



Cisco UCS S3260 Storage Server with MapR Converged Data Platform

Last Updated: October 31, 2016

Cisco Validated Design



About Cisco Validated Designs

The CVD program consists of systems and solutions designed, tested, and documented to facilitate faster, more reliable, and more predictable customer deployments. For more information visit

<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 Unity, Collaboration Without Limitation, EtherFast, EtherSwitch, Event Center, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, iQuick Study, IronPort, the IronPort logo, 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)

© 2016 Cisco Systems, Inc. All rights reserved.

Table of Contents

About Cisco Validated Designs	2
Executive Summary	7
Solution Overview	8
Introduction	8
Solution	8
Audience	8
Solution Summary	9
Scaling/Performance Options	9
MapR Converged Data Platform	10
Data Storage and Retrieval on MapR Converged Platform	11
Technology Overview	12
MapR Reference Architecture	12
Cisco UCS S3260 Storage Server for Big Data and Analytics	13
Cisco UCS S3260 Storage Server	13
Cisco UCS VIC 1387	14
Cisco UCS 6300 Series Fabric Interconnects	15
Cisco UCS Manager	16
MapR Converged Data Platform 5.1	16
MapR Enterprise-Grade Platform Services	17
MapR Open Source Technologies	18
Solution Design	21
Requirements	21
Rack and PDU Configuration	21
Port Configuration on Fabric Interconnects	22
Server Configuration and Cabling for Cisco UCS S3260 Storage Server	22
Cisco UCS S3260 Storage Server Scaling with Cisco Application Centric Infrastructure (ACI)	24
Software Distributions and Versions	29
MapR (5.1)	29
Red Hat Enterprise Linux (RHEL)	29
Software Versions	29
Fabric Configuration	30
Performing Initial Setup of Cisco UCS 6332 Fabric Interconnects	30

Configure Fabric Interconnect A	30
Configure Fabric Interconnect B	31
Logging In to Cisco UCS Manager	32
Adding a Block of IP Addresses for KVM Access	32
Enabling Uplink Ports	33
Configuring VLANs	34
Enabling Server Ports	36
Creating Chassis Profile	37
Creating Disk Zoning Policy	37
Creating Chassis Firmware Package Policy	41
Creating Chassis Profiles Template	43
Associating Chassis Profile to Individual Chassis.	47
Creating a Storage Profile for Boot Drives	50
Creating Pools for Service Profile Templates	55
Creating MAC Address Pools	55
Creating a Server Pool	57
Creating Policies for Service Profile Templates	59
Creating Host Firmware Package Policy	59
Creating QoS Policies	60
Creating the Local Disk Configuration Policy	63
Creating Server BIOS Policy	64
Creating the Boot Policy	66
Creating Power Control Policy	68
Creating a Service Profile Template	70
Configuring the Storage Provisioning for the Template	71
Configuring Network Settings for the Template	73
Configuring the vMedia Policy for the Template	81
Configuring Server Boot Order for the Template	82
Configuring Server Assignment for the Template	84
Configuring Operational Policies for the Template	85
Installing Red Hat Enterprise Linux 7.2	88
Post OS Install Configuration	111
Setting Up Password-less Login	111
Configuring /etc/hosts	112

Creating a Red Hat Enterprise Linux (RHEL) 7.2 Local Repo.....	113
Creating the Red Hat Repository Database.	115
Setting up ClusterShell	116
Installing httpd	117
Disabling SELinux	117
Disabling the Linux Firewall.....	118
Set Up all Nodes to use the RHEL Repository	118
Configuring DNS	119
Upgrading the Cisco Network Driver for VIC1387	120
Setting up JAVA	121
NTP Configuration	123
Enabling Syslog	124
Setting ulimit.....	125
Set TCP Retries	126
Disable Swapping.....	126
Disable Transparent Huge Pages.....	127
Disable IPv6 Defaults	127
Configuring Data Drives	127
Cluster Verification and Micro-Benchmark.....	128
Running the Cluster Verification Script.....	129
Change Permissions to Executable	133
Running STREAM Benchmark	133
Running MapR RPCtest.....	134
Running IOzone Benchmark.....	135
Installing MapR	137
Planning the Cluster.....	137
MapR Services	137
Node Types.....	138
Hostnames and Roles	139
Preparing Packages and Repositories.....	140
RPM Repositories for MapR Core Software.....	141
RPM Repositories for Hadoop Ecosystem Tools.....	141
MapR Software Installation	144
Installing MapR packages	145

Verification of Installation	146
Formatting Disks with the disksetup Script	147
Identify and Format the Data Disks for MapR	147
Bringing Up the Cluster	149
Initialization Sequence	150
Installing Spark	151
Verifying Cluster Status	153
Verify Cluster Status using the Web Interface	153
Verifying Cluster Status Using the Command Line Interface	153
Installing Additional Hadoop Components	153
Troubleshooting	154
Conclusion	155
Bill of Materials	156
About the Authors	161
Acknowledgements	161

Executive Summary

Apache Hadoop enables the creation of big data applications over huge amounts of data and is one of the fastest-growing technologies providing a competitive advantage for businesses across industries. One of the major challenges of big data systems now is managing the rapidly growing amount of data and the corresponding increasing costs.

The Cisco® UCS S3260 Storage Server is specifically designed to address this problem. This next-generation high-density storage system provides up to 600 terabytes (TB) in only four rack units (4RU), providing the best dollar-per-terabyte value while delivering superior computing performance and a balanced core-to-spindle ratio. The Cisco UCS S3260 Storage Server provides superior performance at a lower total cost. Fewer servers mean less rack space, fewer OS and software licenses, and less networking equipment to purchase and maintain, and lower power and cooling costs.

The Cisco UCS S3260 Storage Server is the latest addition to the highly successful Cisco Unified **Computing System™ (Cisco UCS®) reference architecture for big data. It complements Cisco UCS Integrated Infrastructure for Big Data and Analytics**, a highly scalable architecture for big data systems that includes computing, storage, and networking resources fully managed through Cisco UCS Manager and linearly scalable to thousands of nodes using Cisco Nexus® 9000 Series Switches and the Cisco Application Centric Infrastructure (**Cisco ACI™**) platform.

The MapR Converged Data Platform integrates the power of Hadoop with enterprise storage for developing and running innovative data applications built around the Big Data Architecture. This platform is powered by the industry's **fastest, most reliable, secure and open data infrastructure**.

The Cisco UCS S3260 Storage Server for Big Data and Analytics together with MapR Converged Data Platform allow enterprises to build reliable big data applications enabling organizations to tap into the power of big data.

Solution Overview

Introduction

Data and information is being generated at an unmatched scale. Traditional transactional data is being supplemented with data from streaming systems and then stored for long periods of time both for archival and regulatory purposes. Sensors, Internet of Things (IoT) devices, social media, online transactions, and other sources are all generating data that needs to be efficiently captured, processed, and stored.

Applications built on these platforms must reliably process data at large scale, while retrieving data efficiently to meet the business requirements.

The Cisco Unified Computing System (UCS) with Cisco UCS S3260 Storage Server and MapR Converged Data Platform support these capabilities for the broadest set of big data applications. MapR Converged Data Platform integrates the power of Hadoop and enterprise storage for developing and running innovative data applications. MapR enables big data applications using Hadoop and more to serve business-critical needs that cannot afford to lose data, must run on a 24x7 basis and require immediate recovery from node and site failures – all with a smaller data center footprint

Solution

This CVD describes a scalable architecture and deployment procedures for the MapR Converged Data Platform using Cisco UCS S3260 Storage Servers providing all of the benefits of the Cisco UCS Integrated Infrastructure.

As a technology leader in Hadoop, the MapR Converged Data Platform distribution provides enterprise-class big data solutions that are fast to develop and easy to administer. With significant investment in critical technologies, MapR offers a complete Hadoop platform – a platform that is fully optimized for performance and scalability.

Deployed as part of a comprehensive data center architecture, using Cisco UCS S3260 Storage Servers with MapR fundamentally transforms the way that organizations do business with Hadoop technology by delivering a powerful and flexible infrastructure that: increases business and IT agility, reduces total cost of ownership (TCO), and delivers exceptional return on investment (ROI) at scale.

The solution is built on the Cisco Unified Computing System and includes computing, storage, connectivity and unified management capabilities to help companies manage the immense amount of data they collect today. It uses Cisco UCS 6300 Series Fabric Interconnects and Cisco UCS S3260 Storage Servers. This architecture is specifically designed for performance and linear scalability for big data workloads.

Audience

This document describes the architecture and deployment procedures for the MapR Converged Data Platform with eight Cisco UCS S3260 Storage Servers each with two C3x60 M4 server nodes based on the Cisco Unified Computing System (UCS). The intended audience of this document includes, but is

not limited to, sales engineers, field consultants, professional services, IT managers, partner engineering and customers who want to deploy the MapR Converged Data Platform on Cisco UCS.

Solution Summary

This CVD describes in detail the process of installing the MapR Converged Data Platform 5.1 and the configuration details of the cluster. It also details application configuration for MapR, and the installation of additional services, like Spark and MapR Steams.

The configuration using Cisco UCS S3260 Storage Servers is shown in Table 1 . This configuration supports the massive scalability that big data enterprise deployments demand. This architecture can scale to thousands of servers with Cisco Nexus 9000 Series Switches.

Table 1 Cisco UCS S3260 Storage Server Configuration Details

Cisco UCS S3260 Storage Server Data Nodes
Connectivity: Two Cisco UCS 6332 Fabric Interconnects
Eight Cisco UCS S3260 Storage Servers, each with two C3x60 M4 server nodes, each node with:
<ul style="list-style-type: none"> 2 x Intel Xeon processor E5-2680 v4 CPUs (14 cores on each CPU)
<ul style="list-style-type: none"> 256 GB of memory
<ul style="list-style-type: none"> Cisco UCS-C3K-M4RAID SAS Modular RAID Controller with 4-GB FBWC
<ul style="list-style-type: none"> Twenty-four 4-TB 7,200-rpm LFF SAS drives (96 terabytes [TB] per server node)(1.54 petabytes[PB] total)
<ul style="list-style-type: none"> Cisco UCS VIC 1387 (with 2 x 40 Gigabit Ethernet QSFP ports)
<ul style="list-style-type: none"> 2 x 480-GB 6-Gbps 2.5-inch enterprise value SATA SSD drives for boot

Scaling/Performance Options

The server supports up to 28 LFF disk drives of various capacities. The base reference configuration consists of 24 LFF drives each with 4 TB capacity. The storage capacities can be extended as shown in Table 2 below.

Table 2 Disk Drive Options

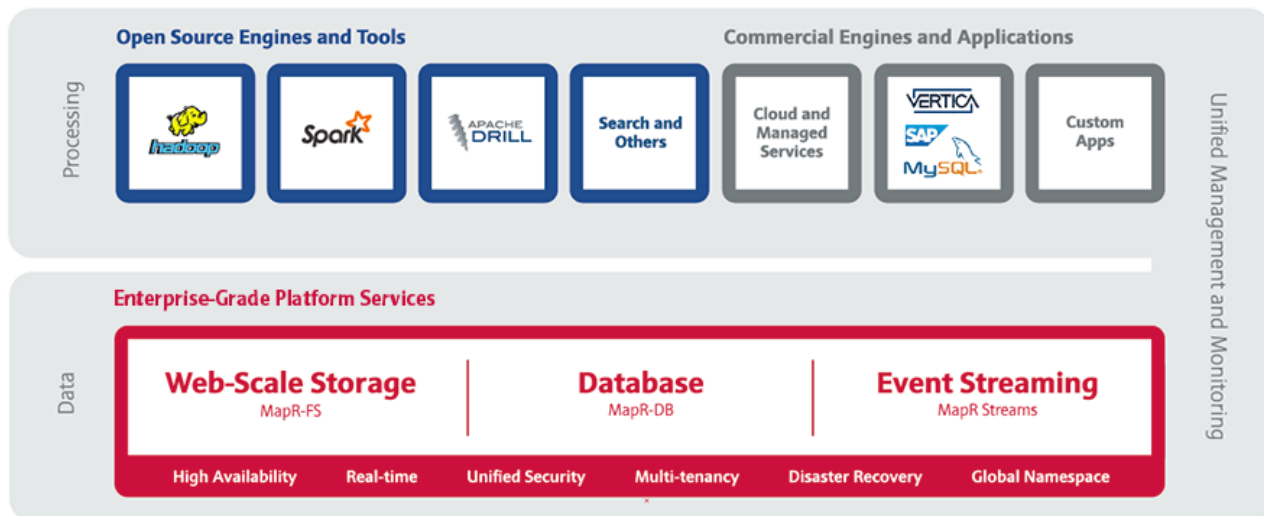
HDD Type	Capacity with 24 disk drives	Capacity with 28 disk drives
4 TB 7200 -rpm LFF SAS drives	1.54 PetaBytes	1.8 PetaBytes
6 TB 7200 -rpm LFF SAS drives	2.38 PetaBytes	2.68 PetaBytes
8 TB 7200 -rpm LFF SAS drives	3.07 PetaBytes	3.58 PetaBytes

10 TB 7200 ~prm LFF SAS drives	3.84 PetaBytes	4.48 PetaBytes
-----------------------------------	----------------	----------------

MapR Converged Data Platform

The MapR Converged Data Platform (Figure 1) integrates Hadoop and scalable enterprise storage to power a new generation of big data applications. The MapR Platform delivers enterprise grade security, reliability and real-time performance while dramatically lowering both hardware and operational costs of your most important applications and data.

Figure 1 The MapR Converged Data Platform



MapR supports dozens of open source projects and is committed to using industry-standard APIs to provide a frictionless method of developing and deploying new applications that can meet the most stringent production runtime requirements.

Enterprise-Grade Platform Services

MapR Platform Services are the core data handling capabilities of the MapR Converged Data Platform. Modules include MapR-FS, MapR-DB and MapR Streams. Its enterprise-friendly design provides a familiar set of file and data management services, including a global namespace, high availability, data protection, self-healing clusters, access control, real-time performance, secure multi-tenancy, and management and monitoring.

Open Source Engines and Tools

MapR packages a broad set of Apache open source ecosystem projects that enable big data applications. The goal is to provide an open platform that provides the right tool for the job. MapR tests and integrates open source ecosystem projects such as Spark, Hive, Drill, Hbase and Mesos, among others.

Commercial Engines & Applications

One of the key developer benefits of the MapR Converged Data Platform is its basis on well known, open APIs and interfaces. This enables commercial software vendors such as SAP HANA and SAS to easily deploy large-scale applications onto the MapR Platform. It also means that even small teams of developers can create enterprise-grade software products by exploiting the built-in protections of the MapR Platform in combination with mature commercial processing engines.

Data Storage and Retrieval on MapR Converged Platform

Companies have realized the power of big data and are now collecting unprecedented amounts of data. They need to get value from this data, often in real-time. Sensors, IoT devices, social network data and online transactions are just a few examples. They are all generating data continuously, 24x7. This data needs to be captured, stored, monitored and processed quickly in order to make informed, data-driven decisions in real-time. In addition, real-time streaming data needs to be sent to data stores where it can be used for traditional analysis and reporting, data discovery and as input to sophisticated machine-learning algorithms.

The MapR Converged Data Platform provides all the technologies to implement this architecture while **also providing additional benefits. With MapR's innovations data can be written directly to the Hadoop** storage while allowing the data processing applications to run as independent services within the cluster. This creates a very resilient architecture. The real-time processing applications become subscribers to the incoming data feeds. If the application goes down due to some failure, there is no data loss. A new instance of the application picks up the data stream where the original left off.

Technology Overview

MapR Reference Architecture

Figure 2 shows the base configuration of 16 nodes.

Figure 2 Reference Architecture for MapR

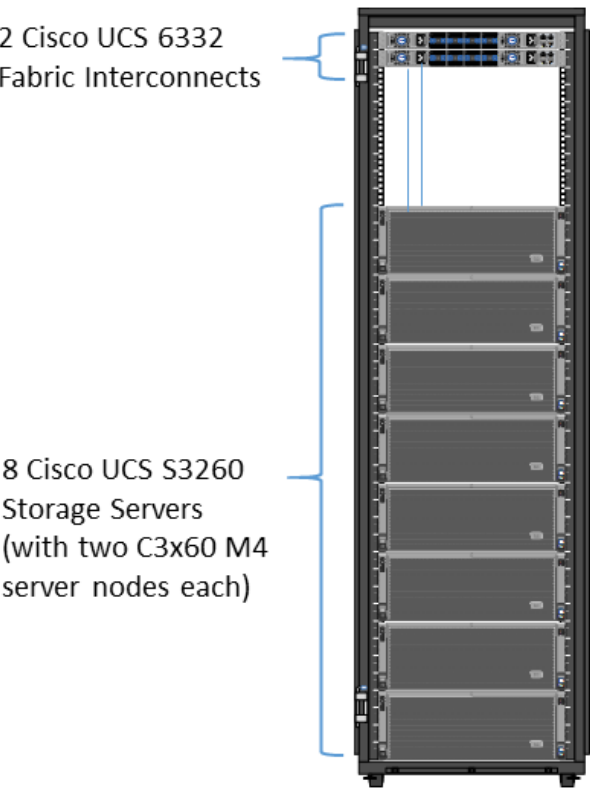


Table 3 Configuration Details

Component	Description
Connectivity	2 x Cisco UCS 6332 32-Port Fabric Interconnects
MapR Nodes	8 x Cisco UCS S3260 Storage Servers with: CLDB, Node Manager, NFS, History Server, Spark-History-Server, Resource Manager, Zookeeper. *Please refer to the Hostname and Roles section for specific service assignment and configuration details.

Cisco UCS S3260 Storage Server for Big Data and Analytics

This solution is based on the Cisco Unified Computing System (Cisco UCS) infrastructure using Cisco UCS 6300 Series Fabric Interconnects, and Cisco UCS S3260 Storage Servers. This architecture is specifically designed for high performance and linear scalability for big data workloads and is built using the following components:

Cisco UCS S3260 Storage Server

The Cisco UCS S3260 Storage Server (Figure 3) is a high-density modular storage server designed to deliver efficient, industry-leading storage for data-intensive workloads. The Cisco UCS S3260 Storage Server is a modular chassis with dual server nodes (two servers per chassis) and up to 60 large-form-factor (LFF) drives in a 4RU form factor. The server uses dual Intel® Xeon® Processor E5-2600 v4 Series CPUs and supports up to 512 GB of main memory and a range of hard-disk-drive (HDD) options. It comes with a pass-through controller or a RAID card with 4 GB cache and host bus adapter (HBA) controller, and up to two internal solid-state-disk (SSD) drives for boot, as shown in Figure 4 .

Figure 3 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 10 TB per HDD (supports 4 TB, 6 TB, 8 TB and 10 TB) and can be mixed with up to 28 SSDs.

Figure 4 Cisco UCSC S3260 Storage Server Chassis –Back view, showing Two Servers



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.

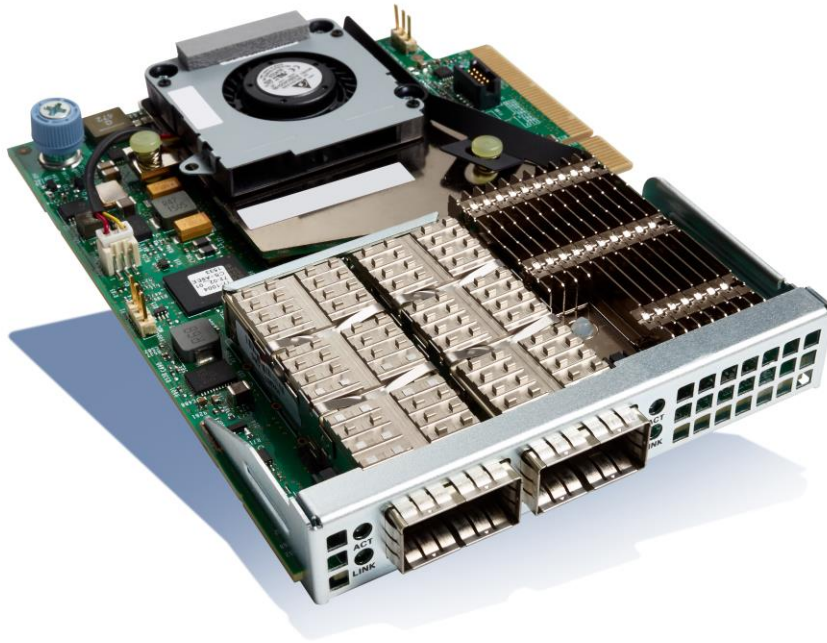
The Cisco UCS S3260 Storage Server Chassis includes a Cisco UCS Virtual Interface Card (VIC) 1300 platform chip onboard the system I/O controller, offering high-performance bandwidth with dual-port 40 Gigabit Ethernet and FCoE interfaces per system I/O controller.

Cisco UCS VIC 1387

Cisco UCS Virtual Interface Cards (VICs) are unique to Cisco. The Cisco UCS VIC 1387 incorporates next-generation converged network adapter (CNA) technology from Cisco, and offers dual 40-Gbps ports designed for use with Cisco UCS Rack-Mount Servers. Optimized for virtualized networking, this card delivers high performance and bandwidth utilization, and supports up to 256 virtual devices.

The Cisco UCS VIC 1387 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 5)

Figure 5 Cisco UCS VIC 1387



Cisco UCS 6300 Series Fabric Interconnects

Cisco UCS 6300 Series Fabric Interconnects as shown in Figure 6 , provide high-bandwidth, low-latency connectivity for servers, with Cisco UCS Manager providing integrated, unified management for all connected devices. The Cisco UCS 6300 Series Fabric Interconnects are a core part of Cisco UCS, providing low-latency, lossless 40 Gigabit Ethernet, Fibre Channel over Ethernet (FCoE), and Fibre Channel functions with management capabilities for systems 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 and automates 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.

Figure 6 Cisco UCS 6332 32-Port Fabric Interconnect

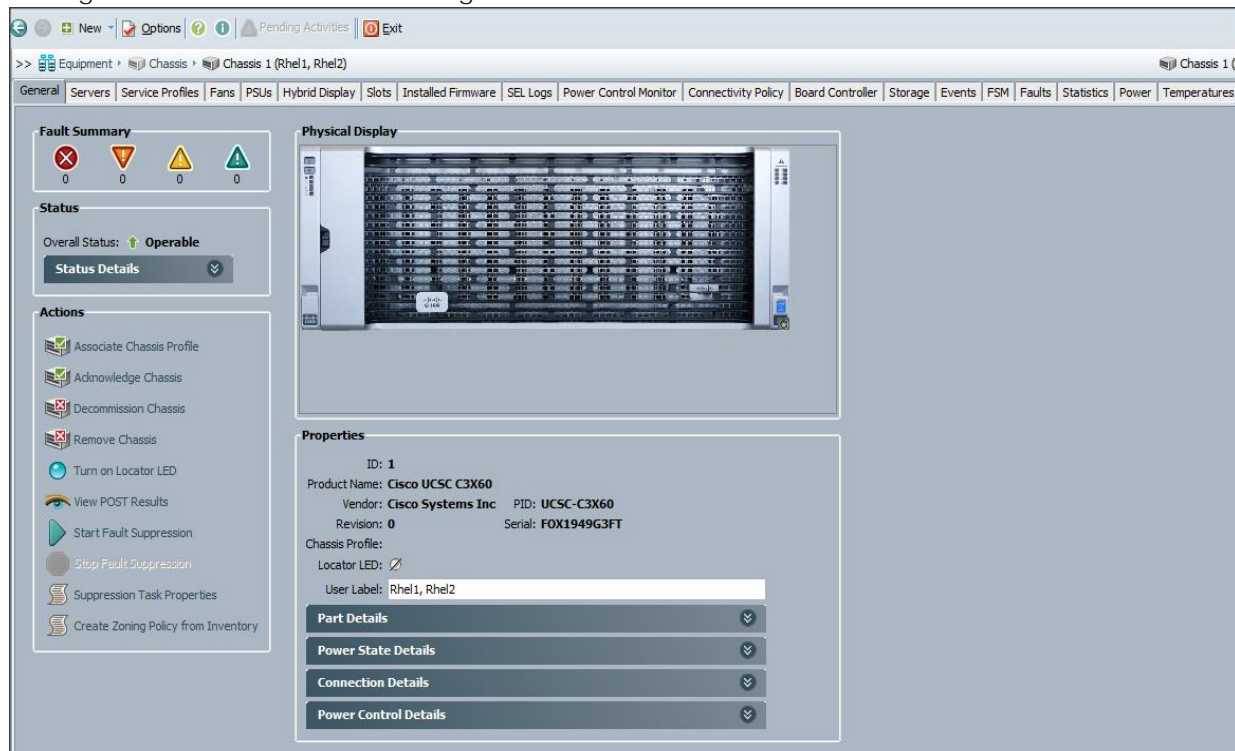


Cisco UCS Manager

Cisco UCS Manager resides within the Cisco UCS 6300 Series Fabric Interconnect. It makes the system self-aware and self-integrating, managing all of the system components as a single logical entity. Cisco UCS Manager can be accessed through an intuitive graphical user interface (GUI), as shown in Figure 7, 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 the process takes minutes instead of days. This simplification allows IT departments to shift their focus from constant maintenance to strategic business initiatives.

The new Cisco UCS Manager has smart capabilities such as predictive drive failure and rebuild. With the integration with Cisco UCS S3260 Storage Server, Cisco UCS Manager can be configured to have hot spare drives in case of any drive failure. In such a case, Cisco UCS Manager will automatically detect the failed drives and replace it with one of the available hot spare drives, rebuild it and make it available to use within the Chassis.

Figure 7 Cisco UCS Manager



MapR Converged Data Platform 5.1

As one of the technology leaders in Hadoop, the MapR Converged Data Platform provides enterprise-class big data solutions that are fast to develop and easy to administer. With significant investment in **critical technologies**, MapR offers one of the industry's most comprehensive Hadoop platforms, fully optimized for performance and **scalability**. MapR's distribution delivers more than a dozen tested and

validated Hadoop software modules over a fortified data platform, offering exceptional ease of use, reliability and performance for big data solutions.

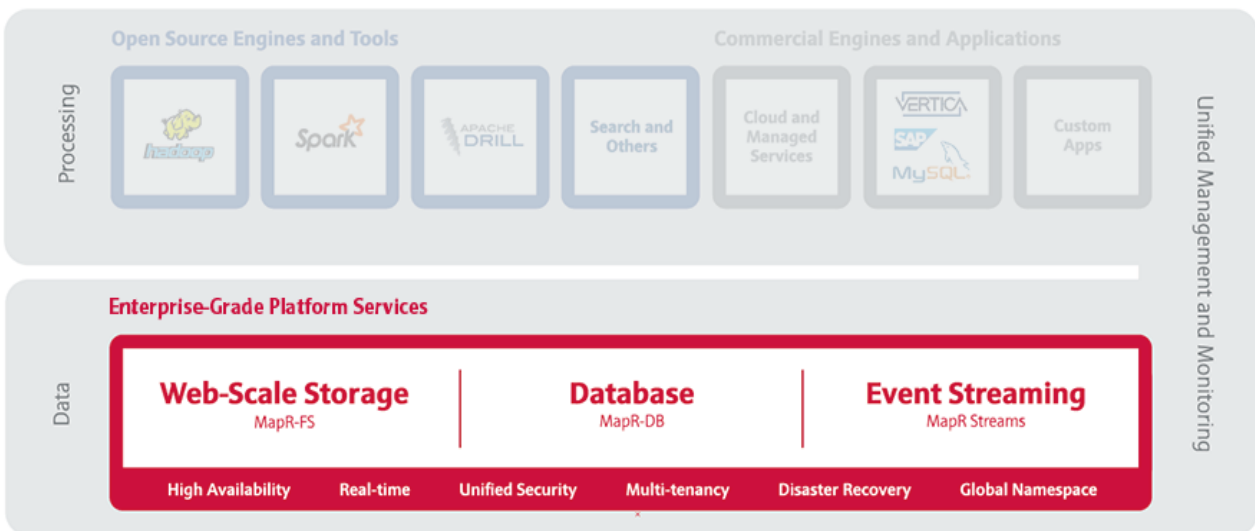
Highlights of MapR Converged Data Platform are:

- Performance – Ultra-fast performance and throughput
- Scalability – Up to a trillion files, with no restrictions on the number of nodes in a cluster
- Standards-based API's and tools – Standard Hadoop API's, ODBC, JDBC, LDAP, Linux PAM, and more
- MapR Direct Access NFS – Random read/write, real-time data flows, existing non-Java applications work seamlessly
- Manageability – Advanced management console, rolling upgrades, REST API support
- Integrated security – Kerberos and non-Kerberos options with wire-level encryption
- Advanced multi-tenancy – Volumes, data placement control, job placement control, queues, and more
- Consistent snapshots – Full data protection with point-in-time recovery
- High availability – Ubiquitous HA with a no-NameNode architecture, YARN HA, NFS HA
- Disaster recovery – Cross-site replication with mirroring
- MapR-DB – Integrated enterprise-grade NoSQL database
- MapR Streams – Global publish-subscribe event streaming system for big data

MapR Enterprise-Grade Platform Services

MapR Platform Services (Figure 8) are the core data handling capabilities of the MapR Converged Data Platform. Modules include MapR-FS, MapR-DB and MapR Streams. Its enterprise-friendly design provides a familiar set of file and data management services, including a global namespace, high availability, data protection, self-healing clusters, access control, real-time performance, secure multi-tenancy, and management and monitoring.

Figure 8 MapR Enterprise-grade Platform Services



Enterprise Storage

MapR-FS is an enterprise standard POSIX file system that provides high-performance read/write data storage for the MapR Converged Data Platform. MapR-FS includes important features for production deployments such as fast NFS access, access controls, and transparent data compression at a virtually unlimited scale.

Database

MapR-DB is an enterprise-grade, high performance, in-Hadoop NoSQL database management system. It is used to add real-time, operational analytics capabilities to applications built on the Hadoop or Spark ecosystems. Because it is integrated into the MapR Converged Data Platform, it inherits the protections and high performance capabilities.

Event Streaming

MapR Streams is a global publish-subscribe event streaming system for big data. It connects data producers and consumers worldwide in real-time, with unlimited scale. MapR Streams is the first big data-scale streaming system built into a converged data platform. It makes data available instantly to stream processing and other applications, and is the only big data streaming system to support global event replication reliably at IoT scale.

MapR Open Source Technologies

MapR packages a broad set of Apache open source ecosystem projects that enable big data applications. The goal is to provide an open platform that provides the right tool for the job. MapR tests and integrates open source ecosystem projects such as Spark, Drill, Solr, HBase, among others.

Figure 9 MapR Open Source Engines and Tools

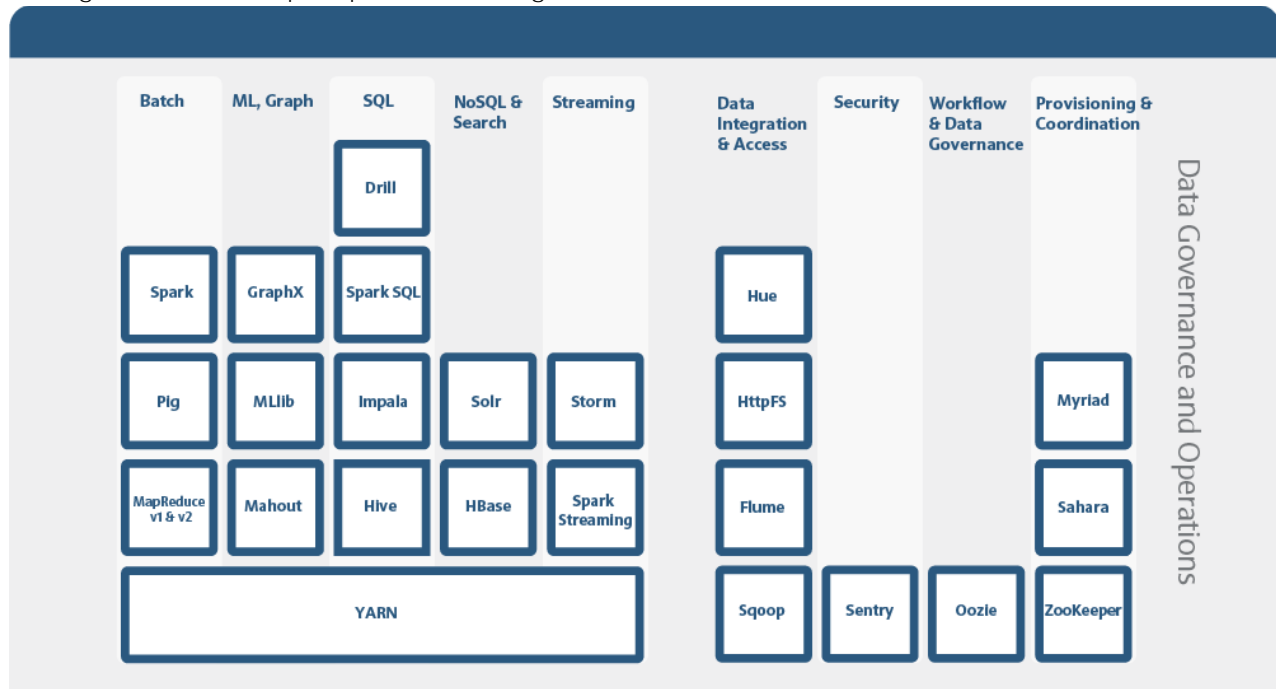


Figure 9 above shows the Apache open source projects supported by the MapR Converged Data Platform. Features of some of the key technologies are highlighted below. In conjunction with the data ingestion capabilities provided by MapR Streams these technologies are building blocks for a system based on the Lambda Architecture.

MapReduce

MapReduce is a powerful framework for processing large, distributed sets of structured or unstructured data on a Hadoop cluster. The key feature of MapReduce is its ability to perform processing across an entire cluster of nodes, with each node processing its local data. This feature makes MapReduce orders of magnitude faster than legacy methods of processing big data. MapReduce is a common choice to perform the pre-compute processing of batch views in the batch layer of the Lambda Architecture.

HBase

HBase is a database that runs on a Hadoop cluster. It is not a traditional relational database management system (RDBMS). Data stored in HBase also does not need to fit into a rigid schema as with an RDBMS, making it ideal for storing unstructured or semi-structured data. HBase stores data in a table-like format with the ability to store billions of rows with millions of columns over multiple nodes in a cluster. HBase can be used to store the pre-computed batch views of data held in the serving layer of the Lambda Architecture.

Solr

Solr is a full-text search and indexing engine that enables large-scale search, navigation and analytics on textual data. It is often used for search and discovery, predictive analytics and data retrievals. In conjunction with the real-time capabilities in MapR, Solr provides near real-time indexing so updated content is immediately available for search.

Drill

Drill is an open source, low-latency query engine for big data that delivers secure and interactive SQL analytics at petabyte scale. It can discover schemas on-the-fly and enable immediate exploration of data stored in Hadoop and NoSQL stores across a variety of data formats and sources.

Drill is fully ANSI SQL compliant, integrates seamlessly with existing BI and visualization tools, and supports thousands of users across thousands of nodes accessing data in the terabyte and petabyte range. Drill can operate on the merged view of data from the serving layer and speed layer of the Lambda Architecture providing a complete historical and real-time picture.

Spark

Spark is a fast and general-purpose engine for large-scale data processing. By adding Spark to the Hadoop deployment and analysis platform, and running it all on Cisco UCS, customers can accelerate streaming, interactive queries, machine learning and batch workloads, and offering experiences that deliver more insights in less time.

It can also be used for fast, interactive analysis on the merged view of data from both real time and batch processing.

Solution Design

Requirements

This CVD describes architecture and deployment procedures for MapR Converged Data Platform (MapR 5.1) on 8 Cisco UCS S3260 Storage Server chassis with two C3x60 M4 server nodes each, and three Cisco UCS C240M4 Rack servers as Hadoop Management nodes for Big Data and Analytics. The solution goes into detail configuring MapR 5.1 on the infrastructure.

The cluster configuration consists of the following:

- Two Cisco UCS 6332 Fabric Interconnects
- Eight Cisco UCS S3260 Storage Servers with two C3x60 M4 server nodes each
- Three Cisco UCS C240 M4 Rack Servers
- One Cisco R42610 Standard Rack
- Two Vertical Power distribution units (PDUs) (Country Specific)

Rack and PDU Configuration

Each rack consists of two vertical PDUs. The rack consists of two Cisco UCS 6332 Fabric Interconnects, eight Cisco UCS S3260 Storage Servers with two C3x60 M4 server nodes. Each chassis is connected to two vertical PDUs for redundancy, to help ensure availability during power source failure.



Note: Please contact your Cisco representative for country specific information.

Table 4 Rack Configuration

Position	Devices
42	Cisco UCS FI 6332
41	Cisco UCS FI 6332
40	Unused
39	Unused
38	Unused
37	Unused
36	Unused
35	Unused
34	Unused
33	Unused
32	Cisco UCS S3260 Storage Server
31	
30	
29	

8	Cisco UCS S3260 Storage Server
27	
26	
25	
24	Cisco UCS S3260 Storage Server
23	
22	
21	
20	Cisco UCS S3260 Storage Server
19	
18	
17	
16	Cisco UCS S3260 Storage Server
15	
14	
13	
12	Cisco UCS S3260 Storage Server
11	
10	
9	
8	Cisco UCS S3260 Storage Server
7	
6	
5	
4	Cisco UCS S3260 Storage Server
3	
2	
1	

Port Configuration on Fabric Interconnects

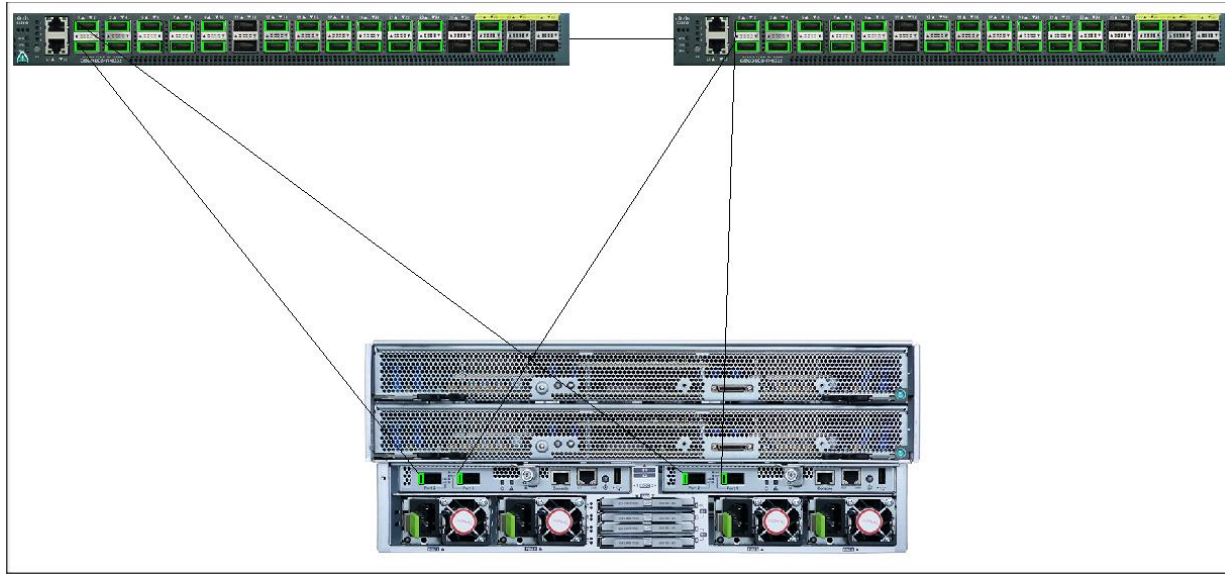
Port Type	Port Number
Network	32
Server	1-16

Server Configuration and Cabling for Cisco UCS S3260 Storage Server

The Cisco UCS S3260 Storage Server chassis is equipped with two Cisco C3x60 M4 server nodes, two Cisco UCSC-C3260-SIOC and four 480 GB SATA SSD. Each server blade is equipped with Intel Xeon E5-2680 v4 processors, 256 GB of memory and a Cisco UCS-C3K-M4RAID SAS Modular RAID Controller with 4-GB FBWC.

Figure 10 illustrates the port connectivity between the Fabric Interconnect and Cisco UCS S3260 Storage Server Chassis. Eight Cisco UCS S3260 Storage Server Chassis are used in single rack configurations.

Figure 10 Fabric Topology for Cisco UCS S3260 Storage Server

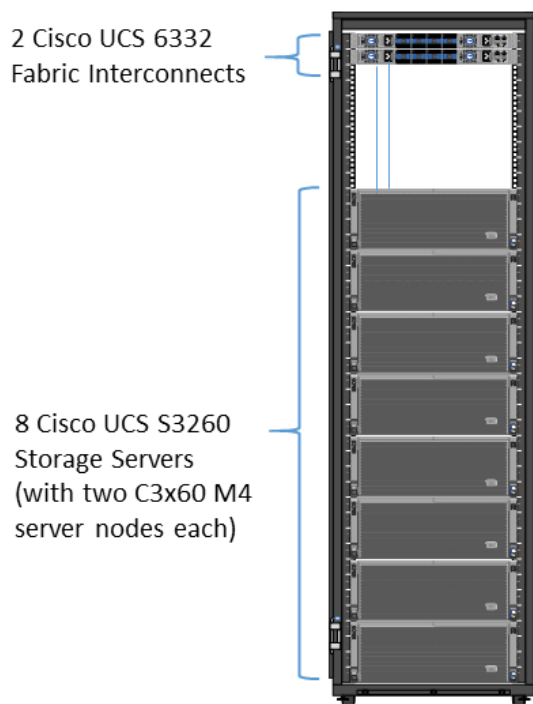


For more information on physical connectivity illustrations and cluster setup, see:

http://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/c-series_integration/ucsm3-1/b_C-Series-Integration_UCSM3-1/b_C-Series-Integration_UCSM3-1_chapter_010.html

Figure 11 depicts an 8-chassis/16-node cluster. Each chassis has two C3x60 M4 server nodes. Each link in the figure represents 40 Gigabit Ethernet link from each chassis connecting to a Fabric Interconnect as a Direct Connect. Every chassis is connected to both Fabric Interconnects represented with dual link.

Figure 11 16 Nodes Cluster Configuration



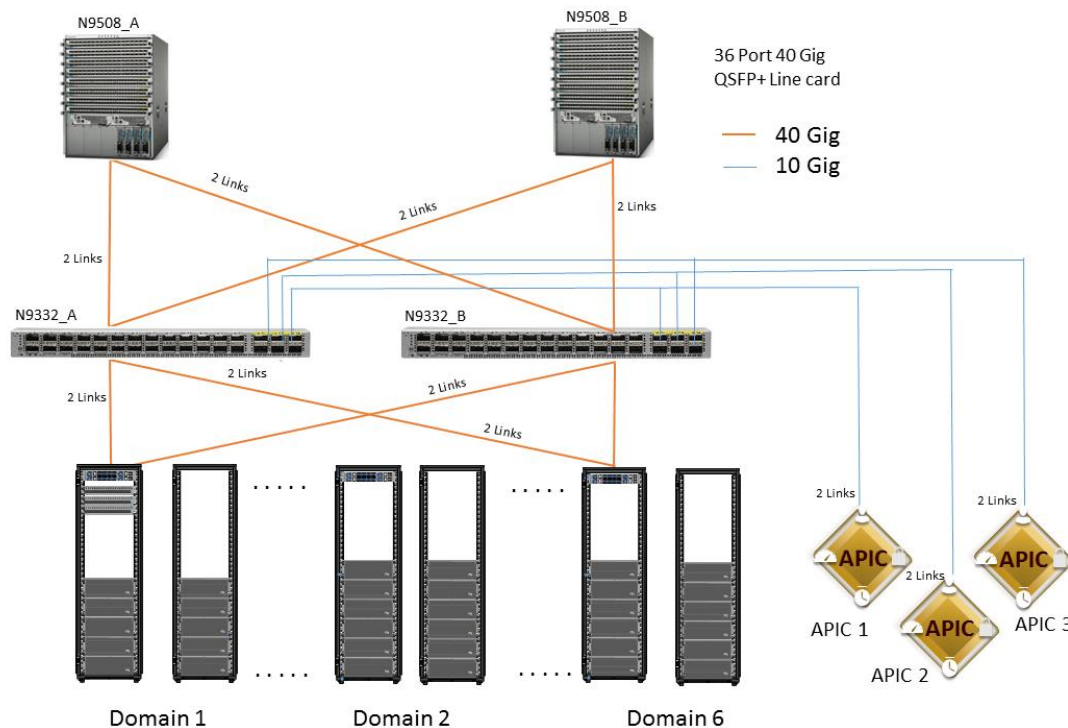
Since each chassis will have two server nodes, the top server node works with the left SIOC and the bottom server node works with right SIOC (as shown in Figure 11). Similarly, for the boot drives, the top two SSD slots are assigned for server blade 1 and bottom two SSD slots are assigned for server blade 2.

Cisco UCS S3260 Storage Server Scaling with Cisco Application Centric Infrastructure (ACI)

The system architecture includes the Cisco UCS S3260 Storage Server chassis. Each Fabric Interconnect domain can have 12 chassis under a single pair of Fabric Interconnects which are interconnected through the Cisco Application Centric Infrastructure (ACI) Fabric.

The ACI Fabric consists of three major components: the Application Policy Infrastructure Controller (APIC), spine switches, and leaf switches. These three components handle both the application of network policy and the delivery of packets.

The system architecture can be scaled up linearly and consists of one domain (one pair of FIs) connecting to ACI having two Nexus 9508 switches acting as a spine and two Nexus 9332PQ as the leaf switches and three APIC-L1 as an APIC appliance.



The following explains the system architecture for the base rack:

- The eight Cisco UCS S3260 Storage Server chassis are rack mounted and connected to a pair of Fabric Interconnects representing a domain through 40GE link (4x40GE link to a pair of FI)
- Multiple such domains can be connected to a pair of ACI leaf switches. Here 40GE x 4 links from each FI are connected to leaf switches. This is done through a virtual port-channel of two links connected to each of the Nexus 9332.
- Nexus 9332 receives the 4x40GE from each pair of Fabric Interconnect as a vPC (Virtual Port-Channel), i.e., two ports coming from each single FI as an uplink to the leaf. There are two vPC for the one domain in each of the 9332's connecting to a single pair of FIs
- Each leaf is connected to each spine via 2 x 40 Gig connectivity cables.
- **The three APIC's are connected to two leaves (Nexus 9332) via 10 gig SFP cable.**

Six UCS domains can be connected to a pair of Leaf switches, this will accommodate up to 70 Cisco UCS S3260 Storage Servers.

- 1 pair of FIs can connect up to 12 chassis
- 1 pair of leaf switches can connect up to six pairs of FI
- 1 Pair of line cards can connect up to nine pairs of leaf switches.

Further scaling can be done based on the requirement and is explained in Table 5 below.

Table 5 Spine to Leaf Connectivity

Spine	Line Card Pair	Ports Used	POD	Chassis	Leaf
N9508_A	Line Card 1	1-2	1	70	9332_1A
	Line Card 1	3-4			9332_1B
	Line Card 1	5-6	2	154	9332_2A
	Line Card 1	7-8			9332_2B
	Line Card 1	9-10	3	238	9332_3A
	Line Card 1	11-12			9332_3B
	Line Card 1	13-14	4	322	9332_4A
	Line Card 1	15-16			9332_4B
	Line Card 1	17-18	5	406	9332_5A
	Line Card 1	19-20			9332_5B

	Line Card 1	33-34	9	742	9332_9A
	Line Card 1	35-36			9332_9B

	Line Card 8	1-2	64	5362	9332_64A
	Line Card 8	3-4			9332_64B
	Line Card 8	1-2	65	5446	9332_65A
	Line Card 8	3-4			9332_65B
	Line Card 8	1-2	66	5530	9332_66A
	Line Card 8	3-4			9332_66B
	Line Card 8	1-2	67	5614	9332_67A
	Line Card 8	3-4			9332_67B
	Line Card 8	5-6	68	5698	9332_68A
	Line Card 8	7-8			9332_68B

	Line Card 8	9-10	72	6034	9332_72A
	Line Card 8	11-12			9332_72B

N9508_B	Line Card 1	1-2			9332_1A
	Line Card 1	3-4	1	70	9332_1B
	Line Card 1	5-6			9332_2A
	Line Card 1	7-8	2	154	9332_2B
	Line Card 1	9-10			9332_3A
	Line Card 1	11-12	3	238	9332_3B
	Line Card 1	13-14			9332_4A
	Line Card 1	15-16	4	322	9332_4B
	Line Card 1	17-18	5		9332_5A
	Line Card 1	19-20		406	9332_5B

	Line Card 1	33-34			9332_9A
	Line Card 1	35-36	9	742	9332_9B

	Line Card 8	1-2			9332_64A
	Line Card 8	3-4	64	5362	9332_64B
	Line Card 8	1-2			9332_65A
	Line Card 8	3-4	65	5446	9332_65B
	Line Card 8	1-2			9332_66A
	Line Card 8	3-4	66	5530	9332_66B
	Line Card 8	1-2			9332_67A
	Line Card 8	3-4	67	5614	9332_67B
	Line Card 8	5-6			9332_68A
	Line Card 8	7-8	68	5698	9332_68B

	Line Card 8	9-10			9332_72A
	Line Card 8	11-12	72	6034	9332_72B

Table 6 Leaf to Fabric Interconnect Connectivity.

LeafF		Ports Used	FI Domain	Chassis
-------	--	------------	-----------	---------

9332_1A		1-4	Domain-1	1-10
9332_1A		5-8	Domain-2	11-22
9332_1A		9-12	Domain-3	23-34
9332_1A		13-16	Domain-4	35-46
9332_1A		17-20	Domain-5	47-58
9332_1A		21-24	Domain-6	59-70
9332_1A		25-27	APIC	
		28-31	Uplink to Spine	
		32	Unused	
9332_2A		1-4	Domain-1	1-10
9332_2A		5-8	Domain-2	11-22
9332_2A		9-12	Domain-3	23-34
9332_2A		13-16	Domain-4	35-46
9332_2A		17-20	Domain-5	47-58
9332_2A		21-24	Domain-6	59-70
9332_2A		25-27	APIC	
		28-31	Uplink to Spine	
		32	Unused	

Based on the system architecture above, only six UCS FI Domains can be connected to the first pair of leaf switches due to the port restrictions, as the leaf switch needs to connect three APIC Appliances, providing the scalability for up to 70 chassis (ten chassis and three management nodes for the first domain and 12 chassis in each additional FI Domain). Each additional leaf pair can have up to seven UCS FI Domain, providing the scalability up to 84 chassis (12 chassis in each FI Domain). The Cisco UCS S3260 Storage Server can be scaled up to 742 chassis with just a pair of line cards on the Nexus 9508 spine switch. Nexus 9508 can have up to eight linecards, and with all eight linecards being used for scaling can connect up to 6034 chassis providing a massive storage solution for the industry.

The architecture above has four unused ports in each FI, these ports can either be used as an uplink to leaf switches or can be connected to external appliances. In that case these unused ports can be used to connect additional management nodes.

If the scaling is performed beyond the pair of leaf switches, it is recommended to connect APIC in three different leaf switches for maximum redundancy.



Note: This example shows a sample scaling capability using ACI. A production implementation might vary based on the customer's network throughput requirements. Please contact a Cisco representative for your specific requirements.

Software Distributions and Versions

The required software distribution versions are listed below.

MapR (5.1)

MapR Hadoop is API-compatible and includes or works with the family of Hadoop ecosystem components such as Spark, Hive, Pig, Flume, and others. For more information visit <https://www.mapr.com/>

Red Hat Enterprise Linux (RHEL)

The operating system supported is Red Hat Enterprise Linux 7.2. For more information visit <http://www.redhat.com>.

Software Versions

The software versions tested and validated in this document are shown in Table 7

Table 7 Software Versions

Layer	Component	Version or Release
Compute (Chassis) System IO Controller	Board Controller	1.0.14
	Chassis Management Controller	2.0(13aS4)
	Shared Adapter	4.1(2a)
	SAS Expander	04.08.01.B073
Compute (Server Nodes)	BIOS	C3x60M4.2.0.13c
	Board Controller	2.0
	CIMC Controller	2.0(13e)
Network	Cisco UCS 6332	3.1(2b)
	Kernel	5.0(3)N2(3.12b)
	Driver	2.3.0.30
Storage	Storage Controller SAS	29.00.1-0042
	Driver	06.810.10.00
Software	Red Hat Enterprise Linux Server	7.2 (x86_64)
	Cisco UCS Manager	3.1(2b)

Layer	Component	Version or Release
	MapR	5.1



The latest drivers can be downloaded from the link below:

[https://software.cisco.com/download/release.html?mdfid=283862063&release=2.0\(13\)&relind=AVAILABLE&flowid=25886&softwareid=283853158&rellifecycle=&reltype=latest](https://software.cisco.com/download/release.html?mdfid=283862063&release=2.0(13)&relind=AVAILABLE&flowid=25886&softwareid=283853158&rellifecycle=&reltype=latest)



The Latest Supported RAID controller Driver is already included with the RHEL 7.2 operating system.

Fabric Configuration

This section provides details for configuring a fully redundant, highly available Cisco UCS 6332 fabric configuration.

- Initial setup of the Fabric Interconnect A and B.
- Connect to Cisco UCS Manager using virtual IP address or using the web browser.
- Launch Cisco UCS Manager.
- Enable server, uplink and appliance ports.
- Start discovery process.
- Create pools and policies for service profile template.
- Create chassis and storage profile.
- Create Service Profile template and 16 Service profiles.
- Associate Service Profiles to servers.

Performing Initial Setup of Cisco UCS 6332 Fabric Interconnects

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

Configure Fabric Interconnect A

1. Connect to the console port on the first Cisco UCS 6332 Fabric Interconnect.
2. At the prompt to enter the configuration method, enter `console` to continue.
3. If asked to either perform a new setup or restore from backup, enter `setup` to continue.
4. Enter `y` to continue to set up a new Fabric Interconnect.
5. Enter `y` to enforce strong passwords.
6. Enter the password for the admin user.

7. Enter the same password again to confirm the password for the admin user.
8. When asked if this fabric interconnect is part of a cluster, answer `y` to continue.
9. Enter `a` for the switch fabric.
10. Enter the cluster name for the system name.
11. Enter the Mgmt0 IPv4 address.
12. Enter the Mgmt0 IPv4 netmask.
13. Enter the IPv4 address of the default gateway.
14. Enter the cluster IPv4 address.
15. To configure DNS, answer `y`.
16. Enter the DNS IPv4 address.
17. Answer `y` to set up the default domain name.
18. Enter the default domain name.
19. Review the settings that were printed to the console, and if they are correct, answer `yes` to save the configuration.
20. Wait for the login prompt to make sure the configuration has been saved.

Configure Fabric Interconnect B

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

For more information on Cisco UCS 6300 Series Fabric Interconnect, see:

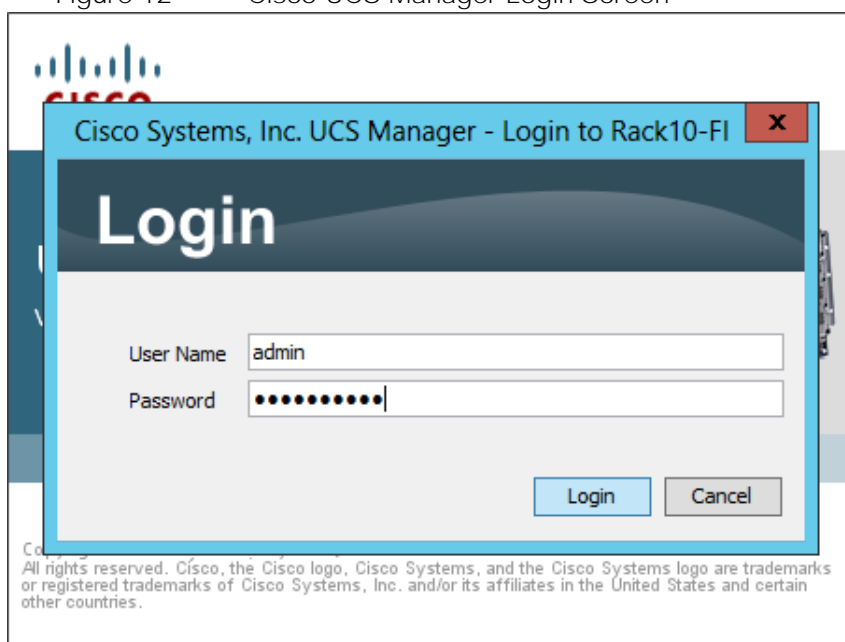
<http://www.cisco.com/c/en/us/products/servers-unified-computing/ucs-6300-series-fabric-interconnects/index.html>

Logging In to Cisco UCS Manager

To login to Cisco UCS Manager, complete the following 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 in to the Cisco UCS Manager. (Figure 12)

Figure 12 Cisco UCS Manager Login Screen

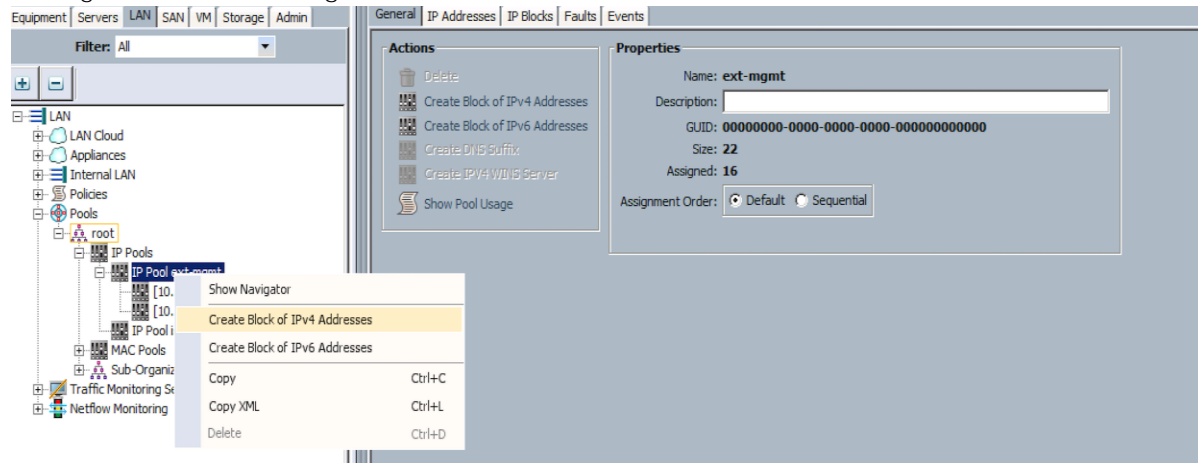


Adding a Block of IP Addresses for KVM Access

These steps provide details for creating a block of KVM IP addresses for server access in the Cisco UCS environment.

1. Select the LAN tab at the top of the left window.
2. Select Pools > IpPools > Ip Pool ext-mgmt.
3. Right-click IP Pool ext-mgmt.
4. Select Create Block of IPv4 Addresses.

Figure 13 Adding a Block of IPv4 Addresses for KVM Access Part 1



5. Enter the starting IP address of the block and number of IPs needed, as well as the subnet and gateway information. Figure 14

Figure 14 Adding Block of IPv4 Addresses for KVM Access Part 2

6. Click OK to create the IP block.
7. Click OK in the message box.

Enabling Uplink Ports

To enable uplinks ports, complete the following steps:

1. Select the **Equipment** tab on the top left of the window.
2. Select **Equipment > Fabric Interconnects > Fabric Interconnect A (primary) > Fixed Module**.
3. Click **Ethernet Ports** section.

4. Select port 32 that is connected to the uplink switch, right-click, and then select Configure as Uplink Port.
5. Select Equipment > Fabric Interconnects > Fabric Interconnect B (subordinate) > Fixed Module.
6. Click Ethernet Ports section.
7. Select port 16 that is connected to the uplink switch, right-click, and then select Configure as Uplink Port. Figure 15

Figure 15 Enabling Uplink Ports

	0	1	CC:46:D6:B3:16:0E	Server	Physical
	0	2	CC:46:D6:B3:16:12	Server	Physical
	0	3	CC:46:D6:B3:16:16	Server	Physical
	0	4	CC:46:D6:B3:16:1A	Server	Physical
	0	5	CC:46:D6:B3:16:1E	Server	Physical
	0	6	CC:46:D6:B3:16:22	Server	LAN Uplinks Manager Show Navigator Enable Disable Configure as Server Port Configure as Uplink Port Configure as FCoE Uplink Port Configure as FCoE Storage Port Configure as Appliance Port
	0	7	CC:46:D6:B3:16:26	Server	
	0	8	CC:46:D6:B3:16:2A	Server	
	0	9	CC:46:D6:B3:16:2E	Server	
	0	10	CC:46:D6:B3:16:32	Server	
	0	11	CC:46:D6:B3:16:36	Server	
	0	12	CC:46:D6:B3:16:3A	Server	
	0	13	CC:46:D6:B3:16:3E	Server	
	0	14	CC:46:D6:B3:16:3F	Server	
	0	15	CC:46:D6:B3:16:40	Server	
	0	16	CC:46:D6:B3:16:44	Server	

Configuring VLANs

VLANs are configured as in shown in Table 8

Table 8 VLAN Configurations

VLAN	NIC Port	Function
VLAN76	Eth0	Management & Data Traffic
VLAN77	Eth1	Data Traffic

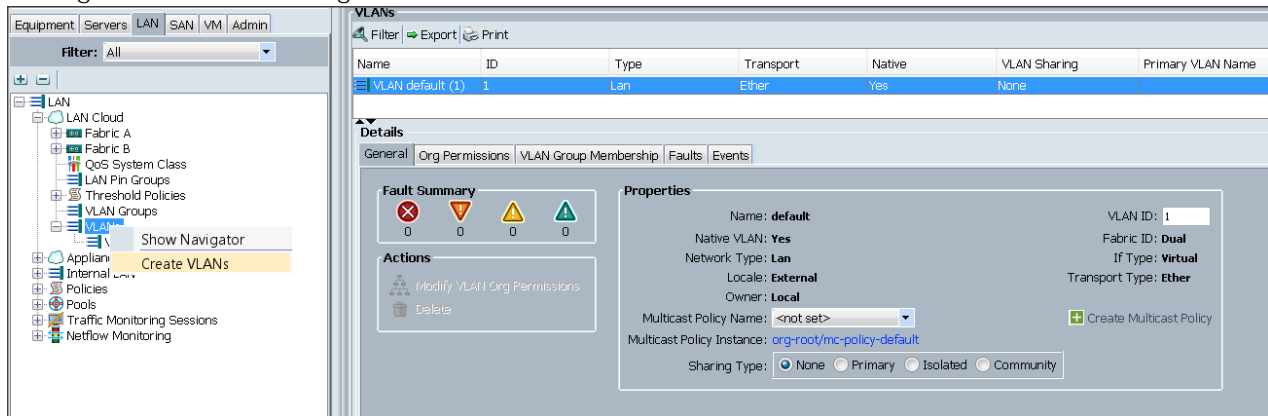
All of the VLANs created need to be trunked to the upstream distribution switch connecting to the fabric interconnects. For this deployment VLAN76 is configured for management access (installing and configuring OS, clustershell commands, set up NTP, user connectivity etc.) and both VLAN76 and VLAN77 are used for Hadoop Data traffic.

To configure VLANs in the Cisco UCS Manager GUI, complete the following steps:

1. Select the LAN tab in the left pane in the UCSM GUI.
2. Select LAN > LAN Cloud > VLANs.
3. Right-click the VLANs under the root organization.

4. Select Create VLANs to create the VLAN. Figure 16

Figure 16 Creating a VLAN



5. Enter vlan76 for the VLAN Name.
6. Keep multicast policy as <not set>.
7. Select Common/Global for vlan76.
8. Enter 76 in the VLAN IDs field for the Create VLAN IDs.
9. Click OK and then, click Finish.
10. Click OK in the success message box.

Figure 17 Creating VLAN for Data

Create VLANs

VLAN Name/Prefix:

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

☒ Common/Global
 ☐ Fabric A
 ☐ Fabric B
 ☐ Both Fabrics Configured Differently

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

VLAN IDs:

Sharing Type:
 ☒ None
 ☐ Primary
 ☐ Isolated
 ☐ Community

11. Click OK and then, click Finish.

12. Follow similar steps to create vlan77

Enabling Server Ports

To enable server ports, complete the following steps:

1. Select the Equipment tab on the top left of the window.
2. Select Equipment > Fabric Interconnects > Fabric Interconnect A (primary) > Fixed Module.
3. Click Ethernet Ports section.

4. Select all the ports that are connected to the Servers right-click them, and select **Configure as a Server Port**. (In this case it is ports 1-16).
5. Select **Equipment > Fabric Interconnects > Fabric Interconnect B (subordinate) > Fixed Module**.
6. Click **Ethernet Ports** section.
7. Select all the ports that are connected to the Servers right-click them, and select **Reconfigure > Configure as a Server Port**. (In this case it is ports 1-16).

Figure 18 Enabling Server Ports

Slot	Aggr. Port ID	Port ID	MAC	If Role	If Type
1	0	1	CC:46:D6:B3:16:0E	Enable	
1	0	2	CC:46:D6:B3:16:12	Disable	
1	0	3	CC:46:D6:B3:16:16	Configure as Server Port	
1	0	4	CC:46:D6:B3:16:1A	Configure as Uplink Port	
1	0	5	CC:46:D6:B3:16:1E	Configure as FCoE Uplink Port	
1	0	6	CC:46:D6:B3:16:22	Configure as FCoE Storage Port	
1	0	7	CC:46:D6:B3:16:26	Configure as Appliance Port	
1	0	8	CC:46:D6:B3:16:2A	Unconfigure	
1	0	9	CC:46:D6:B3:16:2E	Unconfigure FCoE Uplink Port	
1	0	10	CC:46:D6:B3:16:32	Unconfigure Uplink Port	
1	0	11	CC:46:D6:B3:16:36	Unconfigure FCoE Storage Port	
1	0	12	CC:46:D6:B3:16:3A	Unconfigure Appliance Port	
1	0	13	CC:46:D6:B3:16:3E		
1	0	14	CC:46:D6:B3:16:3F		
1	0	15	CC:46:D6:B3:16:40		
1	0	16	CC:46:D6:B3:16:44		

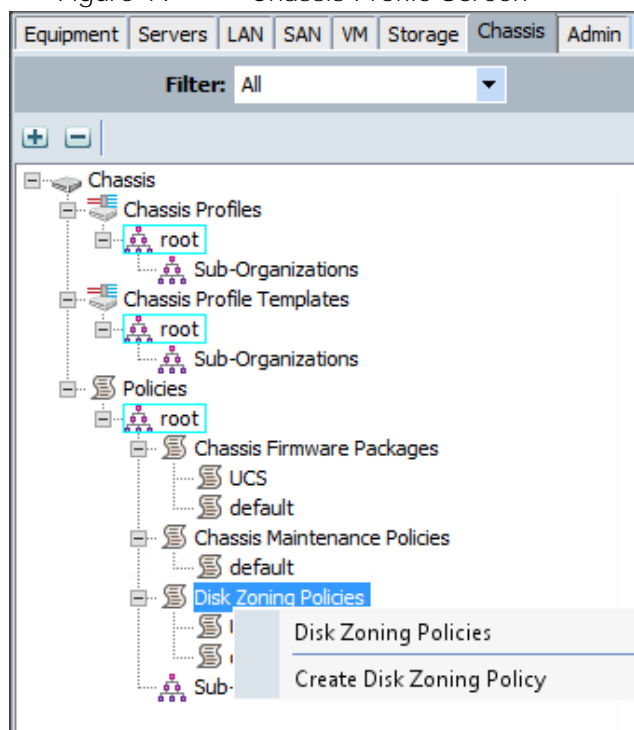
Creating Chassis Profile

Chassis profile is required to assign the number of drives to the particular server nodes and also to upgrade the chassis firmware.

Creating Disk Zoning Policy

1. Click the **Chassis** tab on UCS Manager on the top left menu (Figure 19).
2. Expand **Policies**→**Root**→**Disk Zoning Policies**.
3. Right click on the **Disk Zoning Policies** and click **Create Disk Zoning Policy**.

Figure 19 Chassis Profile Screen



4. On Create Disk Zoning Policy windows enter the Name UCS and click “+” to create the Disk Zoning. (Figure 20)

Figure 20 Disk Zoning Policy Screen

Create Disk Zoning Policy

Name:

Description:

Preserve Config: ☐

Disk Zoning Information

+ - Filter Export Print

Name	Slot Number	Ownership	Assigned to Ser...	Assigned to Contr...	Controller Type
------	-------------	-----------	--------------------	----------------------	-----------------

+ Add - Delete Modify

5. In the Add Slots to Policy window (Figure 21), select the “Dedicated” radio button. From the server drop down list choose “1”, from the controller drop down list choose “1”, in the slot range enter 1-24 and click “OK”.

Figure 21 Add Slots to Policy

Add Slots to Policy

Ownership: ☐ Unassigned ☒ Dedicated ☐ Shared ☐ Chassis Global Hot Spare

Server: 1

Controller: 1

Controller Type: **SAS**

Slot Range: 1-24

OK Cancel

- Click “+” again and In Add Slots to Policy window, select the “Dedicated” radio button. From the server drop down list choose “2”, from the controller drop down list choose “1”, in the slot range enter 29-52 and click “OK”.

Figure 22 Add Slots to Policy Screen 2

Add Slots to Policy

Ownership: ☐ Unassigned ☒ Dedicated ☐ Shared ☐ Chassis Global Hot Spare

Server: 2

Controller: 1

Controller Type: **SAS**

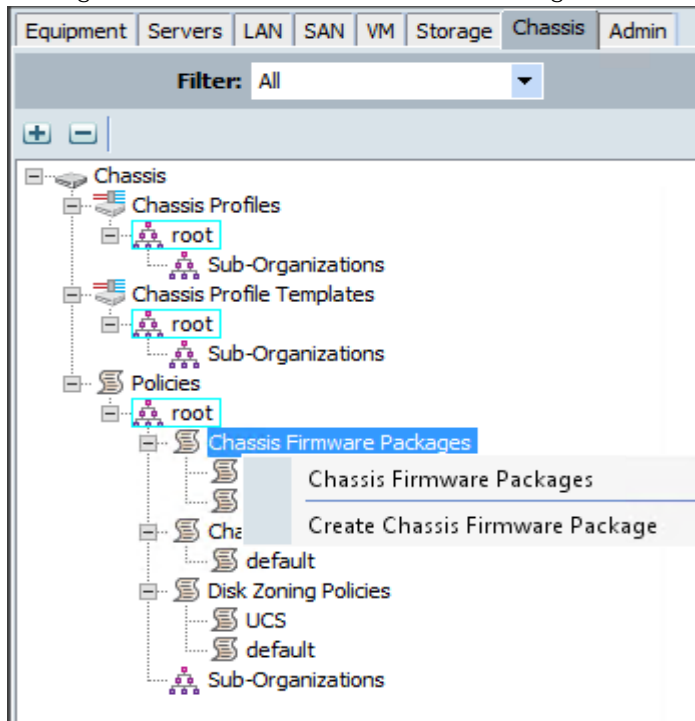
Slot Range: 29-52

OK Cancel

Creating Chassis Firmware Package Policy

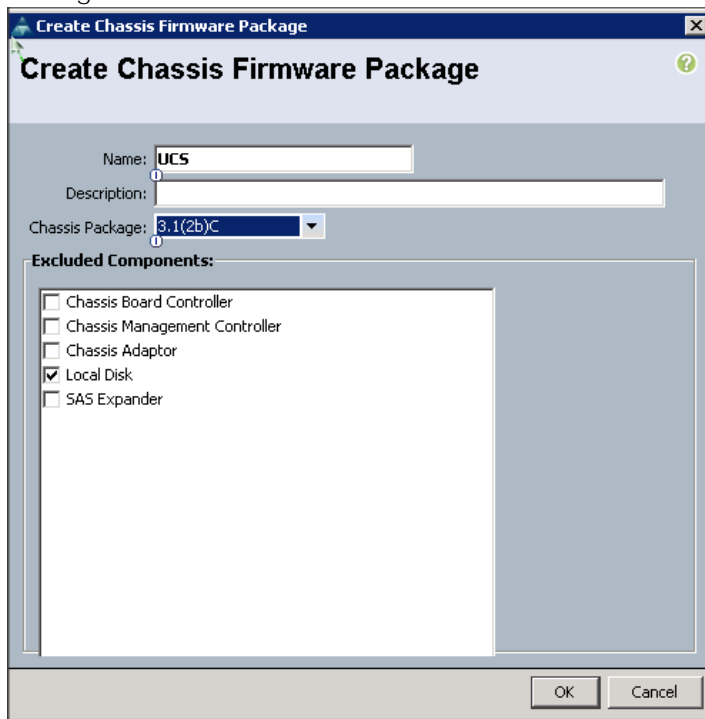
1. Right click on the Chassis Firmware Packages and click “Create Chassis Firmware Packages”. (Figure 23)

Figure 23 Chassis Firmware Packages



2. In Create Chassis Firmware Package window, enter UCS as the Name.
3. From the Chassis Packages drop down list choose the appropriate package and click OK.

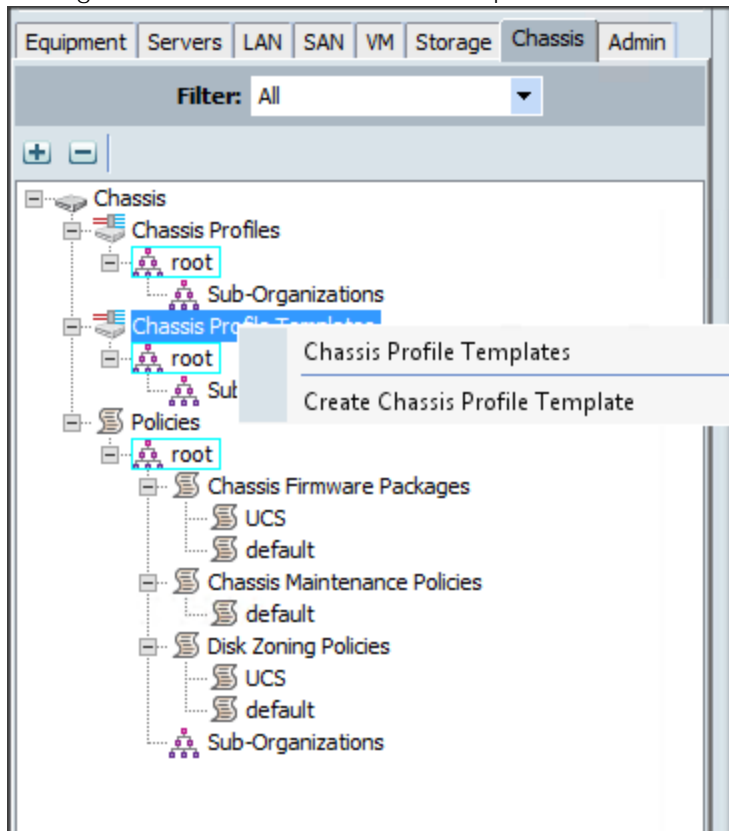
Figure 24 Create Chassis Firmware Screen



Creating Chassis Profiles Template

1. Under Chassis Profile Template, right click and click “Create Chassis Profile Templates” (Figure 25)

Figure 25 Chassis Profile Template



2. Enter the Name “UCS” and select “Updating Template” as the type, and click Next and Next again. (Figure 26)

Figure 26 Identify Chassis Profile Template

The screenshot shows the 'Create Chassis Profile Template' wizard in the Unified Computing System Manager. The title bar reads 'Create Chassis Profile Template'. The main window has a blue header with the text 'Unified Computing System Manager'. On the left, a sidebar titled 'Create Chassis Profile Template' lists four steps: 1. **Identify Chassis Profile Template** (checked), 2. **Maintenance Policy** (checked), 3. **Policies**, and 4. **Disk Zoning Policy**. The main area is titled 'Identify Chassis Profile Template' and contains the following text: 'You must enter a name for the chassis profile template and specify the template type. You can also enter a description of the template.' Below this, there are three sections: 'Name: UCS' with a text input field, 'Where: org-root' with a text input field, and 'Type: Initial Template' and 'Updating Template' with radio buttons. A large text area for a description is at the bottom. At the bottom right, there are four buttons: '< Prev', 'Next >', 'Finish', and 'Cancel'.

Create Chassis Profile Template

1. **Identify Chassis Profile Template**
2. **Maintenance Policy**
3. **Policies**
4. **Disk Zoning Policy**

Identify Chassis Profile Template

You must enter a name for the chassis profile template and specify the template type. You can also enter a description of the template.

Name:

The template will be created in the following organization. Its name must be unique within this organization.

Where: **org-root**

The template will be created in the following organization. Its name must be unique within this organization.

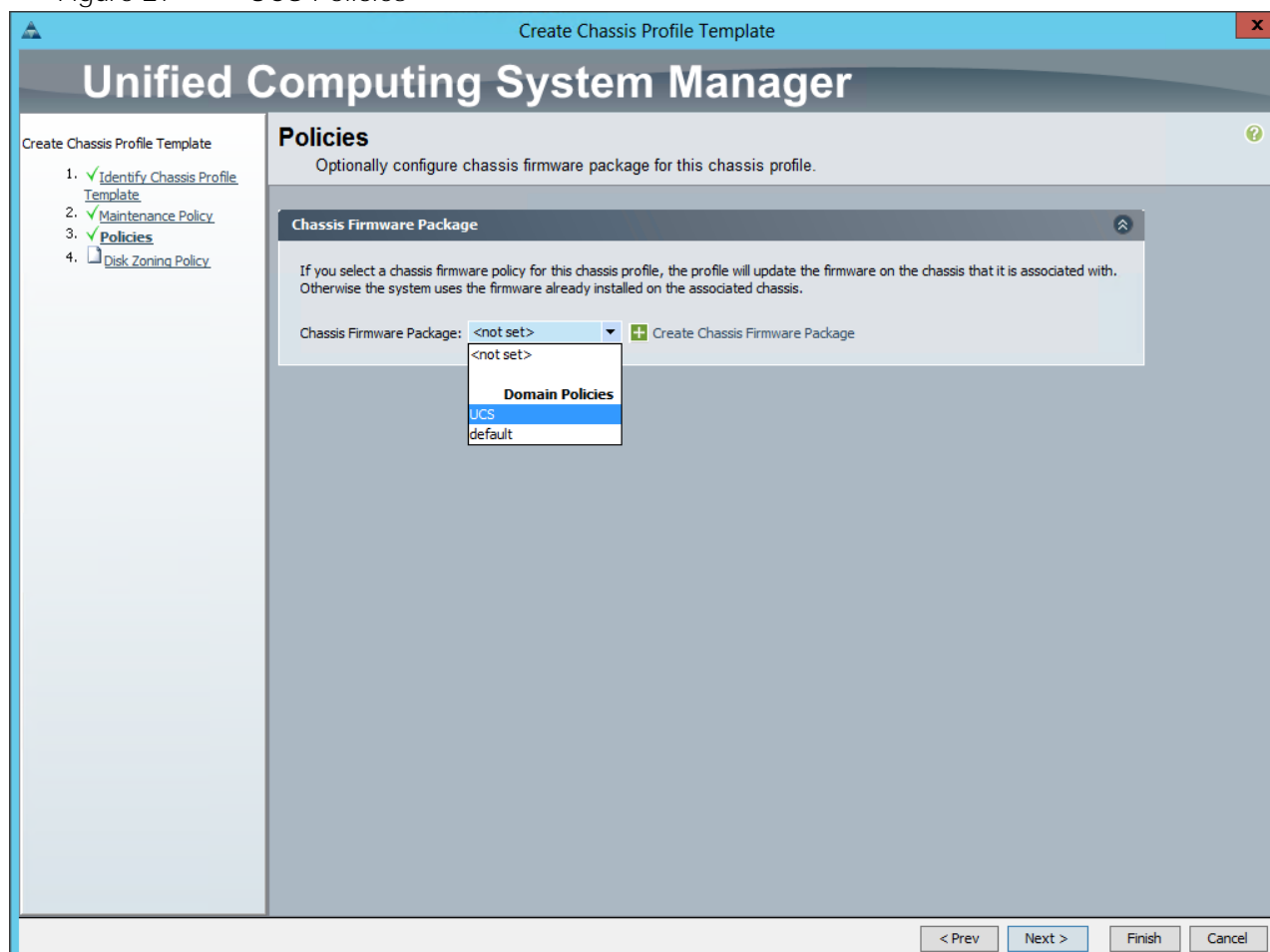
Type: ☐ Initial Template ☒ Updating Template

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

< Prev Next > Finish Cancel

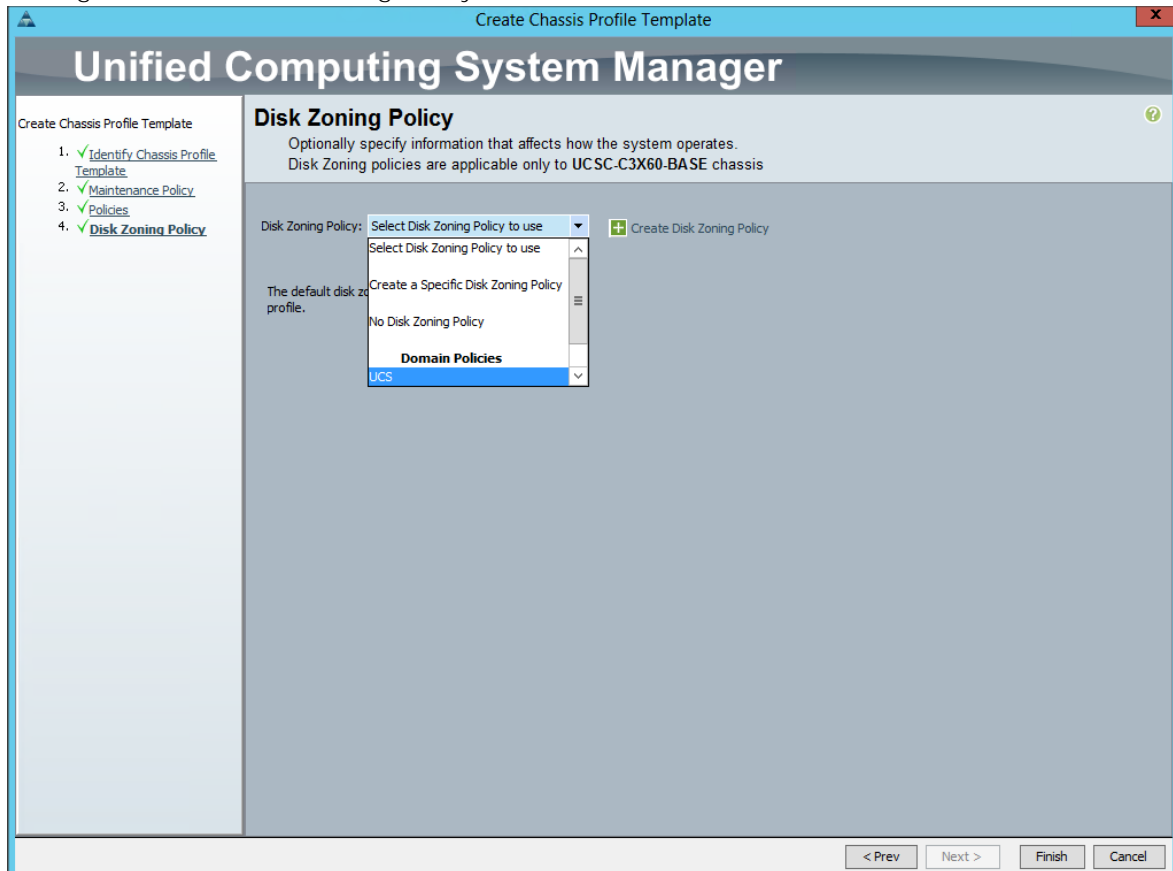
3. From the Chassis Firmware Package drop down list choose UCS and click Next. (Figure 27

Figure 27 UCS Policies



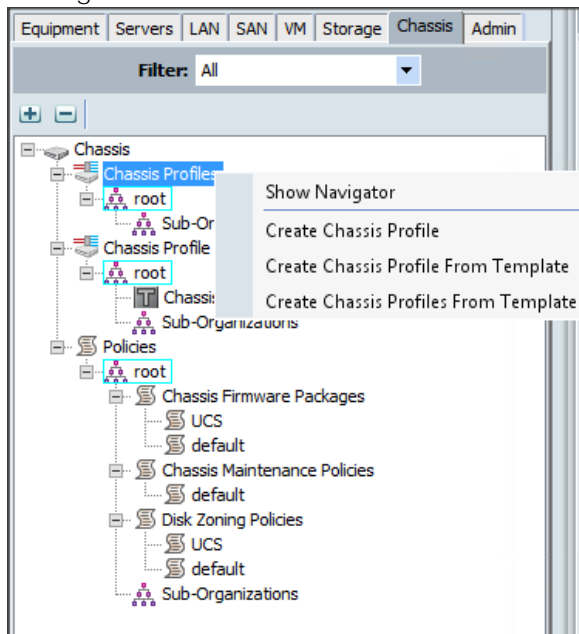
4. From the Disk Zoning Policy drop down list choose UCS and click Finish. (Figure 28)

Figure 28 Disk Zoning Policy



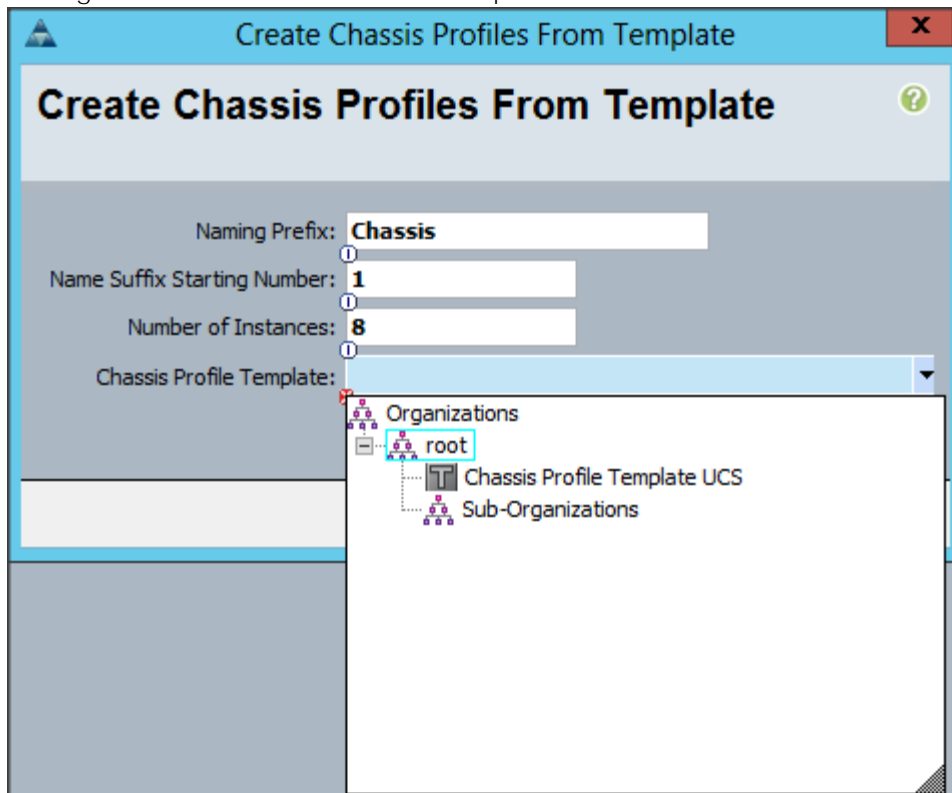
5. Right click on the Chassis Profile and click “Create Chassis Profiles from Template” (Figure 29).

Figure 29 Create Chassis Profile from Templates



6. Enter Chassis as the Naming Prefix, Number of instances is “8” and from the Chassis Profile Template drop down list choose “Chassis Profile Template UCS” and click OK. (Figure 30)

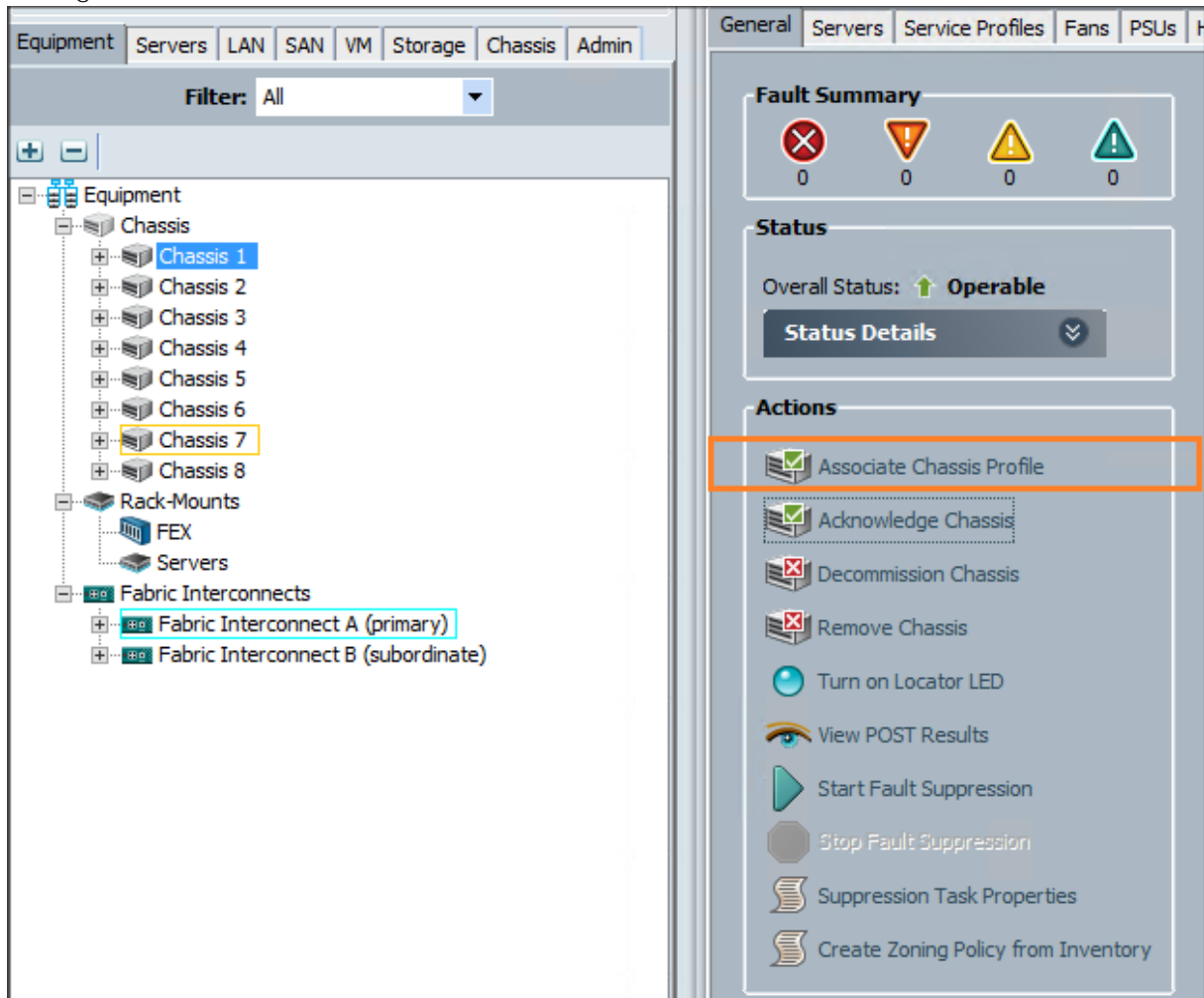
Figure 30 Chassis Profile Template UCS



Associating Chassis Profile to Individual Chassis.

1. On the Cisco UCS Manager UI select Equipment tab. Under Equipment expand chassis.
2. Select the chassis and click Associate Chassis Profile. (Figure 31)

Figure 31 Associate Chassis Profile



3. Select "Chassis Profile Chassis 1" and click "OK". (Figure 32)

Figure 32 Associate Chassis Profile

Associate Chassis Profile

Select an existing chassis profile to associate with the selected chassis.

Chassis Profiles

☒ Available Chassis Profiles ☐ All Chassis Profiles

Select	Name	Org	Assoc State
<input checked="" type="radio"/>	Chassis Profile Chassis1	org-root	Unassociated
<input type="radio"/>	Chassis Profile Chassis2	org-root	Unassociated
<input type="radio"/>	Chassis Profile Chassis3	org-root	Unassociated
<input type="radio"/>	Chassis Profile Chassis4	org-root	Unassociated
<input type="radio"/>	Chassis Profile Chassis5	org-root	Unassociated
<input type="radio"/>	Chassis Profile Chassis6	org-root	Unassociated
<input type="radio"/>	Chassis Profile Chassis7	org-root	Unassociated
<input type="radio"/>	Chassis Profile Chassis8	org-root	Unassociated

OK Cancel

4. Repeat steps 2 and 3 for the rest of the chassis.
5. Once the chassis profile is associated, only 28 disks will be assigned to each server node.
6. To verify that, go to Equipment→Chassis→Chassis 1→Server 2. Click on the Inventory→Storage→Disks. Expand Storage controller SAS 1.

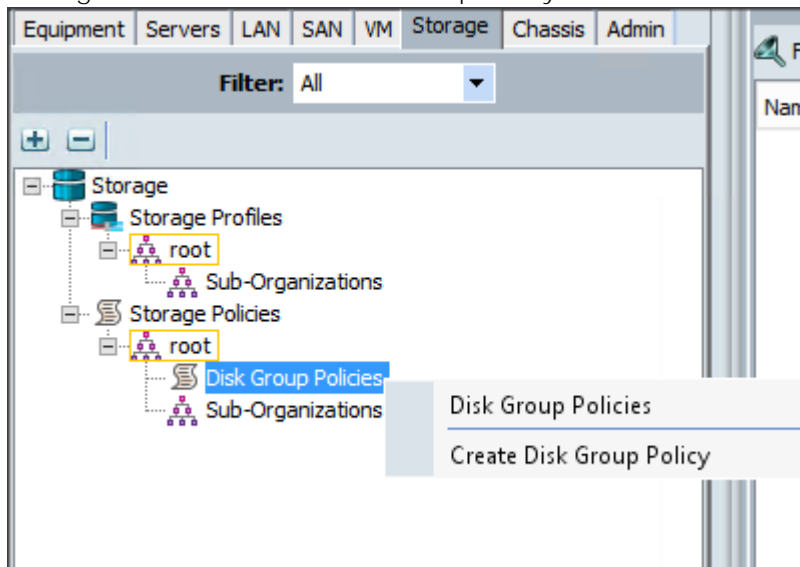
Figure 33 Storage Controller SAS 1

General Inventory Virtual Machines Installed Firmware CIMC Sessions SEL Logs VIF Paths Faults Events FSM Health Statistics Temperatures Power								
Motherboard CIMC CPUs Memory Adapters HBAs NICs iSCSI vNICs Storage GPUs Security								
Controller LUNs Disks								
+ - Filter Export Print								
Name	Size (MB)	Serial	Operability	Drive State	Presence	Technology	Bootable	
Storage Controller PCH 1								
Storage Controller SAS 1								
Disk 1	3814697	Z1223H9T0000C41074VE	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 2	3814697	Z1223HCM0000C4102ZYU	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 3	3814697	Z1211XV3N0000C408CEKU	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 4	3814697	Z1223D0Q0000C4101LSG	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 5	3814697	Z1223HSN0000C41073RB	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 6	3814697	Z121XFGF0000C4095PFD	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 7	3814697	Z121XFKM0000C408KFEW	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 8	3814697	Z1223HQV0000C41073E9	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 9	3814697	Z1220SL30000C4099T1F	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 10	3814697	Z1223HQD0000C41075TF	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 11	3814697	Z1223HVX0000C41073ER	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 12	3814697	Z1223HEB0000C41073YY	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 13	3814697	Z121X6YB0000C408KF7R	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 14	3814697	Z1221GC30000C40967HM	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 15	3814697	Z127P8EE0000C5252QSY	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 16	3814697	Z1223HPG0000C41076PW	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 17	3814697	Z127P8D30000C525H2CT	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 18	3814697	Z127NFI20000C5256745	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 19	3814697	Z121XEX90000C409MAHV	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 20	3814697	Z1223H0E0000C41074PG	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 21	3814697	Z121XGP20000C408LJ1D	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 22	3814697	Z127NICYC0000C52552EY	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 23	3814697	Z127P9S10000C525KH89	Operable	Unconfigured Good	Equipped	HDD	False	
Disk 24	3814697	Z121FNE50000C403T3VD	Operable	Unconfigured Good	Equipped	HDD	False	

Creating a Storage Profile for Boot Drives

1. Go to Storage and expand Storage→Storage Policies. Right click on Disk Group Policies and click Create Disk Group Policies. (Figure 34)

Figure 34 Create Disk Group Policy



2. In the Create Disk Policy window, configure the following parameters and click OK. (Figure 35)
 - a. Name = "Boot_SSD"

- b. RAID Level = RAID 1 Mirrored
- c. Disk Group Configuration=Automatic
- d. Number of Drives=2
- e. Drive Type= SSD
- f. Use Remaining Disks = checked
- g. Strip Size = 64 KB
- h. Access Policy = Platform Default
- i. Read Policy = Read Ahead
- j. Write Cache Policy = Always Write Back
- k. IO Policy and Drive Cache = Platform Default

Figure 35 Create Disk Group Policy

Create Disk Group Policy

Name:

Description:

RAID Level:

☒ Disk Group Configuration (Automatic)
 ☐ Disk Group Configuration (Manual)

Disk Group Configuration (Automatic)

Number of drives: [0-60]

Drive Type: ☐ Unspecified ☐ HDD ☒ SSD

Number of Dedicated Hot Spares: [0-60]

Number of Global Hot Spares: [0-60]

Min Drive Size (GB): [0-10240]

Use Remaining Disks: ☒

Virtual Drive Configuration

Strip Size (KB):

Access Policy:

Read Policy: ☐ Platform Default ☒ Read Ahead ☐ Normal

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

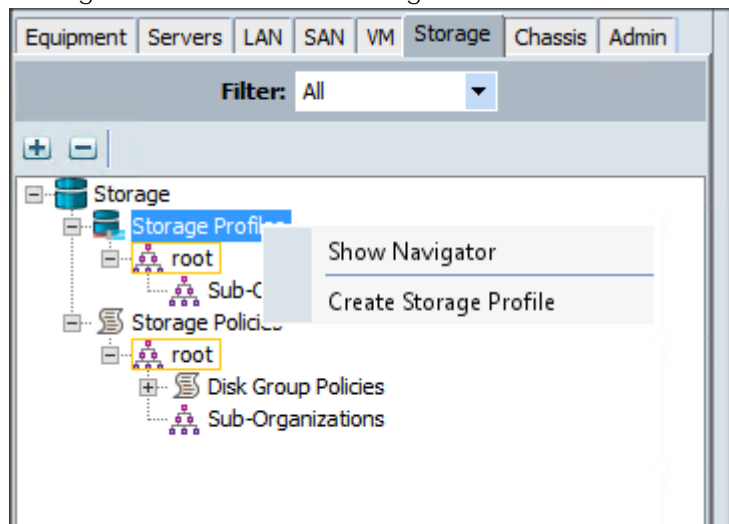
IO Policy: ☒ Platform Default ☐ Direct ☐ Cached

Drive Cache: ☒ Platform Default ☐ No Change ☐ Enable ☐ Disable

OK Cancel

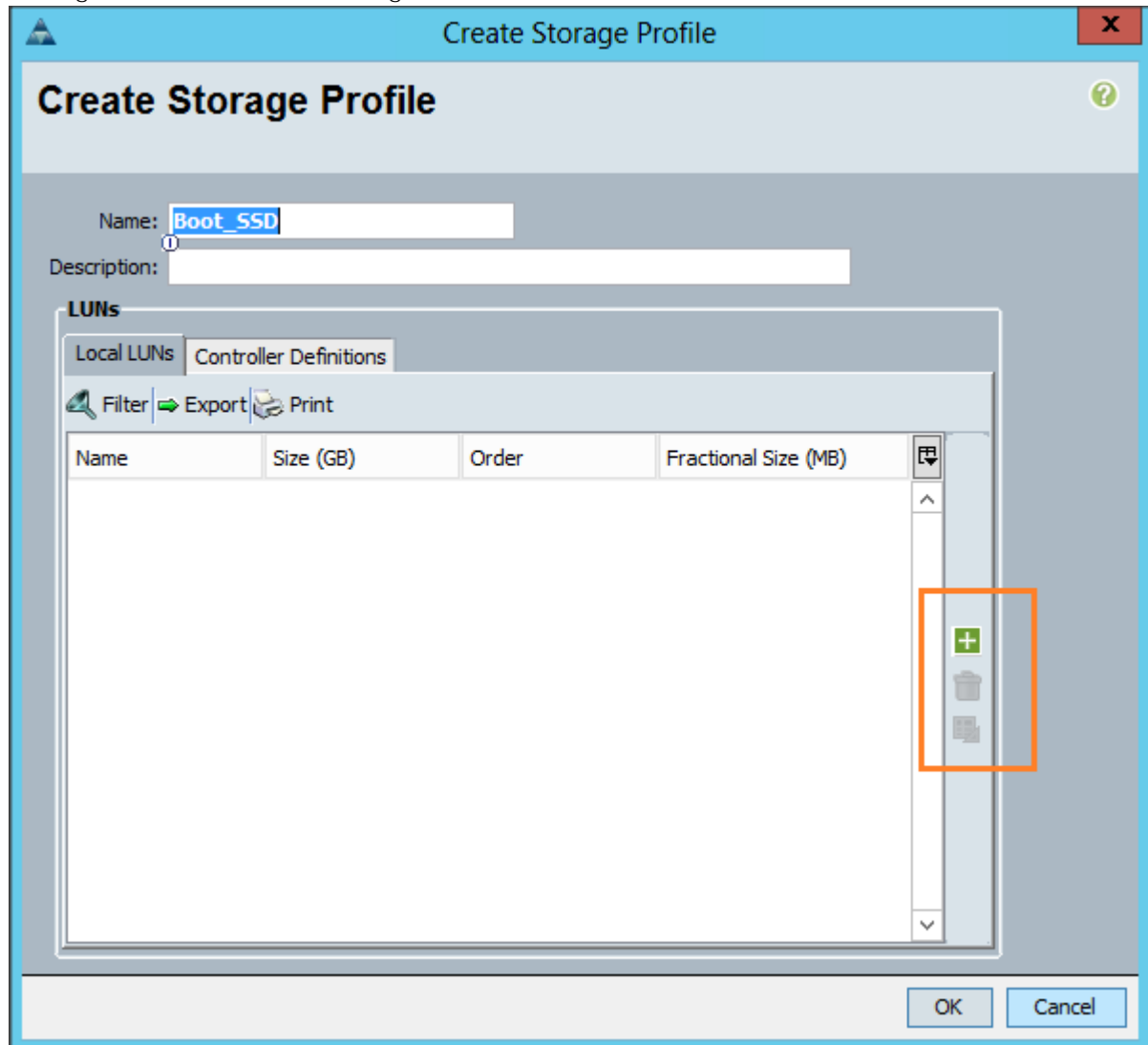
- Click on the Storage tab. Right click on Storage Profile and click Create Storage Profile. (Figure 36)

Figure 36 Create Storage Profile



4. Enter "Boot_SSD" in the name field. Under Local LUNs click "+" to add local lun. (Figure 37)

Figure 37 Create Storage Profile



5. In Create Local LUN window, enter the name “Boot_SSD”.
6. Check the “Expand to Available” checkbox to use all available space.
7. Under Select Disk Group Configuration drop down list choose “Boot_SSD” created earlier and click “OK” and “OK” again to complete the configuration.

Figure 38 Create Local LUN

Create Local LUN

☒ Create Local LUN
 ☐ Prepare Claim Local LUN

Name:

Size (GB): [0-102400]

Fractional Size (MB):

Order: [0-64]

Auto Deploy: ☒ Auto Deploy ☐ No Auto Deploy

Expand To Available: ☒

Select Disk Group Configuration: + Create Disk Group Policy

Domain Policies

Boot_SSD

OK Cancel

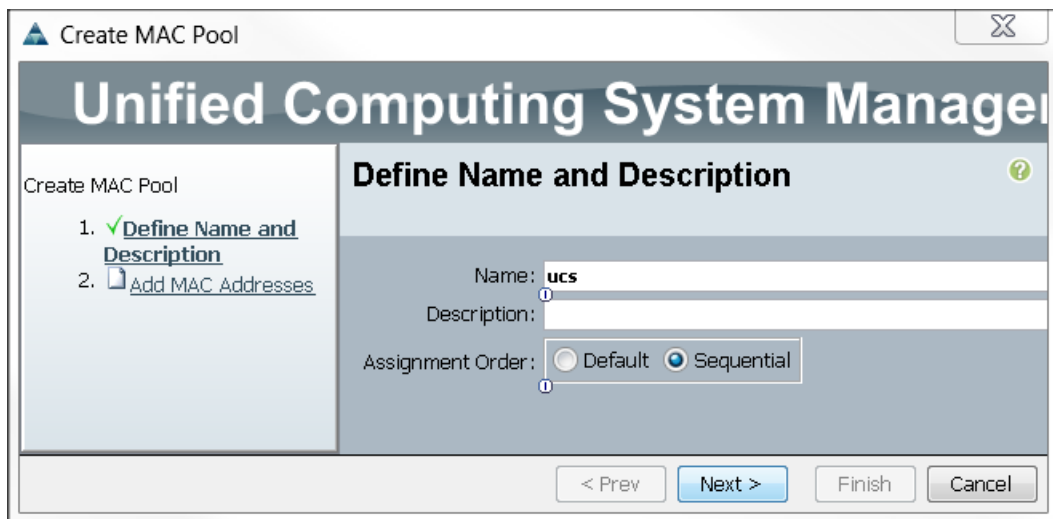
Creating Pools for Service Profile Templates

Creating MAC Address Pools

To create MAC address pools, complete the following steps:

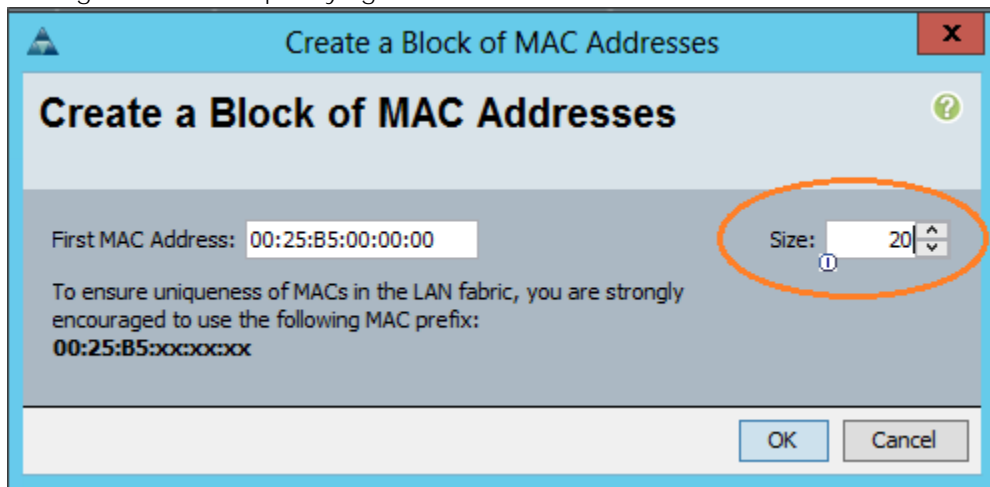
1. Select the LAN tab on the left of the window.
2. Select Pools > root.
3. Right-click MAC Pools under the root organization.
4. Select Create MAC Pool to create the MAC address pool. Enter ucs for the name of the MAC pool.
5. (Optional) Enter a description of the MAC pool.
6. Select Assignment Order Sequential.
7. Click Next.
8. Click Add.
9. Specify a starting MAC address.
10. Specify a size of the MAC address pool, which is sufficient to support the available server resources.

11. Click OK.



The image shows a dialog box titled "Create MAC Pool" from the "Unified Computing System Manager". The dialog has a sidebar on the left with two steps: "1. Define Name and Description" (marked with a green checkmark) and "2. Add MAC Addresses". The main area is titled "Define Name and Description" and contains the following fields: "Name" with the value "ucs", "Description" (empty), and "Assignment Order" with radio buttons for "Default" and "Sequential" (selected). At the bottom are buttons for "< Prev", "Next >", "Finish", and "Cancel".

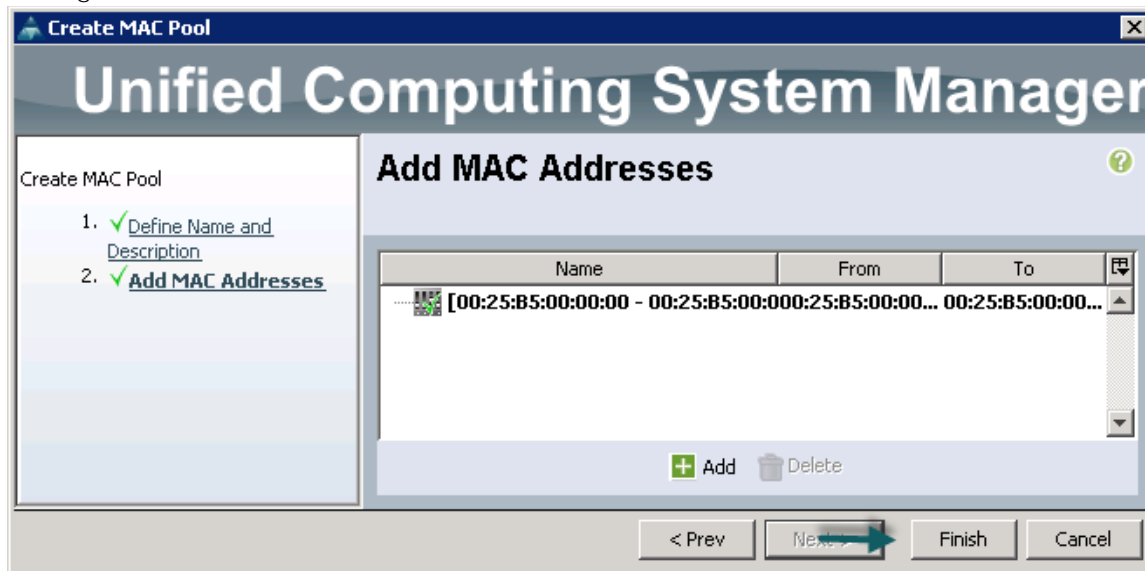
Figure 39 Specifying first MAC Address and Size



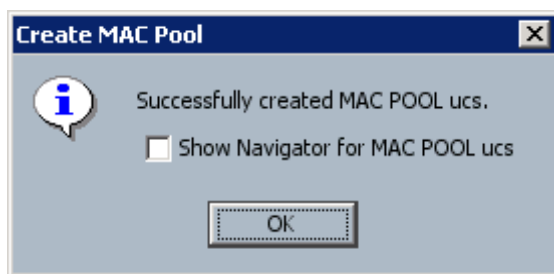
The image shows a dialog box titled "Create a Block of MAC Addresses". It contains the following fields: "First MAC Address" with the value "00:25:B5:00:00:00" and "Size" with a value of "20". The "Size" field is circled in orange. Below these fields is a text block: "To ensure uniqueness of MACs in the LAN fabric, you are strongly encouraged to use the following MAC prefix: 00:25:B5:xx:xx:xx". At the bottom are "OK" and "Cancel" buttons.

12. Click Finish.

Figure 40 Add MAC Addresses



13. When the message box displays, click OK.



Creating a Server Pool

A server pool contains a set of servers. These servers typically share the same characteristics. Those characteristics can be their location in the chassis, or an attribute such as server type, amount of memory, local storage, type of CPU, or local drive configuration. You can manually assign a server to a server pool, or use server pool policies and server pool policy qualifications to automate the assignment.

To configure the server pool within the Cisco UCS Manager GUI, complete the following steps:

1. Select the **Servers** tab in the left pane in the UCS Manager GUI.
2. Select **Pools > root**.
3. Right-click the **Server Pools**.
4. Select **Create Server Pool**.
5. Enter your required name (**ucs**) for the Server Pool in the name text box.
6. (Optional) enter a description for the organization.

- Click **Next >** to add the servers.

The screenshot shows the 'Unified Computing System Manager' window with the 'Create Server Pool' wizard. The wizard has two steps: 1. **Set Name and Description** (current step, marked with a green checkmark) and 2. **Add Servers** (marked with a document icon). The main area is titled 'Set Name and Description' and contains two text input fields: 'Name:' with the value 'ucs' and 'Description:' which is empty. At the bottom right, there are four buttons: '< Prev', 'Next >', 'Finish', and 'Cancel'.

- Select all the Cisco UCS S3260 Storage Servers to be added to the server pool that was previously created (ucs), then click **>>** to add them to the pool.
- Click **Finish**.
- Click **OK** and then click **Finish**.



Creating Policies for Service Profile Templates

Creating Host Firmware Package Policy

Firmware management policies allow the administrator to select the corresponding packages for a given server configuration. These include adapters, BIOS, board controllers, FC adapters, HBA options, and storage controller properties as applicable.

To create a firmware management policy for a given server configuration using the Cisco UCS Manager GUI, complete the following steps:

1. Select the `servers` tab in the left pane in the UCS Manager GUI.
2. Select Policies > root.
3. Right-click Host Firmware Packages.
4. Select Create Host Firmware Package.
5. Enter the required Host Firmware package name (`ucs`).
6. Select `simple` radio button to configure the Host Firmware package.
7. Select the appropriate Rack package that has been installed.

8. Click OK to complete creating the management firmware package
9. Click OK.

Create Host Firmware Package

Name:

Description:

How would you like to configure the Host Firmware Package? ☒ Simple ☐ Advanced

Blade Package:

Rack Package:

Excluded Components:

- ☐ Adapter
- ☐ BIOS
- ☐ CIMC
- ☐ Board Controller
- ☐ Flex Flash Controller
- ☐ GPUs
- ☐ FC Adapters
- ☐ HBA Option ROM
- ☐ Host NIC
- ☐ Host NIC Option ROM
- ☒ Local Disk
- ☐ PSU
- ☐ SAS Expander Regular Firmware
- ☐ SAS Expander
- ☐ Storage Controller
- ☐ Storage Controller Onboard Device
- ☐ Storage Controller Onboard Device Cpld
- ☐ Storage Device Driver

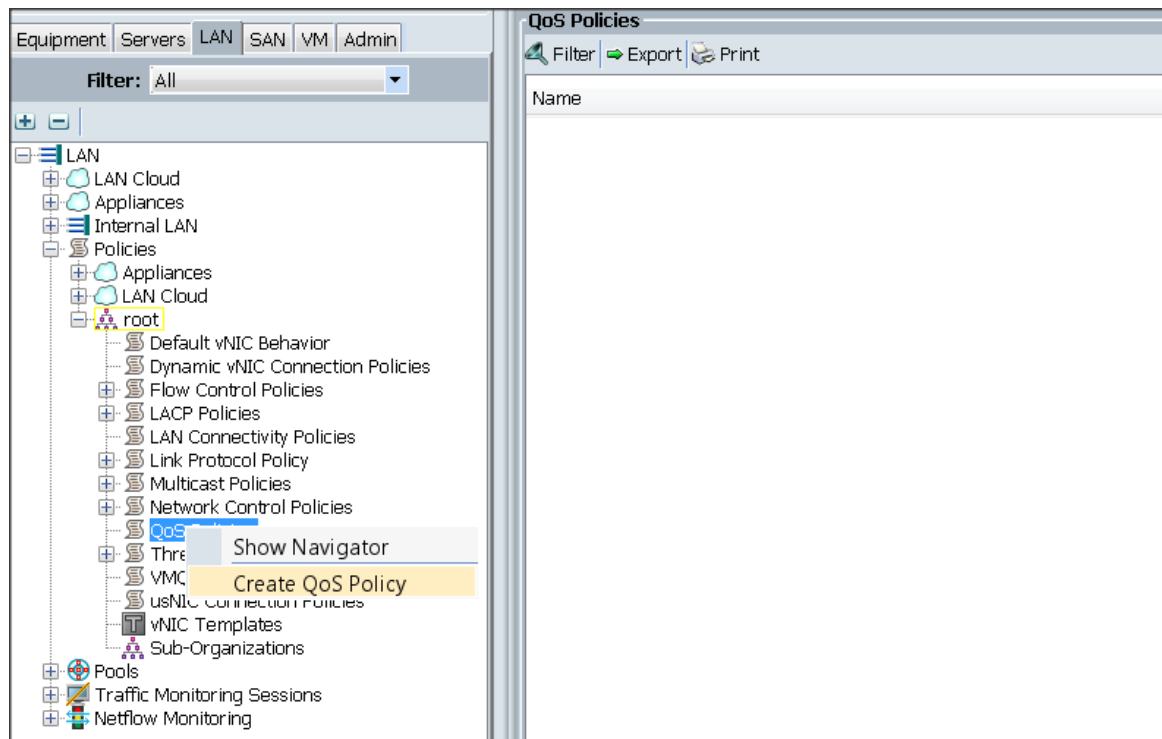
OK Cancel

Creating QoS Policies

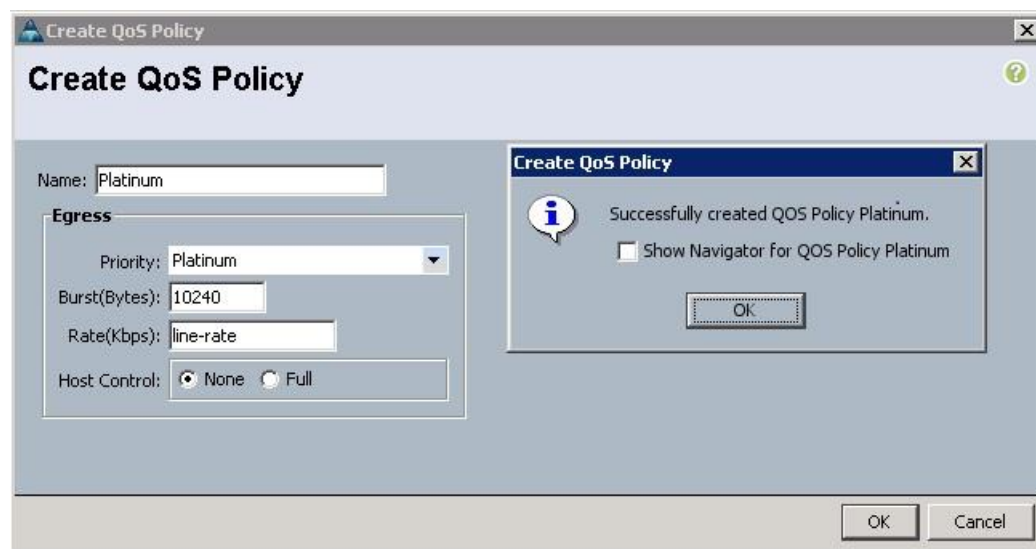
To create the QoS policy for a given server configuration using the Cisco UCS Manager GUI, complete the following steps:

Platinum Policy

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



5. Enter `Platinum` as the name of the policy.
6. Select `Platinum` from the drop down menu.
7. Keep the `Burst (Bytes)` field set to default (10240).
8. Keep the `Rate (Kbps)` field set to default (line-rate).
9. Keep `Host Control` radio button set to default (none).
10. Once the pop-up window appears, click `OK` to complete the creation of the Policy.



Setting Jumbo Frames

To set Jumbo frames and enable QoS, complete the following steps:

1. Select the LAN tab in the left pane in the UCSM GUI.
2. Select LAN Cloud > QoS System Class.
3. In the right pane, select the General tab
4. In the Platinum row, enter 9000 for MTU.
5. Check the Enabled Check box next to Platinum.
6. In the Best Effort row, select none for weight.
7. In the Fiber Channel row, select none for weight.
8. Click Save Changes.
9. Click OK.



Creating the Local Disk Configuration Policy

To create local disk configuration in the Cisco UCS Manager GUI, complete the following steps:

1. Select the Servers tab on the left pane in the UCS Manager GUI.
2. Go to Policies > root.
3. Right-click Local Disk Config Policies.
4. Select Create Local Disk Configuration Policy.
5. Enter `ucs` as the local disk configuration policy name.
6. Change the Mode to Any Configuration. Check the Protect Configuration box.
7. Keep the FlexFlash State field as default (Disable).
8. Keep the FlexFlash RAID Reporting State field as default (Disable).
9. Click OK to complete the creation of the Local Disk Configuration Policy.
10. Click OK.

Create Local Disk Configuration Policy

Name:

Description:

Mode:

Protect Configuration: ☒

If **Protect Configuration** is set, the local disk configuration is preserved if the service profile is disassociated with the server. In that case, a configuration error will be raised when a new service profile is associated with that server if the local disk configuration in that profile is different.

FlexFlash

FlexFlash State: ☒ Disable ☐ Enable

If **FlexFlash State** is disabled, SD cards will become unavailable immediately. Please ensure SD cards are not in use before disabling the FlexFlash State.

FlexFlash RAID Reporting State: ☒ Disable ☐ Enable

OK Cancel

Creating Server BIOS Policy

The BIOS policy feature in Cisco UCS automates the BIOS configuration process. The traditional method of setting the BIOS is manually, and is often error-prone. By creating a BIOS policy and assigning the policy to a server or group of servers, can enable transparency within the BIOS settings configuration.



Note: *BIOS settings* can have a significant performance impact, *depending* on the workload and the *applications*. 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.

To create a server BIOS policy using the Cisco UCS Manager GUI, complete the following steps:

1. Select the `Servers` tab in the left pane in the Cisco UCS Manager GUI.
2. Select `Policies > root`.
3. Right-click `BIOS Policies`.
4. Select `Create BIOS Policy`.
5. Enter your preferred BIOS policy name (`ucs`).
6. Change the BIOS settings as shown in the following figures.
7. Only changes that need to be made are in the `Processor` and `RAS Memory settings`.

Processor

Turbo Boost:	<input type="radio"/> disabled <input checked="" type="radio"/> enabled <input type="radio"/> Platform Default
Enhanced Intel Speedstep:	<input type="radio"/> disabled <input checked="" type="radio"/> enabled <input type="radio"/> Platform Default
Hyper Threading:	<input type="radio"/> disabled <input checked="" type="radio"/> enabled <input type="radio"/> Platform Default
Core Multi Processing:	all
Execute Disabled Bit:	<input type="radio"/> disabled <input type="radio"/> enabled <input checked="" type="radio"/> Platform Default
Virtualization Technology (VT):	<input checked="" type="radio"/> disabled <input type="radio"/> enabled <input type="radio"/> Platform Default
Hardware Pre-fetcher:	<input type="radio"/> disabled <input checked="" type="radio"/> enabled <input type="radio"/> Platform Default
Adjacent Cache Line Pre-fetcher:	<input type="radio"/> disabled <input checked="" type="radio"/> enabled <input type="radio"/> Platform Default
DCU Streamer Pre-fetch:	<input type="radio"/> disabled <input checked="" type="radio"/> enabled <input type="radio"/> Platform Default
DCU IP Pre-fetcher:	<input type="radio"/> disabled <input checked="" type="radio"/> enabled <input type="radio"/> Platform Default
Direct Cache Access:	<input type="radio"/> disabled <input checked="" type="radio"/> enabled <input type="radio"/> auto <input type="radio"/> Platform Default
Processor C State:	<input checked="" type="radio"/> disabled <input type="radio"/> enabled <input type="radio"/> Platform Default
Processor C1E:	<input checked="" type="radio"/> disabled <input type="radio"/> enabled <input type="radio"/> Platform Default
Processor C3 Report:	disabled
Processor C6 Report:	<input checked="" type="radio"/> disabled <input type="radio"/> enabled <input type="radio"/> Platform Default
Processor C7 Report:	disabled
Processor CMCi:	<input type="radio"/> enabled <input type="radio"/> disabled <input checked="" type="radio"/> Platform Default
CPU Performance:	enterprise
Max Variable MTRR Setting:	<input type="radio"/> auto-max <input type="radio"/> 8 <input checked="" type="radio"/> Platform Default
Local X2 APIC:	<input type="radio"/> xapic <input type="radio"/> x2apic <input type="radio"/> auto <input checked="" type="radio"/> Platform Default

Power Technology: **performance**

Energy Performance: **performance**

Frequency Floor Override: ☐ disabled ☒ enabled ☐ Platform Default

P-STATE Coordination: ☒ hw-all ☐ sw-all ☐ sw-any ☐ Platform Default

DRAM Clock Throttling: **performance**

Channel Interleaving: Platform Default

Rank Interleaving: Platform Default

Demand Scrub: ☒ disabled ☐ enabled ☐ Platform Default

Patrol Scrub: ☒ disabled ☐ enabled ☐ Platform Default

Altitude: Package C State Limit: **auto** **auto**

CPU Hardware Power Management: ☐ disabled ☐ hwpm-native-mode ☐ hwpm-oob-mode ☒ Platform Default

Energy Performance Tuning: ☒ os ☐ bios ☐ Platform Default

Workload Configuration: ☐ balanced ☒ io-sensitive ☐ Platform Default

Create BIOS Policy

Unified Computing System Manager

RAS Memory

1. ☒ Main
2. ☒ Processor
3. ☒ Intel Directed IO
4. ☒ **RAS Memory**
5. ☐ Serial Port
6. ☐ USB
7. ☐ PCI
8. ☐ QPI
9. ☐ LOM and PCIe Slots
10. ☐ Trusted Platform
11. ☐ Graphics Configuration
12. ☐ Boot Options
13. ☐ Server Management

Memory RAS Config: **maximum-performance**

NUMA: ☐ disabled ☒ enabled ☐ Platform Default

LV DDR Mode: ☐ power-saving-mode ☒ performance-mode ☐ auto ☐ Platform Default

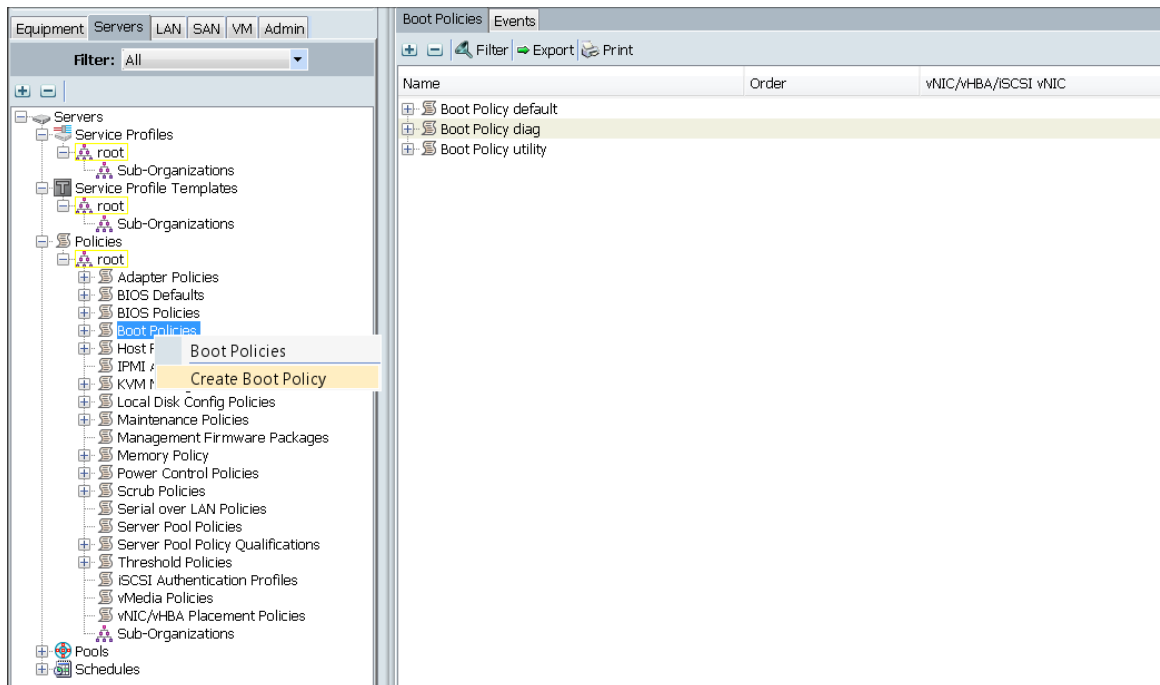
DRAM Refresh Rate: **1x**

DDR3 Voltage Selection: ☐ ddr3-1500mv ☐ ddr3-1350mv ☒ Platform Default

Creating the Boot Policy

To create boot policies within the Cisco UCS Manager GUI, complete the following steps:

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



5. Enter ucs as 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 and select Add Local Lun.
11. In the Add Local LUN Image Path window, select Primary and enter the Name "Boot_SSD" that was created earlier during storage profile creation step.



Note: The LUN name must match with the LUN name created earlier.

Figure 41 Add Local LUN Image Path

Add Local LUN Image Path

Type: ☒ Primary ☐ Secondary ☐ Any

LUN Name:

OK Cancel

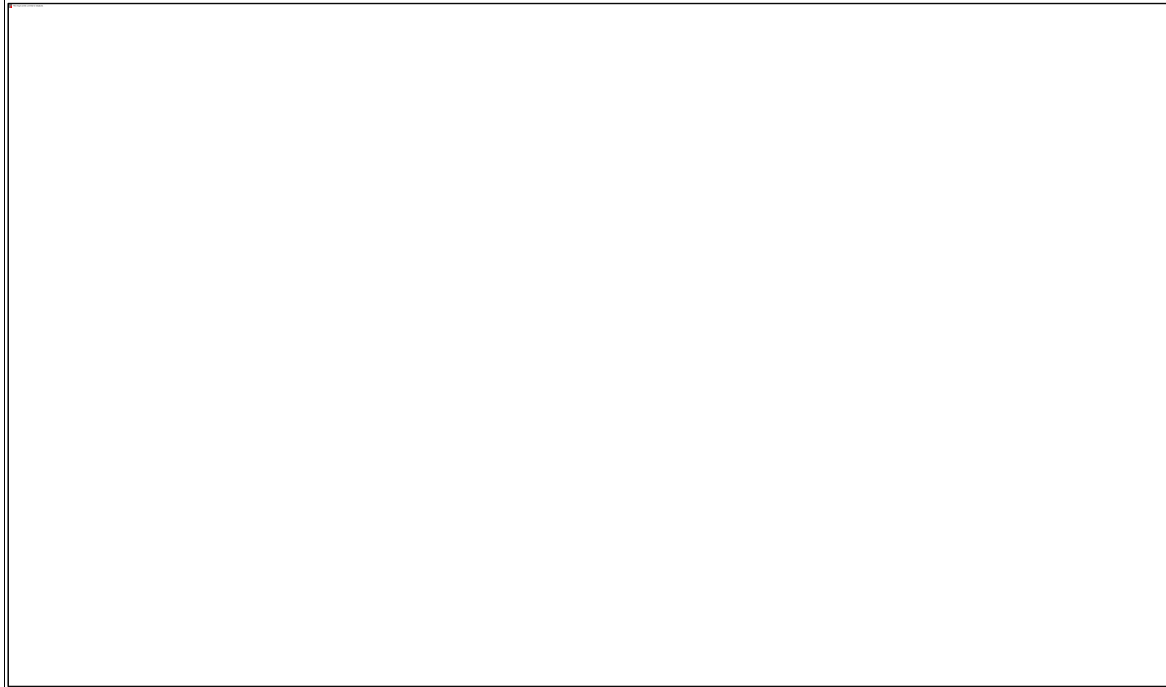
12. Expand Local Devices > Add CD/DVD and select Add Local CD/DVD
13. Expand vNICs and select Add LAN Boot and enter eth0.
14. Click OK to add the Boot Policy.
15. Click OK.

Boot Order						
Name	Order	vNIC/vHBA/ISCSI vNIC	Type	LUN Name	WWN	Slot Number
Local LUN	1					
Local Lun Image			Primary	Boot_SSD		
CD/DVD	2					
LAN	3					
LAN eth0		eth0	Primary			

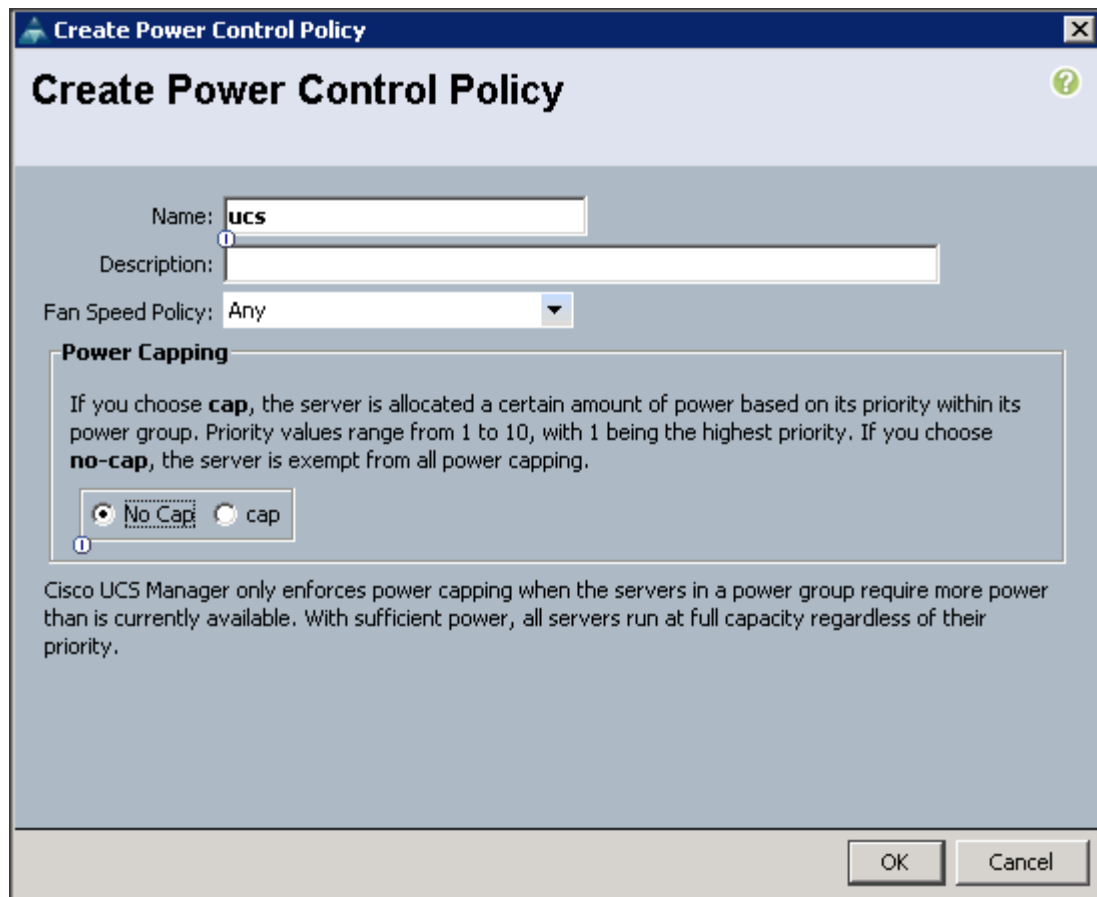
Creating Power Control Policy

To create Power Control policies within the Cisco UCS Manager GUI, complete the following steps:

1. Select the **Servers** tab in the left pane in the UCS Manager GUI.
2. Select **Policies > root**.
3. Right-click the **Power Control Policies**.
4. Select **Create Power Control Policy**.



5. Enter `ucs` as the Power Control policy name.
6. (Optional) enter a description for the boot policy.
7. Select Performance for Fan Speed Policy.
8. Select No cap for Power Capping selection.
9. Click `ok` to create the Power Control Policy.
10. Click `ok`.



Create Power Control Policy

Name:

Description:

Fan Speed Policy:

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

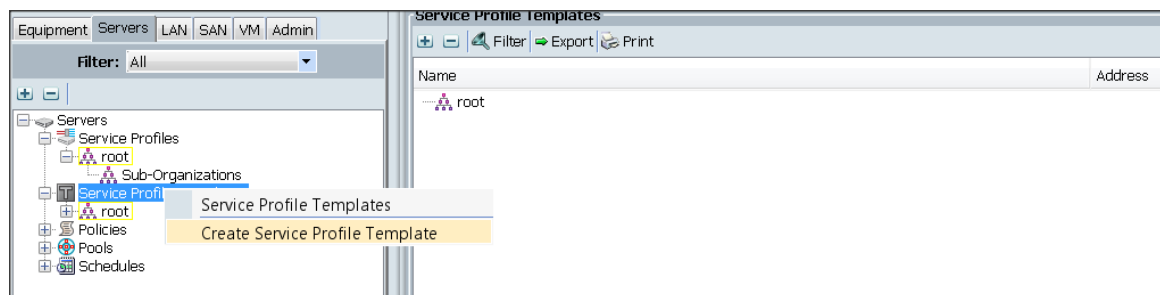
Cisco UCS Manager only enforces power capping when the servers in a power group require more power than is currently available. With sufficient power, all servers run at full capacity regardless of their priority.

OK Cancel

Creating a Service Profile Template

To create a Service Profile Template, complete the following steps:

1. Select the **Servers** tab in the left pane in the UCSM GUI.
2. Right-click **Service Profile Templates**.
3. Select **Create Service Profile Template**.



The Create Service Profile Template window appears.

To identify the service profile template, complete the following steps:

1. Name the service profile template as ucs. Select the Updating Template radio button.
2. In the UUID section, select Hardware Default as the UUID pool.
3. Click Next to continue to the next section.

Create Service Profile Template

Unified Computing System Manager

Create Service Profile Template

1. **Identify Service Profile Template**
2. Storage Provisioning
3. Networking
4. SAN Connectivity
5. Zoning
6. vNIC/vHBA Placement
7. vMedia Policy
8. Server Boot Order
9. Maintenance Policy
10. Server Assignment
11. Operational Policies

Identify Service Profile Template

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

Name:

The template will be created in the following organization. Its name must be unique within this organization.

Where: **org-root**

The template will be created in the following organization. Its name must be unique within this organization.

Type: ☐ Initial Template ☒ Updating Template

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

UUID

UUID Assignment:

The UUID assigned by the manufacturer will be used.
Note: This UUID will not be migrated if the service profile is moved to a new server.

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

< Prev Next > Finish Cancel

Configuring the Storage Provisioning for the Template

To configure storage policies, complete the following steps:

1. Go to Storage Profile Policy tab, and select Boot_SSD from the drop down list.

Create Service Profile Template

Unified Computing System Manager

Create Service Profile Template

1. ☒ Identify Service Profile Template
2. ☒ **Storage Provisioning**
3. ☐ Networking
4. ☐ SAN Connectivity
5. ☐ Zoning
6. ☐ vNIC/vHBA Placement
7. ☐ vMedia Policy
8. ☐ Server Boot Order
9. ☐ Maintenance Policy
10. ☐ Server Assignment
11. ☐ Operational Policies

Storage Provisioning

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

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

Storage Profile: [+ Create Storage Profile](#)

Name: **Boot_SSD**

Description:

LUNs

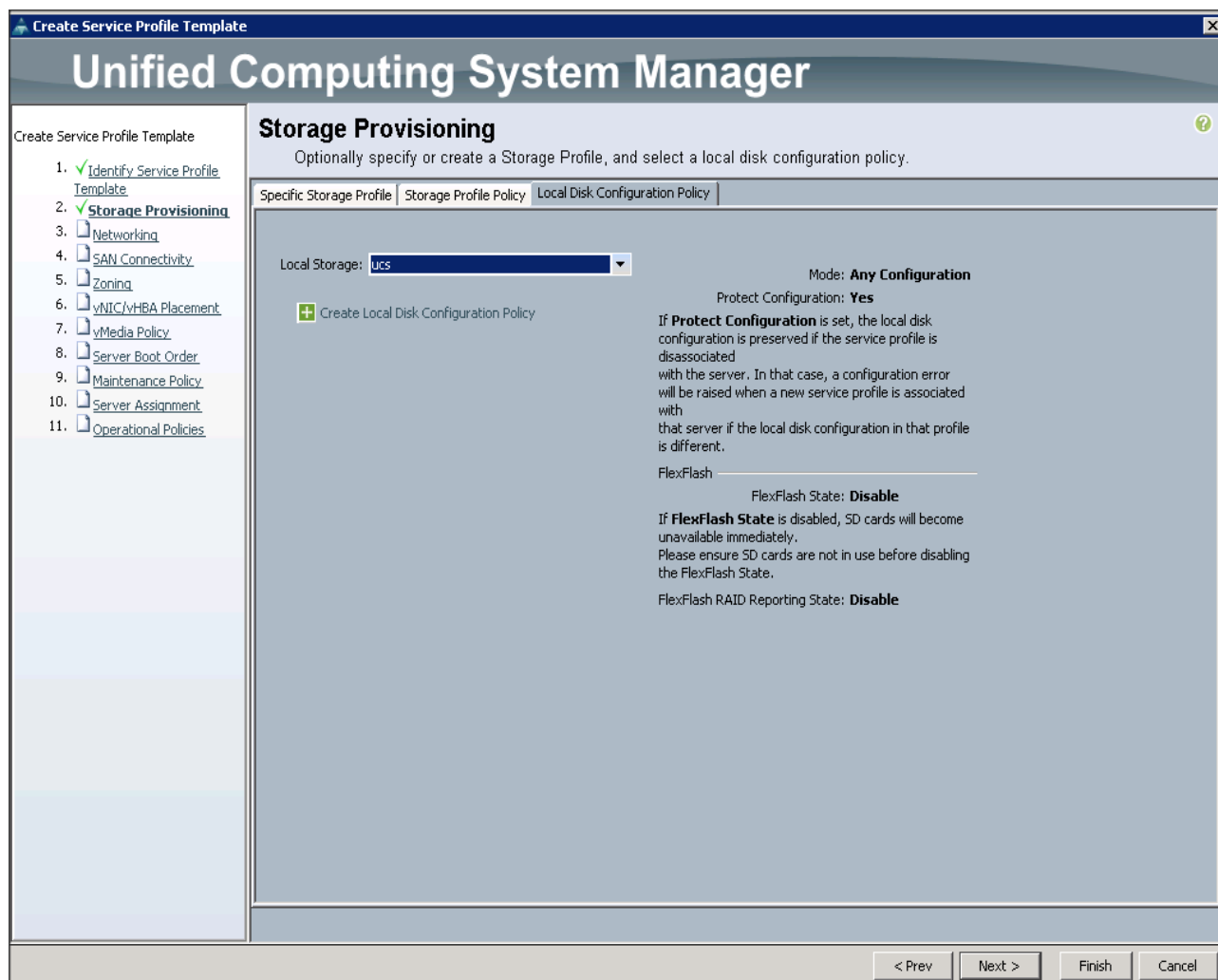
Local LUNs | Controller Definitions

[Filter](#) [Export](#) [Print](#)

Name	Size (GB)	Order	Fractional Size (MB)
Boot_SSD	1	Not Applicable	0

< Prev Next > Finish Cancel

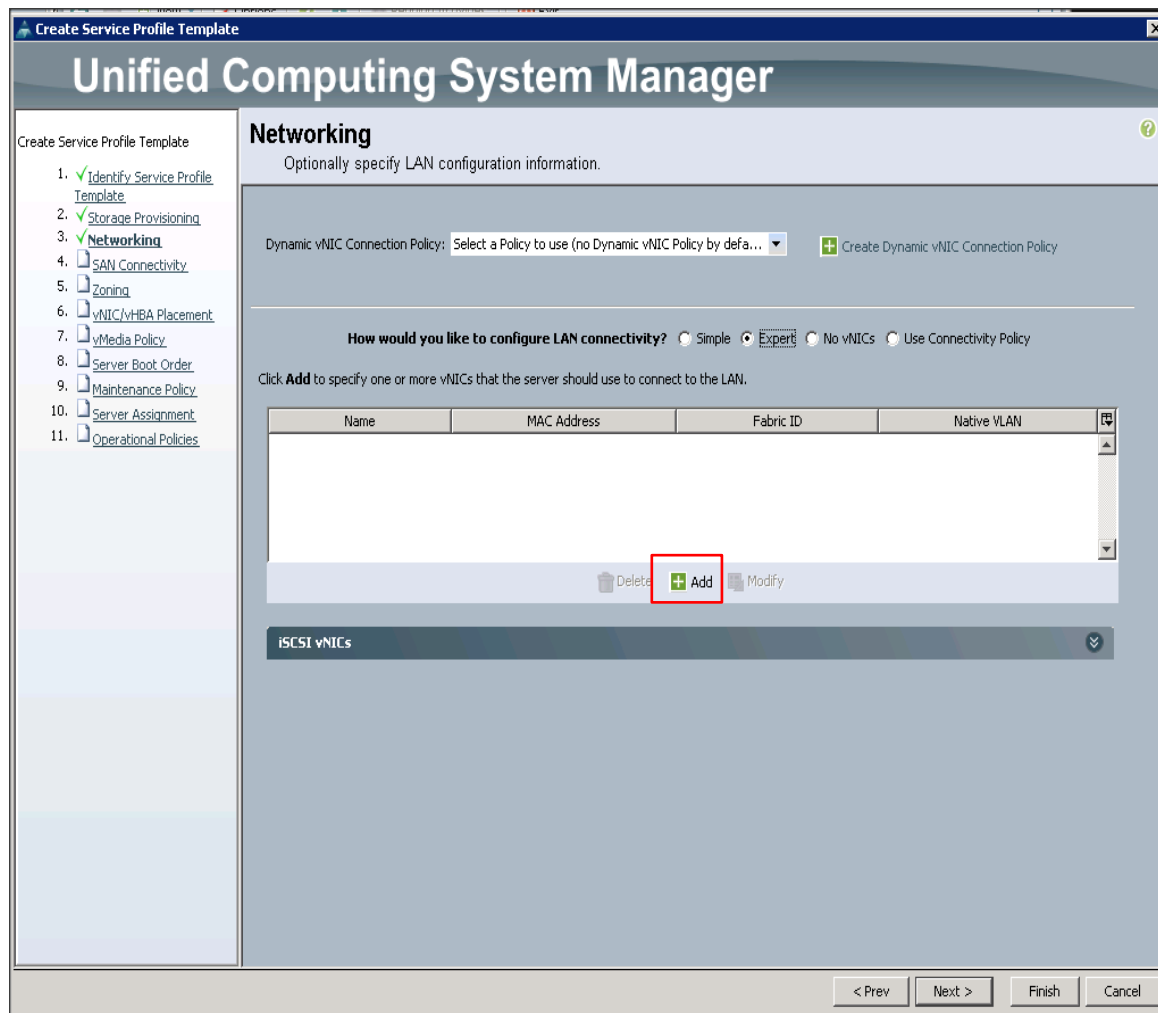
2. Go to the Local Disk Configuration Policy tab, and select ucs for the Local Storage.
3. Click Next to continue to the next section.



4. Click Next once the Networking window appears to go to the next section.

Configuring Network Settings for the Template

1. Keep the Dynamic vNIC Connection Policy field at the default.
2. Select the Expert radio button for the option how would you like to configure LAN connectivity?
3. Click Add to add a vNIC to the template.



4. The Create vNIC window displays. Name the vNIC `eth0`.
5. Select `ucs` in the Mac Address Assignment pool.
6. Select the Fabric `A` radio button and check the `Enable failover` check box for the Fabric ID.
7. Check the `VLAN76` check box for VLANs and select the `Native VLAN` radio button.
8. Select `MTU size` as `9000`.
9. Select adapter policy as `Linux`.
10. Select `QoS Policy` as `Platinum`.
11. Keep the `Network Control Policy` as `Default`.
12. Click `OK`.

Create vNIC

Create vNIC

Name:

1

Use vNIC Template: ☐

MAC Address

MAC Address Assignment:

▼

+

 Create MAC Pool

The MAC address will be automatically assigned from the selected pool.

Fabric ID: ☒ Fabric A ☐ Fabric B ☒ Enable Failover

1

VLAN in LAN cloud will take the precedence over the Appliance Cloud when there is a name clash.

VLANs

Filter

Export

Print

Select	Name	Native VLAN
<input type="checkbox"/>	default	<input type="radio"/>
<input checked="" type="checkbox"/>	vlan76	<input checked="" type="radio"/>

+

 Create VLAN

CDN Source: ☒ vNIC Name ☐ User Defined

MTU:

1

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.

Pin Group:

▼

+

 Create LAN Pin Group

Operational Parameters

Adapter Performance Profile

Adapter Policy:

1

+

 Create Ethernet Adapter Policy

QoS Policy:

1

+

 Create QoS Policy

Network Control Policy:

1

+

 Create Network Control Policy

Connection Policies

☒ Dynamic vNIC ☐ usNIC ☐ VMQ

Dynamic vNIC Connection Policy:

▼

+

 Create Dynamic vNIC Connection Policy

OK

Cancel

75

Configure eth1

1. Click `Add` to add a vNIC to the template.
2. The Create vNIC window displays. Name the vNIC `eth1`.
3. Select `ucs` in the Mac Address Assignment pool.
4. Select the `Fabric B` radio button and check the `Enable failover` check box for the Fabric ID.
5. Check the `VLAN77` check box for VLANs and select the `Native VLAN` radio button.
6. Select `MTU size` as `9000`.
7. Select adapter policy as `Linux`.
8. Select QoS Policy as `Platinum`.
9. Keep the Network Control Policy as `Default`.
10. Click `OK`.

Create vNIC

Name:

Use vNIC Template: ☐

MAC Address

MAC Address Assignment:

Create MAC Pool

The MAC address will be automatically assigned from the selected pool.

Fabric ID: ☐ Fabric A ☒ Fabric B ☒ Enable Failover

VLAN in LAN cloud will take the precedence over the Appliance Cloud when there is a name dash.

VLANs

Filter Export Print

Select	Name	Native VLAN
<input type="checkbox"/>	default	<input type="radio"/>
<input type="checkbox"/>	vlan76	<input type="radio"/>
<input checked="" type="checkbox"/>	vlan 77	<input checked="" type="radio"/>

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.

Pin Group:

Create LAN Pin Group

Operational Parameters

Adapter Performance Profile

Adapter Policy:

QoS Policy:

Network Control Policy:

Dynamic vNIC ☒ usNIC ☐ VMQ

Dynamic vNIC Connection Policy:

OK

Cancel

77

Create Service Profile Template

Unified Computing System Manager

Create Service Profile Template

1. ☒ Identify Service Profile Template
2. ☒ Storage Provisioning
3. ☒ **Networking**
4. ☒ SAN Connectivity
5. ☒ Zoning
6. ☒ vNIC/vHBA Placement
7. ☐ vMedia Policy
8. ☐ Server Boot Order
9. ☐ Maintenance Policy
10. ☐ Server Assignment
11. ☐ Operational Policies

Networking

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	A B		
Network vlan76			<input checked="" type="radio"/>	
vNIC eth 1		B A		
Network vlan77			<input checked="" type="radio"/>	

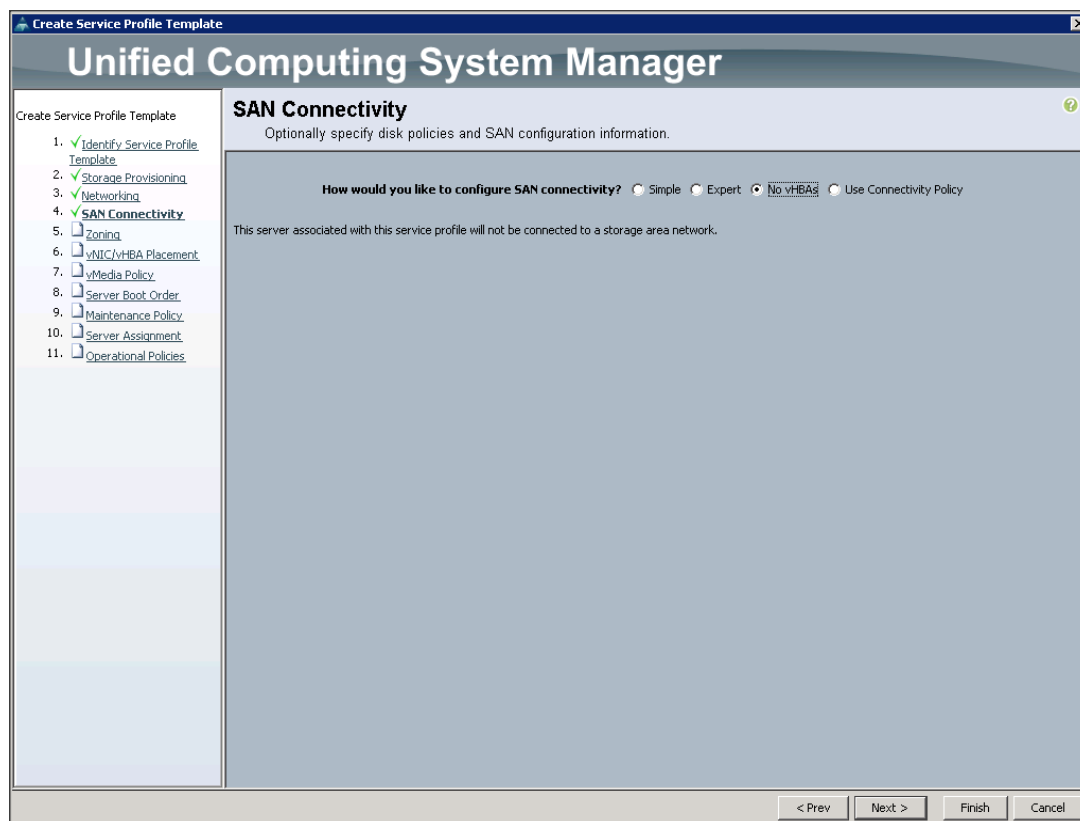
[Delete](#) [+ Add](#) [Modify](#)

iSCSI vNICs

< Prev Next > Finish Cancel

11. Click **Next** to continue with SAN Connectivity.

12. Select **no vHBAs** for How would you like to configure SAN Connectivity?



13. Click Next to continue with Zoning.

The screenshot shows the 'Create Service Profile Template' wizard in the Unified Computing System Manager. The 'Zoning' step is active, showing a warning about end-host mode and instructions for zoning configuration. The left sidebar lists the steps, with 'Zoning' highlighted. The main area contains two tables for selecting vHBA initiators and groups, with a 'Add To' button between them. The bottom navigation bar includes 'Prev', 'Next', 'Finish', and 'Cancel' buttons.

Create Service Profile Template

Unified Computing System Manager

Create Service Profile Template

1. ☒ Identify Service Profile Template
2. ☒ Storage Provisioning
3. ☒ Networking
4. ☒ SAN Connectivity
5. ☒ **Zoning**
6. ☐ vNIC/vHBA Placement
7. ☐ vMedia Policy
8. ☐ Server Boot Order
9. ☐ Maintenance Policy
10. ☐ Server Assignment
11. ☐ Operational Policies

Zoning

Specify zoning information

WARNING: Switch in end-host mode. In end-host mode, zoning configuration will NOT be applied.

Zoning configuration involves the following **steps**:

1. **Select** vHBA Initiator(s) (vHBAs are created on storage page)
2. **Select** vHBA Initiator Group(s)
3. **Add** selected Initiator(s) to selected Initiator Group(s)

Select vHBA Initiators

Name

>> Add To >>

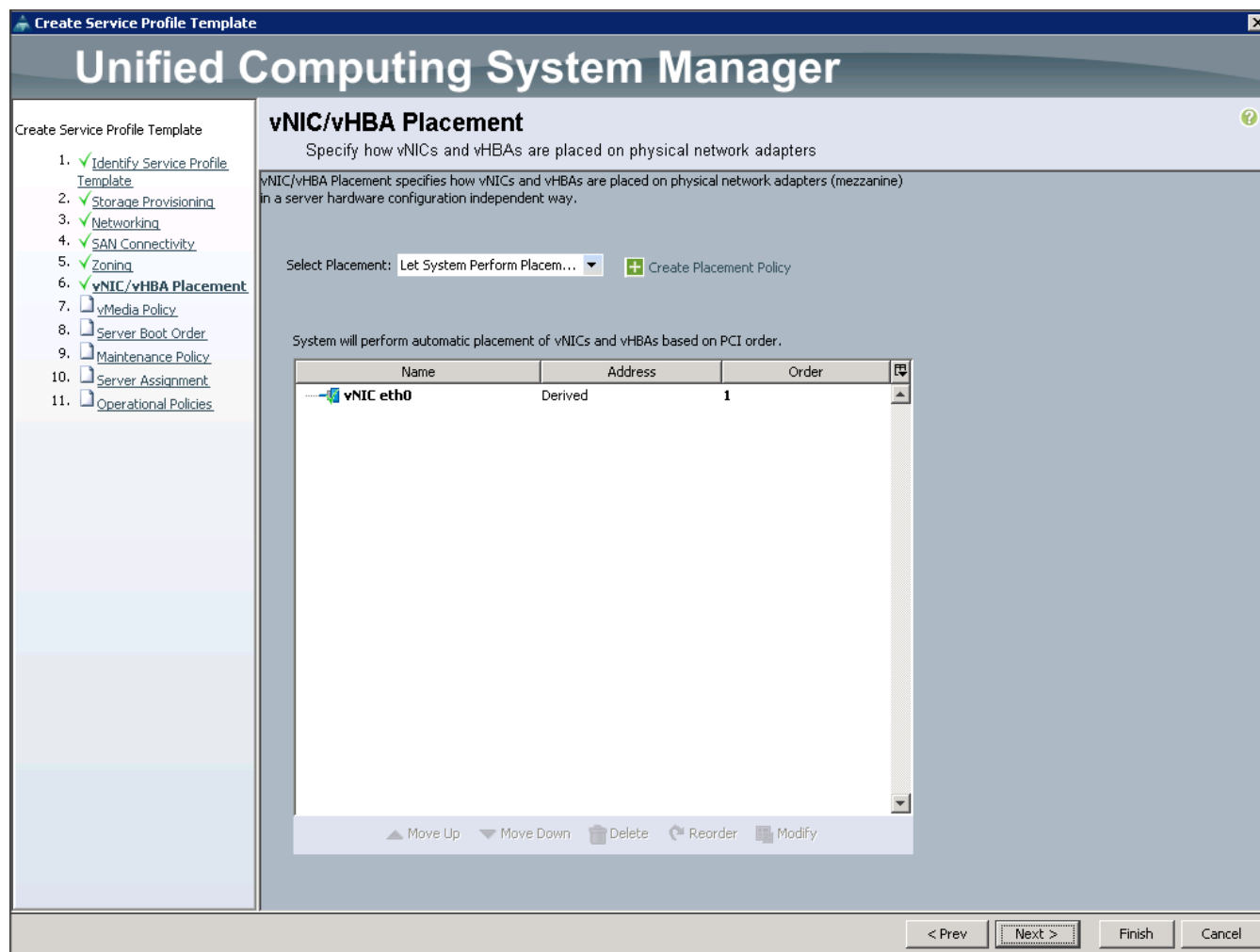
Select vHBA Initiator Groups

Name	Storage Connection Policy Name
------	--------------------------------

Delete Add Modify

< Prev Next > Finish Cancel

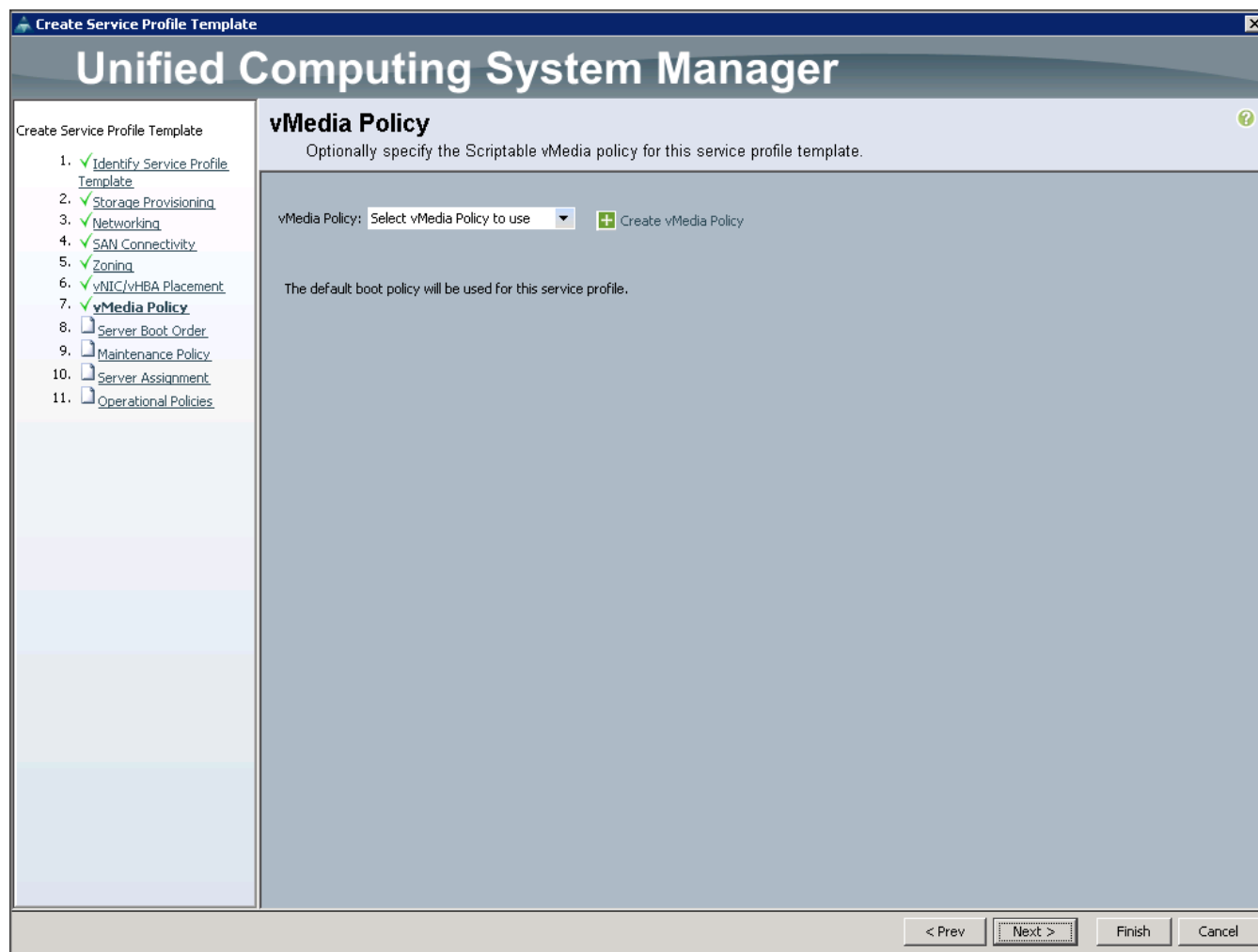
14. Click **Next** to continue with vNIC/vHBA placement.



15. Click **Next** to configure vMedia Policy.

Configuring the vMedia Policy for the Template

1. Click **Next** once the vMedia Policy window appears to go to the next section.



Configuring Server Boot Order for the Template

To set the boot order for the servers, complete the following steps:

1. Select ucs in the Boot Policy name field.
2. Review to make sure that all of the boot devices were created and identified.
3. Verify that the boot devices are in the correct boot sequence.
4. Click OK.
5. Click Next to continue to the next section.

Modify Boot Policy

Boot Policy: ucs + Create Boot Policy

Name: **ucs**

Description:

Reboot on Boot Order Change: **No**

Enforce vNIC/vHBA/iSCSI Name: **Yes**

Boot Mode: **Legacy**

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

Boot Order

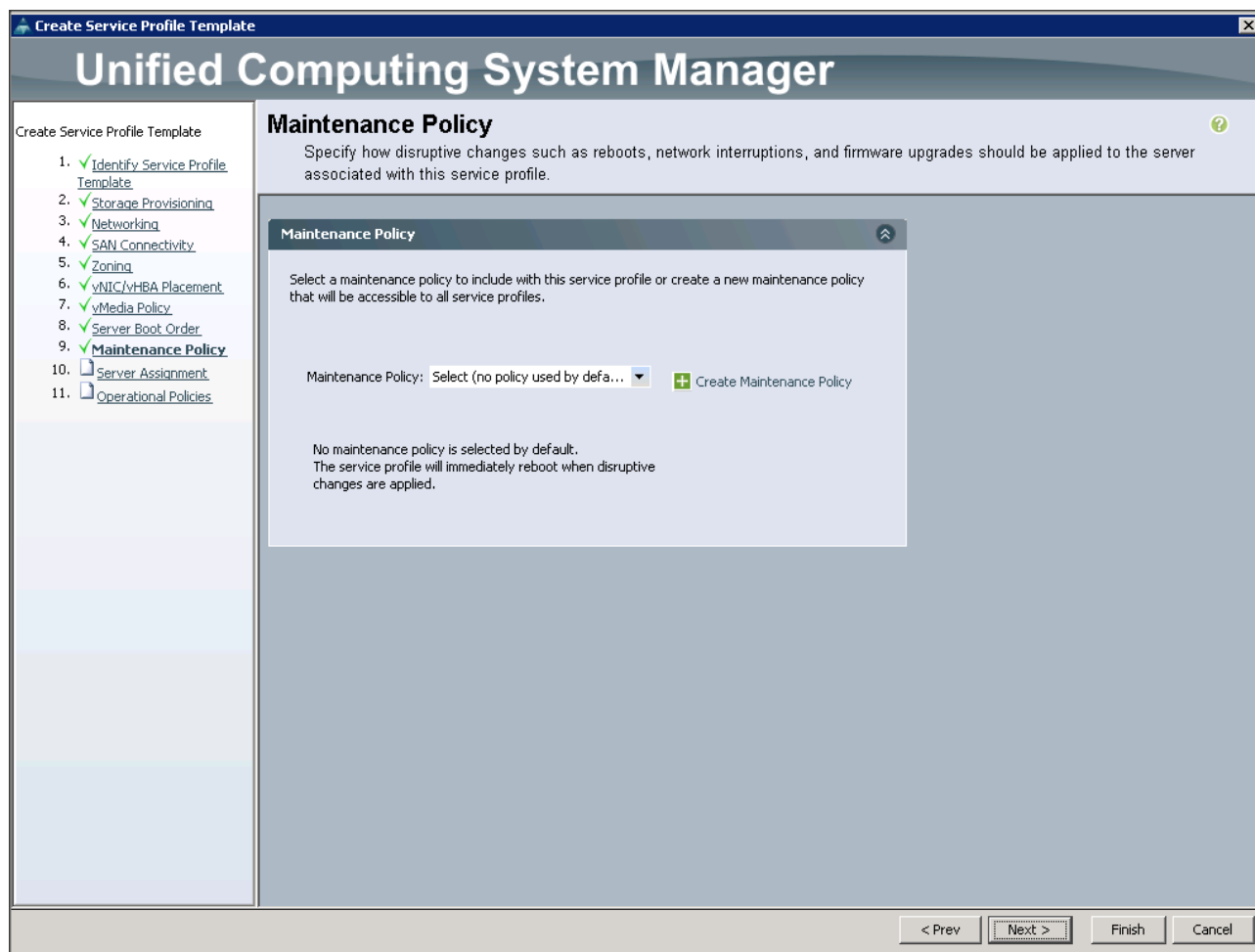
+ - Filter Export Print

Name	Order	vNIC/vHBA/iSCSI vNIC	Type	LUN Name	WWN	Slot Number	Boot Name	Boot Path	Description
Local LUN	1								
Local Lun Image			Primary	Boot_SSD					
CD/DVD	2								
LAN	3								
LAN eth0		eth0	Primary						

Create iSCSI vNIC Set iSCSI Boot Parameters Set Uefi Boot Parameters

OK Cancel

6. In the Maintenance Policy window, apply the maintenance policy.
7. Keep the Maintenance policy at no policy used by default. Click Next to continue to the next section.



Configuring Server Assignment for the Template

In the Server Assignment window, to assign the servers to the pool, complete the following steps:

1. Select ucs for the Pool Assignment field.
2. Select the power state to be Up.
3. Keep the Server Pool Qualification field set to <not set>.
4. Check the Restrict Migration check box.
5. Select ucs in Host Firmware Package.

Create Service Profile Template

Unified Computing System Manager

Create Service Profile Template

1. ☒ [Identify Service Profile Template](#)
2. ☒ [Storage Provisioning](#)
3. ☒ [Networking](#)
4. ☒ [SAN Connectivity](#)
5. ☒ [Zoning](#)
6. ☒ [vNIC/vHBA Placement](#)
7. ☒ [vMedia Policy](#)
8. ☒ [Server Boot Order](#)
9. ☒ [Maintenance Policy](#)
10. ☒ [Server Assignment](#)
11. ☒ [Operational Policies](#)

Server Assignment

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:

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

☒ Up ☐ Down

The service profile template will be associated with one of the servers in the selected pool. If desired, you can specify an additional server pool policy qualification that the selected server must meet. To do so, select the qualification from the list.

Server Pool Qualification:

Restrict Migration: ☒

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.

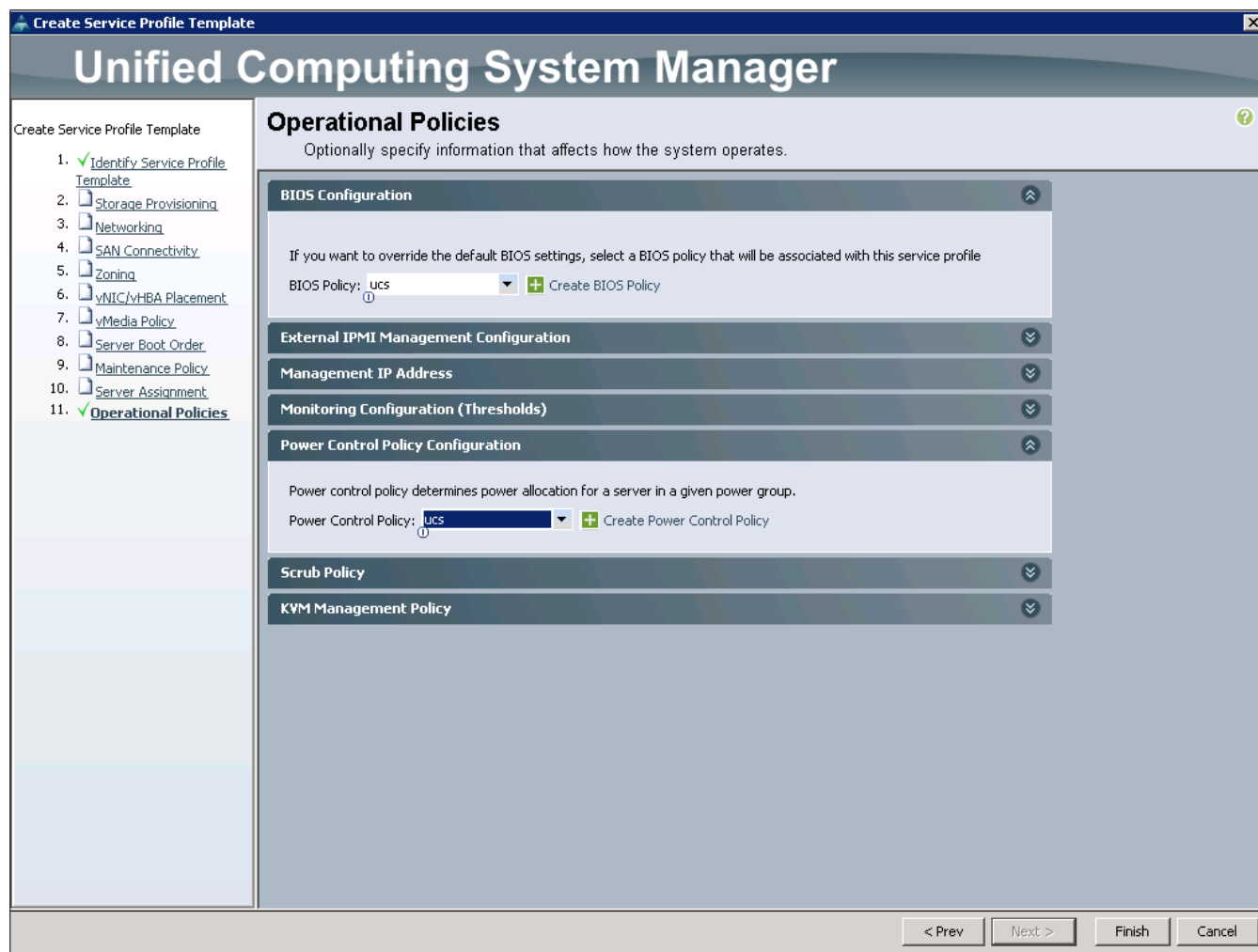
Host Firmware Package:

< Prev Next > Finish Cancel

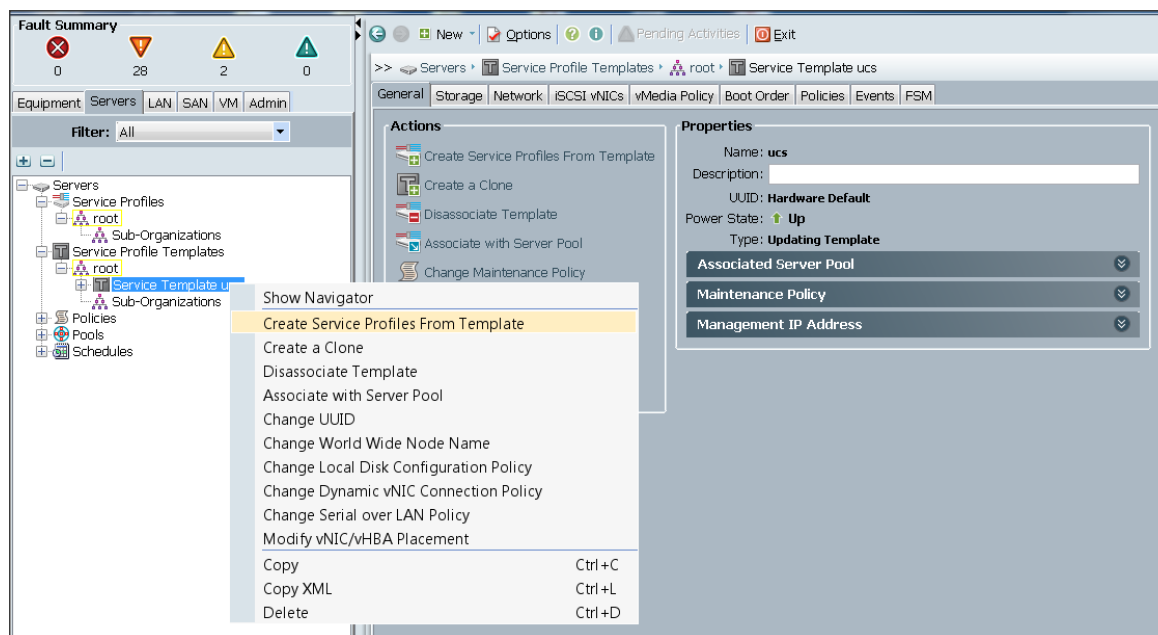
Configuring Operational Policies for the Template

In the Operational Policies Window, complete the following steps:

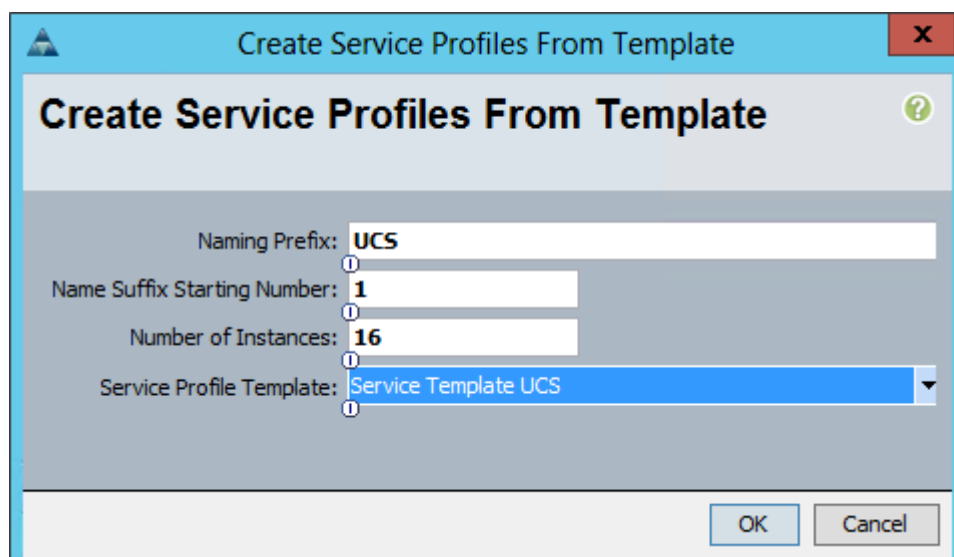
1. Select `ucs` in the BIOS Policy field.
2. Select `ucs` in the Power Control Policy field.



3. Click **Finish** to create the Service Profile template.
4. Click **OK** in the pop-up window to proceed.
5. Select the **Servers** tab in the left pane of the UCS Manager GUI.
6. Go to `Service Profile Templates > root`.
7. Right-click `Service Profile Templates ucs`.
8. Select `Create Service Profiles From Template`.



The Create Service Profiles from Template window appears.



Association of the Service Profiles will take place automatically.

The final Cisco UCS Manager window is shown in below.

Name	Chassis ID	PID	Model	User Label	Cores	Cores Enabled	Memory	Adapters	NICs	HBAs	Overall Status	Operability	Power State	Assoc S
Server 1	1	UCSC-C3K-M45RB	Cisco UCS C3X60M4		28	28	262144	1	6	0	↑ OK	↑ Operable	↑ On	↑ Ass
Server 2	1	UCSC-C3K-M45RB	Cisco UCS C3X60M4		28	28	262144	1	6	0	↑ OK	↑ Operable	↑ On	↑ Ass
Server 1	2	UCSC-C3K-M45RB	Cisco UCS C3X60M4		28	28	262144	1	6	0	↑ OK	↑ Operable	↑ On	↑ Ass
Server 2	2	UCSC-C3K-M45RB	Cisco UCS C3X60M4		28	28	262144	1	6	0	↑ OK	↑ Operable	↑ On	↑ Ass
Server 1	3	UCSC-C3K-M45RB	Cisco UCS C3X60M4		28	28	262144	1	6	0	↑ OK	↑ Operable	↑ On	↑ Ass
Server 2	3	UCSC-C3K-M45RB	Cisco UCS C3X60M4		28	28	262144	1	6	0	↑ OK	↑ Operable	↑ On	↑ Ass
Server 1	4	UCSC-C3K-M45RB	Cisco UCS C3X60M4		28	28	262144	1	6	0	↑ OK	↑ Operable	↑ On	↑ Ass
Server 2	4	UCSC-C3K-M45RB	Cisco UCS C3X60M4		28	28	262144	1	6	0	↑ OK	↑ Operable	↑ On	↑ Ass

Installing Red Hat Enterprise Linux 7.2

The following section provides detailed procedures for installing Red Hat Enterprise Linux 7.2 on Cisco UCS S3260 Storage Servers. There are multiple ways to install the Red Hat Linux operating system. The installation procedure described in this deployment guide uses KVM console and virtual media from Cisco UCS Manager.

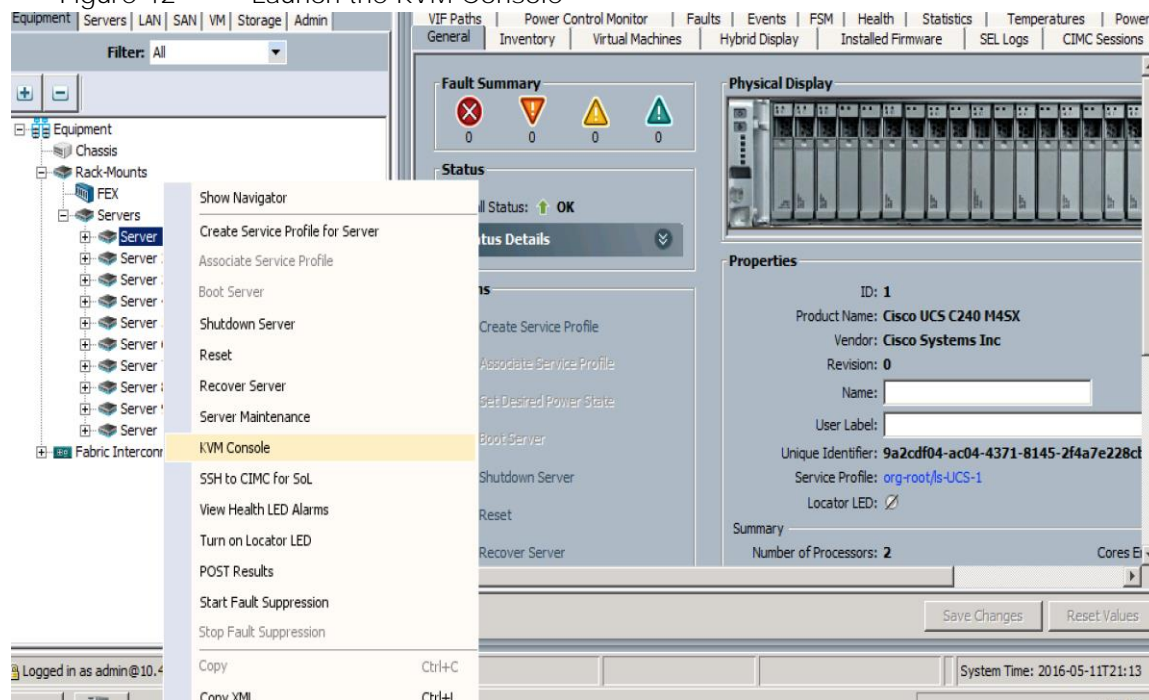


Note: This requires RHEL 7.2 DVD/ISO for the installation

To install the Red Hat Linux 7.2 operating system, complete the following steps:

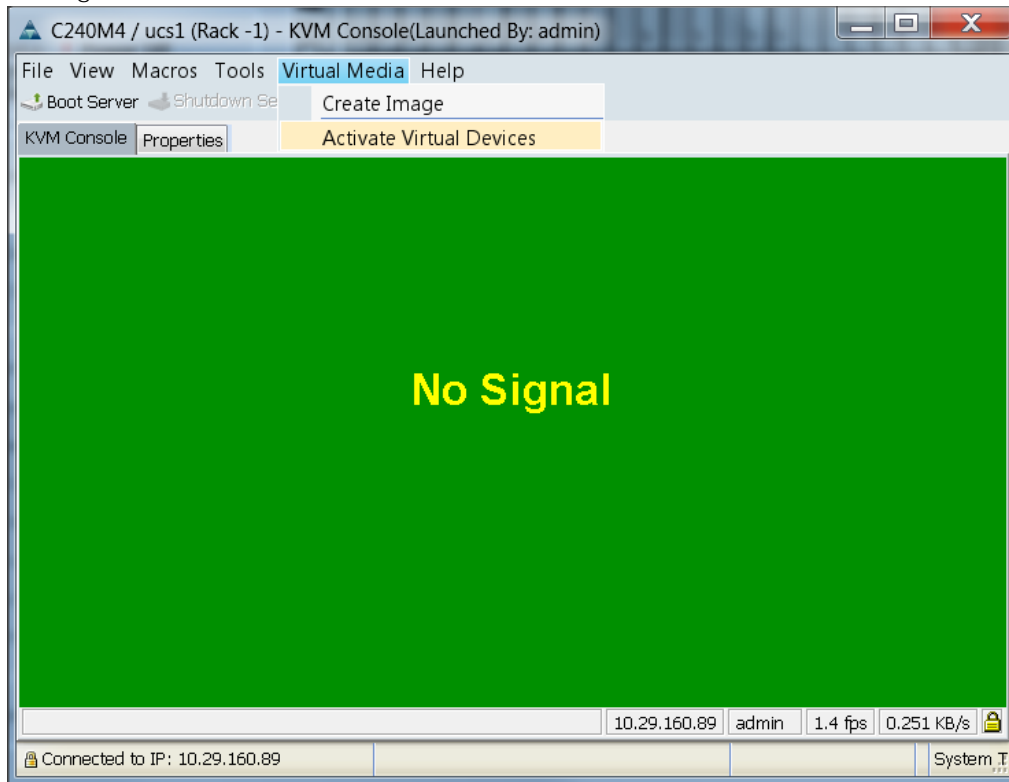
1. Log in to the Cisco UCS 6296 Fabric Interconnect and launch the Cisco UCS Manager application.
2. Select the Equipment tab.
3. In the navigation pane expand Rack-Mounts and then Servers.
4. Right click on the server and select KVM Console (Figure 42).
5. In the KVM window, select the Virtual Media tab.

Figure 42 Launch the KVM Console



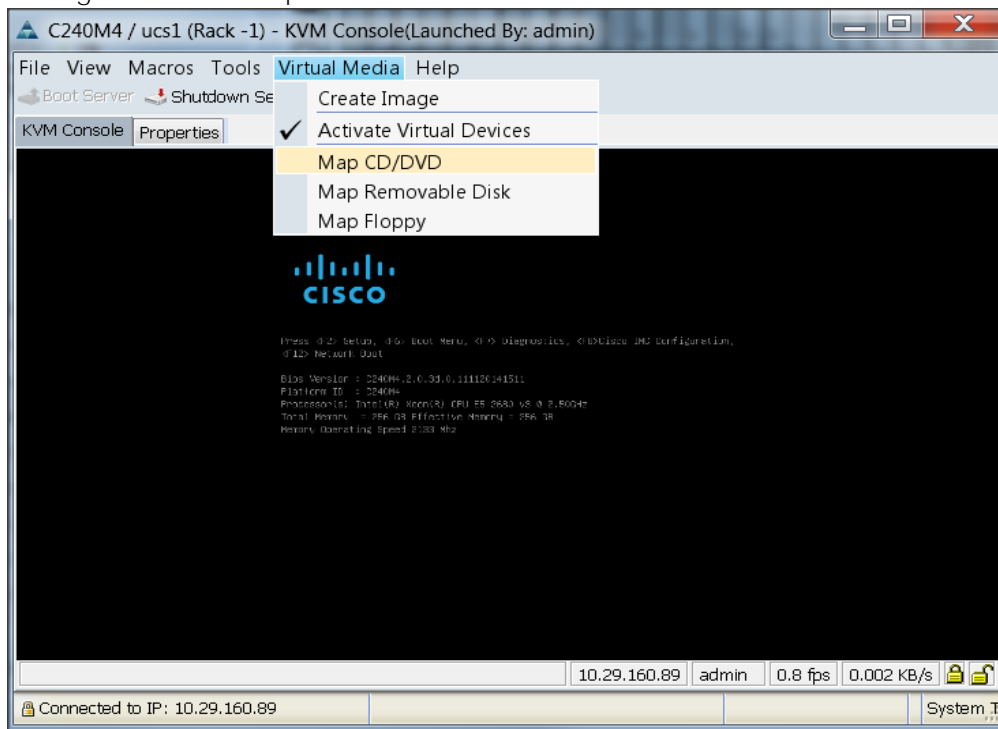
6. Click the Activate Virtual Devices found in Virtual Media tab (Figure 43 Figure 43).

Figure 43 Virtual Media Tab



7. In the KVM window, select the Virtual Media tab and click the Map CD/DVD. (Figure 44)

Figure 44 Map CD/DVD



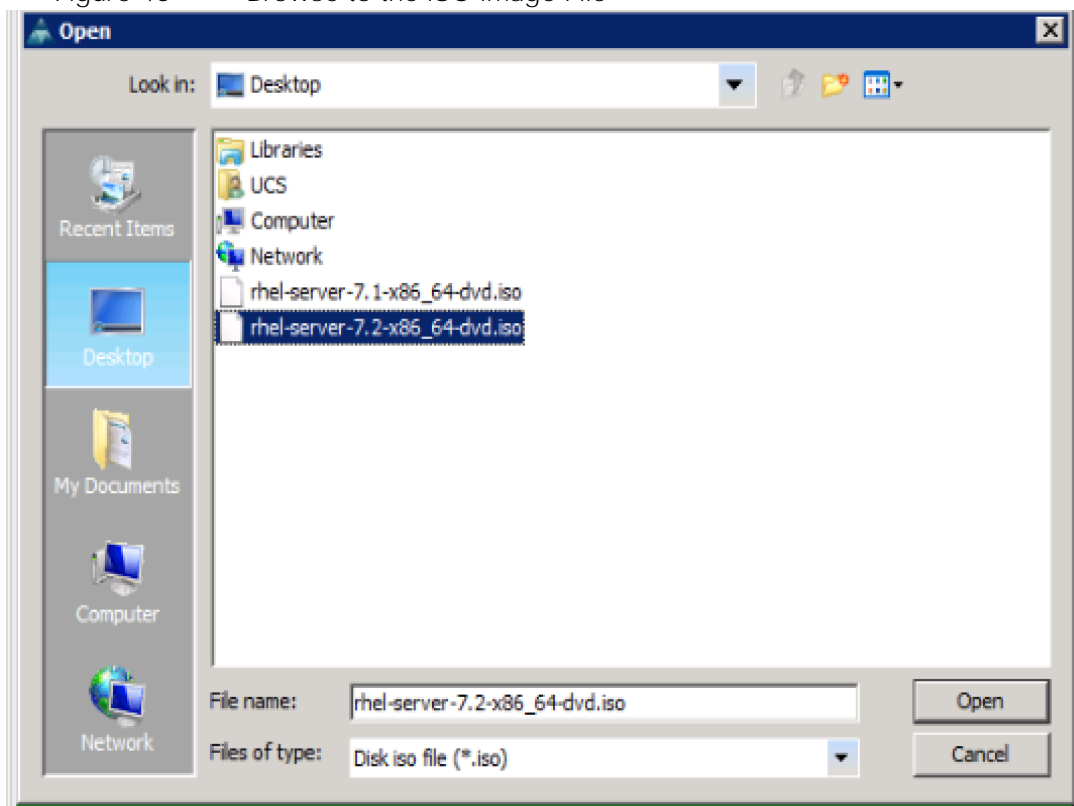
8. Browse to the Red Hat Enterprise Linux Server 7.2 installer ISO image file (Figure 45).



Note: The Red Hat Enterprise Linux 7.2 DVD is assumed to be on the client machine.

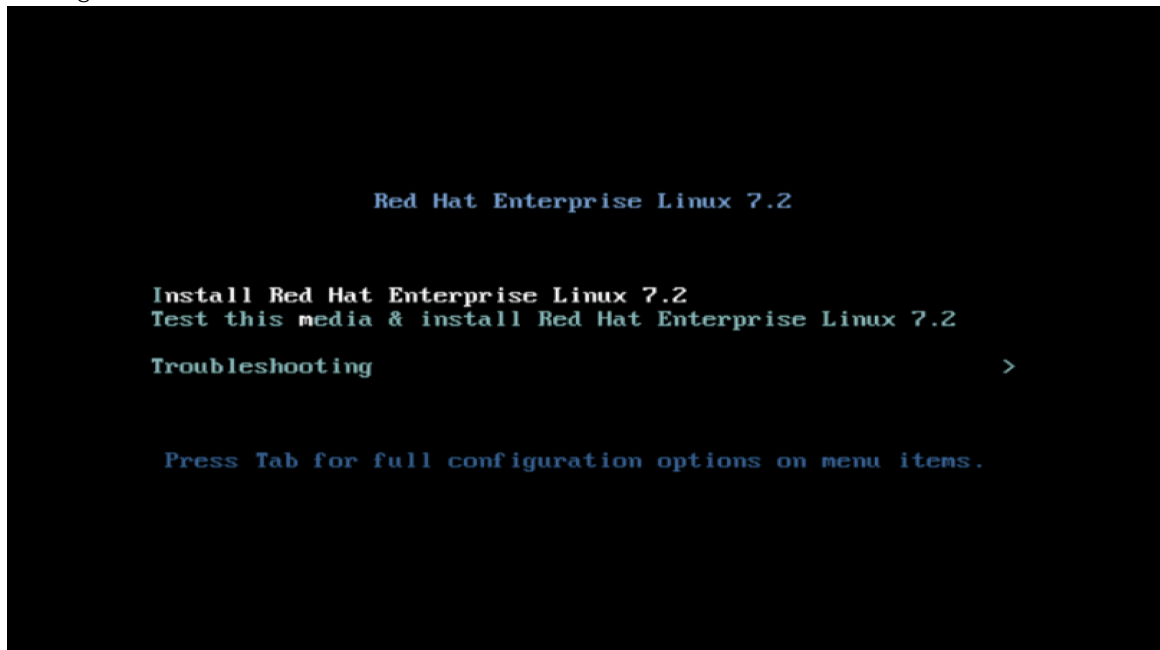
9. Click Open to add the image to the list of virtual media.

Figure 45 Browse to the ISO Image File



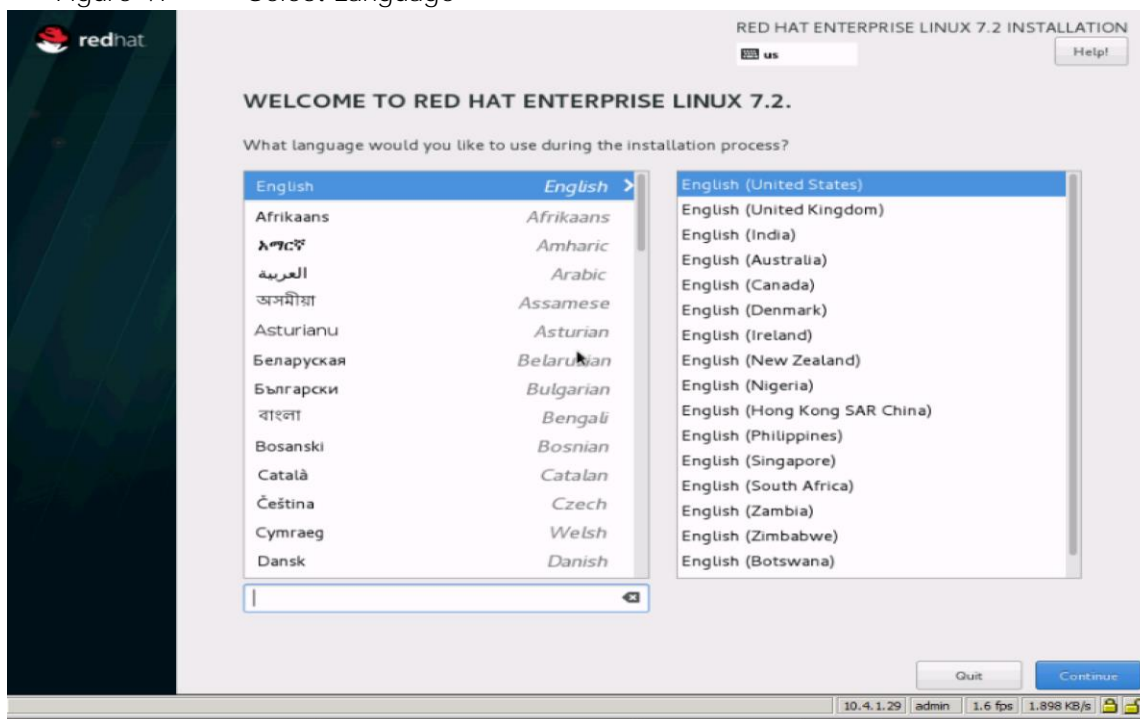
10. In the KVM window, select the KVM tab to monitor during boot.
11. In the KVM window, select the Macros > Static Macros > Ctrl-Alt-Del button in the upper left corner.
12. Click OK.
13. Click OK to reboot the system.
14. On reboot, the machine detects the presence of the Red Hat Enterprise Linux Server 7.2 install media.
15. Select the Install or Upgrade an Existing System. (Figure 46)

Figure 46 Red Hat Linux Installer



16. Skip the Media test and start the installation. Select the language of installation and click Continue. (Figure 47)

Figure 47 Select Language



17. Select Date and Time, which pops up another window as shown below. (Figure 48)

Figure 48 Select Data and Time

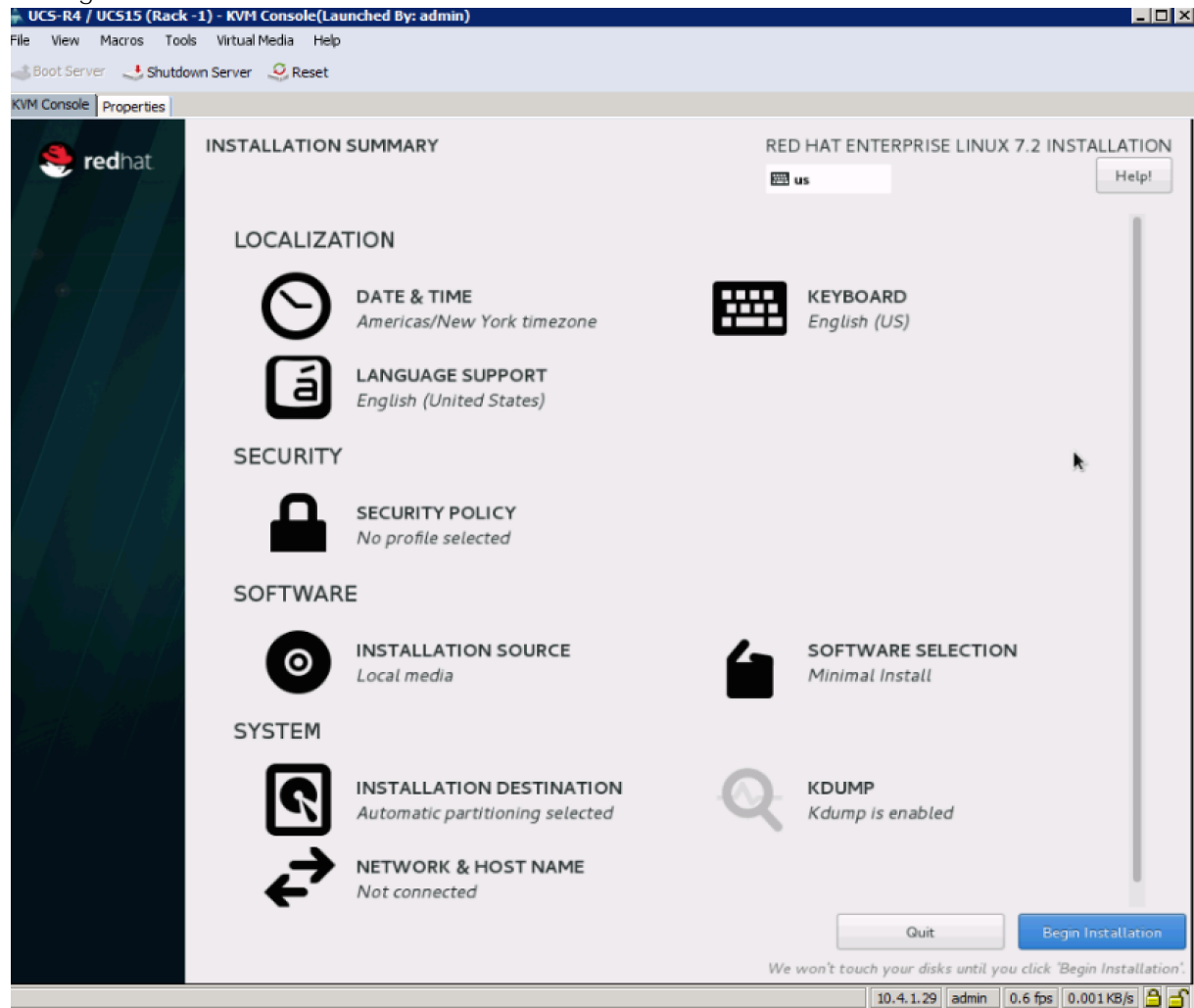
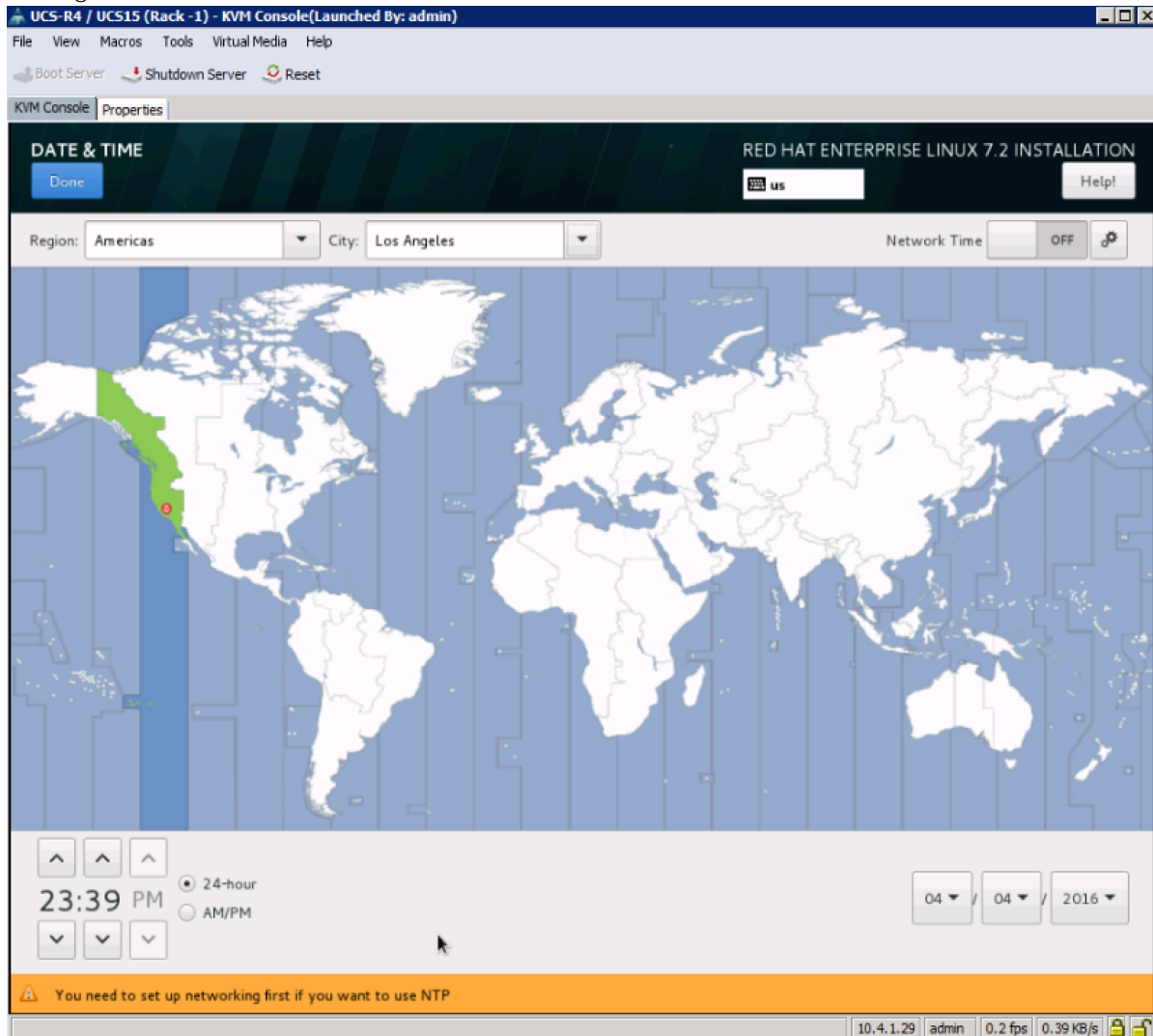


Figure 49 Select Location of Time Zone



18. Select the location on the map, set the time and click Done. (Figure 49)

19. Click on Installation Destination. (Figure 50)

Figure 50 Installation Destination



20. This opens a new window with the boot disks. Make the selection, and choose, "I will configure partitioning". Click Done. (Figure 51)

Figure 51 Select Boot Disks

INSTALLATION DESTINATION RED HAT ENTERPRISE LINUX 7.2 INSTALLATION

[Done](#) [Help!](#)

Device Selection

Select the device(s) you'd like to install to. They will be left untouched until you click on the main menu's "Begin Installation" button.

Local Standard Disks

446.1 GiB

Cisco UCS-C3K-M4RAID

sdd / 446.1 GiB free

Disks left unselected here will not be touched.

Specialized & Network Disks

[Add a disk...](#)

Disks left unselected here will not be touched.

Other Storage Options

Partitioning

☐ Automatically configure partitioning. ☒ I will configure partitioning.

☐ I would like to make additional space available.

Encryption

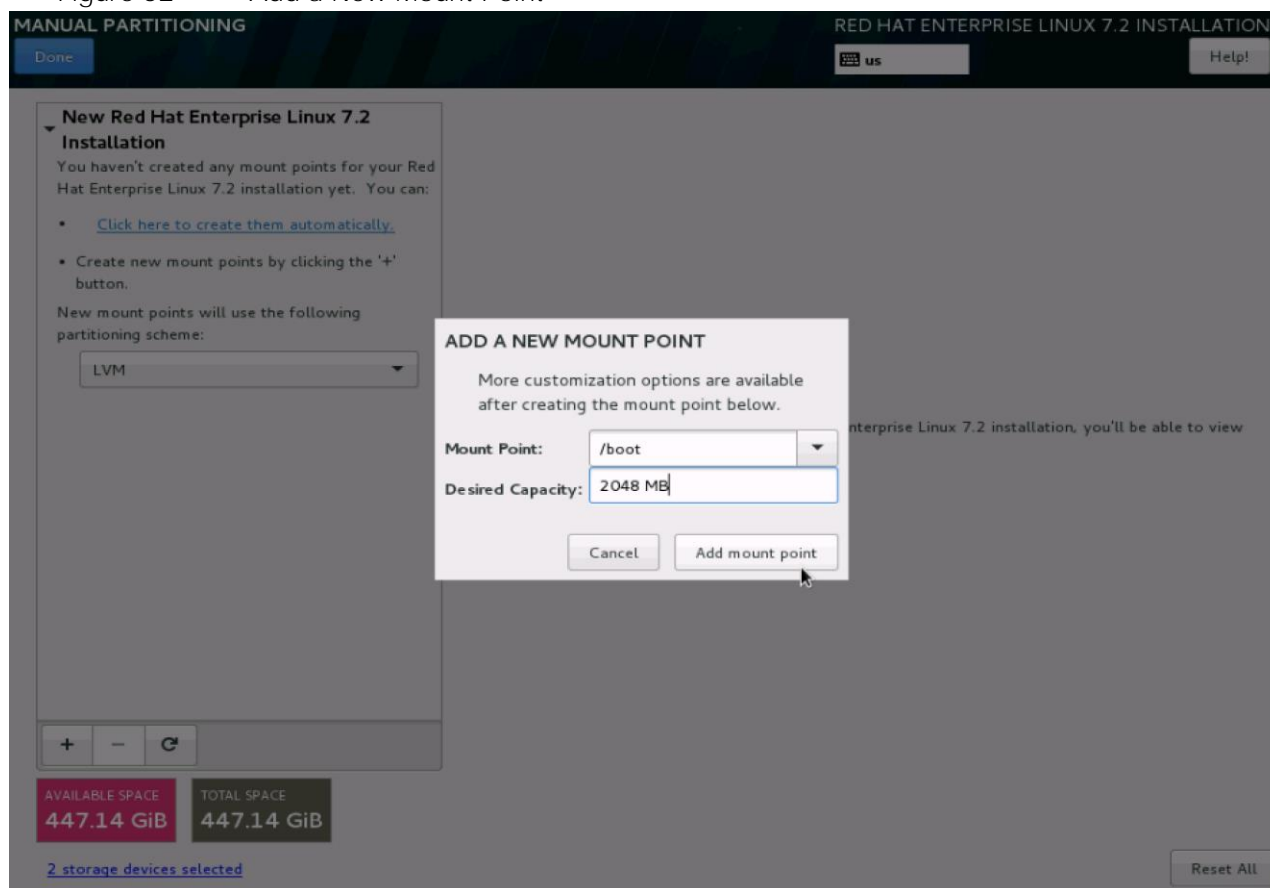
☐ Encrypt my data. *You'll set a passphrase next.*

[Full disk summary and boot loader...](#) 1 disk selected; 446.1 GiB capacity; 446.1 GiB free

21. This opens a window for creating the partitions. Click on the + sign to add a new partition as shown below, boot partition of size 2048 MB. (Figure 52)

22. Click Add MountPoint to add the partition.

Figure 52 Add a New Mount Point



The Manual Partitioning screen opens. (Figure 53)

Figure 53 Manual Partitioning

MANUAL PARTITIONING RED HAT ENTERPRISE LINUX 7.2 INSTALLATION

[Done](#) [Help!](#)

New Red Hat Enterprise Linux 7.2 Installation

SYSTEM

- /boot** 1953 MiB >
 - sdy1

+ - ↺

AVAILABLE SPACE **445.23 GiB** TOTAL SPACE **447.14 GiB**

[2 storage devices selected](#)

sdy1

Mount Point:

Desired Capacity:

Device (s): ATA SAMSUNG MZ7GE240 (sdy) and 1 other

[Modify...](#)

Device Type: ☐ Encrypt

RAID Level:

File System: ☒ Reformat

Label:

Name:

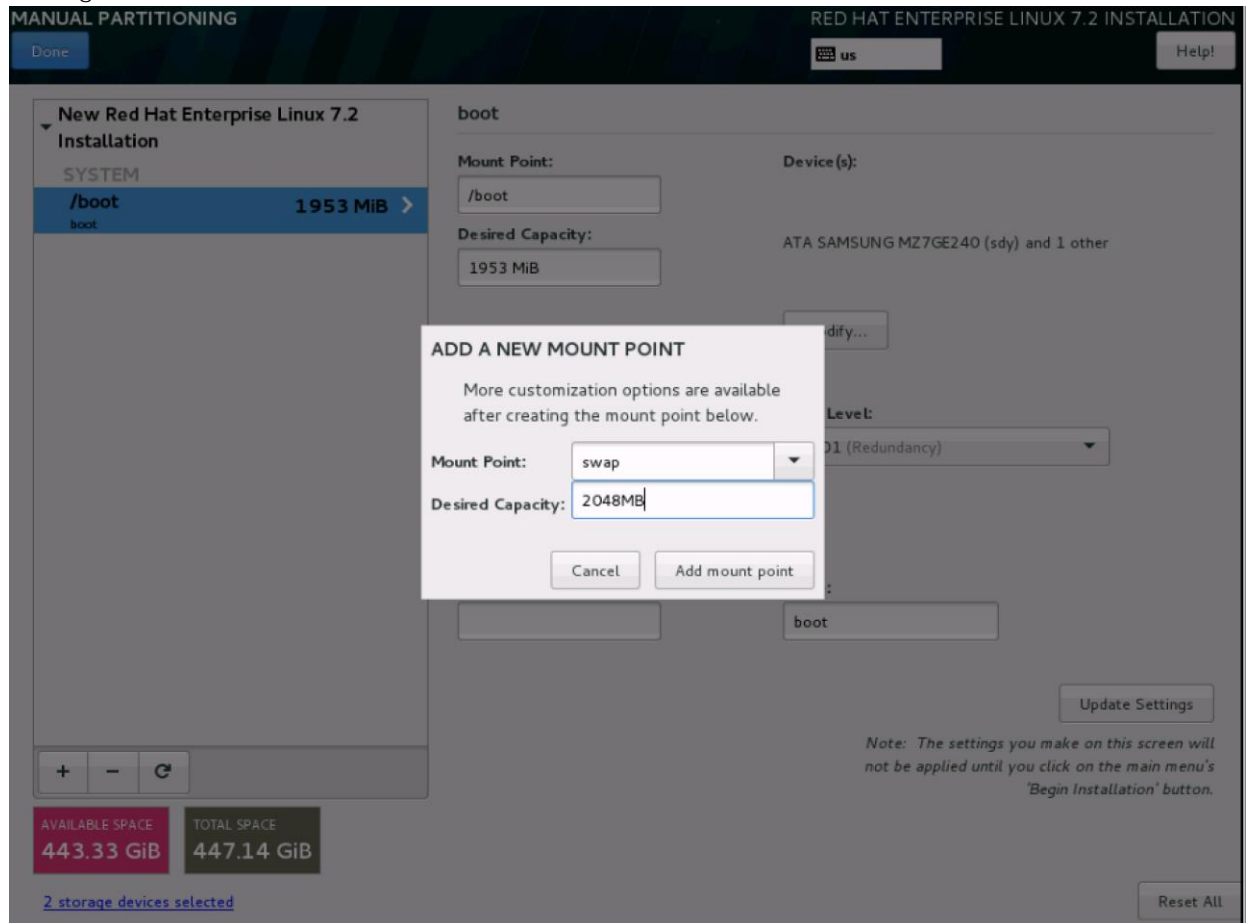
[Update Settings](#)

Note: The settings you make on this screen will not be applied until you click on the main menu's 'Begin Installation' button.

[Reset All](#)

23. Change the Device type to RAID and make sure the RAID Level is RAID1 (Redundancy) and click on Update Settings to save the changes.
24. Click on the + sign to create the swap partition of size 2048 MB as shown below. (Figure 54)

Figure 54 Add a New Mount Point



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

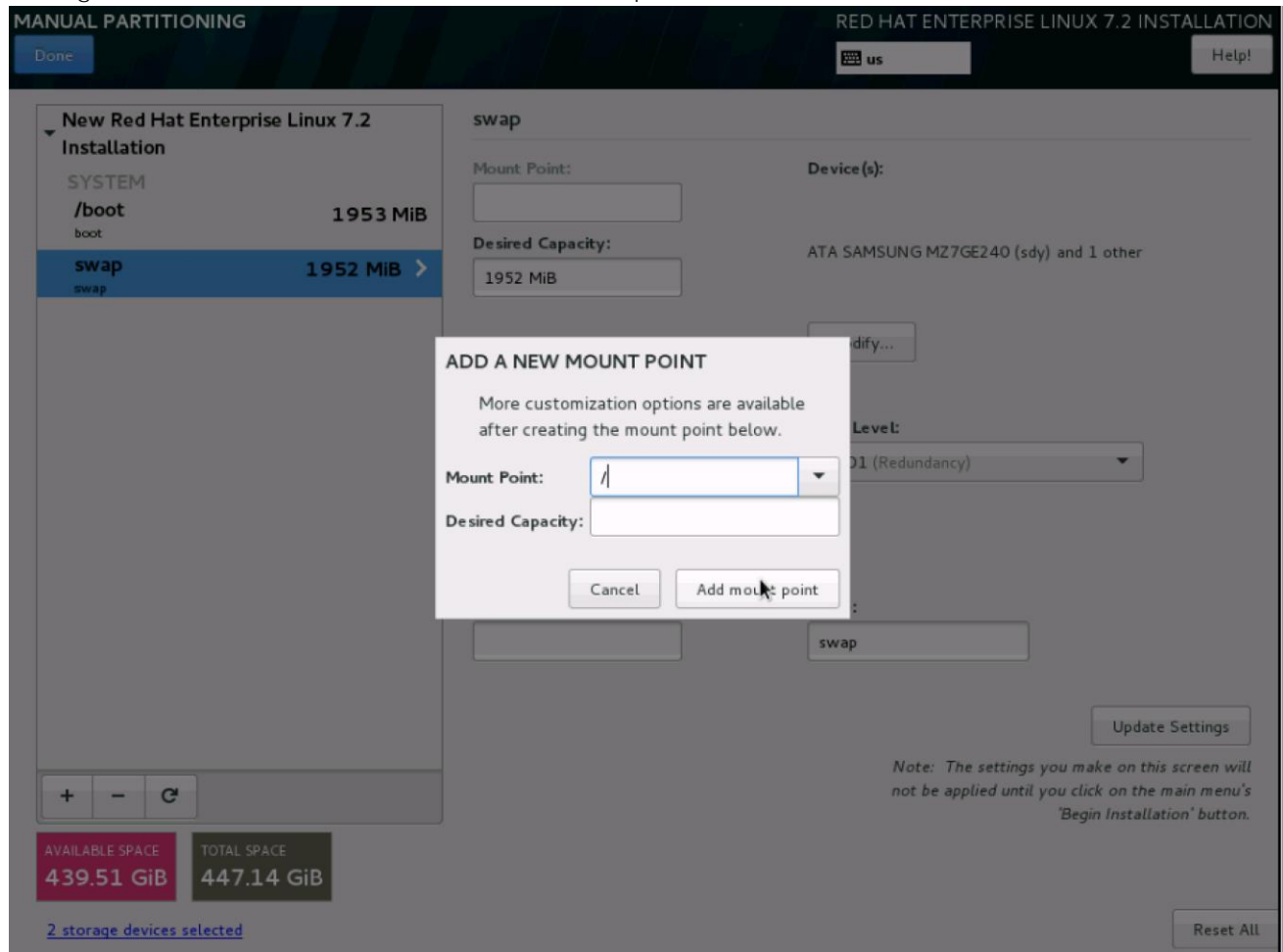
Figure 55 Swap

The screenshot shows the 'MANUAL PARTITIONING' screen for 'RED HAT ENTERPRISE LINUX 7.2 INSTALLATION'. The interface is divided into several sections:

- Left Panel:** A tree view under 'New Red Hat Enterprise Linux 7.2 Installation' showing 'SYSTEM' with sub-items '/boot' (1953 MiB) and 'swap' (1952 MiB). The 'swap' item is selected and highlighted in blue.
- Top Bar:** Includes a 'Done' button, a language selector set to 'us', and a 'Help!' button.
- Main Configuration Area:** Titled 'rheL_rhel8-swap', it contains fields for:
 - Mount Point:** An empty text box.
 - Device(s):** A text box containing 'ATA SAMSUNG MZ7GE240 (sdy) and 1 other'.
 - Desired Capacity:** A text box containing '1952 MiB'.
 - Device Type:** A dropdown menu set to 'RAID'.
 - File System:** A dropdown menu set to 'swap'.
 - RAID Level:** A dropdown menu set to 'RAID1 (Redundancy)'.
 - Label:** An empty text box.
 - Name:** A text box containing 'swap'.
- Buttons:** A 'Modify...' button is located below the 'Device(s)' field. An 'Update Settings' button is at the bottom right of the configuration area.
- Footer:** Displays 'AVAILABLE SPACE 441.41 GiB' and 'TOTAL SPACE 447.14 GiB'. A link '2 storage devices selected' is on the left, and a 'Reset All' button is on the right.
- Notes:** A note at the bottom right states: 'Note: The settings you make on this screen will not be applied until you click on the main menu's 'Begin Installation' button.'

26. Click + to add the / partition. The size can be left empty so it uses the remaining capacity and click Add Mountpoint. (Figure 56)

Figure 56 Add a New Mount Point for Swap



27. Change the Device type to RAID and RAID level to RAID1 (Redundancy). Click Update Settings. (Figure 57)

Figure 57 RAID 1

The screenshot shows the 'New Red Hat Enterprise Linux 7.2 Installation' disk partitioning screen. On the left, a list of partitions is shown: /boot (1953 MiB), / (439.5 GiB, highlighted), and swap (1952 MiB). The / partition is selected, and its settings are shown on the right. The 'Mount Point' is '/', 'Desired Capacity' is 439.5 GiB, and 'Device(s)' is 'ATA SAMSUNG MZ7GE240 (sdy) and 1 other'. The 'Device Type' is set to 'RAID', 'File System' is 'xfs', and 'RAID Level' is 'RAID1 (Redundancy)'. The 'Name' is 'root'. There are buttons for '+', '-', and a refresh icon at the bottom left. At the bottom right, there are buttons for 'Update Settings' and 'Reset All'. A note at the bottom right states: 'Note: The settings you make on this screen will not be applied until you click on the main menu's 'Begin Installation' button.'

New Red Hat Enterprise Linux 7.2 Installation

SYSTEM

Partition	Size
/boot	1953 MiB
/	439.5 GiB
swap	1952 MiB

rhel_rhel8-root

Mount Point: /

Desired Capacity: 439.5 GiB

Device(s): ATA SAMSUNG MZ7GE240 (sdy) and 1 other

Device Type: RAID ☐ Encrypt

File System: xfs ☒ Reformat

RAID Level: RAID1 (Redundancy)

Label:

Name: root

Update Settings

Reset All

AVAILABLE SPACE: 3185 KiB

TOTAL SPACE: 447.14 GiB

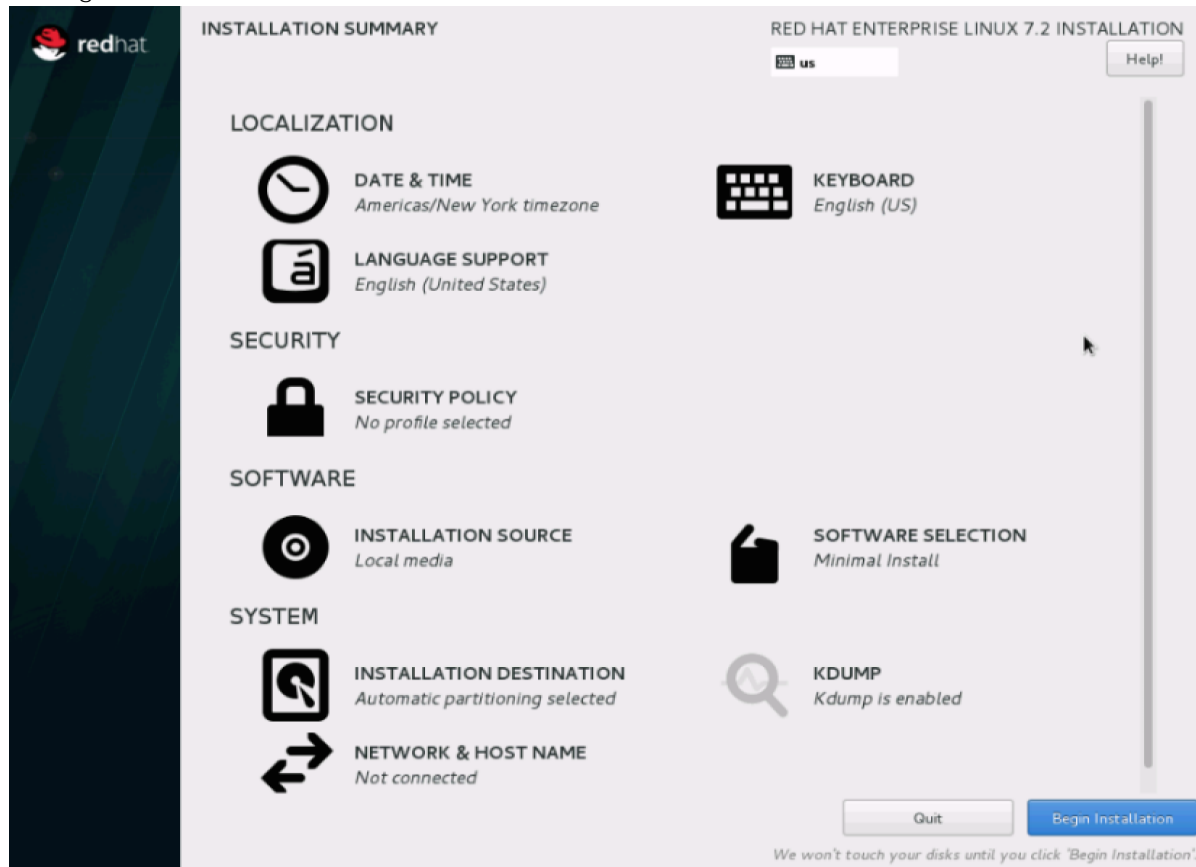
[2 storage devices selected](#)

Note: The settings you make on this screen will not be applied until you click on the main menu's 'Begin Installation' button.

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

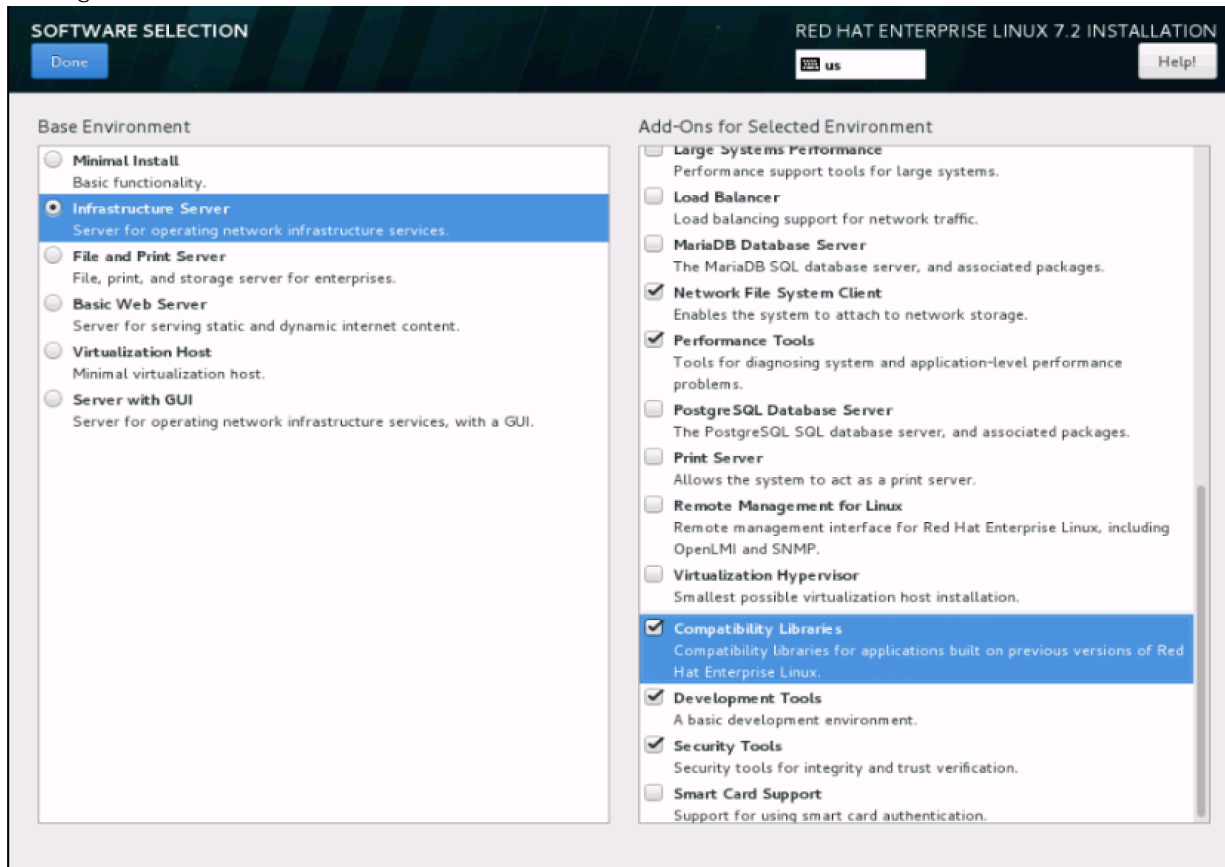
29. Click on Software Selection. (Figure 58)

Figure 58 Software Selection



30. Select Infrastructure Server, and select the Add-Ons as noted below. Click Done. (Figure 59)

Figure 59 Select Add ons



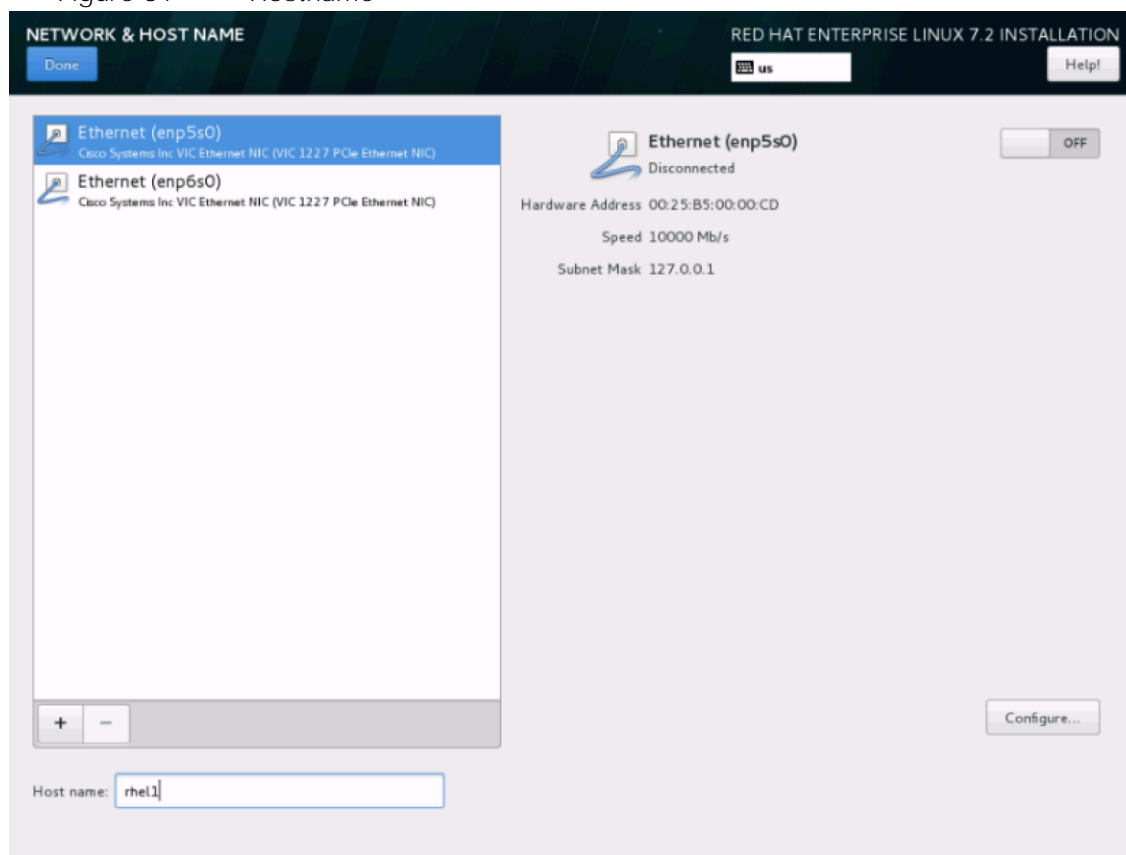
31. Click on Network and Hostname and configure Hostname and Networking for the Host. (Figure 60)

Figure 60 Network and Hostname



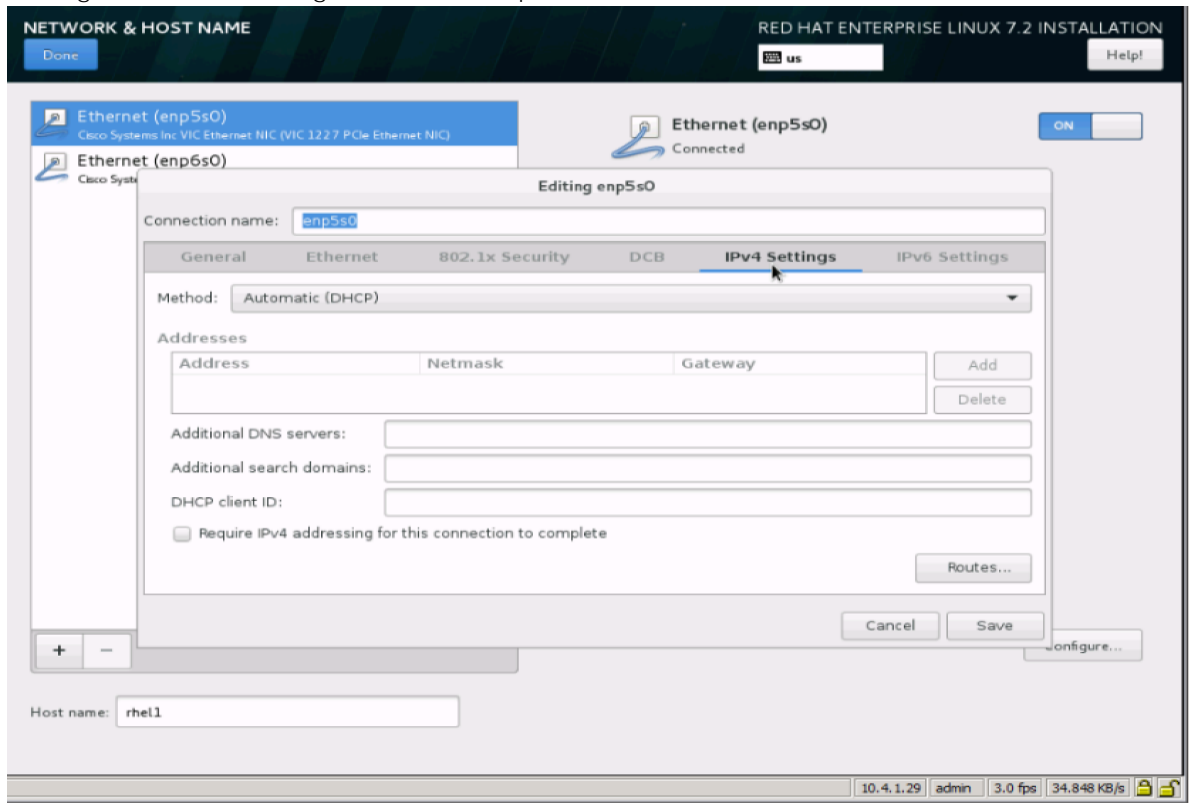
32. Type in the hostname as shown below. (Figure 61)

Figure 61 Hostname



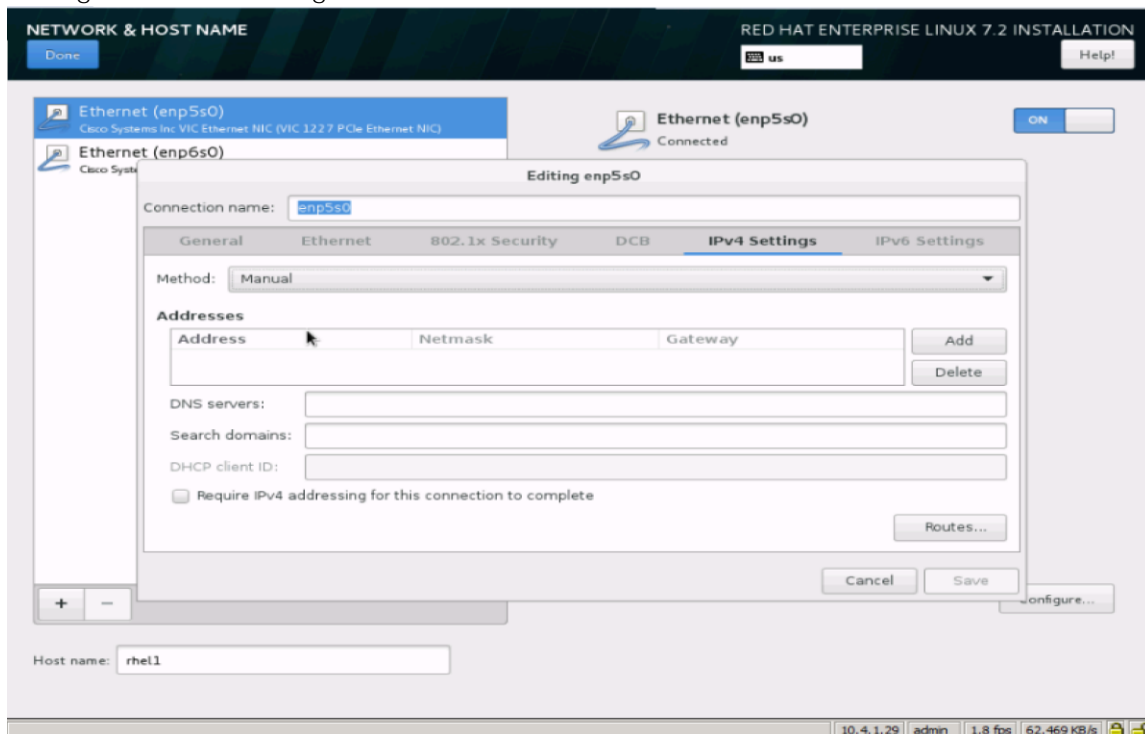
33. Click on Configure to open the Network Connectivity window. Click on IPV4Settings. (Figure 62)

Figure 62 Configure Ethernet enp5s0



34. Change the Method to Manual and click Add to enter the IP Address, Netmask and Gateway details. (Figure 63)

Figure 63 Change Method to Manual



35. Click Save. Update the hostname and turn Ethernet ON. (Figure 64)

Figure 64 Edit enp5s0

Editing enp5s0

Connection name:

General Ethernet 802.1x Security DCB **IPv4 Settings** IPv6 Settings

Method:

Addresses

Address	Netmask	Gateway
172.16.46.14	24	172.16.46.1

DNS servers:

Search domains:

DHCP client ID:

☒ Require IPv4 addressing for this connection to complete

36. Click Done to return to the main menu.



Note: Follow similar steps to assign IP for enp6s0 on different subnet in this case 10.5.1.31

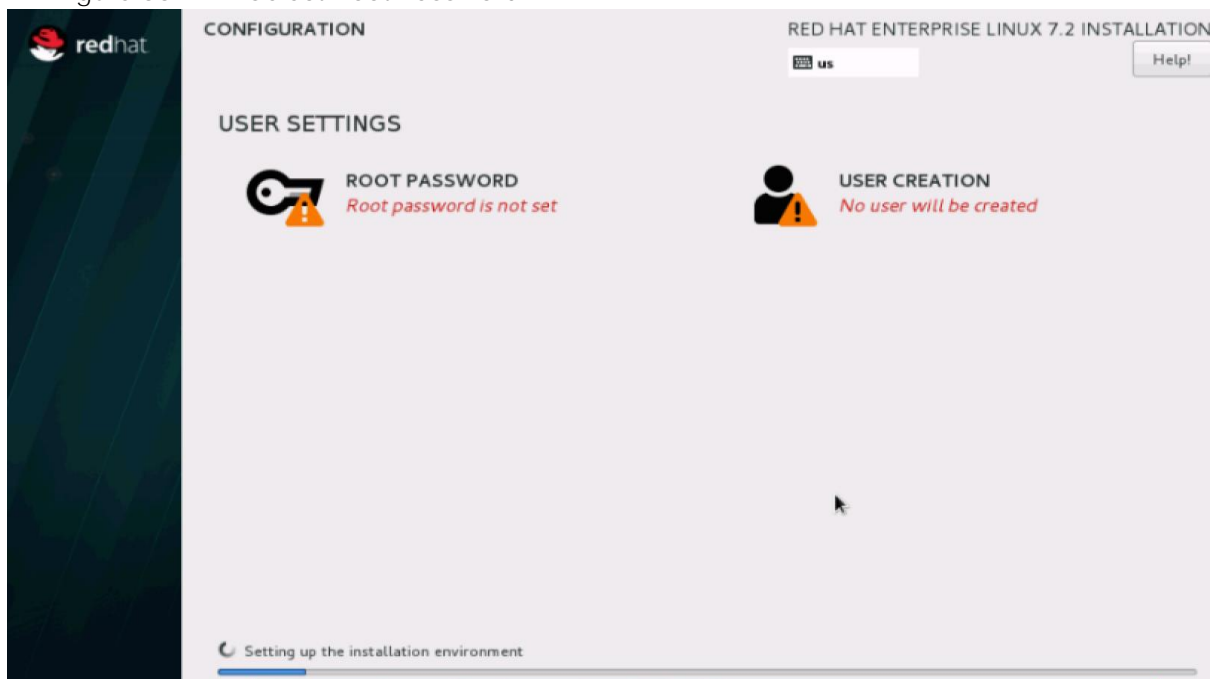
37. Click Begin Installation in the main menu. (Figure 65)

Figure 65 Begin Installation



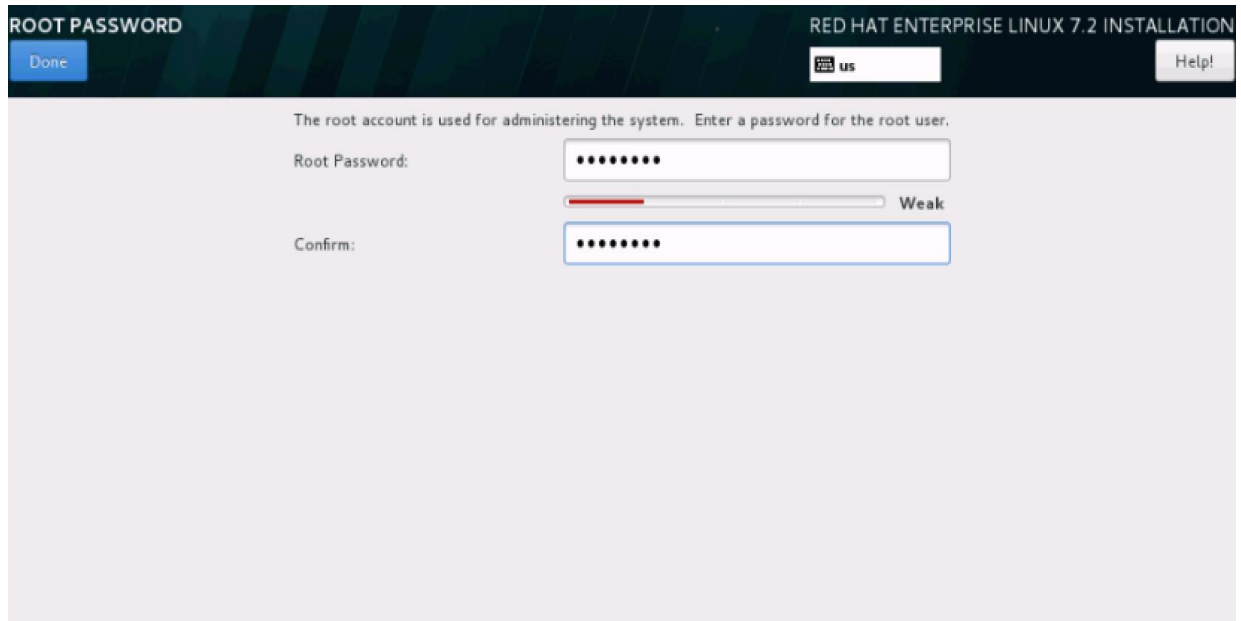
38. Under User Settings, select Root Password. (Figure 66)

Figure 66 Select Root Password




39. Enter the Root Password and click done. (Figure 67)

Figure 67 E



ROOT PASSWORD RED HAT ENTERPRISE LINUX 7.2 INSTALLATION

[Done](#)  **us** [Help!](#)

The root account is used for administering the system. Enter a password for the root user.

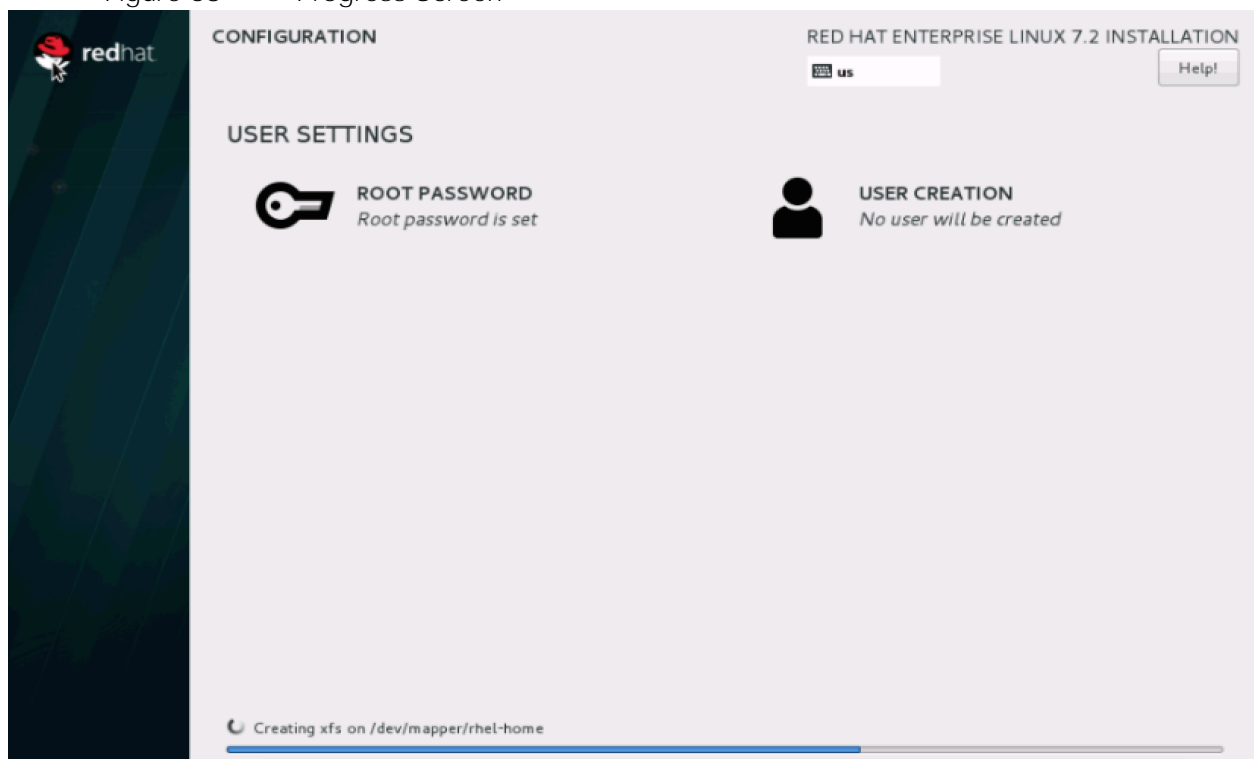
Root Password:

Weak



Confirm:

S
word


Figure 68 Progress Screen




CONFIGURATION RED HAT ENTERPRISE LINUX 7.2 INSTALLATION

 **redhat**  **us** [Help!](#)


USER SETTINGS



ROOT PASSWORD
Root password is set



USER CREATION
No user will be created

 Creating xfs on /dev/mapper/rhel-home

40. Once the installation is complete reboot the system.

41. Repeat steps 1 to 40 to install Red Hat Enterprise Linux 7.2 on Servers 2 through 64.



Note: 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 9 .

Table 9 Hostnames and IP Addresses

Hostname	eth0	eth1
rhel1	172.16.46.11	172.17.46.11
rhel2	172.16.46.12	172.17.46.12
rhel3	172.16.46.13	172.17.46.13
rhel4	172.16.46.14	172.17.46.14
rhel1	172.16.46.15	172.17.46.15
rhel6	172.16.46.16	172.17.46.16
rhel7	172.16.46.17	172.17.46.17
rhel8	172.16.46.18	172.17.46.18
rhel9	172.16.46.19	172.17.46.19
rhel10	172.16.46.20	172.17.46.20
rhel11	172.16.46.21	172.17.46.21
rhel12	172.16.46.22	172.17.46.22
rhel13	172.16.46.23	172.17.46.23
rhel14	172.16.46.24	172.17.46.24
rhel15	172.16.46.25	172.17.46.25
rhel16	172.16.46.26	172.17.46.26



Note: With MapR supporting multiple NICs, Hadoop will use multiple IP subnets for its data traffic, vlan76 and vlan77 can be configured to carry Hadoop data traffic allowing the use of both the fabric interconnects (40 GigE on each fabric allowing 80Gbps active-active connectivity).

Post OS Install Configuration

Choose one of the nodes of the cluster or a separate node as the Admin Node for management such as MapR installation, cluster parallel shell, creating a local Red Hat repo and others. In this document, we use rhel1 for this purpose.

Setting Up Password-less Login

To manage all of the clusters nodes from the admin node, password-less login needs to be setup. It assists in automating common tasks with clustershell (clush, a cluster wide parallel shell), and shell-scripts without having to use passwords.

Once Red Hat Linux is installed across all the nodes in the cluster, follow the steps below in order to enable password-less login across all the nodes.

1. Login to the Admin Node (rhel1).

```
#ssh 172.16.46.11
```

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

```
[root@rhel1 ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
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:
87:78:ad:cc:56:0b:52:e4:0a:86:19:23:cb:27:5e:ed root@rhel1
The key's randomart image is:
+--[ RSA 2048 ]-----+
| . o .
|.o =. o
|.oooo. o
|. +... + o
|. E+ S +
| = = .
| = .
|.
+-----+

```

3. Download sshpass to the node connected to the internet and copy it to the admin node (rhel1) using the command

```
wget ftp://195.220.108.108/linux/dag/redhat/el6/en/x86\_64/dag/RPMS/sshpas-1.05-1.el6.rf.x86\_64.rpm
```

```
scp sshpass-1.05-1.el6.x86_64.rpm rhel1:/root/
```

4. Log in to the admin node and Install the rpm using the command

```
yum -y install sshpass-1.05-1.el6.x86_64.rpm
```

5. Create a file under /.ssh/config and enter the following lines

```
vi ~/.ssh/config
```

```
ServerAliveInterval 99
StrictHostKeyChecking no
```

6. Then 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-**host's** `.ssh/authorized_keys`.

```
#for IP in {11..26}; do echo -n "$IP -> "; sshpass -p secret123 ssh-copy-id -
i ~/.ssh/id_rsa.pub 172.16.46.$IP; done
```

Configuring /etc/hosts

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

To create the host file on the admin node, complete the following steps:

1. Populate the host file with IP addresses and corresponding hostnames on the Admin node (rhel1) and other nodes as follows:
2. On Admin Node (rhel1)

```
#vi /etc/hosts
```

```
127.0.0.1 localhost localhost.localdomain localhost4 \ lo-
calhost4.localdomain4
```

```
:::1 localhost localhost.localdomain localhost6 \ localhost6.localdomain6
```

```
172.16.46.11    rhel1
```

```
172.16.46.12    rhel2
```

```
172.16.46.13    rhel3
```

```
172.16.46.14    rhel4
```

```
172.16.46.15    rhel5
```

```
172.16.46.16    rhel6
```

```
172.16.46.17    rhel7
```

```
172.16.46.18    rhel8
```

```
172.16.46.19    rhel9
```

```
172.16.46.20    rhel10
```

```
172.16.46.21    rhel11
```

172.16.46.22	rhel12
172.16.46.23	rhel13
172.16.46.24	rhel14
172.16.46.25	rhel15
172.16.46.26	rhel16
172.17.46.11	rhel1-1
172.17.46.12	rhel2-2
172.17.46.13	rhel3-3
172.17.46.14	rhel4-4
172.17.46.15	rhel5-5
172.17.46.16	rhel6-6
172.17.46.17	rhel7-7
172.17.46.18	rhel8-8
172.17.46.19	rhel9-9
172.17.46.20	rhel10-10
172.17.46.21	rhel11-11
172.17.46.22	rhel12-12
172.17.46.23	rhel13-13
172.17.46.24	rhel14-14
172.17.46.25	rhel15-15
172.17.46.26	rhel16-16

Creating a Red Hat Enterprise Linux (RHEL) 7.2 Local Repo

To create a repository using RHEL DVD or ISO on the admin node (in this deployment rhel1 is used for this purpose), create a directory with all the required RPMs, run the `createrepo` command and then publish the resulting repository.

1. Log on to rhel1. 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 rhel1.

4. And login back to rhel1 and create the mount directory.

```
#scp rhel-server-7.2-x86_64-dvd.iso rhel1:/root/

#mkdir -p /mnt/rheliso

#mount -t iso9660 -o loop /root/rhel-server-7.2-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
```

```
[root@rhel1 ~]# mkdir -p /var/www/html/rhelrepo
[root@rhel1 ~]# mkdir -p /mnt/rheliso
[root@rhel1 ~]# mount -t iso9660 -o loop /root/rhel-server-7.2-x86_64-dvd.iso /mnt/rheliso/
mount: /dev/loop0 is write-protected, mounting read-only
[root@rhel1 ~]# cp -r /mnt/rheliso/* /var/www/html/rhelrepo
```

6. Now on rhel1 create a .repo file to enable the use of the yum command.

```
#vi /var/www/html/rhelrepo/rheliso.repo

[rhel7.2]

name=Red Hat Enterprise Linux 7.2

baseurl=http://172.16.46.11/rhelrepo

gpgcheck=0

enabled=1
```

7. Now copy rheliso.repo file from /var/www/html/rhelrepo to /etc/yum.repos.d on rhel1.

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



Note: Based on this repo file yum requires httpd to be running on rhel1 for other nodes to access the repository.

8. To make use of repository files on rhel1 without httpd, edit the baseurl of repo file /etc/yum.repos.d/rheliso.repo to point repository location in the file system.



Note: This step is needed to install software on Admin Node (rhel1) using the repo (such as httpd, create-repo, etc.)

```
#vi /etc/yum.repos.d/rheliso.repo

[rhel7.2]

name=Red Hat Enterprise Linux 7.2
```

```
baseurl=file:///var/www/html/rhelrepo

gpgcheck=0

enabled=1
```

Creating the Red Hat Repository Database.

1. Install the createrepo package on admin node (rhel1). Use it to regenerate the repository database(s) for the local copy of the RHEL DVD contents.

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

```
[root@rhel1 ~]# yum -y install createrepo
Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-
: manager
This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.
Resolving Dependencies
--> Running transaction check
--> Package createrepo.noarch 0:0.9.9-23.el7 will be installed
--> Processing Dependency: deltarpm for package: createrepo-0.9.9-23.el7.noarch
--> Processing Dependency: python-deltarpm for package: createrepo-0.9.9-23.el7.noarch
--> Running transaction check
--> Package deltarpm.x86_64 0:3.6-3.el7 will be installed
--> Package python-deltarpm.x86_64 0:3.6-3.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                Arch      Version           Repository        Size
=====
Installing:
createrepo             noarch    0.9.9-23.el7      rhel7.2           92 k
Installing for dependencies:
deltarpm               x86_64    3.6-3.el7         rhel7.2           82 k
python-deltarpm        x86_64    3.6-3.el7         rhel7.2           31 k

Transaction Summary
=====
Install 1 Package (+2 Dependent packages)

Total download size: 205 k
Installed size: 553 k
Downloading packages:
```

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

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

#createrepo .
```

```
[root@rhel1 rhelrepo]# createrepo .
Spawning worker 0 with 3763 pkgs
Workers Finished
Gathering worker results

Saving Primary metadata
Saving file lists metadata
Saving other metadata
Generating sqlite DBs
Sqlite DBs complete
```

Setting up ClusterShell

ClusterShell (or clush) is the cluster-wide shell that runs commands on several hosts in parallel.

1. From the system connected to the Internet download Cluster shell (clush) and install it on rhel1. Cluster shell is available from EPEL (Extra Packages for Enterprise Linux) repository.

```
#wget
http://rpm.phone.net/index.php3/stat/4/idpl/31529309/dir/redhat\_el\_7/com/clustershell-1.7-1.el7.noarch.rpm.html
```

```
#scp clustershell-1.7-1.el7.noarch.rpm rhel1:/root/
```

2. Login to rhel1 and install cluster shell.

```
#yum -y install clustershell-1.7-1.el7.noarch.rpm
```

```
[root@rhel1 ~]# yum -y install clustershell-1.7-1.el7.noarch.rpm
Loaded plugins: product-id, search-disabled-repos, subscription-manager
This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.
Examining clustershell-1.7-1.el7.noarch.rpm: clustershell-1.7-1.el7.noarch
Marking clustershell-1.7-1.el7.noarch.rpm to be installed
Resolving Dependencies
--> Running transaction check
--> Package clustershell.noarch 0:1.7-1.el7 will be installed
--> Processing Dependency: PyYAML for package: clustershell-1.7-1.el7.noarch
--> Running transaction check
--> Package PyYAML.x86_64 0:3.10-11.el7 will be installed
--> Processing Dependency: libyaml-0.50.2()(64bit) for package: PyYAML-3.10-11.el7.x86_64
--> Running transaction check
--> Package libyaml.x86_64 0:0.1.4-11.el7_0 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
clustershell noarch 1.7-1.el7 /clustershell-1.7-1.el7.noarch 1.8 M
Installing for dependencies:
PyYAML x86_64 3.10-11.el7 rhel7.2 153 k
libyaml x86_64 0.1.4-11.el7_0 rhel7.2 55 k
=====

Transaction Summary
=====
Install 1 Package (+2 Dependent packages)
=====

Total size: 2.0 M
Total download size: 208 k
Installed size: 2.5 M
Downloading packages:
-----
Total 98 MB/s | 208 kB 00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : libyaml-0.1.4-11.el7_0.x86_64 1/3
Installing : PyYAML-3.10-11.el7.x86_64 2/3
Installing : clustershell-1.7-1.el7.noarch 3/3
Verifying : libyaml-0.1.4-11.el7_0.x86_64 1/3
Verifying : clustershell-1.7-1.el7.noarch 2/3
Verifying : PyYAML-3.10-11.el7.x86_64 3/3

Installed:
clustershell.noarch 0:1.7-1.el7

Dependency Installed:
PyYAML.x86_64 0:3.10-11.el7 libyaml.x86_64 0:0.1.4-11.el7_0

Complete!
[root@rhel1 ~]#
```


3. Edit `/etc/clusterhell/groups.d/local.cfg` file to include hostnames for all the nodes of the cluster. This set of hosts is taken when running `clush` with the **'-a' option**.
4. For 16-node cluster as in our CVD, set groups file as follows,

```
#vi /etc/clusterhell/groups.d/local.cfg
```

```
Complete!
[root@rhel1 ~]# vi /etc/clusterhell/groups.d/local.cfg
[root@rhel1 ~]#
```

```
all: rhel[1-16]
```



Note: For more information and documentation on ClusterShell, visit <https://github.com/cea-hpc/clusterhell/wiki/UserAndProgrammingGuide>.



Note: ClusterShell will not work if not `ssh` to the machine earlier (as it requires to be in `known_hosts` file), for instance, as in the case below for `rhel<host>`.

Installing httpd

Setting up RHEL repo on the admin node requires `httpd`. To set up RHEL repository on the admin node, complete the following 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 172.16.46.11:80
```

3. Start `httpd`

```
#service httpd start
```

```
#chkconfig httpd on
```

Disabling SELinux

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

1. SELinux can be disabled by editing `/etc/selinux/config` and changing the `SELINUX` line to `SELINUX=disabled`. The following command will disable `SELINUX` on all nodes.

```
#clush -a -b "sed -i 's/SELINUX=enforcing/SELINUX=disabled/g'
/etc/selinux/config"
```

```
[root@rhel1 ~]# clush -a -b "sed -i 's/SELINUX=enforcing/SELINUX=disabled/g' /etc/selinux/config "
```

```
#clush -a -b "setenforce 0"
```



Note: The above command may fail if SELinux is already disabled.

Disabling the Linux Firewall

The default Linux firewall settings are far 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.

```
#clush -a -b " firewall-cmd --zone=public --add-port=80/tcp --permanent"
```

```
#clush -a -b "firewall-cmd --reload"
```

```
#clush -a -b "systemctl disable firewalld"
```

Set Up all Nodes to use the RHEL Repository



Note: Based on this repo file yum requires httpd to be running on rhel1 for other nodes to access the repository.

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

```
#clush -w rhel[2-16] -c /var/www/html/rhelrepo/rheliso.repo --
dest=/etc/yum.repos.d/
```

```
[root@rhel1 ~]# clush -w rhel[2-16] -c /var/www/html/rhelrepo/rheliso.repo --dest=/etc/yum.repos.d/
```

2. Also copy the /etc/hosts file to all nodes.

```
#clush -w rhel[2-16] -c /etc/hosts --dest=/etc/hosts
```

3. Purge the yum caches after this

```
#clush -a -B yum clean all
```

```
#clush -a -B yum repolist
```

```
[root@rhel1 ~]# clush -a -B yum clean all
-----
rhel[1-16](16)
-----
Loaded plugins: product-id, search-disabled-repos, subscription-manager
This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.
Cleaning repos: rhel7.2
Cleaning up everything
[root@rhel1 ~]# clush -a -B yum repolist
-----
rhel[1-16](16)
-----
Loaded plugins: product-id, search-disabled-repos, subscription-manager
This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.
repo id                repo name                status
rhel7.2                 Red Hat Enterprise Linux 7.2 4,687
repolist: 4,687
```



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

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

Configuring DNS

This section details setting up DNS using `dnsmasq` as an example based on the `/etc/hosts` configuration setup in the earlier section.

To create the host file across all the nodes in the cluster, complete the following steps:

1. Disable Network manager on all nodes:

```
#clush -a -b service NetworkManager stop
#clush -a -b chkconfig NetworkManager off
```

2. Update `/etc/resolv.conf` file to point to Admin Node:

```
#vi /etc/resolv.conf

nameserver 172.16.46.11
```



Note: This step is needed if setting up `dnsmasq` on Admin node. Otherwise this file should be updated with the correct nameserver.



Note: Alternatively `#systemctl start NetworkManager.service` can be used to start the service. `#systemctl stop NetworkManager.service` can be used to stop the service. Use `#systemctl disable NetworkManager.service` to stop a service from being automatically started at boot time.

3. Install and Start `dnsmasq` on Admin node:

```
#service dnsmasq start

#chkconfig dnsmasq on
```



Note: A clush copy without -dest copies to the same directory location as the source-file directory

4. Ensure DNS is working fine by running the following command on Admin node and any data-node:

```
[root@rhel2 ~]# nslookup rhel1

Server: 172.16.46.11

Address: 172.16.46.11#53

Name: rhel1

Address: 172.16.46.11 ←
```



Note: clush -a -b yum install -y bind-utils will need to be run for nslookup to utility to run.

Upgrading 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/release.html?mdfid=283862063&release=2.0\(13\)&reind=AVAILABLE&flowid=25886&softwareid=283853158&rellifecycle=&reltype=latest](https://software.cisco.com/download/release.html?mdfid=283862063&release=2.0(13)&reind=AVAILABLE&flowid=25886&softwareid=283853158&rellifecycle=&reltype=latest)

1. In the ISO image, the required driver kmod-enic-2.3.0.30-rhel7u2.el7.x86_64.rpm can be located at \Network\Cisco\VIC\RHEL\RHEL7.2.
2. From a node connected to the Internet, download, extract and transfer kmod-enic-2.3.0.30-rhel7u2.el7.x86_64.rpm to rhel1 (admin node).
3. Install the rpm on all nodes of the cluster using the following clush commands. For this example the rpm is assumed to be in present working directory of rhel1.

```
[root@rhel1 ~]# clush -a -b -c kmod-enic-2.3.0.30-rhel7u2.el7.x86_64.rpm

[root@rhel1 ~]# clush -a -b "rpm -ivh kmod-enic-2.3.0.30-rhel7u2.el7.x86_64.rpm"
```

4. Ensure 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@rhel1 ~]# clush -a -B "modinfo enic | head -5"
```

```
[root@rhel1 ~]# modinfo enic
filename:      /lib/modules/2.6.32-573.el6.x86_64/extra/enic/enic.ko
version:      2.3.0.30
```

- Also it is recommended to download the kmod-megaraid driver for higher performance, the RPM can be found in the same package at:
`\Linux\Storage\LSI\Cisco_Storage_12G_SAS_RAID_controller\RHEL\RHEL7.2`

Setting up JAVA

MapR requires Java 8. To install Java 8, complete the following steps:

- Download `jdk-8u91-linux-x64.rpm` from [oracle.com](http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html) and scp the rpm to
<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
to admin node (rhel1).
- Run the following commands on admin node (rhel1) to install and setup java on all nodes.
- Copy JDK rpm to all nodes.

```
clush -a -b -c /root/jdk-8u91-linux-x64.rpm --dest=/root/
```

- Extract and Install JDK on all nodes.

```
clush -a -b rpm -ivh /root/jdk-8u91-linux-x64.rpm
```

```
-----
rhel[1-16](16)
-----
Preparing...                               #####
Updating / installing...                   #####
jdk1.8.0_91-2000:1.8.0_91-fcs              #####
Unpacking JAR files...
  tools.jar...
  plugin.jar...
  javaws.jar...
  deploy.jar...
  rt.jar...
  jsse.jar...
  charsets.jar...
  localedata.jar...
  jfxrt.jar...
```

- Create the following files `java-set-alternatives.sh` and `java-home.sh` on the admin node (rhel1).

```
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 $item
/usr/java/jdk1.8.0_91/bin/$item 9

alternatives --set $item /usr/java/jdk1.8.0_91/bin/$item

done
```

```
vi java-home.sh
```

```
export JAVA_HOME=/usr/java/jdk1.8.0_91
```

6. Make the two java scripts created above executable.

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

7. Copying java-set-alternatives.sh to all nodes.

```
clush -b -a -c ./java-set-alternatives.sh --dest=/root/
```

8. Setup Java Alternatives.

```
clush -b -a ./java-set-alternatives.sh
```

9. Ensure correct java is setup on all nodes (should point to newly installed java path).

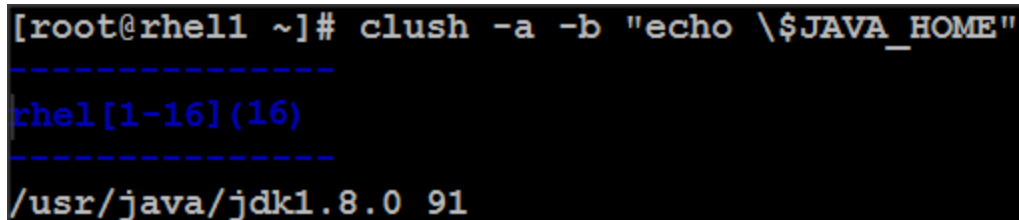
```
clush -b -a "alternatives --display java | head -2"
```

10. Setup JAVA_HOME on all nodes.

```
clush -b -a -c ./java-home.sh --dest=/etc/profile.d
```

11. Display JAVA_HOME on all nodes.

```
clush -a -b "echo \${JAVA_HOME}"
```



```
[root@rhel1 ~]# clush -a -b "echo \${JAVA_HOME}"
-----
rhel[1-16](16)
-----
/usr/java/jdk1.8.0_91
```

12. Display current java -version.

```
[root@rhel1 ~]# clush -B -a java -version
-----
rhel[1-16](16)
-----
java version "1.8.0_91"
Java(TM) SE Runtime Environment (build 1.8.0_91-b14)
Java HotSpot(TM) 64-Bit Server VM (build 25.91-b14, mixed mode)
```

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 (rhel1). 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.

```
#clush -a -b "yum -y install ntp"
```



Note: 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 172.16.46.11

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 command in the admin node (rhel1):

```
#for SERVER in {11..26}; do scp /root/ntp.conf
172.16.46.$SERVER:/etc/ntp.conf; done
```

```
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
ntp.conf 100% 136 0.1KB/s 00:00
```



Note: Instead of the above for loop, this could be run as a clush command with "-w" option.

```
#clush -w rhel[2-16] -b -c /root/ntp.conf --dest=/etc/
```

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

```
#clush -a -b -x rhel1 "service ntpd stop"
```

```
#clush -a -b "ntpdate rhel1"
```

```
#clush -a -b "service ntpd start"
```

5. Ensure restart of NTP daemon across reboots:

```
#clush -a -b "systemctl enable ntpd"
```

Enabling 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. One of the following commands should suffice to confirm that the service is properly configured:

```
#clush -B -a rsyslogd -v
```

```
#clush -B -a service rsyslog status
```



```
[root@rhel1 ~]# clush -B -a rsyslogd -v
-----
rhel[1-16](16)
-----
rsyslogd 7.4.7, compiled with:
    FEATURE_REGEX:                Yes
    FEATURE_LARGEFILE:            No
    GSSAPI Kerberos 5 support:     Yes
    FEATURE_DEBUG (debug build, slow code): No
    32bit Atomic operations supported: Yes
    64bit Atomic operations supported: Yes
    Runtime Instrumentation (slow code): No
    uuid support:                 Yes

See http://www.rsyslog.com for more information.
```

Setting 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.

1. For setting the ulimit on Redhat, edit `/etc/security/limits.conf` on admin node `rhel1` and add the following lines:

```
root soft nofile 64000

root hard nofile 64000
```

```
[root@rhel1 ~]# cat /etc/security/limits.conf | grep 64000
root soft nofile 64000
root hard nofile 64000
```

2. Copy the `/etc/security/limits.conf` file from admin node (`rhel1`) to all the nodes using the following command.

```
#clush -a -b -c /etc/security/limits.conf --dest=/etc/security/
```

```
[root@rhel1 ~]# clush -a -b -c /etc/security/limits.conf --dest=/etc/security/
```

3. Check that the `/etc/pam.d/su` file contains the following settings:

```
##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

account        sufficient      pam_succeed_if.so uid = 0 use_uid quiet

account        include         system-auth

password       include         system-auth

session        include         system-auth

session        optional        pam_xauth.so
```



Note: 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 UCS, this is a safe and recommended change (failures observed at the operating system layer are most likely serious rather than transitory). 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 `rhel1` and add the following lines:

```
net.ipv4.tcp_retries2=5
```

2. Copy the `/etc/sysctl.conf` file from admin node (`rhel1`) to all the nodes using the following command:

```
#clush -a -b -c /etc/sysctl.conf --dest=/etc/
```

3. Load the settings from default `sysctl` file `/etc/sysctl.conf` by running.

```
#clush -B -a sysctl -p
```

Disable Swapping

1. In order to reduce Swapping, run the following on all nodes. Variable `vm.swappiness` defines how often swap should be used, 60 is default.

```
#clush -a -b " echo 'vm.swappiness=1' >> /etc/sysctl.conf"
```

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

```
#clush -a -b "sysctl -p"
```

Disable Transparent Huge Pages

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

```
#clush -a -b "echo never > /sys/kernel/mm/transparent_hugepage/enabled"
```

```
#clush -a -b "echo never > /sys/kernel/mm/transparent_hugepage/defrag"
```

1. The commands above must be run for every reboot, copy these commands to /etc/rc.local to execute automatically for every reboot.
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:

```
#clush -a -b -c /root/thp_disable
```

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

```
#clush -a -b "cat /root/thp_disable >> /etc/rc.local"
```

Disable IPv6 Defaults

1. Disable IPv6 as the addresses used are IPv4.

```
#clush -a -b "echo 'net.ipv6.conf.all.disable_ipv6 = 1' >>  
/etc/sysctl.conf"
```

```
#clush -a -b "echo 'net.ipv6.conf.default.disable_ipv6 = 1' >>  
/etc/sysctl.conf"
```

```
#clush -a -b "echo 'net.ipv6.conf.lo.disable_ipv6 = 1' >>  
/etc/sysctl.conf"
```

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

```
#clush -a -b "sysctl -p"
```

Configuring Data Drives

To configure non-OS disk drives as RAID0 using StorCli command, complete the following steps: These volumes are going to be used for MapR-FS (HDFS supported) Data.

1. From the following link download storcli:

http://docs.avagotech.com/docs/1.19.04_StorCLI.zip

2. Extract the zip file and copy storcli-1.19.04-1.noarch.rpm from the linux directory.
3. Download storcli and its dependencies and transfer to Admin node.

```
#scp storcli-1.19.04-1.noarch.rpm rhell:/root/
```

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

```
#clush -a -b -c /root/storcli-1.19.04-1.noarch.rpm --dest=/root/
```

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

```
#clush -a -b "rpm -ivh storcli-1.19.04-1.noarch.rpm"
```

6. Run the below command to copy storcli64 to root directory.

```
#cd /opt/MegaRAID/storcli/
```

```
#cp storcli64 /root/
```

```
[root@rhell ~]# cd /opt/MegaRAID/storcli/
[root@rhell storcli]# ls
install.log  libstorelibir-2.so  libstorelibir-2.so.14.07-0  storcli64
[root@rhell storcli]# cp storcli64 /root/
```

7. Copy storcli64 to all the nodes using the following commands:

```
#clush -a -b -c /root/storcli64 --dest=/root/
```

8. Run the following command as root user on rhel1

```
# clush -a -B ". ./storcli64 -cfgeachdskraid0 WB RA direct NoCachedBadBBU
strpsz1024 -a0"
```

WB: Write back

RA: Read Ahead

NoCachedBadBBU: Do not write cache when the BBU is bad.

Strpsz1024: Strip Size of 1024K



Note: The command above will not override any existing configuration. To clear and reconfigure existing configurations refer to Embedded MegaRAID Software Users Guide available at www.lsi.com.

Cluster Verification and Micro-Benchmark

This section provides a set of micro-benchmarks and prerequisites scripts to verify that all the systems are configured correctly:

- Prerequisite script to verify configuration across the cluster

- STREAM benchmark to test memory bandwidth
- RPCtest to test network bandwidth
- IOzone to test I/O



Note: Running these tests is optional. Test results can vary based on topology and configuration.

Running the Cluster Verification Script

The section describes the steps to create the script `cluster_verification.sh` that helps to verify CPU, memory, NIC, 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.

1. Create the script `cluster_verification.sh` as shown, on the Admin node (rhel1).

```
#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: '; dmidecode |grep -c \
'^[[:space:]]*Locator:'"

clush -a -B "echo -n 'DIMM count is: '; dmidecode | grep \Size| grep -c
'MB'"

clush -a -B " 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 "ls /sys/class/net | grep ^enp | \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 -srvn | 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: '; cpupower frequency-info \ sta-
tus 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 \ De-
fined!}'

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'

# MapR related RPMs

clush -a -B 'rpm -qa | grep -i nfs |sort'

clush -a -B 'rpm -qa | grep -i nfs |sort'

```



```

clush -a -B 'echo Missing RPMs: ; for each in make patch redhat-lsb
irqbalance syslinux hdparm sdparm dmidecode nc; do rpm -q $each | grep
"is not installed"; done'

clush -a -B "ls -d /opt/mapr/* | head"

# mapr login for hadoop

clush -a -B 'echo "mapr login for Hadoop "; getent passwd mapr'

clush -a -B 'echo "Root login "; getent passwd root'

exit

```

Change Permissions to Executable

```
chmod 755 cluster_verification.sh
```

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
```

Running STREAM Benchmark

The STREAM benchmark measures sustainable memory bandwidth (in MB/s) and the corresponding computation rate for simple vector kernels. To download the STREAM benchmark, go to: <http://www.cs.virginia.edu/stream/>

To run the STREAM benchmark, complete the following steps:

1. Log on to the admin node. Copy and extract STREAM file to each node (/root/).

```
clush -B -a "tar -xvf stream.tgz"
```

2. Change the permissions of the files in the stream directory

```
clush -a -b chmod 755 /root/stream/run.me.sh /root/stream/stream59-
icc-OMP-ff
```

3. Run the following command to run the STREAM benchmark on all nodes:

```
clush -B -a "/root/stream/runme.sh > /root/stream.log"
```

```
[root@rhel1 ~]# clush -B -a "/root/stream/runme.sh > /root/stream.log"
```

4. Run the following command to verify the results:
5. Extract the five lines of the result as shown and verify it on all the nodes.

```
$clush -B -a "grep -A5 \"Function      \" stream.log"
```

Function	Rate (MB/s)	Avg time	Min time	Max time
Copy:	72821.7286	0.0179	0.0176	0.0244
Scale:	90324.5251	0.0147	0.0142	0.0221
Add:	107077.2216	0.0185	0.0179	0.0235
Triad:	107629.6234	0.0182	0.0178	0.0223

rhell1

Function	Rate (MB/s)	Avg time	Min time	Max time
Copy:	72821.7286	0.0179	0.0176	0.0244
Scale:	90324.5251	0.0147	0.0142	0.0221
Add:	107077.2216	0.0185	0.0179	0.0235
Triad:	107629.6234	0.0184	0.0178	0.0223



Note: Results can vary based on the configuration.

Running MapR RPCtest

MapR RPCtest is a network bandwidth measurement test. In this solution the methodology adopted to verify the network bandwidth across the cluster requires configuring half the nodes as senders and the remaining half as receivers. This test is included in the MapR software available at `/opt/mapr/th/tools/rpctest` as part of the installation.

To run the RPCtest, complete the following steps:

1. Log on to the admin node and run the following commands to create the script:

```
#!/bin/bash

# Define sender nodes

# 8 servers in each rack act as servers and the other half as clients

senders=( 172.16.46.11 172.16.46.12 172.16.46.13 172.16.46.14 172.16.46.15
172.16.46.16 172.16.46.17 172.16.46.18)

for node in "${half1[@]"; do

ssh -n $node /opt/mapr/servers/tools/rpctest -server &

done

sleep 9 # let the servers set up
```

```
# Define receiver nodes

receivers=( 172.16.46.19 172.16.46.20 172.16.46.21 172.16.46.22
172.16.46.23 172.16.46.24 172.16.46.25 172.16.46.26)

i=0

for node in "${receivers[@]"; do

ssh -n $node "/opt/mapr/servers/tools/rpctest -client 5000 \ ${senders[$i]}
> rpctest.log" &

    ((i++))

done

#wait $!    # Comment/uncomment this to make it sequential/concurrent

sleep 5

tmp=${half1[@]}

clush -w ${tmp// /,} pkill rpctest
```

2. Run the runRPCtest.sh command from the admin node.

```
[root@rhell1 ~]# ./runRPCtest.sh
```

3. Results are generated on receiver nodes. Verify results for all the nodes.

```
$clush -B -w 172.16.46.[19-26] cat rpctest.log
-----
Rhell1
-----
10:48:21  rpcs 39867,  mb 2608.5
10:48:22  rpcs 39801,  mb 2608.4
Rate: 2606.09 mb/s, time: 2.01337 sec, #rpcs 80063, rpcs/sec 39765.7
-----
```



Note: Results can vary based on the topology and configuration.

Running IOzone Benchmark

IOzone is a filesystem benchmark that measures the performance of various I/O operations, such as read, write, re-read, re-write, fread, fwrite, random read and random write.



Warning: IOzone is data destructive. Do not run the test on disks with data.

To run the IOzone benchmark test, complete the following steps:

1. Download IOzone from <http://www.iozone.org/> and copy to all nodes at /root/.
2. Create the following script, run IOzone.sh on the admin node.

```
#!/bin/bash

# Parallel IOzone tests to stress/measure disk controller

# These tests are destructive therefore

# Test must be run BEFORE MapR filesystem is formatted with disksetup

# Run iozone command once on a single device to verify iozone command

D=$(dirname "$0")

abspath=$(cd "$D" 2>/dev/null && pwd || echo "$D")

# run iozone with -h option for usage, adjust path below for iozone location

# Set list of device names for the 'for' loop

lsblk -id | grep -o ^sd. | sort > /tmp/iozone.disks

for i in `lsblk -i | grep -B2 md[0-1] | grep -v '-' | awk '{print $1}'`; do
sed -i "/$i/d" /tmp/iozone.disks; done

disks=`cat /tmp/iozone.disks | xargs`

echo $disks

set -x

for disk in $disks; do

echo    $abspath/iozone -I -r 1M -s 80G -i 0 -i 1 -i 2 -f /dev/$disk >  $disk-
iozone.log&

sleep 3 #Some controllers seem to lockup without a sleep

done
```

3. Copy runIOzone.sh to all the nodes at location /root/.

4. Run the following command to start the test:

```
clush -B -a runIOzone.sh
```

5. Verify that the tests are running and wait for its completion.

```
clush -B -a "ps -aef | grep iozone | wc -l"
-----
rhel[1-16] (16)
-----
```

6. Run the following command to verify the test results.

The test result is generated for each disk as `sd<x>-iozone.log`, where `<x>` is the device id. These logs have sequential and random write and read latencies from each disks.

```
$ grep " 83886080 " sd*.log

sdb-iozone.log: 83886080      1024 97978 97951 100673 99254 49002 66552

sdc-iozone.log: 83886080      1024 101290 100745 97803 97006 48863 66671

sdd-iozone.log: 83886080      1024 94286 94937 96752 95872 48871 65605
```



Note Results can vary based on configuration.

Installing MapR

Installing MapR software across the cluster involves performing several steps on each node. To make the installation process simpler, start with the installation of core MapR components such as CLDB, MapR-FS, NFS gateway and Yarn. Any additional Hadoop ecosystem components can be easily installed by following instructions on <http://doc.mapr.com/display/MapR/Ecosystem+Guide>. This section will follow Table 10 role assignments for installation of services on the 16-node cluster.

The following sections describe the steps and options for installing MapR software:

- Preparing Packages and Repositories
- MapR Installation
- Installing MapR packages
- Verify successful installation
- Configure the Node with the `configure.sh` script
- Formatting Disks with the `disksetup` script

Planning the Cluster

The first step towards deploying the MapR is planning which nodes contribute to the cluster, and selecting the services that will run on each node.

MapR Services

In a typical cluster, most nodes are dedicated to data processing and storage, and a smaller number of nodes run services that provide cluster coordination and management. Some applications run on cluster nodes and others run on client nodes that can communicate with the cluster.

The following table shows some of the services that can be run on a node.

Table 10 below shows some of the MapR services and corresponding descriptions.

Table 10 MapR Services

MapReduce	Storage	Management	Application
Service	Description		
Warden	Warden runs on every node, coordinating the node's contribution to the cluster.		
NodeManager	Hadoop YARN NodeManager service. The NodeManager manages node resources and monitors the health of the node. It works with the ResourceManager to manage YARN containers that run on the node.		
FileServer	FileServer is the MapR service that manages disk storage for MapR-FS and MapR-DB on each node.		
CLDB	Maintains the container location database (CLDB) service. The CLDB service coordinates data storage services among MapR-FS FileServer nodes, MapR NFS gateways, and MapR clients.		
NFS	Provides read-write MapR Direct Access NFS access to the cluster, with full support for concurrent read and write access.		
MapR HBase Client (optional)	Provides access to MapR-DB tables via HBase APIs. Required on all client nodes that will access table data in MapR-FS		
ResourceManager	Hadoop YARN ResourceManager service. The ResourceManager manages cluster resources, and tracks resource usage and node health.		
ZooKeeper	Enables high availability (HA) and fault tolerance for MapR clusters by providing coordination.		
HistoryServer	Archives MapReduce job metrics and metadata.		
Web Server	Runs the MapR Control System.		
Pig	Pig is a high-level data-flow language and execution framework.		
Hive	Hive is a data warehouse that supports SQL-like ad hoc querying and data summarization.		
Flume	Flume is a service for aggregating large amounts of log data		
Oozie	Oozie is a workflow scheduler system for managing Hadoop jobs.		
Mahout	Mahout is a set of scalable machine-learning libraries that analyze user behavior.		
Spark	Spark is an processing engine for large datasets., you can set it up as standalone or managed by Yarn. In this CVD, we have Yarn manage spark		
Sqoop	Sqoop is a tool for transferring bulk data between Hadoop and relational databases.		

Node Types

The MapR installer categorizes nodes as *control* nodes (which runs only cluster management services to manage the cluster), *data* nodes, *control-as-data* nodes (which combine the functions of control and data nodes), or *client* nodes. For deployment of MapR on Cisco UCS Integrated Infrastructure for Big

Data, control services co-exist on data nodes (control-as-data node) as control services have a small footprint. Client node could be any node accessing the MapR cluster (all nodes in the MapR cluster are also client nodes).

Table 11 shows the Node Types and their descriptions

Table 11 Node Types

Node Type	Description
Data node	Used for processing data, they have FileServer and TaskTracker services installed. If MapR-DB or HBase is run on a data node, the HBase Client service is also installed. Data nodes are used for running YARN applications and MapReduce jobs, and for storing file and table data. These nodes run the FileServer service along with NodeManager (for YARN nodes), TaskTracker (for MapReduce nodes), and HBase client (for MapR-DB and HBase nodes).
Control-as-data node	Acts as both control and data nodes. They perform both functions and have both sets of services installed.
Client node	Provides access to the cluster so the user can communicate via the command line or the MapR Control System. Client nodes provide access to each node on the cluster so the user can submit jobs and retrieve data. A client node can be an edge node of the cluster, laptop, or any Windows machine.

Hostnames and Roles

This section describes the cluster plan of a 16-node cluster with hostnames and roles assignments for the following services as shown in below.

- ResourceManager (RM)
- HistoryServer (HS)
- NodeManager (NM)
- TaskTracker (TT, optional)
- JobTracker (JT, optional), FileServer (FS)
- Container Location Database (CLDB)
- Zookeeper
- Webserver



Note: Starting with MapR version 4.0, both Yarn and MapReduce V1 are supported not only in the same cluster but also on the same node.

Table 12 lists Host Names and Role Assignments.

Table 12 Lists Host Names and Role Assignments

Rack-1	MapR Roles
--------	------------

Hostnames	
rhel1	ZooKeeper
	CLDB,FS, NM, NFS,HS, Spark-history-server
rhel2	ZooKeeper
	CLDB, FS, NM, NFS
rhel3	ZooKeeper
	Webserver, FS, NM, RM NFS
rhel4	FS, NM, NFS, RM
rhel5	FS, NM, NFS, RM
rhel6	FS, NM, NFS
rhel7	FS, NM, NFS
rhel8	FS, NM, NFS
rhel9	FS, NM, NFS
rhel10	FS, NM, NFS
rhel11	FS, NM, NFS
rhel12	FS, NM, NFS
rhel13	FS, NM, NFS
rhel14	FS, NM, NFS
rhel15	FS, NM, NFS
rhel16	FS, NM, NFS



Note: All Job management is performed by the Resource Manager and Node Manager. In this CVD, Task Tracker and Job Tracker are not installed.

Preparing Packages and Repositories

A local repository on the admin node is set up to provide access to installation packages. With this method, the package manager on each node retrieves the installations package from the admin node (rhel1 is used as admin node as already mentioned) and installs the packages. Nodes do not need to have an internet access.

Below are instructions on setting up a local repository for Red Hat Linux distribution. These instructions create a single repository that includes both MapR components and the Hadoop ecosystem components.

RPM Repositories for MapR Core Software

MapR hosts `rpm` repositories for installing the MapR core software using Linux package management tools. For every release of the core MapR software, a repository is created for each supported platform.

These platform-specific repositories are hosted at:

<http://package.mapr.com/releases/<version>/<platform>>

<http://package.mapr.com/releases/v5.1.0/redhat/mapr-v5.1.0GA.rpm.tgz>

<http://archive.mapr.com/releases/ecosystem-all/redhat/mapr-ecosystem-5.x-20160729.rpm.tgz>

RPM Repositories for Hadoop Ecosystem Tools

MapR hosts `rpm` repositories for installing Hadoop ecosystem tools, such as Spark, Flume, Hive, Mahout, Oozie, Pig and Sqoop. At any given time, MapR's recommended versions of ecosystem tools that work with the latest version of MapR core software are available in the link below.

These platform-specific repositories are hosted at: <http://package.mapr.com/releases/ecosystem-5.x>

To create the local repositories, follow the steps below:

1. Login as `root` on the admin node (rhel1).
2. Create the following directory on rhel1

```
mkdir -p /var/www/html/mapr.local
```

3. On a node that is connected to the Internet, download the following files, substituting the appropriate `<version>` and `<datestamp>`:

```
wget http://package.mapr.com/releases/v<version>/redhat/mapr-  
v<version>GA.rpm.tgz
```

```
wget http://package.mapr.com/releases/ecosystem/redhat/mapr-ecosystem-  
<datestamp>.rpm.tgz
```



Note: For this document we use the version 5.1.0. See MapR Repositories and Package Archives for the correct paths for all past releases at <http://archive.mapr.com/releases/>

```
[root@LINUXJB ~]# wget http://package.mapr.com/releases/v5.1.0/redhat/mapr-  
v5.1.0GA.rpm.tgz
```

```
[root@LINUXJB ~]# wget http://package.mapr.com/releases/v5.1.0/redhat/mapr-v5.1.0GA.rpm.tgz
--2016-07-13 02:41:53-- http://package.mapr.com/releases/v5.1.0/redhat/mapr-v5.1.0GA.rpm.tgz
Resolving package.mapr.com... 52.84.243.50, 52.84.243.212, 52.84.243.118, ...
Connecting to package.mapr.com|52.84.243.50|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 858457665 (819M) [application/x-gzip]
Saving to: "mapr-v5.1.0GA.rpm.tgz"

57% [=====] 495,260,128 9.35M/s eta
```

```
[root@LINUXJB ~]# wget http://archive.mapr.com/releases/ecosystem-
all/redhat/mapr-ecosystem-5.x-20160729.rpm.tgz
```

```
[root@LinuxJB ~]# wget http://archive.mapr.com/releases/ecosystem-all/redhat/mapr-ecosystem-5.x-20160729.rpm.tgz
--2016-08-19 10:16:07-- http://archive.mapr.com/releases/ecosystem-all/redhat/mapr-ecosystem-5.x-20160729.rpm.tgz
Resolving archive.mapr.com... 50.19.226.224
Connecting to archive.mapr.com|50.19.226.224|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 6442369585 (6.0G) [application/x-gzip]
Saving to: "mapr-ecosystem-5.x-20160729.rpm.tgz"

0% [ ] 2,127,259 620K/s eta
```



Note: The server internet-host is an edge host that has access to the internet and to the admin node (rhel1). It is not a part of the MapR cluster. It is used to just download and transfer files to the admin node from the internet as the admin node is not directly connected to the internet.

4. Copy the files to /var/www/html/mapr.local on the admin node, and extract them there.

```
[root@LINUXJB ~]# scp mapr-v5.1.0GA.rpm.tgz rhel1:/var/www/html/mapr.local/
```

```
[root@LINUXJB ~]# scp mapr-ecosystem-5.x-20160729.rpm.tgz
rhel1:/var/www/html/mapr.local/
```

Connect to the admin (rhel1) node.

```
[root@rhel1 mapr.local]# tar -xvzf mapr-v5.1.0GA.rpm.tgz
```

```
[root@rhel1 mapr.local]# tar -xvzf mapr-ecosystem-5.x-20160729.rpm.tgz
```

Create the base repository headers:

```
[root@rhel1 mapr.local]# createrepo /var/www/html/mapr.local
```

```
[root@rhel1 mapr.local]# createrepo /var/www/html/mapr.local
Spawning worker 0 with 128 pkgs
Workers Finished
Gathering worker results

Saving Primary metadata
Saving file lists metadata
Saving other metadata
Generating sqlite DBs
Sqlite DBs complete
```

To add the repository on each node, complete the following steps:

1. Create repo file `/etc/yum.repos.d/maprtech.repo` on the admin node (rhel1):

```
vi /etc/yum.repos.d/maprtech.repo

[maprtech]

name=MapR Technologies, Inc.

baseurl=http://172.16.46.11/mapr.local

enabled=1

gpgcheck=0
```

```
[maprtech]
name=MapR Technologies, Inc.
baseurl=http://172.16.46.11/mapr.local
enabled=1
gpgcheck=0
```

2. Copy the `maprtech.repo` specification to all the nodes of the cluster. Then, update the yum metadata cache so that the repository files will be properly accessed.

```
clush -a -c /etc/yum.repos.d/maprtech.repo

clush -a yum makecache
```

```
clush a -c /etc/yum.repos.d/maprtech.repo
clush -a yum makecache
```

3. Create `mapr` user across all nodes

Users of the cluster must have the same credentials and user id on every node in the cluster. Each user (or department) that runs the MapR jobs needs an account and must belong to a common group (gid). If a directory service, such as LDAP, is not used, this user is created on each node. Every user must have the same uid and primary gid on every node.

In addition, a MapR user with full privileges to administer the cluster is created. If a user named 'mapr' does not exist. It is recommended that the user named 'mapr' is created in advance in order to test the connectivity issues prior to the installation step.

```
clush -a groupadd -g 5000 mapr
```

```
clush -a "useradd -g 5000 -u 5000 mapr"
```

```
clush -a -B "echo maprpassw0rd | passwd mapr --stdin"
```

```
clush -a groupadd -g 5000 mapr
clush -a "useradd -g 5000 -u 5000 mapr"
clush -a -B "echo maprpassw0rd | passwd mapr --stdin"
```



Note: Password of mapr user is set to maprpassw0rd

4. Verify mapr user on all nodes

```
clush -a -B id mapr
```

```
[root@rhell ~]# clush -a -B id mapr
uid=5000(mapr) gid=5000(mapr) groups=5000(mapr)
```

MapR Software Installation

Perform the following steps on each node:

1. Install the planned MapR services as shown in Table 13 .
2. Run the `configure.sh` script to configure the node.
3. Format raw drives and partitions allocated to MapR using the `disksetup` script.

Table 13 MapR Services and Packages

Service	Package
MapR core	mapr-core
Cluster location DB (CLDB)	mapr-cldb
History server	mapr-historyserver
ResourceManager and/or JobTracker	mapr-resourcemanager and/or mapr-jobtracker
MapR Control System	mapr-webserver
MapR File Server	mapr-fileserver
NFS	mapr-nfs

NodeManager and/or TaskTracker	mapr-nodemanager and/or mapr-tasktracker
ZooKeeper	mapr-zookeeper
Hadoop Ecosystem Components	Package
Drill	mapr-drill
Spark	mapr-spark
Hive	mapr-hive
Mahout	mapr-mahout
Oozie	mapr-oozie
Pig	mapr-pig
Sqoop	mapr-sqoop

Installing MapR packages

Use the commands in this section to install the appropriate packages for each node, based on the Cluster Plan, as shown in Table 13 above. Configuring the local yum repository ensures that the package dependencies will be managed correctly.

1. Install CLDB using the following command:

```
clush -B -w rhel[1,2] 'yum -y install mapr-cldb'
```

```
[root@rhel1 ~]# clush -B -w rhel[1,2] 'yum -y install mapr-cldb'
```

2. Install ResourceManager:

```
clush -B -w rhel[3,4,5] 'yum -y install mapr-resourcemanager'
```

```
[root@rhel1 ~]# clush -B -w rhel[3,4,5] 'yum -y install mapr-resourcemanager'
```

3. Install Mapr Webserver:

```
clush -B -w rhel3 'yum -y install mapr-webserver'
```

```
[root@rhel1 ~]# clush -B -w rhel3 'yum -y install mapr-webserver'
```



Note: Make sure httpd is not installed on these nodes

4. Install Mapr-Zookeeper:

```
clush -B -w rhel[1,2,3] 'yum -y install mapr-zookeeper'
```

```
[root@rhel1 ~]# clush -B -w rhel[1,2,3] 'yum -y install mapr-zookeeper'
```

5. Install MapR-historyserver:

```
clush -B -w rhell 'yum -y install mapr-historyserver'
```

6. Install NFS, Fileserver and Nodemanager on all cluster nodes:

```
clush -B -a 'yum -y install mapr-fileserver mapr-nfs mapr-nodemanager'
```

```
[root@rhell ~]# clush -B -a 'yum -y install mapr-fileserver mapr-nfs mapr-nodemanager'
```

7. To configure MapR nfs gateway service, run the following commands from the admin node (rhell):

```
clush -a mkdir -p /mapr
```

```
echo "localhost:/mapr /mapr hard,nolock" > /opt/mapr/conf/mapr_fstab
```

```
clush -a -c /opt/mapr/conf/mapr_fstab --dest /opt/mapr/conf/mapr_fstab
```

```
clush -a mkdir -p /mapr
echo "localhost:/mapr /mapr hard,nolock" > /opt/mapr/conf/mapr_fstab
clush -a -c /opt/mapr/conf/mapr_fstab --dest /opt/mapr/conf/mapr_fstab
```

Verification of Installation

To verify that the software has been installed successfully, check the `/opt/mapr/roles` directory on each node. The software is installed in directory `/opt/mapr` and a file is created in `/opt/mapr/roles` for every service that installs successfully. Examine this directory to verify installation for the node. For example:

```
# clush -a -B "ls -l /opt/mapr/roles"
```

1. Configure the Node with the `configure.sh` Script

The script `configure.sh` configures a node to be part of a MapR cluster, or modifies services running on an existing node in the cluster. The script creates (or updates) configuration files related to the cluster and the services running on the node. Before performing this step, make sure to have a list of the hostnames of the CLDB and ZooKeeper nodes. Optionally specify the ports for the CLDB and ZooKeeper nodes as well. If not specified, the default ports are assigned as:

- CLDB – 7222
- ZooKeeper – 5181

The script `configure.sh` takes an optional cluster name and log file, and comma-separated lists of CLDB and ZooKeeper host names or IP addresses (and optionally ports), using the following syntax:

```
/opt/mapr/server/configure.sh -C <host>[:<port>][,<host>[:<port>]...] -Z
<host>[:<port>][,<host>[:<port>]...] [-L <logfile>] [-N <cluster name>]
```

2. Configure nodes with CLDB, Zookeeper and History server Services

```
clush -B -a '/opt/mapr/server/configure.sh -C rhel1,rhel2 -Z rhel1,rhel2,rhel3 -
HS rhel1 -N ciscomapr -no-autostart'
```

Formatting Disks with the disksetup Script

mapr-fileserver is installed on all the nodes, use the following procedure to format disks and partitions to be used by MapR-FS.

The disksetup script is used to format disks to be used by the MapR cluster. The following script creates a text file /tmp/MapR.disks listing the disks and partitions to be used by MapR on the node. Each line lists a single disk.

Identify and Format the Data Disks for MapR

1. Create a list of disks to be formatted. (delete earlier instance of this file)
2. Create the following script on rhel1 and copy it to all the nodes

```
vi mapr_disks.sh

#!/bin/bash

#This script creates file (MapR.disks) containing list of non-os disk #drives
used during MapR Installation

[[ "-x" == "${1}" ]] && set -x && set -v && shift 1

count=1

for HD in /sys/class/scsi_host/host?/scan
do
echo '- - -' > ${HD}

done

for HD in /dev/sd*
do
if [[ -b ${HD} && `/sbin/parted -s ${HD} print quit|/bin/grep -c boot` -ne 0
]]
then
continue
elif [[ `echo ${HD} | grep -c 1 ` -ne 0
]]
then
```

```

continue

elif [[ `echo ${HD} | grep -c 2 ` -ne 0
]]

then

continue

else

echo $HD >> /tmp/MapR.disks

fi

done

```

3. Change the permission and copy mapr_disks.sh to all the nodes

```

chmod +x mapr_disks.sh

clush -a -c mapr_disks.sh

```

4. Run the mapr_disks.sh script on all the nodes.

```

clush -a -B /root/mapr_disks.sh

```

5. Verify the file on all nodes does not contain os drives

```

clush -aB cat /tmp/MapR.disks

```

6. Confirm that the disks are not in use. The cfdisk, mount, and pvdisplay utilities can be used to confirm that the system is not using the disks listed in /tmp/MapR.disks. This confirmation is not necessary during the initial setup, but may be relevant when nodes are removed or re-added to the cluster.

7. Format the disks to MapR-FS

```

clush -B -a "/opt/mapr/server/disksetup -F -W 5 /tmp/MapR.disks"

```



The script disksetup removes all data from the specified disks. Make sure to specify the disks correctly, and that all data has been backed up elsewhere.

This procedure assumes free, unmounted physical partitions or hard disks for use by MapR.

Update Environment Variables in /opt/mapr/conf/env.sh

There are a few key environment variables for the MapR software saved in /opt/mapr/conf/env.sh. These values must be properly configured BEFORE launching the cluster software. The default file is shown below:

```

#!/bin/bash

```



```
# Copyright (c) 2009 & onwards. MapR Tech, Inc., All rights reserved

# Please set all environment variable you want to be used during MapR cluster
# runtime here.

# namely MAPR_HOME, JAVA_HOME, MAPR_SUBNETS

#export JAVA_HOME=

#export MAPR_SUBNETS=

#export MAPR_HOME=

#export MAPR_ULIMIT_U=

#export MAPR_ULIMIT_N=

#export MAPR_SYSCTL_SOMAXCONN=
```

1. For this deployment, explicitly set the values for JAVA_HOME and MAPR_SUBNETS as shown below:

Edit the /opt/mapr/conf/env.sh file with the following environment variables as shown below:

```
export JAVA_HOME=/usr/java/jdk1.8.0_91/

export MAPR_SUBNETS=172.16.46.0/24,172.16.47.0/24
```



Note: By mentioning MAPR_SUBNETS and providing the two vlans, this enables MapR to use both VLANs (NICs) for traffic and thus using full 80 GiGE for Hadoop traffic.

2. Make those changes in rhel1:/opt/mapr/conf/env.sh and then distribute them to the entire cluster with the command

```
$ clush -B -a -c /opt/mapr/conf/env.sh
```

Bringing Up the Cluster

The installation of software across a cluster of nodes will go more smoothly if the services have been pre-planned and each node has been validated. Referring to the cluster design developed in section “Planning the Cluster”, ensure that each node has been prepared and that the MapR packages have been installed on each node in accordance with the plan. The process for launching the cluster can be broken down into several steps:

- Initialization Sequence
- Troubleshooting
- Installing the Cluster License

- Verifying Cluster Status

The initialization sequence involves starting the ZooKeeper service, starting the CLDB service, setting up the administrative user, and installing a MapR license. Once these initial steps are done, the cluster is functional on a limited set of nodes. Not all services are started yet, but the MapR Control System Dashboard, or the MapR Command Line Interface are available, to examine nodes and activity on the cluster.

Initialization Sequence

First, start the ZooKeeper service. It is important that all ZooKeeper instances start up, because the rest of the system cannot start unless a majority of ZooKeeper instances are up and running. Next, start the warden service on each node, or at least on the nodes that host the CLDB and webserver services. The warden service manages all MapR services on the node (except ZooKeeper) and helps coordinate communications. Starting the warden automatically starts the CLDB.

To bring up the cluster, complete the following steps:

1. Start ZooKeeper on all nodes where it is installed, by issuing the following command:

```
clush -B -w rhel[1,2,3] service mapr-zookeeper start
```

2. Verify that the ZooKeeper service is running properly :

```
clush -B -w rhel[1,2,3] service mapr-zookeeper status
```

The servers should display the running pid for the zookeeper process

3. On the nodes running CLDB or webserver, start the warden by issuing the following command:

```
clush -a service mapr-warden start
```



Note: Before continuing, wait 30 to 60 seconds for the warden to start the CLDB service. Calls to MapR (such as maprccli) may fail if executed before the CLDB has started successfully.

4. Log in to rhel1 and issue the following command to give full permission to the chosen administrative user mapr:

```
/opt/mapr/bin/maprccli acl edit -type cluster -user mapr:fc
```

Note: fc is full control.

5. Confirm that the MapR-FS is up by running the following command,

```
hadoop fs -ls /
```

```
drwxr-xr-x - mapr mapr 0 2015-04-28 22:55 /apps
drwxr-xr-x - mapr mapr 0 2015-04-28 22:55 /hbase
drwxr-xr-x - mapr mapr 0 2015-04-28 22:55 /opt
drwxrwxrwx - mapr mapr 3 2015-05-05 22:05 /tmp
drwxr-xr-x - mapr mapr 0 2015-04-28 22:55 /user
drwxr-xr-x - mapr mapr 1 2015-04-28 22:55 /var
```

Installing Spark

To install Spark on Yarn, complete the following steps:

1. Log in to rhel1 (admin node) and install the MapR-Spark package on all the nodes using the clush command.

```
clush -a yum -y install mapr-spark
```

```
[root@rhel1 ~]# clush -a yum -y install mapr-spark
```

2. Install the spark-history-server package on rhel1 server.

```
yum -y install mapr-spark-historyserver
```

```
[root@rhel1 ~]# yum -y install mapr-spark-historyserver
Loaded plugins: product-id, search-disabled-repos, subscription-manager
This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.
Resolving Dependencies
--> Running transaction check
--> Package mapr-spark-historyserver.noarch 0:1.6.1.201607242143-1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                               Arch             Version                               Repository        Size
=====
Installing:
mapr-spark-historyserver              noarch           1.6.1.201607242143-1                 maprtch           3.7 k
Transaction Summary
=====
Install 1 Package

Total download size: 3.7 k
Installed size: 683
Downloading packages:
mapr-spark-historyserver-1.6.1.201607242143-1.noarch.rpm | 3.7 kB 00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : mapr-spark-historyserver-1.6.1.201607242143-1.noarch          1/1
post-install called with argument '1'
post-transaction called with argument '0'
  Verifying  : mapr-spark-historyserver-1.6.1.201607242143-1.noarch          1/1
Installed:
mapr-spark-historyserver.noarch 0:1.6.1.201607242143-1
Complete!
```

3. Create the /apps/spark directory on MapR-FS and set the correct permissions on the directory as follows from the rhel1,

```
hadoop fs -mkdir /apps/spark
```

```
hadoop fs -chmod 777 /apps/spark
```

```
[root@rhel1 ~]# hadoop fs -mkdir /apps/spark
[root@rhel1 ~]# hadoop fs -chmod 777 /apps/spark
```


Verifying Cluster Status

Verify Cluster Status using the Web Interface

1. Log in to the MapR Control System.
2. Under the Cluster group in the left pane, click Dashboard.
3. Check the Services pane and make sure each service is running the correct number of instances, according to the cluster plan.

Verifying Cluster Status Using the Command Line Interface

1. Log in to a cluster node
2. Use the following command to list MapR services:

```
$ maprcli service list
```

```
$ maprcli license list
```

```
$ maprcli disk list -host <name or IP address>
```

```
[root@rhel1 ~]# maprcli disk list -host rhel3
diskname  powerstatus  status  vendor  hostname  modelnum  availablespace  storagepoolid  fstype  mount  firmwareversion
/dev/sda1  running      0       Cisco  rhel3     UCSC-MRAID12G  3664            1              ext4    1       4.25
/dev/sda2  running      0       Cisco  rhel3     UCSC-MRAID12G  1849359         1              ext4    1       4.25
/dev/sda3  running      0       Cisco  rhel3     UCSC-MRAID12G  1143257         1              swap    0       4.25
/dev/sdb   running      0       Cisco  rhel3     UCSC-MRAID12G  1143257         1              MapR-FS 0       4.25
/dev/sdd   running      0       Cisco  rhel3     UCSC-MRAID12G  1143257         1              MapR-FS 0       4.25
/dev/sde   running      0       Cisco  rhel3     UCSC-MRAID12G  1143249         2              MapR-FS 0       4.25
/dev/sdc   running      0       Cisco  rhel3     UCSC-MRAID12G  1143257         1              MapR-FS 0       4.25
/dev/sdf   running      0       Cisco  rhel3     UCSC-MRAID12G  1143249         2              MapR-FS 0       4.25
/dev/sdg   running      0       Cisco  rhel3     UCSC-MRAID12G  1143249         2              MapR-FS 0       4.25
```

Installing Additional Hadoop Components

The final step in installing a MapR cluster is to install and bring up Hadoop ecosystem components such as the following and integrating them with a MapR cluster:

Please refer to the MapR Install guide at <http://doc.mapr.com/display/MapR/Ecosystem+Guide> for detailed instructions on installation and configuration of desired Hadoop components.

- [Apache Drill](#) - Installing and using Drill on a MapR cluster
- [Flume](#) - Installing and using Flume on a MapR cluster
- [Hive](#) - Installing and using Hive on a MapR cluster, and setting up a MySQL metastore
- [Hue](#) - Installing and using Hue on MapR
- [Mahout](#) - Environment variable settings needed to run Mahout on MapR
- [Oozie](#) - Installing and using Oozie on a MapR cluster
- [Pig](#) - Installing and using Pig on a MapR cluster

- [Spark](#) - Installing and running Spark on MapR
- [Sqoop](#) - Installing and using Sqoop on a MapR cluster

Troubleshooting

Difficulty bringing up the cluster can be daunting, but most cluster problems are easily resolved. For the latest support tips, visit <http://answers.mapr.com>.

1. Can each node connect with the others? For a list of ports that must be open, see <http://answers.mapr.com>.
2. Is the warden running on each node? On the node, run the following command as root:

```
$ service mapr-warden status

WARDEN running as process 18732
```

3. If the warden service is not running, check the warden log file, `/opt/mapr/logs/warden.log`, for clues.
4. To restart the warden service run:

```
$ service mapr-warden start
```

The ZooKeeper service is not running on one or more nodes

5. Check the warden log file for errors related to resources, such as low memory
6. Check the warden log file for errors related to user permissions
7. Check for DNS and other connectivity issues between ZooKeeper nodes

The MapR CLI program `/opt/mapr/bin/maprcli` won't run

- Did you configure this node? See Installing MapR Software.
- Permission errors appear in the log

Check that MapR's changes to the following files have not been overwritten by automated configuration management tools:

<code>/etc/sudoers</code>	Allows the mapr user to invoke commands as root
<code>/etc/security/limits.conf</code>	Allows MapR services to increase limits on resources such as memory, file handles, threads and processes, and maximum priority level
<code>/etc/udev/rules.d/99-mapr-disk.rules</code>	Covers permissions and ownership of raw disk devices

Before contacting Support, collect cluster's logs using the `mapr-support-collect` script.

Conclusion

The MapR Converged Data Platform allows enterprises to build reliable, real-time applications by providing: a single cluster for streams, file storage database and analytics, persistence of streaming data, providing direct access to batch and interactive frameworks, a unified security framework for data-in-motion and data-at-rest with authentication, authorization and encryption, and a utility-grade reliability with self-healing and no single point-of-failure architecture.

The Cisco UCS® S3260 Storage Server for Big Data and Analytics with MapR Converged Data Platform enables the next-generation of big data architecture by providing simplified and centralized management, industry-leading performance, and a linearly scaling infrastructure and software platform.

Bill of Materials

This section provides the BOM for the 16 nodes. See Table 14 Bill of Materials for the Cisco UCS Fabric Interconnect 6332, Table 15 for the Cisco UCS S3260 Storage Server Base Rack, Table 16 Bill of Materials for Cisco UCS S3260 Storage Server Capacity Rack, and Table 18 for software components. Error! Reference source not found. lists MapR SKUs available from Cisco.

Table 14 Bill of Materials for Cisco UCS Fabric Interconnect 6332

Part Number	Description	Quantity
UCS-SP-FI6332-2X	(Not sold standalone) UCS 6332 1RU FI/No PSU/32 QSFP+ 2 required	2
UCS-SP-FI6332	UCS SP Select 6332 FI /No PSU/32 QSFP+	2
UCS-SP-FI6332	ONSITE 24X7X4 (Not sold standalone) UCS 6332 1RU FI/No PSU/3	2
QSFP-H40G-CU3M	40GBASE-CR4 Passive Copper Cable, 3m	16
QSFP-40G-SR-BD	QSFP40G BiDi Short-reach Transceiver	8
N10-MGT014	UCS Manager v3.1	2
UCS-FAN-6332	UCS 6332 Fan Module	8
UCS-ACC-6332	UCS 6332 Chassis Accessory Kit	2
CAB-9K12A-NA	Power Cord, 125VAC 13A NEMA 5-15 Plug, North America	4
UCS-PSU-6332-AC	UCS 6332 Power Supply/100-240VAC	4
QSFP-H40G-CU3M=	40GBASE-CR4 Passive Copper Cable, 3m	18
RACK-UCS2	Cisco R42610 standard rack w/side panels	1
RP208-30-1P-U-2=	Cisco RP208-30-U-2 Single Phase PDU 20x C13 4x C19 (Country Specific)	2
CON-UCW3-RPDUX	UC PLUS 24X7X4 Cisco RP208-30-U-X Single Phase PDU 2x (Country Specific)	6

Table 15 Bill of Materials for Cisco UCS S3260 Storage Server Base Rack

Part Number	Description	Quantity
-------------	-------------	----------

UCSS-S3260	Cisco UCS S3260 Base Chassis w/4x, SSD, Railkit	8
CAB-C13-C14-2M	Power Cord Jumper, C13-C14 Connectors, 2 Meter Length	32
UCS-C3K-HD4TB	UCS C3000 4TB NL-SAS 7200 RPM 12Gb HDD w Carrier- Top Load	48
UCSC-C3160-BEZEL	Cisco UCS C3160 System Bezel	8
UCSC-C3X60-RAIL	UCS C3X60 Rack Rails Kit	8
UCSC-PSU1-1050W	UCS C3X60 1050W Power Supply Unit	32
UCSC-C3K-M4SRB	UCS C3000 M4 Server Node for Intel E5-2600 v4	8
UCS-CPU-E52680E	2.40 GHz E5-2680 v4/120W 14C/35MB Cache/DDR4 2400MHz	16
UCS-MR-1X322RV-A	32GB DDR4-2400-MHz RDIMM/PC4-19200/dual rank/x4/1.2v	64
UCS-C3K-M4RAID	Cisco UCS C3000 RAID Controller M4 Server w 4G RAID Cache	8
UCSC-HS-C3X60	Cisco UCS C3X60 Server Node CPU Heatsink	16
UCSC-C3K-M4SRB	UCS C3000 M4 Server Node for Intel E5-2600 v4	8
UCS-CPU-E52680E	2.40 GHz E5-2680 v4/120W 14C/35MB Cache/DDR4 2400MHz	16
UCS-MR-1X322RV-A	32GB DDR4-2400-MHz RDIMM/PC4-19200/dual rank/x4/1.2v	64
UCS-C3K-M4RAID	Cisco UCS C3000 RAID Controller M4 Server w 4G RAID Cache	8
UCSC-HS-C3X60	Cisco UCS C3X60 Server Node CPU Heatsink	16
UCS-S3260-42HD4	Cisco UCS C3X60 Three row of drives containing 42 x 4TB (Tot	8
UCS-C3K-HD4TB	UCS C3000 4TB NL-SAS 7200 RPM 12Gb HDD w Carrier- Top Load	336
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-C3X60-G2SD48	UCSC C3X60 480GB Boot SSD (Gen 2)	32

Table 16 Bill of Materials for Cisco UCS S3260 Storage Server Capacity Rack

Part Number	Description	Quantity
UCSC-C3260	Cisco UCS C3260 Base Chassis w/4x PSU, SSD, Railkit	8
CAB-C13-C14-2M	Power Cord Jumper, C13-C14 Connectors, 2 Meter Length	32
UCSC-C3X60-HD8TB	UCSC 3X60 8TB NL-SAS 7.2K Helium HDD with HDD Carrier	48
UCSC-C3160-BEZEL	Cisco UCS C3160 System Bezel	8
UCSC-C3X60-RAIL	UCS C3X60 Rack Rails Kit	8
UCSC-PSU1-1050W	UCS C3X60 1050W Power Supply Unit	32
UCSC-C3K-M4SRB	UCS C3000 M4 Server Node for Intel E5-2600 v4	8
UCS-CPU-E52680E	2.40 GHz E5-2680 v4/120W 14C/35MB Cache/DDR4 2400MHz	16
UCS-MR-1X322RV-A	32GB DDR4-2400-MHz RDIMM/PC4-19200/dual rank/x4/1.2v	64
UCS-C3K-M4RAID	Cisco UCS C3000 RAID Controller M4 Server w 4G RAID Cache	8
UCSC-HS-C3X60	Cisco UCS C3X60 Server Node CPU Heatsink	16
UCSC-C3K-M4SRB	UCS C3000 M4 Server Node for Intel E5-2600 v4	8
UCS-CPU-E52680E	2.40 GHz E5-2680 v4/120W 14C/35MB Cache/DDR4 2400MHz	16
UCS-MR-1X322RV-A	32GB DDR4-2400-MHz RDIMM/PC4-19200/dual rank/x4/1.2v	64
UCS-C3K-M4RAID	Cisco UCS C3000 RAID Controller M4 Server w 4G RAID Cache	8
UCSC-HS-C3X60	Cisco UCS C3X60 Server Node CPU Heatsink	16
UCSC-C3X60-42HD8	UCS C3X60 3 rows of 8TB NL-SAS7200 RPM SAS-3 (42Total) 336TB	8
UCSC-C3X60-HD8TB	UCSC 3X60 8TB NL-SAS 7.2K Helium HDD with HDD Carrier	336
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-C3X60-G2SD48	UCSC C3X60 480GB Boot SSD (Gen 2)	32



Note: Both Cisco UCS S3260 Storage Server Basic Rack and Cisco UCS S3260 Storage Server Capacity Rack Bundle comes with 24 x 4TB Disk Drives, supports up to 28 x 6TB, 8TB and 10TB Disk drives also.

Table 17 Red Hat Enterprise Linux License

Red Hat Enterprise Linux		
RHEL-2S2V-3A	Red Hat Enterprise Linux	16
CON-ISV1-EL2S2V3A	3 year Support for Red Hat Enterprise Linux	16

Table 18 MapR Software Subscription License SKUs

MapR Software Subscription Licenses				
UCS-BD-MPRMCD-B=	UCS-BD-MCD-B-3Y	MCD-B-36	Base LIC MapR Conv. Ent. MapR-DB 3 Yr.	Base Subscription License for MapR Converged Enterprise Edition, includes base MapR-DB module. Includes Updates and Support. 3 Year Term. Price is Per Node.
UCS-BD-MPRMCH-B=	UCS-BD-MCH-B-3Y	MCH-B-36	Base LIC MapR Conv. Ent. Hadoop-3 Yr.	Base Subscription License for MapR Converged Enterprise Edition, includes base Hadoop module. Includes Updates and Support. 3 Year Term. Price is Per Node.
UCS-BD-MPRMCS-B=	UCS-BD-MCS-B-3Y	MCS-B-36	Base LIC MapR Conv. Ent., MapR Streams-3 Yr.	Base Subscription License for MapR Converged Enterprise Edition, includes base MapR Streams module. Includes Updates and Support. 3 Year Term. Price is Per Node.
UCS-BD-MPRAS-SL=	UCS-BD-AS-SL-3Y	AS-SL-36	Apache Spark, a fast and general engine large-scale data processing-3 Yr.	24/7 support for Apache Spark, a fast and general engine for large-scale data processing; Spark Core Engine, Shark, MLlib, Streaming and GraphX. Per Node Price for a 36-Month period.
UCS-BD-MPRHB-SL=	UCS-BD-HB-SL-3Y	HB-SL-36	Customer's use of Apache HBase-3 Yr.	24/7 Support for Customer's use of Apache HBase. Per Node Price for a 36-Month period.
UCS-BD-MPRIQS-SL=	UCS-BD-IQS-SL-3Y	IQS-SL-36	Customer's use of Impala-3 Yr.	24/7 Support for Customer's use of Impala - a low latency SQL query engine on Hadoop. Per Node Price for a 36-Month period.
UCS-BD-MPRSLR-SL=	UCS-BD-SLR-SL-3Y	SLR-SL-36	Apache SolR search option content on	Apache SolR search option for the content on the MapR Cluster. 3 Year Subscription

MapR Software Subscription Licenses				
			the MapR Cluster-3 Yr.	

About the Authors

Manan Trivedi is a Technical Marketing Engineer in the Data Center Solutions Group, Cisco Systems Inc. He is part of the solution engineering team focusing on big data infrastructure and performance.

James Sun, Senior Solutions Architect (MapR Technologies). James Sun manages the technological relationship with worldwide alliances at MapR Technologies. James has over 15 years of experience in information technology. Prior to MapR, he held several senior technical positions at technological companies such as NetApp, Yahoo and EMC. He holds a PhD. from Stanford University

Acknowledgements

- Jayanth Regula, Hadoop Admin in the Data Center Solutions Group at Cisco Systems
- Karthik Kulkarni, Big Data Solutions Architect, Data Center Solutions Group, Cisco Systems Inc.
- Barbara Dixon, Technical Writer, Data Center Solutions Group, Cisco Systems, Inc.