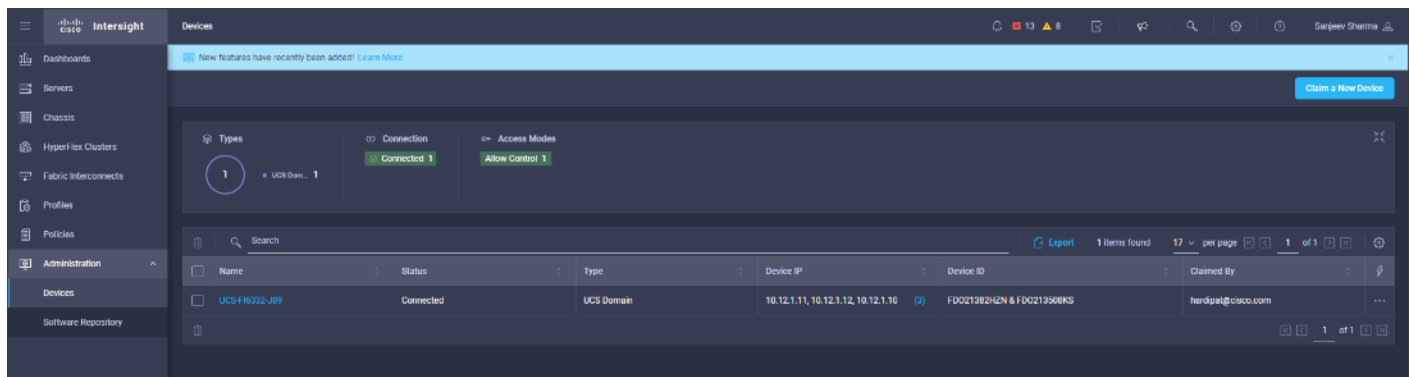
2. Enable Device Connector. Select Allow Control in Access Mode.



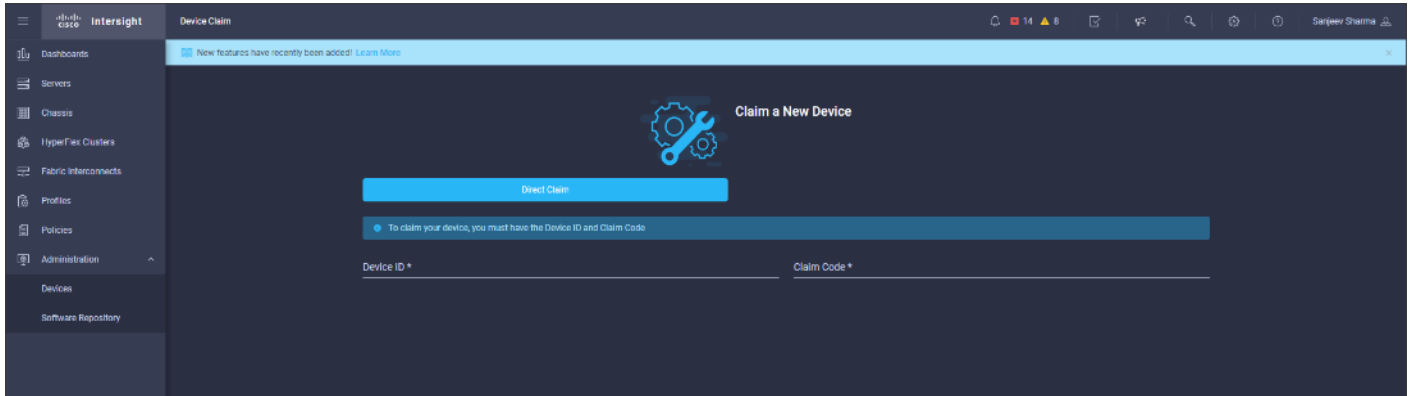
3. Complete the steps for the DNS configuration, NTP Configuration and Proxy Configuration as applicable. Click Save.
4. Make sure UCSM can communicate to Intersight.



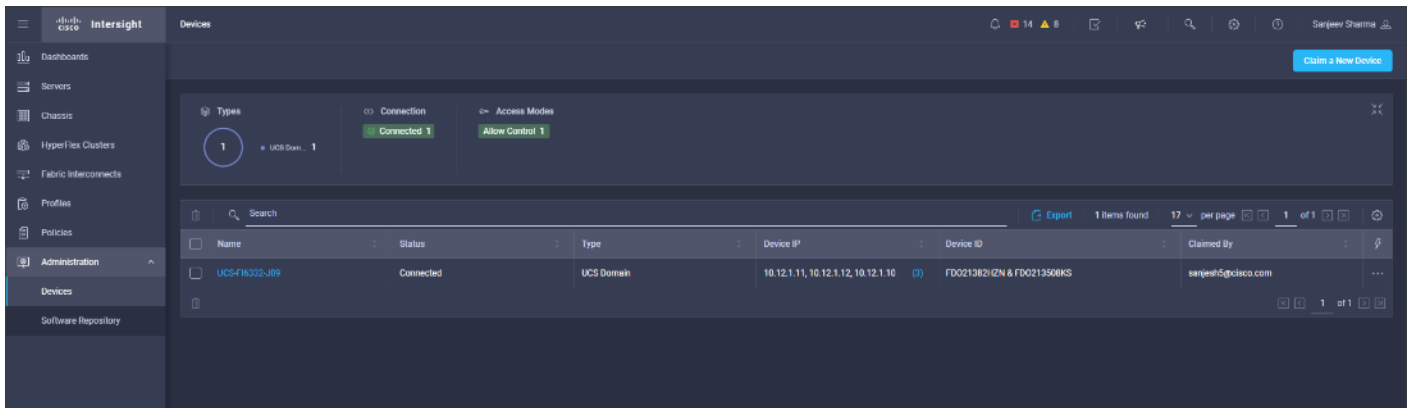
5. Copy the Device Claim ID from the Device Connector screen.
6. Log into intersight.com
7. Select the Devices tab and click Claim a New Device.



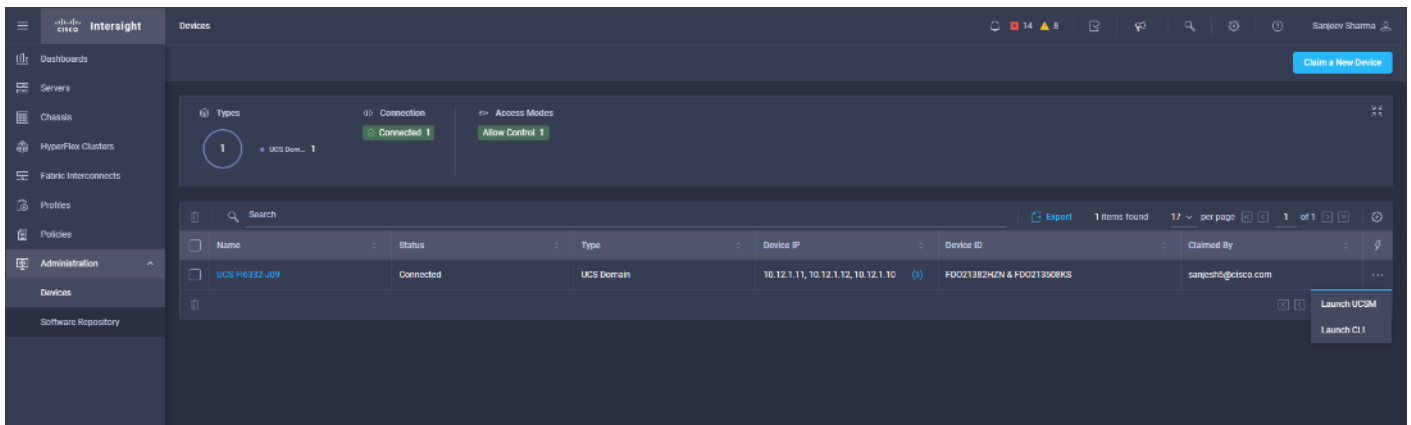
8. Enter the Device ID and Device Claim Code copied from Cisco UCS Manager. Click Claim.



Once Claimed, UCSM can be launched directly from Intersight.



9. Click Launch UCSM.



For more information, go to: [Claiming a Device](#)



For this study, we launched UCSM through Intersight. UCSM WebUI can also be accessed the traditional way which is by entering the IP address of Cisco UCS Manager in a Web Browser. For more information, go to:

[https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/Cisco UCS Data Intelligence Platform with Cloudera and CDSW.html](https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/Cisco_UCS_Data_Intelligence_Platform_with_Cloudera_and_CDSW.html)

Configure Cisco UCS Manager through Intersight

To configure Cisco UCS Manager, follow these high-level steps:

1. Configure Fabric Interconnects for a Cluster Setup.
2. Set Fabric Interconnects to Fibre Channel End Host Mode.
3. Synchronize Cisco UCS to NTP.
4. Configure Fabric Interconnects for Rack or Chassis and Blade Server Discovery.
5. Configure Global Policies.
6. Configure Server Ports.
7. Configure LAN on Cisco UCS Manager.
8. Configure Ethernet LAN Uplink Ports.
9. Set QoS system class and Jumbo Frames in both the Cisco Fabric Interconnect.
10. Create Uplink Port Channels to Cisco Nexus Switches.
11. Configure FC SAN Uplink Ports.
12. Configure VLAN.
13. Configure IP, UUID, Server, MAC Pool and policy:
 - a. IP Pool Creation
 - b. UUID Suffix Pool Creation
 - c. Server Pool Creation
 - d. Configure Server BIOS Policy
 - e. Create Adapter Policy
 - f. Configure Default Maintenance Policy
 - g. Configure vNIC Template
 - h. Create Server Boot Policy

Details for each step are discussed in the following sections.

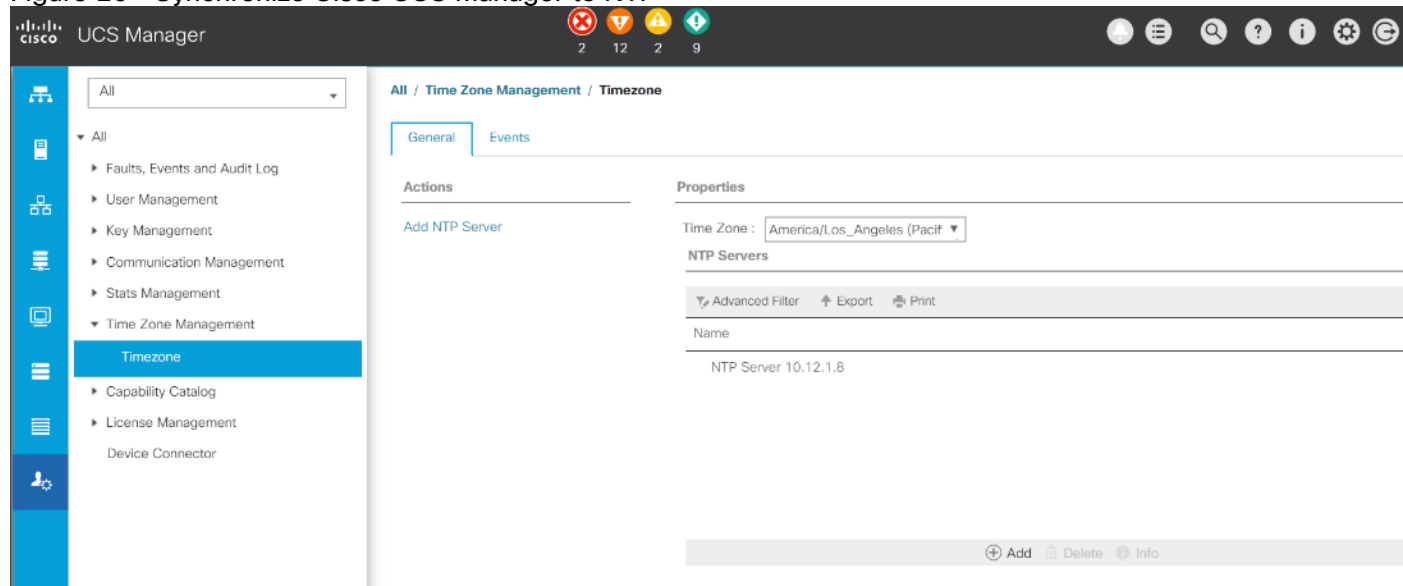
Synchronize Cisco UCSM to NTP

To synchronize the Cisco UCS environment to the NTP server, follow these steps:

1. In Cisco UCS Manager, in the navigation pane, click the Admin tab.
2. Select All > Time zone Management.
3. In the Properties pane, select the appropriate time zone in the Time zone menu.

4. Click Save Changes and then click OK.
5. Click Add NTP Server.
6. Enter the NTP server IP address and click OK.
7. Click OK to finish.
8. Click Save Changes.

Figure 26 Synchronize Cisco UCS Manager to NTP



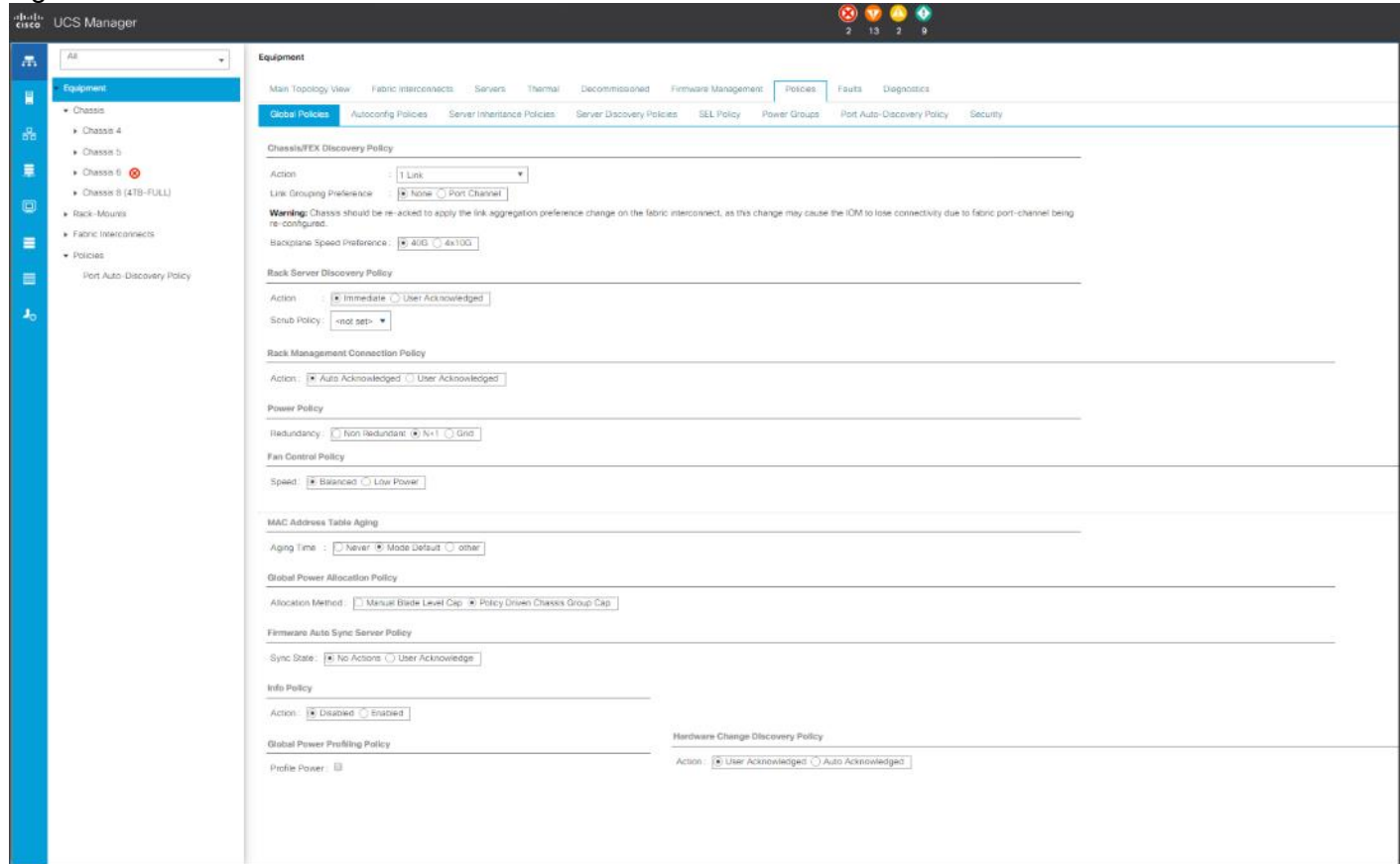
Configure Global Policies

The rack server and chassis discovery policy determine how the system reacts when you add a new rack server or chassis. We recommend using the platform max value as shown. Using platform max helps ensure that Cisco UCS Manager uses the maximum number of IOM uplinks available.

To configure the global policies, follow this step:

1. In Cisco UCS Manager, go to Equipment > Policies (right pane) > Global Policies.

Figure 27 Global Policies in UCSM

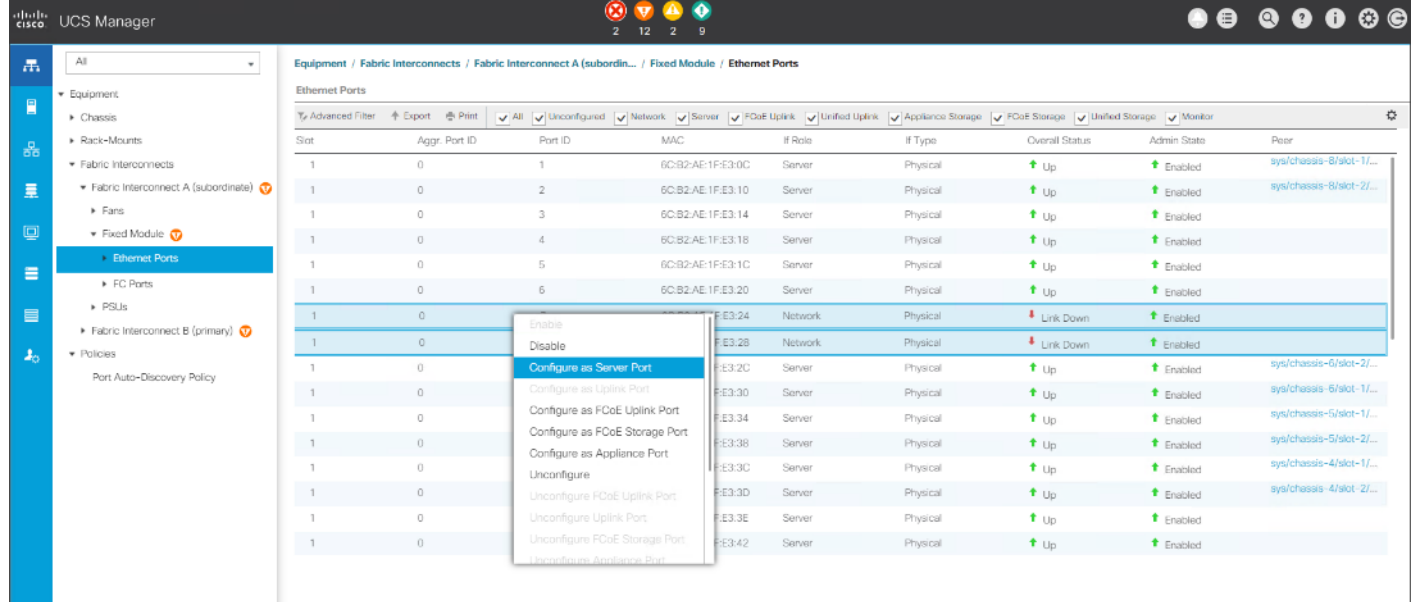


Configure Server Ports

Configure Server Ports to initiate Chassis and Blade discovery. To configure server ports, follow these steps:

1. Go to Equipment > Fabric Interconnects > Fabric Interconnect A > Fixed Module > Ethernet Ports.
2. Select the ports (for this solution ports are 1-16) which are connected to the Cisco UCS VIC 1387 on Cisco UCS C240 M5 rack server.
3. Right-click and select Configure as Server Port.

Figure 28 Configure Server Port on Cisco UCS Manager Fabric Interconnect for Server/Chassis Discovery



Configure Uplink Ports

Configure Network Ports to connect to the data center network switch.

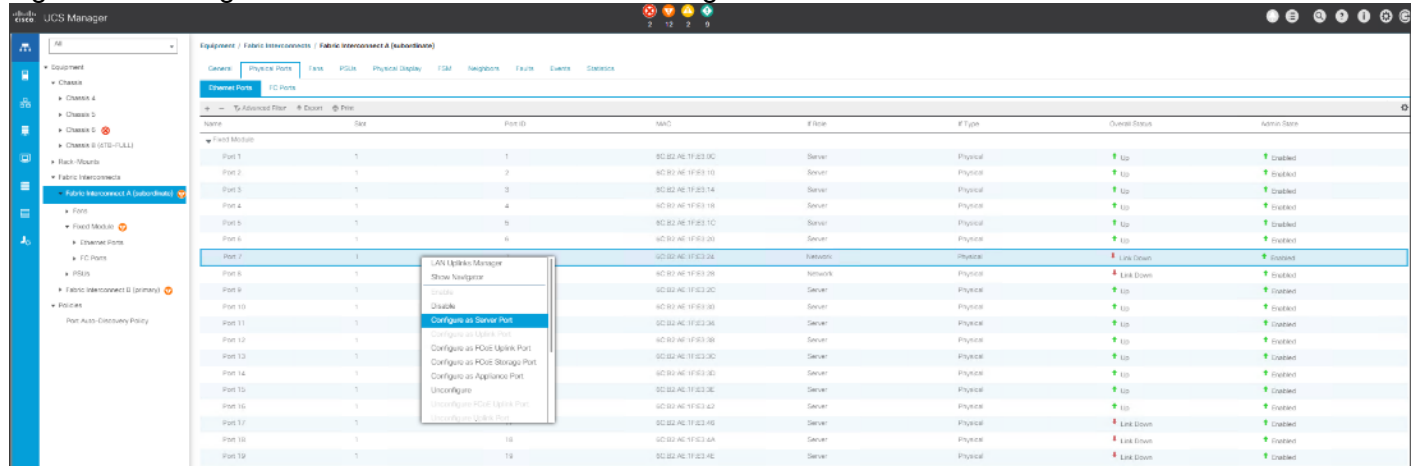


In our solution study, we connected to the Cisco Nexus 9000 series switch.

To configure Network ports, follow these steps:

1. Go to Equipment > Fabric Interconnects > Fabric Interconnect A > Fixed Module > Ethernet Ports.
2. Select the ports (for this solution ports are 29-32) which are connected to the Cisco Nexus 9000 series switch for northbound network connectivity.
3. Right-click and select Configure as Network Port.

Figure 29 Configure Network Port on Cisco UCS Manager Fabric Interconnect

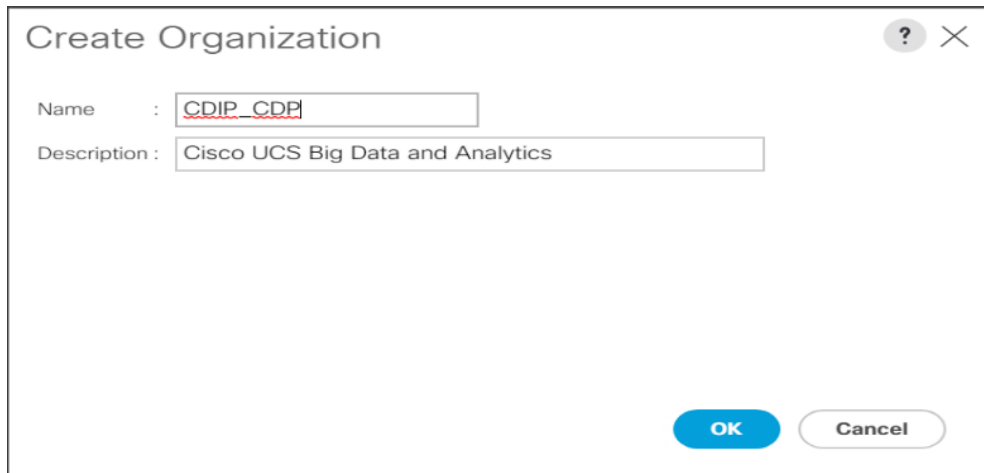
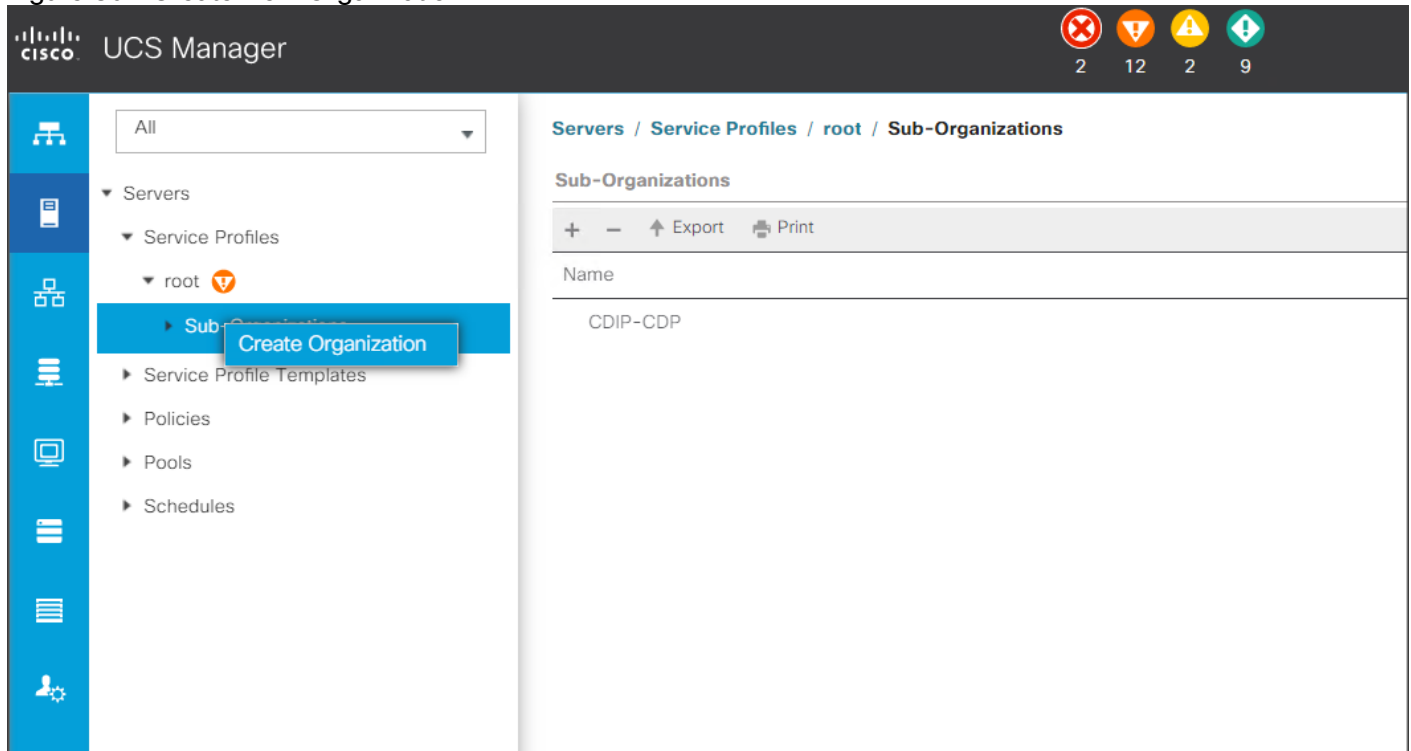


Create New Organization

To configure the necessary Organization for the Cisco UCS environment, follow these steps:

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select root > Sub-Organization.
3. Right-click and select Create Sub-Organization.
4. Enter the name of the Organization.
5. Click OK.

Figure 30 Create New Organization





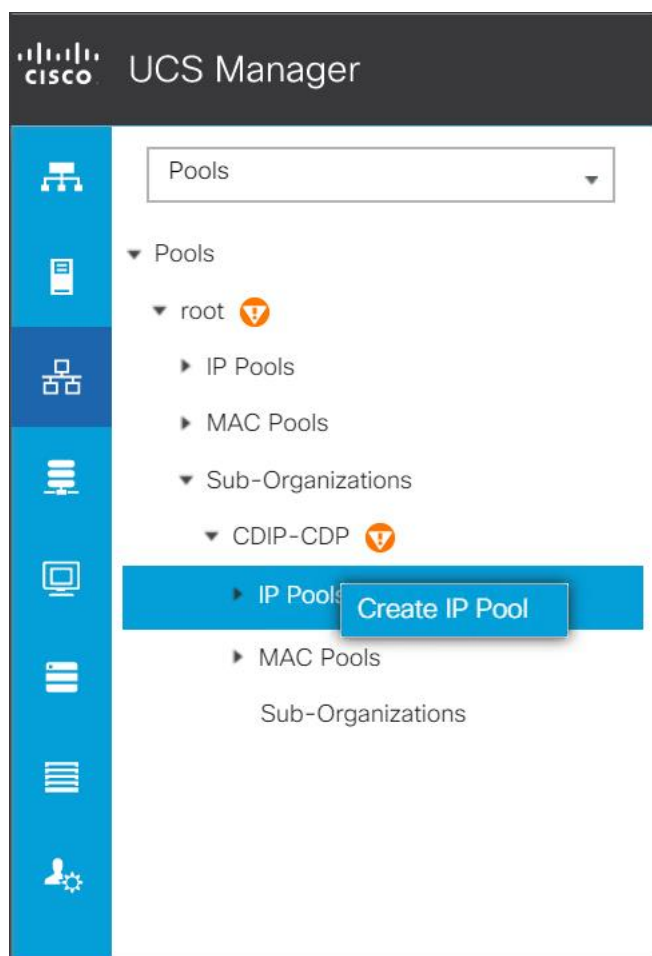
Cisco UCS Manager pools and policies required for this solution were created under new “CDIP-CDP” Organization created.

Configure IP, UUID, Server and MAC Pools

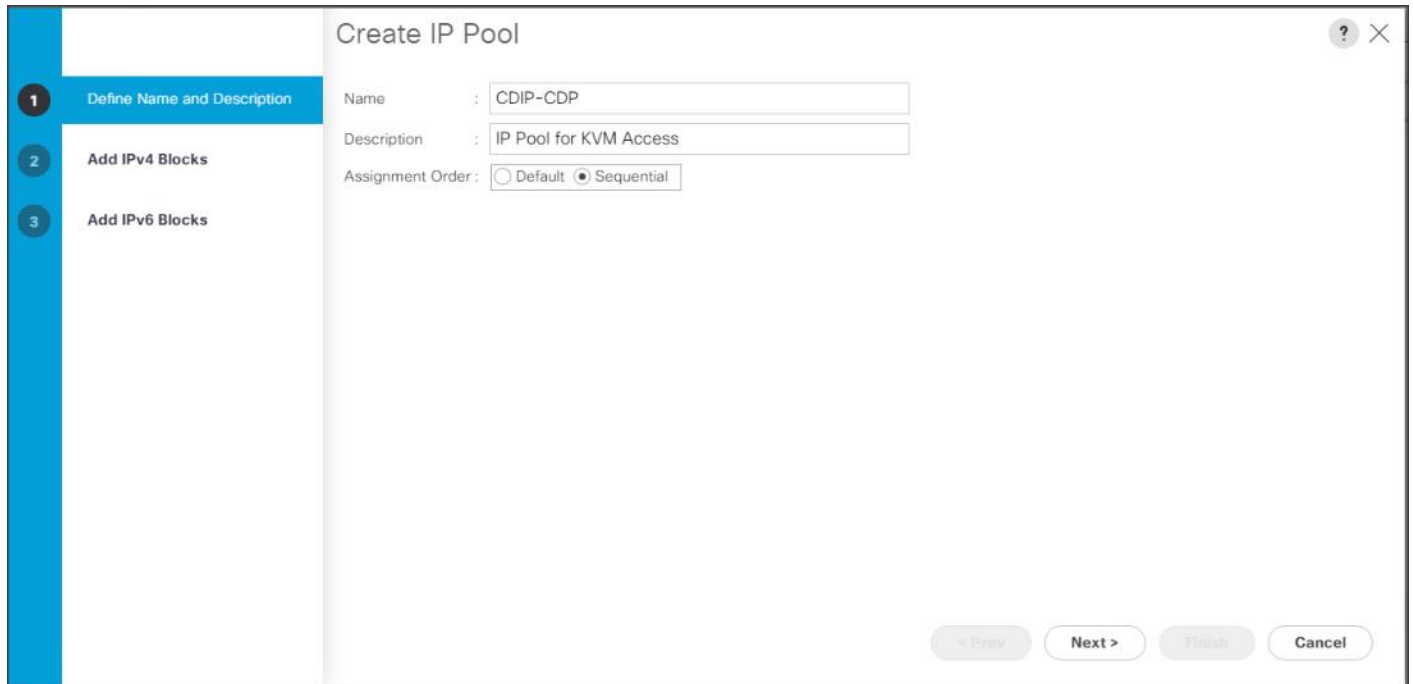
IP Pool Creation

An IP address pool on the out-of-band management network must be created to facilitate KVM access to each compute node in the Cisco UCS domain. To create a block of IP addresses for server KVM access in the Cisco UCS environment, follow these steps:

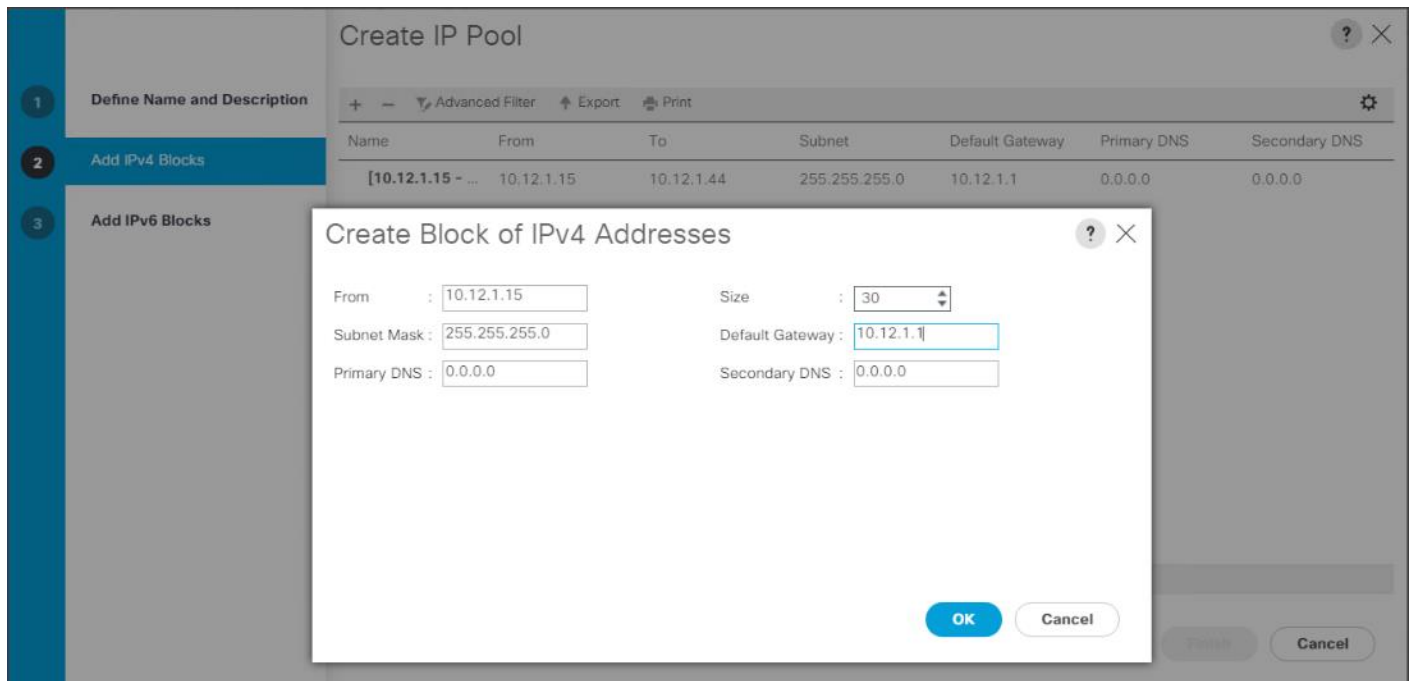
1. In Cisco UCS Manager, in the navigation pane, click the LAN tab.
2. Select Pools > root > Sub-Organizations > CDIP-CDP> IP Pools > click Create IP Pool.



3. Enter name for the IP Pool, select option Sequential to assign IP in sequential order then click Next.



4. Click Add IPv4 Block.
5. Enter the starting IP address of the block and the number of IP addresses required, and the subnet and gateway information as shown below.

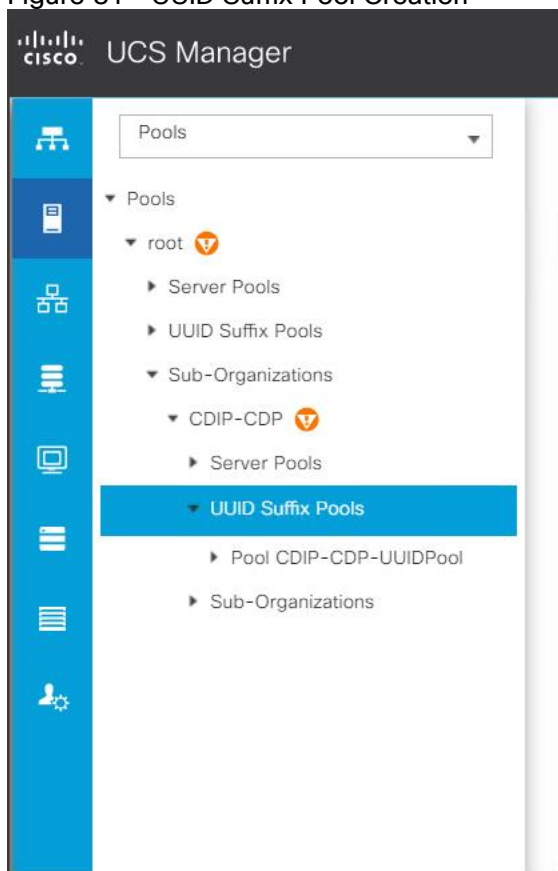


UUID Suffix Pool Creation

To configure the necessary universally unique identifier (UUID) suffix pool for the Cisco UCS environment, follow these steps:

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select Pools > root > Sub-Organization > CDIP-CDP.
3. Right-click UUID Suffix Pools and then select Create UUID Suffix Pool.
4. Enter the name of the UUID name.
5. Optional: Enter a description for the UUID pool.
6. Keep the prefix at the derived option and select Sequential in as Assignment Order then click Next.

Figure 31 UUID Suffix Pool Creation



Create UUID Suffix Pool

1 Define Name and Description

2 Add UUID Blocks

Name : CDIP-CDP

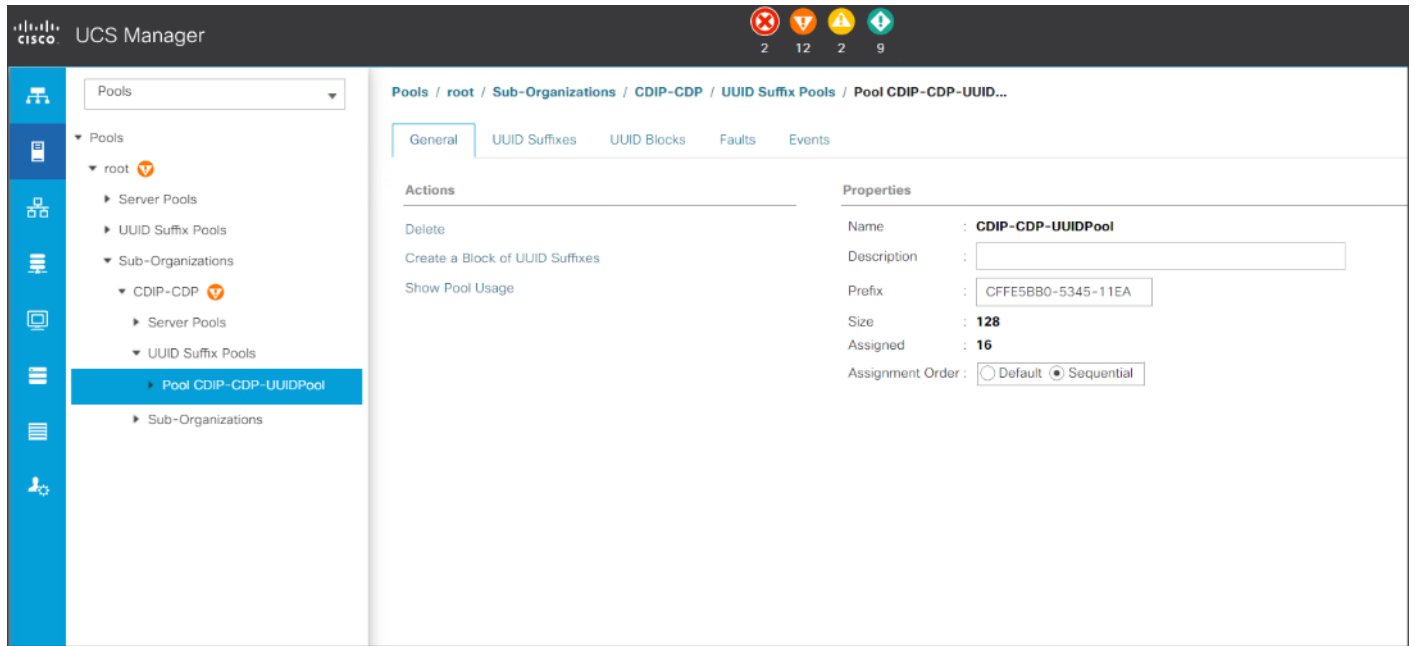
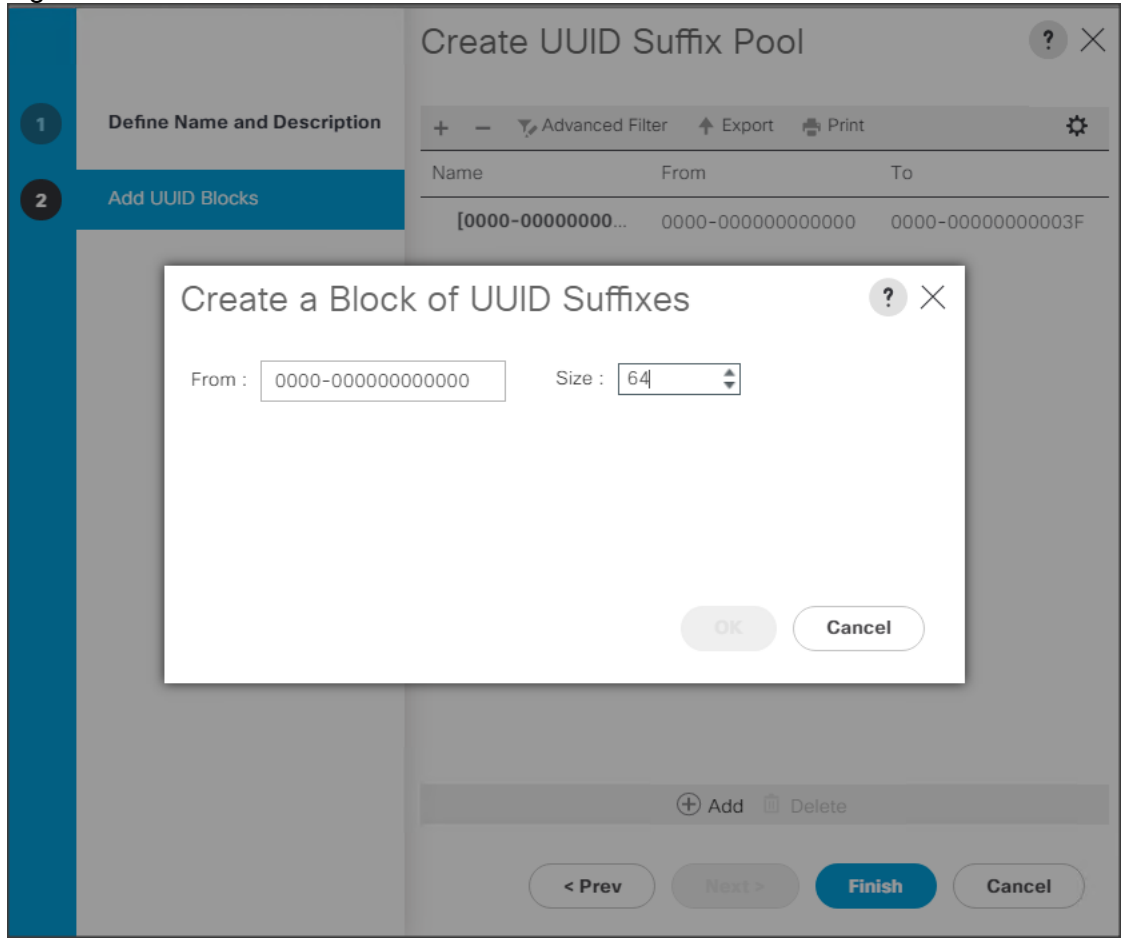
Description : UUID Suffix Pool

Prefix : Derived other

Assignment Order : Default Sequential

< Prev Next > Finish Cancel

Figure 32 Create a Block of UUID Suffixes



Server Pool Creation

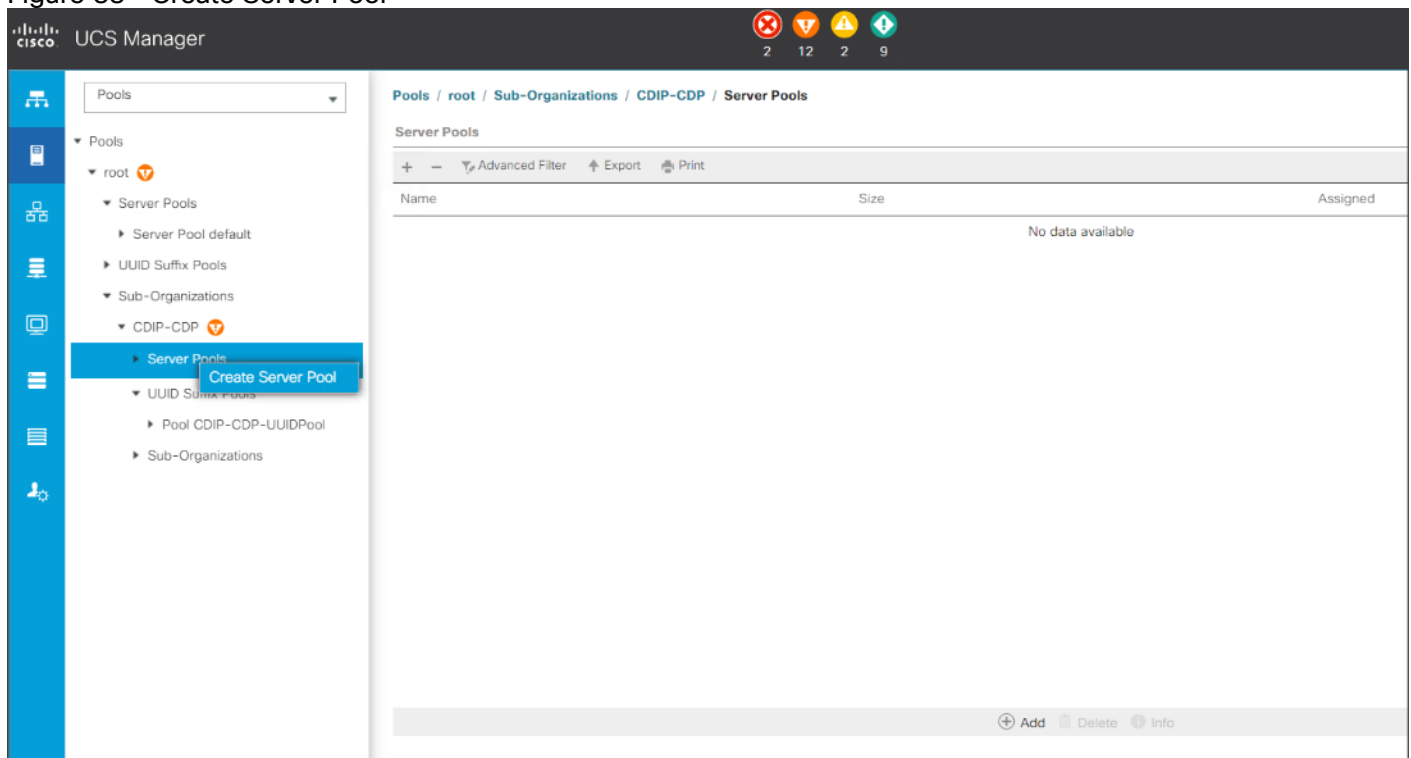
To configure the necessary server pool for the Cisco UCS environment, follow these steps:

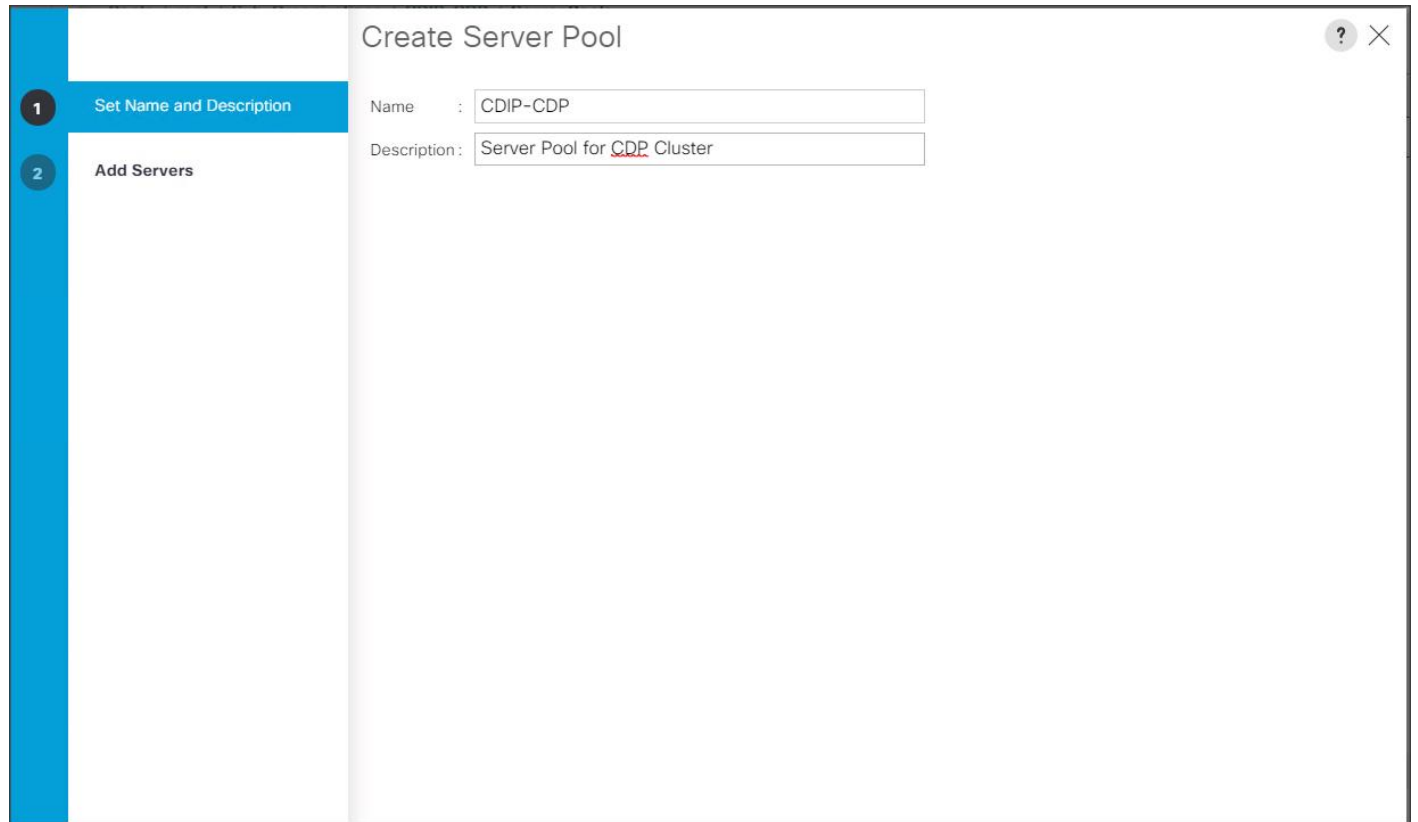


Consider creating unique server pools to achieve the granularity that is required in your environment.

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select Pools > root > Sub-Organization > CDIP-CDP > right-click Server Pools > Select Create Server Pool.
3. Enter name of the server pool.
4. Optional: Enter a description for the server pool then click Next.

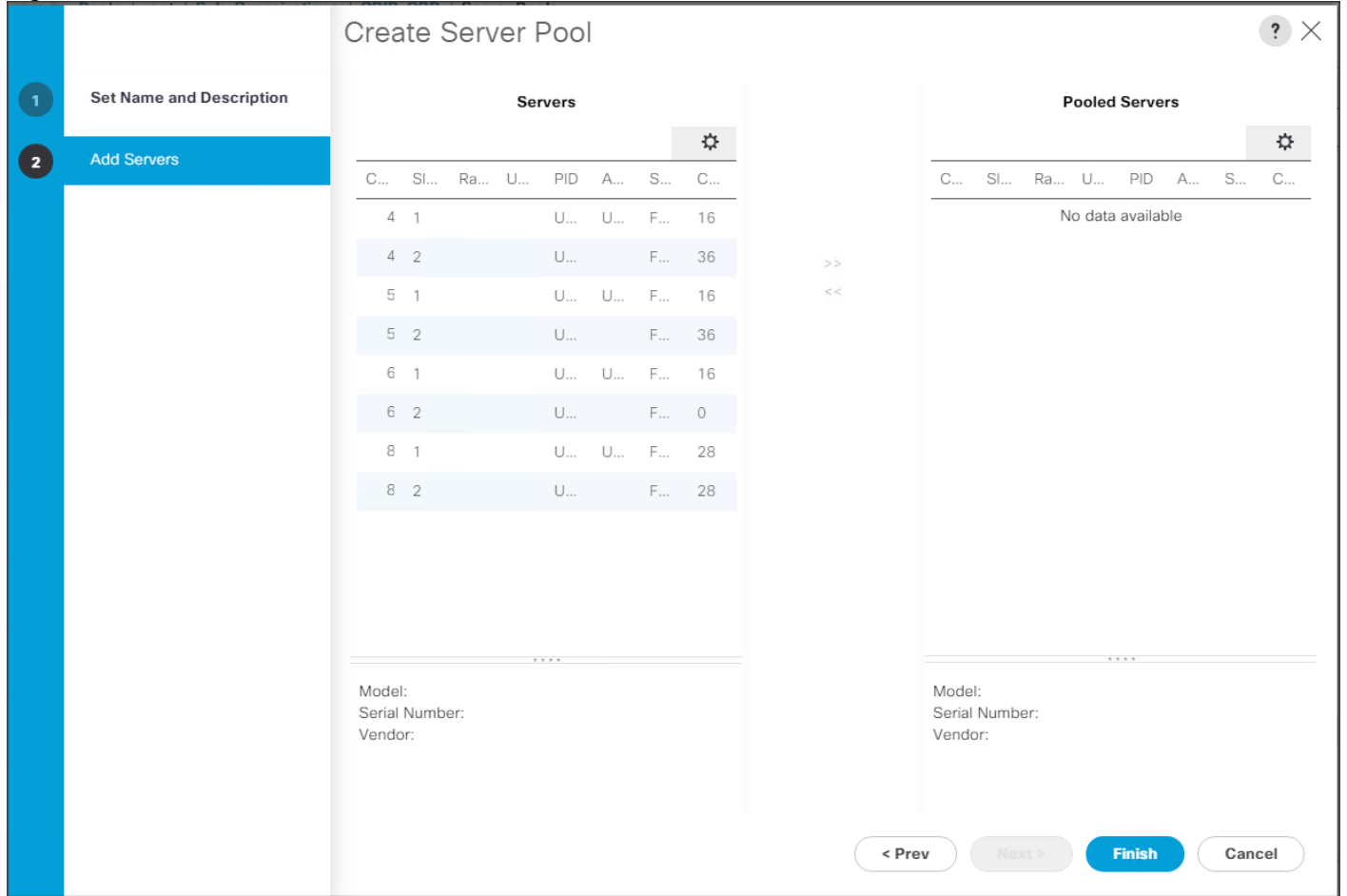
Figure 33 Create Server Pool





5. Select servers to be used for the deployment and click > to add them to the server pool. In our case we added thirty servers in this server pool.
6. Click Finish and then click OK.

Figure 34 Add Server in the Server Pool



7. Once the added Servers are in the Pooled servers, click Finish.

Create Server Pool

1 Set Name and Description

2 Add Servers

Servers

C...	SI...	Ra...	U...	PID	A...	S...	C...
No data available							

>>

<<

Pooled Servers

C...	SI...	Ra...	U...	PID	A...	S...	C...
4	1			U...	U...	F...	16
4	2			U...		F...	36
5	1			U...	U...	F...	16
5	2			U...		F...	36
6	1			U...	U...	F...	16
6	2			U...		F...	0
8	1			U...	U...	F...	28
8	2			U...		F...	28

Model:
Serial Number:
Vendor:

Model:
Serial Number:
Vendor:

< Prev Next > **Finish** Cancel

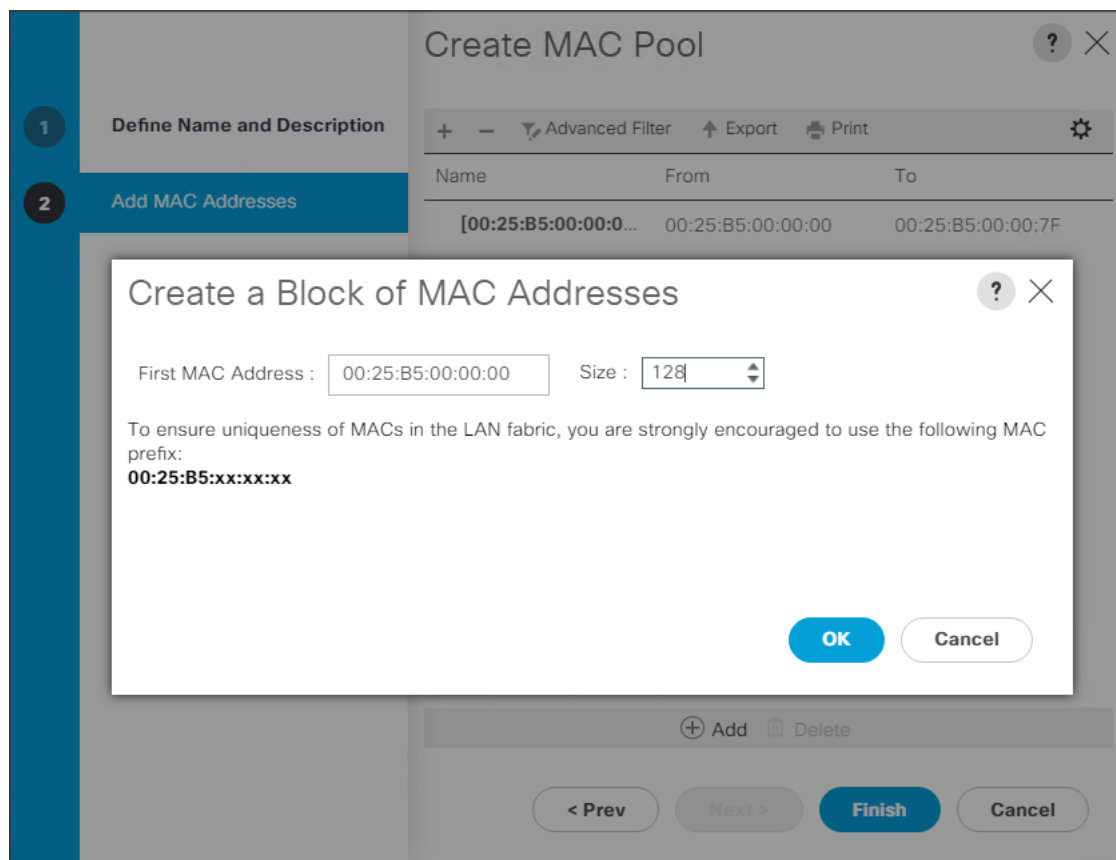
MAC Pool Creation

To configure the necessary MAC address pools for the Cisco UCS environment, follow these steps:

1. In Cisco UCS Manager, click the LAN tab in the navigation pane.
2. Select Pools > root > Sub-Organization > CDIP-CDP> right-click MAC Pools.
3. Select Create MAC Pool to create the MAC address pool.
4. Enter name for MAC pool. Select Assignment Order as "Sequential".
5. Enter the seed MAC address and provide the number of MAC addresses to be provisioned.
6. Click OK and then click Finish.
7. In the confirmation message, click OK.

Figure 35 Creating a Block of MAC Addresses

The screenshot displays the Cisco UCS Manager interface. On the left, a navigation pane shows the hierarchy: Pools > root > MAC Pools. A 'Create MAC Pool' button is highlighted over the 'MAC Pools' folder. The main content area shows the 'MAC Pools' list with columns for Name, Size, and Assigned. Overlaid on this is the 'Create MAC Pool' dialog box. The dialog has a progress indicator on the left with two steps: '1 Define Name and Description' (active) and '2 Add MAC Addresses'. The form fields are: Name: CDIP-CDP; Description: Mac Pool for CDP Cluster; Assignment Order: Sequential (selected). At the bottom, there are buttons for '< Prev', 'Next >', 'Finish' (highlighted), and 'Cancel'.

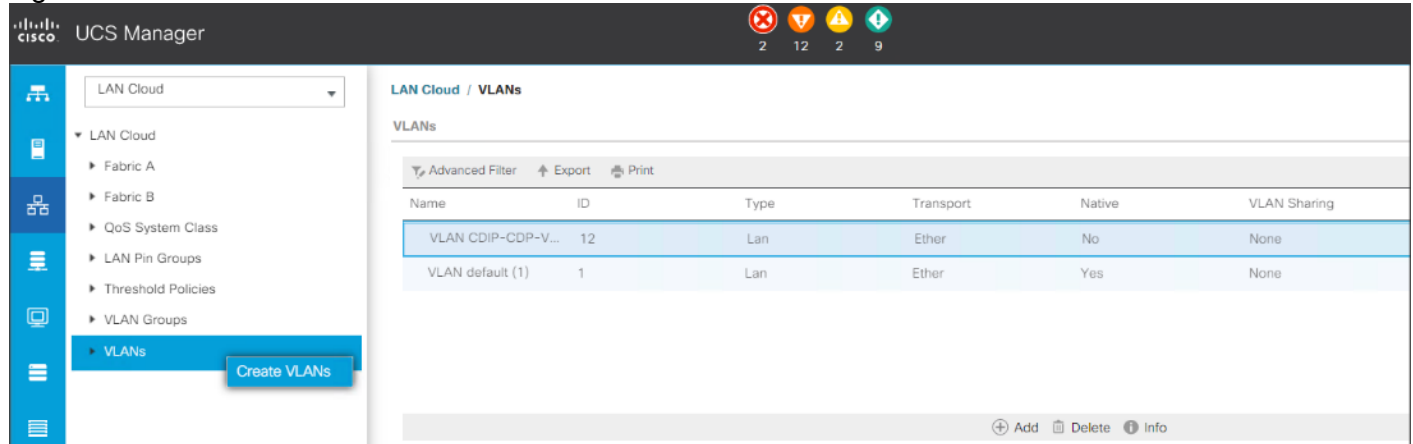


Configure VLAN

To configure the necessary virtual local area networks (VLANs) for the Cisco UCS environment, follow these steps:

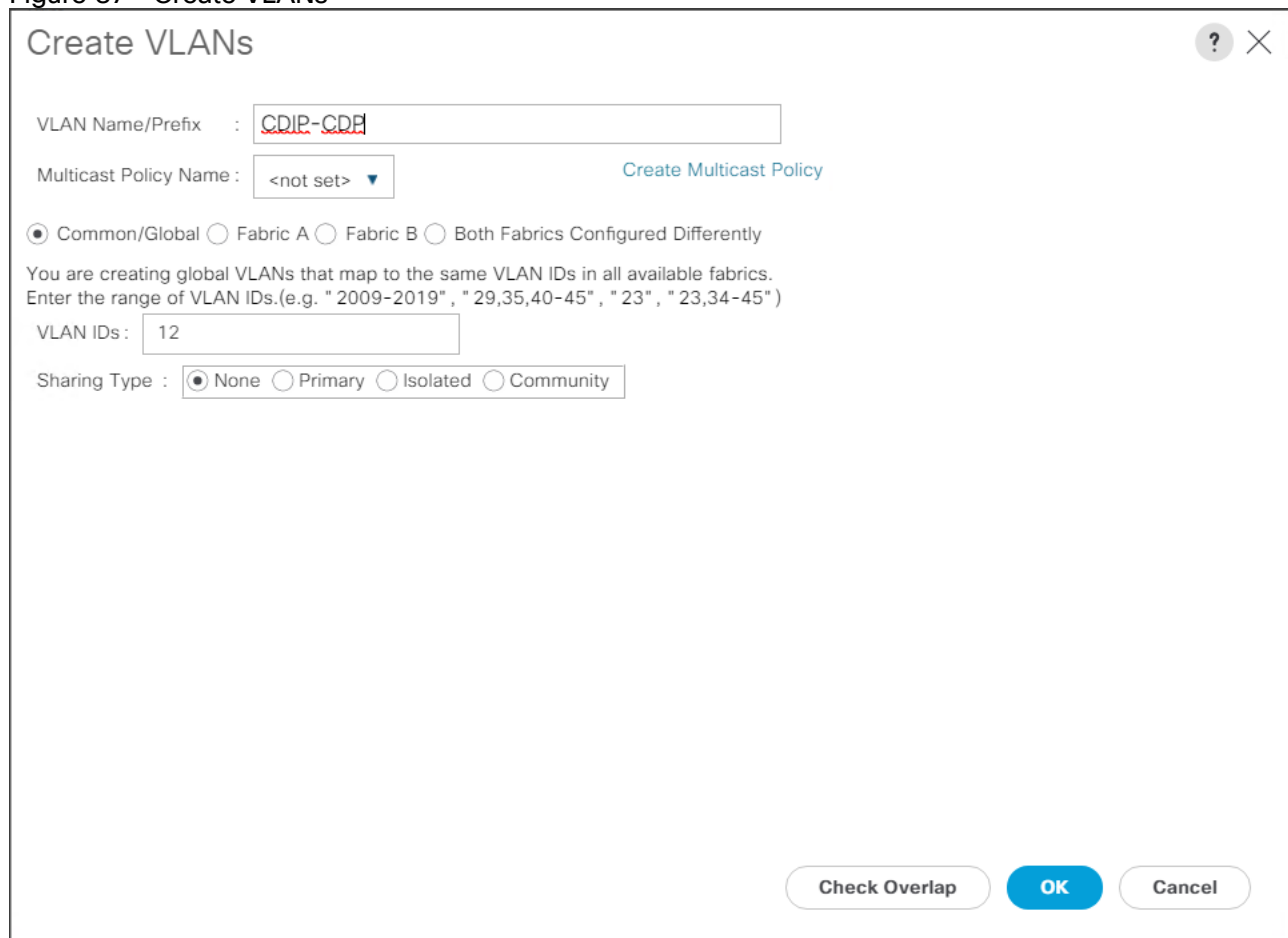
1. In Cisco UCS Manager, click the LAN tab in the navigation pane.
2. Select LAN > LAN Cloud.
3. Right-click VLANs.
4. Select Create VLANs.
5. Enter the name of the VLAN to be used.
6. Keep the Common/Global option selected for the scope of the VLAN.
7. Enter <VLAN Number> as the ID of the VLAN ID.
8. Keep the Sharing Type as None.

Figure 36 Create VLAN



The NIC will carry the data traffic from VLAN12. A single vNIC is used in this configuration and the Fabric Failover feature in Fabric Interconnects will take care of any physical port down issues. It will be a seamless transition from an application perspective.

Figure 37 Create VLANs



Set System Class QoS and Jumbo Frame in Both Cisco Fabric Interconnects

To configure jumbo frames and enable quality of service in the Cisco UCS fabric, follow these steps:

1. In Cisco UCS Manager, click the LAN tab in the navigation pane.
2. Select LAN > LAN Cloud > QoS System Class.
3. In the right pane, click the General tab.
4. On the Platinum row, enter 9216 in the box under the MTU column.
5. Click Save Changes.
6. Click OK.


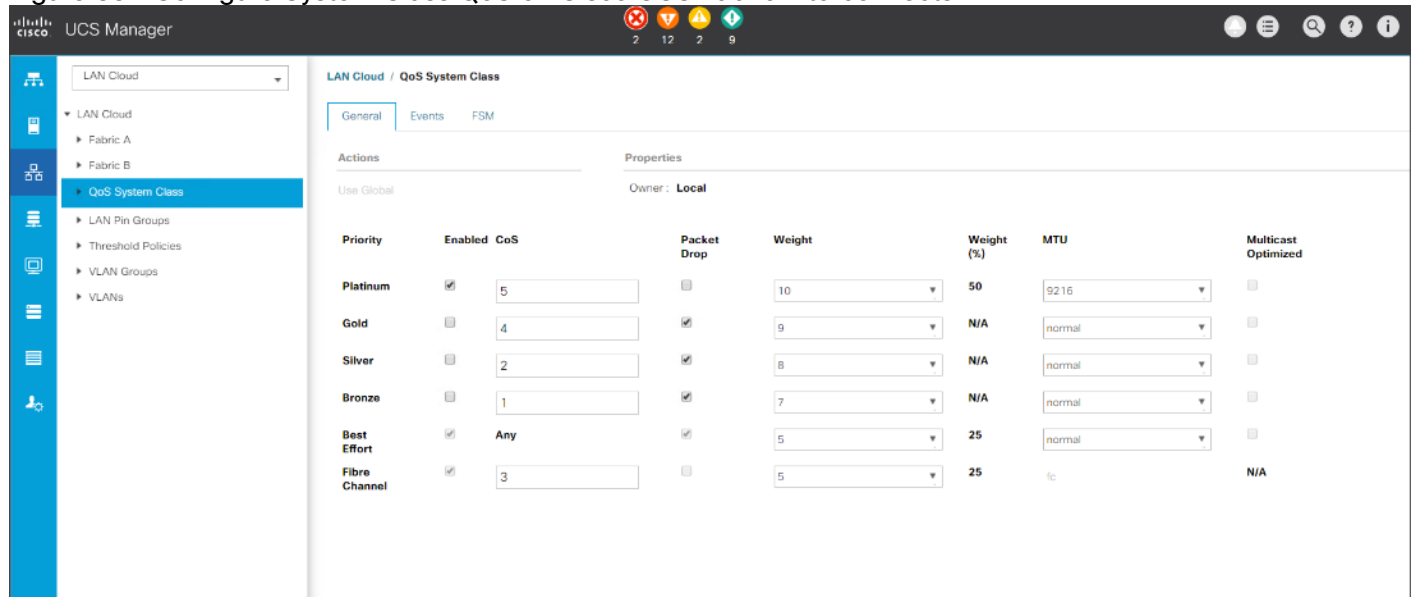
 **Changing the QoS system class MTU requires a reboot of Cisco UCS Fabric Interconnect for changes to be effective.**

Figure 38 Configure System Class QoS on Cisco UCS Fabric Interconnects

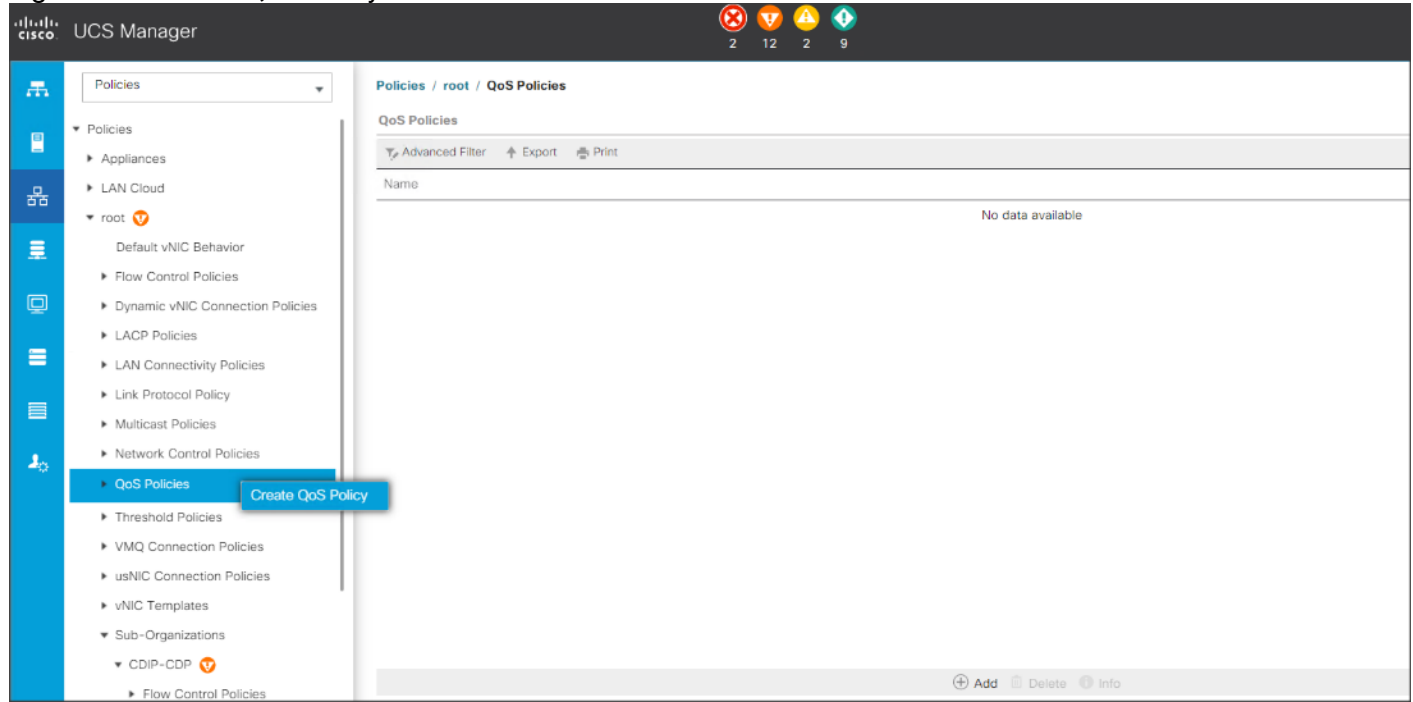


Create QoS Policies

To create the QoS policy to assign priority based on the class using the Cisco UCS Manager GUI, follow these steps:

1. Select LAN tab in the left pane in the Cisco UCS Manager GUI.
2. Select LAN > Policies > root > CDIP-CDP> QoS Policies.
3. Right-click QoS Policies.
4. Select Create QoS Policy.

Figure 39 Create QoS Policy



We created a Platinum class QoS policy for this solution.

Figure 40 Platinum QoS Policy

Create QoS Policy

Name :

Egress

Priority :

Burst(Bytes) :

Rate(Kbps) :

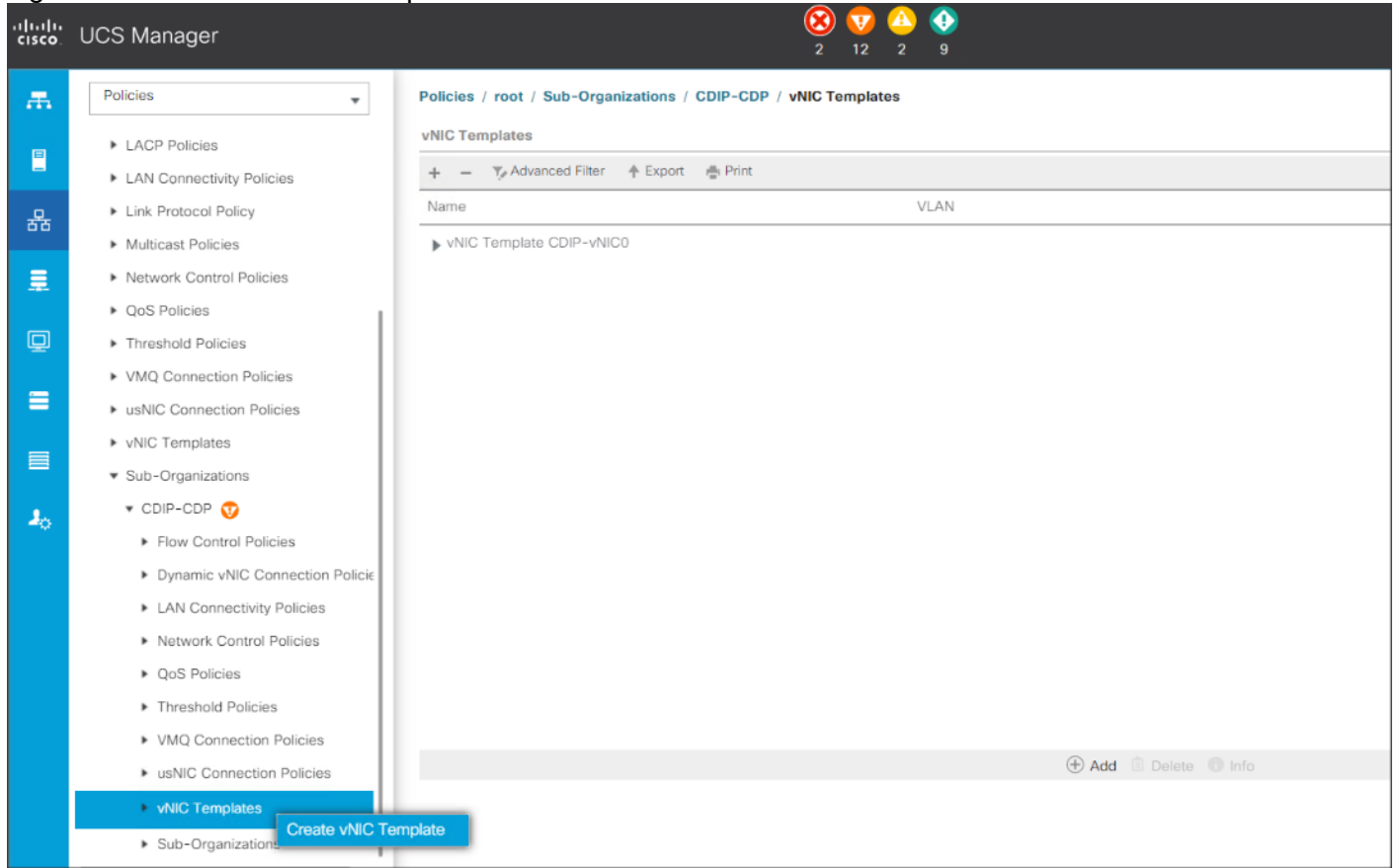
Host Control : None Full

Create vNIC Templates

To create multiple virtual network interface card (vNIC) templates for the Cisco UCS environment, follow these steps:

1. In Cisco UCS Manager, click the LAN tab in the navigation pane.
2. Select Policies > root > Sub-Organization > CDIP-CDP> vNIC Template.
3. Right-click vNIC Templates.
4. Select Create vNIC Template.
5. Enter a name for vNIC template.
6. Keep Fabric A selected. Select the Enable Failover checkbox.
7. Select Updating Template as the Template Type.
8. Under VLANs, select the checkboxes for desired VLANs to add as part of the vNIC Template.
9. Set Native-VLAN as the native VLAN.
10. For MTU, enter 9000.
11. In the MAC Pool list, select MAC Pool configured.
12. Select QOS policy created earlier.
13. Select default Network Control Policy.
14. Click OK to create the vNIC template.

Figure 41 Create the vNIC Template



Create vNIC Template

Name :

Description :

Fabric ID : Fabric A Fabric B Enable Failover

Redundancy

Redundancy Type : No Redundancy Primary Template Secondary Template

Target

Adapter
 VM

Warning

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

Template Type : Initial Template Updating Template

VLANs | VLAN Groups

Advanced Filter | Export | Print

Select	Name	Native VLAN	VLAN ID
<input checked="" type="checkbox"/>	CDIP-CDP-VLAN	<input checked="" type="radio"/>	12
<input type="checkbox"/>	default	<input type="radio"/>	1

OK **Cancel**

Create VLAN

CDN Source : vNIC Name User Defined

MTU :

Warning

Make sure that the MTU has the same value in the [QoS System Class](#) corresponding to the Egress priority of the selected QoS Policy.

MAC Pool :

QoS Policy :

Network Control Policy :

Pin Group :

Stats Threshold Policy :

Connection Policies

Dynamic vNIC usNIC VMQ

Dynamic vNIC Connection Policy :

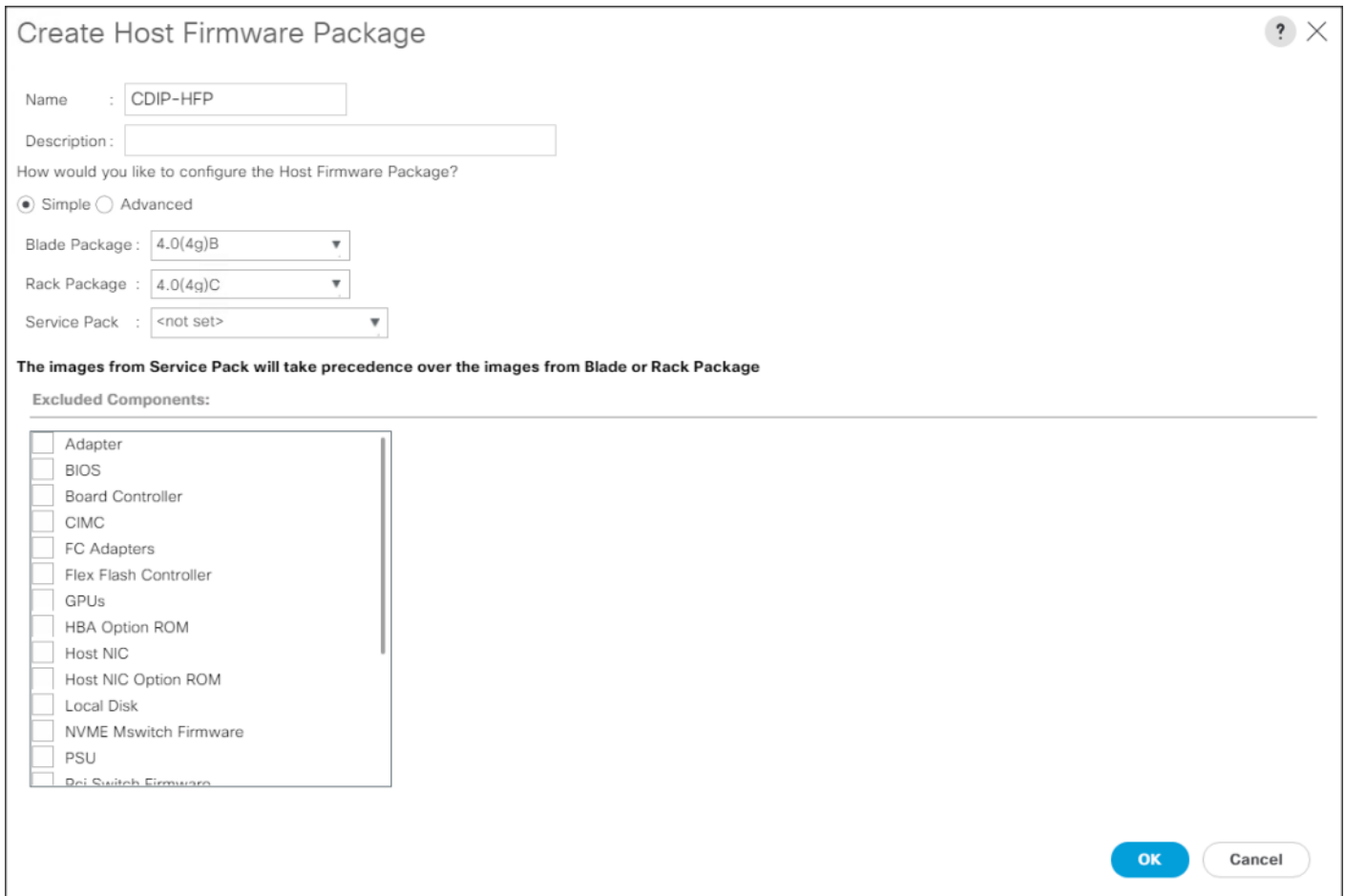
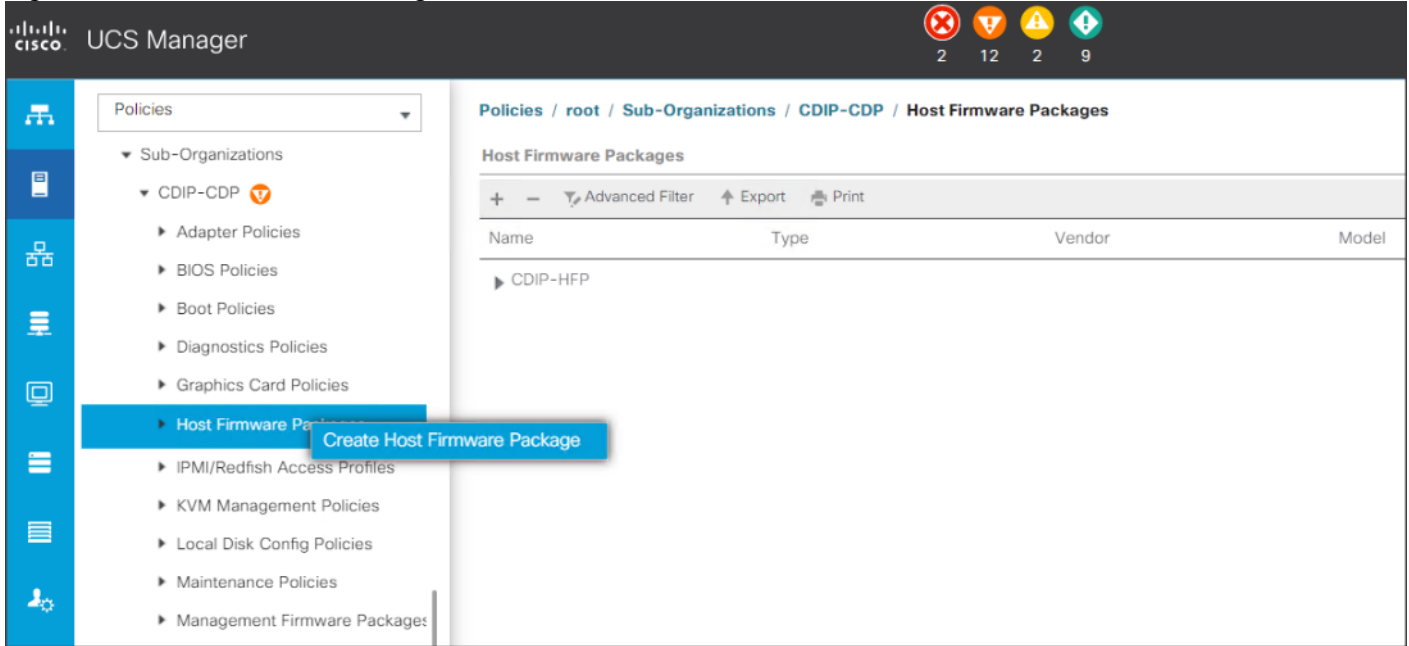
Create Host Firmware Package

Firmware management policies allow the administrator to select the corresponding packages for a given server configuration. These policies often include packages for adapter, BIOS, board controller, FC adapters, host bus adapter (HBA) option ROM, and storage controller properties.

To create a firmware management policy for a given server configuration in the Cisco UCS environment, follow these steps:

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select Policies > root > Sub-Organization > CDIP-CDP > Host Firmware Packages.
3. Right-click Host Firmware Packages.
4. Select Create Host Firmware Package.
5. Enter name of the host firmware package.
6. Leave Simple selected.
7. Select the version.
8. Click OK to create the host firmware package.

Figure 42 Host Firmware Package



Create Power Control Policy

To create a power control policy for the Cisco UCS environment, follow these steps:

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select Policies > root > Sub-Organization > CDIP-CDP > Power Control Policies.
3. Right-click Power Control Policies.
4. Select Create Power Control Policy.
5. Select Fan Speed Policy as “Max Power”.
6. Enter NoPowerCap as the power control policy name.
7. Change the power capping setting to No Cap.
8. Click OK to create the power control policy.

Figure 43 Create Power Control Policy

The screenshot displays the Cisco UCS Manager interface. The breadcrumb navigation at the top reads: **Policies / root / Sub-Organizations / CDIP-CDP / Power Control Policies**. The left-hand navigation pane shows a tree structure under 'Policies' with 'Sub-Organizations' expanded to 'CDIP-CDP', and 'Power Control Policies' selected. A blue callout box labeled 'Create Power Control Policy' points to this selection. The main content area shows the 'Power Control Policies' tab active, with a table listing the policy 'NoPowerCap' and its 'Power Priority' as 'No Cap'.

Name	Power Priority
NoPowerCap	No Cap

Create Power Control Policy

Name : NoPowerCap

Description :

Fan Speed Policy : Max Power

Power Capping

If you choose **cap**, the server is allocated a certain amount of power based on its priority within its power group. Priority values range from 1 to 10, with 1 being the highest priority. If you choose **no-cap**, the server is exempt from all power capping.

No Cap cap

Cisco UCS Manager only enforces power capping when the servers in a power group require more power than is currently available. With sufficient power, all servers run at full capacity regardless of their priority.

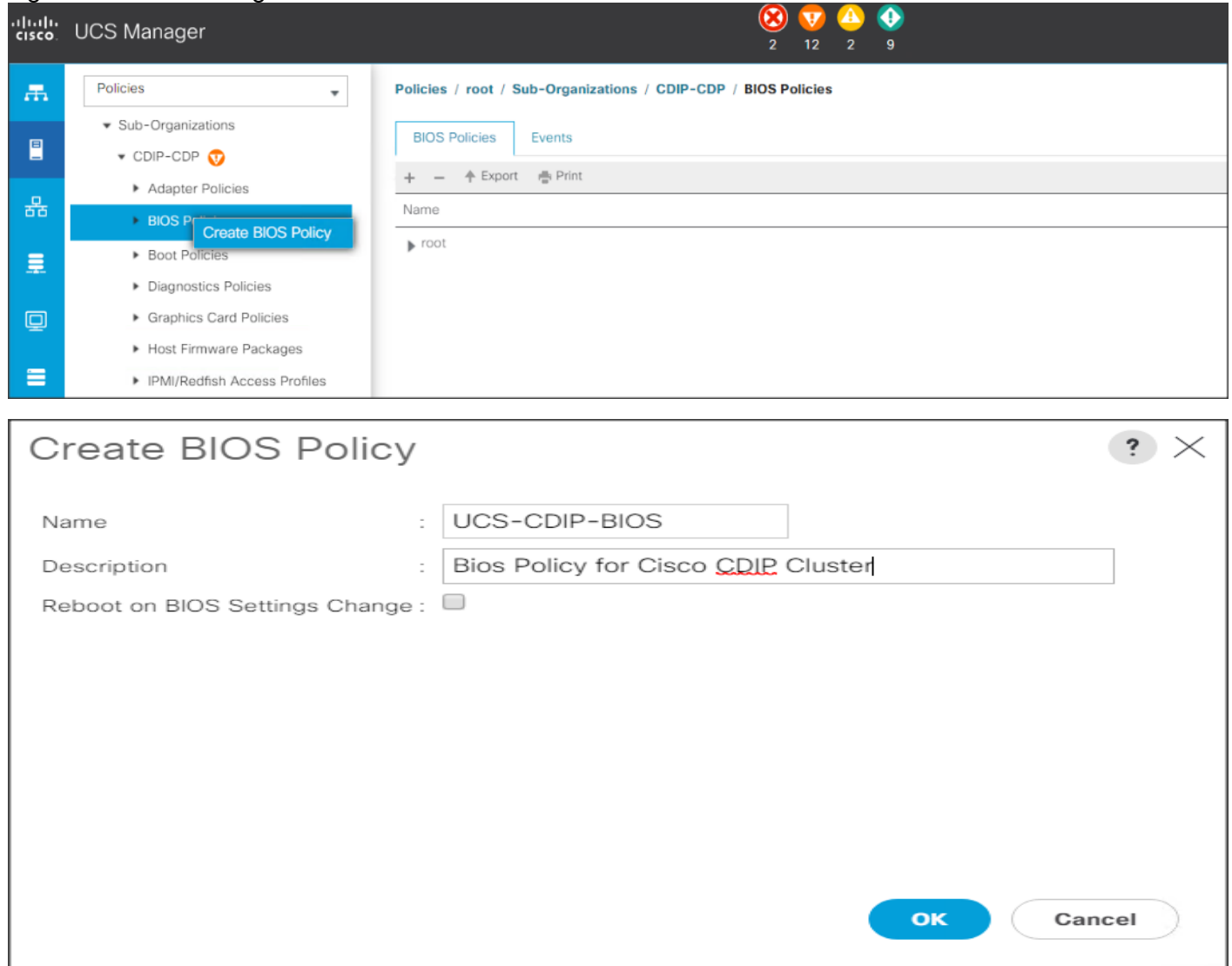
OK **Cancel**

Create Server BIOS Policy

To create a server BIOS policy for the Cisco UCS environment, follow these steps:

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select Policies > root > Sub-Organization > CDIP-CDP > BIOS Policies.
3. Right-click BIOS Policies.
4. Select Create BIOS Policy.
5. Enter the BIOS policy name.

Figure 44 BIOS Configuration



UCS Manager

Policies / root / Sub-Organizations / CDIP-CDP / BIOS Policies / CDIP-CDP-BIOS

Main | **Advanced** | Boot Options | Server Management | Events

Processor | Intel Directed IO | RAS Memory | Serial Port | USB | PCI | QPI | LOM and PCIe Slots | Trusted Platform | Graphics Configuration

Advanced Filter | Export | Print

BIOS Setting	Value
CPU Performance	Enterprise
Core Multi Processing	All
DCPMM Firmware Downgrade	Platform Default
DRAM Clock Throttling	Performance
Direct Cache Access	Enabled
Energy Performance Tuning	Platform Default
Enhanced Intel SpeedStep Tech	Enabled
Execute Disable Bit	Platform Default
Frequency Floor Override	Platform Default
Intel Hyper-Threading Tech	Enabled
Energy Efficient Turbo	Platform Default
Intel Turbo Boost Tech	Enabled
Intel Virtualization Technology	Disabled
Intel Speed Select	Platform Default

Add | Delete | Info

Save Changes | Reset Values

UCS Manager

Policies / root / Sub-Organizations / CDIP-CDP / BIOS Policies / CDIP-CDP-BIOS

Main | **Advanced** | Boot Options | Server Management | Events

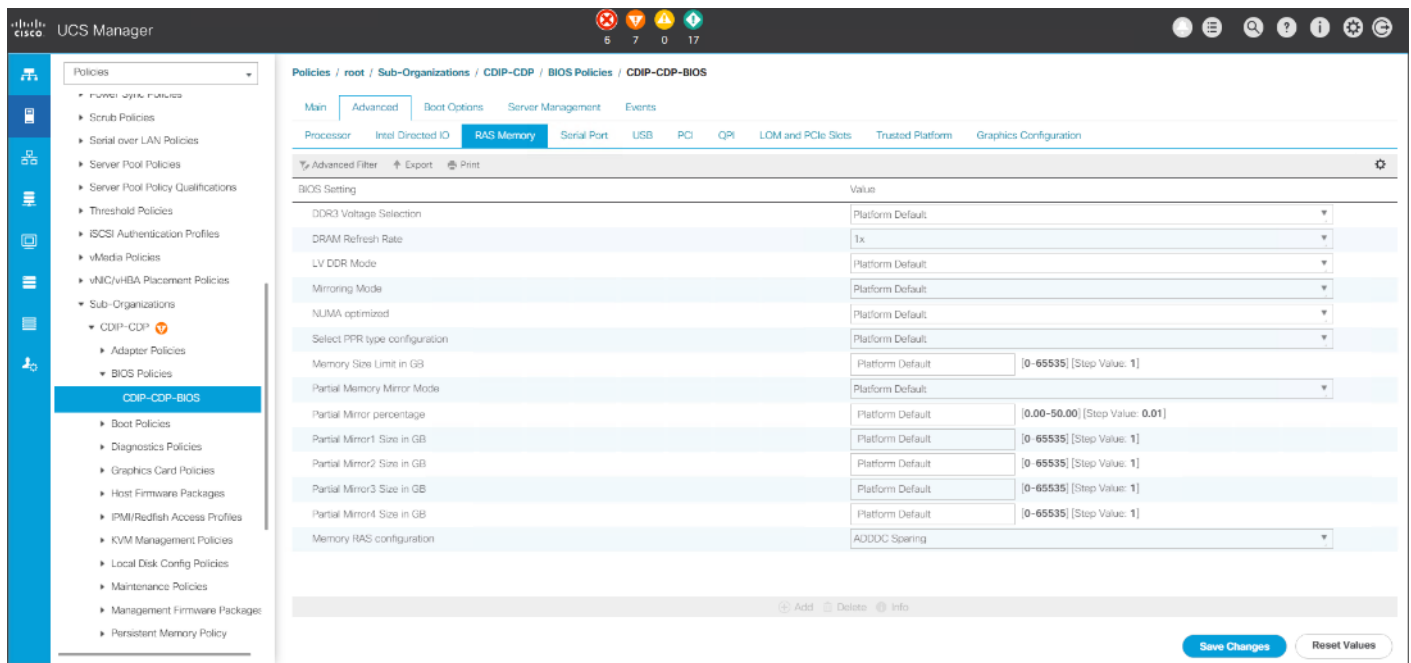
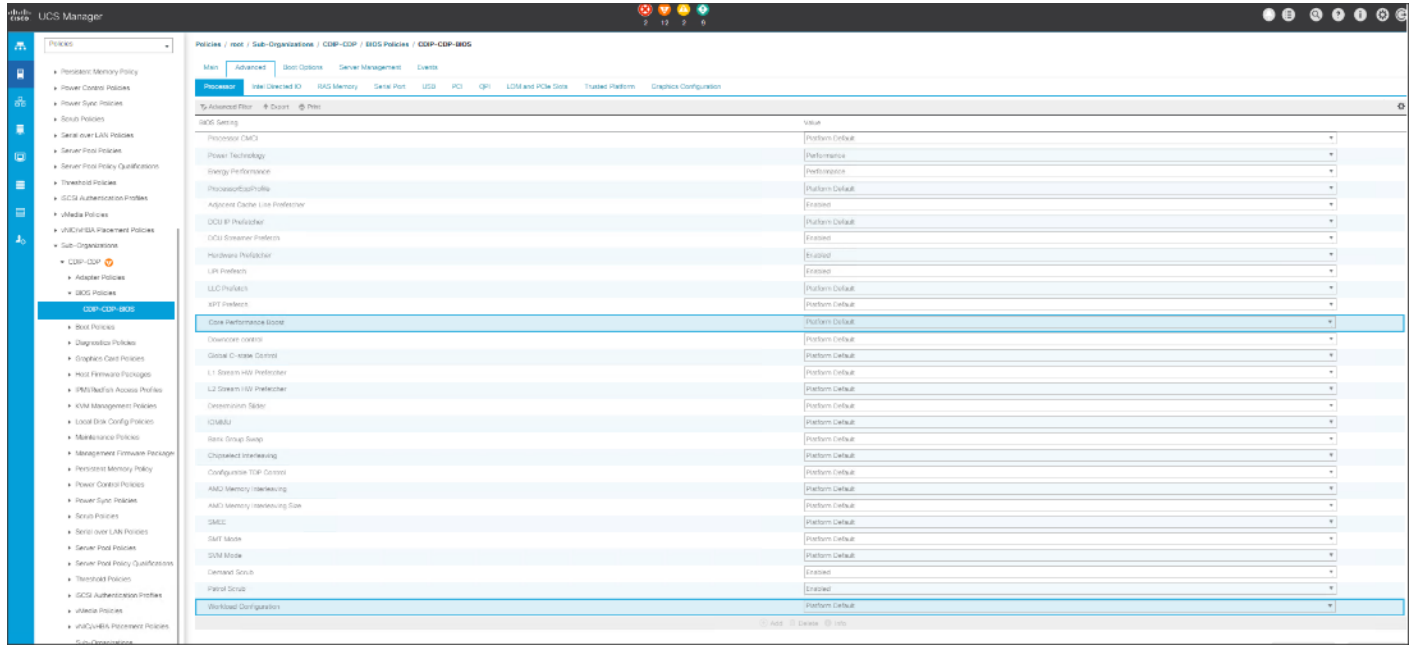
Processor | Intel Directed IO | RAS Memory | Serial Port | USB | PCI | QPI | LOM and PCIe Slots | Trusted Platform | Graphics Configuration

Advanced Filter | Export | Print

BIOS Setting	Value
Local X2 Apic	Platform Default
Max Variable MTRR Setting	Platform Default
P STATE Coordination	HW ALL
Package C State Limit	Platform Default
Autonomous Core C-state	Platform Default
Processor C State	Disabled
Processor C1E	Disabled
Processor C3 Report	Disabled
Processor C6 Report	Disabled
Processor C7 Report	Disabled
Processor CMC1	Platform Default
Power Technology	Performance
Energy Performance	Performance

Add | Delete | Info

Save Changes | Reset Values



For more information, go to: [Performance Tuning Guide](#).



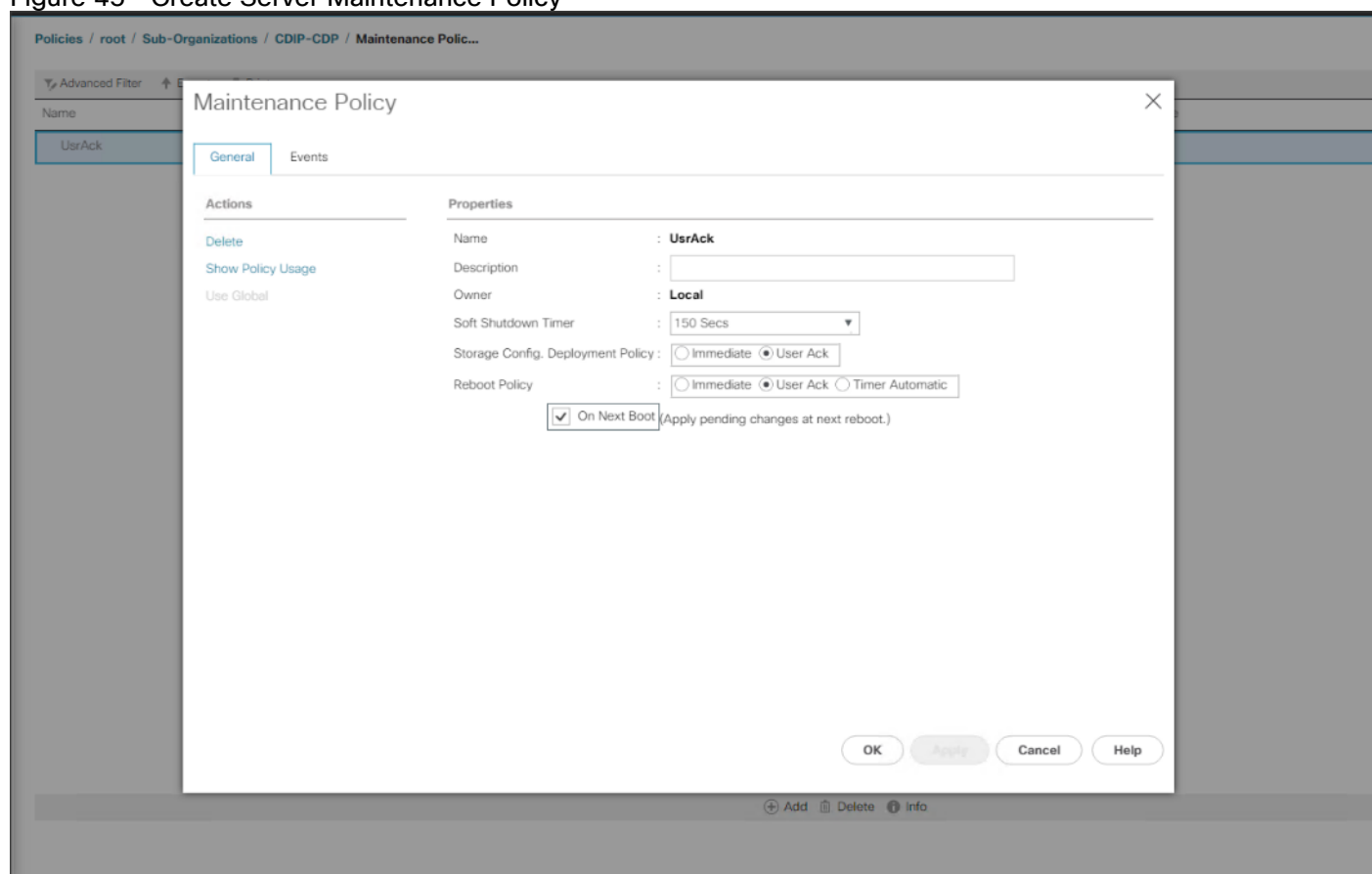
BIOS settings can have a significant performance impact, depending on the workload and the application. The BIOS settings listed in this section is for configurations optimized for best performance which can be adjusted based on the application, performance, and energy efficiency requirements.

Configure Maintenance Policy

To update the default Maintenance Policy, follow these steps:

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select Policies > root > Sub-Organization > CDIP-CDP> Maintenance Policies.
3. Right-click Maintenance Policies to create a new policy.
4. Enter name for Maintenance Policy.
5. Change the Reboot Policy to User Ack.
6. Click Save Changes.
7. Click OK to accept the change.

Figure 45 Create Server Maintenance Policy



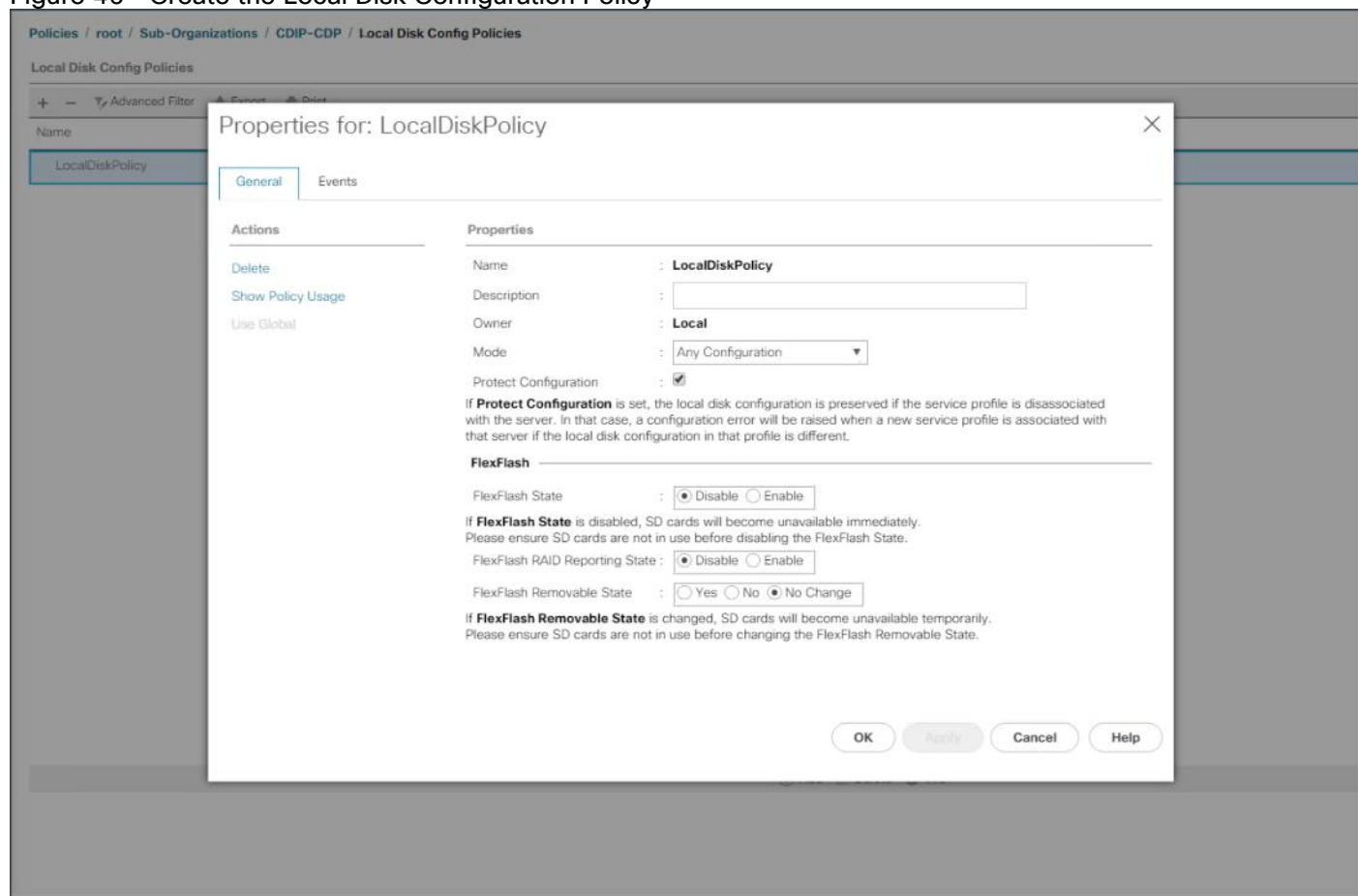
Create the Local Disk Configuration Policy

To create local disk configuration in the Cisco UCS Manager GUI, follow these steps:

1. Select the Servers tab in the Cisco UCS Manager GUI.
2. Select Policies > root > Sub-Organization > CDIP-CDP> Local Disk Config Policies.
3. Right-click Local Disk Config Policies and Select Create Local Disk Config Policies.

4. Enter UCS-Boot as the local disk configuration policy name.
5. Change the Mode to Any Configuration. Check the Protect Configuration box.
6. Keep the FlexFlash State field as default (Disable).
7. Keep the FlexFlash RAID Reporting State field as default (Disable).
8. Click OK to complete the creation of the Local Disk Configuration Policy.
9. Click OK.

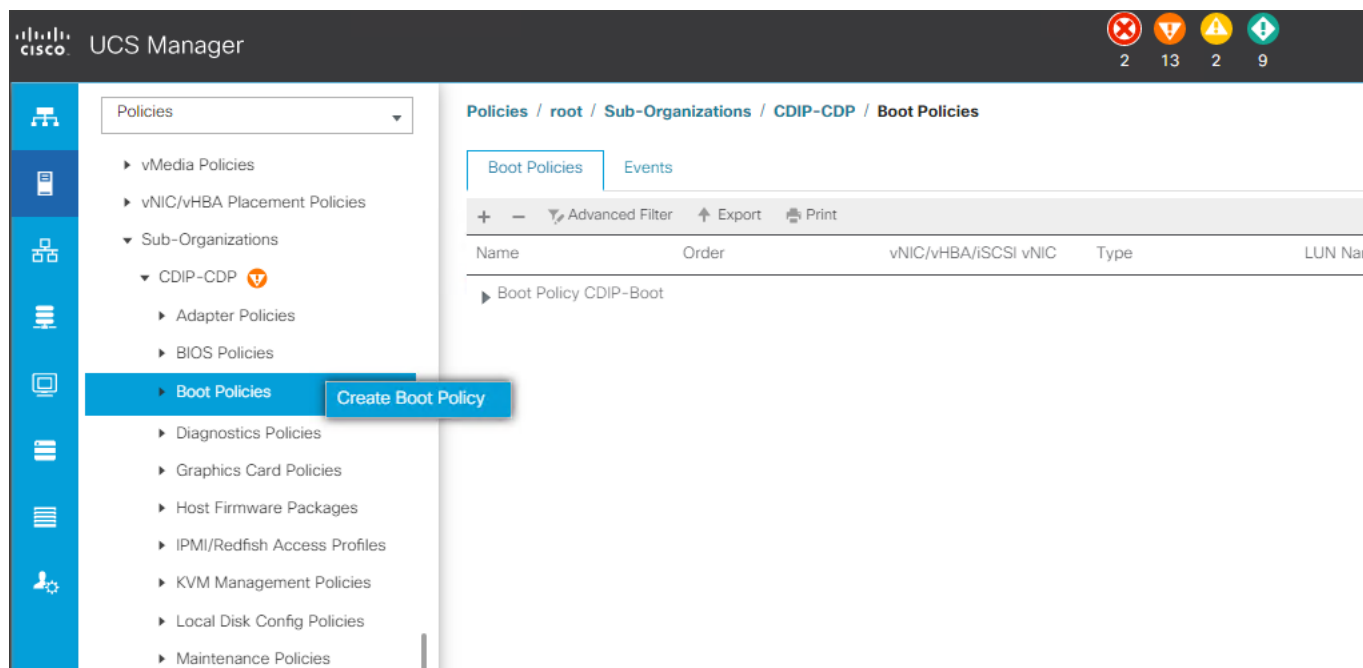
Figure 46 Create the Local Disk Configuration Policy



Create Boot Policy

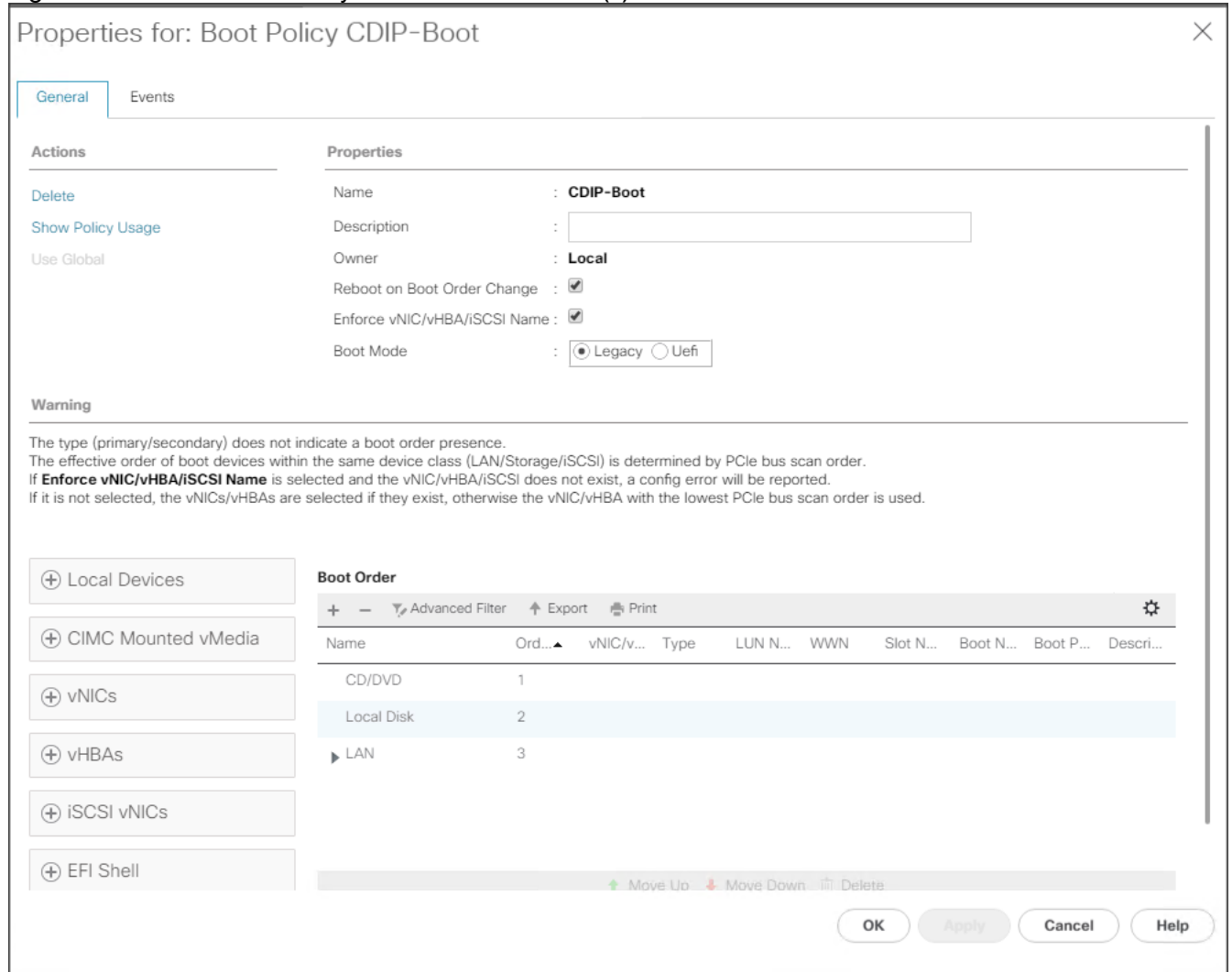
To create boot policies within the Cisco UCS Manager GUI, follow these steps:

1. Select the Servers tab in the Cisco UCS Manager GUI.
2. Select Policies > root.
3. Right-click the Boot Policies.
4. Select Create Boot Policy.



5. Enter ucs for the boot policy name.
6. (Optional) enter a description for the boot policy.
7. Keep the Reboot on Boot Order Change check box unchecked.
8. Keep Enforce vNIC/vHBA/iSCSI Name check box checked.
9. Keep Boot Mode Default (Legacy).
10. Expand Local Devices > Add CD/DVD and select Add Local CD/DVD.
11. Expand Local Devices and select Add Local Disk.
12. Expand vNICs and select Add LAN Boot and enter eth0.
13. Click OK to add the Boot Policy.
14. Click OK.

Figure 47 Create Boot Policy for Cisco UCS Server(s)

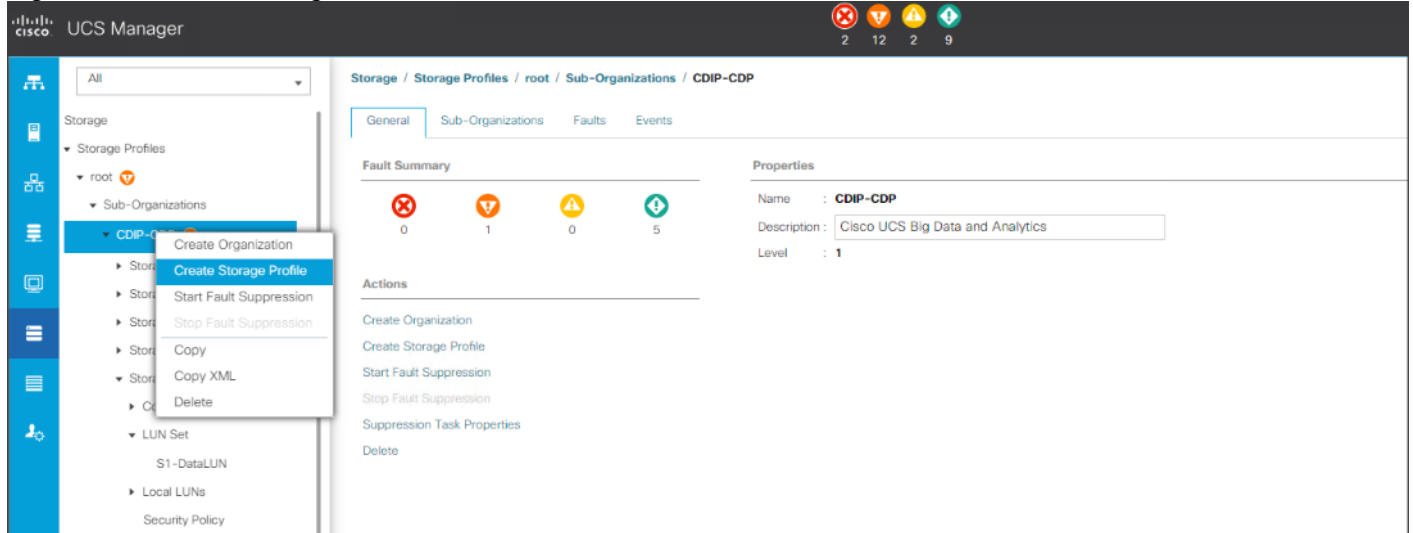


Create Storage Profile for Individual RAID0

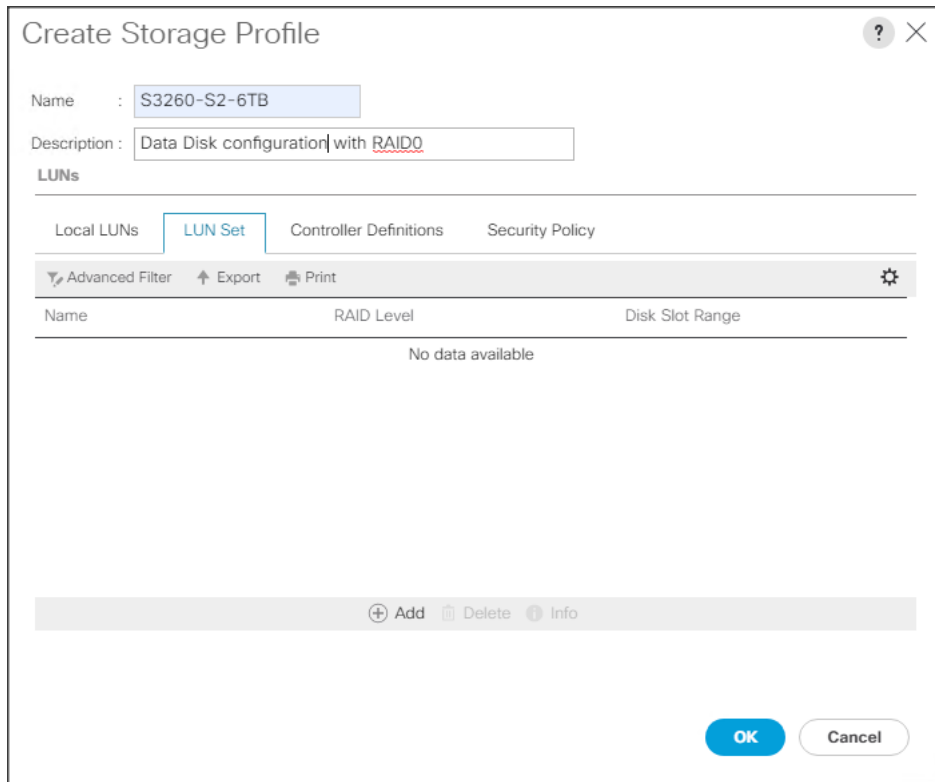
To create the storage profile for the individual RAIDP, follow these steps:


1. On the UCSM navigation page, select the Storage tab.
2. From the Storage Profiles drop-down list, right-click and select Create Storage Profile.

Figure 48 Create Storage Profile



3. Enter a name for the Storage Profile and click the LUN Set tab.
4. Click Add.



 The LUN Set policy configures all disks managed through Cisco UCS S3260 Dual Raid Controller on S3260 and Cisco 12G Modular Raid controller to individual disk RAID0.

5. Select the properties for the LUN set:

- a. Enter a name for LUN set.
- b. Disk Slot Range – 1 – 24/26/56 (Depends on number of drives installed in a server).
- c. Enter Virtual Drive configuration:
 - i. Strip Size(kb) – 1024KB
 - ii. Access Policy – Read Write
 - iii. Read Policy – Read Ahead
 - iv. Write Cache Policy – Write Back Good Bbu
 - v. IO Policy – Direct
 - vi. Drive Cache – Disable

Create LUN Set ? X

Name : S3260-6TB

RAID Level : RAID 0 Striped

Disk Slot Range : 1-14

Virtual Drive Configuration

Strip Size (KB) : Platform Default

Access Policy : Platform Default Read Write Read Only Blocked

Read Policy : Platform Default Read Ahead Normal

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

IO Policy : Platform Default Direct Cached

Drive Cache : Platform Default No Change Enable Disable

Security :

OK Cancel



For a LUN set based configuration, set the JBOD disks to unconfigured by selecting all JBOD disk in Server > Inventory > Disks, right-click and select “Set JBOD to Unconfigured Good.”

Figure 49 Set JBOD Disks to Unconfigured Good

The screenshot shows a server management interface with a table of disks. The table has columns for Name, Size (MB), Serial, Operability, Drive State, Presence, Technology, and Bootable. A context menu is open over the row for Disk 7, showing various actions. The 'Set JBOD to Unconfigured Good' option is highlighted in blue.

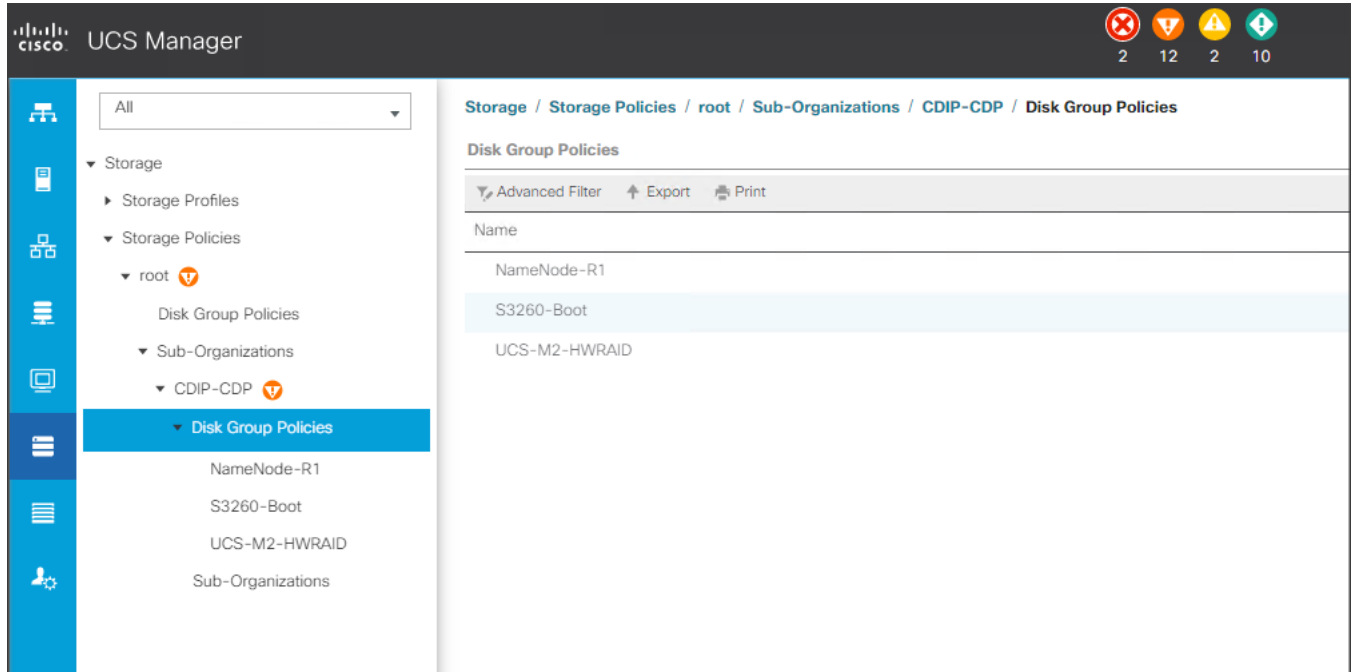
Name	Size (MB)	Serial	Operability	Drive State	Presence	Technology	Bootable
Storage Controller PCH 1							
Storage Controller SAS 1							
Disk 1	5722031	V8H95YRR	Operable	Online	Equipped	HDD	False
Disk 2	5722031	V8H953NR	Operable	Online	Equipped	HDD	False
Disk 3	5722031	V8H95Z6R	Operable	Online	Equipped	HDD	False
Disk 4	5722031	V8H93HTR	Operable	Online	Equipped	HDD	False
Disk 5	5722031	V8GDJ4WR	Operable	Online	Equipped	HDD	False
Disk 6		V8H14BJR	Operable	Online	Equipped	HDD	False
Disk 7		V8H97WKR	Operable	Online	Equipped	HDD	False
Disk 201		18431F0A7720	Operable	Online	Equipped	SSD	False
Disk 202		18431F0A771F	Operable	Online	Equipped	SSD	False

Create Storage Policy and Storage Profile

To create a Storage Profile with multiple RAID LUNs, create Storage Policies and attach them to a Storage Profile.

To create a Storage Policy and attach them to a Storage Profile, follow these steps:

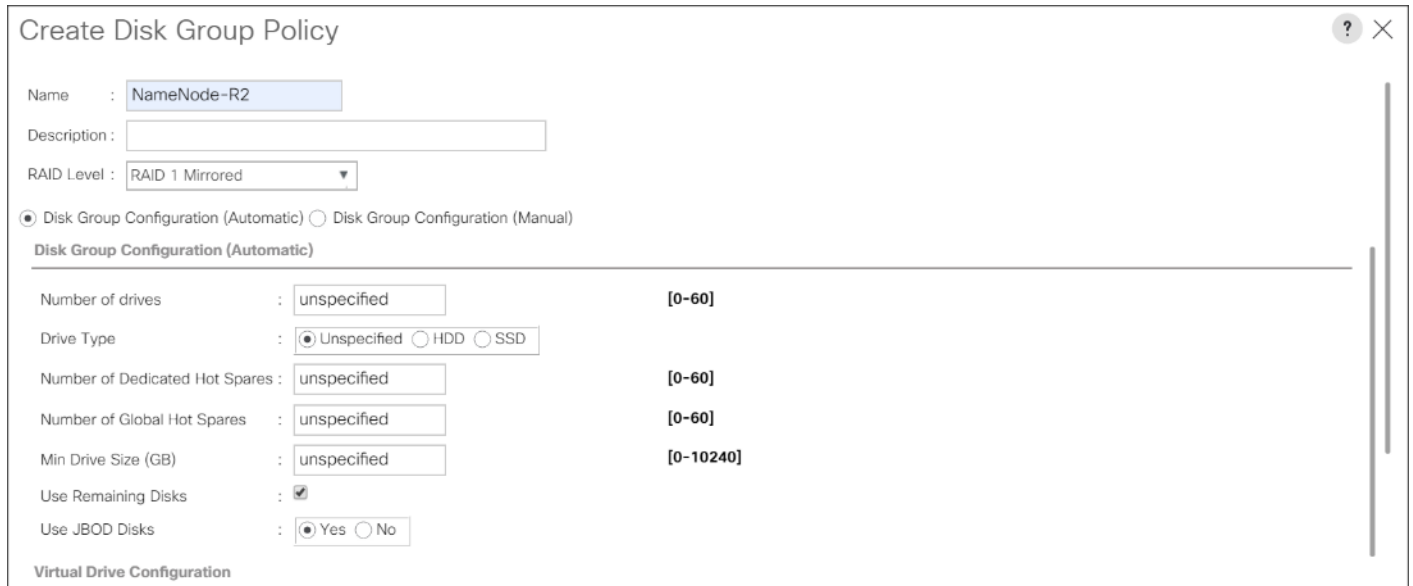
1. Go to the Storage tab on the left side panel selection, select "Storage Policies."
2. From the Storage Policies drop-down list, select and right-click "Disk Group Policies". Select "Create Disk Group Policy."



3. Enter a name for Disk Group Policy and select RAID level.

4. Select “Disk Group Configuration” (Automatic/Manual).

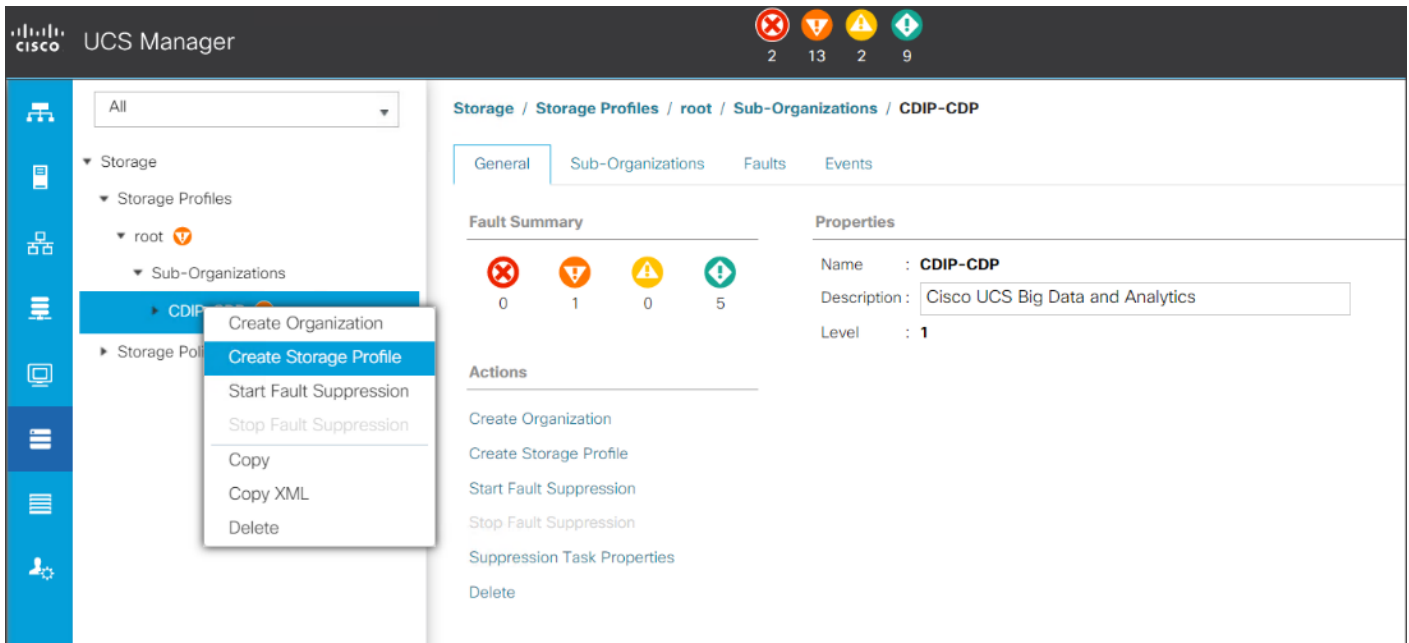
The Disk Group Configuration is shown below:



The Virtual Drive Configuration is shown below:



5. Select Storage Profiles, right-click and select Create Storage Profile.



6. Enter a name for the Storage profile and click Add.

Create Storage Profile ? ✕

Name :

Description :

LUNs

Local LUNs

LUN Set

Controller Definitions

Security Policy

⌵ Advanced Filter
⬆ Export
🖨 Print
⚙

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

⊕ Add
🗑 Delete
ℹ Info

OK
Cancel

7. Enter a Local LUN name and select Auto Deploy.
8. Check the box for Expand to Available and from the drop-down list select the storage policy you want to attach to the Storage Profile. Click OK.

Create Local LUN ? X

Create Local LUN Prepare Claim Local LUN

Name :

Size (GB) : **[0-245760]**

Fractional Size (MB) :

Auto Deploy : Auto Deploy No Auto Deploy

Expand To Available :

Select Disk Group Configuration: Create Disk Group Policy

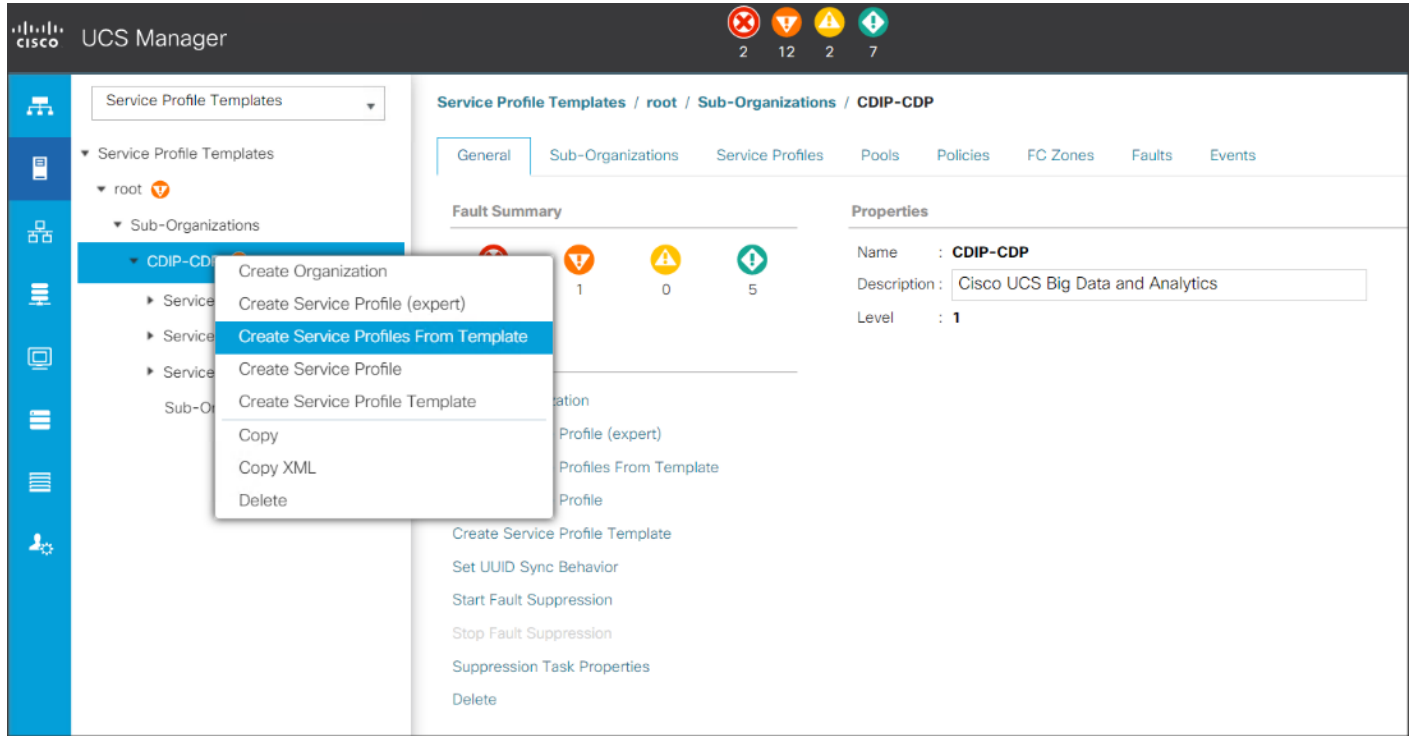


For Cisco UCS S3260, we created a Storage Profile with a Storage Policy to create a Boot LUN and attached it to a Storage Profile as shown above. The LUN set policy for an individual server node (server node 1 and server node 2) to create an individual RAID0.

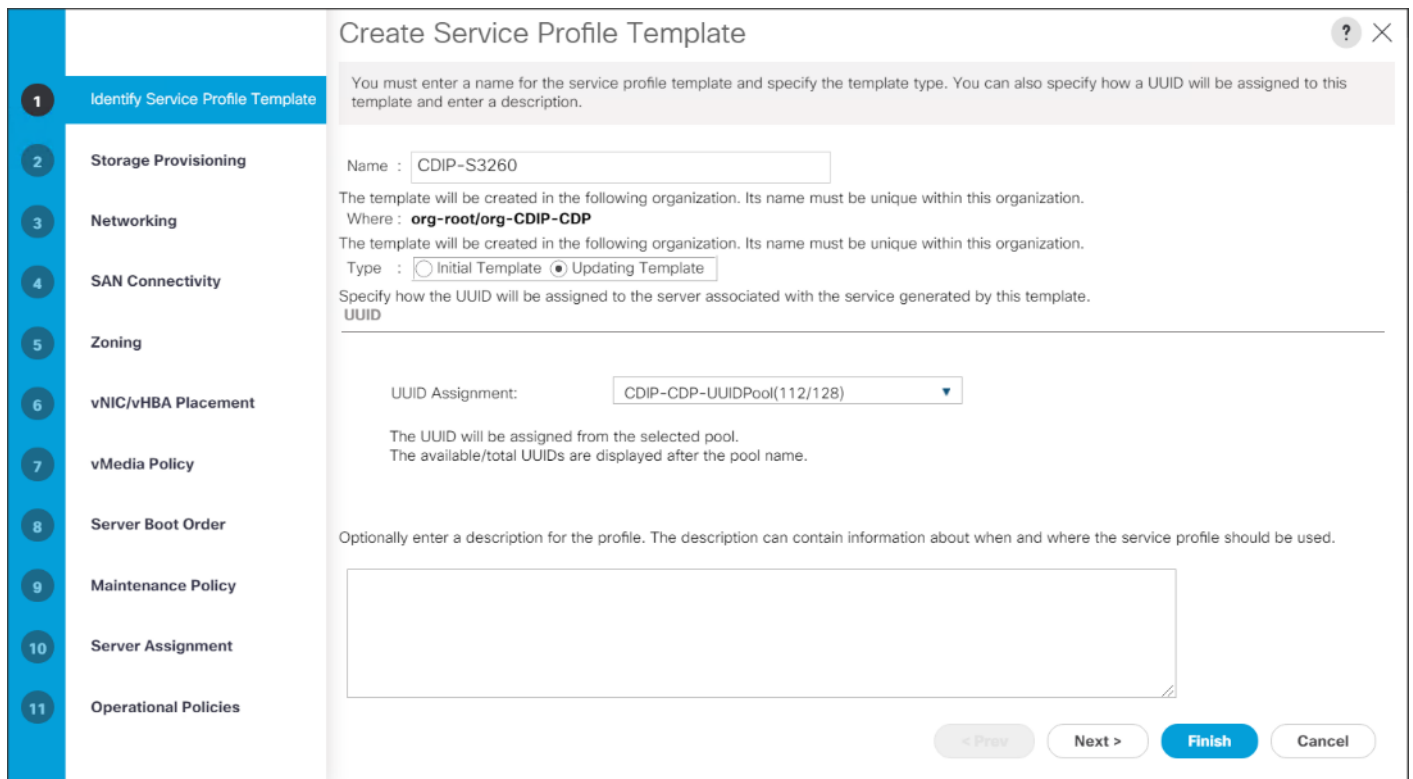
Create Service Profile Template

To create a service profile template, follow these steps:

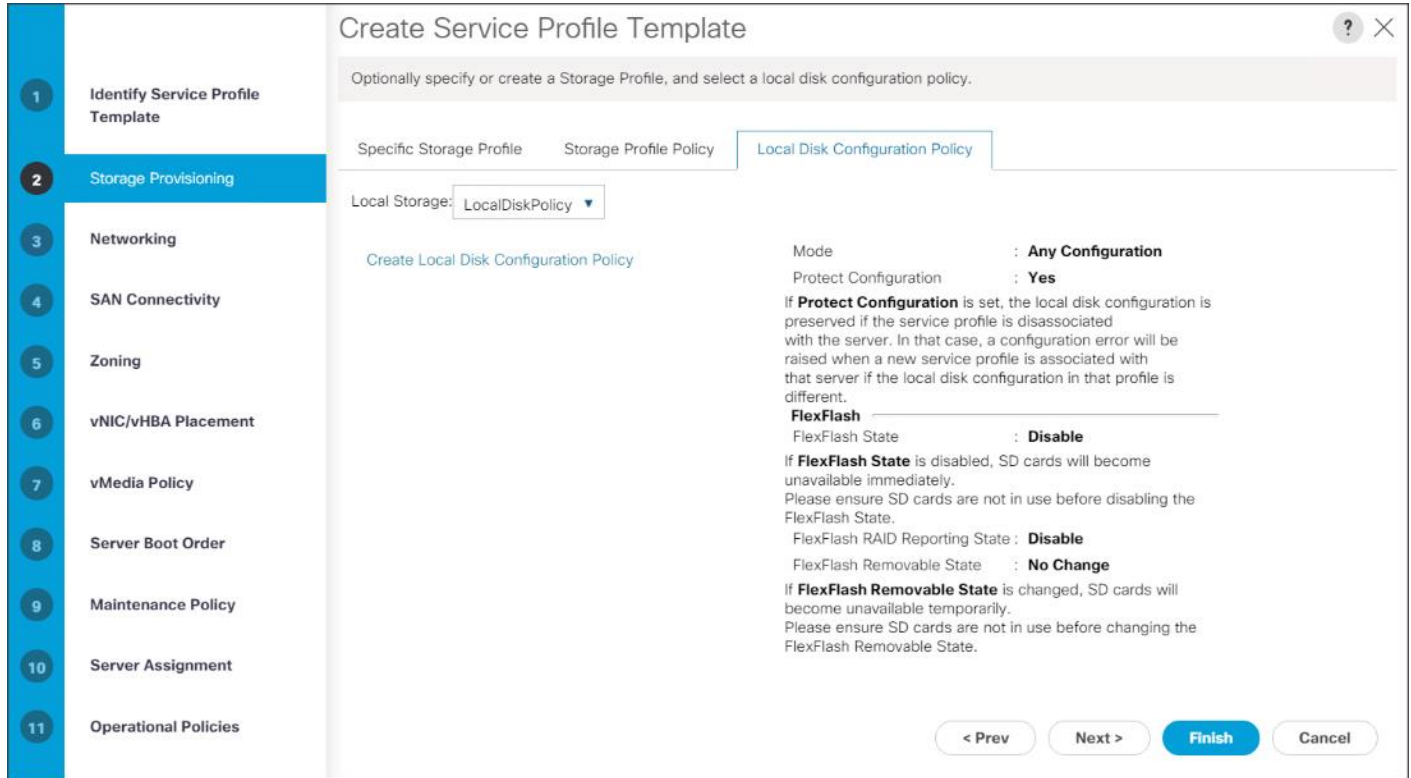
1. In the Cisco UCS Manager, go to Servers > Service Profile Templates > root Sub Organization > CDIP-CDP> and right-click "Create Service Profile Template" as shown below.



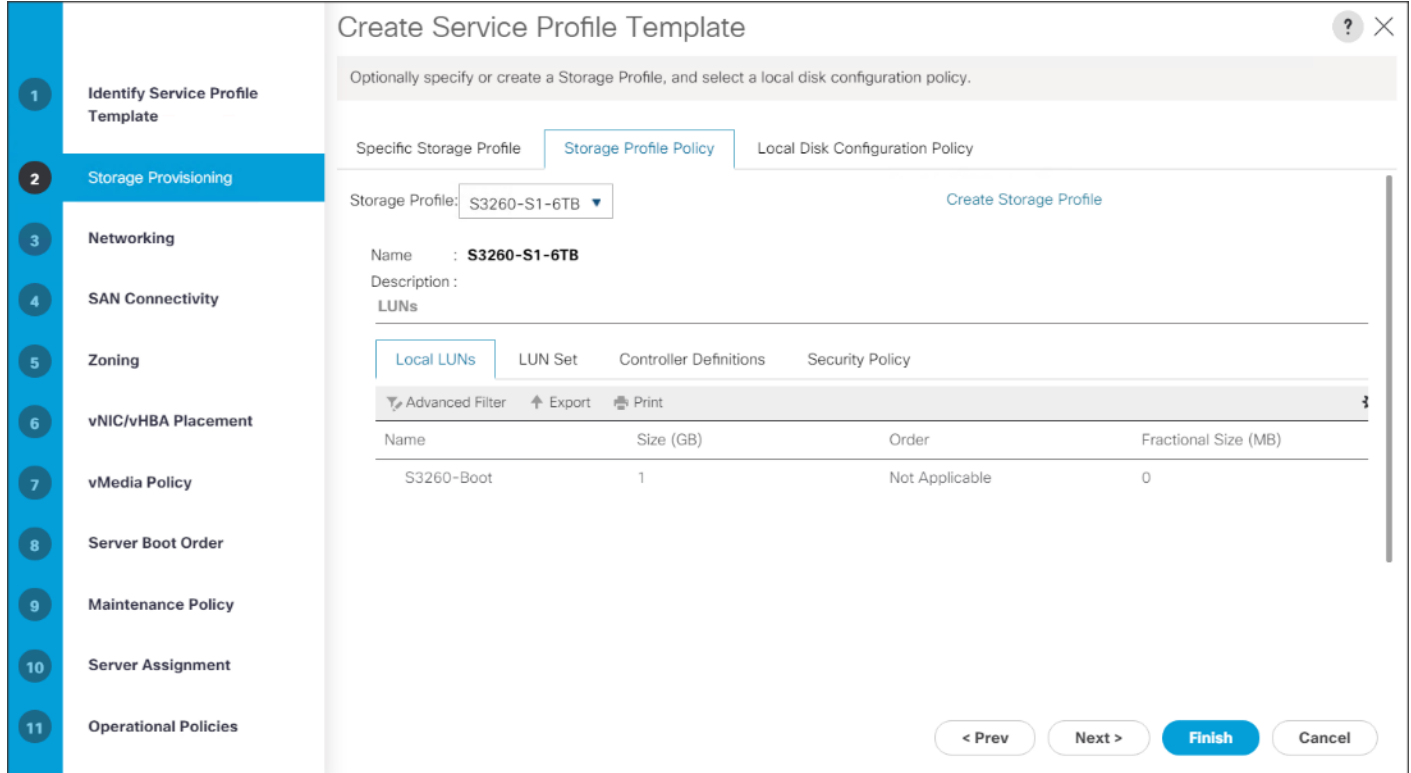
2. Enter the Service Profile Template name, Updating Template as type of template and select the UUID Pool that was created earlier. Click Next.



3. Select Local Disk Configuration Policy tab and select Local Storage policy from the drop-down list.



4. On Storage Profile Policy, select the Storage Profile to attach with the server.





Based on the server model or the role of the server, we created and attached a Storage Profile for NameNode(s), DataNode(s) and Cisco UCS S3260 Storage server in different Service Profile Template for each.

- In the networking window, select Expert and click Add to create vNICs. Add one or more vNICs that the server should use to connect to the LAN.

The screenshot shows the 'Create Service Profile Template' wizard in the Networking step. The left sidebar lists 11 steps, with 'Networking' selected. The main panel contains the following elements:

- Optional LAN configuration section with a dropdown for 'Dynamic vNIC Connection Policy' and a link to 'Create Dynamic vNIC Connection Policy'.
- Configuration options for LAN connectivity: Simple, Expert (selected), No vNICs, and Use Connectivity Policy.
- A table for vNICs with columns: Name, MAC Address, Fabric ID, and Native VLAN. The table is currently empty with the text 'No data available'.
- Buttons for 'Delete', 'Add', and 'Modify'.
- A section for 'iSCSI vNICs' with an 'Add' button.
- Navigation buttons: '< Prev', 'Next >', 'Finish', and 'Cancel'.

- In the create vNIC menu as vNIC name.
- Select vNIC Template as vNIC0 and Adapter Policy as Linux.

Create vNIC ? X

Name :

Use vNIC Template :

Redundancy Pair :

vNIC Template :

Peer Name :

[Create vNIC Template](#)

Adapter Performance Profile

Adapter Policy :

[Create Ethernet Adapter Policy](#)

Create Service Profile Template

Optionally specify LAN configuration information.

Dynamic vNIC Connection Policy:

[Create Dynamic vNIC Connection Policy](#)

How would you like to configure LAN connectivity?

Simple
 Expert
 No vNICs
 Use Connectivity Policy

Click **Add** to specify one or more vNICs that the server should use to connect to the LAN.

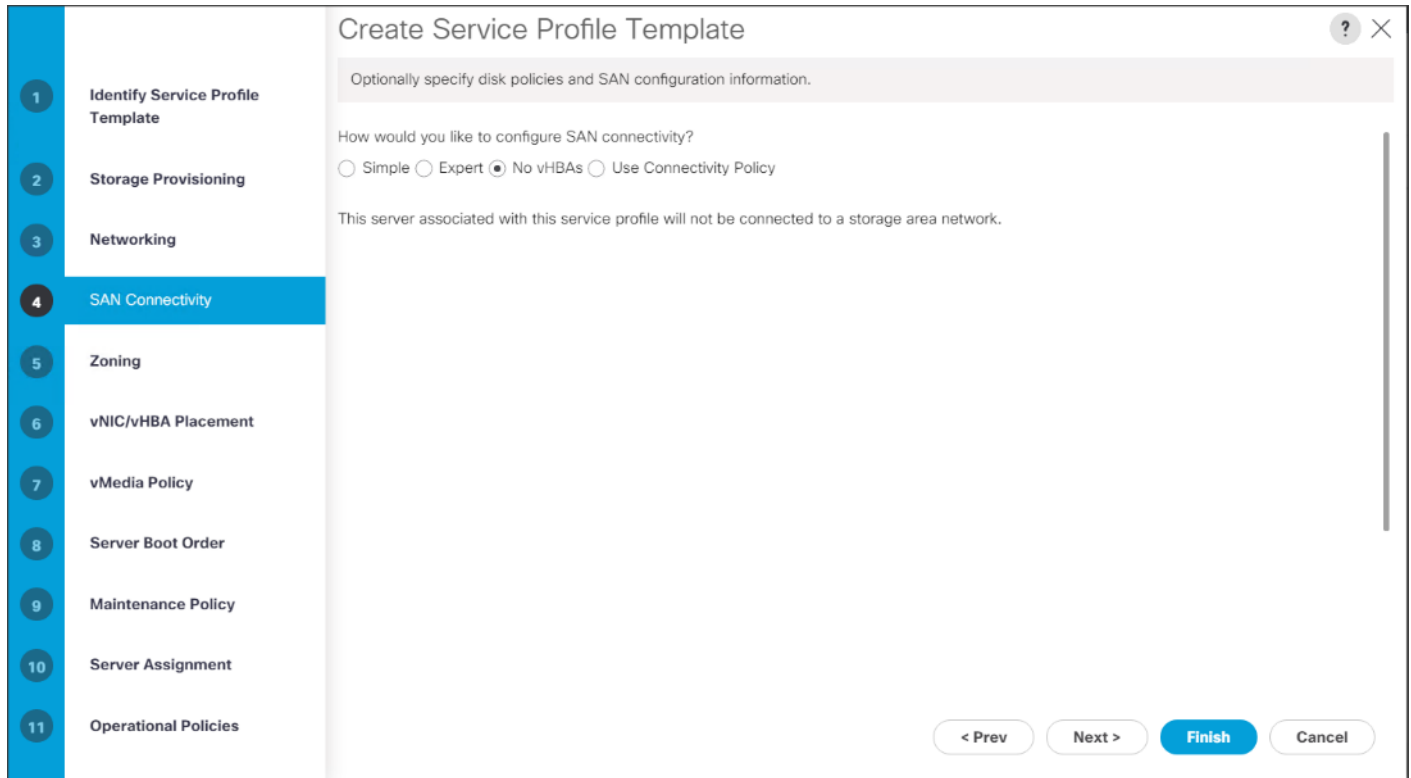
Name	MAC Address	Fabric ID	Native VLAN
vNIC eth0	Derived	derived	

iSCSI vNICs

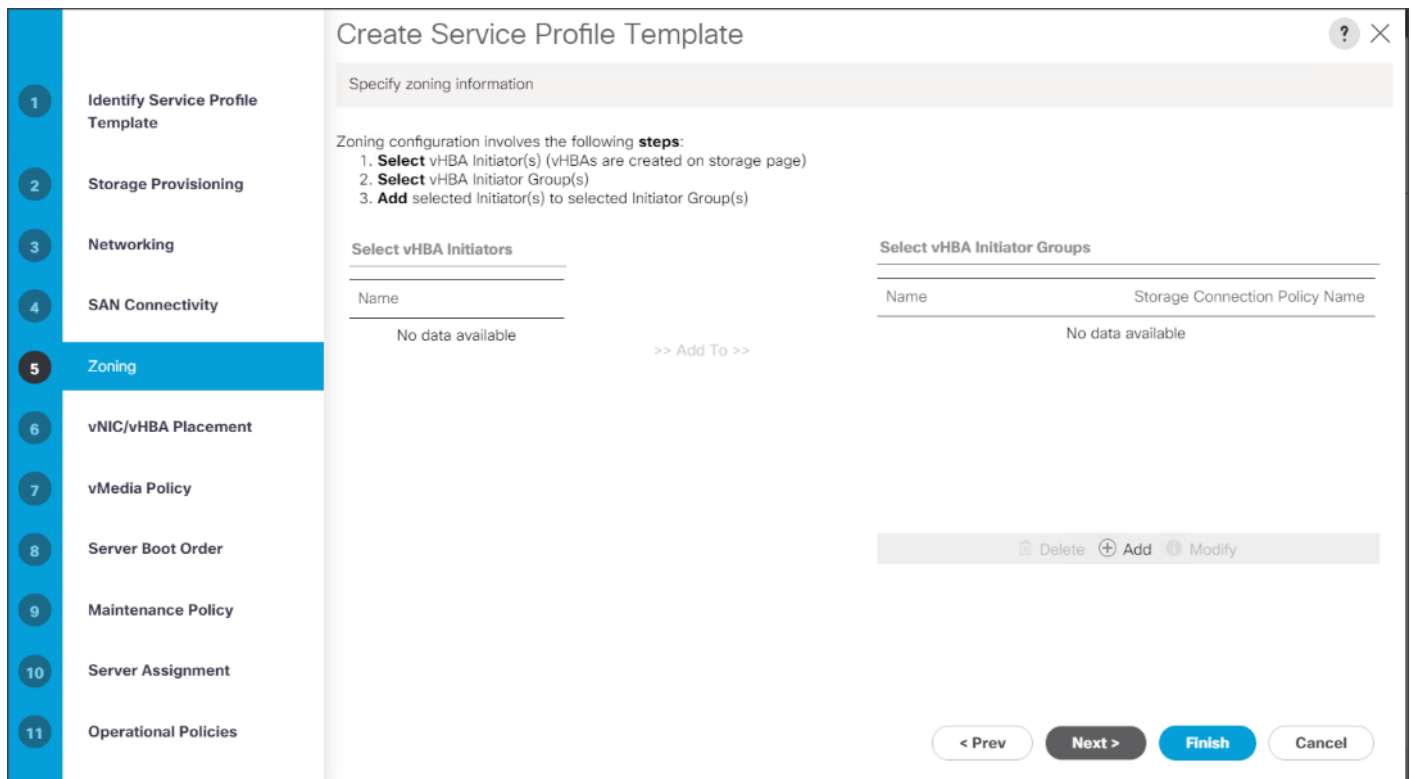


Optionally, Network Bonding can be setup on the vNICs for each host for redundancy as well as for increased throughput.

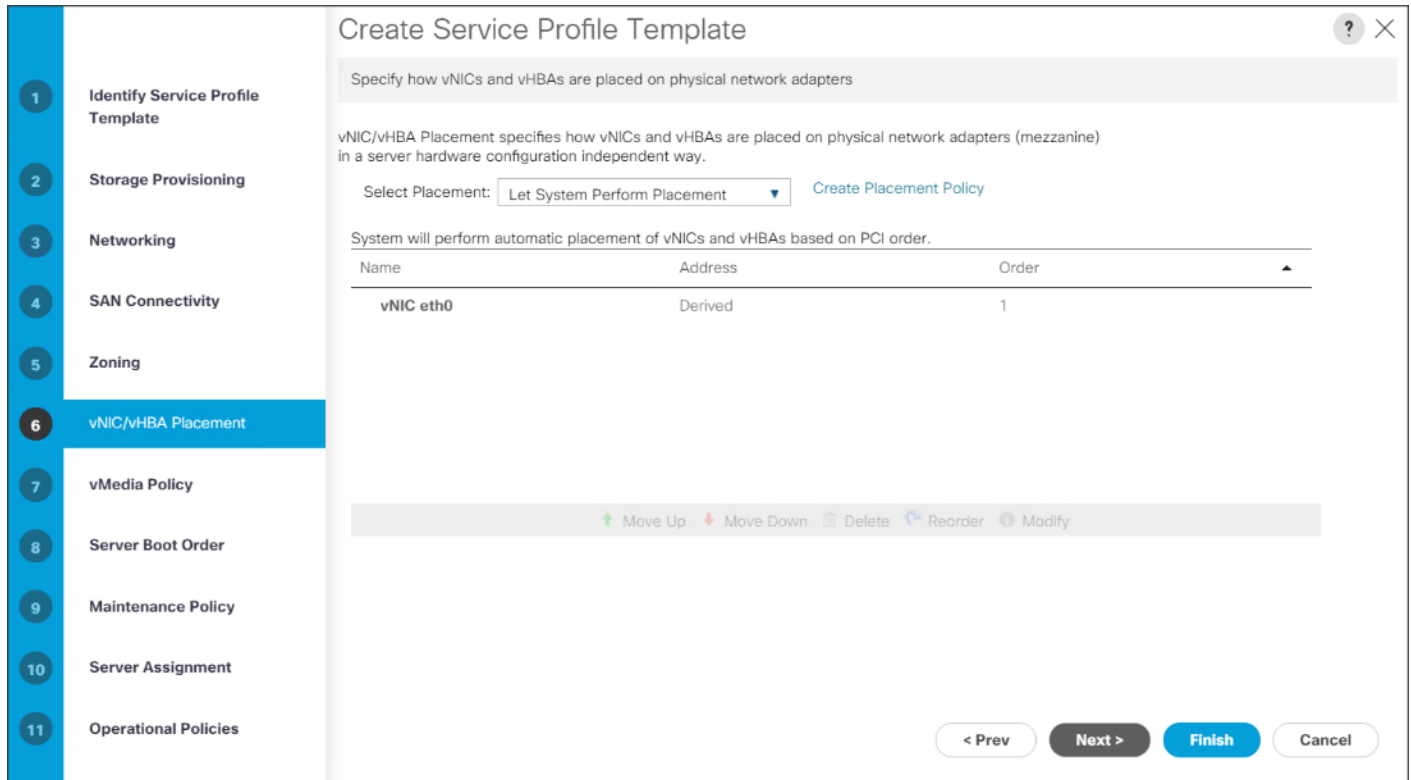
- In the SAN Connectivity menu, select no vHBAs.



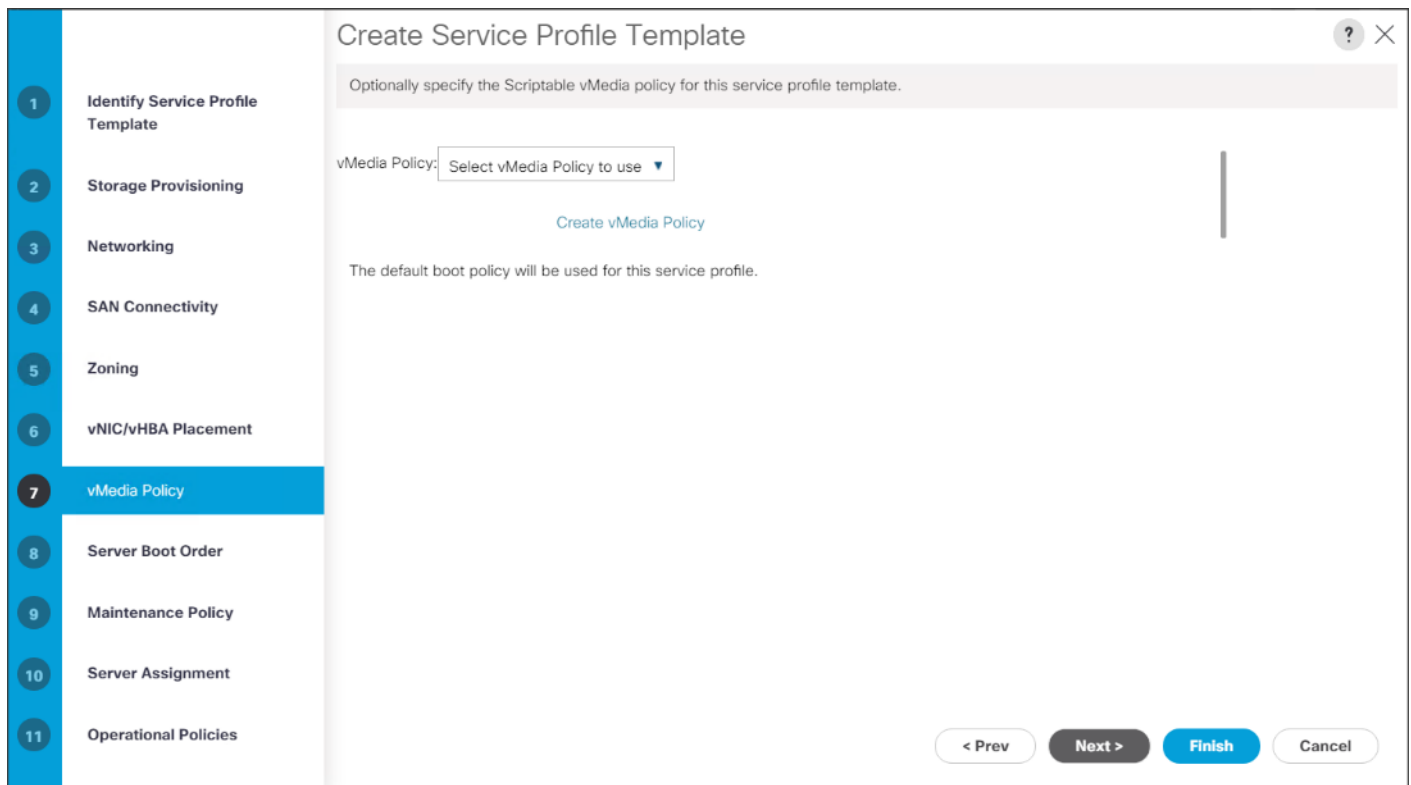
9. Click Next in the Zoning tab.



10. Select Let System Perform Placement for vNIC/vHBA Placement. Click Next.



11. Click Next in the vMedia Policy tab.



12. Select Boot Policy in the Server Boot Order tab.

Create Service Profile Template

Optionally specify the boot policy for this service profile template.

Select a boot policy.

Boot Policy: [Create Boot Policy](#)

Name : **CDIP-Boot**
 Description :
 Reboot on Boot Order Change : **Yes**
 Enforce vNIC/vHBA/iSCSI Name : **Yes**
 Boot Mode : **Legacy**

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

Boot Order

Name	Order	vNIC/vHB...	Type	LUN Name	WWN	Slot Num...	Boot Name	Boot Path	Description
CD/DVD	1								
Local ...	2								
▶ LAN	3								

< Prev Next > **Finish** Cancel

13. Select UserAck maintenance policy, which requires user acknowledgement prior rebooting server when making changes to policy or pool configuration tied to a service profile.

Create Service Profile Template

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

⊖ Maintenance Policy

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

Maintenance Policy: [Create Maintenance Policy](#)

Name : **UsrAck**
 Description :
 Soft Shutdown Timer : **150 Secs**
 Storage Config. Deployment Policy : **User Ack**
 Reboot Policy : **User Ack**

< Prev Next > **Finish** Cancel

14. Select the Server Pool policy to automatically assign a service profile to a server that meets the requirements for server qualification based on the pool configuration. Select Power state when the Service Profile is associated to server
15. On the same page you can configure “Host firmware Package Policy” which helps to keep the firmware in sync when associated to server.

Create Service Profile Template

Optionally specify a server pool for this service profile template.

You can select a server pool you want to associate with this service profile template.

Pool Assignment: [Create Server Pool](#)

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

Up Down

The service profile template is not automatically associated with a server. Either select a server from the list or associate the service profile manually later.

Firmware Management (BIOS, Disk Controller, Adapter)

If you select a host firmware policy for this service profile, the profile will update the firmware on the server that it is associated with. Otherwise the system uses the firmware already installed on the associated server.

Host Firmware Package: [Create Host Firmware Package](#)

< Prev Next > **Finish** Cancel



On the Operational Policy page, we configured the BIOS policy for a Cisco UCS C240 M5 Rack server with the Power Control Policy set to “NoPowerCap” for maximum performance.

Create Service Profile Template

Optionally specify information that affects how the system operates.

BIOS Configuration
 If you want to override the default BIOS settings, select a BIOS policy that will be associated with this service profile
 BIOS Policy :

External IPMI/Redfish Management Configuration
 Management IP Address
 Monitoring Configuration (Thresholds)

Power Control Policy Configuration
 Power control policy determines power allocation for a server in a given power group.
 Power Control Policy : [Create Power Control Policy](#)

Scrub Policy
 KVM Management Policy
 Graphics Card Policy
 Persistent Memory Policy

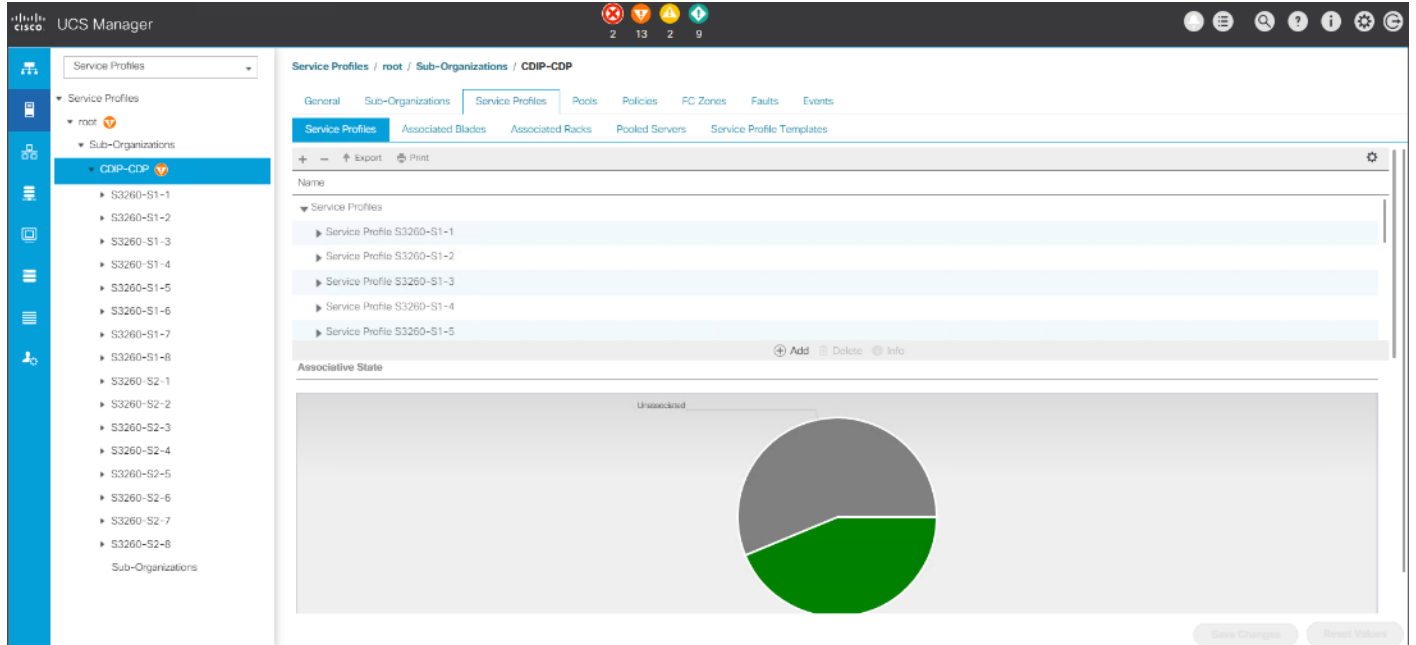
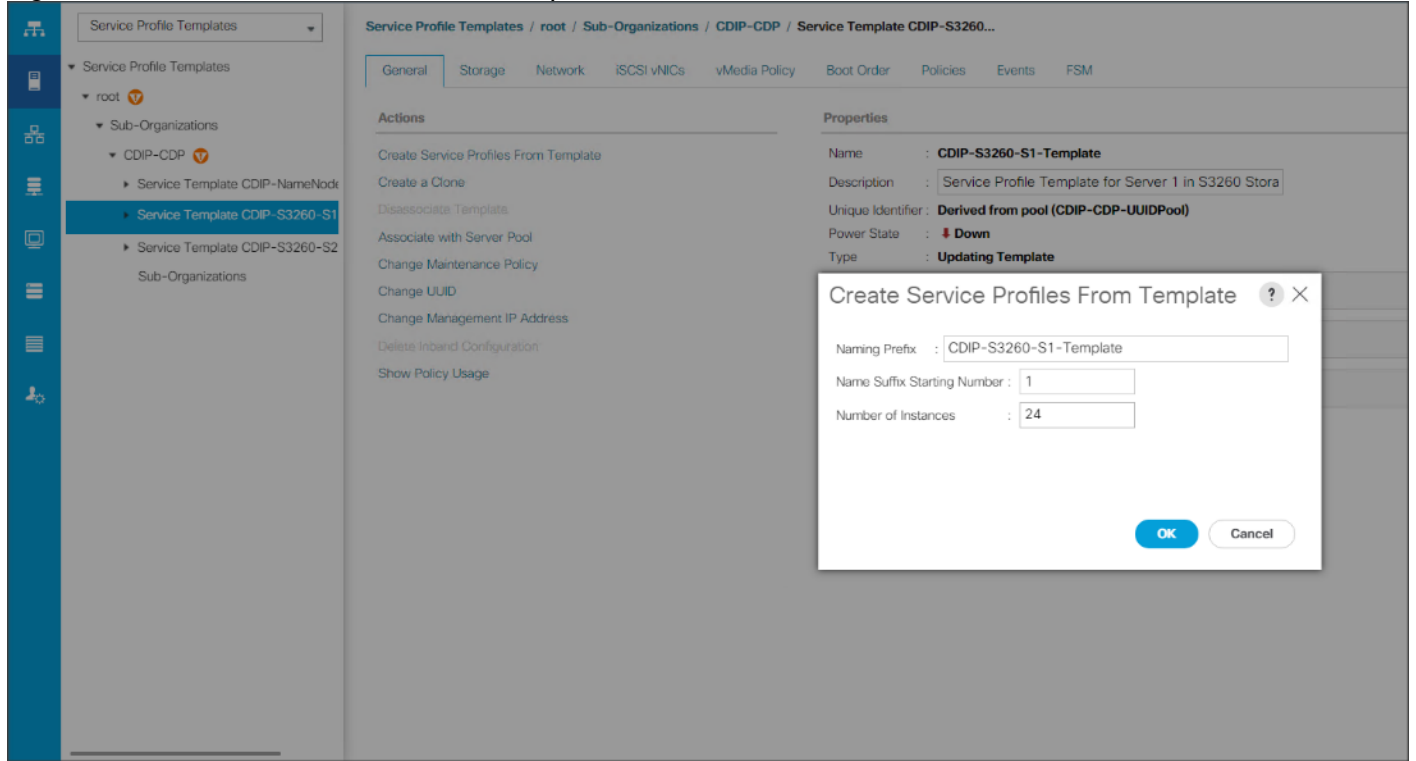
16. Click Finish to create the Service Profile template.


Create Service Profiles from Template

To create a Service Profile from a template, follow these steps:

1. Right-click the Service Profile Template and select Create Service profile from Template.

Figure 50 Create Service Profile from Template



 The Service profile will automatically assign to servers discovered and meets the requirement of Server Pool.

- Repeat this step to create service profile template(s) and service profile(s) according to different deployment scenario.

Install Red Hat Enterprise Linux 7.7

This section provides detailed procedures for installing Red Hat Enterprise Linux Server using Software RAID (OS based Mirroring) on Cisco UCS C240 M5 servers. There are multiple ways to install the RHEL operating system. The installation procedure described in this deployment guide uses KVM console and virtual media from Cisco UCS Manager.



In this study, RHEL version 7.7 DVD/ISO was utilized for OS the installation on Cisco UCS C240 M5 Rack Servers.

To install the Red Hat Enterprise Linux 7.7 operating system, follow these steps:

1. Log into the Cisco UCS Manager.
2. Select the Equipment tab.
3. In the navigation pane expand Rack-Mounts and then Servers.
4. Right-click the server and select KVM console.
5. In the right pane, click the KVM Console >>.

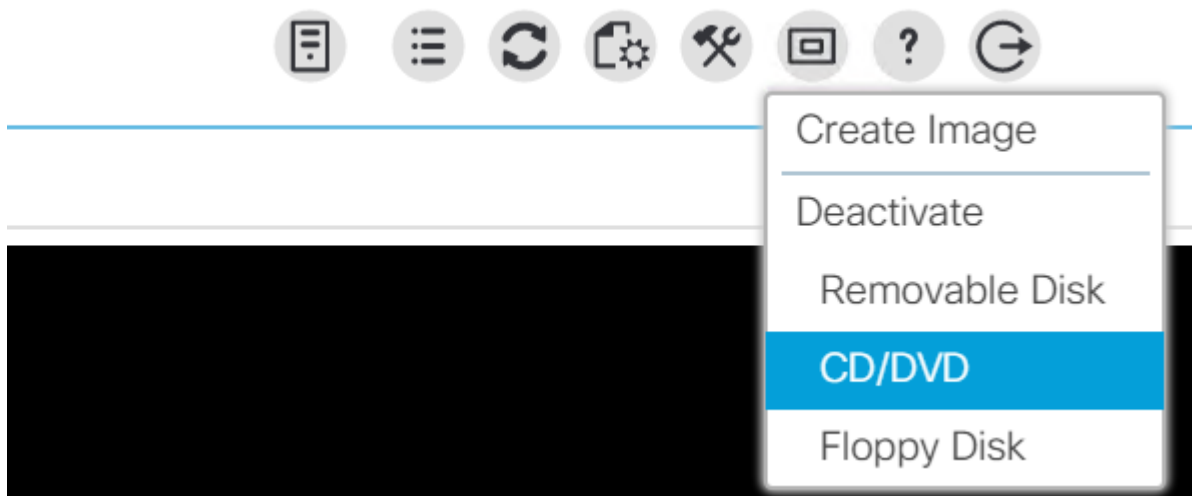
The screenshot displays the Cisco UCS Manager interface. The left navigation pane shows the hierarchy: Equipment > Chassis > Chassis 4 (Name Node) > Servers > Server 2 (Name Node-02). The main content area is titled 'Equipment / Chassis / Chassis 4 (Name Node) / Servers / Server 2'. It features a 'Fault Summary' section with four status indicators (0 errors), a 'Status' section showing 'Overall Status: OK', and a 'Physical Display' image of the server. The 'Actions' section lists various server management tasks. The 'Properties' section on the right provides detailed information:

Property	Value	Property	Value
Slot ID	2	Chassis ID	4
Product Name	Cisco UCS S3260 M5	PID	UCS-S3260-M5SRB
Vendor	Cisco Systems Inc	Serial	FCH23107MB
Revision	0	Health LED	Normal
Manufacturing Date	2019-03-15	Oper Qualifier Reason	
Asset Tag		Locator LED	
Name		FP Buttons	Unlocked
User Label	Name Node-02		
Unique Identifier	cfe5bb0-5345-11ea-0000-abc09000000c		
Service Profile	org-root/org-CDIP-CDP/ls-S3260-S2-4		

6. Click the link to launch the KVM console.
7. Point the cursor over the top right corner and select the Virtual Media tab.
8. Click the Activate Virtual Devices found in Virtual Media tab.



9. Click the Virtual Media tab to select CD/DVD.



10. Select Map Drive in the Virtual Disk Management windows.

11. Browse to the Red Hat Enterprise Linux 7.7 installer ISO image file.



The Red Hat Enterprise Linux 7.7 Server DVD is assumed to be on the client machine.

Virtual Disk Management



CD/DVD rhel-server-7...6_64-dvd.iso

Read Only

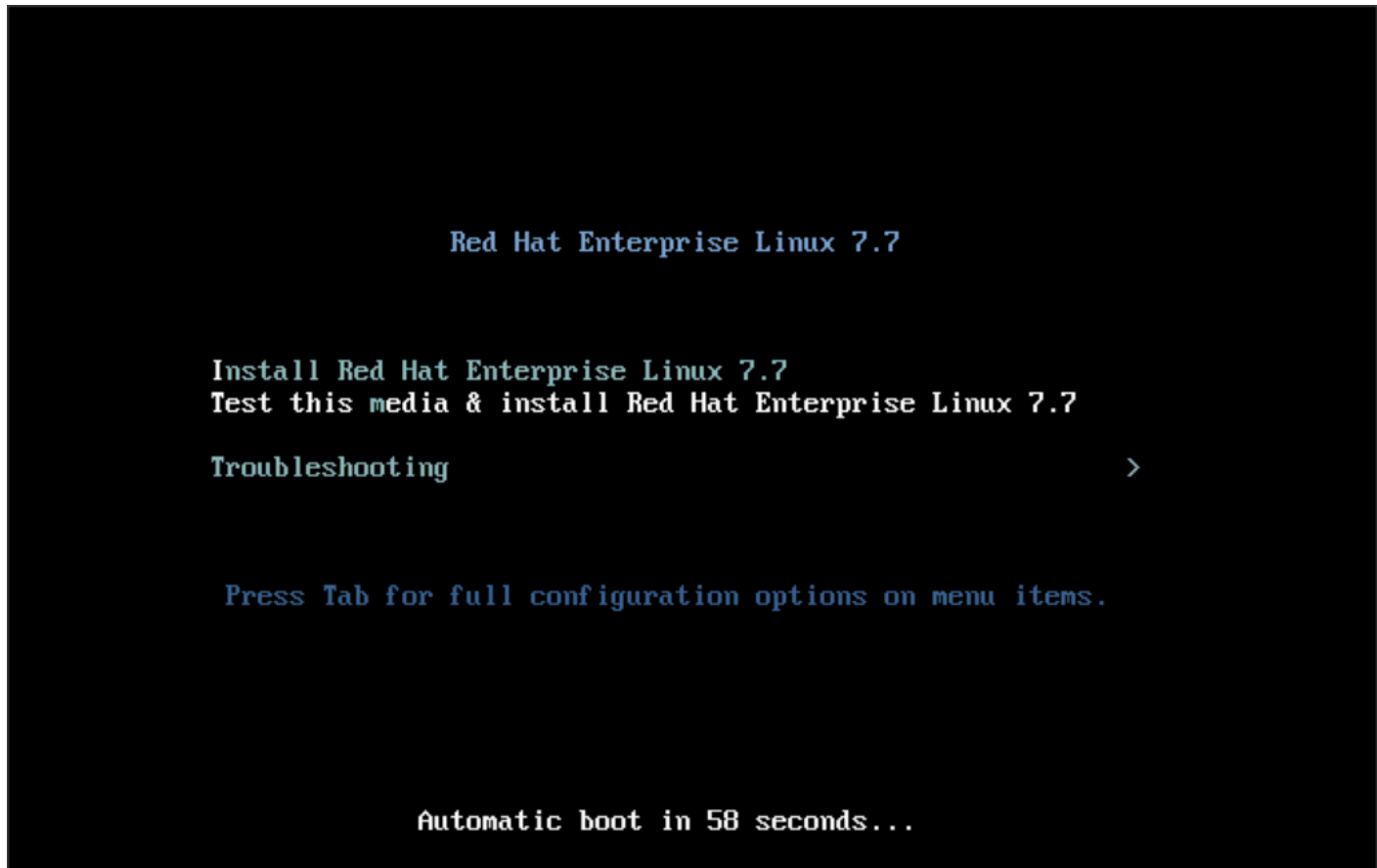
To share files/folders you can drag and drop them in the area below or in the video display area.

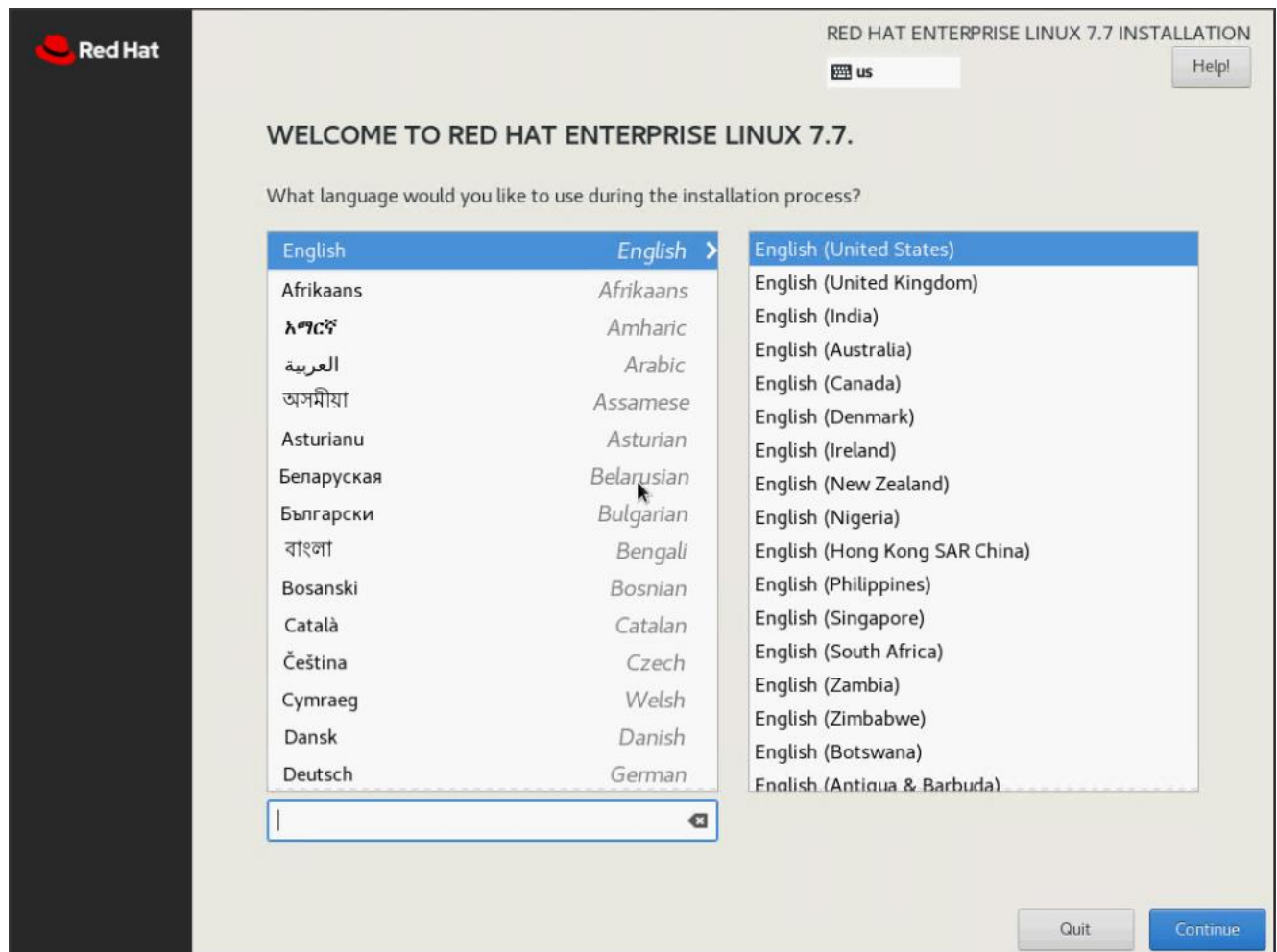
Drop files/folders here

Virtual Disk Management

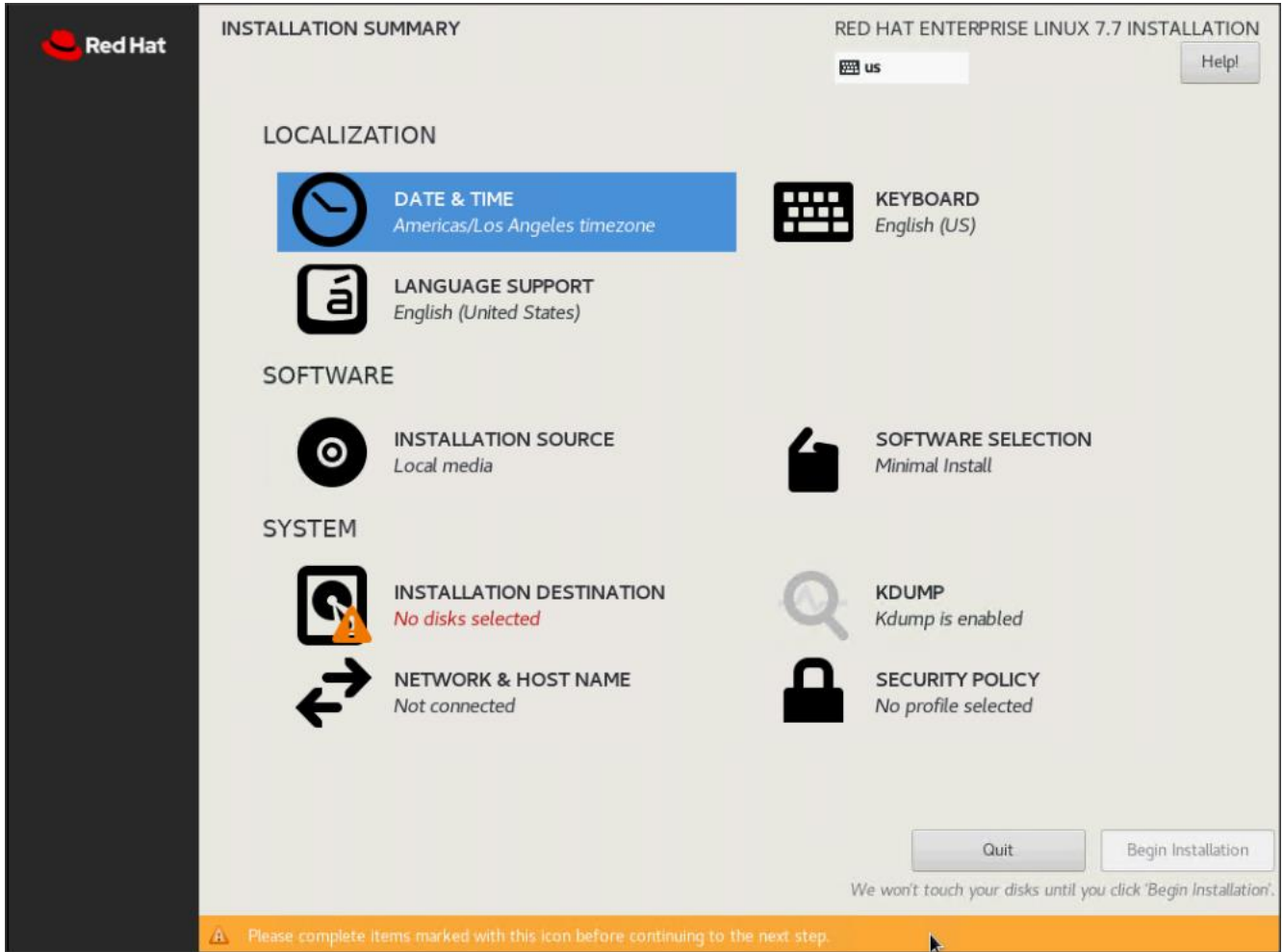


CD/DVD rhel-server-7.7-x86_64-dvd.iso
mapped -- Read Only





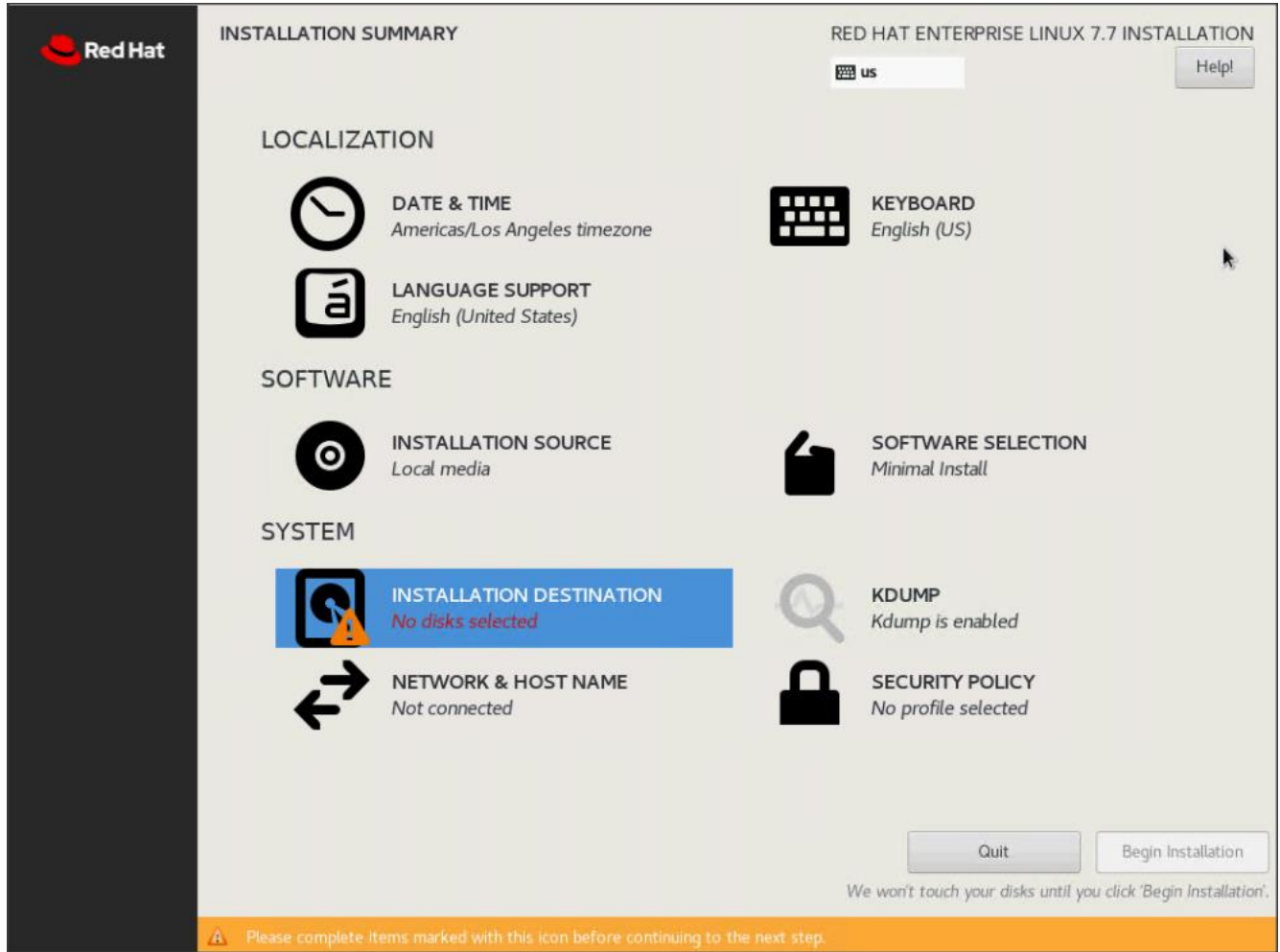
12. Click Open to add the image to the list of virtual media.
13. Select the Installation option from Red Hat Enterprise Linux 7.7
14. Select date and time as shown below.



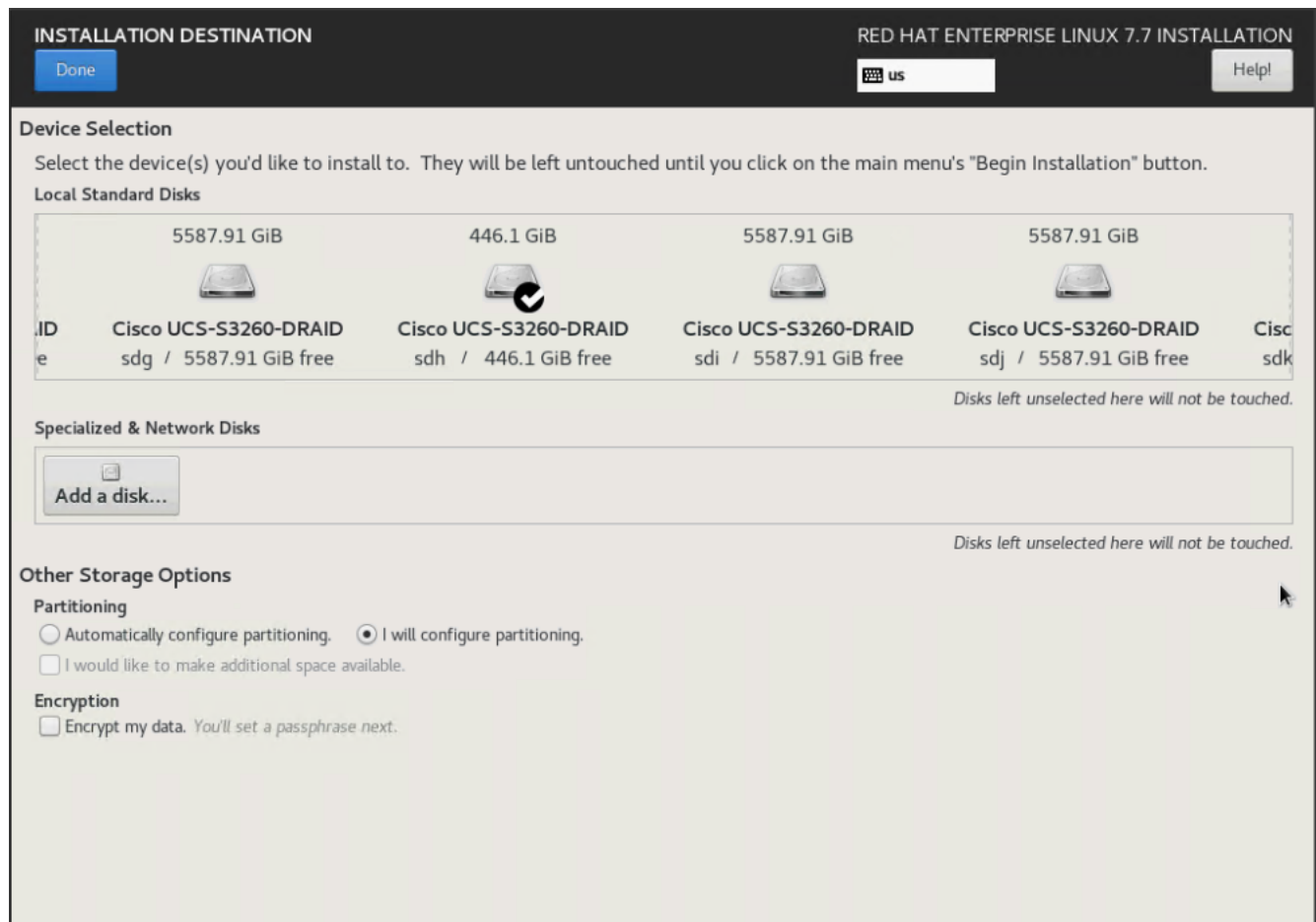
15. Select the location on the map, set the time, and click Done.



16. Click Installation Destination.

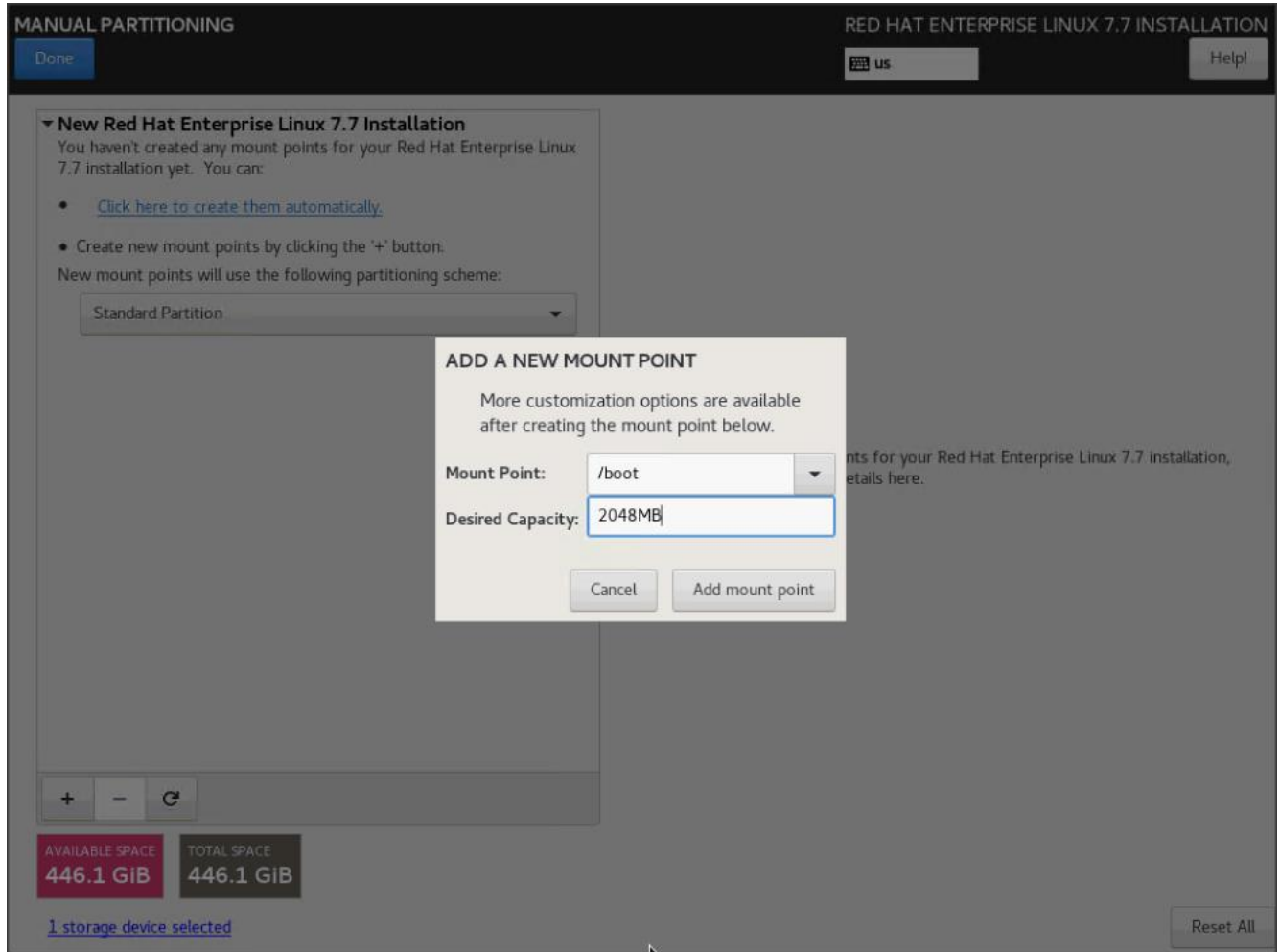


17. This opens a new window with the boot disks. Make the selection and choose “I will configure partitioning”. Click Done. We selected two M.2 SATA SSDs.

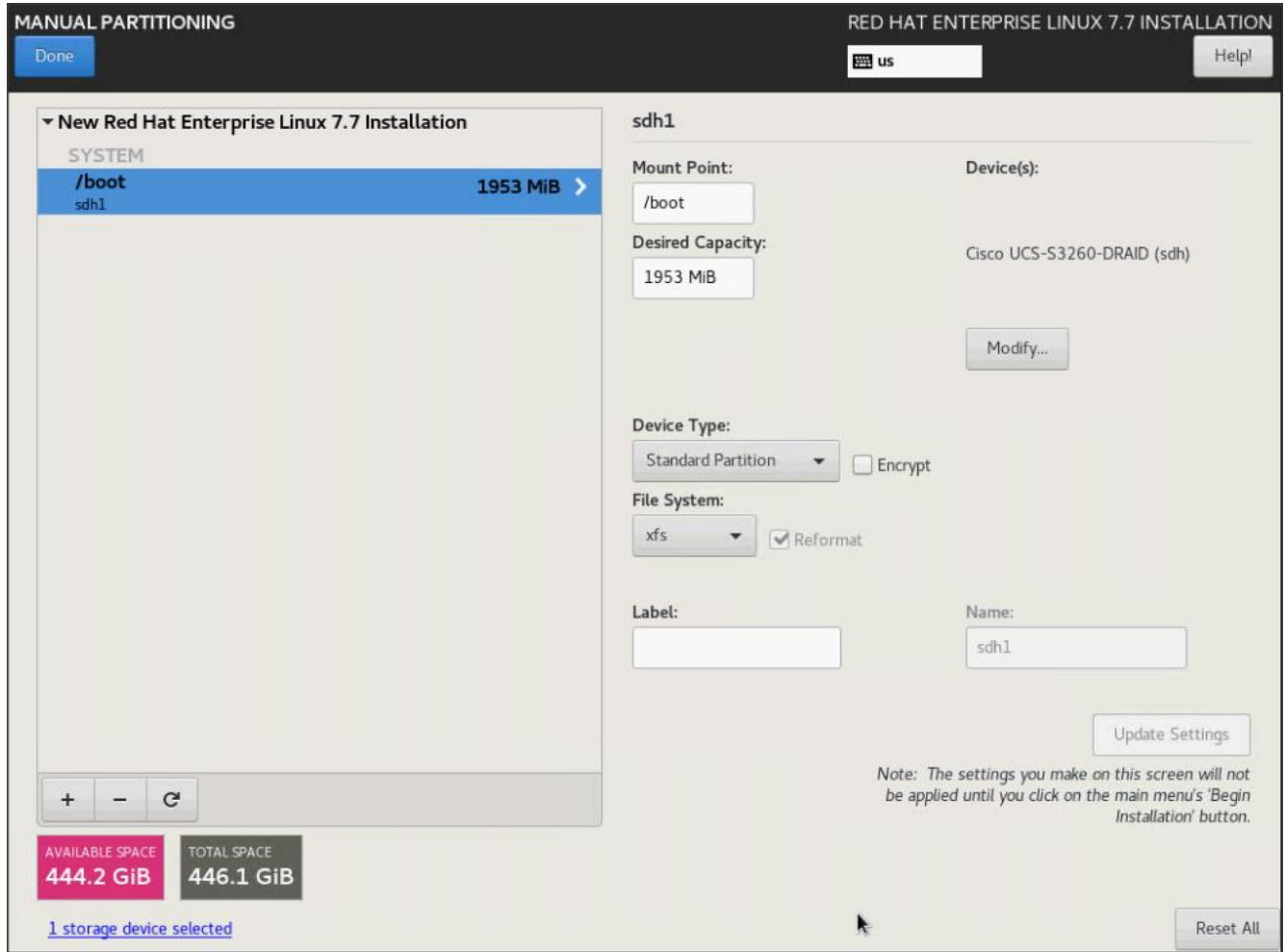


18. This opens a window to create the partitions. Click the + sign to add a new partition as shown below with a boot partition size 2048 MB.

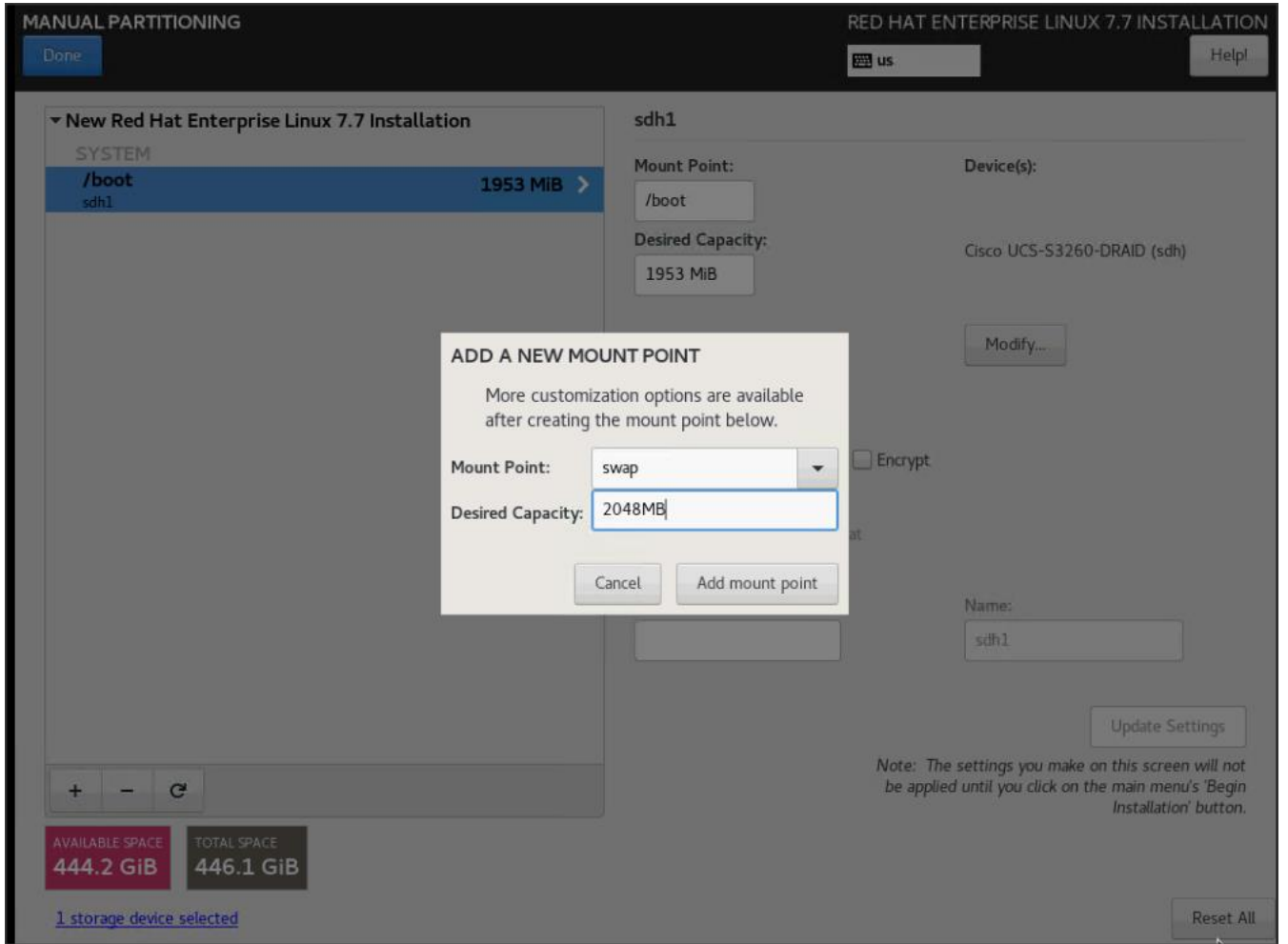
19. Click Add Mount Point to add the partition.



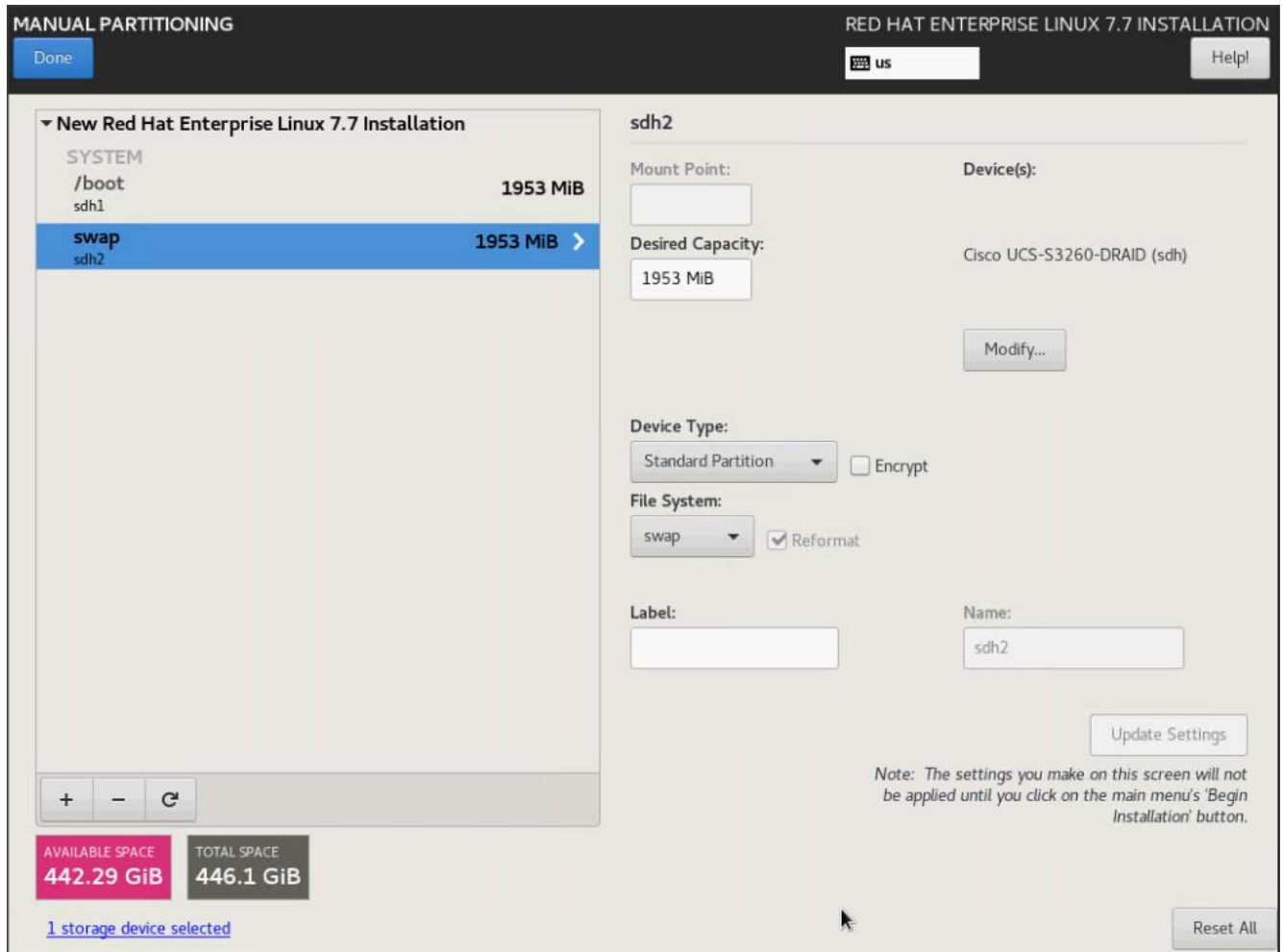
20. Change the device type to RAID and make sure the RAID level is RAID1 (redundancy) and click Update Settings to save the changes.



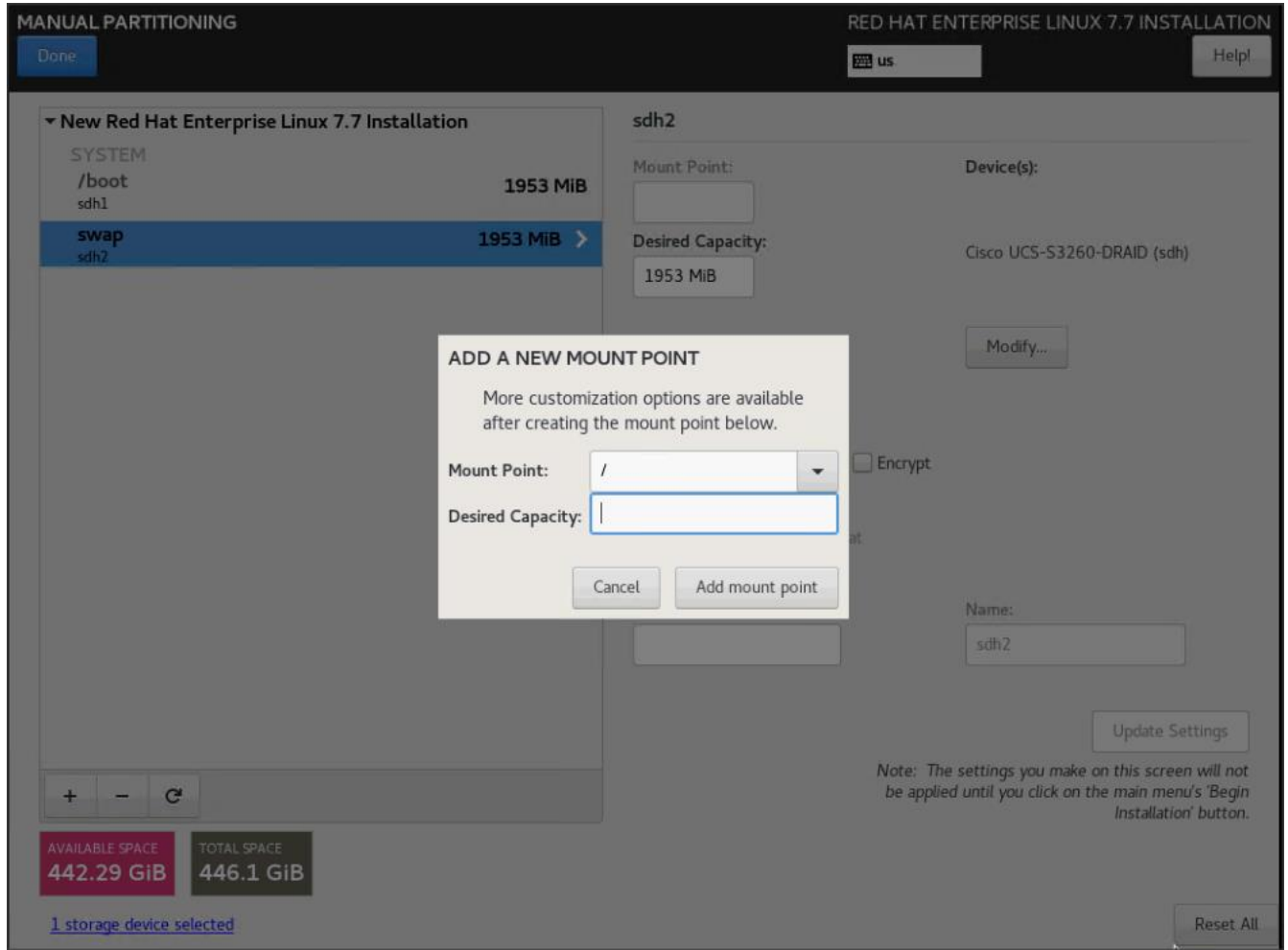
21. Click the + sign to create the swap partition of size 2048 MB. Click Add Mount Point.



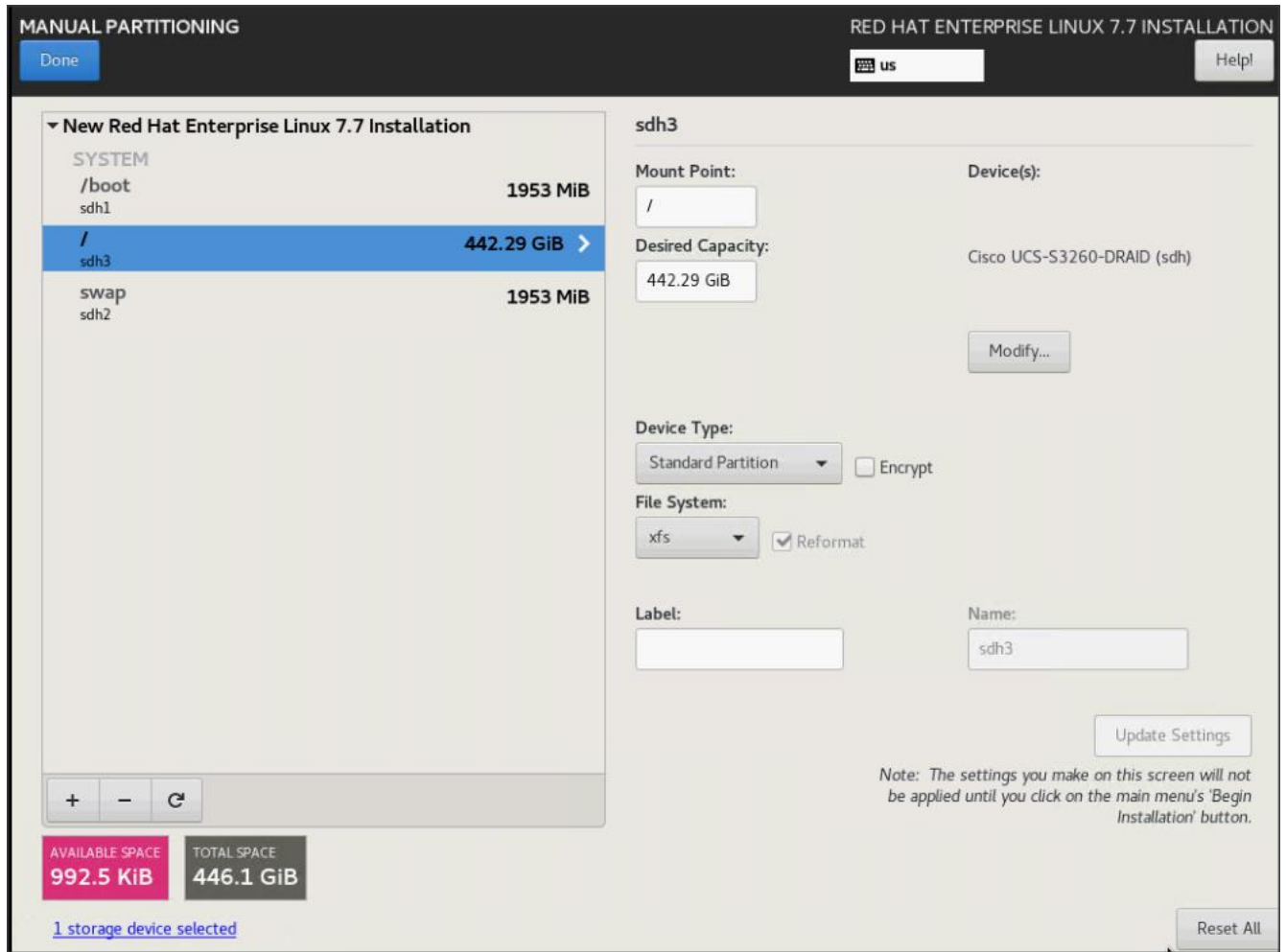
22. Change the Device type to RAID and RAID level to RAID1 (Redundancy) and click Update Settings.



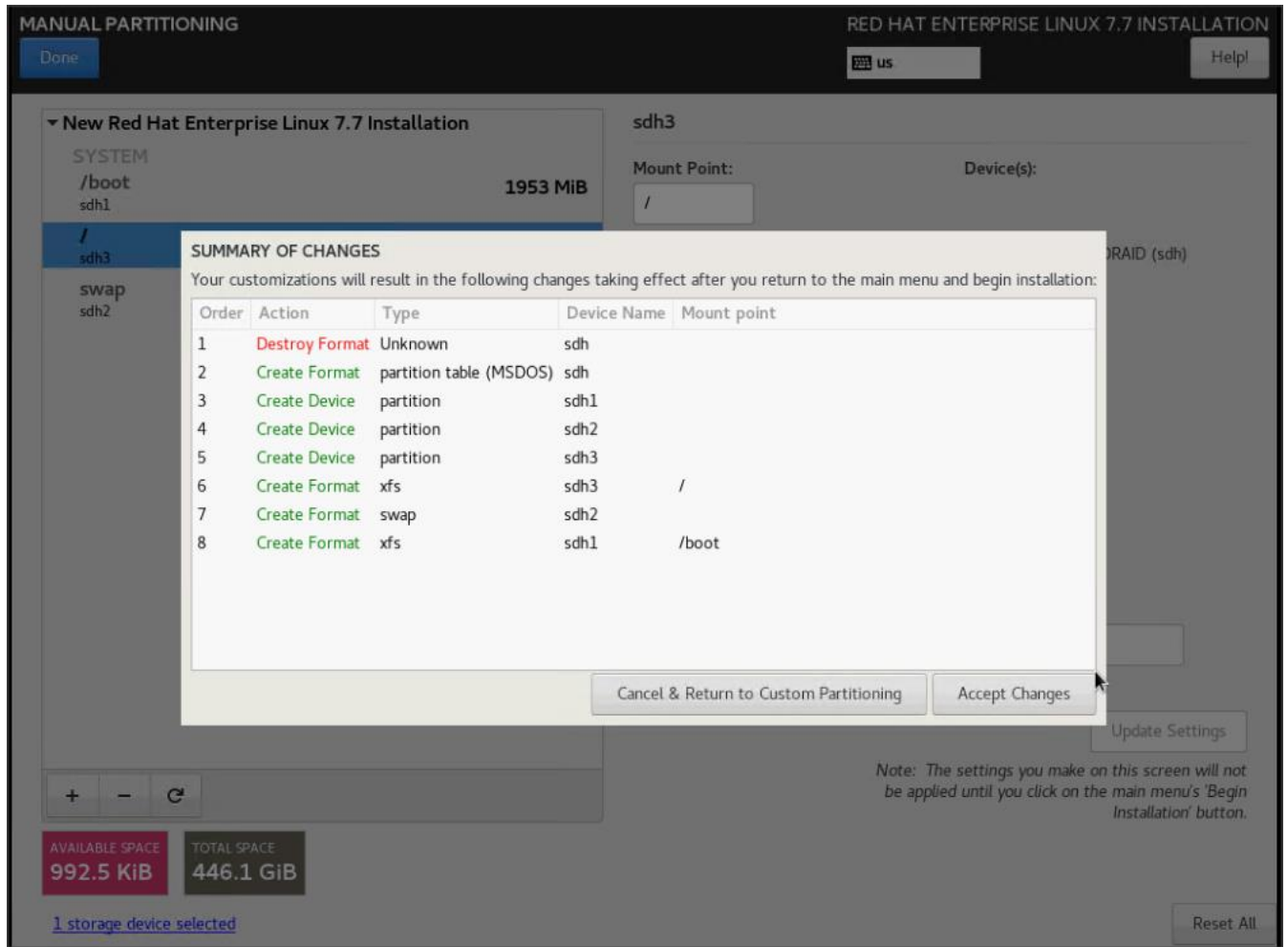
23. Click + to add the / partition. The size can be left empty so it will use the remaining capacity. Click Add Mountpoint.



24. Change the Device type to RAID and RAID level to RAID1 (Redundancy). Click Update Settings.

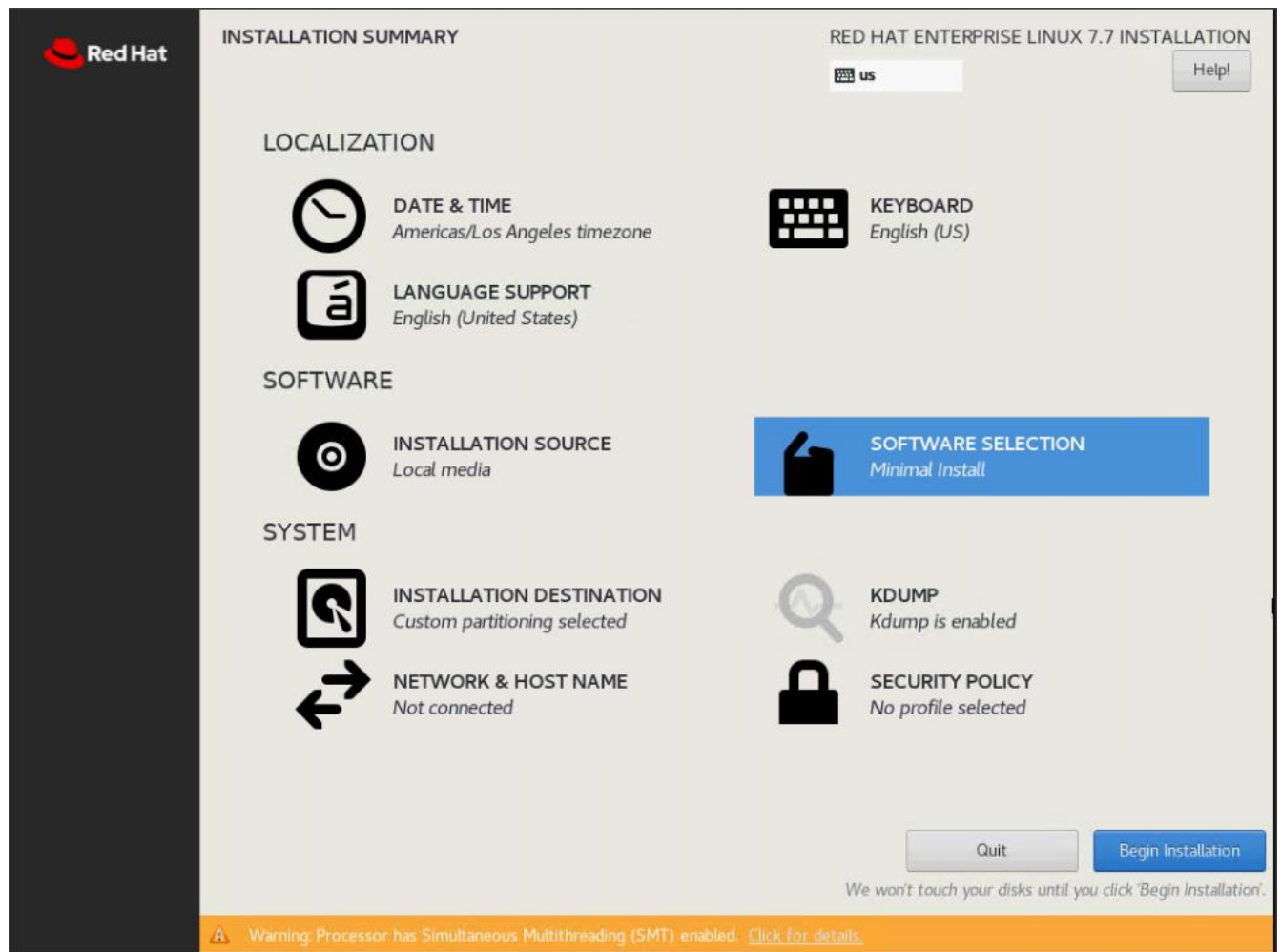


25. Accept Changes.



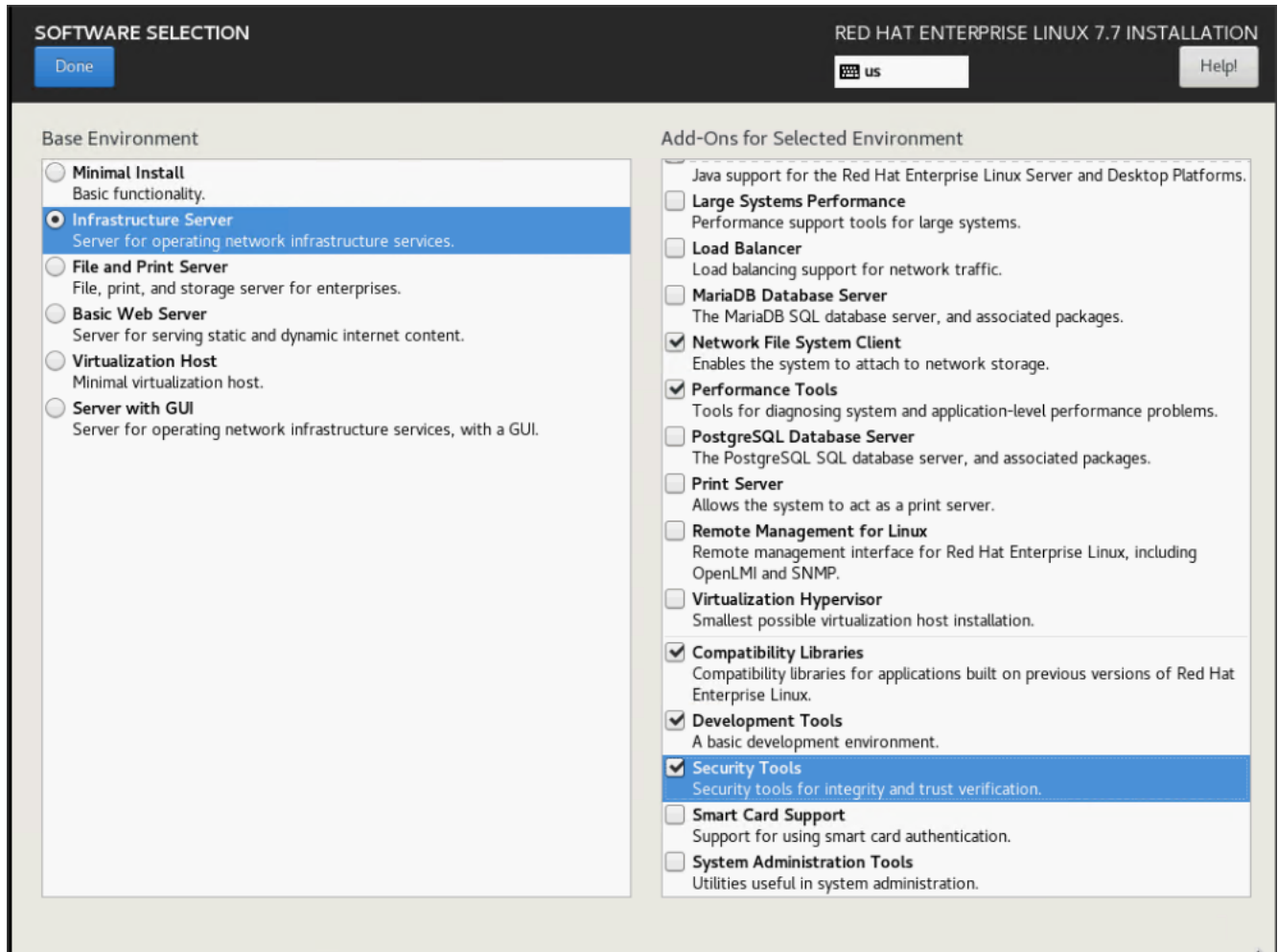
26. Click Done to go back to the main screen and continue the Installation.

27. Click Software Selection.



28. Select Infrastructure Server and select the Add-Ons as noted below, then click Done:

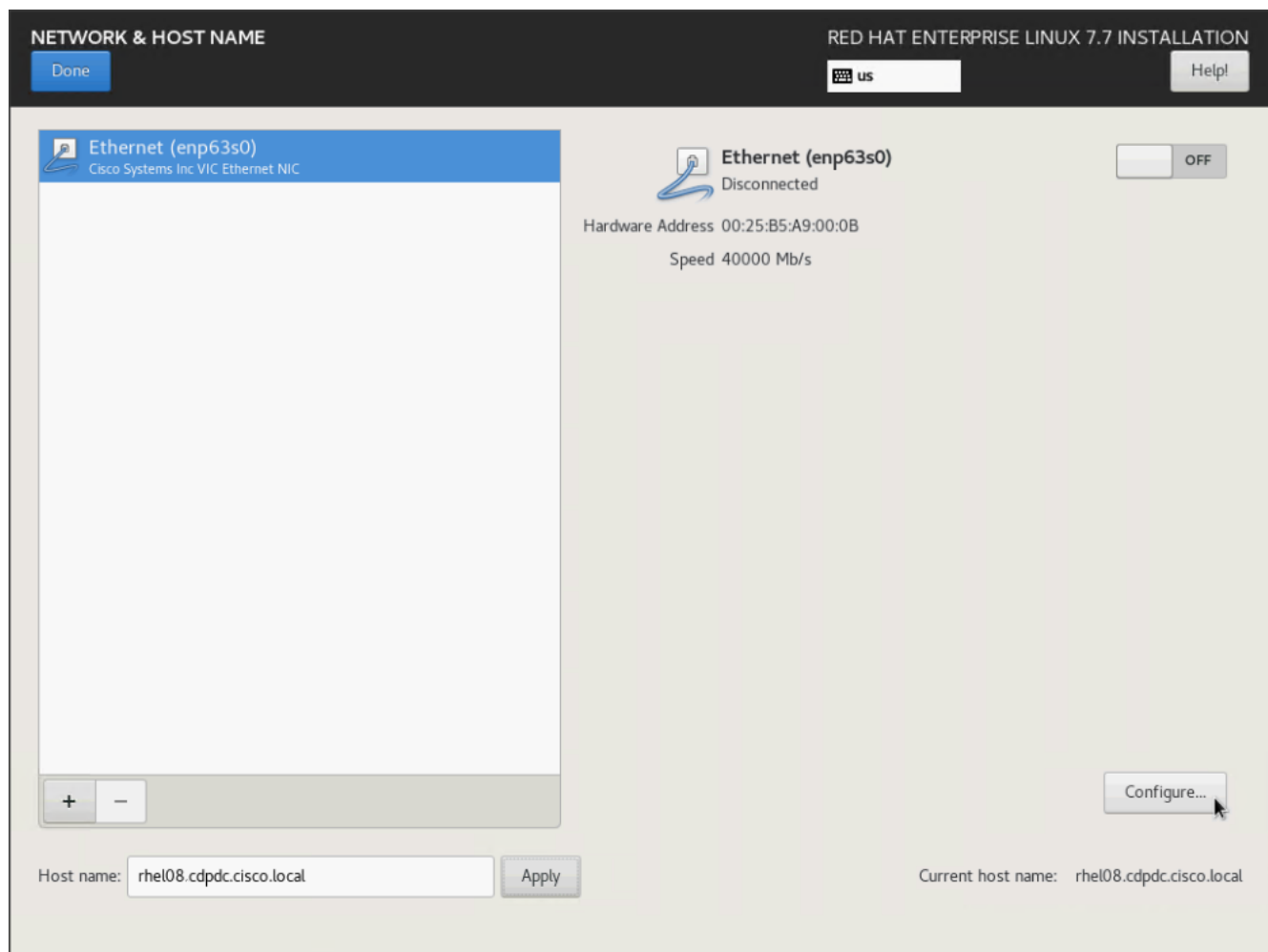
- a. Network File System Client
- b. Performance Tools
- c. Compatibility Libraries
- d. Development Tools
- e. Security Tools



29. Click Network and Hostname and configure Hostname and Networking for the Host.

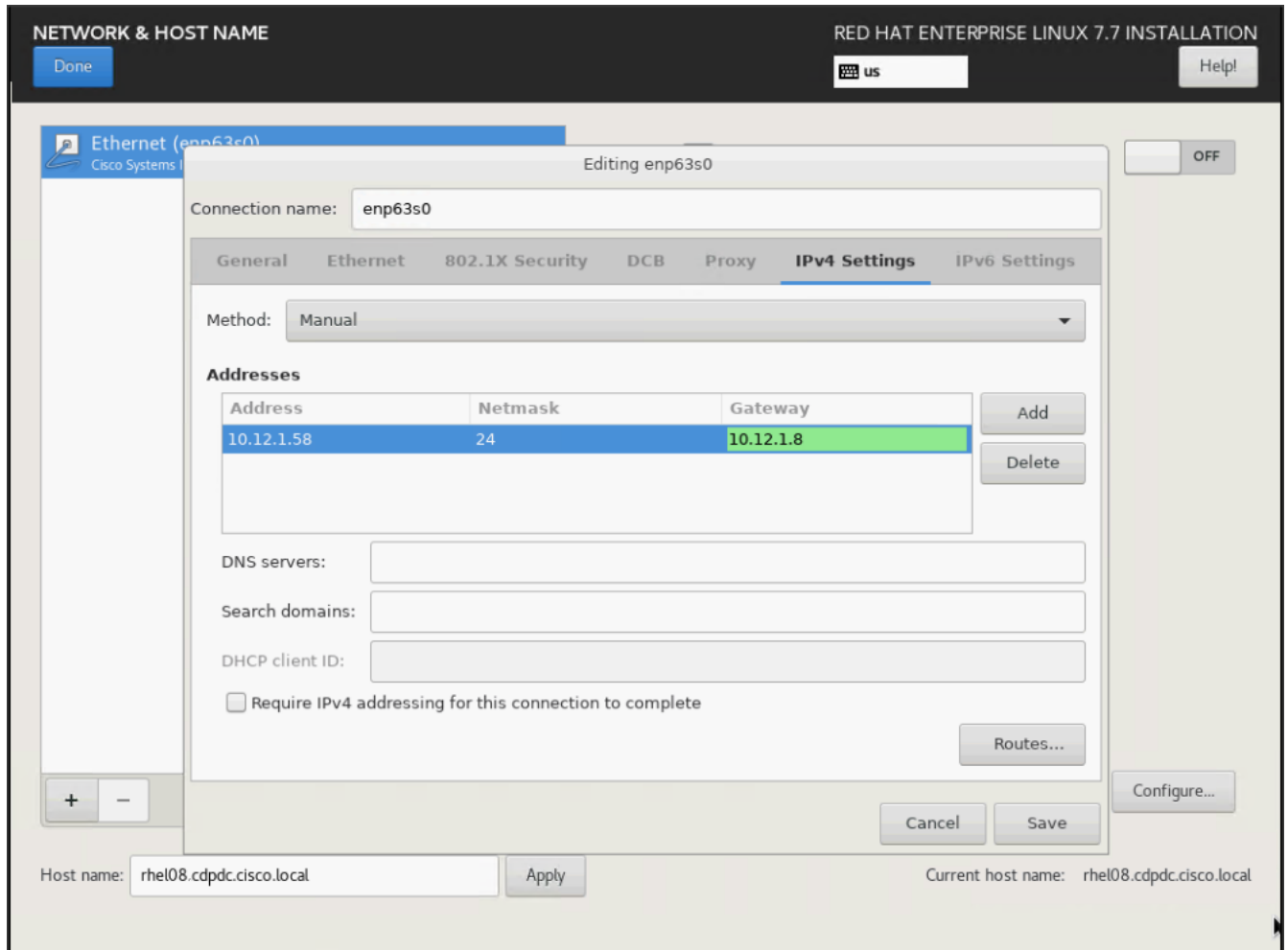


30. Type in the hostname as shown below.



31. Click Configure to open the Network Connectivity window. Click IPv4 Settings.

32. Change the Method to Manual and click Add to enter the IP Address, Netmask and Gateway details.



33. Click Save, update the hostname, and turn Ethernet ON. Click Done to return to the main menu.

34. Click Begin Installation in the main menu.

NETWORK & HOST NAME RED HAT ENTERPRISE LINUX 7.7 INSTALLATION

Ethernet (enp63s0)
Cisco Systems Inc VIC Ethernet NIC

Ethernet (enp63s0)

Connected

Hardware Address 00:25:B5:A9:00:0B

Speed 40000 Mb/s

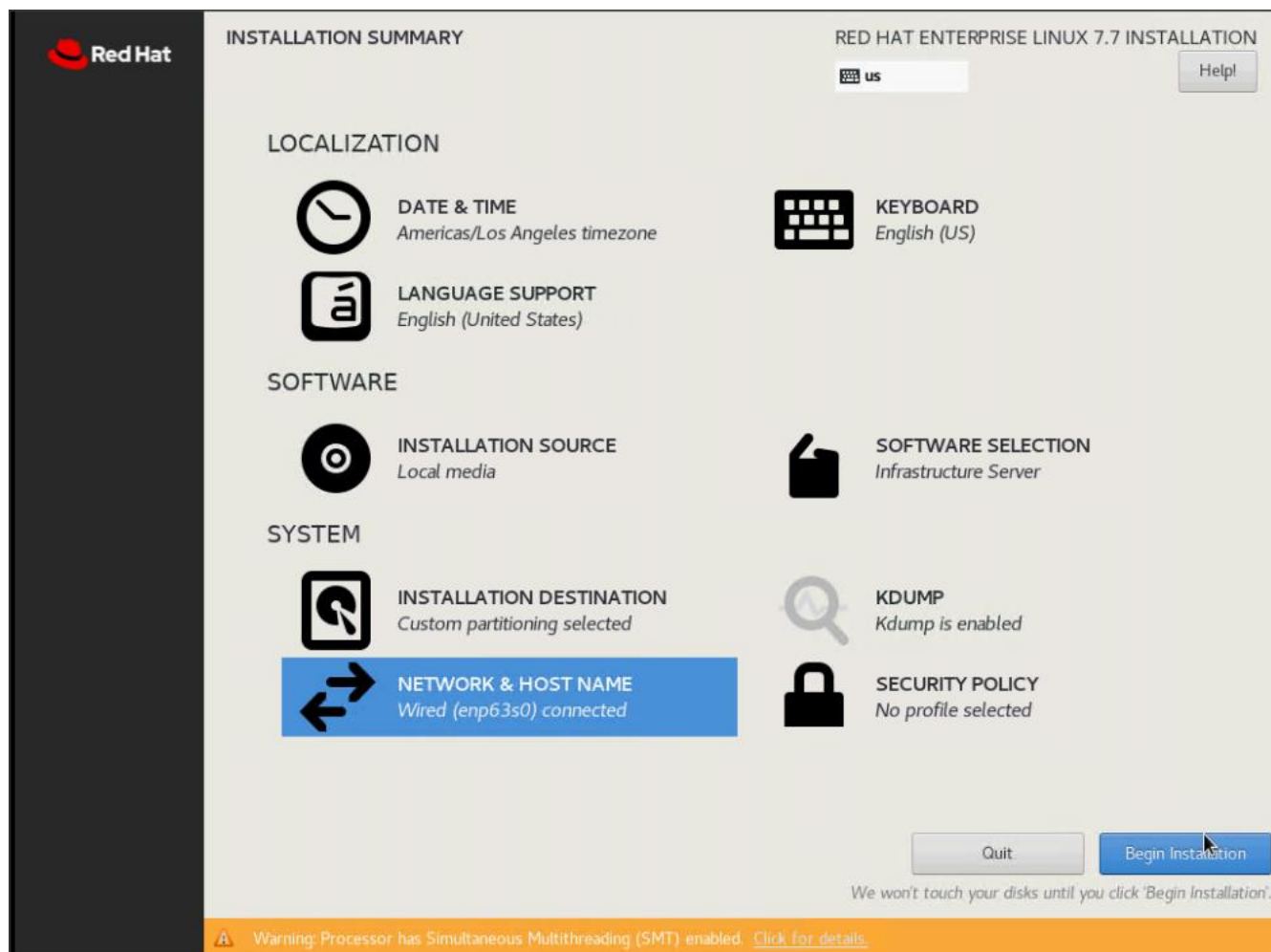
IP Address 10.12.1.58

Subnet Mask 255.255.255.0

Default Route 10.12.1.8

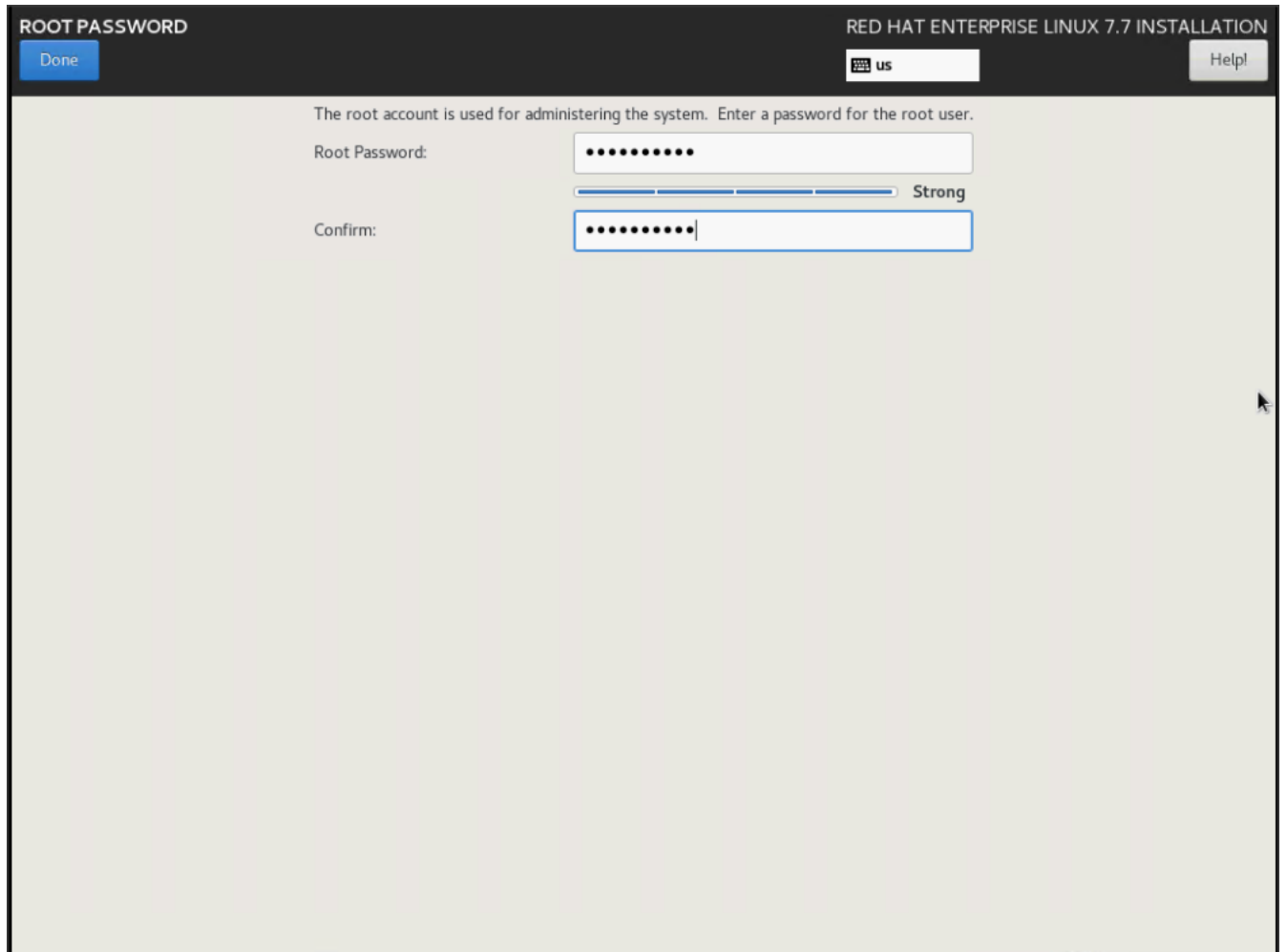
DNS

Host name: Current host name: rhel08.cdpcd.cisco.local

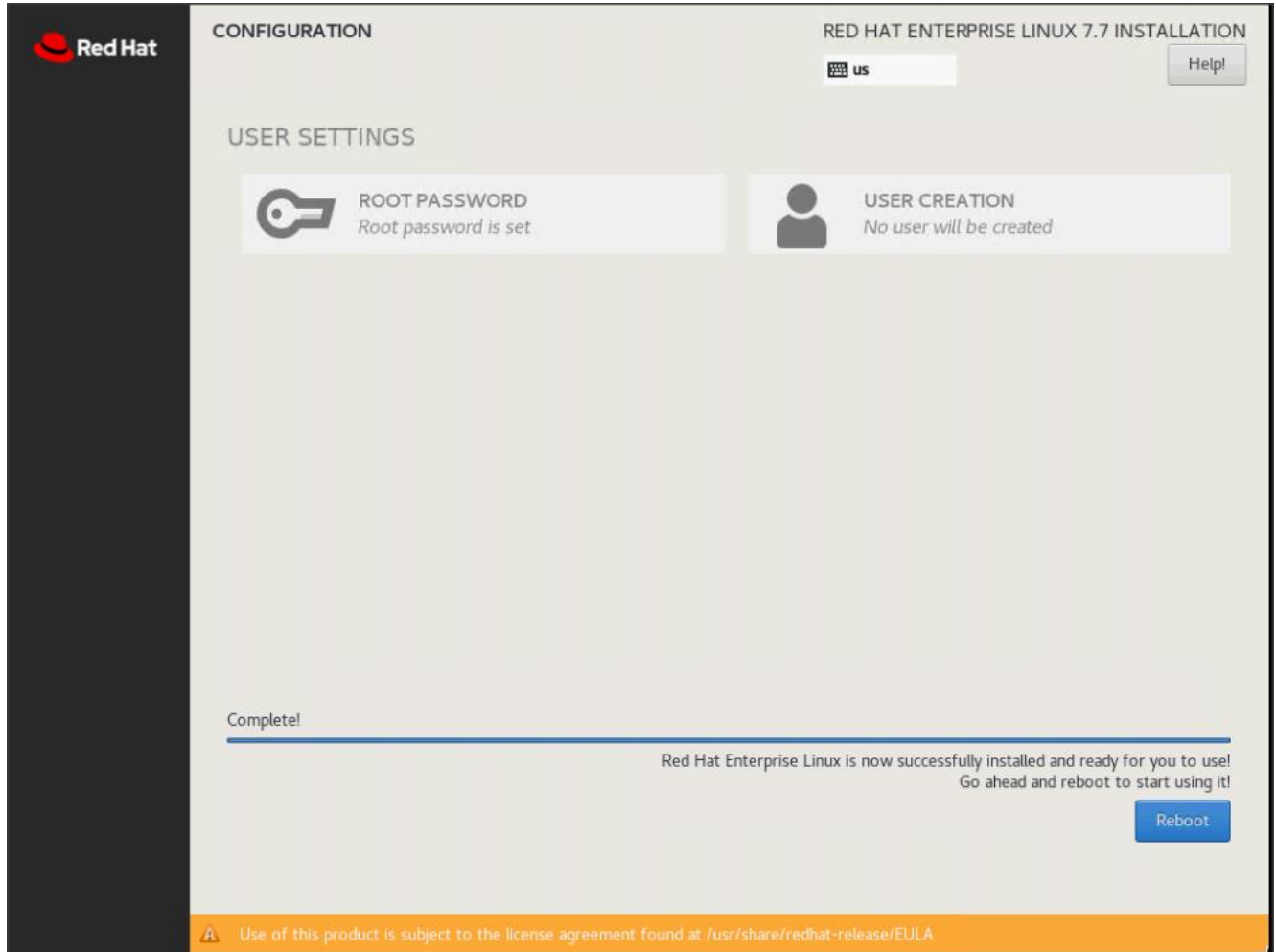


35. Select Root Password in the User Settings.

36. Enter the Root Password and click Done.



37. Once the installation is complete, reboot the system.



38. Repeat steps 1 to 37 to install Red Hat Enterprise Linux 7.7 on Servers 2 through 30.



The OS installation and configuration of the nodes that is mentioned above can be automated through PXE boot or third-party tools.

The hostnames and their corresponding IP addresses are shown in [Table 4](#).

Table 4 Hostname and IP address

Hostname	Eth0
rhel01	10.12.1.51
rhel02	10.12.1.52
rhel03	10.12.1.53
rhel04	10.12.1.54
.....

Hostname	Eth0
rhel15	10.12.1.65
rhel16	10.12.1.66



Multi-homing configuration is not recommended in this design, so please assign only one network interface on each host.



For simplicity, outbound NATing is configured for internet access when desired, such as accessing public repos and/or accessing Red Hat Content Delivery Network. However, configuring outbound NAT is beyond the scope of this document.

Post OS Install Configuration

Choose one of the nodes of the cluster or a separate node as the Admin Node for management, such as CDP DC installation, Ansible, creating a local Red Hat repo, and others. In this document, we used rhelnn01 for this purpose.

Configure /etc/hosts

Setup `/etc/hosts` on the Admin node; this is a pre-configuration to setup DNS as shown in the next section.



For the purpose of simplicity, the `/etc/hosts` file is configured with hostnames in all the nodes. However, in large scale production grade deployment, DNS server setup is highly recommended. Furthermore, `/etc/hosts` file is not copied into containers running on the platform.

Below are the sample A records for DNS configuration within Linux environment:

```
ORIGIN cdpdc.cisco.local
rhel01  A 10.12.1.51
rhel02  A 10.12.1.52
rhel03  A 10.12.1.53
...
...
rhel15  A 10.12.1.65
rhel16  A 10.12.1.66
rhelnn01 A 10.12.1.67
rhelnn02 A 10.12.1.68
rhelnn03 A 10.12.1.69
```

To create the host file on the admin node, follow these steps:

1. Log into the Admin Node (rhelnn01).


```
#ssh 10.12.1.57
```
2. Populate the host file with IP addresses and corresponding hostnames on the Admin node (rhelnn01) and other nodes as follows:
3. On the Admin Node (rhelnn01):

```
[root@rhelnn07 ~]# cat /etc/hosts
127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
10.12.1.51  rhel01.cdpdc.cisco.local rhel01
10.12.1.52  rhel02.cdpdc.cisco.local rhel02
10.12.1.53  rhel03.cdpdc.cisco.local rhel03
10.12.1.54  rhel04.cdpdc.cisco.local rhel04
10.12.1.55  rhel05.cdpdc.cisco.local rhel05
10.12.1.56  rhel06.cdpdc.cisco.local rhel06
10.12.1.57  rhel07.cdpdc.cisco.local rhel07
10.12.1.58  rhel08.cdpdc.cisco.local rhel08
10.12.1.59  rhel09.cdpdc.cisco.local rhel09
10.12.1.60  rhel10.cdpdc.cisco.local rhel10
10.12.1.61  rhel01.cdpdc.cisco.local rhel11
10.12.1.62  rhel02.cdpdc.cisco.local rhel12
10.12.1.63  rhel03.cdpdc.cisco.local rhel13
10.12.1.64  rhel04.cdpdc.cisco.local rhel14
10.12.1.65  rhel05.cdpdc.cisco.local rhel15
10.12.1.66  rhel06.cdpdc.cisco.local rhel16
10.12.1.67  rhel07.cdpdc.cisco.local rhelnn01
10.12.1.68  rhel08.cdpdc.cisco.local rhelnn02
10.12.1.69  rhel09.cdpdc.cisco.local rhelnn03
```

Set Up Passwordless Login

To manage all the nodes in a cluster from the admin node password-less login needs to be setup. It assists in automating common tasks with Ansible, and shell-scripts without having to use passwords.

To enable password-less login across all the nodes when Red Hat Linux is installed across all the nodes in the cluster, follow these steps:

1. Log into the Admin Node (rhelnn01).

```
#ssh 10.12.1.67
```

2. Run the ssh-keygen command to create both public and private keys on the admin node.

```
# ssh-keygen -N '' -f ~/.ssh/id_rsa
```

Figure 51 Ssh-keygen

```
[root@rhel07 /]#
[root@rhel07 /]# ssh-keygen -N '' -f ~/.ssh/id_rsa
Generating public/private rsa key pair.
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:kUuqzZ1qmaEhj0Kma/aeLnDogkcYjWt21uPT0G/nqZQ root@rhel07.cdpdc.cisco.local
The key's randomart image is:
+----[RSA 2048]-----+
|
| o      +
| o .    o o
| B . o S
|Bo=o.B.o ..
|Ooo.oo*++E
|oB. oo+o.o ..
|Oo== .o ..+o
+----[SHA256]-----+
[root@rhel07 /]#
[root@rhel07 /]# █
```

3. Run the following command from the admin node to copy the public key `id_rsa.pub` to all the nodes of the cluster. `ssh-copy-id` appends the keys to the remote-hosts `.ssh/authorized_keys`.

```
# for i in {01..16}; do echo "copying rhel$i.cdpdc.cisco.local"; ssh-copy-id -i
~/.ssh/id_rsa.pub root@rhel$i.cdpdc.cisco.local; done;
# for i in {01..03}; do echo "copying rhelnn$i.cdpdc.cisco.local"; ssh-copy-id -i
~/.ssh/id_rsa.pub root@rhelnn$i.cdpdc.cisco.local; done;
```

4. Enter yes for Are you sure you want to continue connecting (yes/no)?
5. Enter the password of the remote host.

Create a Red Hat Enterprise Linux (RHEL) 7.7 Local Repository

To create a repository using RHEL DVD or ISO on the admin node (in this deployment `rhelnn01` is used for this purpose), create a directory with all the required RPMs, run the `createrepo` command and then publish the resulting repository.

To create a RHEL 7.7 local repository, follow these steps:

1. Log into `rhelnn01`. Create a directory that would contain the repository.

```
# mkdir -p /var/www/html/rhelrepo
```

2. Copy the contents of the Red Hat DVD to `/var/www/html/rhelrepo`
3. Alternatively, if you have access to a Red Hat ISO Image, Copy the ISO file to `rhelnn01`.
4. Log back into `rhelnn01` and create the mount directory.

```
# scp rhel-server-7.7-x86_64-dvd.iso rhelnn01:/root/
# mkdir -p /mnt/rheliso
# mount -t iso9660 -o loop /root/rhel-server-7.7-x86_64-dvd.iso /mnt/rheliso/
```

5. Copy the contents of the ISO to the `/var/www/html/rhelrepo` directory.

```
# cp -r /mnt/rheliso/* /var/www/html/rhelrepo
```

6. On `rhelnn01` create a `.repo` file to enable the use of the `yum` command.

```
# vi /var/www/html/rhelrepo/rheliso.repo
[rhel7.7]
name=Red Hat Enterprise Linux 7.7
baseurl=http://10.12.1.67/rhelrepo
gpgcheck=0
enabled=1
```

7. Copy `rheliso.repo` file from `/var/www/html/rhelrepo` to `/etc/yum.repos.d` on `rhelnn01`.

```
# cp /var/www/html/rhelrepo/rheliso.repo /etc/yum.repos.d/rheliso.repo
```



Based on this repository file, yum requires httpd to be running on rhelnn01 for other nodes to access the repository.

8. To make use of repository files on `rhelnn01` without `httpd`, edit the `baseurl` of `repo` file `/etc/yum.repos.d/rheliso.repo` to point repository location in the file system.



This step is needed to install software on Admin Node (rhelnn01) using the repo (such as `httpd`, `create-repo`, and so on.)

```
# vi /etc/yum.repos.d/rheliso.repo
[rhel7.7]
name=Red Hat Enterprise Linux 7.7
baseurl=file:///var/www/html/rhelrepo
gpgcheck=0
enabled=1
```

Create the Red Hat Repository Database

To create the Red Hat repository database, follow these steps:

1. Install the “`createrepo`” package on admin node (`rhelnn01`). Use it to regenerate the repository database(s) for the local copy of the RHEL DVD contents:

```
# yum -y install createrepo
```

2. Run “`createrepo`” on the RHEL repository to create the repo database on admin node:

```
# cd /var/www/html/rhelrepo
# createrepo .
```

Figure 52 createrepo

```

[root@rhel07 rhelrepo]#
[root@rhel07 rhelrepo]# cd /var/www/html/rhelrepo
[root@rhel07 rhelrepo]#
[root@rhel07 rhelrepo]#
[root@rhel07 rhelrepo]# createrepo .
Spawning worker 0 with 167 pkgs
Spawning worker 1 with 167 pkgs
Spawning worker 2 with 167 pkgs
Spawning worker 3 with 167 pkgs
Spawning worker 4 with 167 pkgs
Spawning worker 5 with 167 pkgs
Spawning worker 23 with 167 pkgs
Spawning worker 24 with 167 pkgs
Spawning worker 25 with 167 pkgs
Spawning worker 26 with 166 pkgs
Spawning worker 27 with 166 pkgs
Spawning worker 28 with 166 pkgs
Spawning worker 29 with 166 pkgs
Spawning worker 30 with 166 pkgs
Spawning worker 31 with 166 pkgs
Workers Finished
Saving Primary metadata
Saving file lists metadata
Saving other metadata
Generating sqlite DBs
Sqlite DBs complete
[root@rhel07 rhelrepo]# █

```

Set Up Ansible

To set up Ansible, follow these steps:

1. Download Ansible rpm from the following link: https://releases.ansible.com/ansible/rpm/release/epel-7-x86_64/ansible-2.9.6-1.el7.ans.noarch.rpm

```
# wget https://releases.ansible.com/ansible/rpm/release/epel-7-x86_64/ansible-2.9.6-1.el7.ans.noarch.rpm
```

2. Run the following command to install ansible:

```
# yum localinstall -y ansible-2.9.6-1.el7.ans.noarch.rpm
```

3. Verify Ansible installation by running the following commands:

```
# ansible --version
ansible 2.9.6
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules',
u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
```

```
python version = 2.7.5 (default, Jun 11 2019, 14:33:56) [GCC 4.8.5 20150623 (Red
Hat 4.8.5-39)]

# ansible localhost -m ping
localhost | SUCCESS => {
  "changed": false,
  "ping": "pong"
}
```

4. Prepare the host inventory file for Ansible as shown below. Various host groups have been created based on any specific installation requirements of certain hosts.

```
[root@rhelnn01 ~]# cat /etc/ansible/hosts
[admin]
rhelnn01.cdpdc.cisco.local

[namenodes]
rhelnn01.cdpdc.cisco.local
rhelnn02.cdpdc.cisco.local
rhelnn03.cdpdc.cisco.local

[datanodes]
rhel01.cdpdc.cisco.local
rhel02.cdpdc.cisco.local
rhel03.cdpdc.cisco.local
rhel04.cdpdc.cisco.local
rhel05.cdpdc.cisco.local
rhel06.cdpdc.cisco.local
rhel07.cdpdc.cisco.local
rhel08.cdpdc.cisco.local
rhel09.cdpdc.cisco.local
rhel10.cdpdc.cisco.local
rhel11.cdpdc.cisco.local
rhel12.cdpdc.cisco.local
rhel13.cdpdc.cisco.local
rhel14.cdpdc.cisco.local
rhel15.cdpdc.cisco.local
rhel16.cdpdc.cisco.local

[nodes]
rhelnn01.cdpdc.cisco.local
rhelnn02.cdpdc.cisco.local
rhelnn03.cdpdc.cisco.local
rhel01.cdpdc.cisco.local
rhel02.cdpdc.cisco.local
rhel03.cdpdc.cisco.local
rhel04.cdpdc.cisco.local
rhel05.cdpdc.cisco.local
rhel06.cdpdc.cisco.local
rhel07.cdpdc.cisco.local
rhel08.cdpdc.cisco.local
rhel09.cdpdc.cisco.local
rhel10.cdpdc.cisco.local
rhel11.cdpdc.cisco.local
rhel12.cdpdc.cisco.local
```



```
rhel13.cdpdc.cisco.local
rhel14.cdpdc.cisco.local
rhel15.cdpdc.cisco.local
rhel16.cdpdc.cisco.local
```

5. Verify host group by running the following commands. Error! Reference source not found. shows the outcome of the ping command.

```
# ansible datanodes -m ping
```

Install httpd

Setting up the RHEL repository on the admin node requires httpd. To set up RHEL repository on the admin node, follow these steps:

1. Install httpd on the admin node to host repositories:



The Red Hat repository is hosted using HTTP on the admin node; this machine is accessible by all the hosts in the cluster.

```
# yum -y install httpd
```

2. Add ServerName and make the necessary changes to the server configuration file:

```
# vi /etc/httpd/conf/httpd.conf
ServerName 10.12.1.67:80
```

3. Start httpd:

```
# service httpd start
# chkconfig httpd on
```

Disable the Linux Firewall



The default Linux firewall settings are too restrictive for any Hadoop deployment. Since the Cisco UCS Big Data deployment will be in its own isolated network there is no need for that additional firewall.

```
# ansible all -m command -a "firewall-cmd --zone=public --add-port=80/tcp --
permanent"
# ansible all -m command -a "firewall-cmd --reload"
# ansible all -m command -a "systemctl disable firewalld"
```

Set Up All Nodes to use the RHEL Repository

To set up all nodes to use the RHEL repository, follow these steps:



Based on this repository file, yum requires httpd to be running on rhelnn01 for other nodes to access the repository.

1. Copy the rheliso.repo to all the nodes of the cluster:

```
# ansible nodes -m copy -a "src=/var/www/html/rhelrepo/rheliso.repo
dest=/etc/yum.repos.d/."
```

2. Copy the /etc/hosts file to all nodes:

```
# ansible nodes -m copy -a "src=/etc/hosts dest=/etc/hosts"
```

3. Purge the yum caches:

```
# ansible nodes -a "yum clean all"
# ansible nodes -a "yum repolist"
```



While the suggested configuration is to disable SELinux as shown below, if for any reason SELinux needs to be enabled on the cluster, run the following command to make sure that the httpd can read the Yum repofiles.

```
#chcon -R -t httpd_sys_content_t /var/www/html/
```

Disable SELinux



SELinux must be disabled during the install procedure and cluster setup. SELinux can be enabled after installation and while the cluster is running.

SELinux can be disabled by editing `/etc/selinux/config` and changing the `SELINUX` line to `SELINUX=disabled`.

To disable SELinux, follow these steps:

1. The following command will disable `SELINUX` on all nodes:

```
# ansible nodes -m shell -a "sed -i 's/SELINUX=enforcing/SELINUX=disabled/g'
/etc/selinux/config"
# ansible nodes -m shell -a "setenforce 0"
```



The above command may fail if SELinux is already disabled. This requires reboot to take effect.

2. Reboot the machine, if needed for SELinux to be disabled in case it does not take effect. It can be checked using the following command:

```
# ansible namenodes -a "sestatus"

rhel07.cdpdc.cisco.local | CHANGED | rc=0 >>
SELinux status:          disabled

rhel09.cdpdc.cisco.local | CHANGED | rc=0 >>
SELinux status:          disabled

rhel11.cdpdc.cisco.local | CHANGED | rc=0 >>
```

```
SELinux status:                disabled
```

Upgrade the Cisco Network Driver for VIC1387

The latest Cisco Network driver is required for performance and updates. The latest drivers can be downloaded from the link below:

[https://software.cisco.com/download/home/283862063/type/283853158/release/4.1\(1a\)](https://software.cisco.com/download/home/283862063/type/283853158/release/4.1(1a))

In the ISO image, the required driver `kmod-enic-4.0.0.8-802.24.rhel7u7.x86_64.rpm` can be located at `\Network\Cisco\VIC\RHEL\RHEL7.7\`.
To upgrade the Cisco Network Driver for VIC1387, follow these steps:
From a node connected to the Internet, download, extract and transfer `kmod-enic-.rpm` to `rhelnn01` (admin node).

1. Copy the rpm on all nodes of the cluster using the following Ansible commands. For this example, the rpm is assumed to be in present working directory of `rhelnn01`:

```
[root@rhelnn01 ~]# ansible all -m copy -a "src=/root/kmod-enic-4.0.0.8-802.24.rhel7u7.x86_64.rpm dest=/root/."
```

2. Use the yum module to install the enic driver rpm file on all the nodes through Ansible:

```
[root@rhelnn01 ~]# ansible all -m yum -a "name=/root/kmod-enic-4.0.0.8-802.24.rhel7u7.x86_64.rpm state=present"
Make sure that the above installed version of kmod-enic driver is being used on all nodes by running the command "modinfo enic" on all nodes:
[root@rhel102 ~]# ansible all -m shell -a "modinfo enic | head -5"
```

3. It is recommended to download the `kmod-megaraid` driver for higher performance on Name nodes. The RPM can be found in the same package at:
`\Storage\LSI\Cisco_Storage_12G_SAS_RAID_controller\RHEL\RHEL7.7\kmod-megaraid_sas-07.710.06.00_e17.7-1.x86_64.rpm`:
4. Copy the rpm on all nodes of the cluster using the following Ansible commands. For this example, the rpm is assumed to be in present working directory of `rhelnn01`:

```
[root@rhelnn01 ~]# ansible namenodes -m copy -a "src=/root/kmod-megaraid_sas-07.710.06.00_e17.7-1.x86_64.rpm dest=/root/."
```

5. Use the yum module to install the enic driver rpm file on all the nodes through Ansible:

```
[root@rhelnn01 ~]# ansible namenodes -m yum -a "name=/root/kmod-megaraid_sas-07.710.06.00_e17.7-1.x86_64.rpm state=present"
Make sure that the above installed version of kmod-megaraid driver is being used on all nodes by running the command "modinfo megaraid_sas" on all nodes:
[root@rhelnn01 ~]# ansible all -m shell -a "modinfo megaraid_sas | head -5"
```

Set Up JAVA

To setup JAVA, follow these steps:



CDP DC 7 requires JAVA 8.

1. Download `jdk-8u241-linux-x64.rpm` and src the rpm to admin node (rhelnn01) from the link: <https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

2. Copy JDK rpm to all nodes:

```
# ansible nodes -m copy -a "src=/root/jdk-8u241-linux-x64.rpm dest=/root/."
```

3. Extract and Install JDK all nodes:

```
# ansible all -m command -a "rpm -ivh jdk-8u241-linux-x64.rpm "
```

4. Create the following files `java-set-alternatives.sh` and `java-home.sh` on admin node (rhelnn01):

```
# vi java-set-alternatives.sh
#!/bin/bash
for item in java javac javaws jar jps javah javap jcontrol jconsole jdb; do
  rm -f /var/lib/alternatives/$item
  alternatives --install /usr/bin/$item $item /usr/java/jdk1.8.0_241-amd64/bin/$item
done

# vi java-home.sh
export JAVA_HOME=/usr/java/jdk1.8.0_241-amd64
```

5. Make the two java scripts created above executable:

```
chmod 755 ./java-set-alternatives.sh ./java-home.sh
```

6. Copying `java-set-alternatives.sh` to all nodes.

```
ansible nodes -m copy -a "src=/root/java-set-alternatives.sh dest=/root/."
ansible nodes -m file -a "dest=/root/java-set-alternatives.sh mode=755"
ansible nodes -m copy -a "src=/root/java-home.sh dest=/root/."
ansible nodes -m file -a "dest=/root/java-home.sh mode=755"
```

7. Setup Java Alternatives

```
[root@rhelnn01 ~]# ansible all -m shell -a "/root/java-set-alternatives.sh"
```

8. Make sure correct java is setup on all nodes (should point to newly installed java path).

```
# ansible all -m shell -a "alternatives --display java | head -2"
rhel10.cdpdc.cisco.local | CHANGED | rc=0 >>
java - status is manual.
  link currently points to /usr/java/jdk1.8.0_241-amd64/bin/java
rhel08.cdpdc.cisco.local | CHANGED | rc=0 >>
java - status is manual.
  link currently points to /usr/java/jdk1.8.0_241-amd64/bin/java
rhel09.cdpdc.cisco.local | CHANGED | rc=0 >>
java - status is manual.
```

```
link currently points to /usr/java/jdk1.8.0_241-amd64/bin/java
rhel11.cdpdc.cisco.local | CHANGED | rc=0 >>
java - status is manual.
link currently points to /usr/java/jdk1.8.0_241-amd64/bin/java
```

9. Setup JAVA_HOME on all nodes.

```
# ansible all -m copy -a "src=/root/java-home.sh dest=/etc/profile.d"
```

10. Display JAVA_HOME on all nodes.

```
# ansible all -m command -a "echo $JAVA_HOME"
rhel08.cdpdc.cisco.local | CHANGED | rc=0 >>
/usr/java/jdk1.8.0_241-amd64
rhel10.cdpdc.cisco.local | CHANGED | rc=0 >>
/usr/java/jdk1.8.0_241-amd64
rhel11.cdpdc.cisco.local | CHANGED | rc=0 >>
/usr/java/jdk1.8.0_241-amd64
rhel09.cdpdc.cisco.local | CHANGED | rc=0 >>
/usr/java/jdk1.8.0_241-amd64
```

11. Display current java -version.

```
# ansible all -m command -a "java -version"
rhel10.cdpdc.cisco.local | CHANGED | rc=0 >>
java version "1.8.0_241"
Java(TM) SE Runtime Environment (build 1.8.0_241-b07)
Java HotSpot(TM) 64-Bit Server VM (build 25.241-b07, mixed mode)
rhel08.cdpdc.cisco.local | CHANGED | rc=0 >>
java version "1.8.0_241"
Java(TM) SE Runtime Environment (build 1.8.0_241-b07)
Java HotSpot(TM) 64-Bit Server VM (build 25.241-b07, mixed mode)
rhel11.cdpdc.cisco.local | CHANGED | rc=0 >>
java version "1.8.0_241"
Java(TM) SE Runtime Environment (build 1.8.0_241-b07)
Java HotSpot(TM) 64-Bit Server VM (build 25.241-b07, mixed mode)
```

Enable Syslog

Syslog must be enabled on each node to preserve logs regarding killed processes or failed jobs. Modern versions such as syslog-ng and rsyslog are possible, making it more difficult to be sure that a syslog daemon is present.

Use one of the following commands to confirm that the service is properly configured:

```
# ansible all -m command -a "rsyslogd -v"
# ansible all -m command -a "service rsyslog status"
```

Set the ulimit

On each node, `ulimit -n` specifies the number of inodes that can be opened simultaneously. With the default value of 1024, the system appears to be out of disk space and shows no inodes available. This value should be set to 64000 on every node.

Higher values are unlikely to result in an appreciable performance gain.

To set ulimit, follow these steps:

1. For setting the ulimit on Redhat, edit “/etc/security/limits.conf” on admin node rhelnn01 and add the following lines:

```
# vi /etc/security/limits.conf
* soft nofile 1048576
* hard nofile 1048576
```

2. Copy the /etc/security/limits.conf file from admin node (rhelnn01) to all the nodes using the following command:

```
# ansible nodes -m copy -a "src=/etc/security/limits.conf
dest=/etc/security/limits.conf"
```

3. Make sure that the /etc/pam.d/su file contains the following settings:

```
# cat /etc/pam.d/su
#%PAM-1.0
auth            sufficient      pam_rootok.so
# Uncomment the following line to implicitly trust users in the "wheel" group.
#auth           sufficient      pam_wheel.so trust use_uid
# Uncomment the following line to require a user to be in the "wheel" group.
#auth           required        pam_wheel.so use_uid
auth            include          system-auth
auth            include          postlogin
account         sufficient      pam_succeed_if.so uid = 0 use_uid quiet
account         include          system-auth
password        include          system-auth
session         include          system-auth
session         include          postlogin
session         optional        pam_xauth.so
```



The ulimit values are applied on a new shell, running the command on a node on an earlier instance of a shell will show old values.

Set TCP Retries

Adjusting the tcp_retries parameter for the system network enables faster detection of failed nodes. Given the advanced networking features of Cisco UCS, this is a safe and recommended change (failures observed at the operating system layer are most likely serious rather than transitory).

To set TCP retries, follow these steps:



On each node, set the number of TCP retries to 5 can help detect unreachable nodes with less latency.

1. Edit the file /etc/sysctl.conf and on admin node rhelnn01 and add the following lines:

```
net.ipv4.tcp_retries2=5
Copy the /etc/sysctl.conf file from admin node (rhelnn01) to all the nodes using the
following command:
# ansible nodes -m copy -a "src=/etc/sysctl.conf dest=/etc/sysctl.conf"
```

2. Load the settings from default sysctl file `/etc/sysctl.conf` by running the following command:

```
# ansible nodes -m command -a "sysctl -p"
```

Disable IPv6 Defaults

To disable IPv6 defaults, follow these steps:

1. Run the following command:

```
# ansible all -m shell -a "echo 'net.ipv6.conf.all.disable_ipv6 = 1' >> /etc/sysctl.conf"
# ansible all -m shell -a "echo 'net.ipv6.conf.default.disable_ipv6 = 1' >> /etc/sysctl.conf"
# ansible all -m shell -a "echo 'net.ipv6.conf.lo.disable_ipv6 = 1' >> /etc/sysctl.conf"
```

2. Load the settings from default sysctl file `/etc/sysctl.conf`:

```
# ansible all -m shell -a "sysctl -p"
```

Disable Swapping

To disable swapping, follow these steps:

1. Run the following on all nodes. Variable `vm.swappiness` defines how often swap should be used, 60 is default:

```
# ansible all -m shell -a "echo 'vm.swappiness=0' >> /etc/sysctl.conf"
```

2. Load the settings from default sysctl file `/etc/sysctl.conf` and verify the content of `sysctl.conf`:

```
# ansible all -m shell -a "sysctl -p"
# ansible all -m shell -a "cat /etc/sysctl.conf"
```

Disable Memory Overcommit

To disable Memory Overcommit, follow these steps:

1. Run the following on all nodes. Variable `vm.overcommit_memory=0`

```
# ansible all -m shell -a "echo 'vm.overcommit_memory=0' >> /etc/sysctl.conf"
```

2. Load the settings from default sysctl file `/etc/sysctl.conf` and verify the content of `sysctl.conf`:

```
# ansible all -m shell -a "sysctl -p"
# ansible all -m shell -a "cat /etc/sysctl.conf"
Rhel07.cdpdc.cisco.local | CHANGED | rc=0 >>
# sysctl settings are defined through files in
# /usr/lib/sysctl.d/, /run/sysctl.d/, and /etc/sysctl.d/.
#
# Vendors settings live in /usr/lib/sysctl.d/.
# To override a whole file, create a new file with the same in
```

```
# /etc/sysctl.d/ and put new settings there. To override
# only specific settings, add a file with a lexically later
# name in /etc/sysctl.d/ and put new settings there.
#
# For more information, see sysctl.conf(5) and sysctl.d(5).
net.ipv4.tcp_retries2=5
net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
vm.swappiness = 0
vm.overcommit_memory = 0
```

Disable Transparent Huge Pages

Disabling Transparent Huge Pages (THP) reduces elevated CPU usage caused by THP.

To disable Transparent Huge Pages, follow these steps:

1. You must run the following commands for every reboot; copy this command to `/etc/rc.local` so they are executed automatically for every reboot:

```
# ansible all -m shell -a "echo never > /sys/kernel/mm/transparent_hugepage/enabled"
# ansible all -m shell -a "echo never > /sys/kernel/mm/transparent_hugepage/defrag"
```

2. On the Admin node, run the following commands:

```
#rm -f /root/thp_disable
#echo "echo never > /sys/kernel/mm/transparent_hugepage/enabled" >>
/root/thp_disable
#echo "echo never > /sys/kernel/mm/transparent_hugepage/defrag " >>
/root/thp_disable
```

3. Copy file to each node:

```
# ansible nodes -m copy -a "src=/root/rhell/thp_disable
dest=/root/rhell/thp_disable"
```

4. Append the content of file `thp_disable` to `/etc/rc.d/rc.local`:

```
# ansible nodes -m shell -a "cat /root/rhell/thp_disable >> /etc/rc.d/rc.local"
# ansible nodes -m shell -a "chmod +x /etc/rc.d/rc.local"
```

NTP Configuration

The Network Time Protocol (NTP) is used to synchronize the time of all the nodes within the cluster. The Network Time Protocol daemon (`ntpd`) sets and maintains the system time of day in synchronism with the timeserver located in the admin node (`rhelnn01`). Configuring NTP is critical for any Hadoop Cluster. If server clocks in the cluster drift out of sync, serious problems will occur with HBase and other services.

To configure NTP, follow these steps:

```
# ansible all -m yum -a "name=ntp state=present"
```




Installing an internal NTP server keeps your cluster synchronized even when an outside NTP server is inaccessible.

1. Configure `/etc/ntp.conf` on the admin node only with the following contents:

```
# vi /etc/ntp.conf
driftfile /var/lib/ntp/drift
restrict 127.0.0.1
restrict -6 ::1
server 127.127.1.0
fudge 127.127.1.0 stratum 10
includefile /etc/ntp/crypto/pw
keys /etc/ntp/keys
```

2. Create `/root/ntp.conf` on the admin node and copy it to all nodes:

```
# vi /root/ntp.conf
server 10.12.1.67
driftfile /var/lib/ntp/drift
restrict 127.0.0.1
restrict -6 ::1
includefile /etc/ntp/crypto/pw
keys /etc/ntp/keys
```

3. Copy `ntp.conf` file from the admin node to `/etc` of all the nodes by executing the following commands in the admin node (`rhelnn01`):

```
# ansible nodes -m copy -a "src=/root/ntp.conf dest=/etc/ntp.conf"
```

4. Run the following to synchronize the time and restart NTP daemon on all nodes:

```
# ansible all -m service -a "name=ntpd state=stopped"
# ansible all -m command -a "ntpdate rhelnn01.cdpcdc.cisco.local"
# ansible all -m service -a "name=ntpd state=started"
```

5. Make sure to restart of NTP daemon across reboots:

```
# ansible all -a "systemctl enable ntpd"
```

6. Verify NTP is up and running in all nodes by running the following commands:

```
# ansible all -a "systemctl status ntpd"
```



Alternatively, the new Chrony service can be installed, which is quicker to synchronize clocks in mobile and virtual systems.

7. Install the Chrony service:

```
# ansible all -m yum -a "name=chrony state=present"
```

8. Activate the Chrony service at boot:

```
# ansible all -a "systemctl enable chronyd"
```

9. Start the Chrony service:

```
# ansible all -m service -a "name=chronyd state=started"
# systemctl start chronyd
```

10. The Chrony configuration is in the `/etc/chrony.conf` file, configured similar to `/etc/ntp.conf`.

Install Megaraid StorCLI

This section explains the steps to install StorCLI (Storage Command Line Tool) which is a command line interface designed to be easy to use, consistent, and script. For more details, go to:

<https://docs.broadcom.com/docs/12352476>

To install StorCLI, follow these steps:

1. Download StorCLI: <https://www.broadcom.com/support/download-search/?pg=&pf=&pn=&po=&pa=&dk=storcli>
2. Extract the .zip file and copy `storcli-1.23.02-1.noarch.rpm` from the linux directory.
3. Download StorCLI and its dependencies and transfer to Admin node:

```
# scp storcli-1.23.02-1.noarch.rpm rhelnn01:/root/
```

4. Copy storcli rpm to all the nodes using the following commands:

```
# ansible all -m copy -a "src=/root/storcli-1.23.02-1.noarch.rpm dest=/root/."
```

5. Run this command to install storcli on all the nodes:

```
# ansible all -m shell -a "rpm -ivh storcli-1.23.02-1.noarch.rpm"
```

6. Run this command to copy storcli64 to root directory:

```
# ansible all -m shell -a "cp /opt/MegaRAID/storcli/storcli64 /root/."
```

7. Run this command to check the state of the disks:

```
# ansible all -m shell -a "./storcli64 /c0 show all"
```



The Cisco UCS Manager configuration explains the steps to deploy the required storage configuration via Storage Policy and Storage Profile attached to Service Profile Template for NameNode(s), Management Node(s), GPU Node(s) and DataNode(s). To configure Storage with StorCLI, go to section [Configure Data Drives on Name Node and Other Management Nodes](#).

Configure the Filesystem for NameNodes and DataNodes

The following script formats and mounts the available volumes on each node whether it is NameNode or Data node. OS boot partition will be skipped. All drives are mounted based on their UUID as `/data/disk1`, `/data/disk2`, etc. To configure the filesystem for NameNodes and DataNodes, follow these steps:

1. On the Admin node, create a file containing the following script:

```
#vi /root/driveconf.sh
```

2. To create partition tables and file systems on the local disks supplied to each of the nodes, run the following script as the root user on each node:



This script assumes there are no partitions already existing on the data volumes. If there are partitions, delete them before running the script. This process is detailed in section [Delete Partitions](#).

```
#vi /root/driveconf.sh
#!/bin/bash
[[ "-x" == "${1}" ]] && set -x && set -v && shift 1
count=1
for X in /sys/class/scsi_host/host?/scan
do
echo '- - -' > ${X}
done
for X in /dev/sd?
do
list+=$(echo $X " ")
done
for X in /dev/sd??
do
list+=$(echo $X " ")
done
for X in $list
do
echo "======"
echo $X
echo "======"
if [[ -b ${X} && ` /sbin/parted -s ${X} print quit | /bin/grep -c boot ` -
ne 0
]]
then
echo "$X bootable - skipping."
continue
else
Y=${X##*/}1
echo "Formatting and Mounting Drive => ${X}"
166
/sbin/mkfs.xfs -f ${X}
(( $? )) && continue
#Identify UUID
UUID=`blkid ${X} | cut -d " " -f2 | cut -d "=" -f2 | sed 's"/"/g'`
/bin/mkdir -p /data/disk${count}
(( $? )) && continue
echo "UUID of ${X} = ${UUID}, mounting ${X} using UUID on
/data/disk${count}"
/bin/mount -t xfs -o inode64,noatime,nobarrier -U ${UUID}
/data/disk${count}
(( $? )) && continue
echo "UUID=${UUID} /data/disk${count} xfs inode64,noatime,nobarrier 0
0" >> /etc/fstab
((count++))
```

```
fi
done
```

3. Run the following command to copy `driveconf.sh` to all the nodes:

```
# chmod 755 /root/driveconf.sh
# ansible datanodes -m copy -a "src=/root/driveconf.sh dest=/root/."
# ansible nodes -m file -a "dest=/root/driveconf.sh mode=755"
```

4. Run the following command from the admin node to run the script across all data nodes:

```
# ansible datanodes -m shell -a "/root/driveconf.sh"
```

5. Run the following from the admin node to list the partitions and mount points:

```
# ansible datanodes -m shell -a "df -h"
# ansible datanodes -m shell -a "mount"
# ansible datanodes -m shell -a "cat /etc/fstab"
```

Delete Partitions

To delete a partition, follow these steps:

1. Run the `mount` command (`'mount'`) to identify which drive is mounted to which device `/dev/sd<?>`
2. `umount` the drive for which partition is to be deleted and run `fdisk` to delete as shown below.



Be sure not to delete the OS partition since this will wipe out the OS.

```
# mount
# umount /data/disk1 ← (disk1 shown as example)
# (echo d; echo w;) | sudo fdisk /dev/sd<?>
```

Cluster Verification

This section explains the steps to create the script `cluster_verification.sh` that helps to verify the CPU, memory, NIC, and storage adapter settings across the cluster on all nodes. This script also checks additional prerequisites such as NTP status, SELinux status, ulimit settings, JAVA_HOME settings and JDK version, IP address and hostname resolution, Linux version and firewall settings.

To verify a cluster, follow these steps:



The following script uses cluster shell (clush) which needs to be installed and configured.

1. Create the script `cluster_verification.sh` as shown, on the Admin node (rhelnn01).

```
#vi cluster_verification.sh
#!/bin/bash
shopt -s expand_aliases,
# Setting Color codes
green='\e[0;32m'
```

```

red='\e[0;31m'
NC='\e[0m' # No Color
echo -e "${green} === Cisco UCS Integrated Infrastructure for Big Data and Analytics
\ Cluster Verification === ${NC}"
echo ""
echo ""
echo -e "${green} ==== System Information ==== ${NC}"
echo ""
echo ""
echo -e "${green}System ${NC}"
clush -a -B "`which dmidecode` |grep -A2 '^System Information'"
echo ""
echo ""
echo -e "${green}BIOS ${NC}"
clush -a -B "`which dmidecode` | grep -A3 '^BIOS I'"
echo ""
echo ""
echo -e "${green}Memory ${NC}"
clush -a -B "cat /proc/meminfo | grep -i ^memt | uniq"
echo ""
echo ""
echo -e "${green}Number of Dimms ${NC}"
clush -a -B "echo -n 'DIMM slots: '; `which dmidecode` |grep -c \
'^[[:space:]]*Locator:'"
clush -a -B "echo -n 'DIMM count is: '; `which dmidecode` | grep \"Size\" | grep -c
\"MB\""
clush -a -B "`which dmidecode` | awk '/Memory Device$/,/^$/ {print}' | grep -e
'^Mem' -e Size: -e Speed: -e Part | sort -u | grep -v -e 'NO \ DIMM' -e 'No Module
Installed' -e Unknown"
echo ""
echo ""
# probe for cpu info #
echo -e "${green}CPU ${NC}"
clush -a -B "grep '^model name' /proc/cpuinfo | sort -u"
echo ""
clush -a -B "`which lscpu` | grep -v -e op-mode -e ^Vendor -e family -e Model: -e
Stepping: -e Bogomips -e Virtual -e ^Byte -e ^NUMA node(s)'"
echo ""
echo ""
# probe for nic info #
echo -e "${green}NIC ${NC}"
clush -a -B "`which ifconfig` | egrep '(^e|^p)' | awk '{print \$1}' | \ xargs -l
`which ethtool` | grep -e ^Settings -e Speed"
echo ""
clush -a -B "`which lspci` | grep -i ether"
echo ""
echo ""
# probe for disk info #
echo -e "${green}Storage ${NC}"
clush -a -B "echo 'Storage Controller: '; `which lspci` | grep -i -e \ raid -e
storage -e lsi"
echo ""
clush -a -B "dmesg | grep -i raid | grep -i scsi"
echo ""
clush -a -B "lsblk -id | awk '{print \$1,\$4}'|sort | nl"
echo ""
echo ""

```

```

echo -e "${green} ===== Software ===== ${NC}"
echo ""
echo ""
echo -e "${green}Linux Release ${NC}"
clush -a -B "cat /etc/*release | uniq"
echo ""
echo ""
echo -e "${green}Linux Version ${NC}"
clush -a -B "uname -srvm | fmt"
echo ""
echo ""
echo -e "${green}Date ${NC}"
clush -a -B date
echo ""
echo ""
echo -e "${green}NTP Status ${NC}"
clush -a -B "ntpstat 2>&1 | head -1"
echo ""
echo ""
echo -e "${green}SELINUX ${NC}"
clush -a -B "echo -n 'SElinux status: '; grep ^SELINUX= \ /etc/selinux/config 2>&1"
echo ""
echo ""
clush -a -B "echo -n 'CPUspeed Service: '; `which service` cpuspeed \ status 2>&1"
clush -a -B "echo -n 'CPUspeed Service: '; `which chkconfig` --list \ cpuspeed 2>&1"
echo ""
echo ""
echo -e "${green}Java Version${NC}"
clush -a -B 'java -version 2>&1; echo JAVA_HOME is ${JAVA_HOME:-Not \ Defined!}'
echo ""
echo ""
echo -e "${green}Hostname LoOKup${NC}"
clush -a -B " ip addr show"
echo ""
echo ""
echo -e "${green}Open File Limit${NC}"
clush -a -B 'echo -n "Open file limit(should be >32K): "; ulimit -n'

```

2. Change permissions to executable:

```
# chmod 755 cluster_verification.sh
```

3. Run the Cluster Verification tool from the admin node. This can be run before starting Hadoop to identify any discrepancies in Post OS Configuration between the servers or during troubleshooting of any cluster / Hadoop issues:

```
#./cluster_verification.sh
```

Install Cloudera Data Platform

This section provides instructions for installing Cloudera software, including Cloudera Manager, Cloudera Runtime, and other managed services, in a production environment.

Review the [Cloudera Production Installation: Before You Install](#) steps prior to the production installation of Cloudera Manager, Cloudera Runtime, and other managed services, review the Cloudera Data Platform 7 Requirements and Supported Versions, in addition to the Cloudera Data Platform Release Notes.

Prerequisites for CDP DC Installation

This section details the prerequisites for the CDP DC installation, such as setting up Cloudera Repo.

Cloudera Manager Repository

To setup the Cloudera Manager Repository, follow these steps:

1. From a host connected to the Internet, download the Cloudera's repositories as shown below and transfer it to the admin node:

```
#mkdir -p /tmp/cloudera-repos/
```

2. Download Cloudera Manager Repository:

```
#cd /tmp/cloudera-repos/
# wget https://archive.cloudera.com/cm7/7.1.1/redhat7/yum/cloudera-manager-trial.repo
# reposync --config=./cloudera-manager-trial.repo --repoid=cloudera-manager
# wget https://archive.cloudera.com/cm7/7.1.1/allkeys.asc
```



This downloads the Cloudera Manager RPMs needed for the Cloudera repository.

3. Run the following command to move the RPMs:
4. Copy the repository directory to the admin node (rhelnn01):

```
# scp -r /tmp/cloudera-repos/ rhelnn01:/var/www/html/
# mkdir -p /var/www/html/cloudera-repos/cloudera-manager (On admin node rhelnn01)
# scp allkeys.asc rhelnn01:/var/www/html/cloudera-repos/cloudera-manager/
```

5. On admin node (rhelnn01) run create repo command:

```
#cd /var/www/html/cloudera-repos/
#createrepo --baseurl http://10.12.1.67/cloudera-repos/cloudera-manager/
/var/www/html/cloudera-repos/cloudera-manager/
```



Go to: <http://10.12.1.67/cloudera-repos/cloudera-manager/> to verify the files.

6. Create the Cloudera Manager repo file with following contents:

```
# vi /var/www/html/cloudera-repos/cloudera-manager/cloudera-manager.repo
# cat /var/www/html/cloudera-repos/cloudera-manager/cloudera-manager.repo
[cloudera-repo]
name=Cloudera Manager 7.1.1
baseurl=http://10.12.1.67/cloudera-repos/cloudera-manager/
gpgcheck=0
```

```
enabled=1
```

- Copy the file `cloudera-repo.repo` into `/etc/yum.repos.d/` on the admin node to enable it to find the packages that are locally hosted:

```
#cp /var/www/html/cloudera-repos/cloudera-manager/cloudera-manager.repo
/etc/yum.repos.d/
From the admin node copy the repo files to /etc/yum.repos.d/ of all the nodes of the
cluster:
# ansible all -m copy -a "src=/etc/yum.repos.d/cloudera-manager.repo
dest=/etc/yum.repos.d/."
```

Set Up the Local Parcels for CDP DC 7.1.1

From a host connected the internet, download CDP DC 7.1.1 parcels that are meant for RHEL7.7 from the URL: <https://archive.cloudera.com/cdh7/7.1.10/parcels/> and place them in the directory `/var/www/html/cloudera-repos/` of the Admin node.

The following are the required files for RHEL7.7:

- `CDH-7.1.1-1.cdh7.1.1.p0.3266817-el7.parcel`
- `CDH-7.1.1-1.cdh7.1.1.p0.3266817-el7.parcel.sha256`
- `manifest.json`

Download Parcels

To download parcels, follow these steps:

- From a host connected to the Internet, download the Cloudera's parcels as shown below and transfer it to the admin node:

```
#mkdir -p /tmp/cloudera-repos/CDH7.1.1.0parcels
```

- Download parcels:

```
#cd /tmp/cloudera-repos/CDH7.1.1.0parcels
# wget https://archive.cloudera.com/cdh7/7.1.1.0/parcels/CDH-7.1.1-
1.cdh7.1.1.p0.3266817-el7.parcel
# wget https://archive.cloudera.com/cdh7/7.1.1.0/parcels/CDH-7.1.1-
1.cdh7.1.1.p0.3266817-el7.parcel.sha256
# wget https://archive.cloudera.com/cdh7/7.1.1.0/parcels/manifest.json
```

- Copy `/tmp/cloudera-repos/CDH7.1.1.0parcels` to the admin node (`rhelnn01`):

```
# mkdir -p /var/www/html/cloudera-repos/cdh7/7.1.1.0/parcels/ (on rhelnn01)
# scp -r /tmp/cloudera-repos/CDH7.1.1.0parcels/ rhelnn01:/var/www/html/cloudera-
repos/cdh7/7.1.1.0/parcels/
# chmod -R ugo+rx /var/www/html/cloudera-repos/cdh7
```

- Verify that these files are accessible by visiting the URL <http://10.12.1.67/cloudera-repos/cdh7/7.1.1.0/parcels/> in admin node.

5. Download Sqoop Connectors.

```
# mkdir -p /tmp/cloudera-repos/sqoop-connectors
# wget --recursive --no-parent --no-host-directories
http://archive.cloudera.com/sqoop-connectors/parcels/latest/ -P /tmp/cloudera-repos/
```

6. Copy /tmp/cloudera-repos/sqoop-connectors to the admin node (rhelnn01).

```
# scp -r /tmp/cloudera-repos/sqoop-connectors rhelnn01:/var/www/html/cloudera-repos/
# sudo chmod -R ugo+rX /var/www/html/cloudera-repos/sqoop-connectors
```

Install and Configure Database for Cloudera Manager

You will set up the following for Cloudera Manager:

- Install the PostgreSQL Server
- Installing the psycopg2 Python Package
- Configure and Start the PostgreSQL Server

Install PostgreSQL Server

To install the PostgreSQL packages on the PostgreSQL server, follow these steps:

1. In the admin node where Cloudera Manager will be installed, use the following command to install PostgreSQL server.

```
#yum -y install postgresql10-server postgresql10
```

2. Install psycopg2 Python package 2.7.5 or higher if lower version is installed.

```
# yum install -y python-pip
# pip install psycopg2==2.7.5 --ignore-installed
```



Check installing dependencies for hue:

https://docs.cloudera.com/documentation/enterprise/upgrade/topics/ug_cdh_upgrade_hue_psycopg2.html

Configure and Start PostgreSQL Server

To configure and start the PostgreSQL server, follow these steps:

1. To configure and start the PostgreSQL Server, stop PostgreSQL server if it is running.

```
# systemctl stop postgresql-10.service
```



Backup the existing database.



By default, PostgreSQL only accepts connections on the loopback interface. You must reconfigure PostgreSQL to accept connections from the fully qualified domain names (FQDN) of the hosts hosting the services for which you are configuring databases. If you do not make these changes, the services cannot connect to and use the database on which they depend.

2. Make sure that LC_ALL is set to en_US.UTF-8 and initialize the database as follows:

```
# echo 'LC_ALL="en_US.UTF-8"' >> /etc/locale.conf
# sudo /usr/pgsql-10/bin/postgresql-10-setup initdb
#
```

3. To enable MD5 authentication, edit /var/lib/pgsql/data/pg_hba.conf by adding the following line:

```
# host all all 127.0.0.1/32 md5
```



The host line specifying md5 authentication shown above must be inserted before this ident line:

```
# host all 127.0.0.1/32 ident
```

Failure to do so may cause an authentication error when running the scm_prepare_database.sh script. You can modify the contents of the md5 line shown above to support different configurations. For example, if you want to access PostgreSQL from a different host, replace 127.0.0.1 with your IP address and update postgresql.conf, which is typically found in the same place as pg_hba.conf, to include:

```
# listen_addresses = '*'
```

4. Configure settings to ensure your system performs as expected. Update these settings in the /var/lib/pgsql/data/postgresql.conf file. Settings vary based on cluster size and resources as follows:

```
max_connection - 500
shared_buffers - 1024 MB
wal_buffers - 16 MB
max_wal_size - 6GB (checkpoint_segments - 128)
checkpoint_completion_target - 0.9
```



Refer to section Configuration and Starting the PostgreSQL Server, in the Cloudera Data Platform Data Center Installation guide: <https://docs.cloudera.com/cloudera-manager/7.1.1/installation/topics/cdpdc-install-config-postgresql-for-cdp.html>

5. Start the PostgreSQL Server and configure to start at boot.

```
# systemctl start postgresql-10.service
# systemctl enable postgresql-10.service
```

Databases for CDP

Create databases and service accounts for components that require a database.

Create databases for the following components:

- Cloudera Manager Server
- Cloudera Management Service Roles: Activity Monitor, Reports Manager, Hive Metastore Server, Data Analytics Studio, Ranger, hue, and oozie.

The databases must be configured to support the PostgreSQL UTF8 character set encoding.

Record the values you enter for database names, usernames, and passwords. The Cloudera Manager installation wizard requires this information to correctly connect to these databases.

To create databases for CDP, follow these steps:

1. In the admin node, connect to PostgreSQL:

```
# sudo -u postgres psql
```

2. Create databases using the following command:

```
CREATE ROLE scm LOGIN PASSWORD 'password';
CREATE DATABASE scm OWNER scm ENCODING 'UTF8';

CREATE ROLE amon LOGIN PASSWORD 'password';
CREATE DATABASE amon OWNER amon ENCODING 'UTF8';

CREATE ROLE rman LOGIN PASSWORD 'password';
CREATE DATABASE rman OWNER rman ENCODING 'UTF8';

CREATE ROLE hue LOGIN PASSWORD 'password';
CREATE DATABASE hue OWNER hue ENCODING 'UTF8';

CREATE ROLE hive LOGIN PASSWORD 'password';
CREATE DATABASE metastore OWNER hive ENCODING 'UTF8';

CREATE ROLE nav LOGIN PASSWORD 'password';
CREATE DATABASE nav OWNER nav ENCODING 'UTF8';

CREATE ROLE navms LOGIN PASSWORD 'password';
CREATE DATABASE navms OWNER navms ENCODING 'UTF8';

CREATE ROLE oozie LOGIN PASSWORD 'password';
CREATE DATABASE oozie OWNER oozie ENCODING 'UTF8';

CREATE ROLE rangeradmin LOGIN PASSWORD 'password';
CREATE DATABASE ranger OWNER rangeradmin ENCODING 'UTF8';

CREATE ROLE das LOGIN PASSWORD 'password';
CREATE DATABASE das OWNER das ENCODING 'UTF8';

ALTER DATABASE metastore SET standard_conforming_strings=off;
ALTER DATABASE oozie SET standard_conforming_strings=off;
```



For Apache Ranger specific configuration for PostgreSQL, see: [Configuring a PostgreSQL Database for Ranger](#)

Cloudera Manager Installation

The following sections describe how to install Cloudera Manager and then using Cloudera Manager to install CDP DC 7.1.1.

Install Cloudera Manager

Cloudera Manager, an end-to-end management application, is used to install and configure CDP DC. During CDP Installation, Cloudera Manager's Wizard will help to install Hadoop services and any other role(s)/service(s) on all nodes using the following procedure:

- Discovery of the cluster nodes
- Configure the Cloudera parcel or package repositories
- Install Hadoop, Cloudera Manager Agent (CMA) and Impala on all the cluster nodes.
- Install the Oracle JDK or Open JDK if it is not already installed across all the cluster nodes.
- Assign various services to nodes.
- Start the Hadoop services



Please see the [JAVA requirements](#) for CDP DC.

To install Cloudera Manager, follow these steps:

1. Update the repo files to point to local repository.

```
#rm -f /var/www/html/clouderarepo/*.repo
#cp /etc/yum.repos.d/c*.repo /var/www/html/clouderarepo/
```

2. Install the Oracle Java Development Kit on the Cloudera Manager Server host.

```
# ansible nodes -m shell -a "yum install -y java-1.8.0-openjdk-devel"
```



Please see the CDP DC documentation for more information: [Manually Installing OpenJDK](#) and [Manually Installing Oracle JDK](#)

3. Install the Cloudera Manager Server packages either on the host where the database is installed, or on a host that has access to the database:

```
#yum install -y cloudera-manager-agent cloudera-manager-daemons cloudera-manager-server
```

Set Up the Cloudera Manager Server Database

The Cloudera Manager Server Database includes a script that can create and configure a database for itself.

The script can:

- Create the Cloudera Manager Server database configuration file.
- (PostgreSQL) Create and configure a database for Cloudera Manager Server to use.
- (PostgreSQL) Create and configure a user account for Cloudera Manager Server.

The following sections describe the syntax for the script and demonstrate how to use it.

Prepare a Cloudera Manager Server External Database

To prepare a Cloudera Manager Server external database, follow these steps:

1. Run the `scm_prepare_database.sh` script on the host where the Cloudera Manager Server package is installed (rhelnn01) admin node:

```
# cd /opt/cloudera/cm/schema/  
# ./scm_prepare_database.sh postgresql scm scm <password>  
# ./scm_prepare_database.sh postgresql amon amon <password>  
# ./scm_prepare_database.sh postgresql rman rman <password>  
# ./scm_prepare_database.sh postgresql hue hue <password>  
# ./scm_prepare_database.sh postgresql metastore hive <password>  
# ./scm_prepare_database.sh postgresql oozie oozie<password>  
# ./scm_prepare_database.sh postgresql das das <password>  
# ./scm_prepare_database.sh postgresql ranger rangeradmin <password>
```

Start the Cloudera Manager Server

To start the Cloudera Manager Server, follow these steps:

1. Start the Cloudera Manager Server:

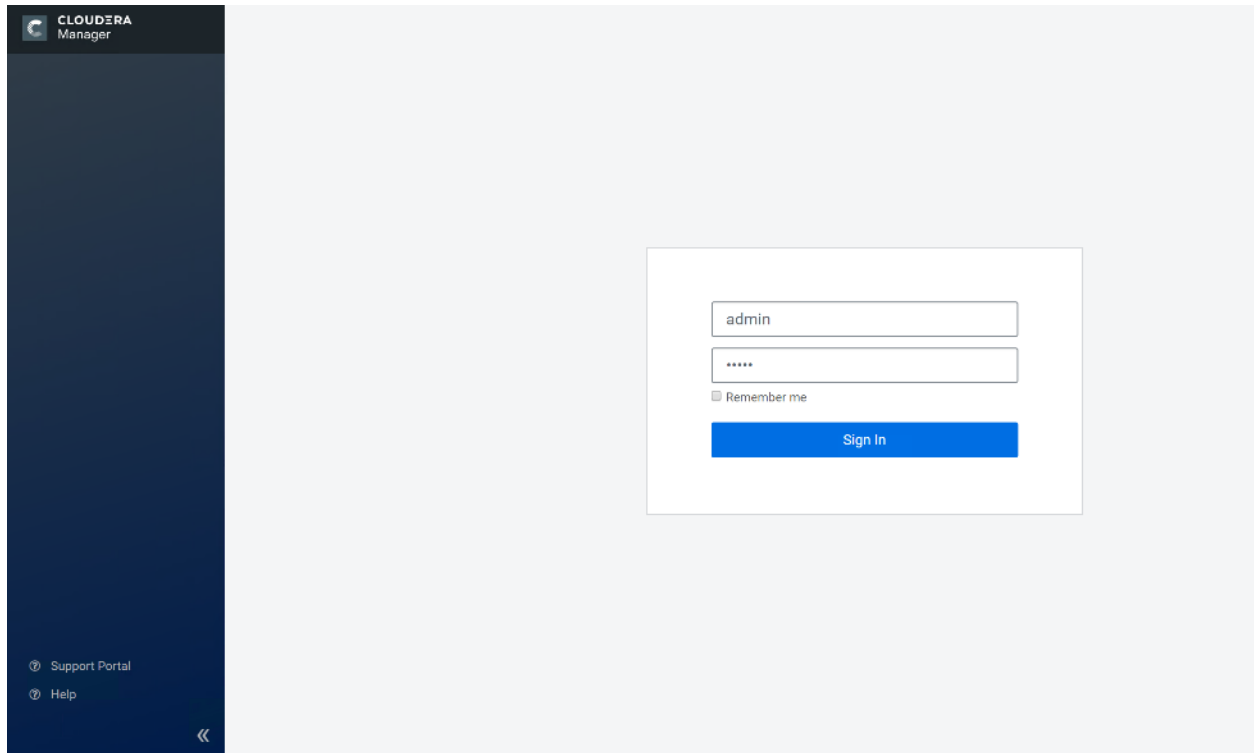
```
#systemctl start cloudera-scm-server
```

2. Access the Cloudera Manager using the URL, <http://10.12.1.57:7180> to verify that the server is up.
3. Once the installation of Cloudera Manager is complete, install CDP DC 7 using the Cloudera Manager Web interface.

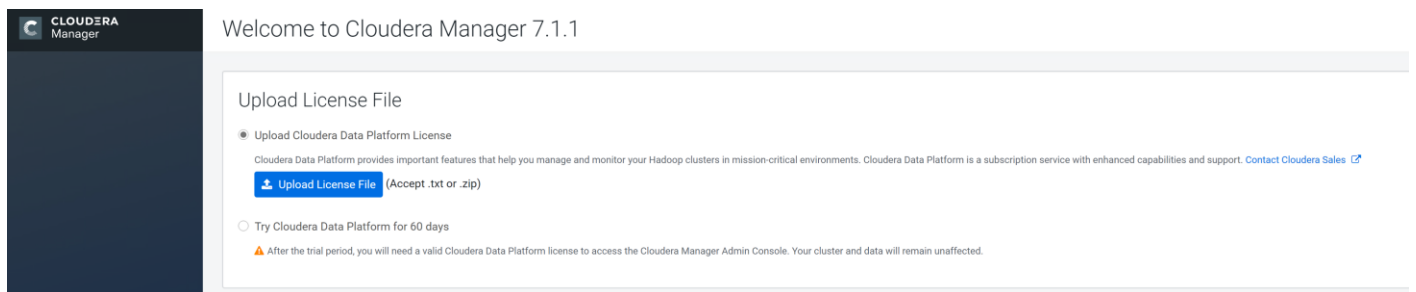
Install Cloudera Data Platform Data Center (CDP DC 7)

To install the Cloudera Data Platform Data Center, follow these steps:

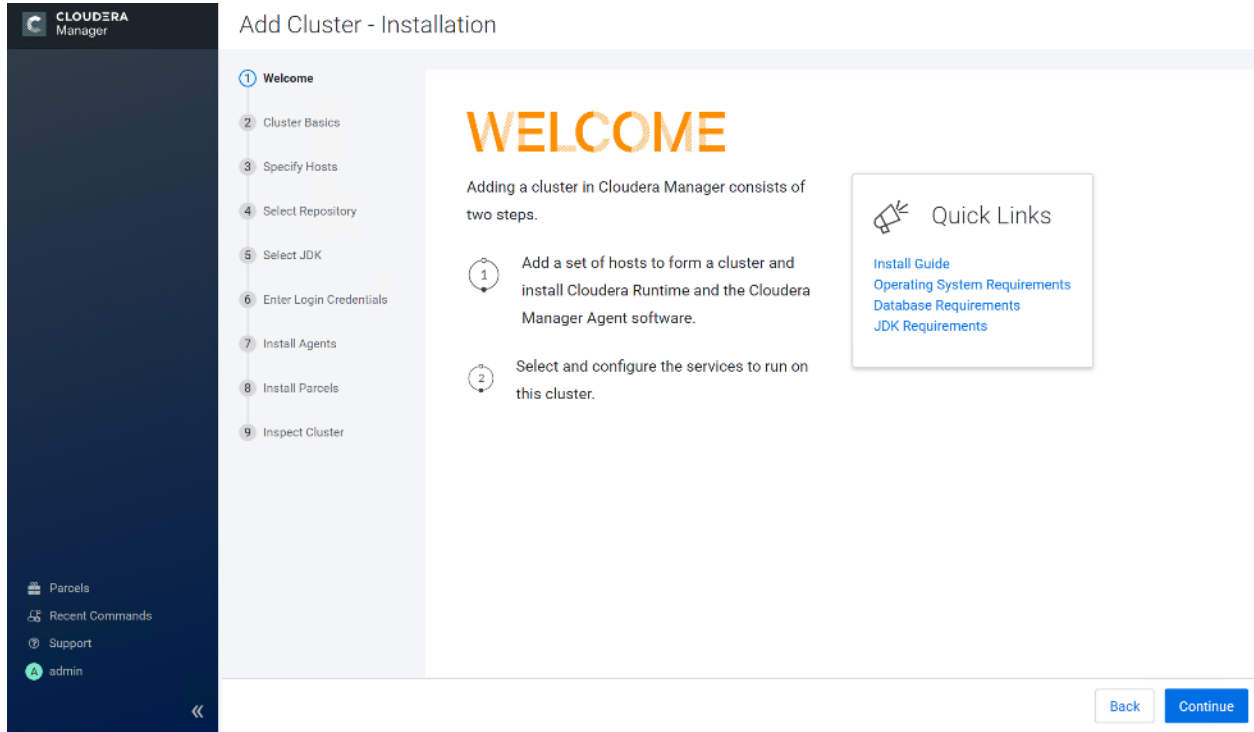
1. Log into the Cloudera Manager. Enter "admin" for both the Username and Password fields.



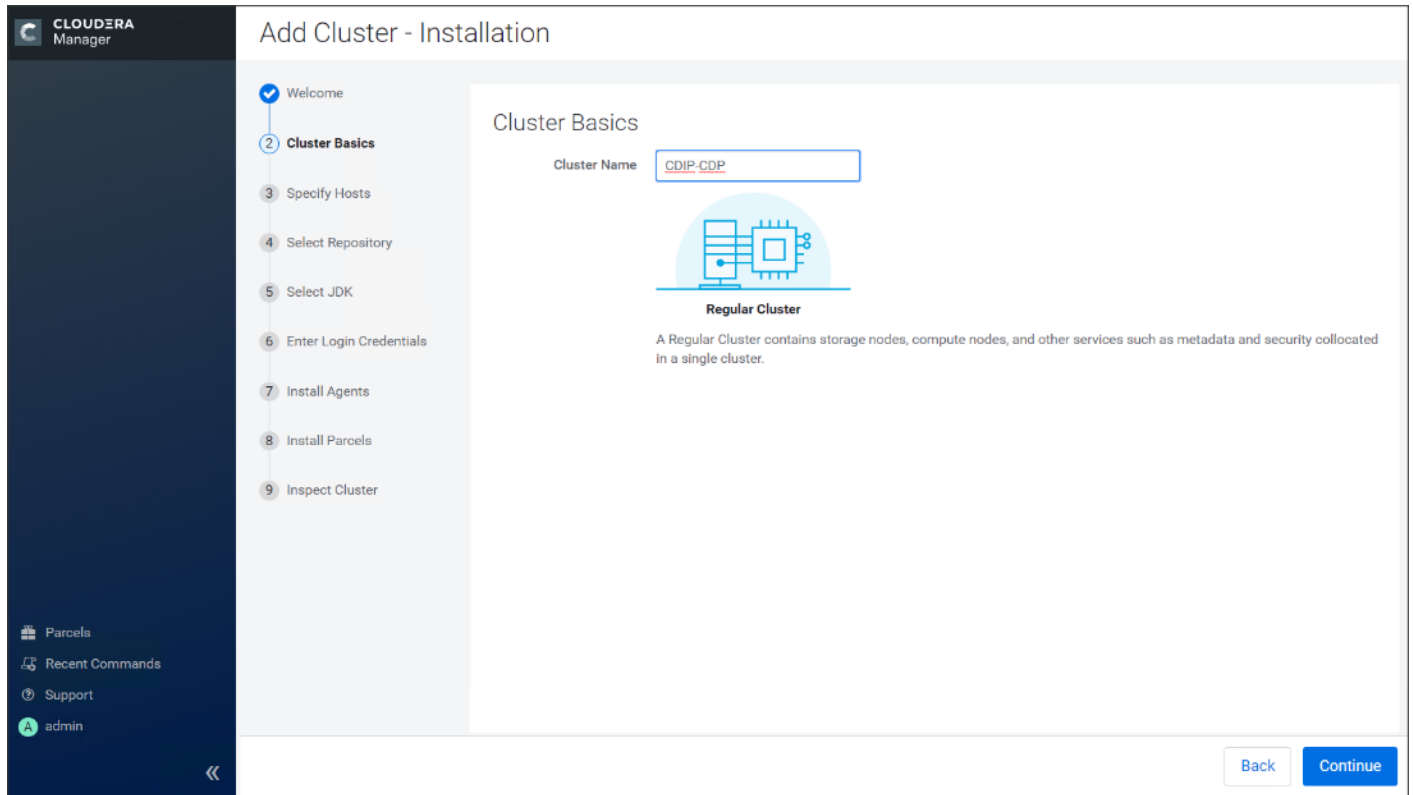
2. Upload license file. Click Continue after successfully uploading license for CDP DC.



3. Click Continue on the Welcome screen.



4. Enter name for the Cluster.



5. Specify the hosts that are part of the cluster using their IP addresses or hostname. The figure below shows a pattern that specifies the IP addresses range.

```
10.12.1.[51-66] or rhel[01-16].cdpdc.cisco.local
10.12.1.[67-69] or rhelnn[01-03].cdpdc.cisco.local
```

6. After the IP addresses or hostnames are entered, click Search.

Specify Hosts

Hosts should be specified using the same hostname (FQDN) that they will identify themselves with. Cloudera recommends including Cloudera Manager Server's host. This also enables health monitoring for that host.

Hostname:

Hint: Search for hostnames or IP addresses using [patterns](#)

SSH Port:

8 hosts scanned, 7 running SSH.
Click the first checkbox, hold down the Shift key and click the last checkbox to select a range.

<input checked="" type="checkbox"/>	Expanded Query	Hostname (FQDN)	IP Address	Currently Managed	Result
<input checked="" type="checkbox"/>		rhel07	10.12.1.57	No	Host was successfully scanned.
<input checked="" type="checkbox"/>		rhel08	10.12.1.58	No	Host was successfully scanned.
<input checked="" type="checkbox"/>		rhel09	10.12.1.59	No	Host was successfully scanned.
<input checked="" type="checkbox"/>		rhel10	10.12.1.60	No	Host was successfully scanned.
<input checked="" type="checkbox"/>		rhel11	10.12.1.61	No	Host was successfully scanned.
<input type="checkbox"/>		rhel12	10.12.1.62	No	Could not connect to host.
<input checked="" type="checkbox"/>		rhel15	10.12.1.65	No	Host was successfully scanned.
<input checked="" type="checkbox"/>		rhel16	10.12.1.66	No	Host was successfully scanned.

1 - 8 of 8

7. Cloudera Manager will "discover" the nodes in the cluster. Verify that all desired nodes have been found and selected for installation.

Edit the Cloudera Data Platform Data Center Parcel Settings to Use the CDP 7.1.1 Parcels

To edit the CDP DC Parcel settings, follow these steps:

1. Add custom repository path for Cloudera Manager local repository created.

CLUSTER
Add Cluster - Installation

1 Welcome
2 Cluster Basics
3 Specify Hosts
4 **Select Repository**
5 Select JDK
6 Enter Login Credentials
7 Install Agents
8 Install Parcels
9 Inspect Cluster

Select Repository

Cloudera Manager Agent

Cloudera Manager Agent 7.1.1 (#3274282) needs to be installed on all new hosts.

Repository Location Public Cloudera Repository

Ensure the above version is listed in <https://archive.cloudera.com/p/cm7> and that you have access to that repository. Requires direct Internet access on all hosts.

Custom Repository

Example: `http://LOCAL_SERVER/c/Cloudera-repos/cm7/7.1.1`
Do not include operating system-specific paths in the URL. The path will be automatically derived.
Learn more at [How to set up a custom repository](#).

CDH and other software

Cloudera recommends the use of parcels for installation over packages, because parcels enable Cloudera Manager to easily manage the software on your cluster, automating the deployment and upgrade of service binaries. Electing not to use parcels will require you to manually upgrade packages on all hosts in your cluster when software updates are available, and will prevent you from using Cloudera Manager's rolling upgrade capabilities.

Install Method Use Packages

Use Parcels (Recommended) [Parcel Repositories & Network Settings](#) [Other Parcel Configurations](#)

Version Versions that are too new for this version of Cloudera Manager (7.1.1) will not be shown.

Cloudera Runtime 7.1.1-1.cdh7.1.1.p0.3266817

Additional Parcels ACCUMULO 1.9.2-1.ACCUMULO6.1.0.p0.908695
 ACCUMULO 1.7.2-5.5.0.ACCUMULO5.5.0.p0.8

[Back](#) [Continue](#)

2. On the Cloudera Manager installation wizard, click Parcels.
3. Click Parcel Repositories and Network Settings.

CLUSTER
Parcels

Parcel Usage **Parcel Repositories & Network Settings** Other Parcel Configurations Check for New Parcels

Search

Clusters
Hosts
Diagnostics
Audits
Charts
Replication
Administration

Location

Available Remotely

Filters

No clusters found.

4. Click to remove the entire remote repository URLs and add the URL to the location where we kept the CDP DC 7.1.1 parcels i.e. <http://10.12.1.67/cloudera-repos/cdh7/7.1.1.0/parcels/>

Parcel Repository & Network Settings

Cloudera Manager checks the connection to the configured parcel repository URLs. A valid license is required to access most Cloudera parcel repositories. Last Updated: Jun 23, 4:17:16 PM PDT

> 8/8 URL(s) - The repository was successfully accessed and the manifest downloaded and validated. (HTTP Status: 200)

Remote Parcel Repository URLs

[Undo](#)

Enable Automatic Authentication for Cloudera Repositories

HTTP authentication username override for Cloudera Repositories

HTTP authentication password override for Cloudera Repositories

Proxy Server

Proxy Port

Proxy User

Reason for change: [Close](#) [Save & Verify Configuration](#)

5. Click Save Changes to finish the configuration.
6. Click Continue on the confirmation page.
7. For the method of installation, select the Use Parcels (Recommended) radio button.
8. For the CDP DC 7 version, select the Cloudera Runtime 7.1.1-1.cdh7.1.1.p0.3266817 radio button.
9. For the specific release of Cloudera Manager, select the Custom Repository radio button.
10. Enter the URL for the repository within the admin node. <http://10.12.1.67/clouderarepo/cloudera-manager> and click Continue.

The screenshot shows the 'Add Cluster - Installation' wizard in Cloudera Manager. The left sidebar lists steps: Welcome, Cluster Basics, Specify Hosts, **Select Repository** (current), Select JDK, Enter Login Credentials, Install Agents, Install Parcels, and Inspect Cluster. The main content area is titled 'Select Repository' and 'Cloudera Manager Agent'. It states that Cloudera Manager Agent 7.1.1 (#3274282) needs to be installed on all new hosts. Under 'Repository Location', the 'Custom Repository' option is selected. A text input field contains the URL 'http://10.12.1.57/cloudera-repos/cloudera-manager/'. Below this, an example URL is provided: 'Example: http://LOCAL_SERVER/cloudera-repos/cm7/7.1.1'. A note says 'Do not include operating system-specific paths in the URL. The path will be automatically derived.' Under 'CDH and other software', the 'Use Packages' option is selected. A note says 'Versions that are too new for this version of Cloudera Manager (7.1.1) will not be shown.' A single version is listed: 'Cloudera Runtime 7.1.1-1.odh7.1.1.p0.3266817'. At the bottom right, there are 'Back' and 'Continue' buttons.

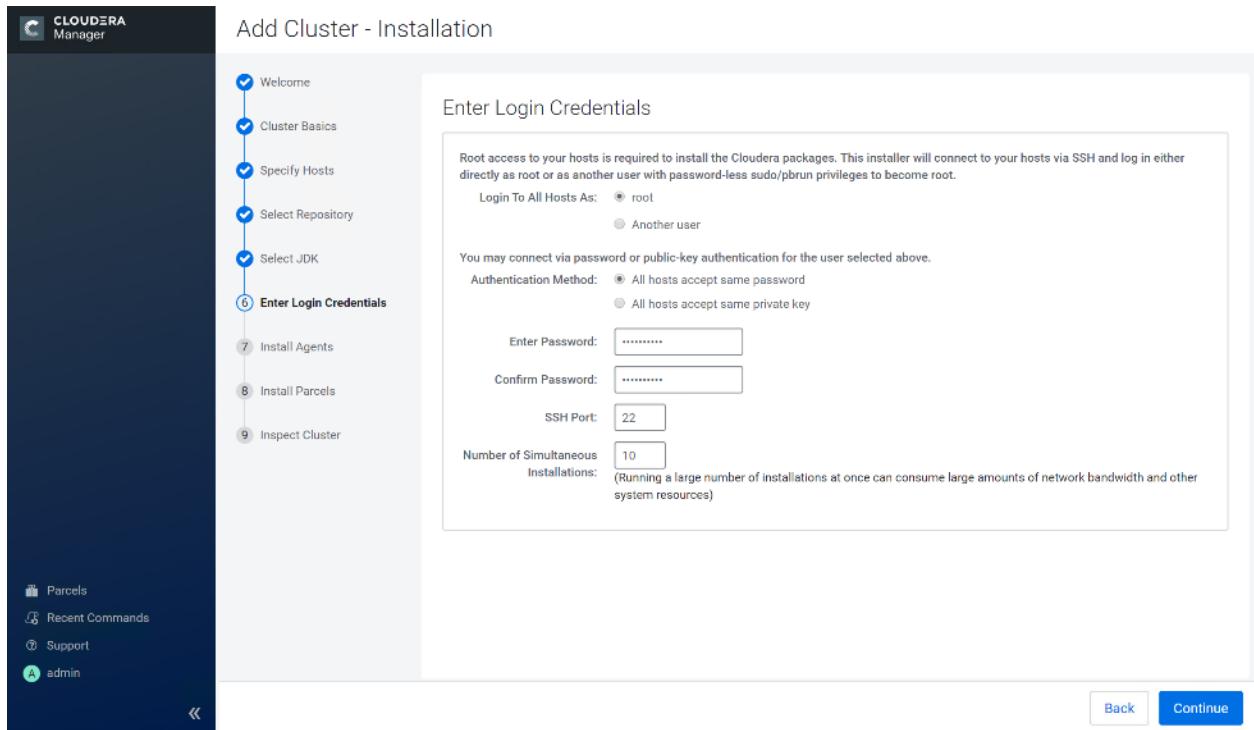
11. Select appropriate option for JDK.

The screenshot shows the 'Add Cluster - Installation' wizard in Cloudera Manager. The left sidebar lists steps: Welcome, Cluster Basics, Specify Hosts, Select Repository, **Select JDK** (current), Enter Login Credentials, Install Agents, Install Parcels, and Inspect Cluster. The main content area is titled 'Select JDK'. It shows a table with 'Selected Version' as 'Cloudera Runtime 7.1' and 'Supported JDK Version' as 'OpenJDK 8, 11 or Oracle JDK 8, 11'. A link for 'More details on supported JDK version' is present. A note states: 'If you plan to use JDK 11, you will need to install it manually on all hosts and then select the **Manually manage JDK** option below.' The 'Manually manage JDK' option is selected. A blue information box contains the text: 'Please ensure that a supported JDK is **already installed** on all hosts. You will need to manage installing the unlimited strength JCE policy file, if necessary.' Below this, two other options are listed: 'Install a Cloudera-provided version of OpenJDK' and 'Install a system-provided version of OpenJDK'. At the bottom right, there are 'Back' and 'Continue' buttons.

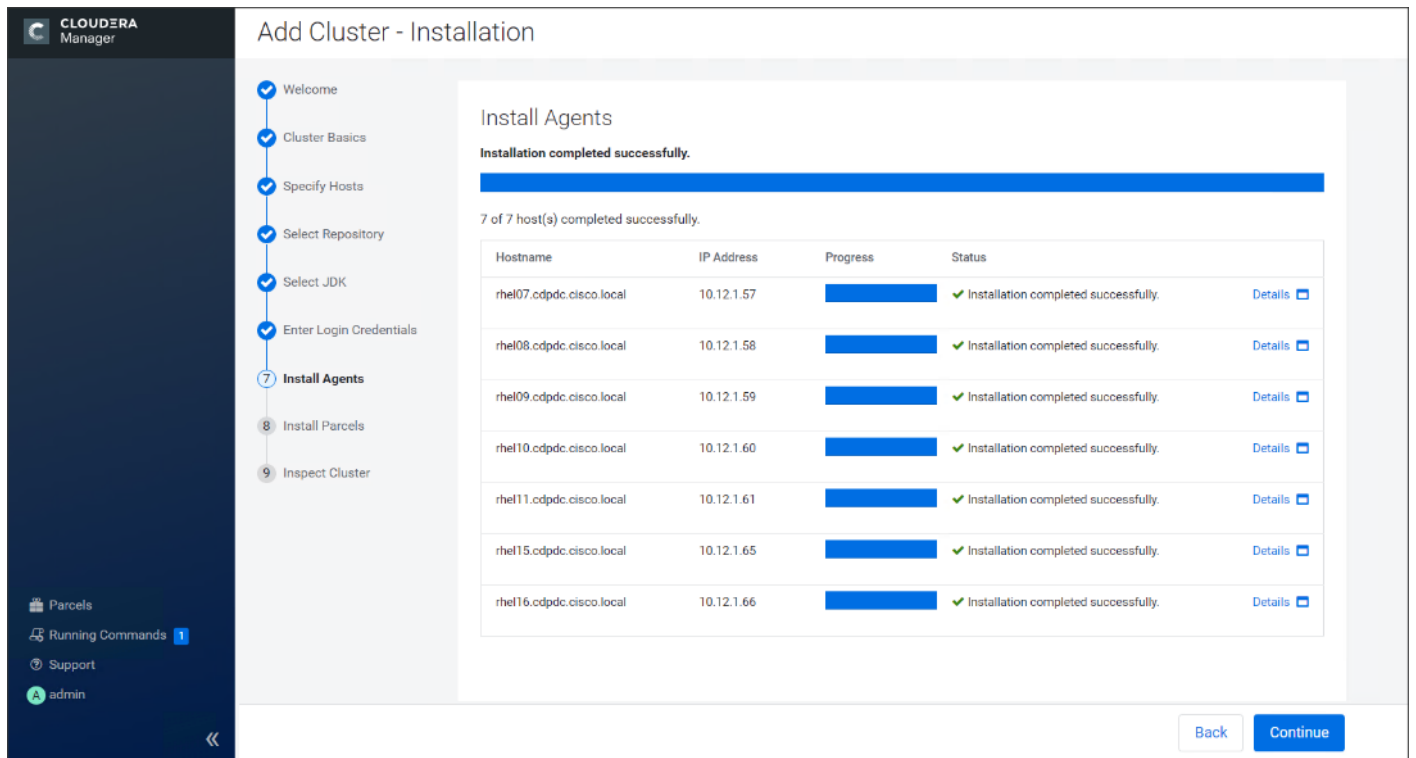


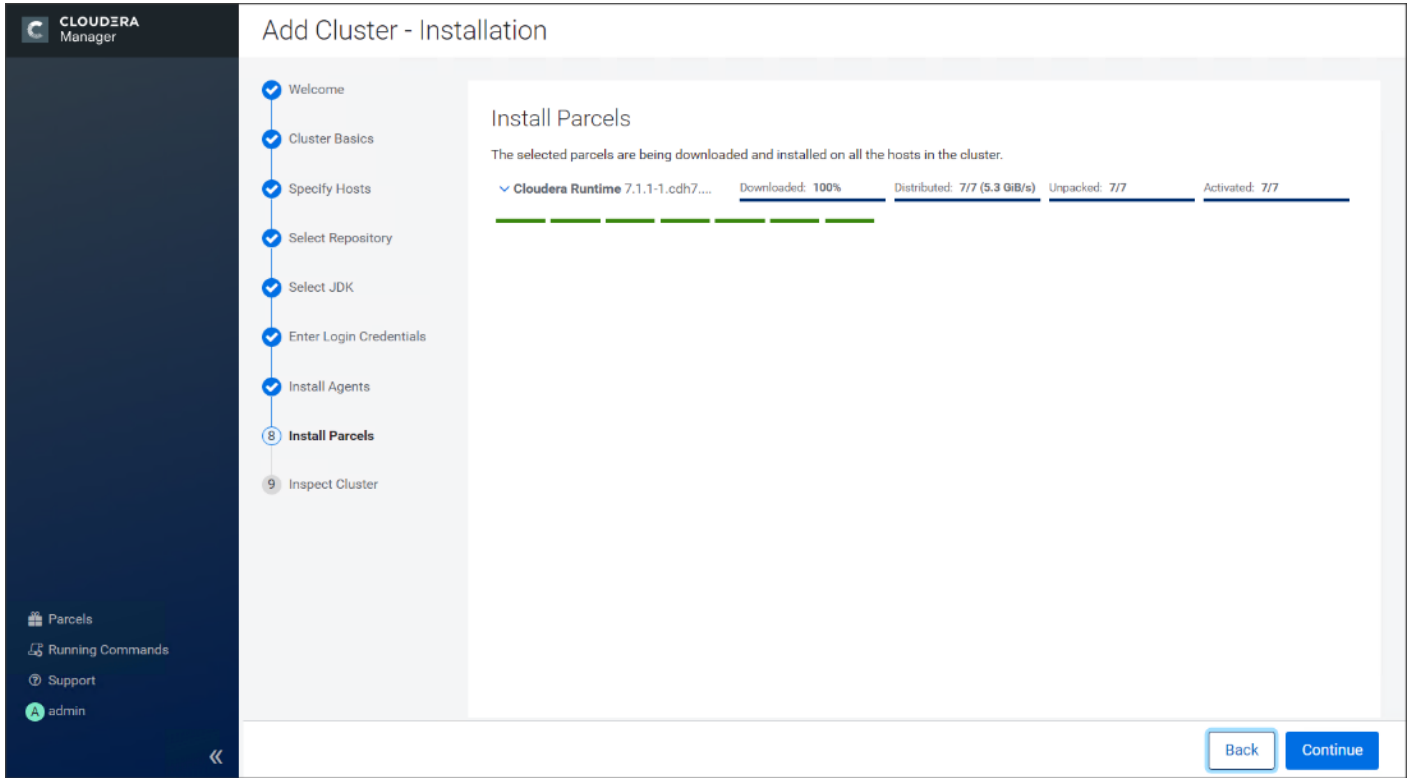
We selected the Manually Manager JDK option as shown in the screenshot above.

12. Provide SSH login credentials for the cluster and click Continue.



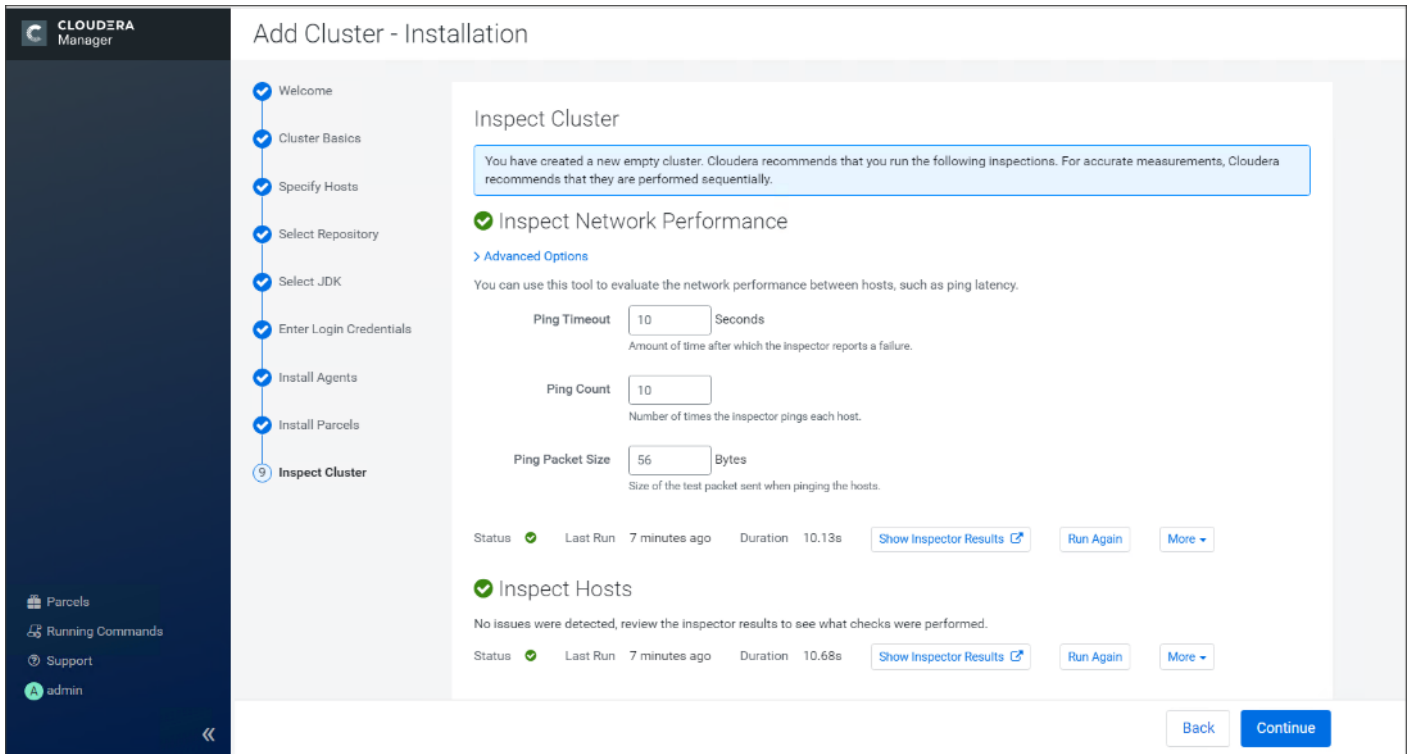
The installation of the local Cloudera repository and using parcels begins.



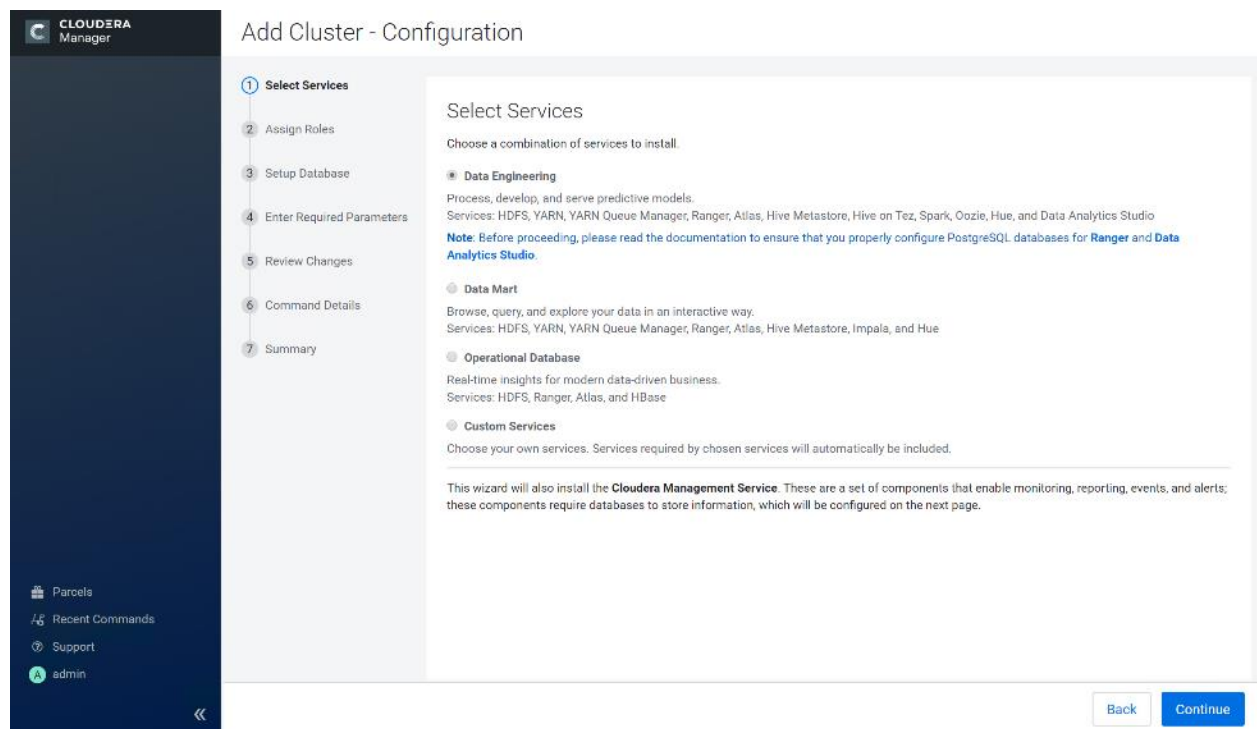


13. Run the inspect the hosts and network performance test through Cloudera Manager on which it has just performed the installation.

14. Review and verify the summary. Click Continue.



15. Select services that need to be started on the cluster.



 We selected Custom Services for this study.

16. This is a critical step in the installation: Inspect and customize the role assignments of all the nodes based on your requirements and click Continue.

17. Reconfigure the service assignment to match [Table 5](#).

Table 5 Service/Role Assignment

Service Name	Host
NameNode	rhelnn01, rhelnn02 (HA)
HistoryServer	rhelnn02
JournalNodes	rhelnn01, rhelnn02, rhelnn03
ResourceManager	rhelnn02, rhelnn03 (HA)
Hue Server	rhelnn02
HiveMetastore Server	rhelnn01
HiveServer2	rhelnn02
HBase Master	rhelnn02
Oozie Server	rhelnn01
ZooKeeper	rhelnn01, rhelnn02, rhelnn03

Service Name	Host
DataNode	rhel01 to rhel16
NodeManager	rhel01 to rhel16
RegionServer	rhel01 to rhel16
Sqoop Server	rhelnn01
Impala Catalog Server Daemon	rhelnn01
Impala State Store	rhelnn02
Impala Daemon	rhel01 to rhel16
Solr Server	rhel01 (can be installed on all hosts if needed if there is a search use case)
Spark History Server	rhelnn01
Spark Executors	rhel01 to rhel16

Figure 53 Assign Roles in Cloudera Manager; Cluster Creation Wizard Example

The screenshot displays the 'Add Cluster - Configuration' wizard in Cloudera Manager, specifically the 'Assign Roles' step. The left sidebar shows a progress indicator with seven steps: 1. Select Services, 2. Assign Roles (current), 3. Setup Database, 4. Enter Required Parameters, 5. Review Changes, 6. Command Details, and 7. Summary. The main content area is titled 'Assign Roles' and includes a warning about incorrect assignments. Below this, there are sections for assigning roles to various services:

- Data Analytics Studio:** Two roles are shown: 'Data Analytics Studio Webapp Server' and 'Data Analytics Studio Eventprocessor', both assigned to 'rhel07.cdpsc.cisco.local'.
- HBase:** Three roles are shown: 'Master' (3 New) assigned to 'rhel[07, 09, 11].cdpsc.cisco.local', 'HBase REST Server' (1 New) assigned to 'rhel07.cdpsc.cisco.local', and 'HBase Thrift Server' (1 New) assigned to 'rhel09.cdpsc.cisco.local'. A 'RegionServer' (7 New) role is also shown, assigned to 'Same As DataNode'.
- HDFS:** Three roles are shown: 'NameNode' (1 New) assigned to 'rhel09.cdpsc.cisco.local', 'SecondaryNameNode' (1 New) assigned to 'rhel11.cdpsc.cisco.local', and 'Balancer' (1 New) assigned to 'rhel07.cdpsc.cisco.local'. Additionally, 'HttpFS' and 'NFS Gateway' roles are shown with 'Select hosts' as their assignment, and a 'DataNode' (7 New) role is assigned to 'rhel[07-11, 15-16].cdpsc.cisco.local'.

At the bottom right of the main area, there are 'Back' and 'Continue' buttons.

CLUSTER
CLUSTER NAME:

HDFS

- NameNode × 1 New:
- SecondaryNameNode × 1 New:
- Balancer × 1 New:
- HttpFS:
- NFS Gateway:
- DataNode × 7 New:

Hive

- Gateway × 7 New:
- Hive Metastore Server × 3 New:
- WebHCat Server × 1 New:
- HiveServer2 × 3 New:

Hive on Tez

- Gateway × 7 New:
- HiveServer2 × 3 New:

Hue

- Hue Server × 3 New:
- Load Balancer × 3 New:

Impala

- Impala StateStore × 1 New:
- Impala Catalog Server × 1 New:
- Impala Daemon × 7 New:

Navigation: Parcels, Running Commands, Support, admin

Buttons: Back, Continue

CLUSTER
CLUSTER NAME:

Key-Value Store Indexer

- Lily HBase Indexer × 1 New:

Cloudera Management Service

- Service Monitor × 1 New:
- Activity Monitor × 1 New:
- Host Monitor × 1 New:
- Reports Manager × 1 New:
- Event Server × 1 New:
- Alert Publisher × 1 New:
- Telemetry Publisher:

Oozie

- Oozie Server × 1 New:

Solr

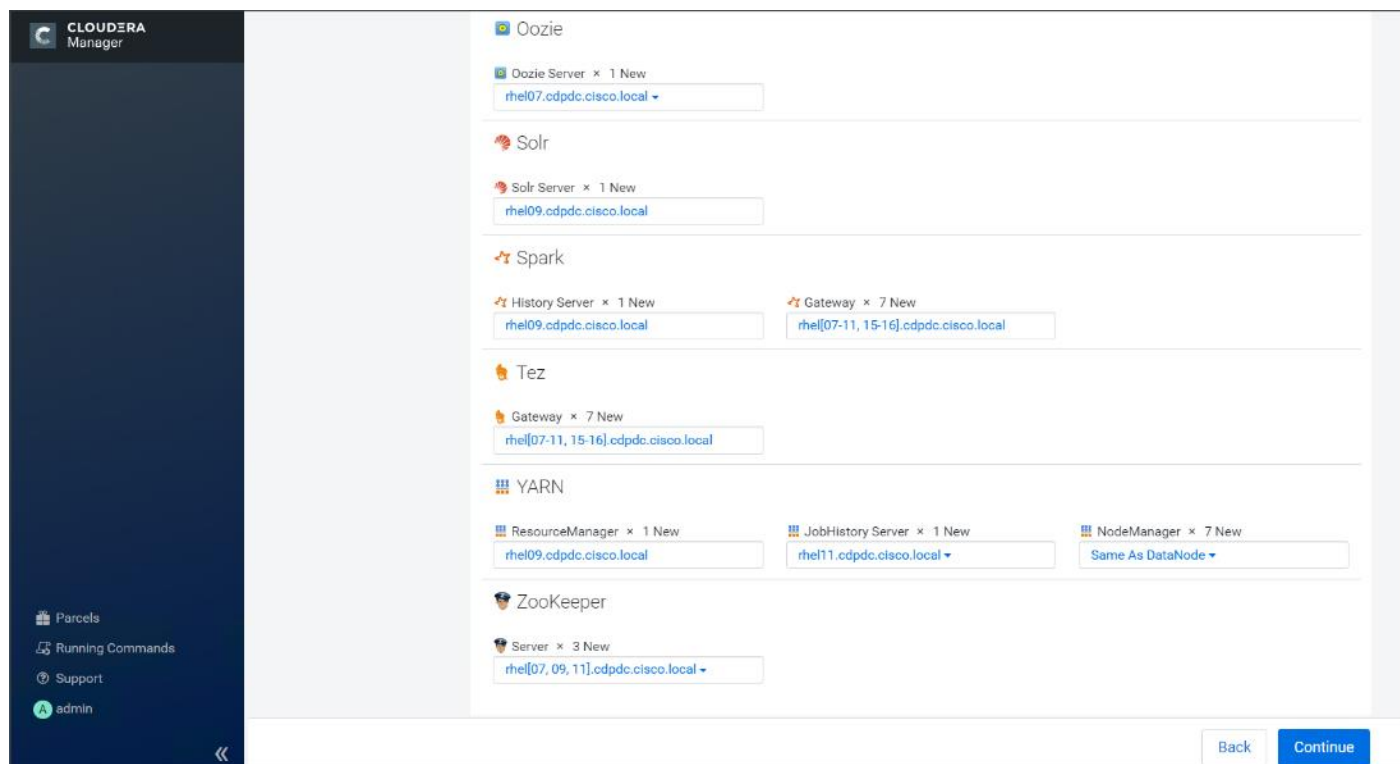
- Solr Server × 1 New:

Spark

- History Server × 1 New:
- Gateway × 7 New:

Navigation: Parcels, Running Commands, Support, admin

Buttons: Back, Continue

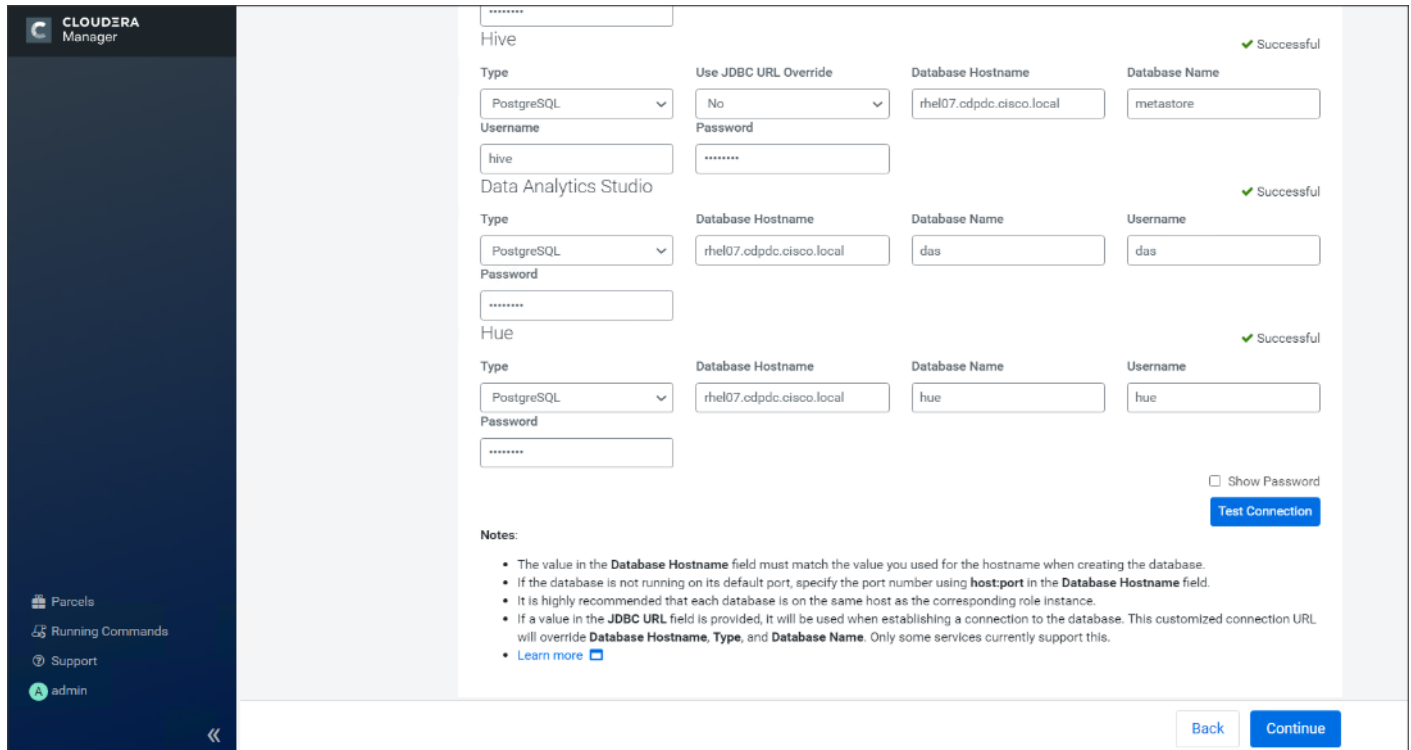


Set Up the Database

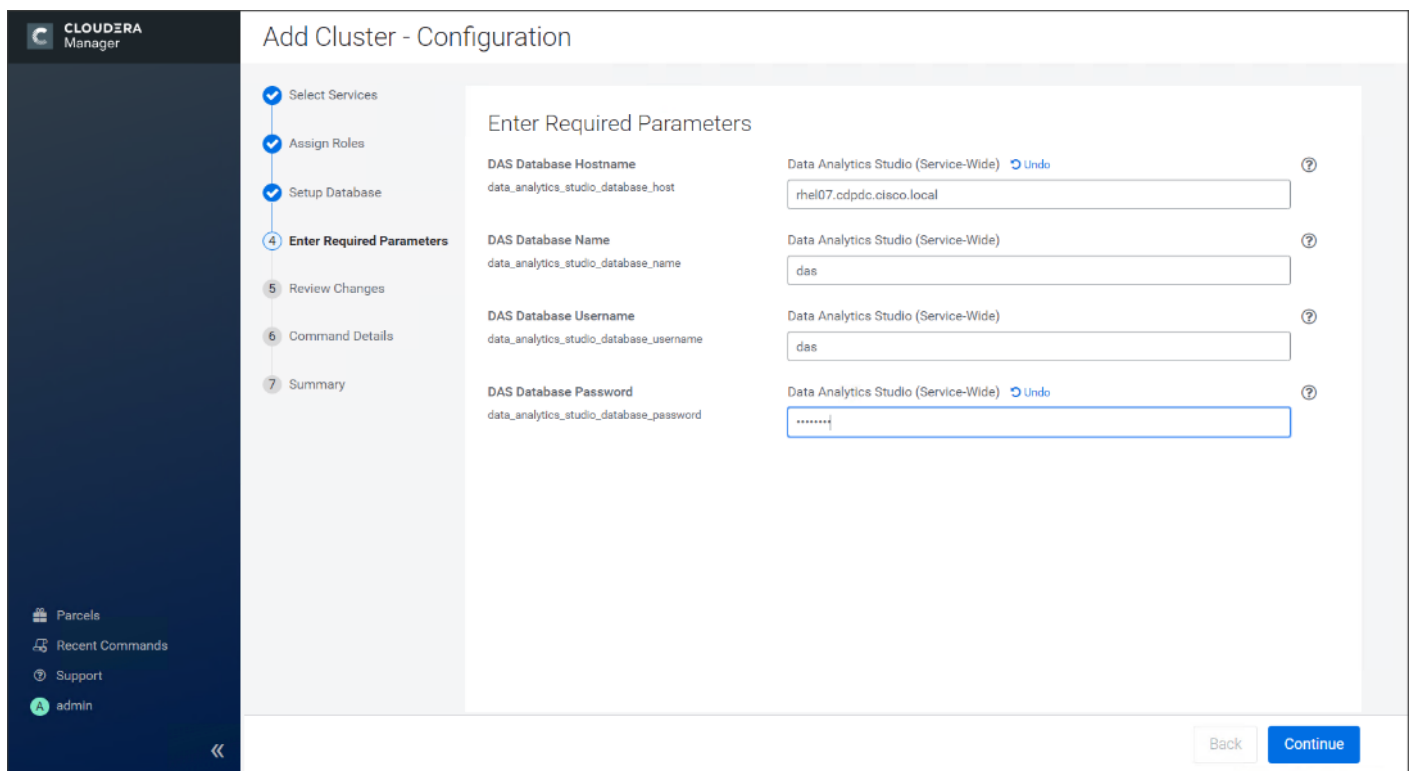
The role assignment recommendation above is for clusters of up to 64 servers. For clusters larger than 64 nodes, use the high availability recommendation defined in [Table 5](#).

To set up the database, follow these steps:

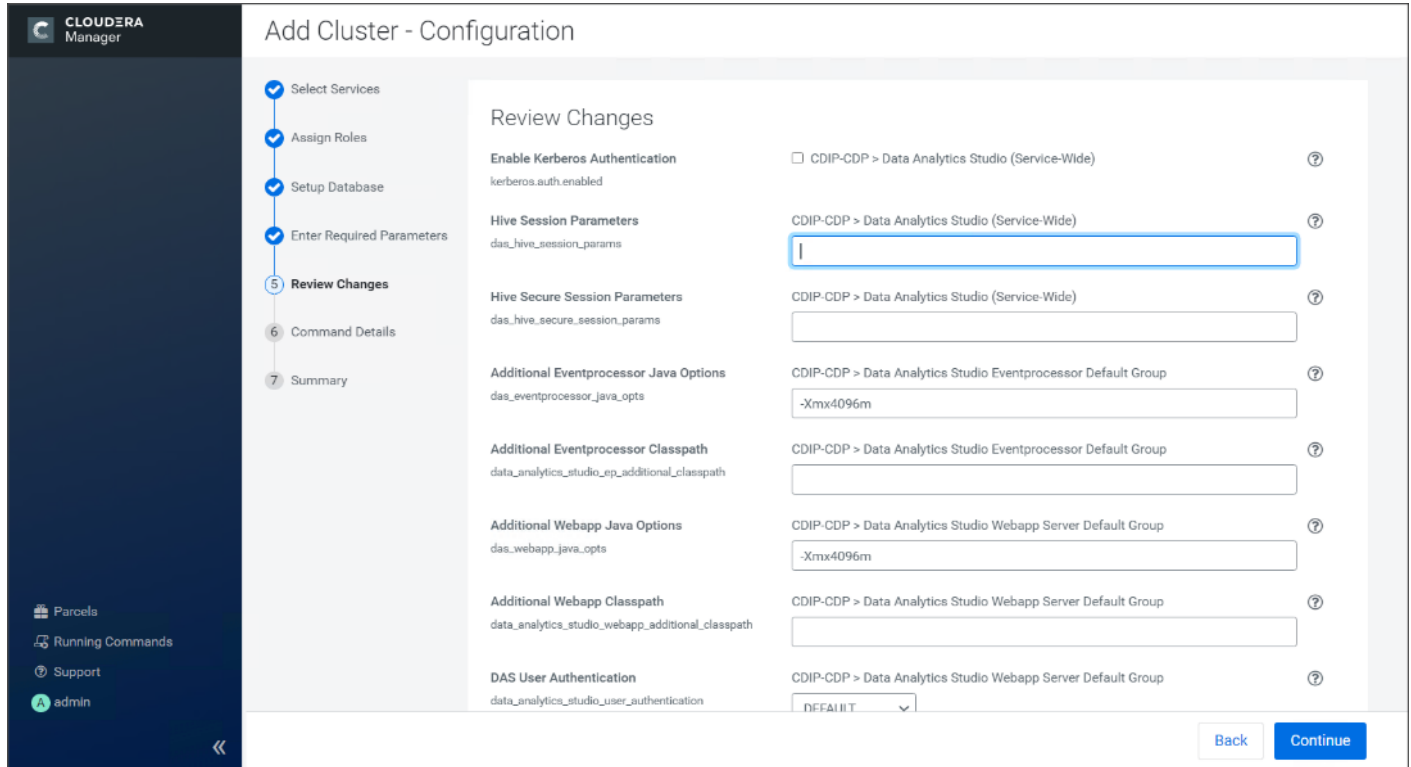
1. In the Database Host Name sections use port 3306 for TCP/IP because connection to the remote server always uses TCP/IP.
2. Enter the Database Name, username and password that were used during the database creation stage earlier in this document.
3. Click Test Connection to verify the connection and click Continue.



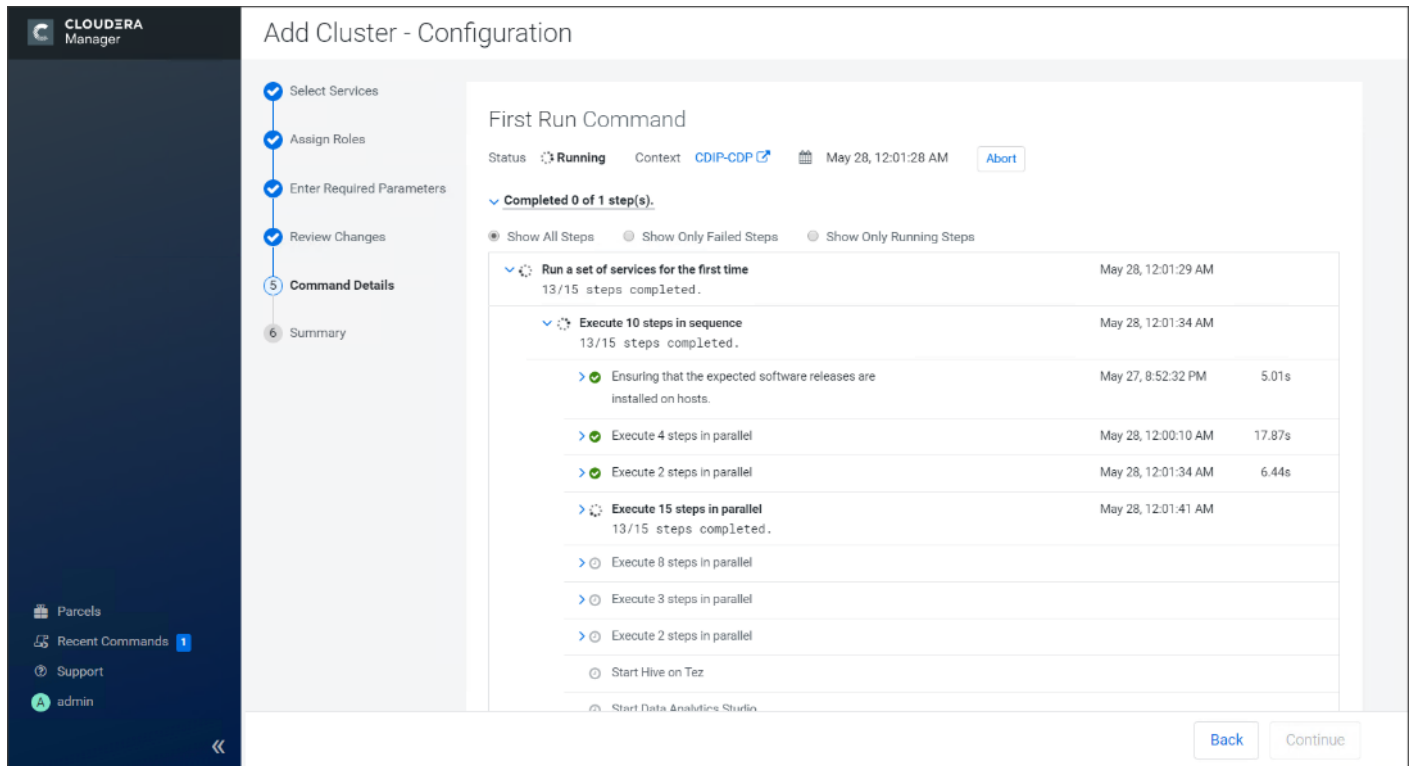
4. Enter required parameters for Data Analytics Studio.



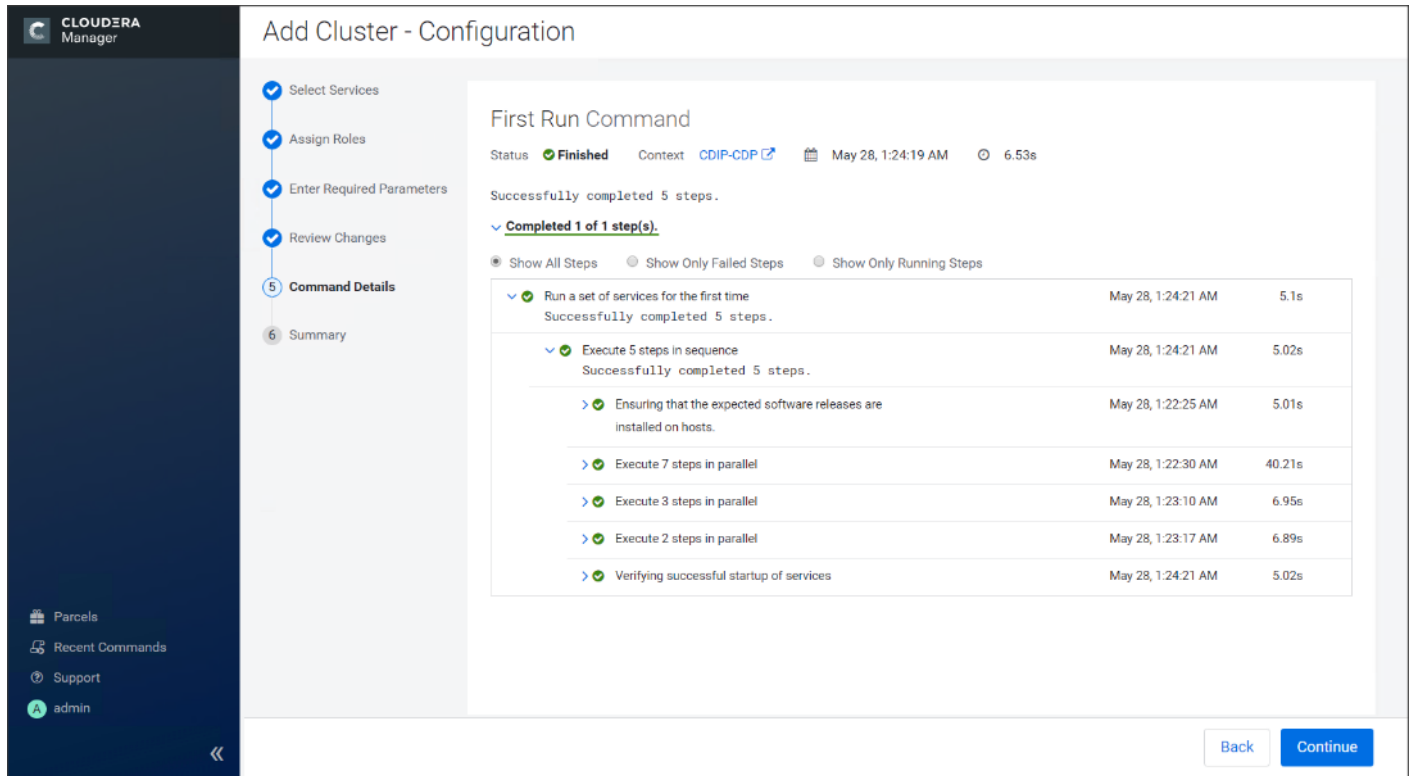
5. Review Data Analytics Studio (DAS) configuration.



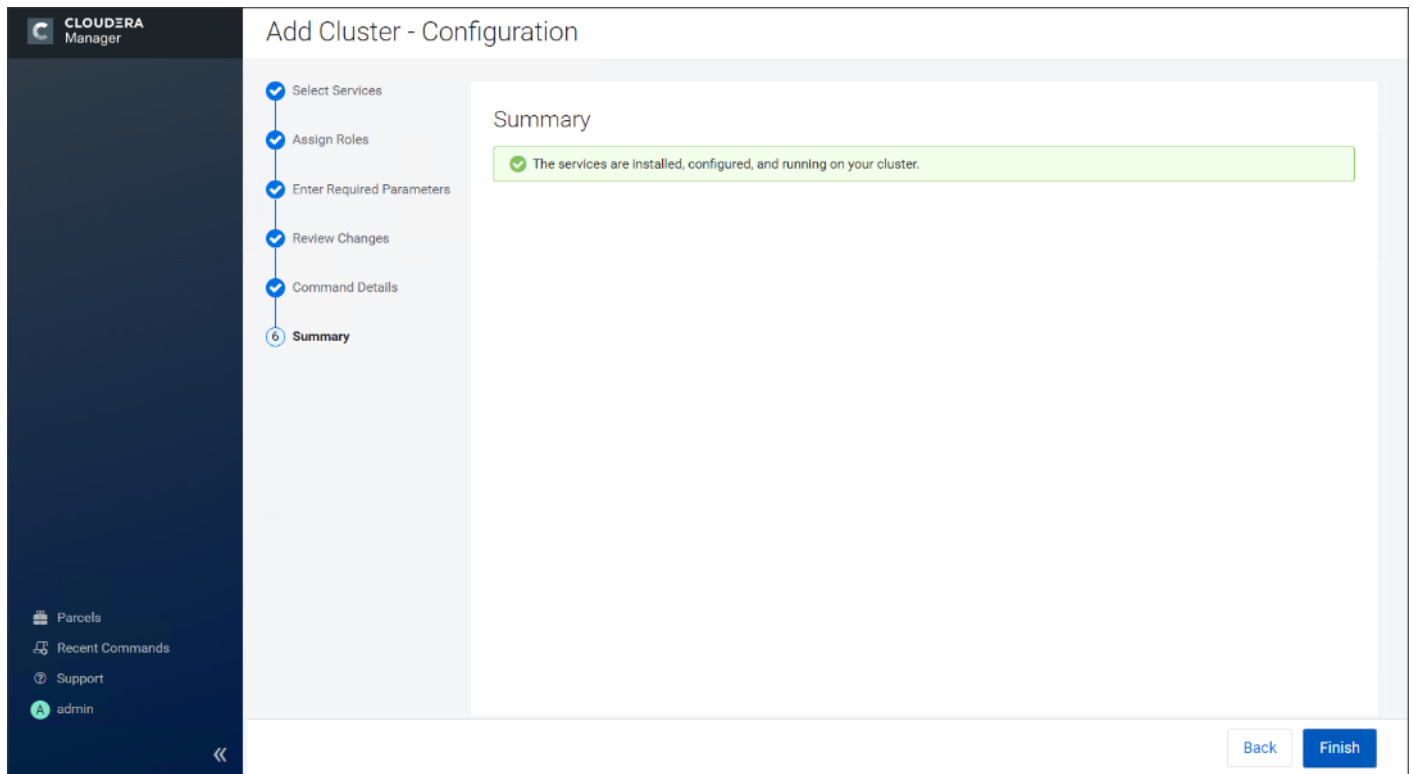
6. Review and customize the configuration changes based on your requirements.



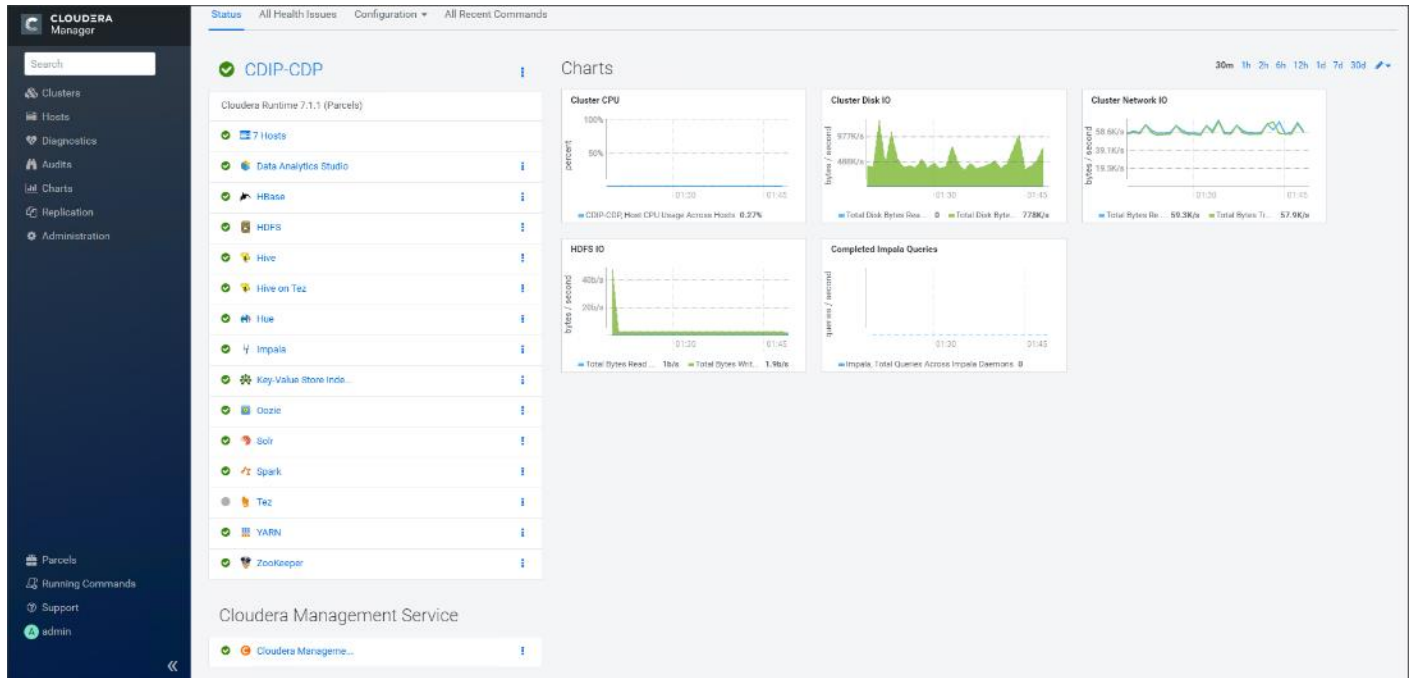
7. Click Continue to start running the cluster services.



8. Hadoop services are installed, configured, and now running on all the nodes of the cluster. Click Finish to complete the installation.



9. Cloudera Manager now displays the status of all Hadoop services running on the cluster.



Scale the Cluster

The role assignment recommendation above is for cluster with at least 64 servers and in High Availability. For smaller cluster running without High Availability the recommendation is to dedicate one server for NameNode and a second server for secondary name node and YARN Resource Manager. For larger clusters larger than 16 nodes the recommendation is to dedicate one server each for name node, YARN Resource Manager and one more for running both NameNode (High Availability) and Resource Manager (High Availability) as in the table (no Secondary NameNode when in High Availability).



For production clusters, it is recommended to set up NameNode and Resource manager in High Availability mode.

This implies that there will be at least 3 master nodes, running the NameNode, YARN Resource manager, the failover counter-part being designated to run on another node and a third node that would have similar capacity as the other two nodes.

All the three nodes will also need to run zookeeper and quorum journal node services. It is also recommended to have a minimum of 7 DataNodes in a cluster. Please refer to the next section for details on how to enable HA.

Rack-Aware Replica Placement

Data is stored in blocks on the HDFS filesystem. Data blocks are replicated, three times by default, for data durability and high availability. Block replicas are distributed across the cluster's racks and nodes using a rack placement policy. NameNode uses a rack-aware replica placement policy to improve data reliability, durability, and network bandwidth utilization. Using the rack-aware replica placement policy replicas of a block are placed on different DataNodes in the cluster on different racks so that failure of a single DataNode or rack does not make all replicas of a block unavailable. Block replicas on other DataNodes/racks are used in the event of failure of a DataNode or a rack. The rack-aware replica placement policy is implemented by the NameNode.

Using rack-aware replica placement improves data reliability, availability, and network utilization. To get maximum performance, it is important to configure rack-aware policy in CDP so that it knows the topology of your network. Network locations such as hosts and racks are represented in a tree, which reflects the network “distance” between locations. HDFS will use the network location to be able to place block replicas more intelligently to trade off performance and resilience. When placing jobs on hosts, CDP will prefer within-rack transfers (where there is more bandwidth available) to off-rack transfers; the MapReduce and YARN schedulers use network location to determine where the closest replica is as input to a map task. These computations are performed with the assistance of rack awareness scripts. Cloudera Manager includes internal rack awareness scripts, but you must specify the racks where the hosts in your cluster are located. If your cluster contains more than 10 hosts, Cloudera recommends that you specify the rack for each host. HDFS, MapReduce, and YARN will automatically use the racks you specify.

Customers can avoid complete data loss of data when a worker node goes down. No duplicated replicas are on the same node or nodes under the same rack. First replica is on the local rack or on nodes under the same rack.

Cloudera Manager supports nested rack specifications. For example, you could specify the rack `/rack3`, or `/group5/rack3` to indicate the third rack in the fifth group. All hosts in a cluster must have the same number of path components in their rack specifications.

To specify racks for hosts, follow these steps:

1. Click the Hosts tab.
2. Check the checkboxes next to the host(s) for a particular rack, such as all hosts for `/rack123`.
3. Click Actions for Selected (n) > Assign Rack, where n is the number of selected hosts.
4. Enter a rack name or ID that starts with a slash / such as `/rack123` or `/aisle1/rack123`, and then click Confirm.
5. Optionally restart affected services. Rack assignments are not automatically updated for running services.

To configure rack awareness on a Hadoop cluster, follow these steps:

1. Using Cloudera Manager, configure the following in safety valves:

In HDFS - Cluster-wide Advanced Configuration Snippet (Safety Valve) for `core-site.xml`:

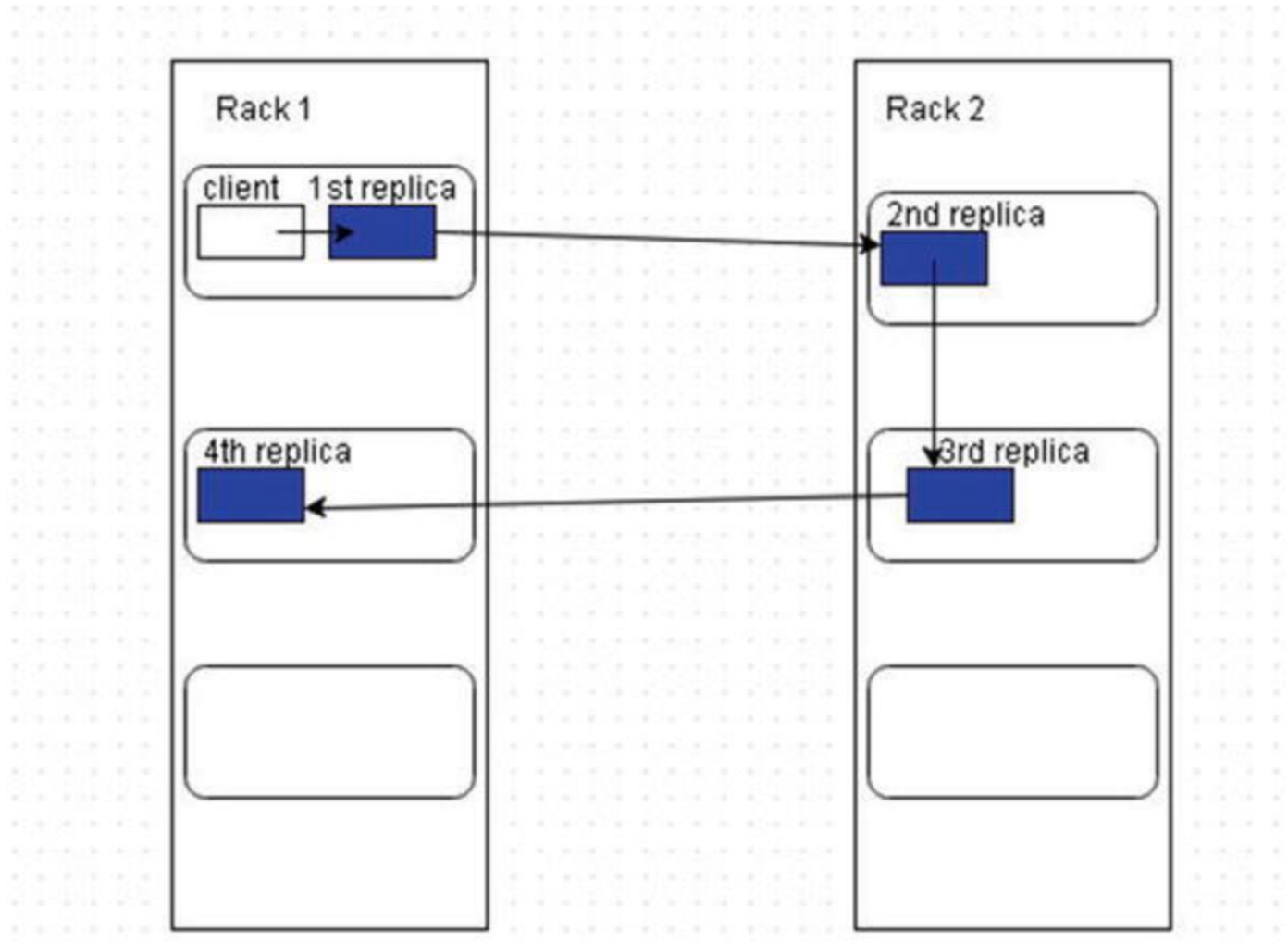
```
<property>
<name>net.topology.impl</name>
<value> org.apache.hadoop.net.NetworkTopologyWithNodeGroup</value>
</property>
<property>
<name>net.topology.nodegroup.aware</name>
<value> true</value>
</property>
<property>
<name>dfs.block.replicator.classname</name>
<value>org.apache.hadoop.hdfs.server.blockmanagement.BlockPlacementPolicyWithNodeGroup</value>
</property>
```

2. In YARN Service MapReduce Advanced Configuration Snippet (Safety Valve) `mapred.xml`, add the following properties and values:

```
<property>
<name>mapred.jobtracker.nodegroup.aware</name>
<value>>true</value>
</property>
<property>
<name>mapred.task.cache.levels</name>
<value>3</value>
</property>
```

To set rack location of hosts, follow these steps:

1. Select each node from the Hosts page and then assign a rack, following the format of /chassis\$ID/rack\$ID (shown below)



The screenshot shows the Cloudera Manager interface. On the left is a navigation sidebar with 'Hosts' selected. The main area is titled 'All Hosts' and contains a table of host roles. An 'Assign Rack' modal is open, listing actions for the selected 7 hosts. The table below shows the following data:

Roles	Commission State	Last Heartbeat	Load Average	Disk Usage	Physical Memory	Swap Space
> 30 Role(s)	Commissioned	4.87s	0.26 0.39 0.43	87.6 GB / 16.8 TiB	41.3 GB / 376.4 GiB	0 B / 1.9 GiB
> 8 Role(s)	Commissioned	12.25s	0.03 0.04 0.05	21.3 GiB / 444.0 GiB	6.2 GiB / 376.4 GiB	0 B / 1.9 GiB
> 17 Role(s)	Commissioned	9.72s	0.10 0.06 0.05	27.3 GiB / 16.8 TiB	14.3 GiB / 376.4 GiB	0 B / 1.9 GiB
> 8 Role(s)	Commissioned	8.97s	0.03 0.02 0.05	21 GiB / 76.8 TiB	6.2 GiB / 376.4 GiB	0 B / 1.9 GiB
> 16 Role(s)	Commissioned	307ms	0.09 0.06 0.05	27.3 GiB / 16.8 TiB	16.2 GiB / 376.4 GiB	0 B / 1.9 GiB
> 8 Role(s)	Commissioned	14.51s	0.12 0.05 0.09	20.6 GiB / 102.2 TiB	4.9 GiB / 219.6 GiB	0 B / 1.9 GiB
> 8 Role(s)	Commissioned	1.35s	0.04 0.03 0.05	22.4 GiB / 102.2 TiB	6.2 GiB / 376.4 GiB	0 B / 1.9 GiB

The 'Assign Rack' dialog box shows the following information:

Host	Current Rack
rhel[07-11, 15-16].cdpdc.cisco.local	/chassis1/rack1

Below the table, there is a 'New Rack' label and an empty text input field. Below the input field, there is explanatory text:

Rack names are slash-separated identifiers, like Unix paths. For example, "/rack1" and "/cabinet3/rack4" are both valid.

Changing the rack configuration might result in a transient state of mis-replicated blocks in HDFS until the old blocks are correctly placed using this new rack configuration.

At the bottom right, there are 'Cancel' and 'Confirm' buttons.



Follow the rack name format for each worker node in the Hadoop cluster.

Hadoop uses the rack information to place replica blocks on redundant racks.

After adding the safety valves and the rack names for each server, follow these steps:

1. Stop the cluster

2. Deploy client config
3. Start zookeeper
4. Start HDFS
5. Start all the other services.

Enable High Availability



Setting up High Availability is done after the Cloudera Installation is completed.

HDFS High Availability

The HDFS High Availability feature provides the option of running two NameNodes in the same cluster, in an Active/Passive configuration. These are referred to as the Active NameNode and the Standby NameNode. Unlike the Secondary NameNode, the Standby NameNode is a hot standby, allowing a fast failover to a new NameNode in case that a machine crashes, or a graceful administrator-initiated failover for the purpose of planned maintenance. There cannot be more than two NameNodes.

For more information go to: <https://docs.cloudera.com/content/www/en-us/documentation/enterprise/6/6.3/PDF/cloudera-administration.pdf>

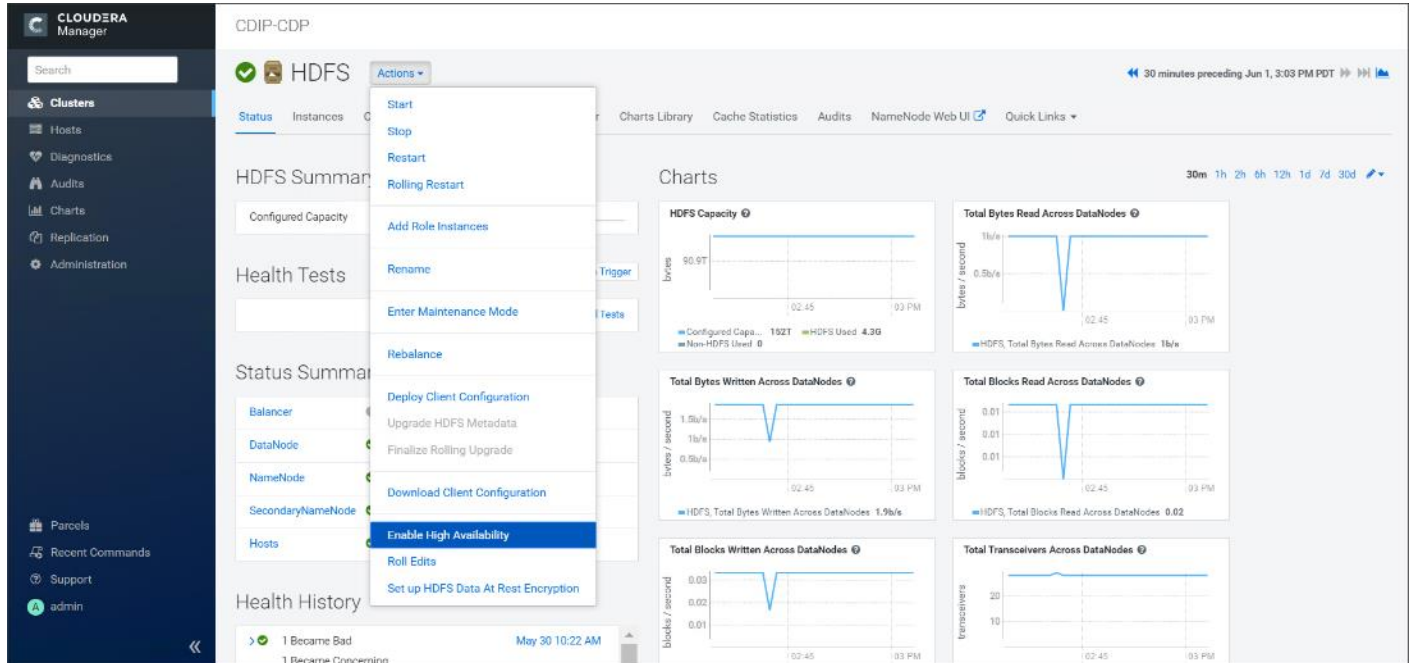
Set Up HDFS High Availability

The Enable High Availability workflow leads through adding a second (standby) NameNode and configuring JournalNodes. During the workflow, Cloudera Manager creates a federated namespace. To set up HDFS High Availability, follow these steps:

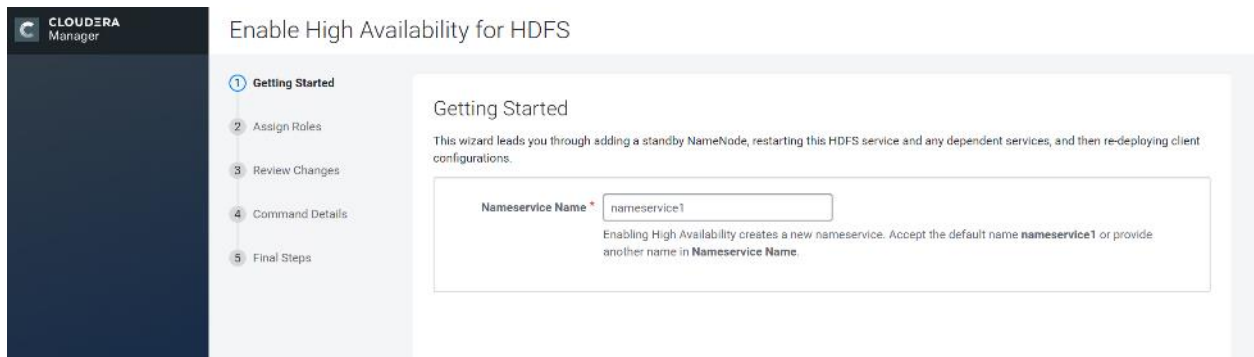
1. Log into the admin node (rhelnn01) and create the Edit directory for the JournalNode:

```
# ansible namenodes -m shell -a "mkdir -p /data/disk1/namenode-edits"  
# ansible namenodes -m shell -a "chmod 777 /data/disk1/namenode-edits"
```

2. Log into the Cloudera manager and go to the HDFS service.
3. Select Actions> Enable High Availability. A screen showing the hosts that are eligible to run a standby NameNode and the JournalNodes displays.



- Specify a name for the nameservice or accept the default name nameservice1 and click Continue.



- In the NameNode Hosts field, click Select a host. The host selection dialog displays.
- Check the checkbox next to the hosts (rhel2) where the standby NameNode is to be set up and click OK.
- In the JournalNode Hosts field, click Select hosts. The host selection dialog displays.
- Check the checkboxes next to an odd number of hosts (a minimum of three) to act as JournalNodes and click OK. We used the same nodes for the Zookeeper nodes.
- Click Continue.



The standby NameNode cannot be on the same host as the active NameNode, and the host that is chosen should have the same hardware configuration (RAM, disk space, number of cores, and so on) as the active NameNode.

Enable High Availability for HDFS

Assign Roles

NameNode Hosts:

JournalNode Hosts:

We recommend that JournalNodes be hosted on machines of similar hardware specifications as the NameNodes. The hosts of NameNodes and the ResourceManager are generally good options. You must have a minimum of three and an odd number of JournalNodes.

Back Continue

10. In the JournalNode Edits Directory property, enter a directory location created earlier in step 1 for the JournalNode edits directory into the fields for each JournalNode host.

Enable High Availability for HDFS

Review Changes

Set the following configuration values for your new role(s). Required values are marked with *.

Parameter	Group	Value	Description
Service HDFS			
NameNode Data Directories* dfs.namenode.name.dir	rhel07	/data/disk1/dfs/nn Inherited from: NameNode Default Group	Determines where on the local file system the NameNode should store the name table (fsimage). For redundancy, enter a comma-delimited list of directories to replicate the name table in all of the directories. Typical values are /data/N/dfs/nn where N=1..3.
	rhel09	/data/disk1/dfs/nn Inherited from: NameNode Default Group	
JournalNode Edits Directory* dfs.journalnode.edits.dir	rhel07	/data/disk1/namenode-e Reset to empty default value	Directory on the local file system where NameNode edits are written.
	rhel09	/data/disk1/namenode-e Reset to empty default value	
	rhel11	/data/disk1/namenode-e Reset to empty default value	

Extra Options

- Force initialize the ZooKeeper ZNode for autofalover. Any previous ZNode used for this nameservice will be overwritten.
- Clear any existing data present in name directories of Standby NameNode.

Back Continue



The directories specified should be empty and must have the appropriate permissions.

11. Extra Options: Decide whether Cloudera Manager should clear existing data in ZooKeeper, Standby NameNode, and JournalNodes. If the directories are not empty (for example, re-enabling a previous HA configuration), Cloudera Manager will not automatically delete the contents—select to delete the contents by keeping the default checkbox selection. The recommended default is to clear the directories.



If you choose not to configure any of the extra options, the data should be in sync across the edits directories of the JournalNodes and should have the same version data as the NameNodes.

12. Click Continue.
13. Cloudera Manager executes a set of commands that will stop the dependent services, delete, create, and configure roles and directories as appropriate, create a nameservice and failover controller, and restart the dependent services and deploy the new client configuration.

Enable High Availability for HDFS

Enable High Availability Command

Status ✔ **Finished** Context [HDFS](#) Jun 24, 1:02:53 PM 8.6m

Successfully enabled High Availability and Automatic Failover

Completed 20 of 20 step(s).

Show All Steps Show Only Failed Steps Show Only Running Steps

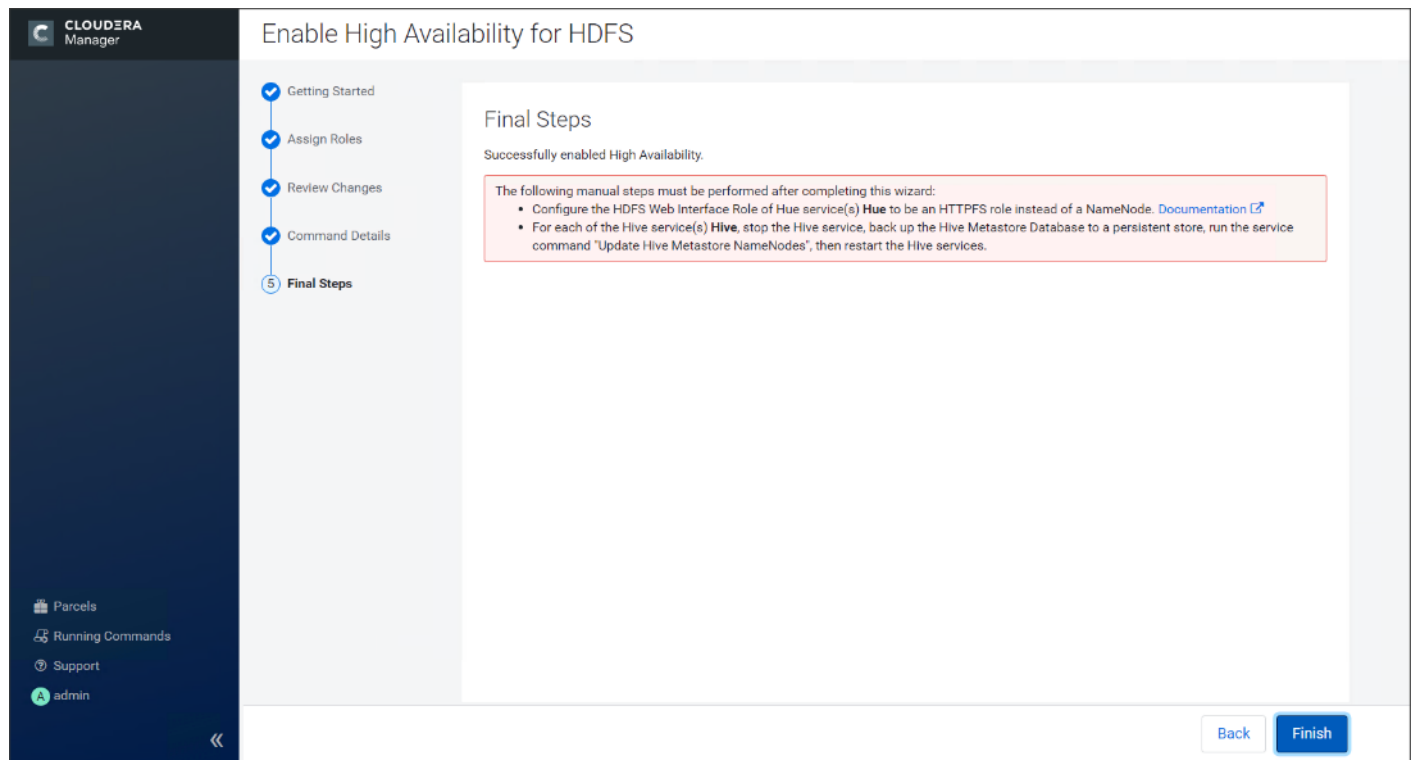
✔ Check that name directories for the new Standby NameNode either do not exist or are writable and empty. Can optionally clear directories.	rhel07.odpdc.cisco.local	Jun 24, 1:02:53 PM	2.71s
✔ Check that edits directories for the nameservice either do not exist or are writable and empty. Can optionally clear directories.		Jun 24, 1:02:55 PM	2.69s
✔ Stop hdfs and its dependent services	CDIP-CDP	Jun 24, 1:02:58 PM	2.5m
✔ Creating roles to enable High Availability		Jun 24, 1:05:27 PM	38ms
✔ Deleting the SecondaryNameNode role. The checkpoint directories of the SecondaryNameNode will not be deleted.		Jun 24, 1:05:27 PM	62ms
✔ Configuring NameNodes and the HDFS service to enable High Availability.		Jun 24, 1:05:27 PM	3ms
✔ Initializing High Availability state in ZooKeeper.	Failover Controller (rhel09)	Jun 24, 1:05:27 PM	18.72s
✔ Starting the JournalNodes		Jun 24, 1:05:46 PM	23.84s

[Back](#) [Continue](#)



Formatting the name directory is expected to fail if the directories are not empty.

14. In the next screen additional steps are suggested by the Cloudera Manager to update the Hue and Hive metastore. Click Finish.

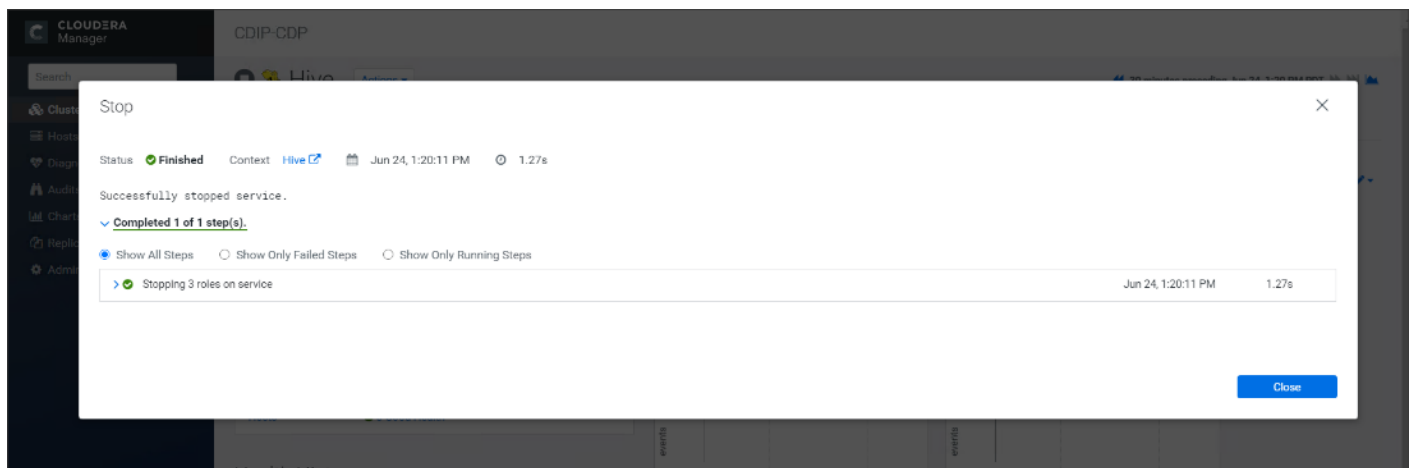
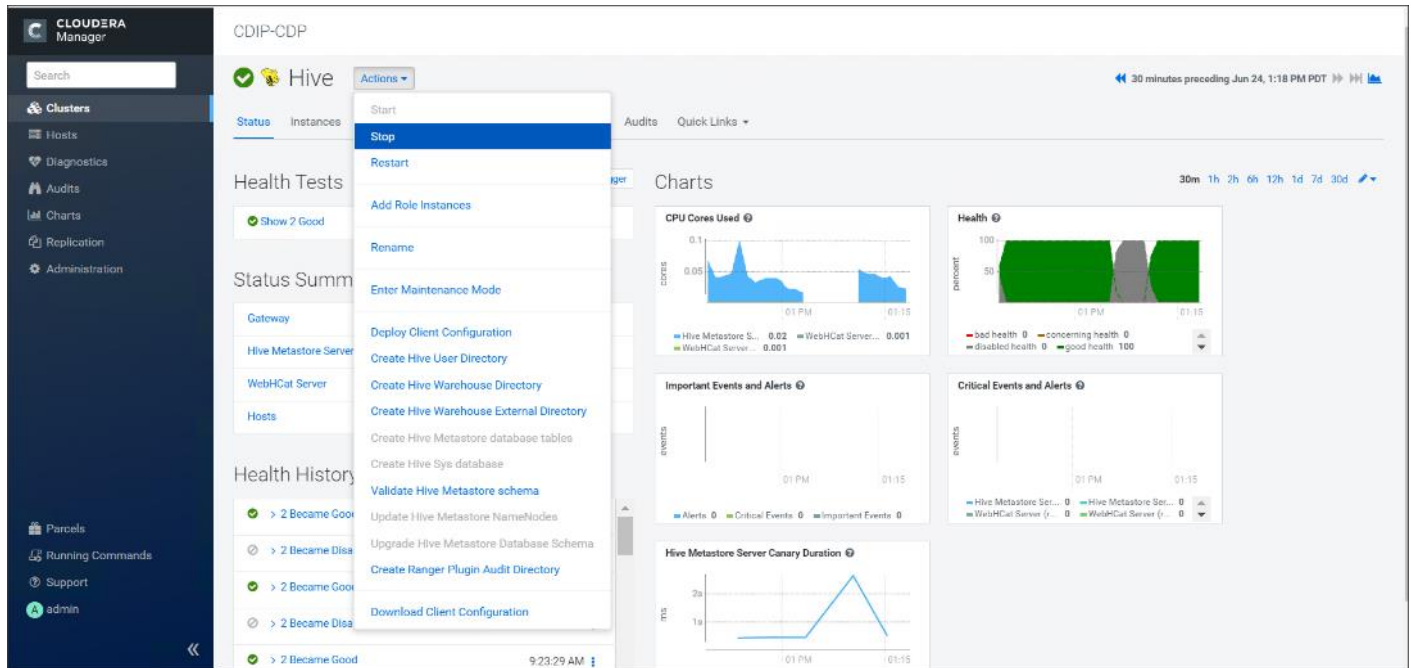


The following subsections explain configuring Hue and Hive for High Availability as needed.

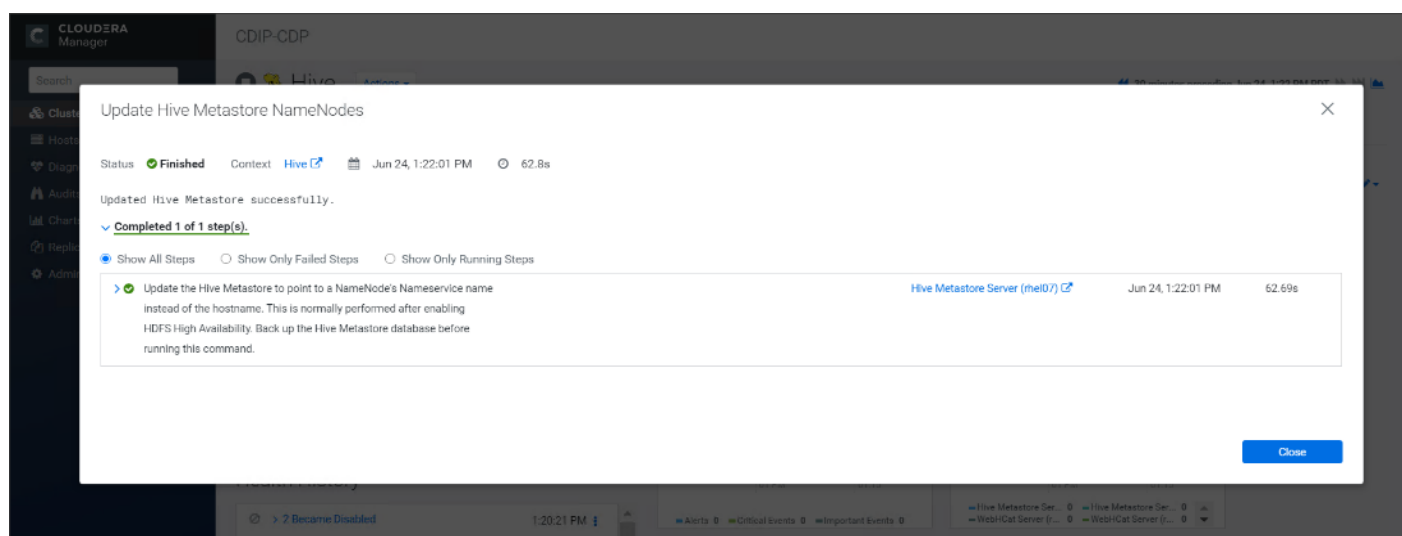
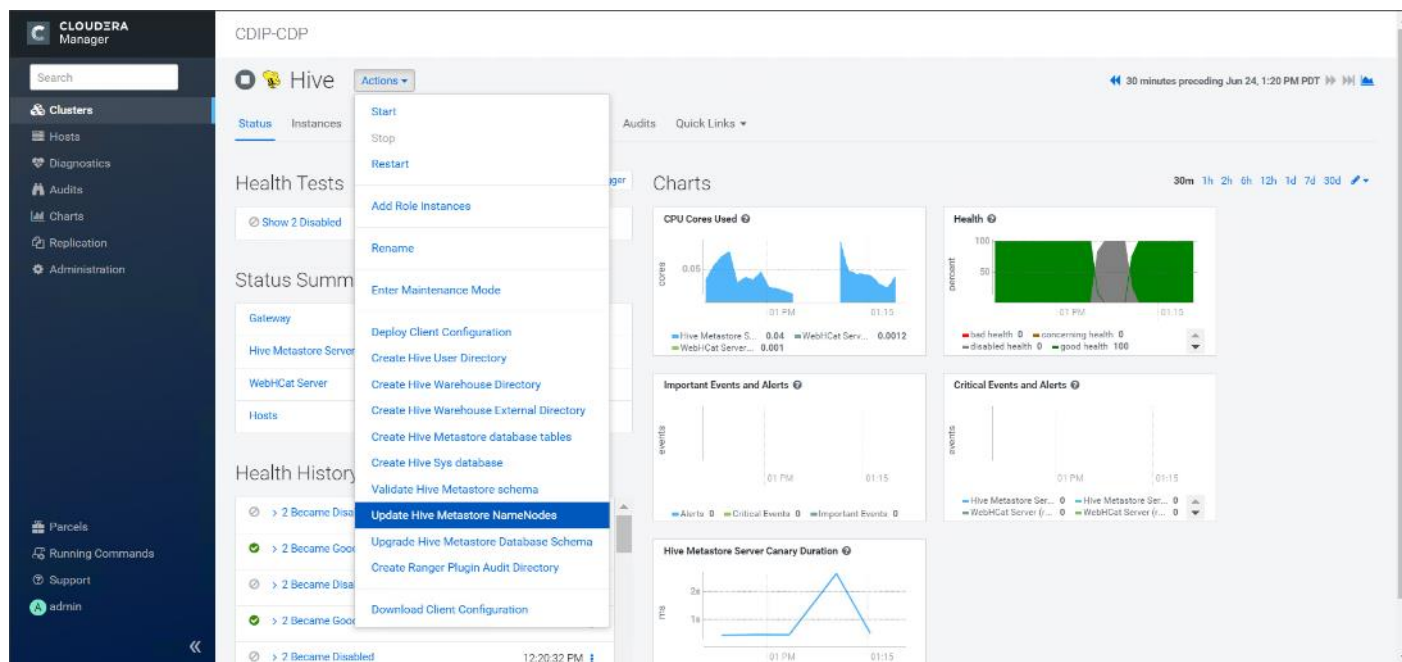
Configure Hive Metastore to Use HDFS High Availability

To configure the Hive Megastore to use HDFS High Availability, follow these steps:

1. Go the Hive service.
2. Select Actions > Stop.



3. Click Stop to confirm the command.
4. Back up the Hive Metastore Database (if any existing data is present).
5. Select Actions> Update Hive Metastore NameNodes and confirm the command.

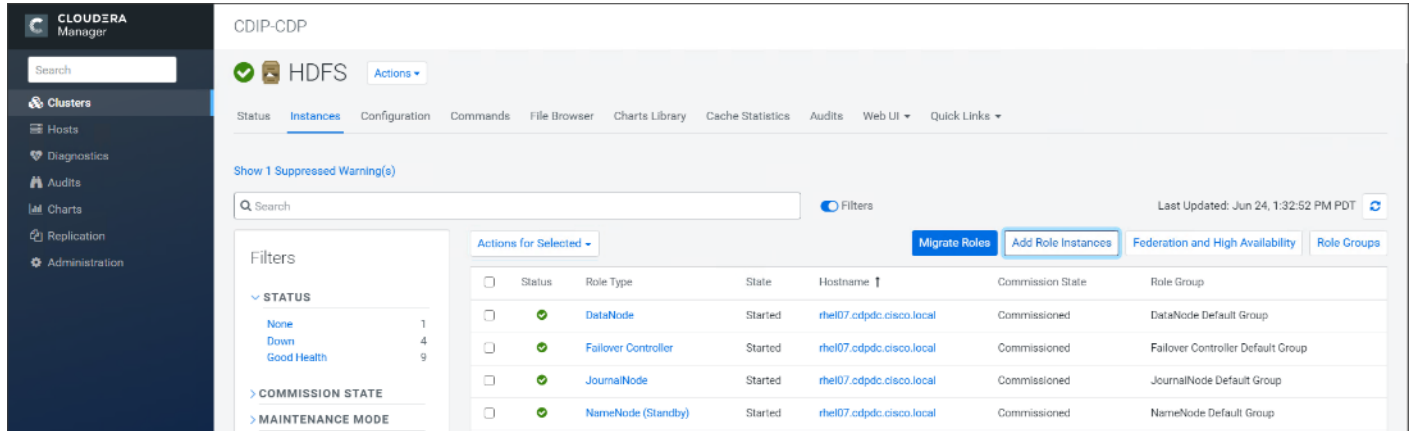


6. Select Actions> Start.
7. Restart the Hue and Impala services if stopped prior to updating the Metastore.

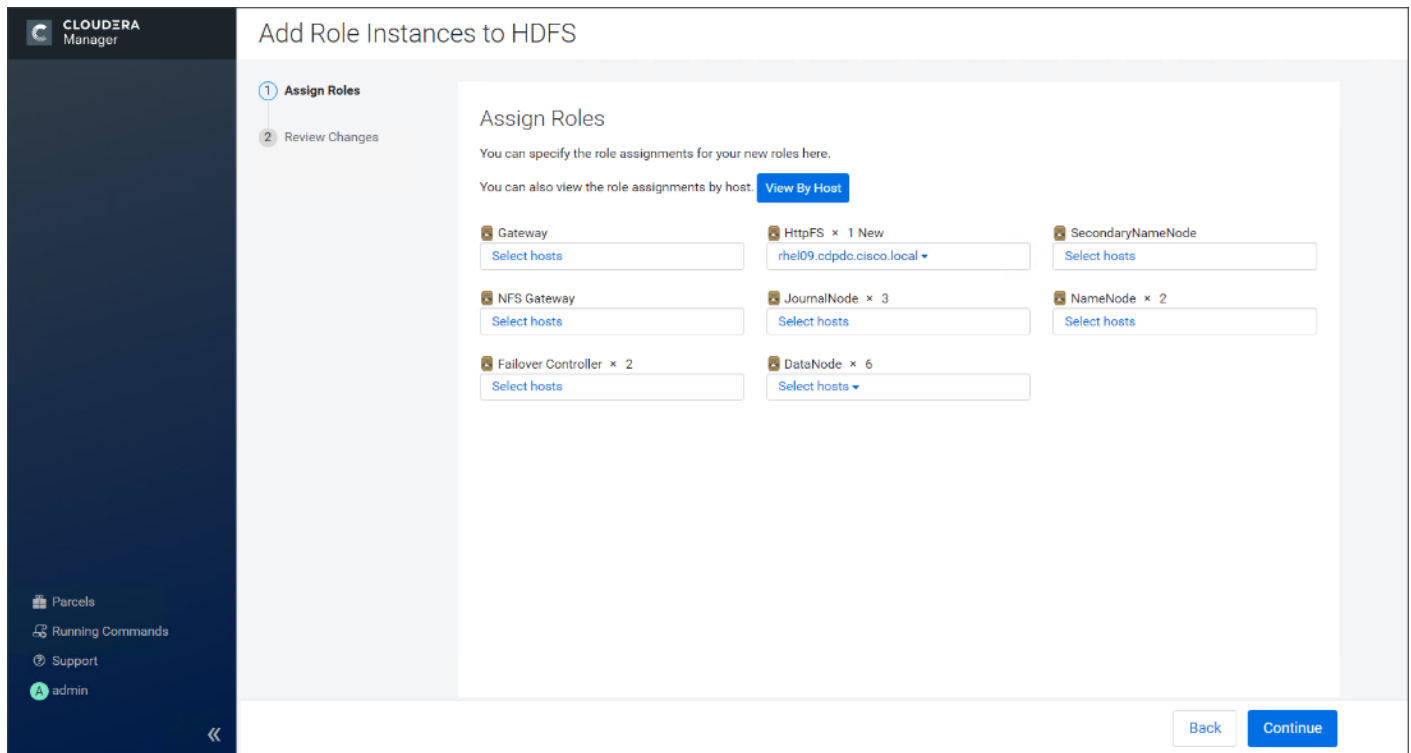
Configure Hue to Work with HDFS High Availability

To configure Hue to work with HDFS High Availability, follow these steps:

1. Go to the HDFS service.
2. Click the Instances tab.
3. Click Add Role Instances.



4. Select the text box below the HttpFS field. The Select Hosts dialog displays.
5. Select the host on which to run the role and click OK.
6. Click Continue.
7. Check the checkbox next to the HttpFS role and select Actions for Selected> Start.



CDIP-CDP

HDFS

Status **Instances** Configuration Commands File Browser Charts Library Cache Statistics Audits Web UI Quick Links

Show 1 Suppressed Warning(s)

Search Filters Last Updated: Jun 24, 1:38:39 PM PDT

Actions for Selected (1) Migrate Roles Add Role Instances Federation and High Availability Role Groups

Status	Role Type	State	Hostname	Commission State	Role Group
<input checked="" type="checkbox"/>	HttpFS	Stopped	rhel09.cdpc.dcsco.local	Commissioned	HttpFS Default Group

1 - 1 of 1

Filters

- STATUS
 - None: 0
 - Down: 0
 - Good Health: 0
 - Stopped: 1
- COMMISSION STATE
- MAINTENANCE MODE

8. After the command has completed, go to the Hue service.

9. Click the Configuration tab.

10. Locate the HDFS Web Interface Role property or search for it by typing its name in the Search box.

11. Select the HttpFS role that was just created instead of the NameNode role and save your changes.

12. Restart the Hue service.

CDIP-CDP

Hue

Status Instances **Configuration** Commands Charts Library Audits Web UI Quick Links

Search Filters Role Groups History and Rollback

Filters

- SCOPE
 - Hue (Service-Wide): 68
 - Hue Server: 48
 - Kerberos Ticket Renewer: 30
 - Load Balancer: 36
- CATEGORY
 - Advanced: 24
 - Cloudera Navigator Optimizer: 4
 - Database: 8
 - Logs: 3
 - Main: 25
 - Monitoring: 31
 - Performance: 1
 - Ports and Addresses: 3
 - Resource Management: 5
 - Security: 29
- STATUS
 - Error: 0

HDFS Web Interface Role: webhdfs.url

- Hue (Service-Wide) Undo
 - HttpFS (rhel09)
 - NameNode (rhel07)
 - NameNode (rhel09)

HTTPFS role is recommended for Web interface if HDFS is HA or federated. Suppress...

Hdfs Service: Hue (Service-Wide) ?

- HDFS

Oozie Service: Hue (Service-Wide) ?

- Oozie

HBase Service: Hue (Service-Wide) ?

- HBase

HiveServer2 Service: Hue (Service-Wide) ?

- Hive on Tez

1 Edited Value Reason for change: Modified HDFS Web Interface Role Save Changes (CTRL+S)



Refer to the High Availability section in the Cloudera Management document:

https://www.cloudera.com/documentation/enterprise/6/6.2/topics/admin_ha.html for more information on setting up High Availability for other components like Impala, Oozie, and so on.

YARN High Availability

The YARN Resource Manager (RM) is responsible for tracking the resources in a cluster and scheduling applications (for example, MapReduce jobs). Before CDH 5, the RM was a single point of failure in a YARN cluster. The RM high availability (HA) feature adds redundancy in the form of an Active/Standby RM pair to remove this single point of failure. Furthermore, upon failover from the Standby RM to the Active, the applications can resume from their last check-pointed state; for example, completed map tasks in a MapReduce job are not re-run on a subsequent attempt. This allows events such the following to be handled without any significant performance effect on running applications.

- Unplanned events such as machine crashes.
- Planned maintenance events such as software or hardware upgrades on the machine running the ResourceManager.

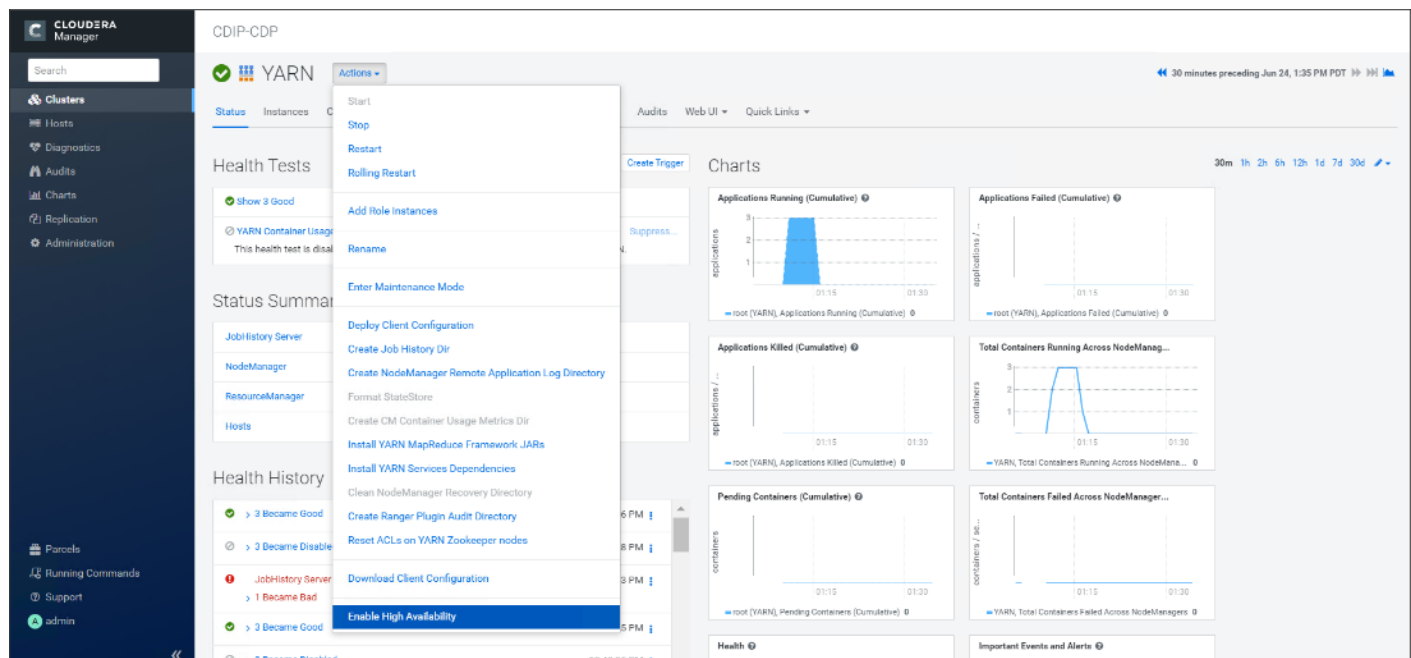
For more information, go to:

https://www.cloudera.com/documentation/enterprise/latest/topics/cdh_hag_rm_ha_config.html#xd_583c10bfdbd326ba--43d5fd93-1410993f8c2--7f77

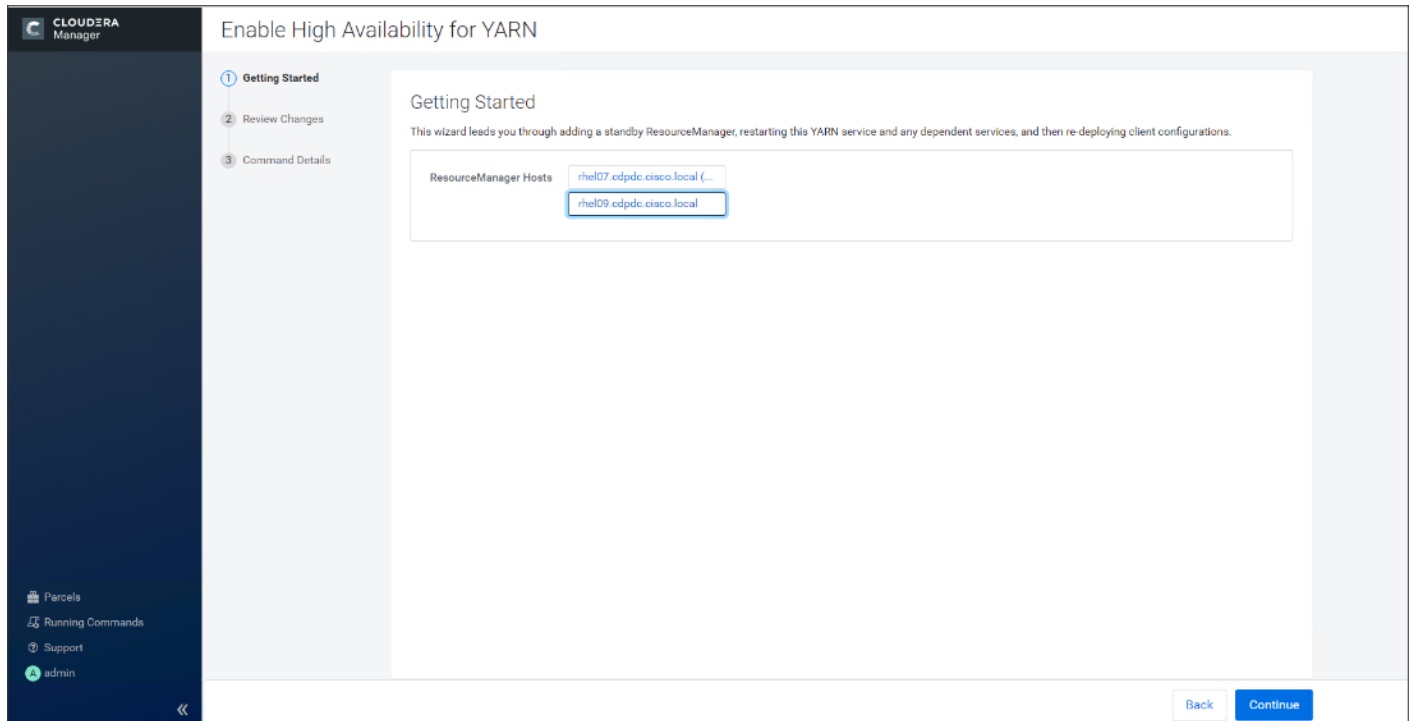
Set Up YARN High Availability

To set up YARN high availability, follow these steps:

1. Log into the Cloudera manager and go to the YARN service.
2. Select Actions> Enable High Availability.



3. A screen showing the hosts that are eligible to run a standby ResourceManager displays.
4. The host where the current ResourceManager is running is not available as a choice.
5. Select the host (rhel3) where the standby ResourceManager is to be installed and click Continue.



6. Cloudera Manager proceeds to execute a set of commands that stop the YARN service, add a standby ResourceManager, initialize the ResourceManager high availability state in ZooKeeper, restart YARN, and re-deploy the relevant client configurations.
7. Click Finish once the installation is completed successfully.

Configure Yarn (MR2 Included) and HDFS Services

The parameters in [Table 6](#) and [Table 7](#) are used for Cisco UCS Integrated Infrastructure for Big Data and Analytics Performance Optimized cluster configuration described in this document. These parameters are to be changed based on the cluster configuration, number of nodes and specific workload.

Table 6 YARN

Service	Value
mapreduce.output.fileoutputformat.compress.type	BLOCK
mapreduce.output.fileoutputformat.compress.codec	org.apache.hadoop.io.compress.DefaultCodec
mapreduce.map.output.compress.codec	org.apache.hadoop.io.compress.SnappyCodec
mapreduce.map.output.compress	True
zlib.compress.level	BEST_SPEED
mapreduce.task.io.sort.factor	64
mapreduce.map.sort.spill.percent	0.9
mapreduce.reduce.shuffle.parallelcopies	20
yarn.nodemanager.resource.memory-mb	320GB

Service	Value
yarn.nodemanager.resource.cpu-vcores	64
yarn.scheduler.maximum-allocation-vcores	64
yarn.scheduler.maximum-allocation-mb	320GB
mapreduce.task.io.sort.mb	2047
mapreduce.job.reduce.slowstart.completedmap	0.8
yarn.app.mapreduce.am.resource.cpu-vcores	1
mapreduce.map.memory.mb	5G
mapreduce.reduce.memory.mb	5G
mapreduce.job.heap.memory-mb.ratio	0.8
mapreduce.job.shuffle.merge.percent	0.95
mapreduce.job.shuffle.input.buffer.percent	0.7
mapreduce.job.reduce.input.buffer.percent	0.7
mapreduce.input.fileinputformat.split.minsize	4096000000
mapreduce.ifile.readahead.bytes	16777216
mapreduce.inmem.merge.threshold	0
Enable Optimized Map-side Output Collector	Enable – Gateway Default Group

Table 7 HDFS

Service	Value
dfs.datanode.failed.volumes.tolerated	6
dfs.datanode.du.reserved	50 GiB
dfs.datanode.data.dir.perm	755
Java Heap Size of Namenode in Bytes	4096 MiB
dfs.namenode.handler.count	55
dfs.namenode.service.handler.count	55
Java Heap Size of Secondary namenode in Bytes	4096 MiB

Configure Spark

The two main resources that Spark (and YARN) are dependent on are CPU and memory. Disk and network I/O play a part in Spark performance as well, but neither Spark nor YARN currently can actively manage them. Every Spark executor in any application has the same fixed number of cores and same fixed heap size. The number of cores can be specified with the `executor-cores` flag when invoking `spark-submit`, `spark-shell`, and `pyspark` from

the command line, or by setting the `spark.executor.cores` property in the `spark-defaults.conf` file or in the `SparkConf` object.

The heap size can be controlled with the `executor-memory` flag or the `spark.executor.memory` property. The `cores` property controls the number of concurrent tasks an executor can run, `executor-cores = 5` mean that each executor can run a maximum of five tasks at the same time. The memory property impacts the amount of data Spark can cache, as well as the maximum sizes of the shuffle data structures used for grouping, aggregations, and joins.

The `num-executors` command-line flag or `spark.executor.instances` configuration property control the number of executors requested. Dynamic Allocation can be enabled from CDH5.4 instead setting the `spark.dynamicAllocation.enabled` to `true`. Dynamic allocation enables a Spark application to request executors when there is a backlog of pending tasks and free up executors when idle.

Asking for five executor cores will result in a request to YARN for five virtual cores. The memory requested from YARN is a little more complex for a couple reasons:

- `executor-memory/spark.executor.memory` controls the executor heap size, but JVMs can also use some memory off heap, for example for VM overhead, interned Strings and direct byte buffers. The value of the `spark.yarn.executor.memoryOverhead` property is added to the executor memory to determine the full memory request to YARN for each executor. It defaults to `max (384, 0.10 * spark.executor.memory)`.
- YARN may round the requested memory up a little. YARN's `yarn.scheduler.minimum-allocation-mb` and `yarn.scheduler.increment-allocation-mb` properties control the minimum and increment request values respectively.
- The application master is a non-executor container with the special capability of requesting containers from YARN, takes up resources of its own that must be budgeted in. In *yarn-client* mode, it defaults to a 1024MB and one vcore. In *yarn-cluster* mode, the application master runs the driver, so it's often useful to add its resources with the `-driver-memory` and `-driver-cores` properties.
- Running executors with too much memory often results in excessive garbage collection delays. 64GB is a rough guess at a good upper limit for a single executor.
- A good estimate is that at most five tasks per executor can achieve full write throughput, so it's good to keep the number of cores per executor around that number.
- Running tiny executors (with a single core and just enough memory needed to run a single task, for example) throws away the benefits that come from running multiple tasks in a single JVM. For example, broadcast variables need to be replicated once on each executor, so many small executors will result in many more copies of the data.

Tune Resource Allocation for Spark

Below is an example of configuring a Spark application to use as much of the cluster as possible, we are using an example cluster with 16 nodes running NodeManagers, each equipped with 56 cores and 256GB of memory. `yarn.nodemanager.resource.memory-mb` and `yarn.nodemanager.resource.cpu-vcores` should be set to $180 * 1024 = 184320$ (megabytes) and 48 respectively.

```
spark.default.parallelism=10000
spark.driver.memoryOverhead=4096
spark.executor.memoryOverhead=4096
spark.executor.extraJavaOptions=-XX:+UseParallelGC -XX:ParallelGCThreads=4
```

```
spark.shuffle.file.buffer=1024k
spark.broadcast.compress=true
spark.shuffle.compress=true
spark.io.compression.codec=org.apache.spark.io.SnappyCompressionCodec
spark.io.compression.snappy.blockSize=512k
```

This configuration results in four executors on all nodes except for the one with the AM, which will have three executors.

```
executor-memory is derived as (180/4 executors per node) = 45; 45 * 0.10 = 4.5 45 -
4.5 ~ 40.
For taking care of long running processes use 2G for the spark driver
spark.driver.memory = 2G
```

Submit a Job

```
--driver -memory 2G -executor -memory 40G --num-executors 63 --executor-cores 5 --
properties-file /opt/cloudera/parcels/CDH/etc/spark/conf.dist/spark-defaults.conf
```

In yarn-cluster mode, the local directories used by the Spark executors and the Spark driver will be the local directories configured for YARN (Hadoop YARN config yarn.nodemanager.local-dirs). If the user specifies spark.local.dir, it will be ignored.

In yarn-client mode, the Spark executors will use the local directories configured for YARN while the Spark driver will use those defined in spark.local.dir. The Spark driver does not run on the YARN cluster in yarn-client mode, only the Spark executors do.

```
spark.local.dir /tmp (Directory to use for "scratch" space in Spark, including map
output files and RDDs that get stored on disk. This should be on a fast, local disk
in your system).
```

Every Spark stage has several tasks, each of which processes data sequentially. In tuning Spark jobs, this parallelism number is the most important parameter in determining performance. The number of tasks in a stage is the same as the number of partitions in the last RDD in the stage. The number of partitions in an RDD is the same as the number of partitions in the RDD on which it depends, with a couple exceptions: the coalesce transformation allows creating an RDD with fewer partitions than its parent RDD, the union transformation creates an RDD with the sum of its parents' number of partitions, and Cartesian creates an RDD with their product.

```
RDDs produced by a file have their partitions determined by the underlying MapReduce
InputFormat that's used. Typically there will be a partition for each HDFS block
being read. Partitions for RDDs produced by parallelize come from the parameter
given by the user, or spark.default.parallelism if none is given.
```

The primary concern is that the number of tasks will be too small. If there are fewer tasks than slots available to run them in, the stage won't be taking advantage of all the CPU available.

If the stage in question is reading from Hadoop, your options are:

- Use the repartition transformation, which will trigger a shuffle.
- Configure your InputFormat to create more splits.
- Write the input data out to HDFS with a smaller block size.

If the stage is getting its input from another stage, the transformation that triggered the stage boundary will accept a numPartitions argument.

The most straightforward way to tune the number of partitions is experimentation: Look at the number of partitions in the parent RDD and then keep multiplying that by 1.5 until performance stops improving.

In contrast with MapReduce for Spark when in doubt, it is almost always better to be on the side of a larger number of tasks (and thus partitions).

Shuffle Performance Improvement

`spark.shuffle.compress true` (compress map output files)

`spark.broadcast.compress true` (compress broadcast variables before sending them)

`spark.io.compression.codec org.apache.spark.io.SnappyCompressionCodec` (codec used to compress internal data such as RDD partitions, broadcast variables and shuffle outputs)

`spark.shuffle.spill.compress true` (Whether to compress data spilled during shuffles.)

`spark.shuffle.io.numConnectionsPerPeer 4` (Connections between hosts are reused in order to reduce connection buildup for large clusters. For clusters with many hard disks and few hosts, this may result in insufficient concurrency to saturate all disks, and so users may consider increasing this value.)

`spark.shuffle.file.buffer 64K` (Size of the in-memory buffer for each shuffle file output stream. These buffers reduce the number of disk seeks and system calls made in creating intermediate shuffle file)

Improve Serialization Performance

Serialization plays an important role in the performance of any distributed application. Often, this will be the first thing that should be tuned to optimize a Spark application.

`spark.serializer org.apache.spark.serializer.KryoSerializer` (when speed is necessary)

`spark.kryo.referenceTracking false`

`spark.kryoserializer.buffer 2000` (If the objects are large, may need to increase the size further to fit the size of the object being deserialized).

SparkSQL is ideally suited for mixed procedure jobs where SQL code is combined with Scala, Java, or Python programs. In general, the SparkSQL command line interface is used for single user operations and ad hoc queries.

For multi-user SparkSQL environments, it is recommended to use a Thrift server connected via JDBC.

Spark SQL Tuning

Below are some guidelines for Spark SQL tuning:

- To compile each query to Java bytecode on the fly, turn on `sql.codegen`. This can improve performance for large queries but can slow down very short queries.

`spark.sql.codegen true`

`spark.sql.unsafe.enabled true`

- Configuration of in-memory caching can be done using the `setConf` method on `SQLContext` or by running `SET key=value` commands using SQL.

- `spark.sql.inMemoryColumnarStorage.compressed true` (will automatically select a compression codec for each column based on statistics of the data)
- `spark.sql.inMemoryColumnarStorage.batchSize 5000` (Controls the size of batches for columnar caching. Larger batch sizes can improve memory utilization and compression, but risk OOMs when caching data)
- The columnar nature of the ORC format helps avoid reading unnecessary columns, but it is still possible to read unnecessary rows. ORC avoids this type of overhead by using predicate push-down with three levels of built-in indexes within each file: file level, stripe level, and row level. This combination of indexed data and columnar storage reduces disk I/O significantly, especially for larger datasets where I/O bandwidth becomes the main bottleneck for performance.
- By default, ORC predicate push-down is disabled in Spark SQL. To obtain performance benefits from predicate push-down, enable it explicitly, as follows:

```
spark.sql.orc.filterPushdown=true
```

- In SparkSQL to automatically determine the number of reducers for joins and groupbys, use the parameter:
`spark.sql.shuffle.partitions 200, (default value is 200)`
- This property can be put into `hive-site.xml` to override the default value.
- Set log to WARN in `log4j.properties` to reduce log level.



Running the Thrift server and connecting to spark-sql through beeline is the recommended option for multi-session testing.

Compression for Hive

Set the following Hive parameters to compress the Hive output files using Snappy compression:

```
hive.exec.compress.output=true
hive.exec.orc.default.compress=SNAPPY
```

Change the Log Directory for All Applications

To change the default log from the `/var` prefix to `/data/disk1`, follow these steps:

1. Log into the cloudera home page and click My Clusters.
2. From the configuration drop-down list select "All Log Directories."
3. Click Save.

Summary

When building an infrastructure to enable this modernized architecture which could scale to thousands of nodes, operational efficiency can't be an afterthought.

To achieve a seamless operation of the application at this scale, you need:

- Infrastructure automation of Cisco UCS servers with service profiles and Cisco Data Center network automation with application profiles with Cisco ACI.
- Centralized Management and Deep telemetry and Simplified granular trouble-shooting capabilities and Multi-tenancy allowing application workloads including containers, micro-services, with the right level of security and SLA for each workload.
- Cisco UCS with Cisco Intersight and Cisco ACI can enable this cloud scale architecture deployed and managed with ease.
- Cisco UCS S3260 M5 with CDP on CDP is an ideal system for wide variety of big data analytics solution delivering new approach to data where machine learning intelligently auto scale workloads up and down for more cost-effective use of private cloud infrastructure.

For More Information

For additional information, see the following resources:

- To find out more about Cisco UCS big data solutions, see <http://www.cisco.com/go/bigdata>.
- To find out more about Cisco Data Intelligence Platform, see <https://www.cisco.com/c/dam/en/us/products/servers-unified-computing/ucs-c-series-rack-servers/solution-overview-c22-742432.pdf>
- To find out more about Cisco UCS big data validated designs, see http://www.cisco.com/go/bigdata_design
- To find out more about Cisco UCS AI/ML solutions, see <http://www.cisco.com/go/ai-compute>
- To find out more about Cisco ACI solutions, see <http://www.cisco.com/go/aci>
- To find out more about Cisco validated solutions based on Software Defined Storage, see <https://www.cisco.com/c/en/us/solutions/data-center-virtualization/software-defined-storage-solutions/index.html>
- Cloudera Data Platform Data Center 7 release note, see <https://docs.cloudera.com/cdp/latest/release-guide/topics/cdpdc-release-guide-overview.html>
- CDP Data Center Requirements and Supported Versions, see <https://docs.cloudera.com/cdp/latest/release-guide/topics/cdpdc-requirements-supported-versions.html>

Bill of Materials

This section provides the BoM for the 8 S3260 Storage Server Hadoop Base Rack. See [Table 8](#) for BOM for the Hadoop Base rack and [Table 9](#) for Red Hat Enterprise Linux License.

Table 8 Bill of Materials for Cisco UCS C240M5SX Hadoop Nodes Base Rack

Part Number	Description	Qty
UCSS-S3260	Cisco UCS S3260 Storage Server Base Chassis	8
CON-OSP-UCSS3260	SNTC 24X7X4OS, Cisco UCS S3260 Storage Server Base Chassis	8
UCSC-PSU1-1050W	Cisco UCS 1050W AC Power Supply for Rack Server	32
CAB-N5K6A-NA	Power Cord, 200/240V 6A North America	32
CIMC-LATEST	IMC SW (Recommended) latest release for C-Series Servers.	8
UCSC-C3X60-RAIL	UCS C3X60 Rack Rails Kit	8
UCSS-S3260-BBEZEL	Cisco UCS S3260 Bezel	8
N20-BKVM	KVM local IO cable for UCS servers console port	16
UCSC-C3260-SIOC	Cisco UCS C3260 System IO Controller with VIC 1300 incl.	8
UCSC-C3260-SIOC	Cisco UCS C3260 System IO Controller with VIC 1300 incl.	8
UCS-S3260-M5SRB	UCS S3260 M5 Server Node for Intel Scalable CPUs	8
UCS-S3260-DRAID	UCS S3260 Dual Raid based on LSI 3316	8
UCS-S3260-M5HS	UCS S3260 M5 Server Node HeatSink	16
UCS-S3260-M5SRB	UCS S3260 M5 Server Node for Intel Scalable CPUs	8
UCS-S3260-DRAID	UCS S3260 Dual Raid based on LSI 3316	8
UCS-S3260-M5HS	UCS S3260 M5 Server Node HeatSink	16
UCS-C3K-HD4TB	UCS C3000 4TB NL-SAS 7200 RPM 12Gb HDD w Carrier- Top Load	448
UCS-S3260-G3SD48	UCS S3260 480G Boot SSD (Micron 6G SATA)	32
UCS-S3260-56HD4	Cisco UCS C3X60 Four row of drives containing 56 x 4TB	8
UCS-CPU-I6230R	Intel 6230R 2.1GHz/150W 26C/35.75MB DDR4 2933MHz	16
UCS-MR-X32G2RT-H	32GB DDR4-2933-MHz RDIMM/2Rx4/1.2v	96
RHEL-2S2V-3A	Red Hat Enterprise Linux (1-2 CPU,1-2 VN); 3-Yr Support Req	28
CON-ISV1-EL2S2V3A	ISV 24X7 RHEL Server 2Socket-OR-2Virtual; ANNUAL List Price	28
CAB-N5K6A-NA	Power Cord, 200/240V 6A North America	56
RACK2-UCS2	Cisco R42612 standard rack, w/side panels	2

Part Number	Description	Qty
CON-SNT-RCK2UCS2	SNTC 8X5XNBD, Cisco R42612 standard rack, w side panels	2
UCS-SP-FI6332	(Not sold standalone) UCS 6332 1RU FI/12 QSFP+	4
CON-OSP-SPFI6332	ONSITE 24X7X4 (Not sold standalone) UCS 6332 1RU FI/No PSU/3	4
UCS-PSU-6332-AC	UCS 6332 Power Supply/100-240VAC	8



For NameNode, we configured ten 1.8TB 10K RPM SAS HDD.

Table 9 Red Hat Enterprise Linux License

Part Number	Description	Qty
RHEL-2S2V-3A	Red Hat Enterprise Linux	30
CON-ISV1-EL2S2V3A	3-year Support for Red Hat Enterprise Linux	30



For Cloudera Data Platform Data Center (CDP DC) software licensing requirement, contact [Cloudera Data Platform software - Sales](#)

Appendix

Configure Data Drives on Name Node and Other Management Nodes

This section describes the steps needed to configure non-OS disk drives as RAID1 using the StorCli command. All drives are part of a single RAID1 volume. This volume can be used for staging any client data to be loaded to HDFS. This volume will not be used for HDFS data.



To configure data drives on the Name node and other nodes, if the drive state displays as JBOD, creating RAID in the subsequent steps will fail with the error *“The specified physical disk does not have the appropriate attributes to complete the requested command.”*

To configure data drive on the Name node and other management nodes, follow these steps:

1. If the drive state shows up as JBOD, it can be converted into Unconfigured Good using Cisco UCSM or storcli64 command. The following steps should be performed if the state is JBOD.
2. Get the enclosure id as follows:

```
ansible all -m shell -a './storcli64 pdlist -a0 | grep Enc | grep -v 252 | awk
'{print $4}' | sort | uniq -c | awk '{print $2}'"
```

```
[root@rhe101 ~]# ansible all -m shell -a './storcli64 pdlist -a0 | grep Enc | grep -v 252 | awk '{print $4}' | sort | uniq -c | awk '{print $2}'"
rhe106.hdp3.cisco.local | CHANGED | rc=0 >>
 24 Enclosure Device ID: 66
 24 Enclosure position: 0
rhe104.hdp3.cisco.local | CHANGED | rc=0 >>
 24 Enclosure Device ID: 66
 24 Enclosure position: 0
rhe108.hdp3.cisco.local | CHANGED | rc=0 >>
 24 Enclosure Device ID: 66
 24 Enclosure position: 0
```



It has been observed that some earlier versions of storcli64 complains about the above mentioned command as if it is deprecated. In this case, please use `“./storcli64 /c0 show all| awk '{print $1}' | sed -n '/[0-9]:[0-9]/p|awk '{print substr($1,1,2)}'|sort -u”` command to determine enclosure id.



With S3260, use `-a0` and `-a1` or `c0` and `c1` since there are two controller per node.

3. Convert to unconfigured good:

```
ansible datanodes -m command -a './storcli64 /c0 /e18 /sall set good force"
```



The Cisco UCS S3260 with Dual Storage Controller requires running same command mentioned above as `c0` – Storage Controller 1 and `c1` – Storage Controller 2 and enclosure 18 for Storage Controller 1 and enclosure 36 for Storage Controller 2.

4. Verify status by running the following command:

```
# ansible datanodes -m command -a "./storcli64 /c0 /e18 /sall show"
```

5. Run this script as root user on rhelnn01 to rhel3 to create the virtual drives for the management nodes:

```
#vi /root/raid1.sh
./storcli64 -cfgldadd
r1[$1:1,$1:2,$1:3,$1:4,$1:5,$1:6,$1:7,$1:8,$1:9,$1:10,$1:11,$1:12,$1:13,$1:14,$1:15,
$1:16,$1:17,$1:18,$1:19,$1:20,$1:21,$1:22,$1:23,$1:24] wb ra nocachedbadbbu
strpsz1024 -a0
```



The script (above) requires enclosure ID as a parameter.

6. Run the following command to get enclosure id:

```
#!/storcli64 pdlist -a0 | grep Enc | grep -v 252 | awk '{print $4}' | sort | uniq -c
| awk '{print $2}'
#chmod 755 raid1.sh
```

7. Run MegaCli script:

```
#!/raid1.sh <EnclosureID> obtained by running the command above
WB: Write back
RA: Read Ahead
NoCachedBadBBU: Do not write cache when the BBU is bad.
Strpsz1024: Strip Size of 1024K
```



The command (above) will not override any existing configuration. To clear and reconfigure existing configurations refer to Embedded MegaRAID Software Users Guide available: www.broadcom.com.

8. Run the following command. State should change to Online:

```
ansible namenodes -m command -a "./storcli64 /c0 /e18 /sall show"
```

9. State can also be verified in UCSM as show below in Equipment>Rack-Mounts>Servers>Server # under Inventory/Storage/Disk tab:

Name	Size (MB)	Serial	Operability	Disk State	Presence	Technology	Bootable
Storage Controller PCI-0							
Storage Controller SAS 1							
Disk 1	1718555	S1Z107N6u10000704MCTA	Operable	Online	Equipped	HDD	False
Disk 2	1718555	S1Z1080F6000033465J2	Operable	Online	Equipped	HDD	False
Disk 3	1718555	S1Z110P701000003158J5	Operable	Online	Equipped	HDD	False
Disk 4	1718555	S1Z101Z3N0000083531J32	Operable	Online	Equipped	HDD	False
Disk 5	1718555	S1Z101Y900000003318E95	Operable	Online	Equipped	HDD	False
Disk 6	1718555	S1Z11V83000000033P1CJ	Operable	Online	Equipped	HDD	False
Disk 7	1718555	S1Z11V84E00000033a071L	Operable	Online	Equipped	HDD	False

Configure Data Drives on Data Nodes

To configure non-OS disk drives as individual RAID0 volumes using StorCli command, follow this step. These volumes will be used for HDFS Data.

1. Issue the following command from the admin node to create the virtual drives with individual RAID 0 configurations on all the data nodes:

```
[root@rhelnn01 ~]# ansible datanodes -m command -a "./storcli64 -cfgeachdskraid0 WB
RA direct NoCachedBadBBU strpsz1024 -a0"

rhel7.hdp3.cisco.local | SUCCESS | rc=0 >>
Adapter 0: Created VD 0
Configured physical device at Encl-18:Slot-7.
Adapter 0: Created VD 1
Configured physical device at Encl-18:Slot-6.
Adapter 0: Created VD 2
Configured physical device at Encl-18:Slot-8.
Adapter 0: Created VD 3
Configured physical device at Encl-18:Slot-5.
Adapter 0: Created VD 4
Configured physical device at Encl-18:Slot-3.
Adapter 0: Created VD 5
Configured physical device at Encl-18:Slot-4.
Adapter 0: Created VD 6
Configured physical device at Encl-18:Slot-1.
Adapter 0: Created VD 7
Configured physical device at Encl-18:Slot-2.
..... Omitted Ouput
14 physical devices are Configured on adapter 0.

Exit Code: 0x00
```



The command (above) will not override existing configurations. To clear and reconfigure existing configurations, refer to the Embedded MegaRAID Software Users Guide available at www.broadcom.com.

About the Author

Sajeev Sharma, Engineer, Cisco Systems Inc.

Sanjeev Sharma is a Performance Engineer in the Computing Systems Product Group. He is part of solution engineering team focusing on big data infrastructure and performance.

Hardik Patel, Big Data Solutions Architect, Cisco Systems, Inc.

Hardik Patel is a Big Data Solutions Architect in the Computing Systems Product Group. He is part of the solution engineering team focusing on big data infrastructure, solutions, and performance.

Acknowledgements

For their support and contribution to the design, validation, and creation of this Cisco Validated Design, the author would like to thank:

- Karthik Kulkarni, Architect, Computing Systems Product Group, Cisco Systems, Inc.