

Cisco UCS Integrated Infrastructure for Big Data with Splunk Enterprise

Last Updated: September 23, 2016





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	8
Solution Overview	9
Audience	9
Purpose of the Document	9
Technology Overview	9
Cisco UCS 6200 Series Fabric Interconnects	10
Cisco UCS 6300 Series Fabric Interconnects	10
Cisco UCS C-Series Rack Mount Servers	10
Cisco UCS C3260 Rack Server	11
Cisco UCS Virtual Interface Cards (VICs)	12
Solution Design	13
Cisco UCS Manager	13
Splunk for Big Data Analytics	13
Key Features of Splunk Enterprise	13
Deployment Hardware and Software	15
Architecture	15
Rack and PDU Configuration	15
Port Configuration on Fabric Interconnects	17
Configuration and Cabling for Cisco UCS C240 M4 Rack Servers	17
Configuration and Cabling for Cisco UCS C220 M4 Rack Servers	18
Configuration and Cabling for the Cisco C3260 Rack Server	19
Rack Appearance	20
Software Distributions and Versions	21
Fabric Configuration	22
Performing Initial Setup of Cisco UCS 6296 Fabric Interconnects	22
Configure Fabric Interconnect A	22
Configure Fabric Interconnect B	23
Logging Into Cisco UCS Manager	24
Upgrading UCSM Software to Version 3.1(2b)	24
Adding a Block of IP Addresses for KVM Access	24
Enabling Uplink Ports	25
Configuring VLANs	26
Enabling Server Ports	28
Creating a Storage Profile for Boot Drives	29
Creating Pools for Service Profile Templates	32
Creating an Organization	32
Creating MAC Address Pools	32

Creating Server Pools	34
Creating Policies for Service Profile Templates	36
Creating Host Firmware Package Policy	36
Creating QoS Policies	37
Creating Local Disk Configuration Policy.....	39
Creating Server BIOS Policy	40
Creating Boot Policy.....	43
Creating Power Control Policy.....	45
Creating a Maintenance Policy.....	47
Creating Chassis Profiles for C3260 Rack Servers	49
Creating Disk Zoning Policy	49
Creating Chassis Firmware Package Policy.....	51
Creating a Chassis Profile Template	53
Associating Chassis Profile to Individual Chassis	56
Creating a Service Profile Template.....	58
Configuring the Storage Policy for the Template	59
Configuring Network Settings for the Template	60
Configuring SAN Connectivity for the Template	64
Configuring vNIC/vHBA Placement Policy for the Template.....	65
Configuring vMedia Policy for the Template	66
Configuring Server Boot Order for the Template	66
Configuring the Maintenance Policy for the Template	67
Configuring Server Assignment for the Template	68
Configuring Operational Policies for the Template	69
Creating a Service Profile Template for C3260	70
Creating Service Profiles from Template.....	76
Identifying the Servers.....	77
Installing Red Hat Enterprise Linux 6.8 on C220 M4 Systems	79
Creating a Virtual Drive Using Cisco 12G SAS RAID Controller Utility.....	79
Installing the Operating System.....	84
Installing Red Hat Enterprise Linux 6.8 using Software RAID on C240 M4 Systems	93
Installing Red Hat Enterprise Linux 6.8 on the C3260.....	111
Post OS Install Configuration.....	112
Configuring /etc/hosts.....	112
Setting Up Password-less Login.....	113
Setting up ClusterShell.....	114
Creating Red Hat Enterprise Linux (RHEL) 6.8 Local Repo	115
Creating the Red Hat Repository Database	116
Installing httpd	117
Verify Cisco Network Driver for VIC1227.....	118

Disabling SELinux	119
Disabling the Linux Firewall	119
Installing xfsprogs.....	120
NTP Configuration.....	120
Enabling Syslog.....	122
Setting Ulimit	122
Set TCP Retries	123
Configure VM Swapping.....	124
Disable IPv6 Defaults	124
Disable Transparent Huge Pages	124
Installing LSI StorCLI utility on all Indexers and Archival Nodes	125
Configuring the Virtual Drive on the Indexers	125
Configuring the XFS File System	126
Configuring Data Drives on Archival Nodes	128
Configuring the XFS File System	129
Cluster Verification	132
Installing Splunk Enterprise 6.4	137
Splunk Architecture and Terminology.....	137
Splunk Services and Processes	138
Planning the Installation	138
Installing Splunk	140
Setting Up Login for Splunk User	142
Starting Splunk Enterprise Cluster	143
Logging in for the First Time.....	144
Creating User Accounts	145
Initializing Splunk on Boot	145
Default Ports.....	145
NFS Configurations for Splunk Frozen Data Storage.....	146
Create the User Splunk in the Storage Servers.....	146
NFS Server Setup on Archival Nodes	146
Scenario A: 4 Indexers, 6 Indexers, 8 Indexers (or) 16 Indexers with only One Storage Node	146
Scenario B: 8 Indexers (or) 16 Indexers with Two Storage Nodes	147
NFS Client Configurations on the Indexers	149
Configuring the Splunk Enterprise Cluster	153
Configuring Splunk Enterprise Licenses.....	153
Setup License Master.....	153
Configure the Indexers, Search Heads, and Admin Nodes as License Slaves	155
Configure all the License Slaves at Once Using CLI (clush)	155
(Optional) Configure License Slaves Individually Using the Web Interface	156
Verifying License-Slave Relationships	158

Configuring the Master Node (aka: Cluster Master)	160
Configure Indexing Peers	163
Configuring Indexer Clusters	163
Configure All Indexing Peers Using CLI (clush).....	163
Configure Indexing Peers Individually Using the Web Interface (Optional)	164
Setting Dedicated Replication Address	169
Verify Cluster Configuration.....	170
Configure Receiving on the Peer Nodes	170
Configure Master to Forward All its Data to the Indexer Layer	173
Configure Search Head Clustering	174
Add Search Heads to Master Node	174
Configure the Deployer.....	177
Configure Search Head Cluster Members	178
Elect a Search Head Captain	179
Configure Search Heads to Forward their Data to the Indexer Layer	180
Configure Search Head Load-Balancing	182
Configuring the Distributed Management Console	185
Configure Search Heads in Distributed Management Console	187
Configuring Archive of Data from Cold to Frozen	190
Configuring the Deployment Server	191
Installing a Universal Forwarder on a Test Server.....	191
Register Universal Forwarder with the Deployment Server	192
Configure an App within the Deployment Server.....	192
Installation Verification	196
Verifying from DMC	196
Verifying Master and Peer Replication	198
Verifying Data Replication	201
Verifying Transfer of Frozen Buckets to Archival Storage	204
Post-Test Cleanup.....	205
Removing Test Data.....	205
Removing Test Indexes.....	205
Removing the Universal Forwarder	207
Remove Deployment Server App	207
Hardening the Splunk Installation.....	207
Turn Off Web Servers.....	207
Best Practices for Onboarding Data	209
The Universal Forwarder.....	209
Advantages of the Splunk Universal Forwarder	209
What About Systems Where the Splunk Universal Forwarder is Not Supported?.....	210
Additional Terminology.....	211

Recommended Apps and Add-ons for Data Collection	212
Conclusion	214
Bill of Materials	215
About Authors	219
Acknowledgements	219
Appendix A	220
Provisioning a Splunk Cluster with UCSDE	220
Requirements	220
Creating MAC Address Pools	220
Creating MAC Pool Window	221
Specifying First MAC Address and Size	222
Creating Server Pools	222
Communication Settings	224
QOS System Class	225
UCS Director Express Deployment and Configuration	225
Baremetal Agent Deployment and Configuration	230
License Administration	233
License Administration	236
License Administration	238
Building the Software Catalog	241
Creating IP Pools	245
Creating an Instant Splunk Cluster	247

Executive Summary

Traditional tools for managing and monitoring IT infrastructures are out of step with the constant change happening in today's data centers. When problems arise, finding the root cause or gaining visibility across the infrastructure to pro-actively identify and prevent outages is nearly impossible. Virtualization and cloud infrastructures introduce additional complexity, resulting in an environment that is more challenging to control and manage.

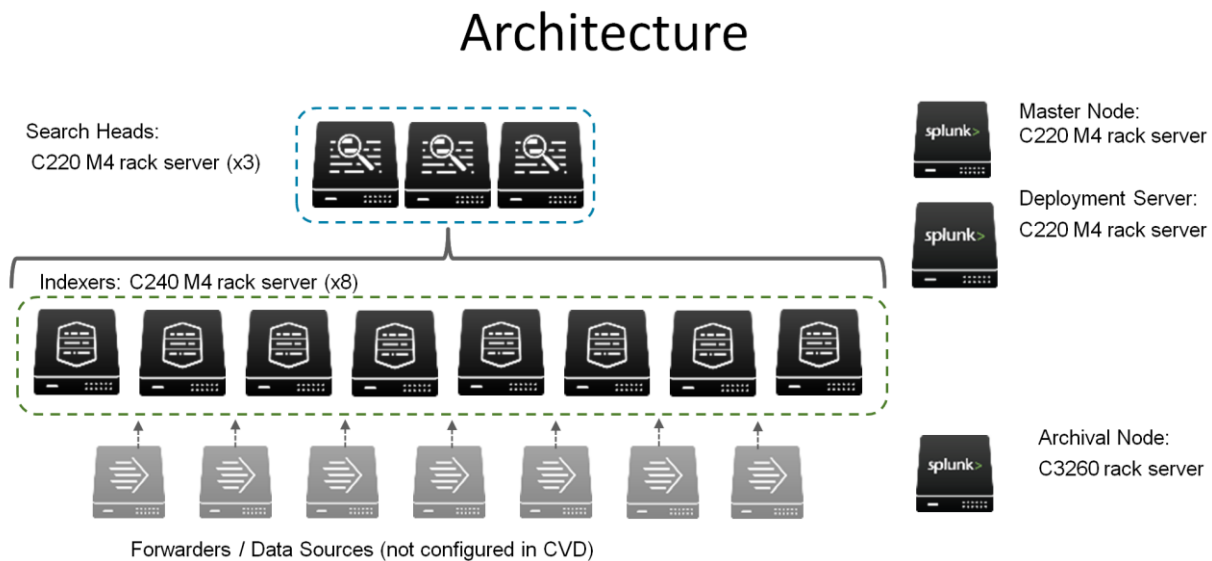
Splunk software reliably collects and indexes machine data, from a single source to tens of thousands of sources, all in real time. Organizations typically start with Splunk to solve a specific problem, and then expand from there to address a broad range of use cases, such as application troubleshooting, IT infrastructure monitoring, security, business analytics, Internet of Things, and many others. As operational analytics become increasingly critical to day-to-day decision-making and Splunk deployments expand to terabytes of data, a high-performance, highly scalable infrastructure is critical to ensuring rapid and predictable delivery of insights. Cisco UCS's ability to expand to thousands of servers allows the Splunk deployments to scale horizontally while continuously delivering exceptional performance.

The Cisco Validated Design (CVD) for Splunk Enterprise describes the architecture and deployment procedures for Splunk Enterprise on a Distributed High Performance reference architecture based on Cisco UCS Integrated Infrastructure for Big Data (see Distributed Splunk Reference Architecture Solution Brief). The configuration consists of eight (8) Cisco UCS C240 M4 rack servers as indexers, three (3) Cisco C220 M4 rack servers as search heads and two (2) Cisco C220 M4 rack servers to perform administrative functions, along with one (1) archival node (Cisco UCS C3260 rack servers) for frozen data.

Solution Overview

This CVD describes architecture and deployment procedures for Splunk Enterprise using eight (8) Cisco UCS C240 M4 rack servers as indexers, three (3) Cisco UCS C220 M4 rack servers as search heads, and two (2) Cisco UCS C220 M4 rack servers to perform administrative functions, along with 1 archival server for frozen data (Cisco UCS C3260 rack server). This architecture is based on the Cisco UCS Integrated Infrastructure for Big Data with Splunk. The reference architecture named Distributed Deployment with High Capacity consists of 16 indexers for storage, of which 8 are considered for this CVD as well as an additional archival server for attached storage. The solution goes into detail configuring distributed search on Splunk Enterprise platform along with the Archival node (Cisco UCS C3260 rack server with one server blade). Figure 1 shows the architecture for the Splunk Enterprise deployment.

Figure 1 Clustered Distributed Search Deployment Architecture of Splunk Enterprise



Audience

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 Splunk Enterprise on Cisco UCS Integrated Infrastructure for Big Data.

Purpose of the Document

This CVD offers a dependable deployment model for Splunk Enterprise which can be implemented rapidly and customized to meet Splunk requirements. The configuration detailed in the document can be extended to larger clusters. In this CVD, eight Splunk Indexers provide capacity to index up to 2 TB of data per day (without replication). This configuration can scale to index hundreds of terabytes to petabytes of data every 24 hours, delivering real-time search results and meeting Splunk application demands with seamless data integration and analytics to multiple users across the globe.

Technology Overview

The Cisco UCS solution for Splunk Enterprise is based on Cisco UCS Integrated Infrastructure for Big Data and Analytics, a highly scalable architecture designed to meet a variety of scale-out application demands with seamless data integration and management integration capabilities built using the following components:

Cisco UCS 6200 Series Fabric Interconnects

Cisco UCS 6200 Series Fabric Interconnects provide high-bandwidth, low-latency connectivity for servers, with integrated, unified management provided for all connected devices by Cisco UCS Manager. Deployed in redundant pairs, Cisco Fabric Interconnects offer the full active-active redundancy, performance, and exceptional scalability needed to support the large number of nodes that are typical in clusters serving big data applications. Cisco UCS Manager enables rapid and consistent server configuration using service profiles, automating ongoing system maintenance activities such as firmware updates across the entire cluster as a single operation. Cisco UCS Manager also offers advanced monitoring with options to raise alarms and send notifications about the health of the entire cluster.

The Cisco UCS 6296UP 96-Port Fabric Interconnect is shown in Figure 2

Figure 2 Cisco UCS 6296UP 96-Port Fabric Interconnect



Cisco UCS 6300 Series Fabric Interconnects

Cisco UCS 6300 Series Fabric Interconnects 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. Figure 3 shows the Cisco UCS 6332 UP-Port Fabric Interconnect.

Figure 3 Cisco UCS 6332 UP 32-Port Fabric Interconnect



Note: This Cisco Validated Design is built using second generation Fabric Interconnects (Cisco UCS 6296, as shown above), but the Cisco UCS 6332 can be used as well.

Cisco UCS C-Series Rack Mount Servers

Cisco UCS C-Series Rack Mount C220 M4 High-Density Rack servers (Small Form Factor Disk Drive Model) and Cisco UCS C240 M4 High-Density Rack servers (Small Form Factor Disk Drive Model) are enterprise-

class systems that support a wide range of computing, I/O, and storage-capacity demands in compact designs. Cisco UCS C-Series Rack-Mount Servers are based on the Intel Xeon E5-2600 v4 product family with 12-Gbps SAS throughput, delivering significant performance and efficiency gains over the previous generation of servers. The servers use dual Intel Xeon processor E5-2600 v4 series CPUs and support up to 768 GB of main memory (128 or 256 GB is typical for big data applications) and a range of disk drive and SSD options. The Performance-optimized option supports 24 Small Form Factor (SFF) disk drives. The Capacity-optimized option supports 12 Large Form Factor (LFF) disk drives, along with 4 Gigabit Ethernet LAN-on-motherboard (LOM) ports. Cisco UCS virtual interface cards 1227 (VICs) designed for the M4 generation of Cisco UCS C-Series Rack Servers are optimized for high-bandwidth and low-latency cluster connectivity, with support for up to 256 virtual devices that are configured on demand through Cisco UCS Manager. The Cisco UCS C240 M4 Rack Server is shown in Figure 4

Figure 4 Cisco UCS C240 M4 Rack Server



Cisco UCS C3260 Rack Server

Cisco UCS C3260 Rack Server is an advanced, modular rack server with extremely high storage density. Based on the Intel Xeon processor E5-2600 v4 series, it offers up to 560 TB of local storage in a compact 4-rack-unit (4RU) form factor. With its individually hot-swappable hard-disk drives, and its built-in enterprise-class Redundant Array of Independent Disks (RAID) redundancy, the Cisco UCS C3260 helps you achieve the highest levels of data availability. The Cisco UCS C3260 is ideal for Snapshots, active archiving, compliance, media storage, and distributed file systems for scenarios in which high storage capacity is important. Cisco UCS virtual interface cards 1300 (VICs) designed for the M4 generation of Cisco UCS C-Series Rack Servers and C3260 are optimized for high-bandwidth and low-latency cluster connectivity, with support for up to 256 virtual devices that are configured on demand through Cisco UCS Manager. The Cisco UCS C3260 Server is shown in Figure 5

Figure 5 Cisco UCS C3260 Server



Cisco UCS Virtual Interface Cards (VICs)

Cisco UCS Virtual Interface Cards (VICs), unique to Cisco, incorporate next-generation converged network adapter (CNA) technology from Cisco, and offer dual 10-Gbps ports designed for use with Cisco UCS C-Series Rack-Mount Servers. Optimized for virtualized networking, these cards deliver high performance and bandwidth utilization and support up to 256 virtual devices. The Cisco UCS Virtual Interface Card (VIC) 1227 is a dual-port, Enhanced Small Form-Factor Pluggable (SFP+), 10 Gigabit Ethernet and Fiber Channel over Ethernet (FCoE)-capable, PCI Express (PCIe) modular LAN on motherboard (mLOM) adapter. Cisco VIC 1227 provide dual 10 Gigabit Ethernet. The Cisco VIC 1387 can also be used in conjunction with 3rd generation Cisco UCS Fabric Interconnects 6332 for taking advantage of 40 Gigabit Ethernet connectivity. The System IO Controller (SIOC) with VIC1300 on C3260 can work as dual 40 Gigabit Ethernet or dual 10 Gigabit Ethernet ports (with appropriate QSFP to SFP+ convertors).

Figure 6 Cisco UCS VIC 1227



Figure 7 Cisco UCS VIC 1387

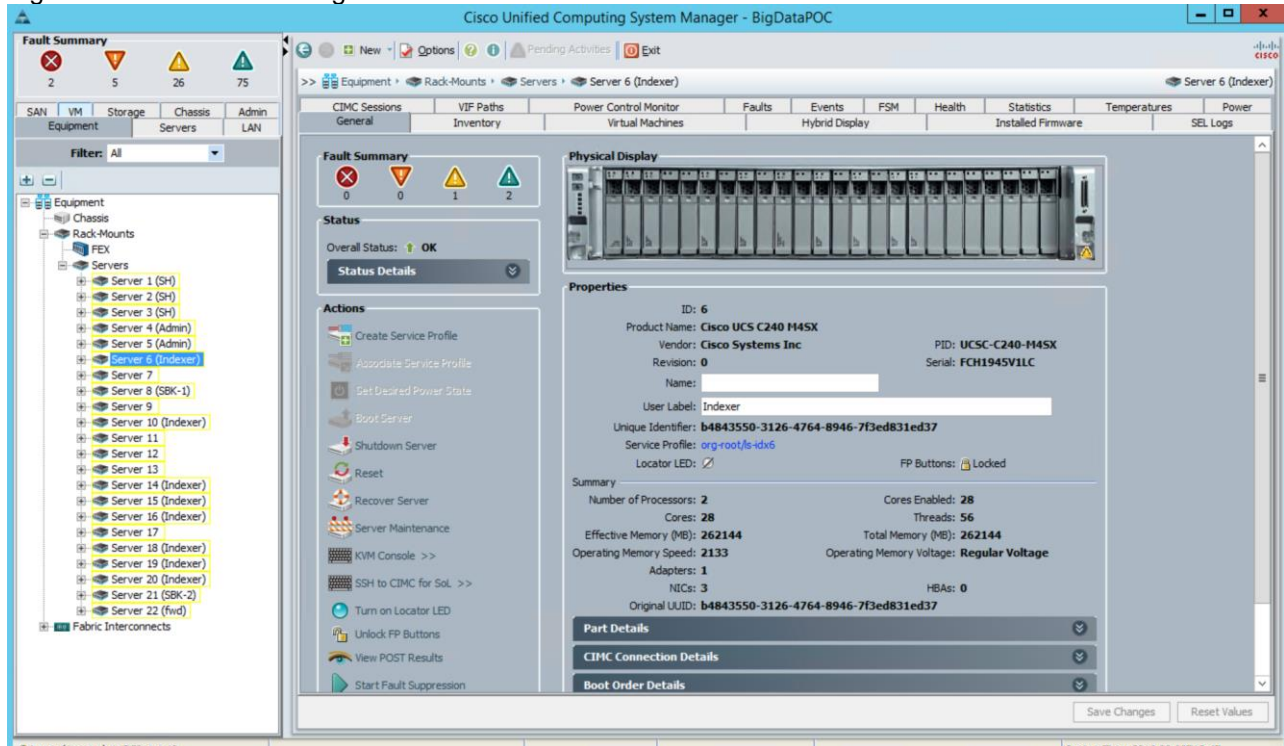


Solution Design

Cisco UCS Manager

Cisco UCS Manager resides within the Cisco UCS 6200 Series Fabric Interconnects. 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), a command-line interface (CLI), or an XML application-programming interface (API). Cisco UCS Manager uses service profiles to define the personality, configuration, and connectivity of all resources within Cisco UCS, radically simplifying provisioning of resources so that the process takes minutes instead of days. This simplification allows IT departments to shift their focus from constant maintenance to strategic business initiatives.

Figure 8 Cisco UCS Manager



Splunk for Big Data Analytics

All your IT applications, systems, and technology infrastructure generate data every millisecond of every day. This machine data is one of the fastest growing, most complex areas of big data. It is also one of the most valuable, containing a definitive record of user transactions, customer behavior, sensor activity, machine behavior, security threats, fraudulent activity, and more.

Splunk Enterprise provides a holistic way to organize and extract real-time insights from massive amounts of machine data from virtually any source. This includes data from websites, business applications, social media platforms, app servers, hypervisors, sensors, traditional databases, and open source data stores. Splunk Enterprise scales to collect and index tens of terabytes of data per day, across multi-geography, multi-datacenter, and hybrid cloud infrastructures.

Key Features of Splunk Enterprise

Splunk Enterprise provides the end-to-end, real-time solution for machine data, delivering the following core capabilities:

- Universal collection and indexing of machine data, from virtually any source

- Powerful search processing language (SPL) to search and analyze real-time and historical data
- Real-time monitoring for patterns and thresholds; real-time alerts when specific conditions arise
- Powerful reporting and analysis
- Custom dashboards and views for different roles
- Resilience and horizontal scalability
- Granular role-based security and access controls
- Support for multi-tenancy and flexible, distributed deployments on-premises or in the cloud
- Robust, flexible platform for big data apps

Deployment Hardware and Software

Architecture

Reference Architecture for the Splunk Enterprise deployment is shown in Table 1

Table 1 Cisco UCS Reference Architecture for Splunk Enterprise (with Archival Nodes)

Indexer	8 Cisco UCS C240 M4 Rack Servers, each with: <ul style="list-style-type: none"> • 2 Intel Xeon processor E5-2680 v4 CPUs (28 cores) • 256 GB of memory • Cisco 12-Gbps SAS modular RAID controller with 2-GB flash-backed write cache • Cisco UCS VIC 1227 • 24 1.8-TB 10K SFF SAS drives in a RAID10 configuration • 2 120-GB (or 240-GB) SSDs for the operating system
Search head	3 Cisco UCS C220 M4 Rack Servers, each with: <ul style="list-style-type: none"> • 2 Intel Xeon processor E5-2680 v4 CPUs (28 cores) • 256 GB of memory • Cisco 12-Gbps SAS modular RAID controller with 2-GB flash-backed write cache • Cisco UCS VIC 1227 • 2 600-GB 10K SFF SAS drives
Administration and master nodes	2 Cisco UCS C220 M4 Rack Servers, each with: <ul style="list-style-type: none"> • 2 Intel Xeon processor E5-2620 v4 • 128 GB of memory • Cisco 12-Gbps SAS modular RAID controller with 2-GB flash-backed write cache • Cisco UCS VIC 1227 • 2 600-GB 10K SFF SAS drives
Archival Storage (Frozen data)	1 Cisco UCS C3260 Rack Server with: <ul style="list-style-type: none"> • 2 Intel Xeon processor E5-2620 v4 CPUs (16 cores) • 256 GB of memory • Cisco 12-Gbps SAS modular RAID controller with 4-GB flash-backed write cache • Cisco UCS VIC 1227 • 2 600-GB 10K SFF SAS drives • 60 X 4TB 7200 RPM drives
Networking	2 Cisco UCS 6296UP 96-Port Fabric Interconnects
Recommended indexing capacity	Up to 2 TB per day
Retention capability	2 TB per day with 4 month retention
Recommended indexing capacity with replication	Up to 1 TB per day
Total storage capacity	172 TB
Servers	14
Rack space	29 RU

Rack and PDU Configuration

The rack consists of two vertical power distribution units (PDU), two Cisco UCS 6296UP Fabric Interconnects, eight Cisco UCS C240 M4 servers, five Cisco UCS C220 M4 servers, and one Cisco UCS C3260 server. All the devices are connected to each of the vertical PDUs for redundancy, thereby ensuring availability during power source failure.

Table 2 describes the rack configurations used in this CVD.

Table 2 Rack Configurations

Cisco 42 RU Rack	Master Rack
------------------	-------------

Cisco 42 RU Rack	Master Rack
42	Cisco UCS FI 6296UP
41	
40	Cisco UCS FI 6296UP
39	
38	Cisco UCS C220 M4
37	Cisco UCS C220 M4
36	Unused
35	Cisco UCS C220 M4
34	Cisco UCS C220 M4
33	Cisco UCS C220 M4
32	Cisco UCS C240 M4
31	
30	Cisco UCS C240 M4
29	
28	Cisco UCS C240 M4
27	
26	Cisco UCS C240 M4
25	
24	Cisco UCS C240 M4
23	
22	Cisco UCS C240 M4
21	
20	Cisco UCS C240 M4
19	
18	Cisco UCS C240 M4
17	
16	Unused
15	Unused

Cisco 42 RU Rack	Master Rack
14	Unused
13	Unused
12	Unused
11	Unused
10	Unused
9	Unused
8	Unused
7	Unused
6	Unused
5	Unused
4	Cisco UCS C3260
3	
2	
1	

Port Configuration on Fabric Interconnects

Table 3 shows the network connectivity configurations used for developing this CVD.

Table 3 Port Types and Port Numbers

Port Type	Description	Port Number
Network	Uplink port	1
Server	Cisco UCS C220 M4 Servers	4 to 8
Server	Cisco UCS C240 M4 Servers	9 to 16
Server	Cisco C3260 Server	17

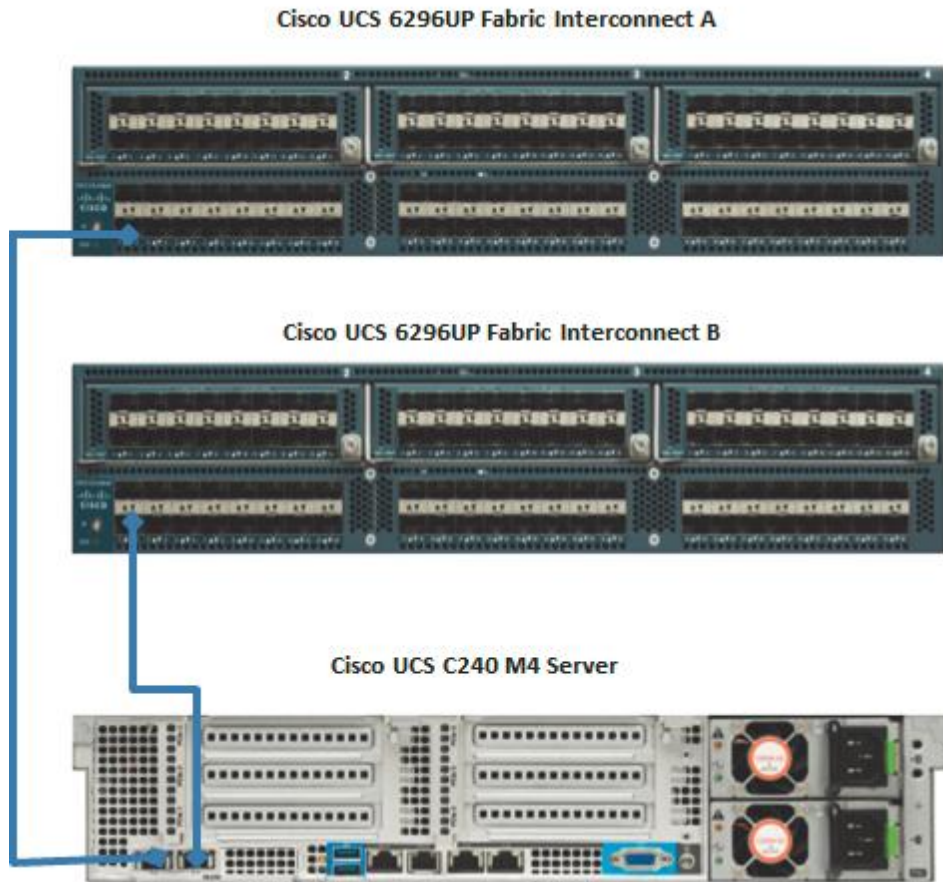
Configuration and Cabling for Cisco UCS C240 M4 Rack Servers

The Cisco UCS C240 M4 rack server is equipped with Intel Xeon E5-2680 v4 processors, 256 GB of memory, Cisco UCS Virtual Interface Card 1227, Cisco 12-Gbps SAS Modular Raid Controller with 2-GB FBWC, twenty-four 1.8-TB 10K SFF SAS drives, and two 120-GB SATA SSD for Boot.

All eight servers of this category are directly connected to the ports on the Cisco UCS FI 6296 Fabric Interconnects as shown below. These ports are configured as server ports in the Cisco UCS Manager.

Figure 9 illustrates the port connectivity between the Fabric Interconnect and Cisco UCS C240 M4 server. Eight Cisco UCS C240 M4 servers are used as indexers in this rack configuration.

Figure 9 Fabric Topology for Cisco UCS C240 M4 Rack Server



Configuration and Cabling for Cisco UCS C220 M4 Rack Servers

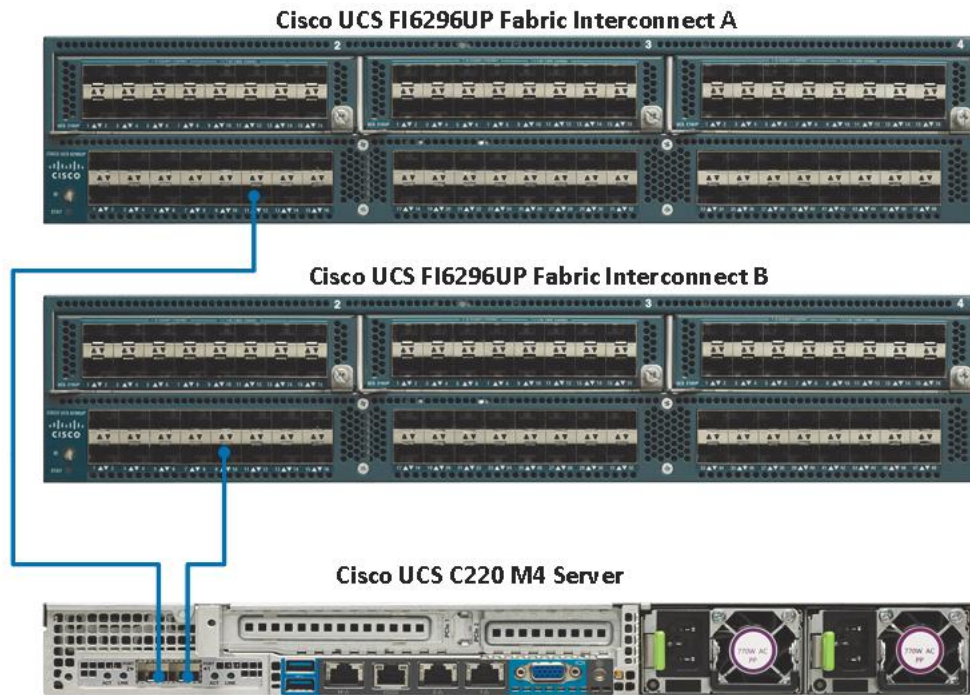
This solution makes use of five C220 M4 rack servers that are configured with two different classes of CPUs.

- The servers that function as the search heads are equipped with Intel Xeon E5-2680 v4 processors, 256 GB of memory, Cisco UCS Virtual Interface Card 1227, Cisco 12-Gbps SAS Modular Raid Controller with 2-GB FBWC, and 2 600 GB 10K SFF SAS drives.
- The servers that function as the admin nodes are equipped with Intel Xeon E5-2620 v4 processors, 128 GB of memory, Cisco UCS Virtual Interface Card 1227, Cisco 12-Gbps SAS Modular Raid Controller with 2-GB FBWC, and 2 600 GB 10K SFF SAS drives.

All five servers of this category are directly connected to the ports on the Cisco UCS Fabric Interconnects 6296 Fabric Interconnects as shown below. These ports are configured as server ports in the UCS Manager.

Figure 10 illustrates the port connectivity between the Fabric Interconnect and Cisco UCS C220 M4 servers. Five Cisco UCS C220 M4 servers are used in the rack configuration.

Figure 10 Fabric Topology for Cisco UCS C220 M4 Rack Server



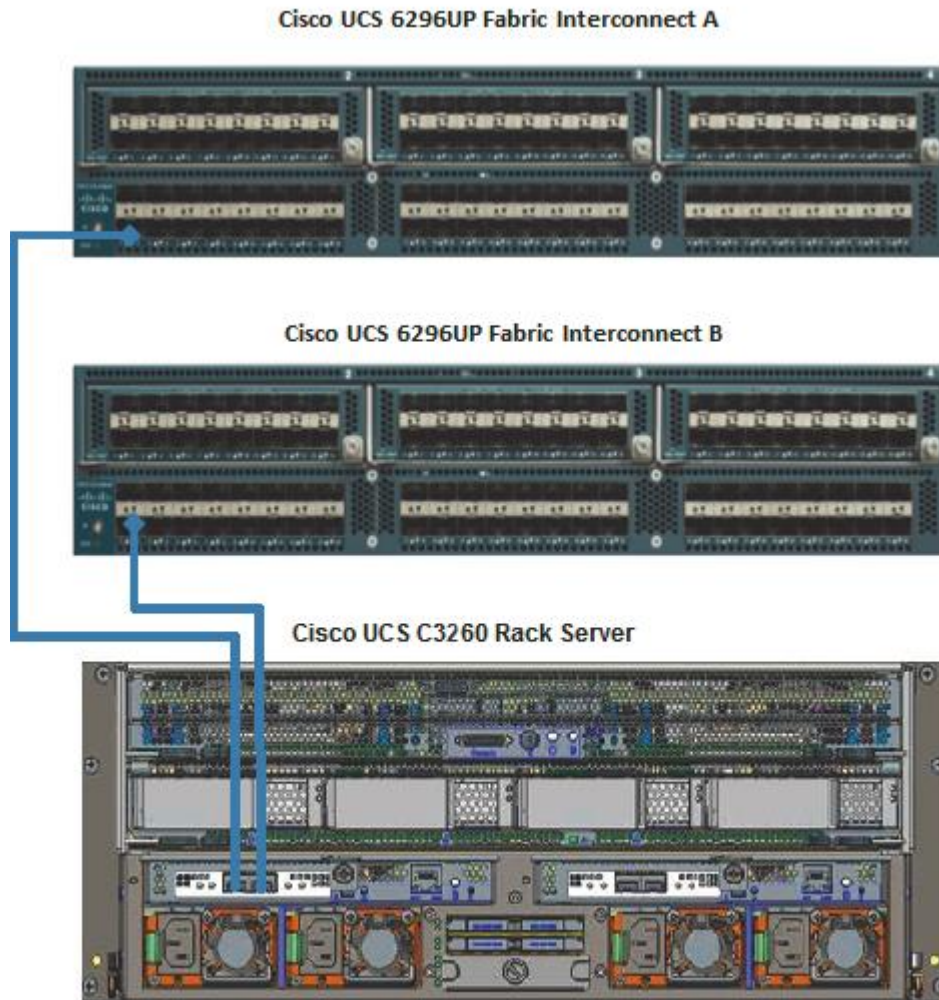
Configuration and Cabling for the Cisco C3260 Rack Server

The C3260 rack server is equipped with two Intel Xeon E5-2680 v4 processors, 256 GB of memory, one SIOC containing Cisco UCS 1300 series Virtual Interface Card, Cisco 12-Gbps SAS Modular Raid Controller with 4-GB FBWC, 60 6-TB 7.2K LFF SAS drives, and two 120-GB SATA SSD for Boot.

The servers of this category are all directly connected to the ports on the Cisco UCS FI6296 Fabric Interconnects as shown below. These ports are configured as server ports in the UCS Manager.

Figure 11 illustrates the port connectivity between the Fabric Interconnect and Cisco UCS C3260 server as a server port. One Cisco UCS C3260 server is used in master rack configurations.

Figure 11 Fabric Topology for Cisco UCS C3260 Rack Server



The SIOC card has two 40 Gigabit Ethernet ports. In order to connect these ports to the Cisco UCS Fabric Interconnect 6296, we will need make use of the QSFP-to-SFP+ converters per port (PID: CVR-QSFP-SFP10G).

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

http://www.cisco.com/en/US/docs/unified_computing/ucs/c-series_integration/ucsm2.1/b_UCSM2-1_C-Integration_chapter_010.html

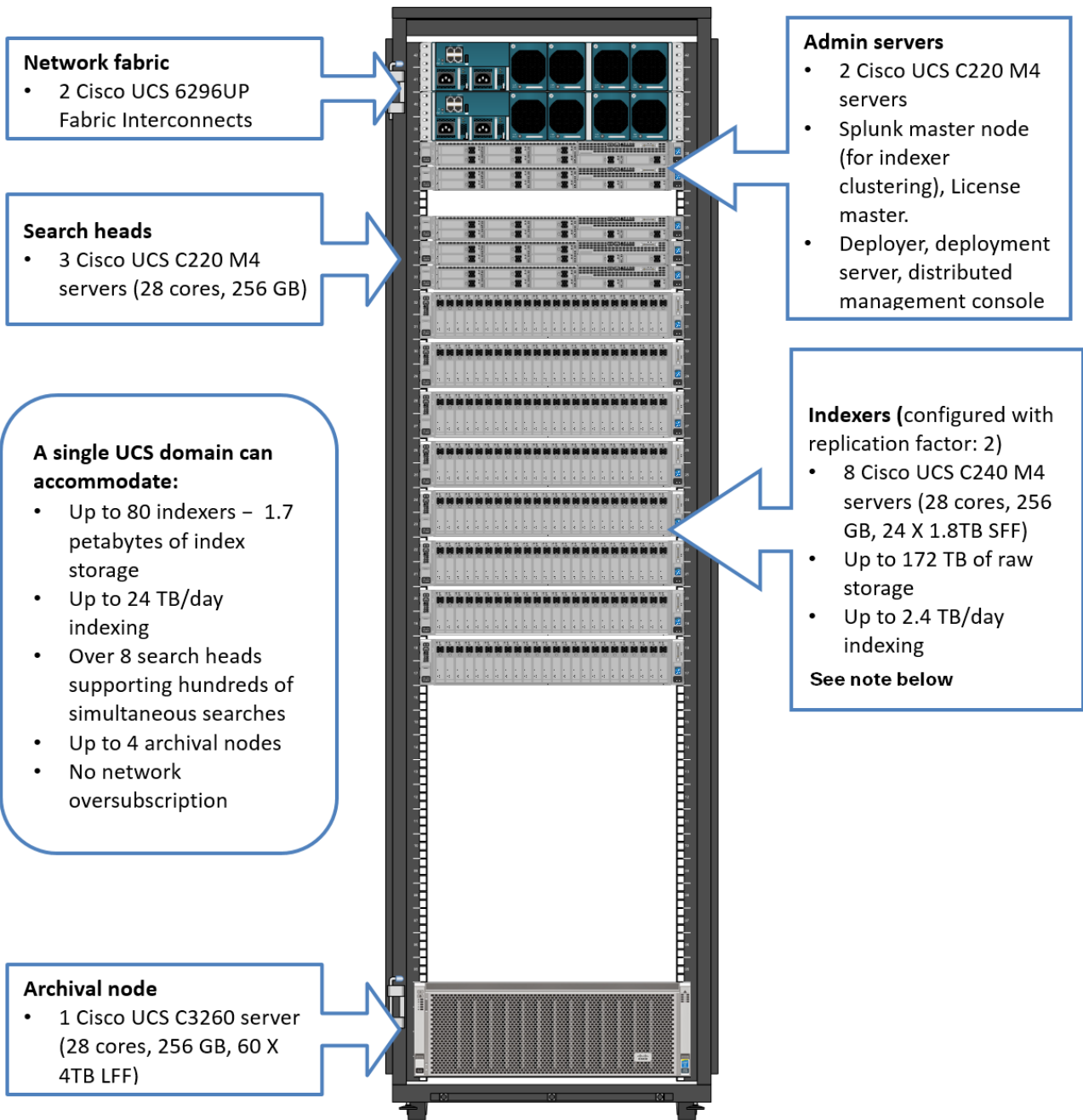
For more information on physical connectivity illustrations and cluster setup, go to:

http://www.cisco.com/en/US/docs/unified_computing/ucs/c-series_integration/ucsm2.1/b_UCSM2-1_C-Integration_chapter_010.html

Rack Appearance

Figure 12 shows the single rack configuration containing five Cisco C220 M4 servers and eight Cisco UCS C240 M4 servers, along with one Cisco UCS C3260 as an archival server. Each server is connected to each Fabric Interconnect by means of a dedicated (i.e. directly) 10 Gigabit Ethernet link. Individual server connectivity diagrams can be seen above.

Figure 12 Splunk Distributed Search with Indexer and Search Head Clustering Configuration



Note: 2.4 TB/day is computed based on the indexer’s capability of indexing 300 Gigabytes per day for core IT operational analytics use cases.

Software Distributions and Versions

The software versions tested and validated in this document are shown in Table 4

Table 4 Software Versions

Layer	Component	Version or Release
Compute	Cisco UCS C240 M4	C240M4.2.0.13d

	Cisco UCS C220 M4	C220M4.2.0.13d
	Cisco UCS C3260	C3260M4.2.0.13c
Network	Cisco UCS 6296UP	UCS 3.1(2b) A
	Cisco UCS VIC1227 Firmware	4.1 (2d)
	Cisco UCS VIC1227 Driver	2.3.0.20
Storage	LSI SAS 3108	24.12.1-0049
Software	Red Hat Enterprise Linux Server	6.8 (x86_64)
	Cisco UCS Manager	3.1 (2b)
	Splunk Enterprise	6.4.3

To learn more about Splunk Enterprise, visit: <http://www.splunk.com>.



Note: The latest drivers can be downloaded from the link below:

<https://software.cisco.com/download/release.html?mdfid=283862063&flowid=25886&softwareid=283853158&release=1.5.7d&relind=AVAILABLE&rellifecycle=&reltype=latest>.

The latest supported RAID controller driver is already included with the RHEL 6.8 operating system.

Broadwell CPUs, i.e. E5-2600 v4 processors, are supported from Cisco UCS firmware 3.1(2b) onward.

Fabric Configuration

This section provides details for configuring a fully redundant, highly available Cisco UCS 6296 Fabric Interconnect configuration.

1. Initial setup of Fabric Interconnect A and B.
2. Connect to UCS Manager with the virtual IP address using a web browser.
3. Launch UCS Manager.
4. Enable server, uplink, and appliance ports.
5. Start discovery process.
6. Create pools and policies for the Service Profile template.
7. Create Service Profile template and 13 Service Profiles.
8. Associate Service Profiles to servers.

Performing Initial Setup of Cisco UCS 6296 Fabric Interconnects

This section describes the steps to perform initial setup of the Cisco UCS 6296 Fabric Interconnects A and B.

Configure Fabric Interconnect A

1. Connect to the console port on the first Cisco UCS 6296 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. When asked to `Join centralized management environment (UCS Central)?`, select `No`.



Note: UCS Central extends the policy-based functions and concepts of Cisco UCS Manager across multiple Cisco UCS domains in one or more physical locations. If you are using multiple UCS domains, select `Yes` for this question.

20. Review the settings that were printed to the console, and if they are correct, answer `yes` to save the configuration.
21. 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 6296 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 configuring Cisco UCS 6200 Series Fabric Interconnect, go to:

http://www.cisco.com/en/US/docs/unified_computing/ucs/sw/gui/config/guide/2.0/b_UCSM_GUI_Configuration_Guide_2_0_chapter_0100.html.

Logging Into Cisco UCS Manager

To login to Cisco UCS Manager, complete the following steps:

1. Open a web browser and navigate to the Cisco UCS 6296 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 user name and enter the administrative password.
5. Click `Login` to log in to the Cisco UCS Manager.

Upgrading UCSM Software to Version 3.1(2b)

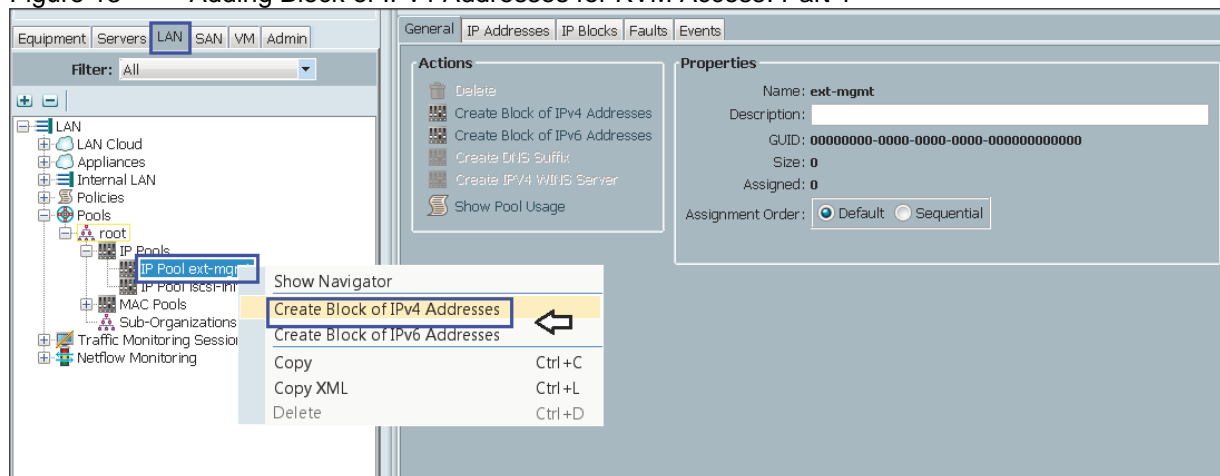
This document assumes the use of UCS 3.1(2b). Refer to [Cisco UCS 3.1 Release](#). Upgrade the Cisco UCS Manager software and UCS 6296 Fabric Interconnect software to version 3.1(2b). Also, make sure the UCS C-Series version 3.1(2b) software bundles are installed on the Fabric Interconnects.

Adding a Block of IP Addresses for KVM Access

To create a block of KVM IP addresses for server access in the Cisco UCS environment, complete the following steps:

1. Select the `LAN` tab at the top of the left window.
2. Select `Pools > IP Pools > IP Pool ext-mgmt`.
3. Right-click `IP Pool ext-mgmt`.
4. Select `Create Block of IPv4 Addresses` as shown in Figure 13

Figure 13 Adding 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)
6. Click **OK** to create the IP block.
7. Click **OK** in the message box.

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

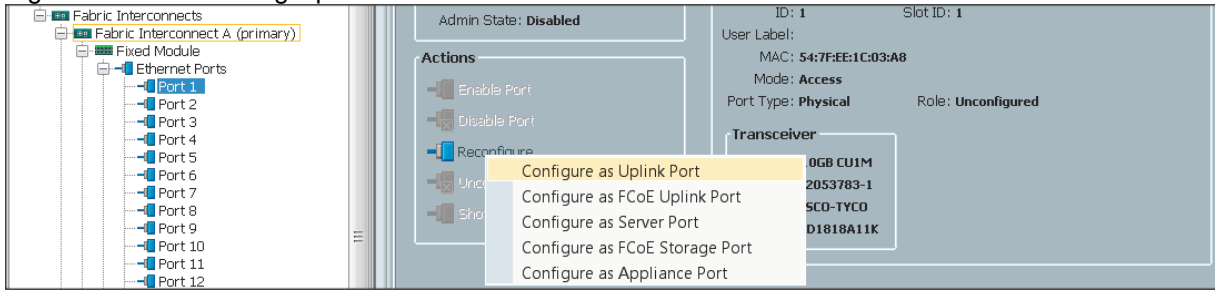
Enabling Uplink Ports

To enable uplink 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. Expand the **Ethernet Ports** section.
4. Select **Port 1** (which is connected to the uplink switch), right-click, then select **Reconfigure > Configure as Uplink Port**. (Figure 15)
5. Select **Show Interface** and select **10GB** for Uplink Connection.
6. A pop-up window appears, asking to confirm your selection. Click **Yes**, then click **OK** to continue.
7. Select **Equipment > Fabric Interconnects > Fabric Interconnect B (subordinate) > Fixed Module**.
8. Expand the **Ethernet Ports** section.
9. Select **Port 1** (which is connected to the uplink switch), right-click, then select **Reconfigure > Configure as Uplink Port**.
10. Select **Show Interface** and select **10GB** for Uplink Connection.

11. A pop-up window appears, asking to confirm your selection. Click **Yes**, then click **OK** to continue.

Figure 15 Enabling Uplink Ports



Configuring VLANs

VLANs are configured as shown in Table 6.

Table 5 VLAN Configurations

VLAN	Fabric	NIC Port	Function	Failover
default(VLAN1)	A	eth0	Management, User connectivity, Data Ingestion	Fabric Failover to B
vlan11_DATA1	B	eth1	Data Replication	Fabric Failover to A

Both VLANs must be trunked to the upstream distribution switch connecting the Fabric Interconnects. For this deployment, default VLAN1 is configured for management access (Installing and configuring OS, clustershell commands, setup NTP, user connectivity, etc) and for Splunk data ingestion from the forwarders. VLAN vlan11_DATA1 will be used for the replication traffic between the indexers. This enables Splunk to take advantage of the UCS dual 10Gigabit Ethernet (10GigE) links to isolate the inter-server traffic (i.e. index replication) from the ingress traffic (data ingestion from forwarders) on separate 10GigE links.

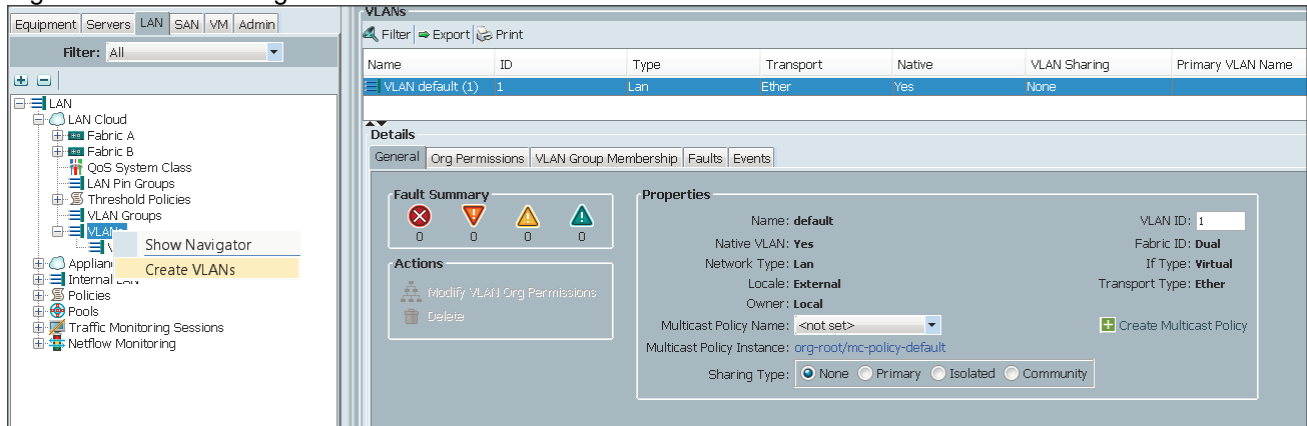


Note: We are using default VLAN1 for management traffic.

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

1. Select the **LAN** tab in the left pane in the UCS Manager GUI.
2. Select **LAN > VLANs**.
3. Right-click the **VLANs** under the root organization.
4. Select **Create VLANs** to create the VLAN. (Figure 16)

Figure 16 Creating VLAN



5. Enter vlan11_DATA1 for the VLAN Name.
6. Click the Common/Global radio button for vlan11_DATA1.
7. Enter 11 in the VLAN IDs field.
8. Click OK and then, click Finish.
9. Click OK in the success message box.

Figure 17 Creating VLANs for Splunk Data Traffic

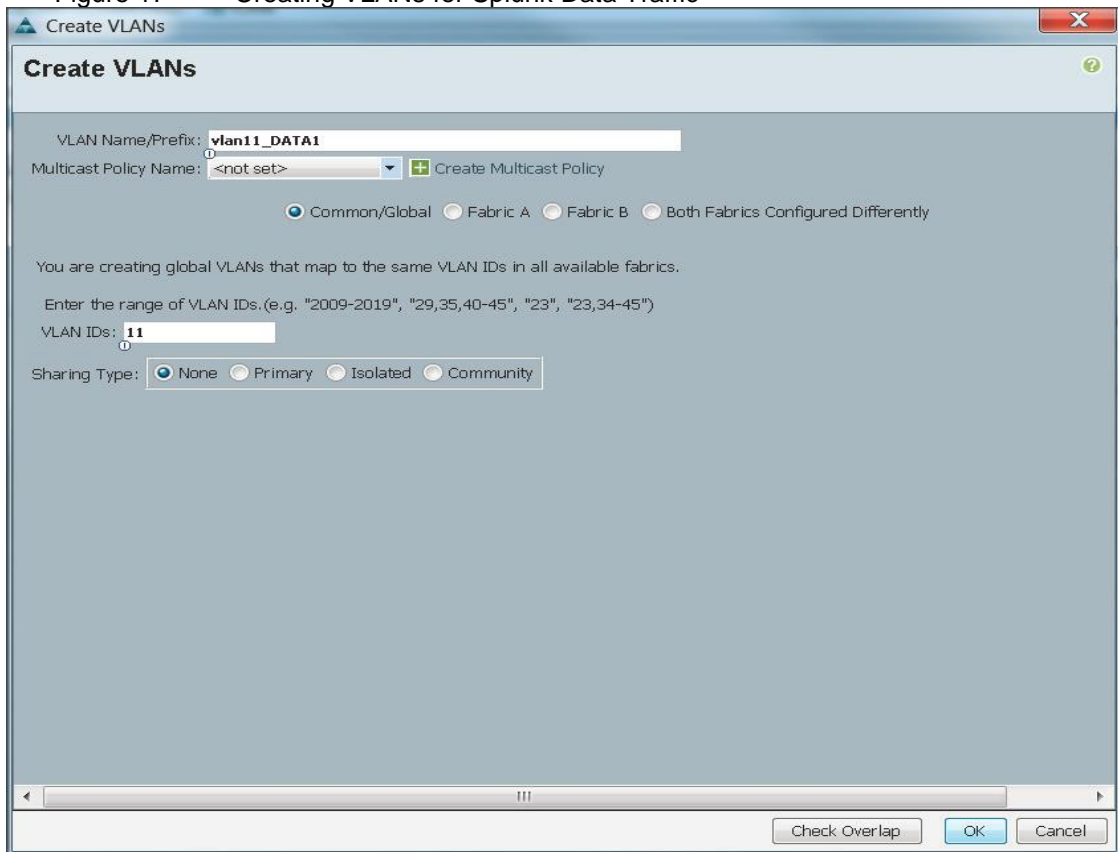


Figure 18 shows the created VLANs.

Figure 18 List of VLANs

Name	ID	Type	Transport	Native	VLAN Sharing	Primary VLAN Name
VLAN default (1)	1	Lan	Ether	Yes	None	
VLAN vlan11_DATA1 (11)	11	Lan	Ether	No	None	

Enabling Server Ports

These steps provide details for enabling server ports:

1. Select the **Equipment** tab on the top left of the window.
2. Select **Equipment > Fabric Interconnects > Fabric Interconnect A (primary) > Fixed Module**.
3. Expand the **Ethernet Ports** section. (Figure 19)
4. Select all the ports that are connected to the servers (including the C3260), right-click them, and select **Reconfigure > Configure as a Server Port**.
5. A pop-up window appears to confirm your selection. Click **Yes**, then **OK** to continue.
6. Select **Equipment > Fabric Interconnects > Fabric Interconnect B (subordinate) > Fixed Module**.
7. Expand the **Ethernet Ports** section.
8. Select all the ports that are connected to the servers (including the C3260), right-click them, and select **Reconfigure > Configure as a Server Port**.
9. A pop-up window appears, asking to confirm your selection. Click **Yes** then **OK** to continue.

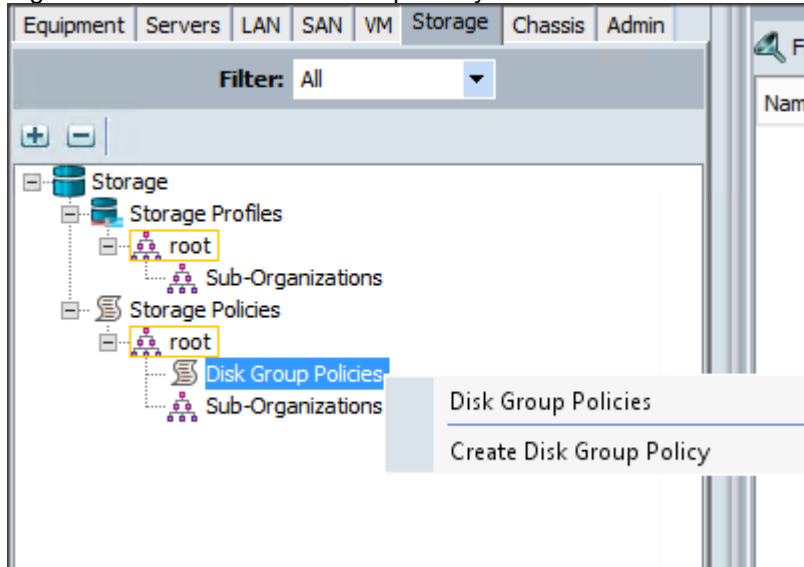
Figure 19 Enabling Server Ports

Slot	Port ID	MAC	IF Role
1	1	54:7F:EE:A0:3D:08	Network
1	2	54:7F:EE:A0:3D:09	Unconfigured
1	3	54:7F:EE:A0:3D:0A	Unconfigured
1	4	54:7F:EE:A0:3D:0B	Enable
1	5	54:7F:EE:A0:3D:0C	Disable
1	6	54:7F:EE:A0:3D:0D	Configure as Server Port
1	7	54:7F:EE:A0:3D:0E	Configure as Uplink Port
1	8	54:7F:EE:A0:3D:0F	Configure as FCoE Uplink Port
1	9	54:7F:EE:A0:3D:10	Configure as FCoE Storage Port
1	10	54:7F:EE:A0:3D:11	Configure as Appliance Port
1	11	54:7F:EE:A0:3D:12	Unconfigure
1	12	54:7F:EE:A0:3D:13	Unconfigure FCoE Uplink Port
1	13	54:7F:EE:A0:3D:14	Unconfigure Uplink Port
1	14	54:7F:EE:A0:3D:15	Unconfigure FCoE Storage Port
1	15	54:7F:EE:A0:3D:16	Unconfigure Appliance Port
1	16	54:7F:EE:A0:3D:17	Unconfigure both
1	17	54:7F:EE:A0:3D:18	

Creating a Storage Profile for Boot Drives

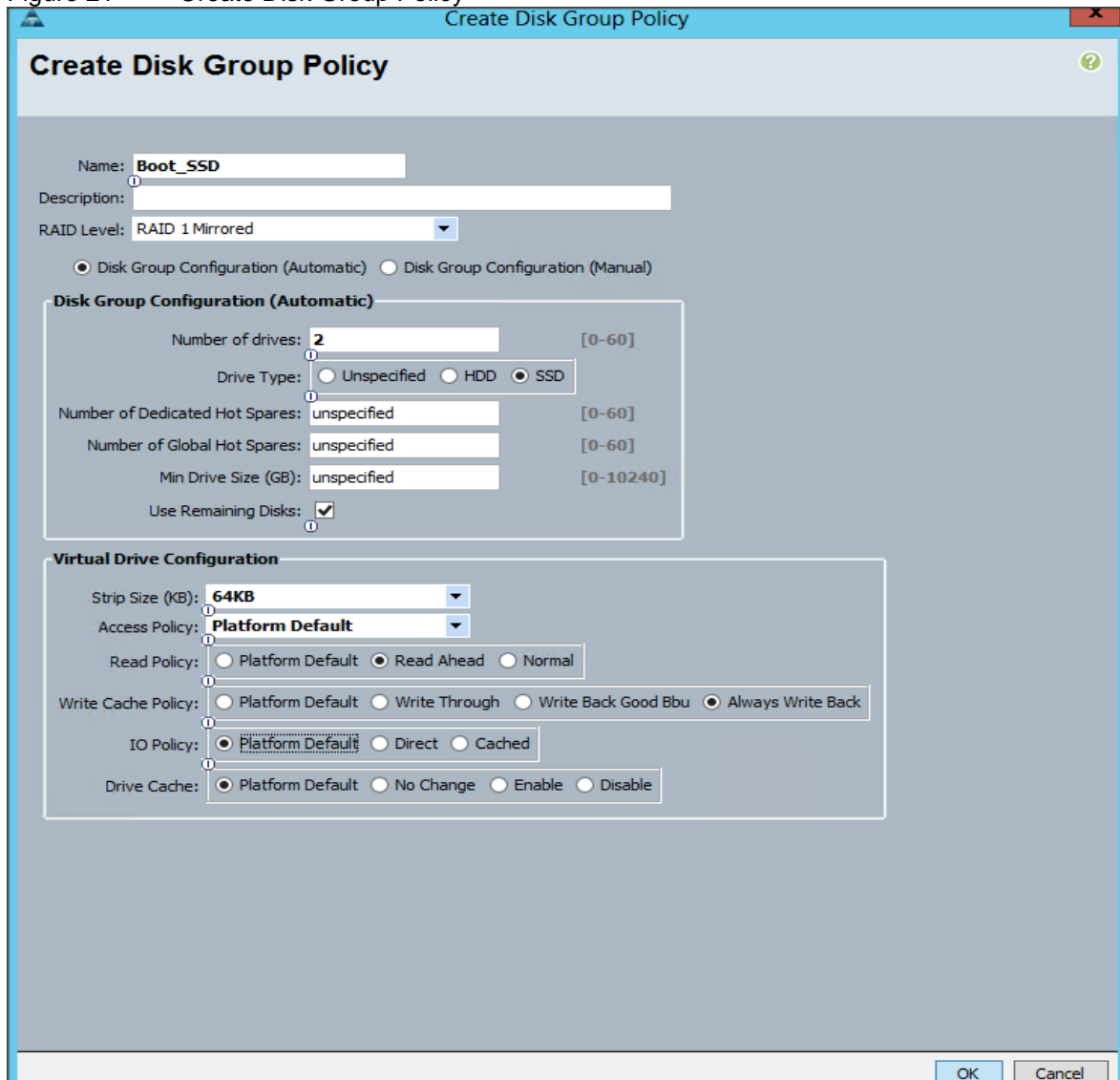
1. Go to the Storage tab and expand Storage → Storage Policies.
2. Right click on Disk Group Policies and click Create Disk Group Policies as shown in Figure 20

Figure 20 Create Disk Group Policy



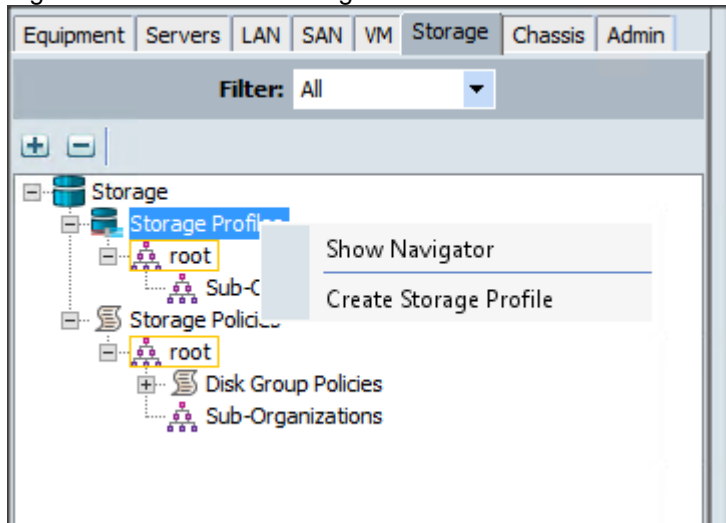
3. In the Create Disk Policy window, configure the following parameters and click OK as shown in Figure 21
 - a. Name = Boot_SSD
 - b. RAID Level = RAID 1 Mirrored
 - c. Disk Group Configuration = Automatic
 - d. Number of Drives = 2
 - e. Drive Type = SSD
 - f. Use Remaining Disks = checked
 - g. Strip 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 21 Create Disk Group Policy



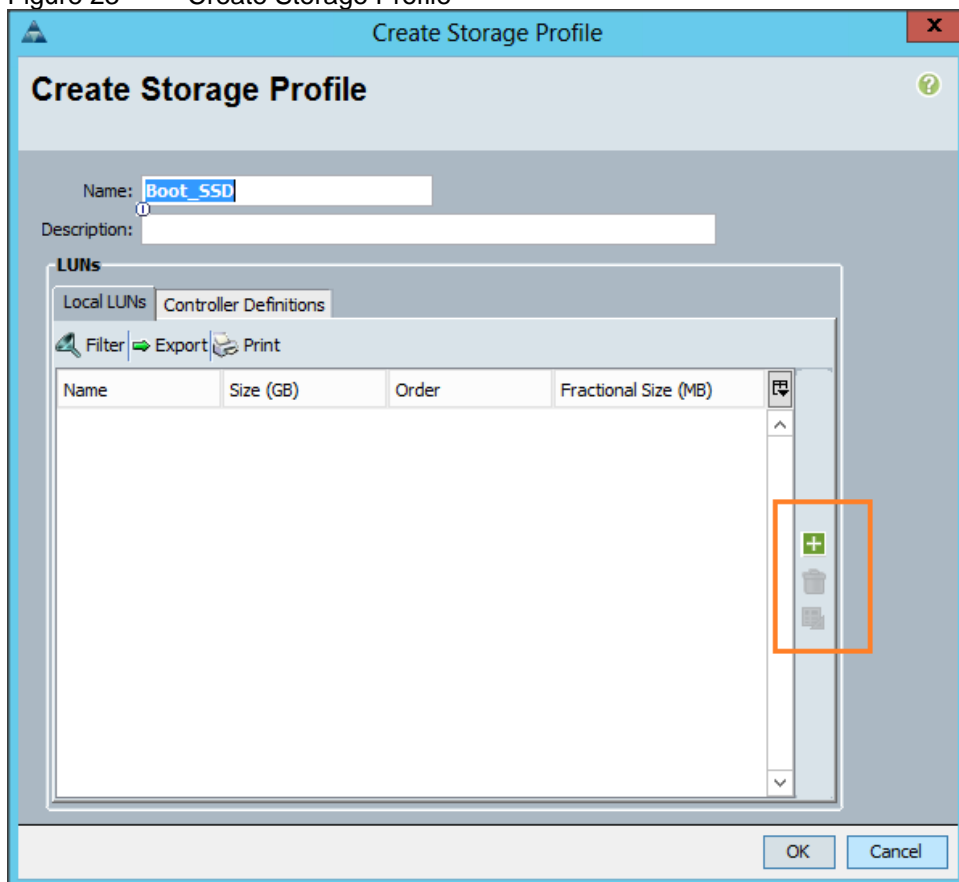
4. In the Storage tab, right click on Storage Profile, and click Create Storage Profile as shown in Figure 22

Figure 22 Create Storage Profile



5. Enter "Boot_SSD" in the name field. Under Local LUNs click "+" to add local LUN, as shown in Figure 23

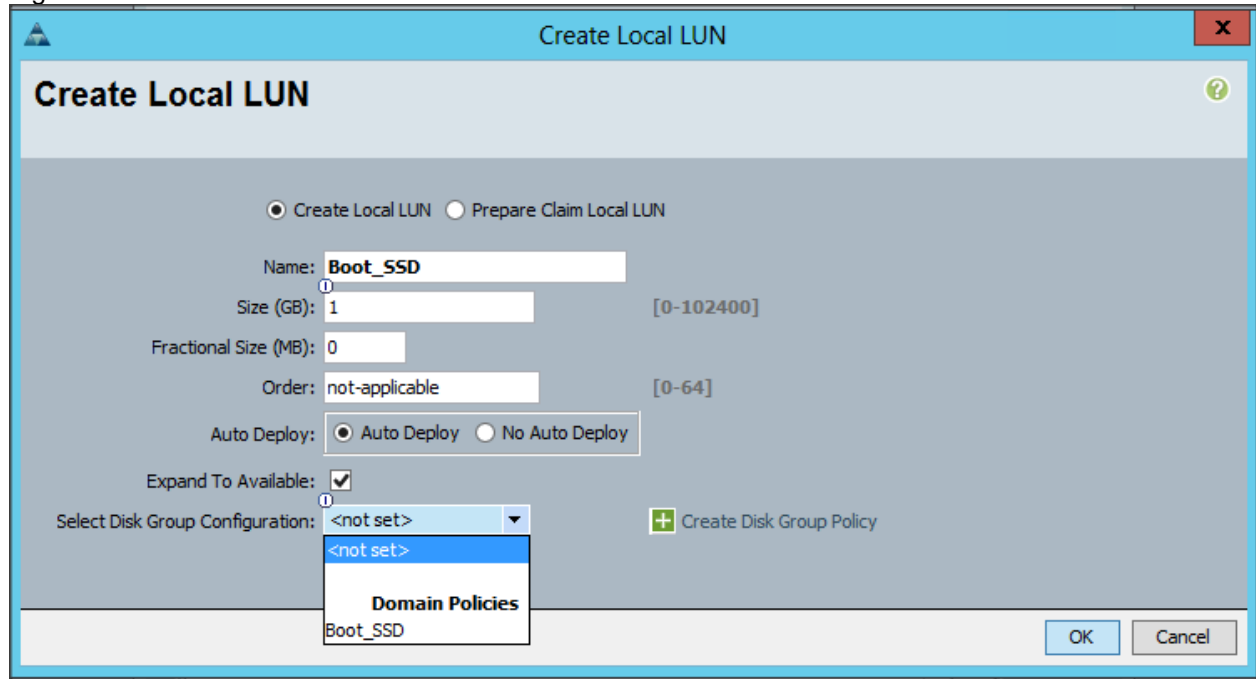
Figure 23 Create Storage Profile



6. In the Create Local LUN window, enter the name Boot_SSD as shown in Figure 24
7. Check the Expand to Available checkbox to use all available space.

8. Under the `Select Disk Group Configuration` drop down list, choose `Boot_SSD`, which was created earlier.
9. Click `OK` and `OK` again to complete the configuration.

Figure 24 Create Local LUN



Creating Pools for Service Profile Templates

Creating an Organization

Organizations are used as a means to arrange and restrict access to various groups within the IT organization, thereby enabling multi-tenancy of the compute resources. This document does not assume the use of organizations; however, the necessary steps are provided for future reference. If you create an organization, you can choose it instead of root in the remaining instructions of this document (e.g., step 2 of the next section, Creating MAC Address Pools).

To configure an organization within the Cisco UCS Manager GUI, complete the following steps:

1. Click `New` on the top left corner in the right pane in the UCS Manager GUI.
2. Select `Create Organization` from the options.
3. Enter a name for the organization.
4. (Optional) Enter a description for the organization.
5. Click `OK`.
6. Click `OK` in the success message box.

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 as shown in Figure 25
5. Enter the MAC Pool name, which is ucs.
6. (Optional) Enter a description of the MAC pool.
7. Select the Assignment Order to be Sequential.
8. Click Next.
9. Click Add.
10. Specify a starting MAC address.
11. Specify a size of the MAC address pool, which is sufficient to support the available server resources, as shown in Figure 26
12. Click OK.

Figure 25 Creating MAC Pool Window

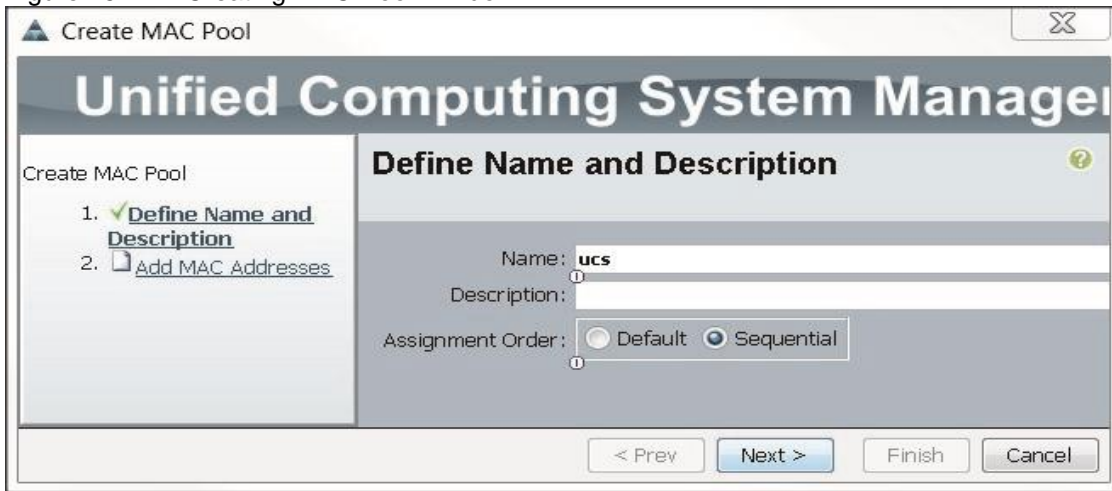
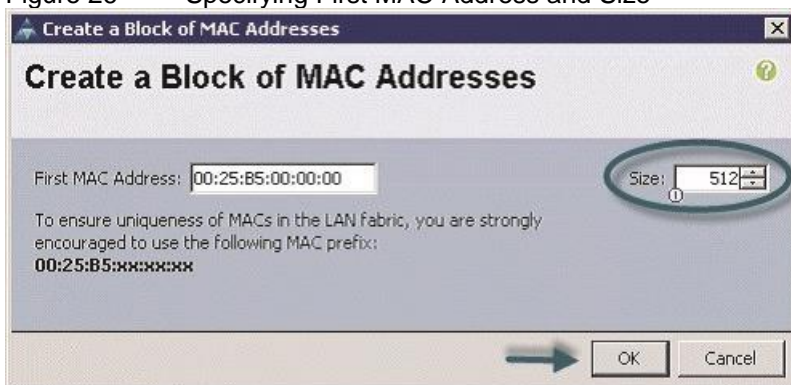
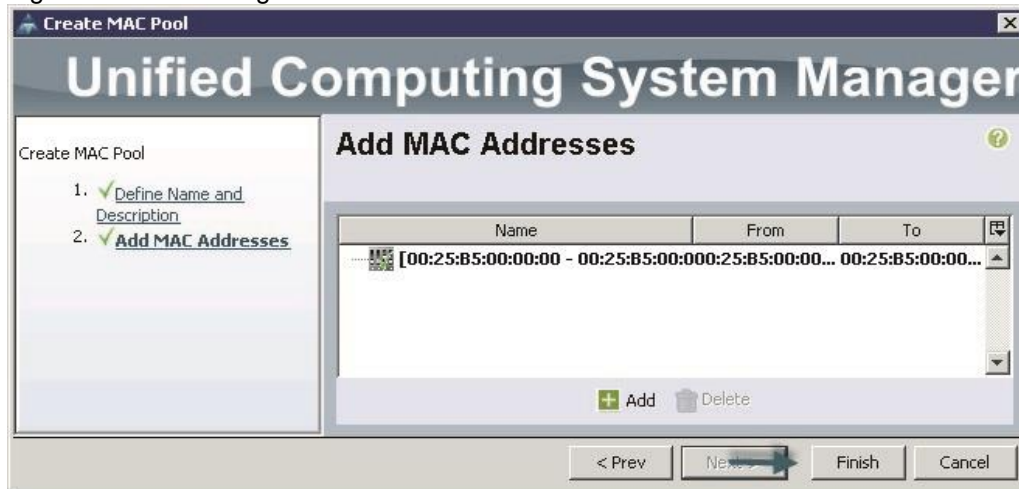


Figure 26 Specifying First MAC Address and Size



- Click `Finish` as shown in Figure 27

Figure 27 Adding MAC Addresses



- When the message box displays, click `OK`, as shown in Figure 28

Figure 28 Confirming Newly Added MAC Pool



Creating Server Pools

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. Follow these steps to configure the server pool within the Cisco UCS Manager GUI:

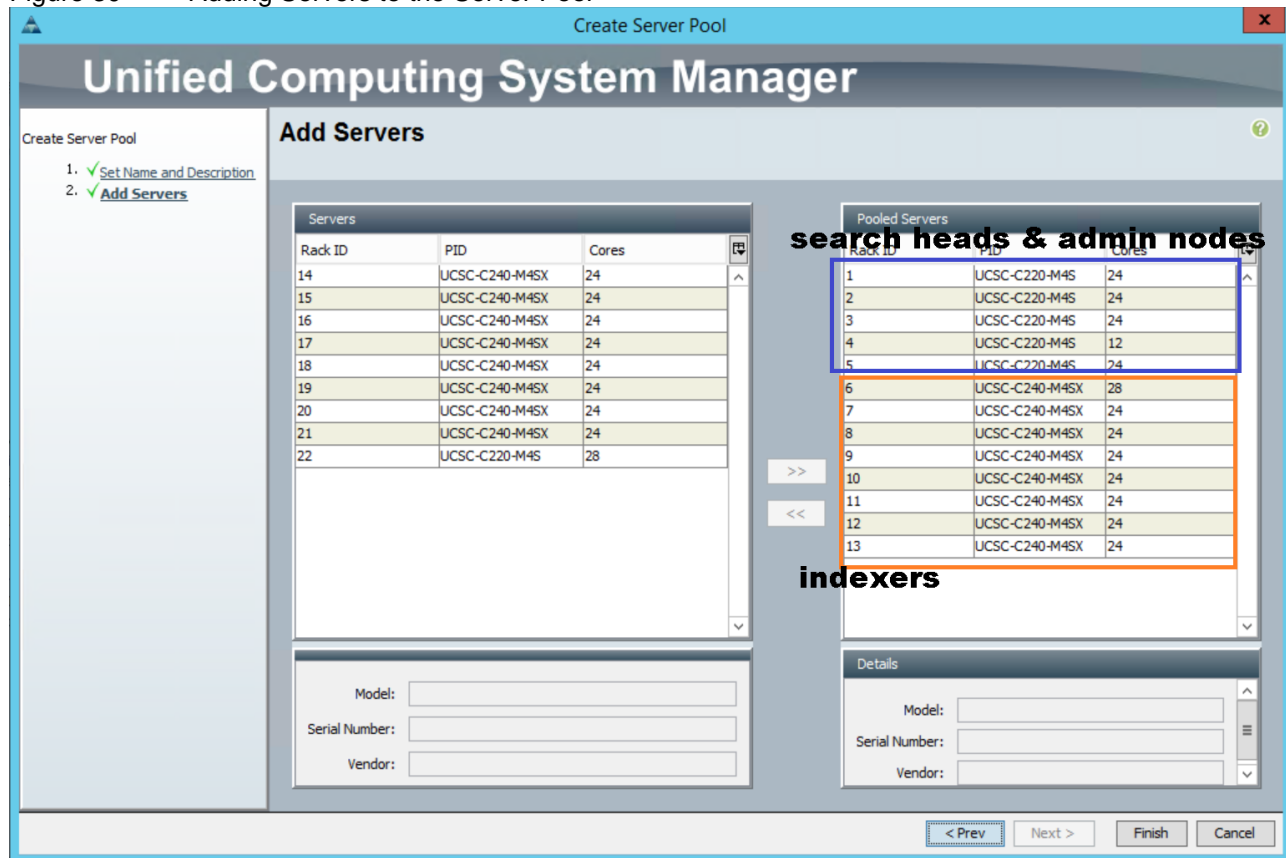
- Select the `Servers` tab in the left pane in the UCS Manager GUI.
- Select `Pools > root`.
- Right-click `Server Pools`.
- Select `Create Server Pool`.
- Enter the server pool name, which is `ucs`, as shown in Figure 29
- (Optional) Enter a description for the server pool.
- Click `Next` to add the servers.

Figure 29 Setting Name and Description of Server Pool

The screenshot shows the 'Unified Computing System Manager' interface for the 'Create Server Pool' wizard. The title bar reads 'Create Server Pool' and the main window title is 'Unified Computing System Manager'. The wizard is currently on the 'Set Name and Description' step, which is highlighted in a light purple header. On the left side, a sidebar shows the progress: '1. ✓ Set Name and Description' and '2. Add Servers'. The main area contains two text input fields: 'Name:' with the value 'ucs' and 'Description:'. At the bottom right, there are four buttons: '< Prev', 'Next >', 'Finish', and 'Cancel'.

8. Select all the Cisco UCS C240 M4 and all five Cisco UCS C220 M4 servers to be added to the server pool and then click >> to add them to the pool, as shown in Figure 30
9. Click **Finish**.
10. Click **OK**, and then click **Finish**.

Figure 30 Adding Servers to the Server Pool



Creating Policies for Service Profile Templates

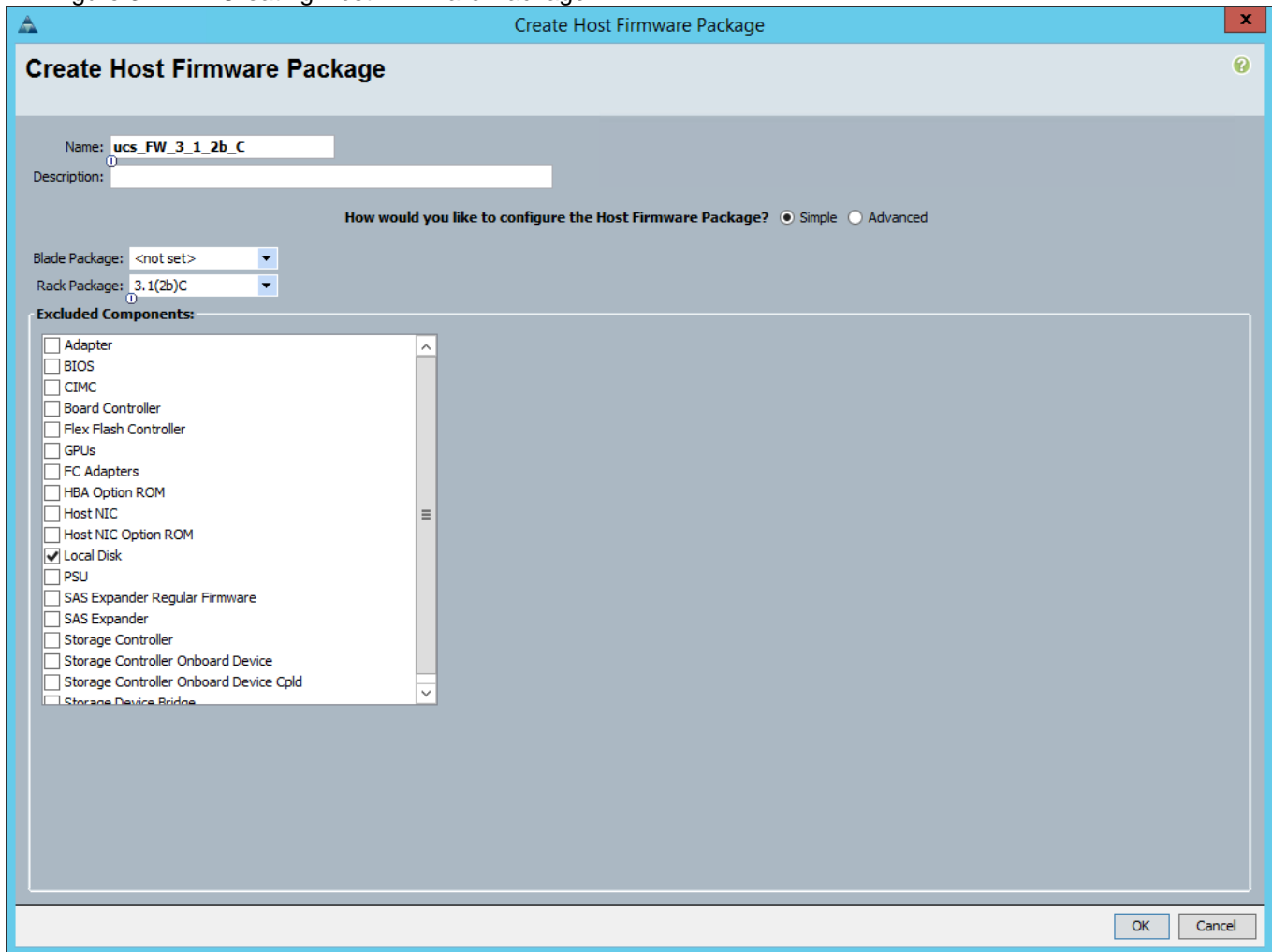
Creating Host Firmware Package Policy

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, ROM, and storage controller properties as applicable. Follow these steps to create a firmware management policy for a given server configuration using the Cisco UCS Manager GUI:

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 host firmware package name, which is `ucs_FW_3_1_2b_C`, as shown in Figure 31
6. Name the Host Firmware Package appropriately to include the actual firmware version and package.
7. Click the `Simple` radio button to configure the host firmware package.
8. Select the appropriate `Rack Package` that you have.
9. Click `OK` to complete creating the management firmware package.

10. Click **OK**.

Figure 31 Creating Host Firmware Package



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, as shown in Figure 32
6. Select `Platinum` from the drop down menu.

7. Keep the `Burst (Bytes)` field as default (10240).
8. Keep the `Rate (Kbps)` field as default (line-rate).
9. Keep `Host Control` radio button as default (none).
10. Once the pop-up window appears, click `OK` to complete the creation of the policy.

Figure 32 Creating Platinum QoS Policy

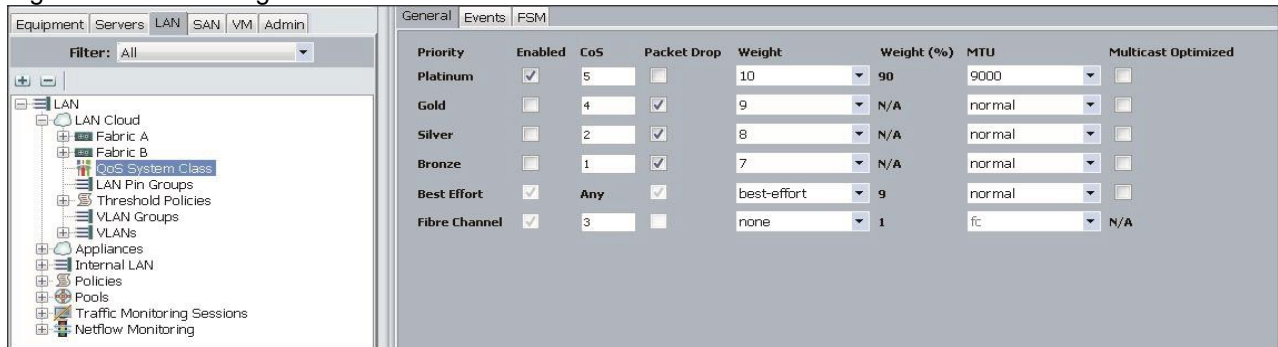


Setting Jumbo Frames

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

1. Select the `LAN` tab in the left pane in the UCS Manager GUI.
2. Select `LAN Cloud > QoS System Class`, as shown in Figure 33
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 `best-effort` for `weight`.
7. In the `Fiber Channel` row, select `none` for `weight`.
8. Click `Save Changes`.
9. Click `OK`.

Figure 33 Setting Jumbo Frames



Creating 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 the local disk configuration policy name, which is `ucs`, as shown in Figure 34
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`.

Figure 34 Configuring Local Disk Policy

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 done manually and is often error-prone. By creating a BIOS policy and assigning the policy to a server or group of servers, you 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 UCS Manager GUI.
2. Select `Policies > root`.
3. Right-click `BIOS Policies`.
4. Select `Create BIOS Policy`.
5. Enter the BIOS policy name, which is `ucs`. Figure 35

6. Change the BIOS settings to match the following figures:

Figure 35 Creating Server BIOS Policy – Main Screen

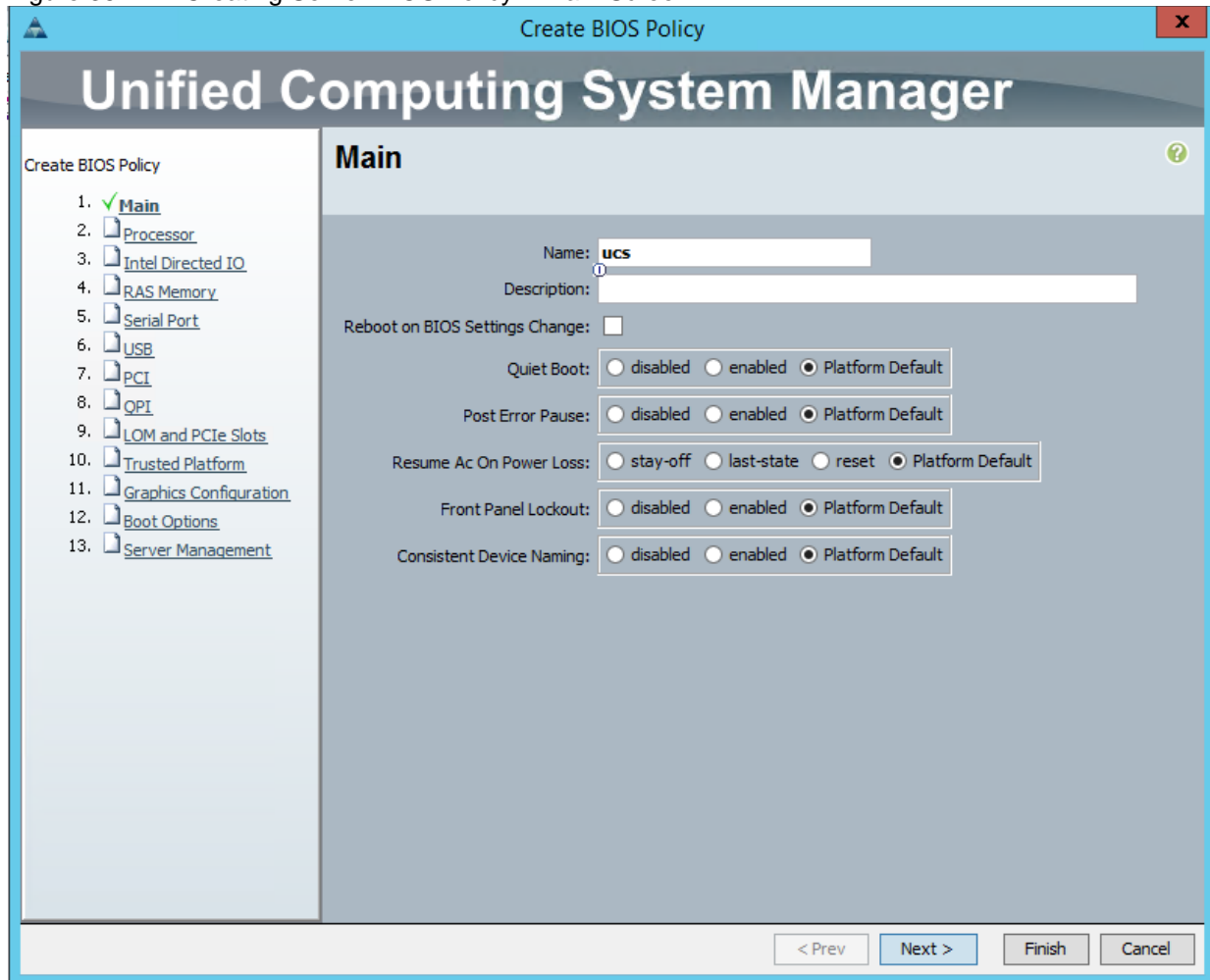


Figure 36 Creating Server BIOS Policy for Processor

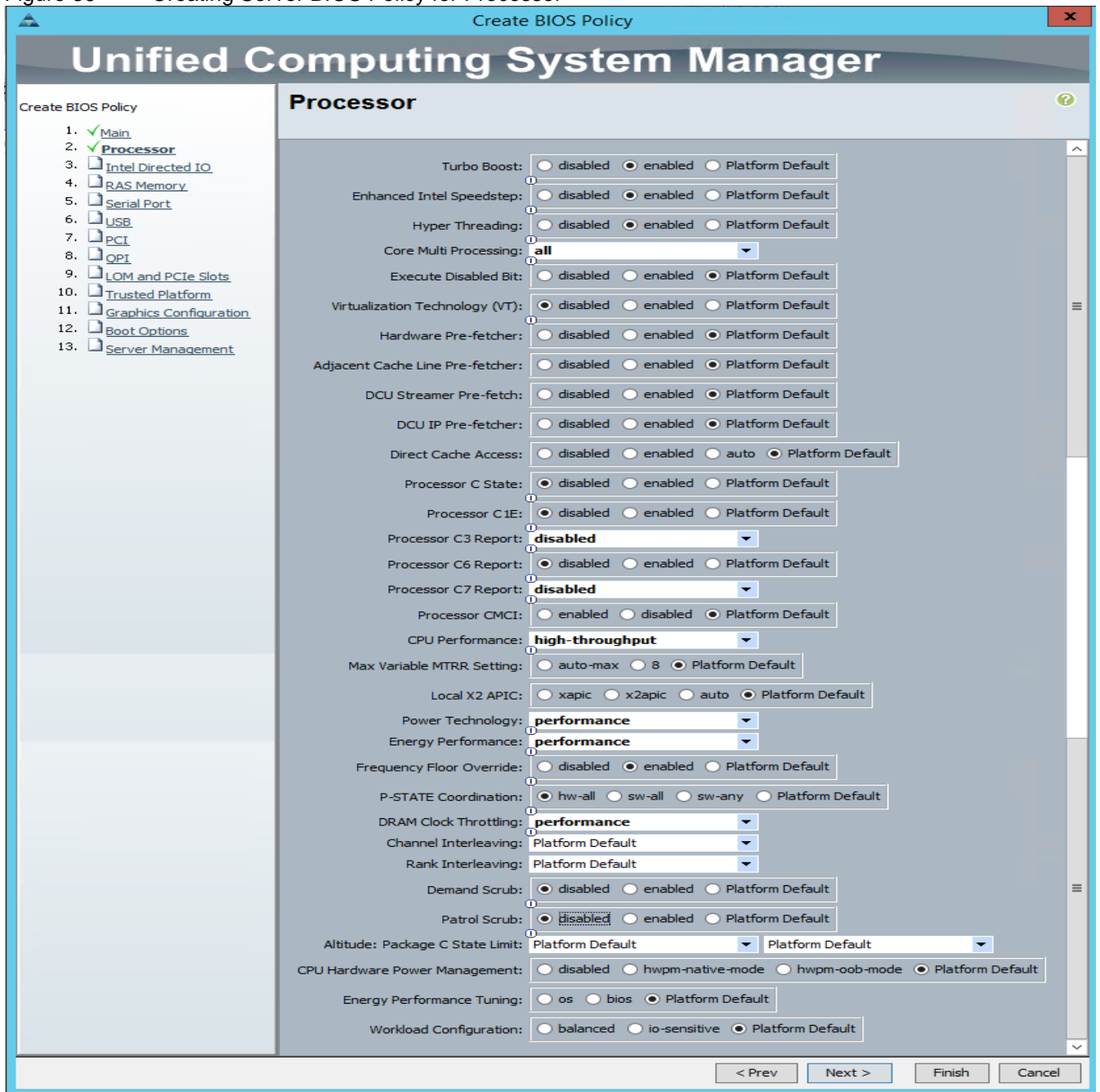


Figure 37 Creating Server BIOS Policy for Intel Directed IO

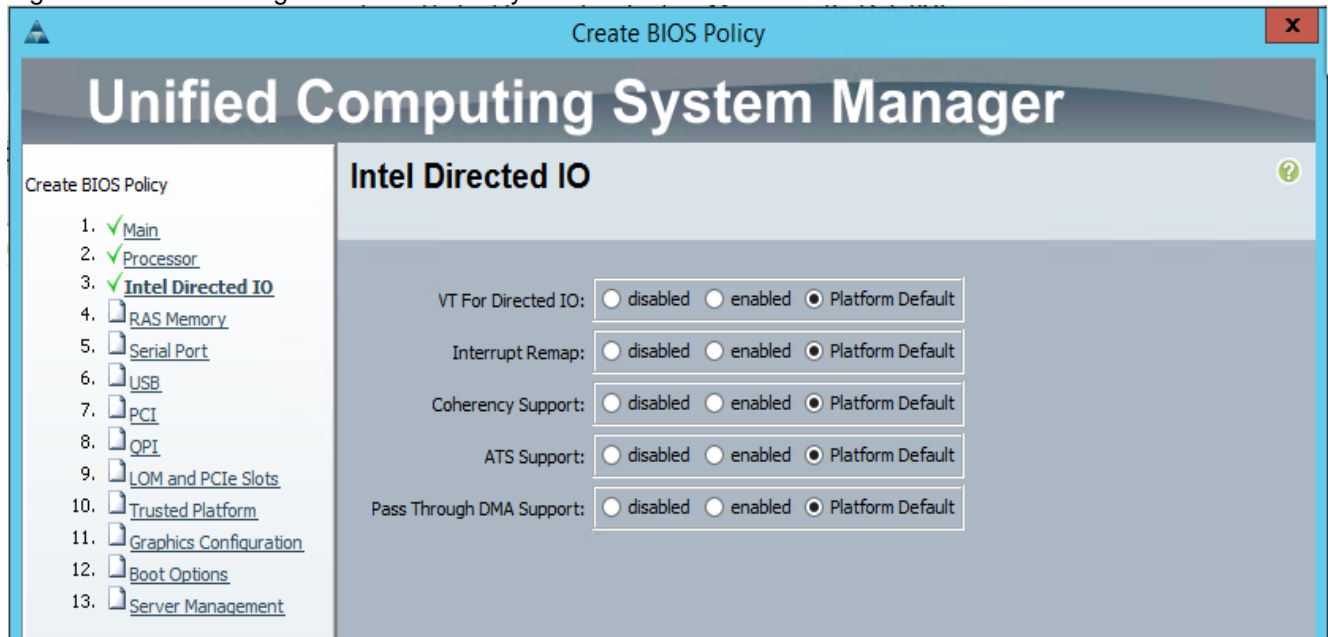
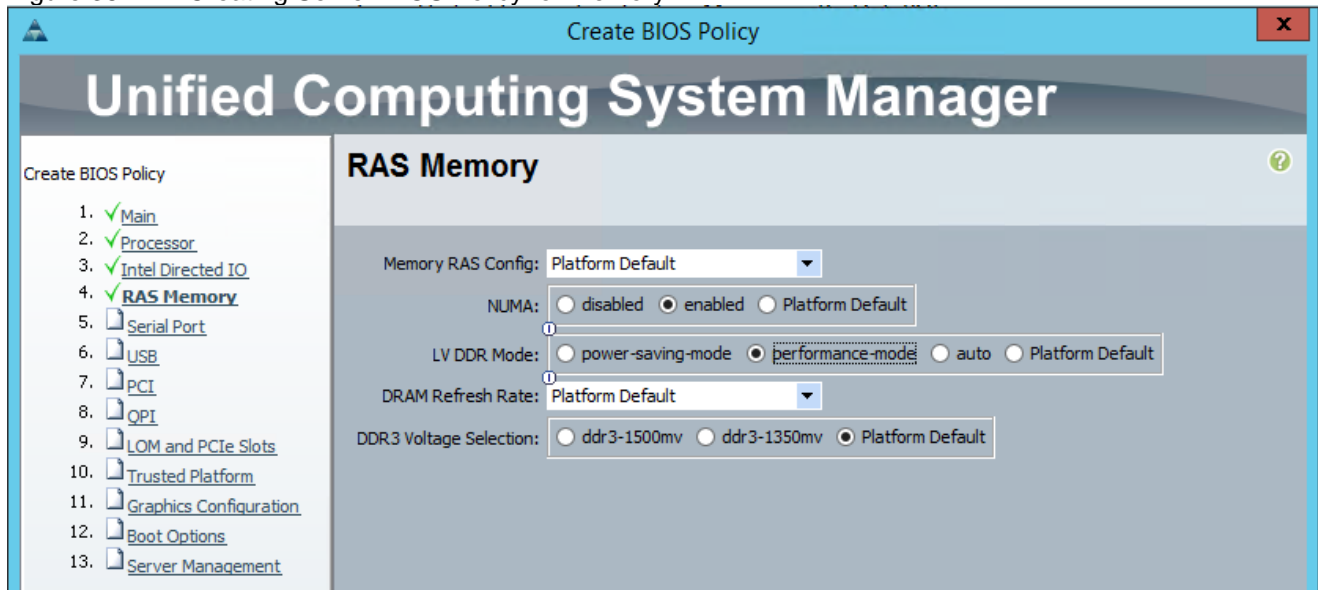


Figure 38 Creating Server BIOS Policy for Memory



7. Click **Finish** to complete creating the BIOS policy.
8. Click **OK**.

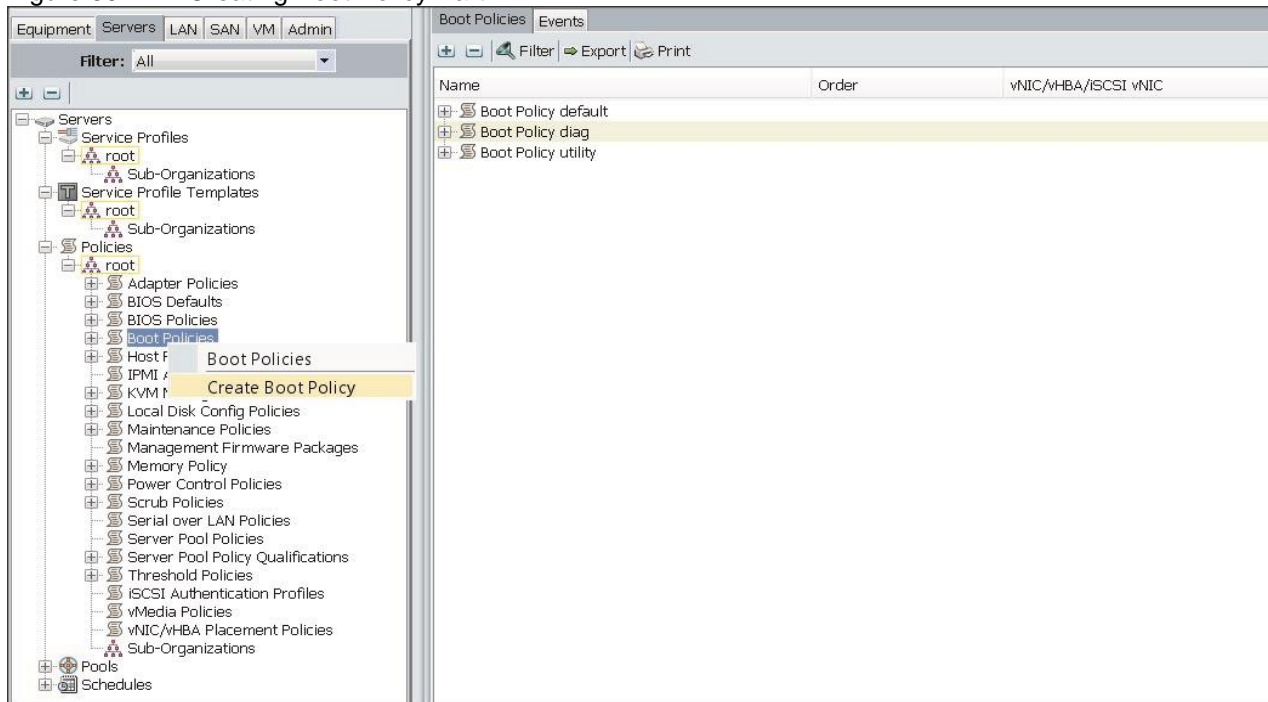
Creating Boot Policy

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. (Figure 39)
2. Select **Policies > root**.

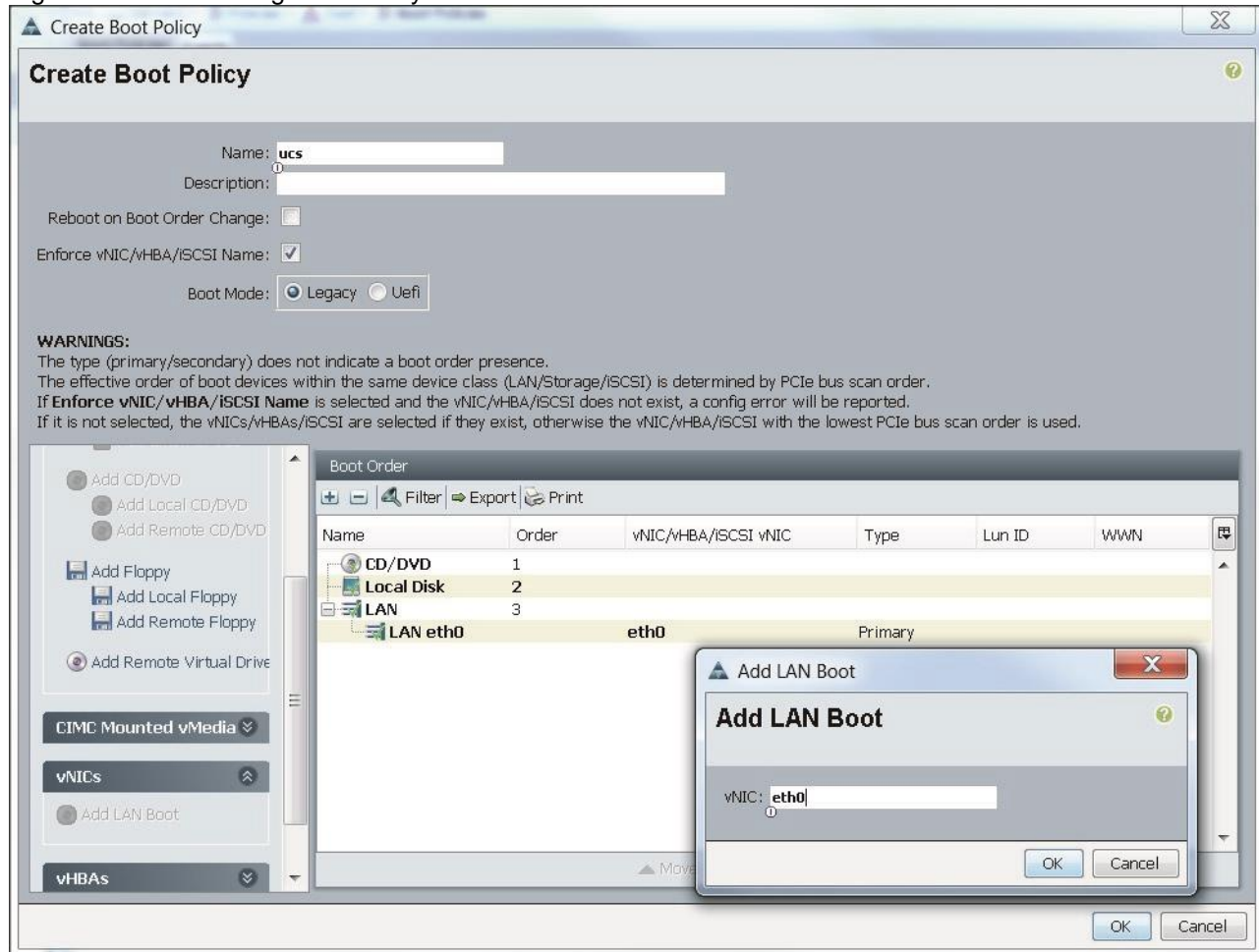
3. Right-click the `Boot Policies`.
4. Select `Create Boot Policy`.

Figure 39 Creating Boot Policy Part 1



5. Enter the boot policy name, which is `ucs`. (Figure 40)
6. (Optional) Enter a description for the boot policy.
7. Keep the `Reboot on Boot Order Change` check box unchecked.
8. Keep the `Enforce vNIC/vHBA/iSCSI Name` check box checked.
9. Keep `Boot Mode` as the default (`Legacy`).
10. Expand `Local Devices > Add CD/DVD` and select `Add CD/DVD`.
11. Expand `Local Devices` and select `Add Local Disk`.
12. Expand `vNICs` and select `Add LAN Boot` and enter `eth0`.
13. Click `OK` to add the boot policy.
14. Click `OK`.

Figure 40 Creating Boot Policy Part 2

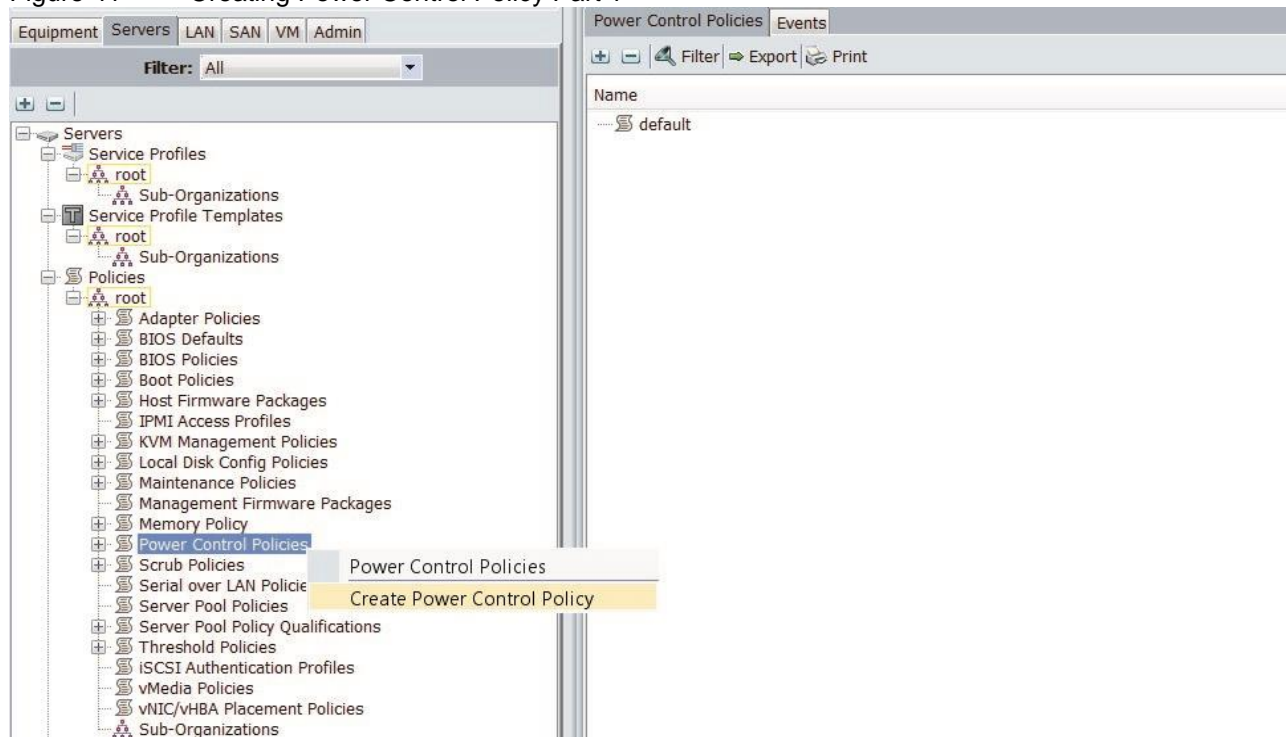


Creating Power Control Policy

To create the 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. (Figure 41)
2. Select **Policies > root**.
3. Right-click **Power Control Policies**.
4. Select **Create Power Control Policy**.

Figure 41 Creating Power Control Policy Part 1



5. Enter the power control policy name: `ucs`. (Figure 42)
6. (Optional) Enter a description for the power control policy.
7. Select `No cap` for the Power Capping selection.
8. Click `OK` to create the power control policy.
9. Click `OK`.

Figure 42 Creating Power Control Policy Part 2

Create Power Control Policy

Name:

Description:

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.

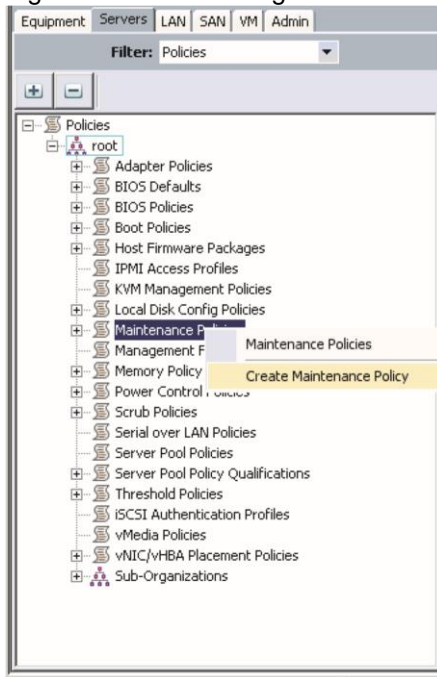
OK Cancel

Creating a Maintenance Policy

To create a maintenance policy, complete the following steps:

1. Select the `Servers` tab in the left pane in the UCS Manager GUI. (Figure 43)
2. Select `Policies > root`.
3. Right-click `Maintenance Policies`.
4. Select `Create Maintenance Policy`.

Figure 43 Creating Maintenance Policy - Part 1



5. Enter the maintenance policy name: `user_ack`.
6. Select `User Ack` as the `Reboot Policy` by clicking on it.
7. Click `OK` to create the maintenance policy.
8. Click `OK`.

Figure 44 Creating Maintenance Policy - Part 2

 A screenshot of a dialog box titled 'Create Maintenance Policy'. The dialog has a blue header bar with a close button (X) on the right. The main area is light gray and contains the following fields:

- Name:** A text input field containing the text 'user_ack'.
- Description:** An empty text input field.
- Soft Shutdown Timer:** A dropdown menu set to '150 Secs'.
- Reboot Policy:** A group box containing three radio buttons: 'Immediate', 'User Ack' (which is selected), and 'Timer Automatic'.
- On Next Boot:** A checkbox that is currently unchecked.
- Below the checkbox, the text '(Apply pending changes at next reboot.)' is displayed.

 At the bottom right of the dialog, there are two buttons: 'OK' and 'Cancel'.

Creating Chassis Profiles for C3260 Rack Servers

A chassis profile is required for discovering the C3260 server nodes. To create a chassis profile the following steps need to be performed.

- Create Disk zoning policy to discover and allocate the hard disk drives per server node.
- Assign chassis firmware policy
- Create a chassis profile template.
- Create and associate a chassis profile with the C3260 chassis.

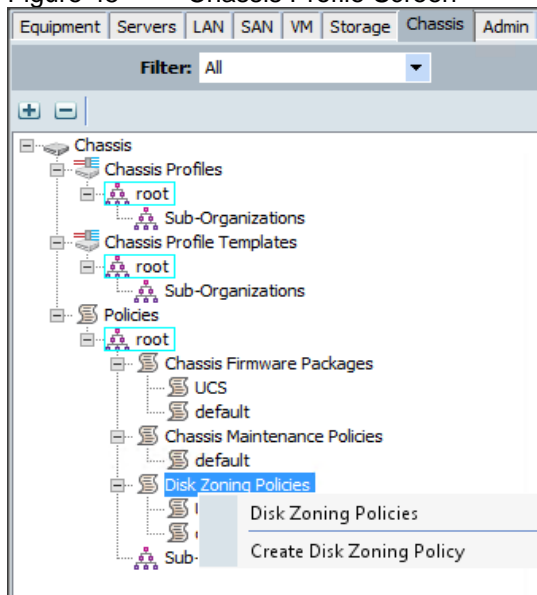


Note: In this solution, all the 60 drives of the chassis are allocated to one server node.

Creating Disk Zoning Policy

1. Click the **Chassis** tab on the top of the left navigation pane in UCS Manager.
1. Expand **Policies -> Root -> Disk Zoning Policies**. (Figure 45)
2. Right-click on **Disk Zoning Policies** and click **Create Disk Zoning Policy**.

Figure 45 Chassis Profile Screen



3. In the **Create Disk Zoning Policy** window, enter the Name **UCS** and click “+” to add the disk zoning information. (Figure 46)

Figure 46 Disk Zoning Policy Screen

Create Disk Zoning Policy

Name:

Description:

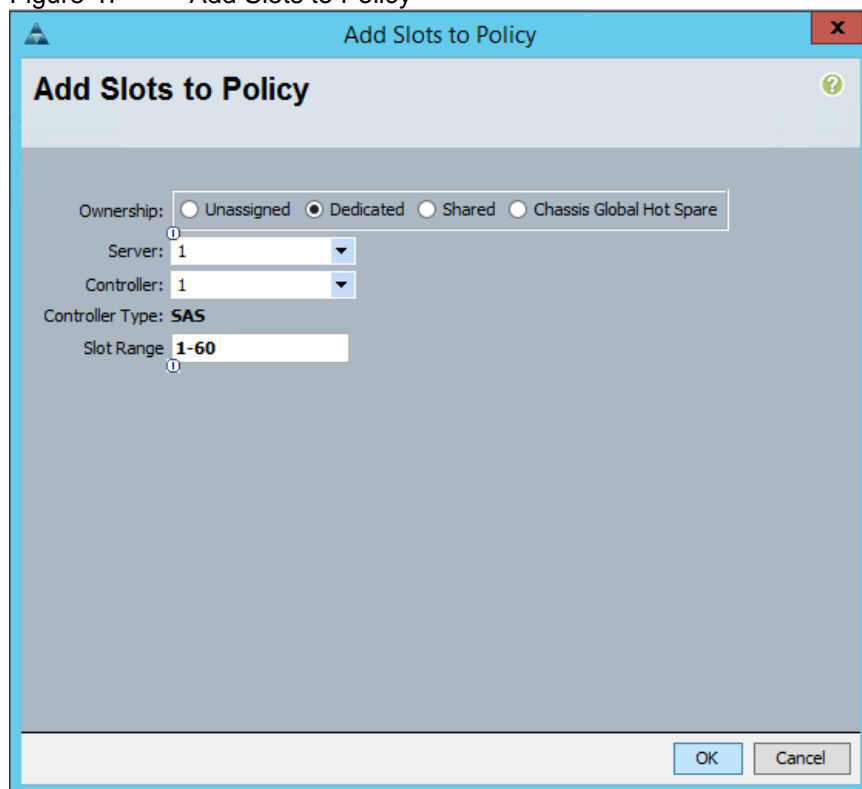
Preserve Config:

Disk Zoning Information

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

4. In the Add Slots to Policy window, select the Dedicated radio button.
5. From the Server drop down list choose "1".
6. From the Controller drop down list, choose "1".
7. For the Slot Range field, enter 1-60 and click OK.

Figure 47 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: 1-60

OK Cancel

8. Click **OK** to finish creating the disk zoning policy.

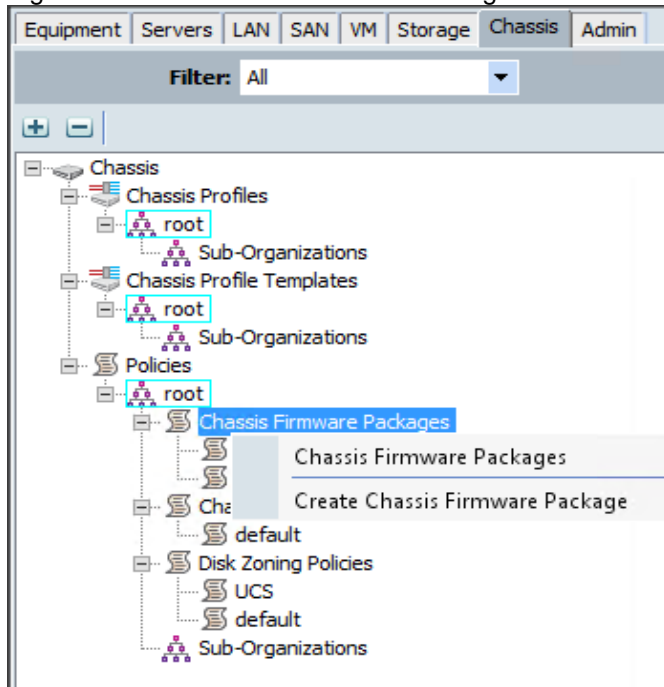


Note: In this configuration, a single server is used, which holds 60 drives. In a two-server configuration, there would only be room for 56 drives, with each server assigned to 28 drives.

Creating Chassis Firmware Package Policy

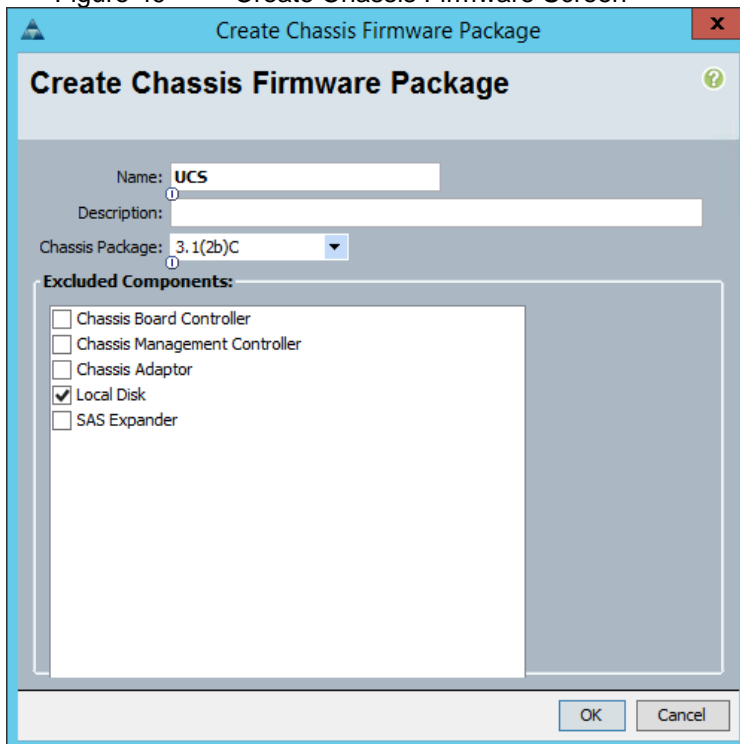
1. In the **Chassis** tab, expand **Chassis > Policies > Root**.
2. Right click on **Chassis Firmware Packages** and click **Create Chassis Firmware Packages**. (Figure 48)

Figure 48 Chassis Firmware Packages



3. In the Create Chassis Firmware Package window, enter UCS as the Name.
4. From the Chassis Packages drop down list, choose the appropriate package (must be 3.1(2b) or above) and click OK.

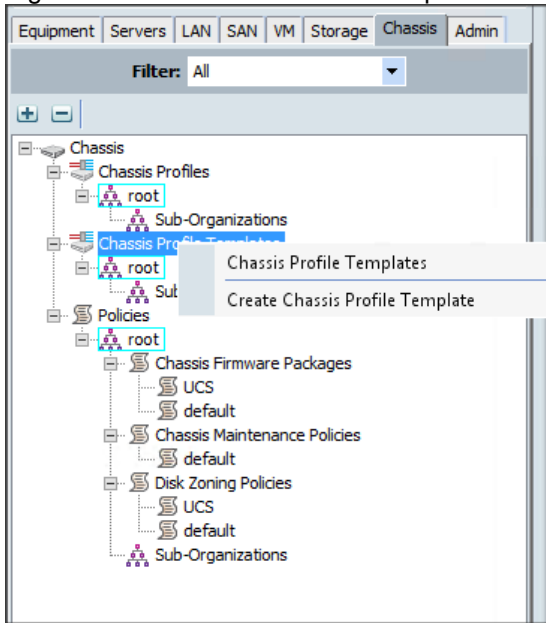
Figure 49 Create Chassis Firmware Screen



Creating a Chassis Profile Template

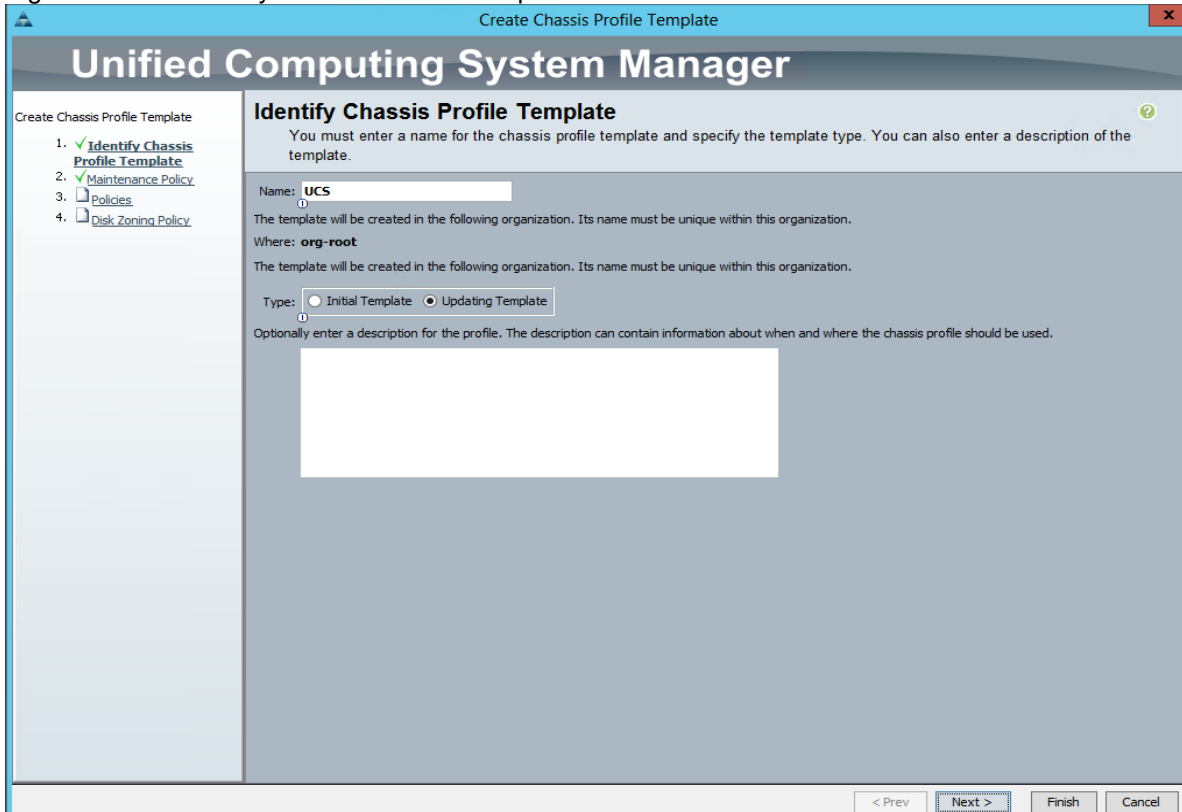
1. Under Chassis Profile Template, right-click and click Create Chassis Profile Template. (Figure 50)

Figure 50 Chassis Profile Template



2. Enter UCS for the Name. Select Updating Template for Type.

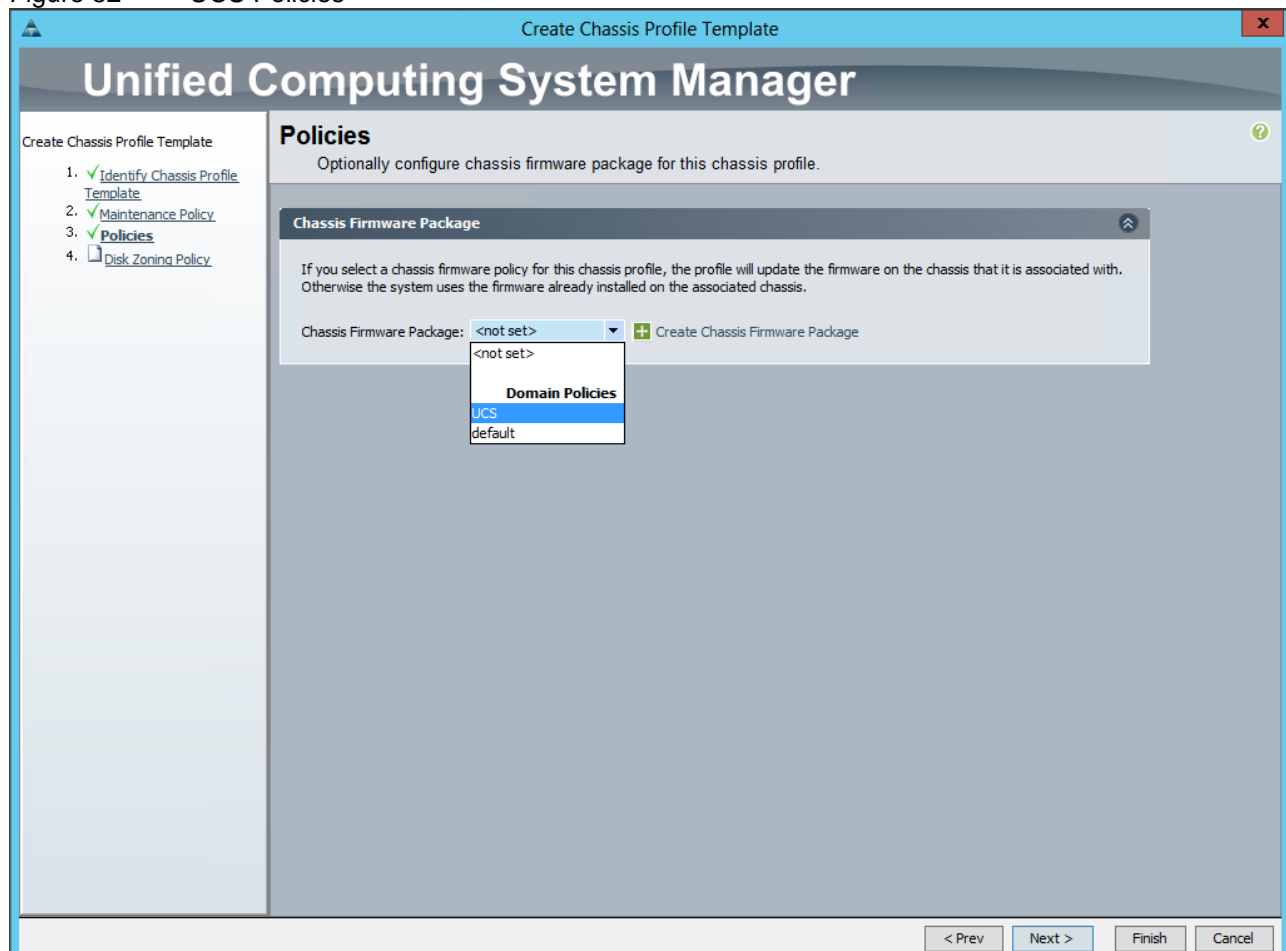
Figure 51 Identify Chassis Profile Template



3. Click Next and Next again.

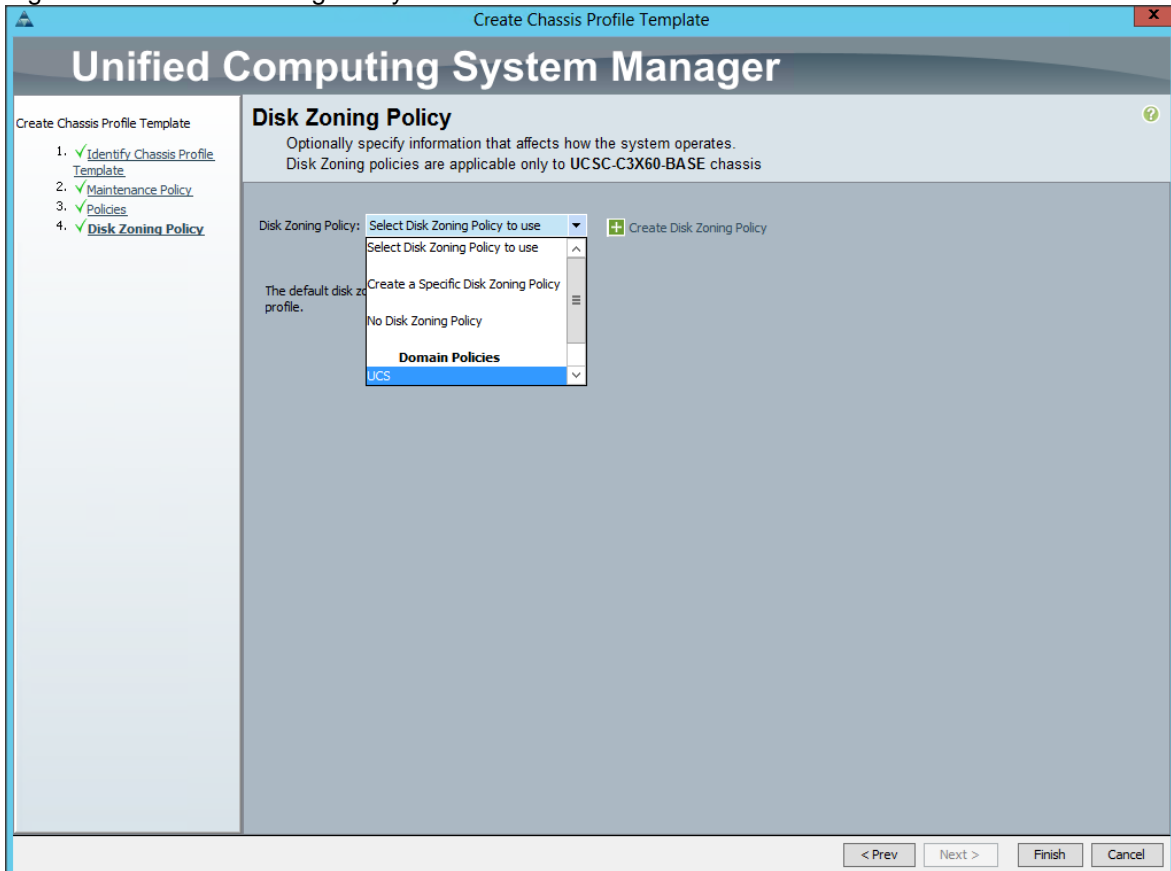
4. Expand the Chassis Firmware Package section. From the Chassis Firmware Package drop down list choose UCS and click Next. (Figure 52)

Figure 52 UCS Policies



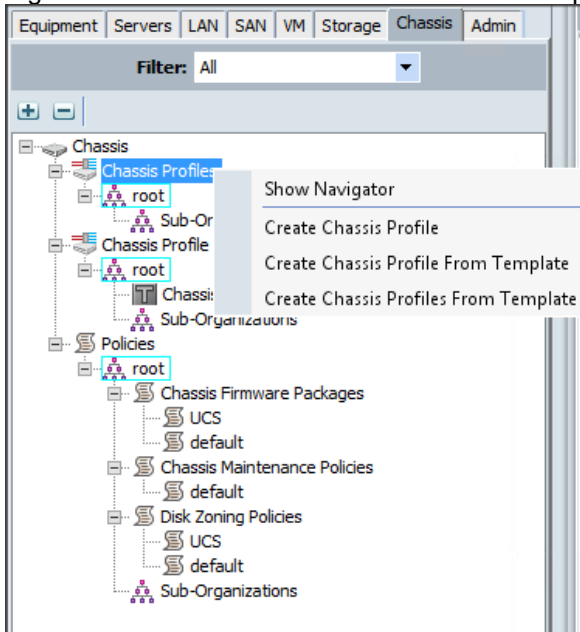
5. From the Disk Zoning Policy drop down list, choose UCS and click Finish. Click OK at the success message. (Figure 53)

Figure 53 Disk Zoning Policy



6. Right-click on Chassis Profiles and click Create Chassis Profile from Templates (not to be confused with the option to create multiple chassis profiles).(Figure 54)

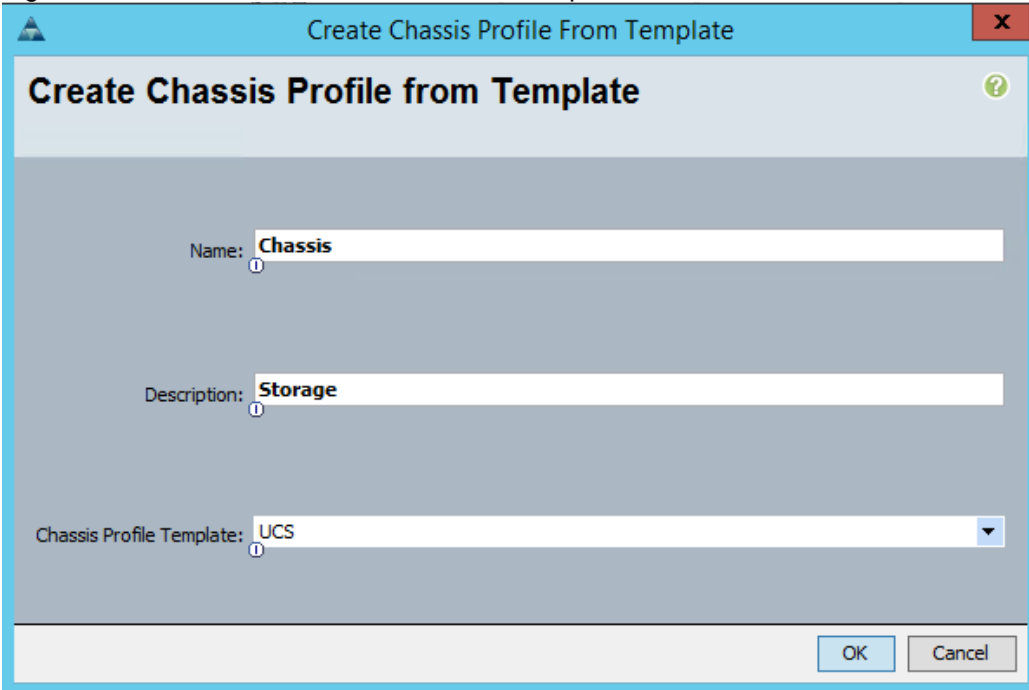
Figure 54 Create Chassis Profile from Templates



7. The Create Chassis Profile From Template window will appear. For Name, enter Chassis.

8. Enter a description (optional).
9. From the `Chassis Profile Template` drop down list, choose `UCS`. Click `OK`. (Figure 55)

Figure 55 Create Chassis Profile from Template



The screenshot shows a dialog box titled "Create Chassis Profile From Template". The dialog has a blue header bar with a close button (X) and a help icon (?). The main content area is light gray and contains three input fields:

- Name:** A text input field containing the text "Chassis".
- Description:** A text input field containing the text "Storage".
- Chassis Profile Template:** A dropdown menu with "UCS" selected.

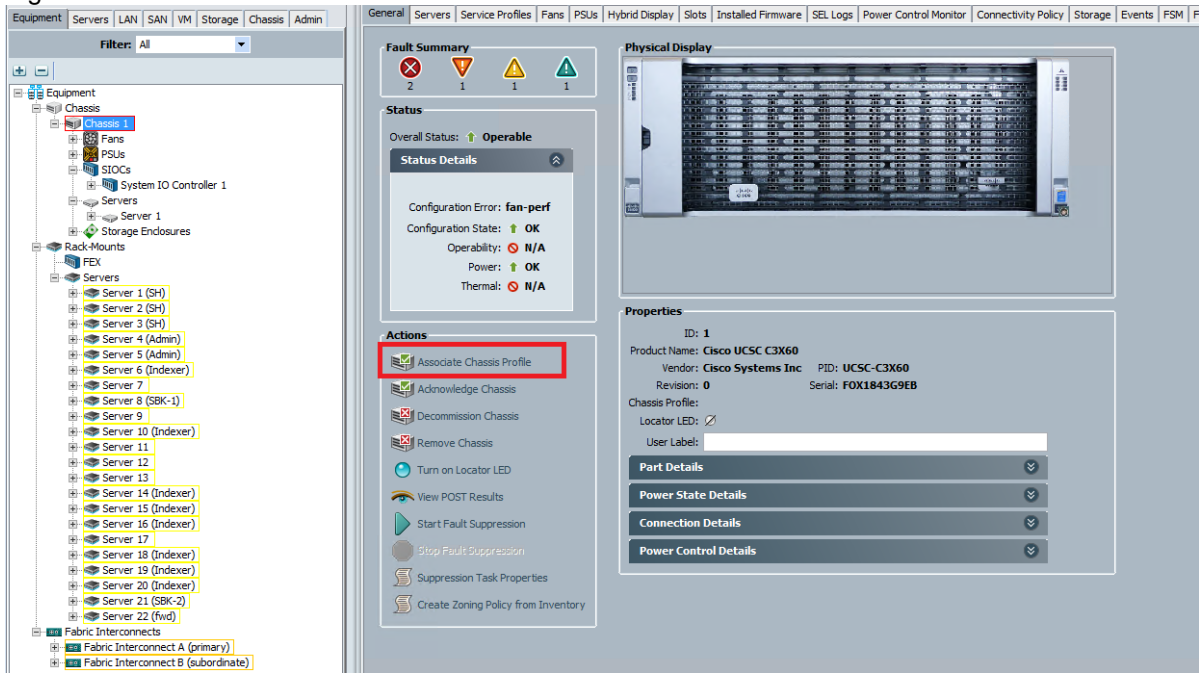
At the bottom right of the dialog, there are two buttons: "OK" and "Cancel".

10. Click `OK` on the success dialog box.

Associating Chassis Profile to Individual Chassis

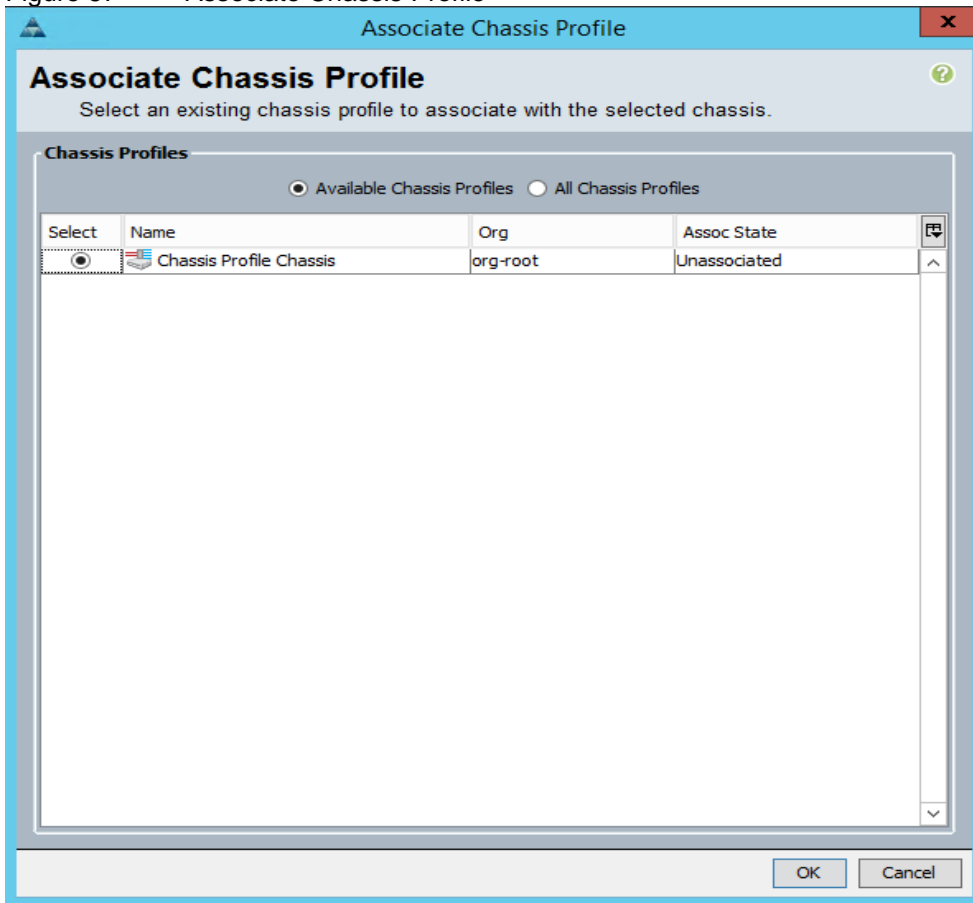
1. In the Cisco UCS Manager UI, select the `Equipment` tab. Under `Equipment`, expand `Chassis`.
2. Select the chassis and click `Associate Chassis Profile`. (Figure 56)

Figure 56 Associate Chassis Profile



3. Select Chassis Profile Chassis and click OK. (Figure 57)

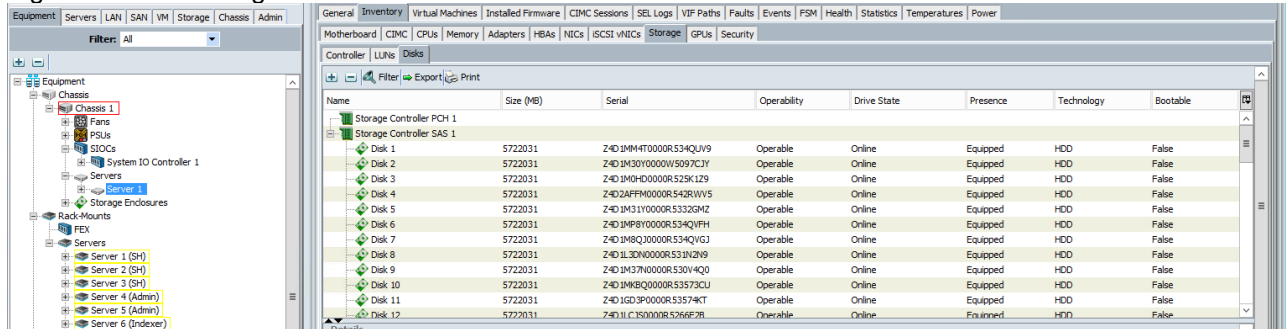
Figure 57 Associate Chassis Profile



4. Once the chassis profile is associated, all 60 disks will be assigned to the server node.

5. To verify that, go to Equipment → Chassis → Chassis 1 → Servers → Server 1.
6. Click on the tabs Inventory → Storage → Disks.
7. Expand Storage Controller SAS 1. (Figure 58)

Figure 58 Storage Controller SAS 1

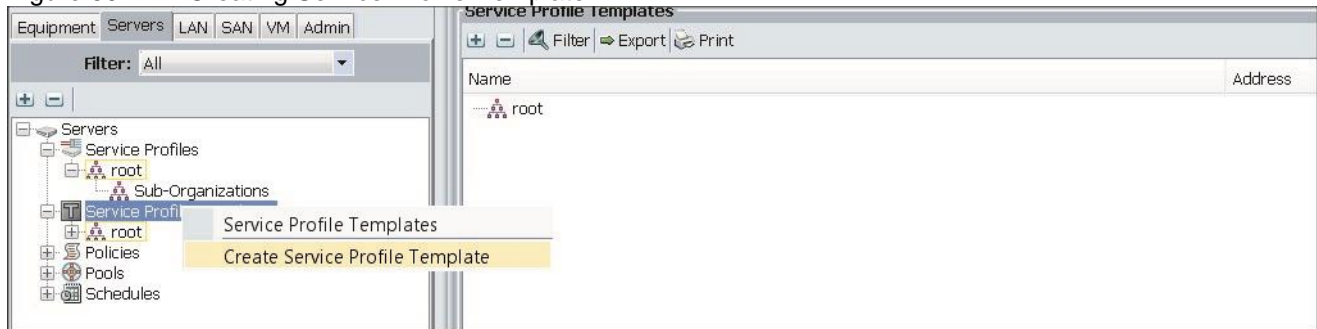


Creating a Service Profile Template

To create a service profile template, complete the following steps:

1. Select the Servers tab in the left pane in the UCS Manager GUI.
2. Right-click Service Profile Templates.
3. Select Create Service Profile Template. (Figure 59)

Figure 59 Creating Service Profile Template



The Create Service Profile Template window appears.

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

4. Enter the service profile template name: ucs. (Figure 60)
5. Click the Updating Template radio button.
6. In the UUID section, select Hardware Default as the UUID Assignment.
7. Click Next to continue to the next section.

Figure 60 Identify Service Profile Template

Create Service Profile Template

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.

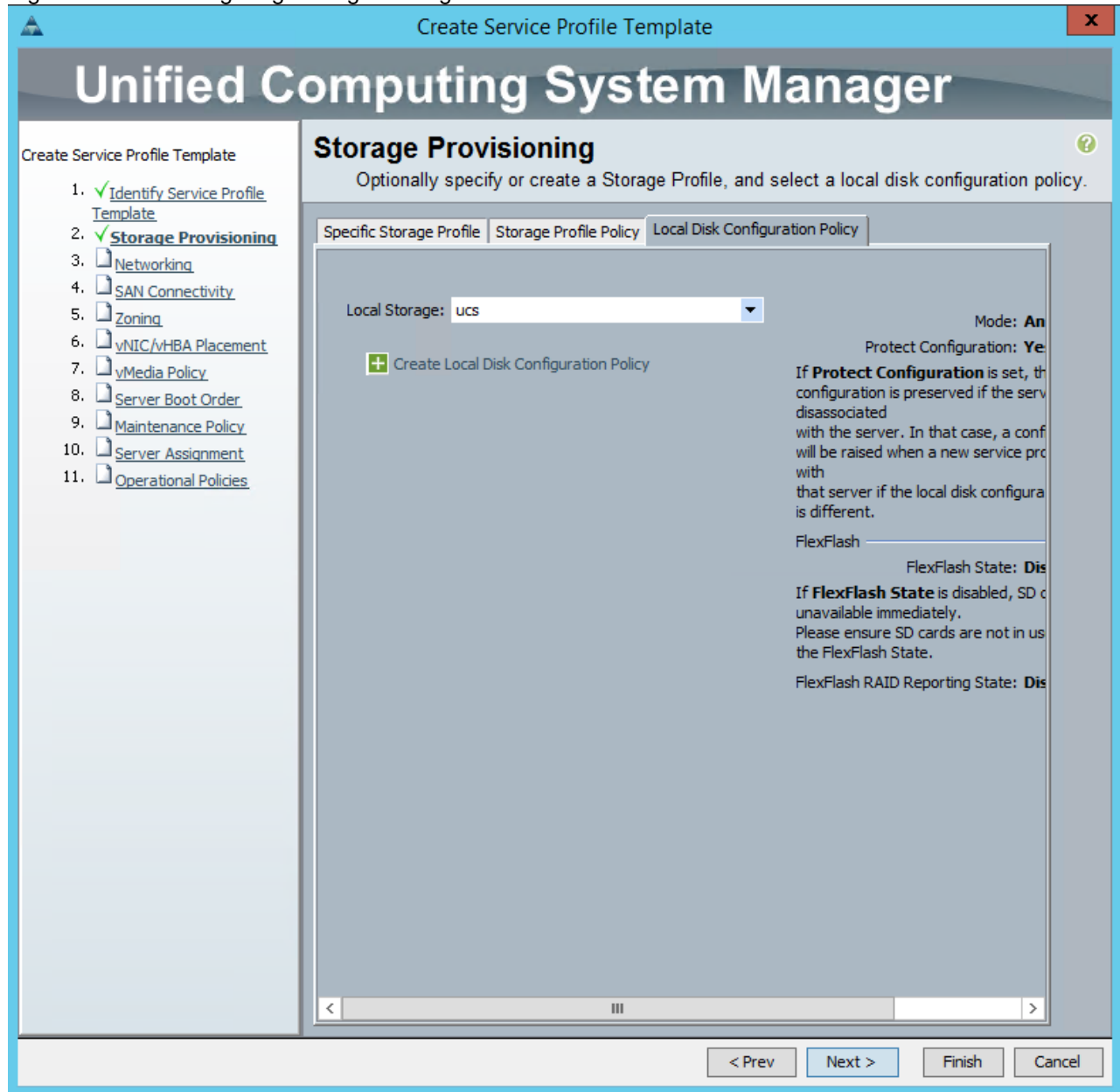
< Prev Next > Finish Cancel

Configuring the Storage Policy for the Template

To configure storage policies for the template, complete the following steps:

1. Click on the `Local Disk Configuration Policy` tab.
2. Select `ucs` for the local disk configuration policy. (Figure 61)
3. Click `Next` to continue to the next section.

Figure 61 Configuring Storage Settings

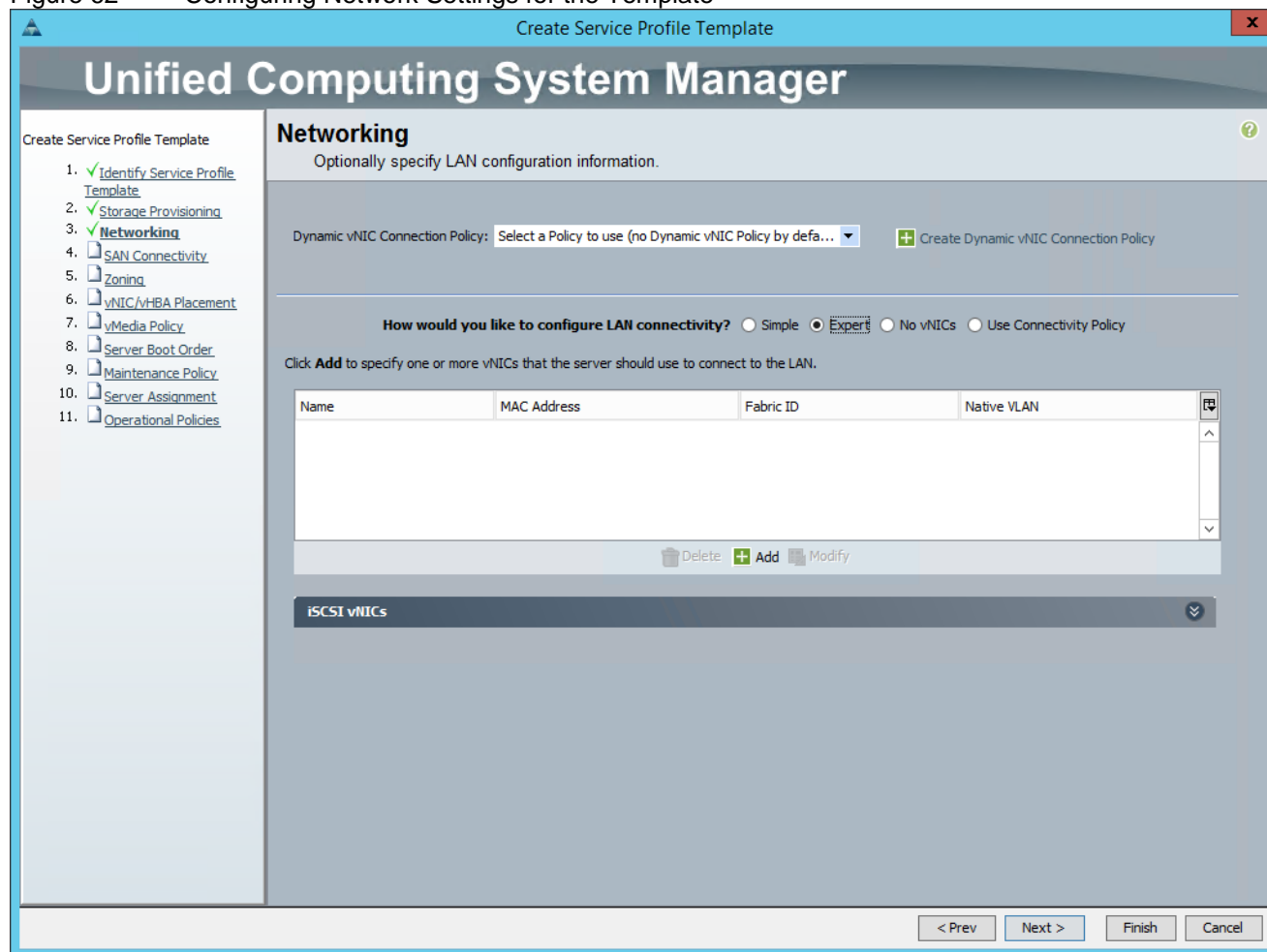


Configuring Network Settings for the Template

To configure the network settings for the template, complete the following steps:

1. Keep the `Dynamic vNIC Connection Policy` field set to default. (Figure 62)
2. Click the `Expert` radio button for the option, `How would you like to configure LAN connectivity?`

Figure 62 Configuring Network Settings for the Template



3. Click `Add` to add a vNIC to the template.
4. The `Create vNIC` window displays. Name the vNIC `eth0`. (Figure 63)
5. Select `ucs` in the `Mac Address Assignment` pool.
6. For `Fabric ID`, click the `Fabric A` radio button and check the `Enable failover` check box.
7. Check the default check box for VLANs and click the `Native VLAN` radio button.
8. Set `MTU size` to `9000`.
9. In the `Adapter Performance Profile` section, set `Adapter Policy` to `Linux`. Set `QoS Policy` to `Platinum`. Set `Network Control Policy` to `Default`.
10. In the `Connection Policies` section, keep the `Connection Policies` set at `Dynamic vNIC`. Keep the `Dynamic vNIC Connection Policy` as `<not set>`.
11. Click `OK`.

Figure 63 Configuring vNIC eth0

Create vNIC

Name:

Use vNIC Template:

MAC Address

MAC Address Assignment:

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 clash.

VLANs

Select	Name	Native VLAN
<input checked="" type="checkbox"/>	default	<input checked="" type="radio"/>
<input type="checkbox"/>	vlan11_Appliance	<input type="radio"/>
<input type="checkbox"/>	vlan11_DATA1	<input type="radio"/>
<input type="checkbox"/>	vlan12_DATA2	<input type="radio"/>

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:

Operational Parameters

Adapter Performance Profile

Adapter Policy:

QoS Policy:

Network Control Policy:

Connection Policies

Dynamic vNIC usNIC VMQ

Dynamic vNIC Connection Policy:

12. Click **Add** to add a vNIC to the template.

13. The **Create vNIC** window appears. Name the vNIC `eth1`. (Figure 64)

14. Select `ucs` in the `Mac Address Assignment pool`.
15. Click the `Fabric B` radio button and check the `Enable failover` check box for the `Fabric ID`.
16. Check the `vlan11_DATA1` check box for VLANs, and click the `Native VLAN` radio button
17. Set `MTU size` as `9000`.
18. In the `Adapter Performance Profile` section, set `Adapter Policy` as `Linux`. Set `QoS Policy` as `Platinum`. Set `Network Control Policy` as `Default`.
19. In the `Connection Policies` section, keep the `Connection Policies` as `Dynamic vNIC`. Keep the `Dynamic vNIC Connection Policy` as `<not set>`.
20. Click `OK`.

Figure 64 Configuring vNIC eth1

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 clash.

VLANs

Filter Export Print

Select	Name	Native VLAN
<input type="checkbox"/>	default	<input type="radio"/>
<input type="checkbox"/>	vlan11_Appliance	<input type="radio"/>
<input checked="" type="checkbox"/>	vlan11_DATA1	<input checked="" type="radio"/>
<input type="checkbox"/>	vlan12_DATA2	<input type="radio"/>

+ Create VLAN

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: **+ Create Ethernet Adapter Policy**

QoS Policy: **+ Create QoS Policy**

Network Control Policy: **+ Create Network Control Policy**

Connection Policies

Dynamic vNIC usNIC VMQ

Dynamic vNIC Connection Policy: **+ Create Dynamic vNIC Connection Policy**

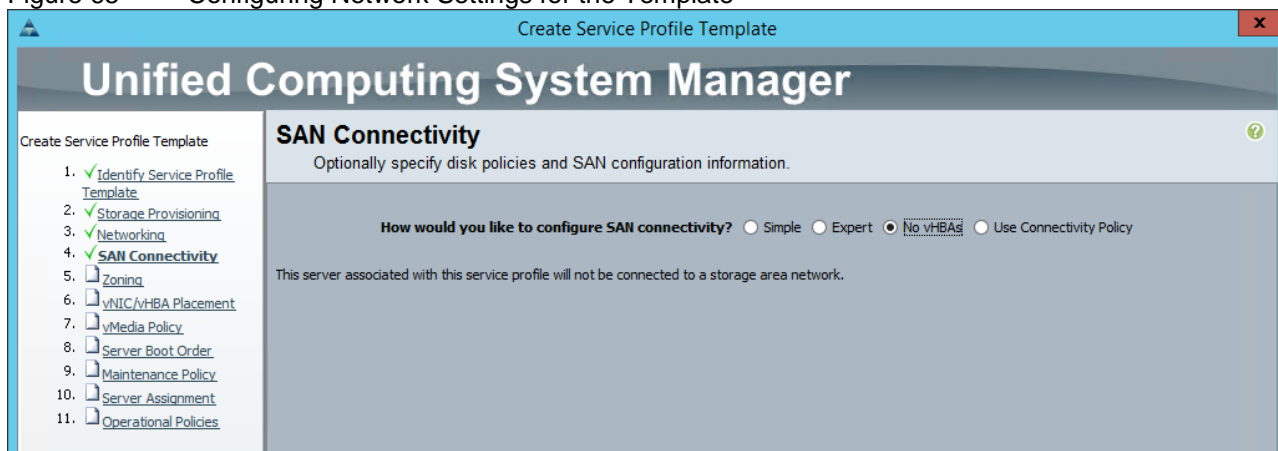
OK Cancel

Configuring SAN Connectivity for the Template

To configure SAN connectivity, complete the following steps:

1. For How would you like to configure SAN connectivity?, select no vHBAs. (Figure 65)
2. Click Next to go to the next section.
3. Zoning information is not specified. Click Next to go to the next section.

Figure 65 Configuring Network Settings for the Template

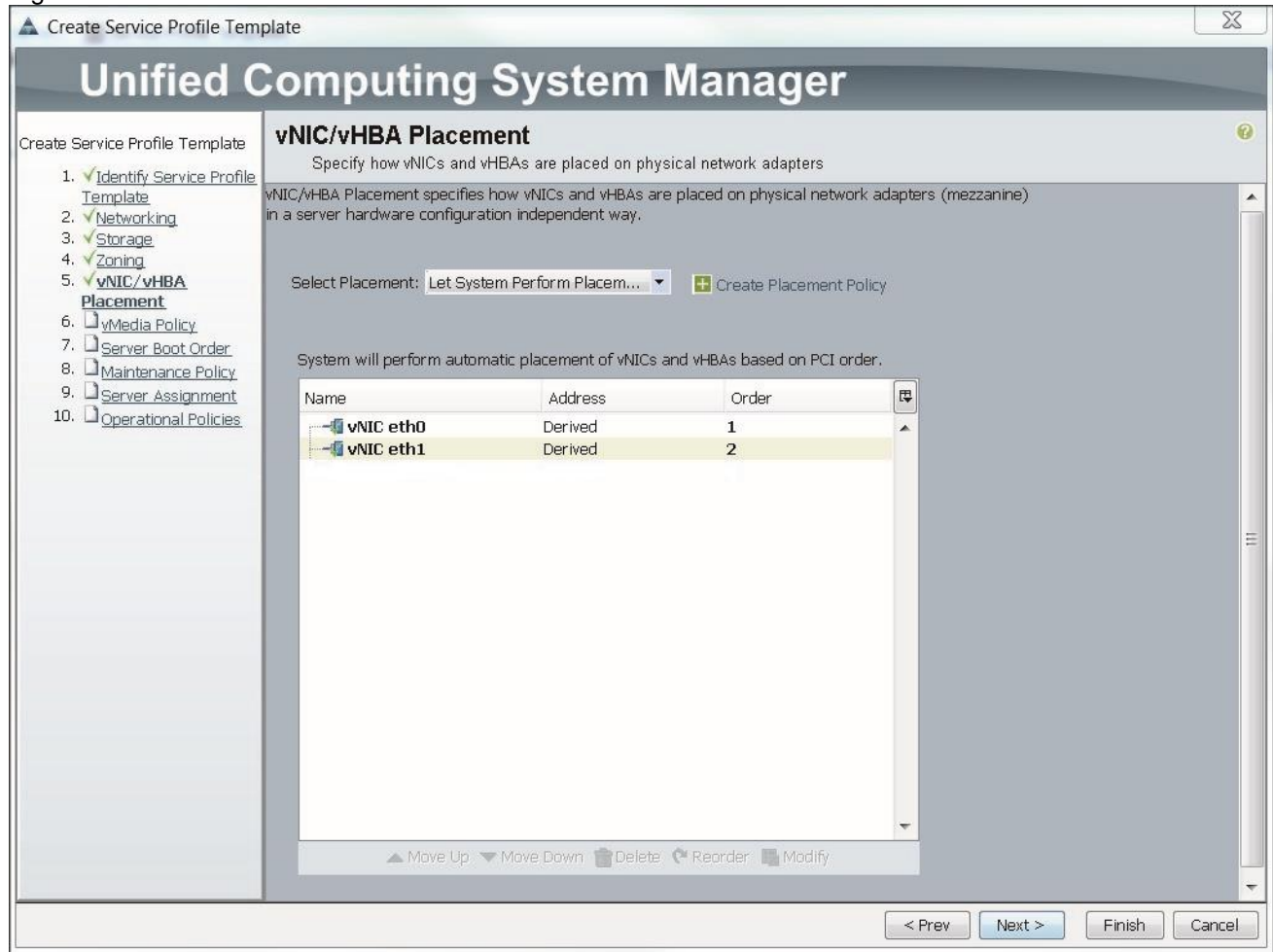


Configuring vNIC/vHBA Placement Policy for the Template

To configure the vNIC/vHBA placement policy, complete the following steps:

1. Keep the default option for the `Select Placement` field. (Figure 66)
2. Select `eth0` and `eth1` and assign the vNICs in the following order:
 - a. `eth0`
 - b. `eth1`
3. Review to make sure that both vNICs were assigned in the appropriate order.
4. Click `Next` to continue to the next section.

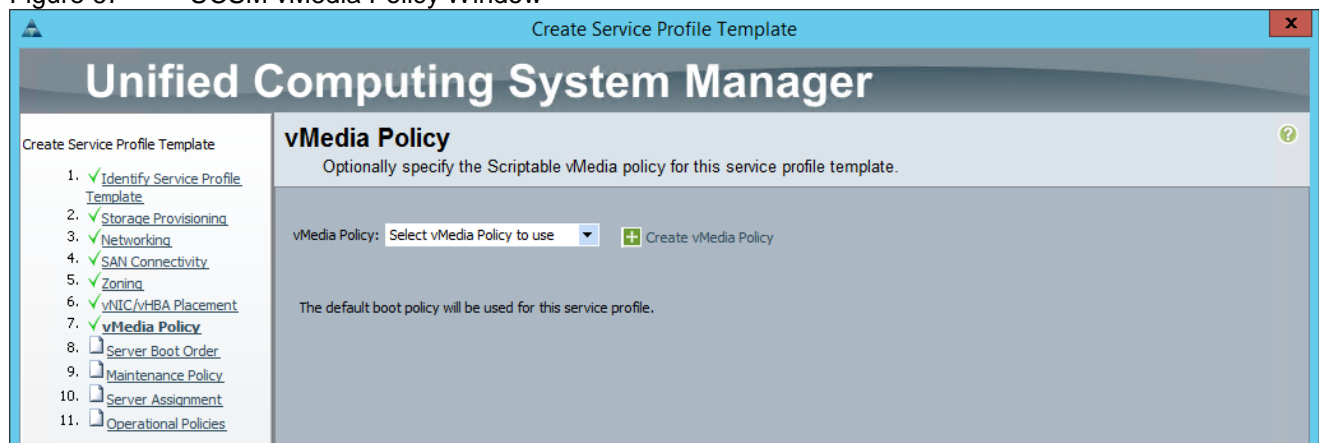
Figure 66 vNIC/vHBA Placement



Configuring vMedia Policy for the Template

1. Skip the vMedia policy. (Figure 67) Click **Next** to go to the next section.

Figure 67 UCSM vMedia Policy Window

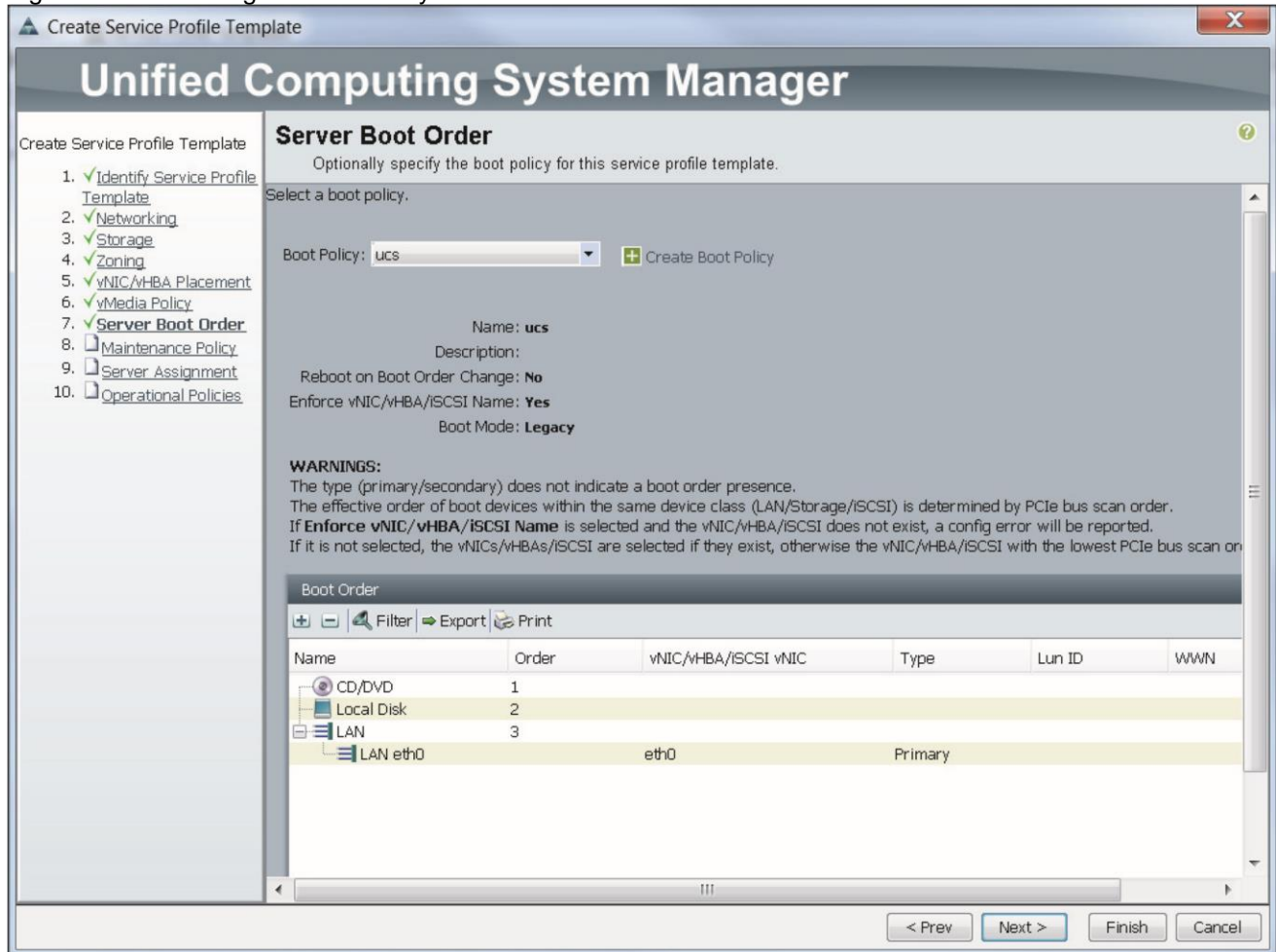


Configuring Server Boot Order for the Template

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

1. Select `ucs` in the `Boot Policy` name field. (Figure 68)
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.

Figure 68 Configure Boot Policy

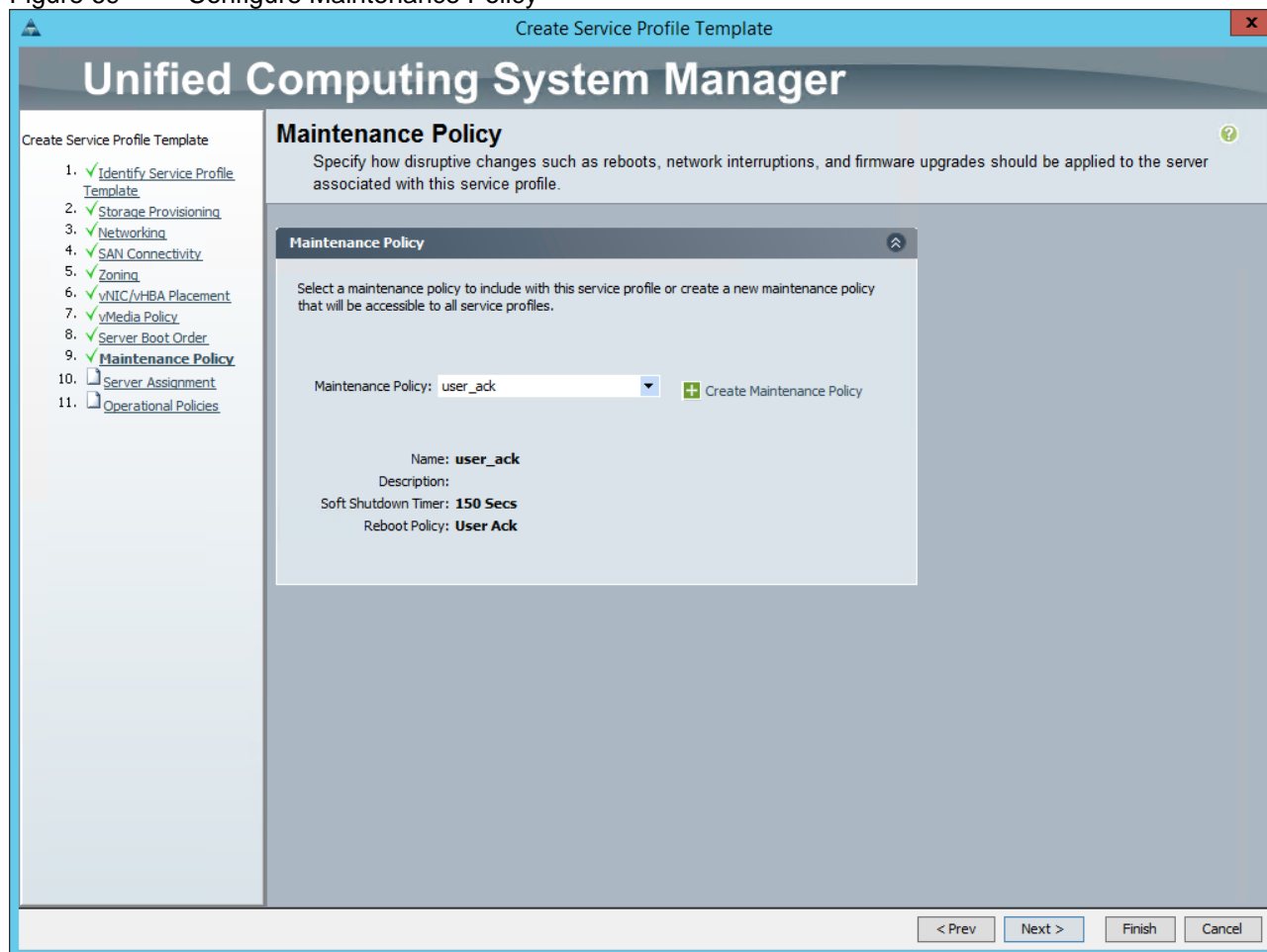


Configuring the Maintenance Policy for the Template

To configure the maintenance policy, complete the following steps:

1. Select `user_ack` at the `Maintenance Policy` field. (Figure 69)
2. Click `Next` to continue to the next section.

Figure 69 Configure Maintenance Policy

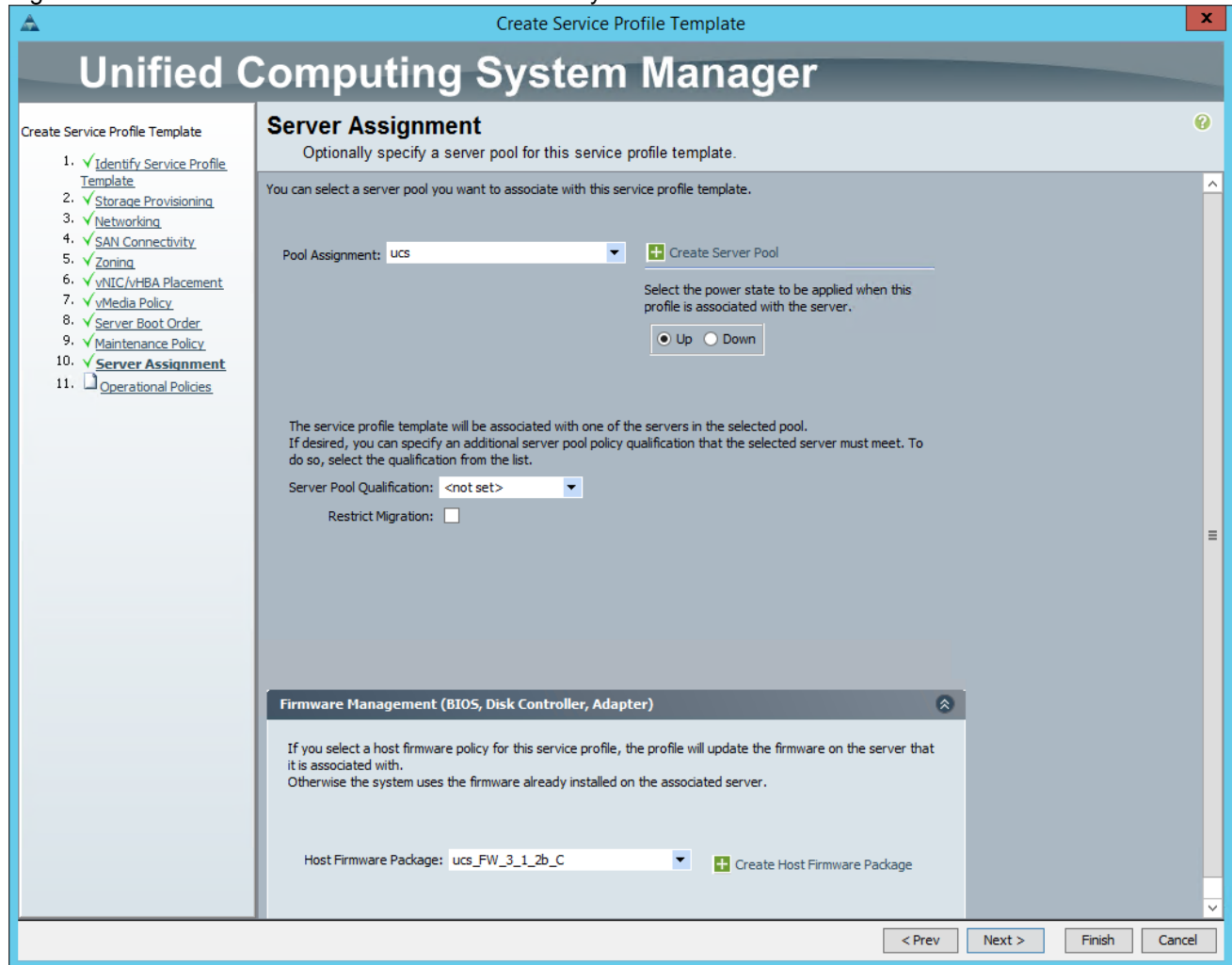


Configuring Server Assignment for the Template

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

1. Select `ucs` for the `Pool Assignment` field. (Figure 70)
2. Keep the `Server Pool Qualification` field at default.
3. Expand the `Firmware Management` section.
4. For the `Host Firmware Package`, select `ucs_FW_3_1_1h_C` from the drop-down list.

Figure 70 Server Pool and Host Firmware Policy Selection

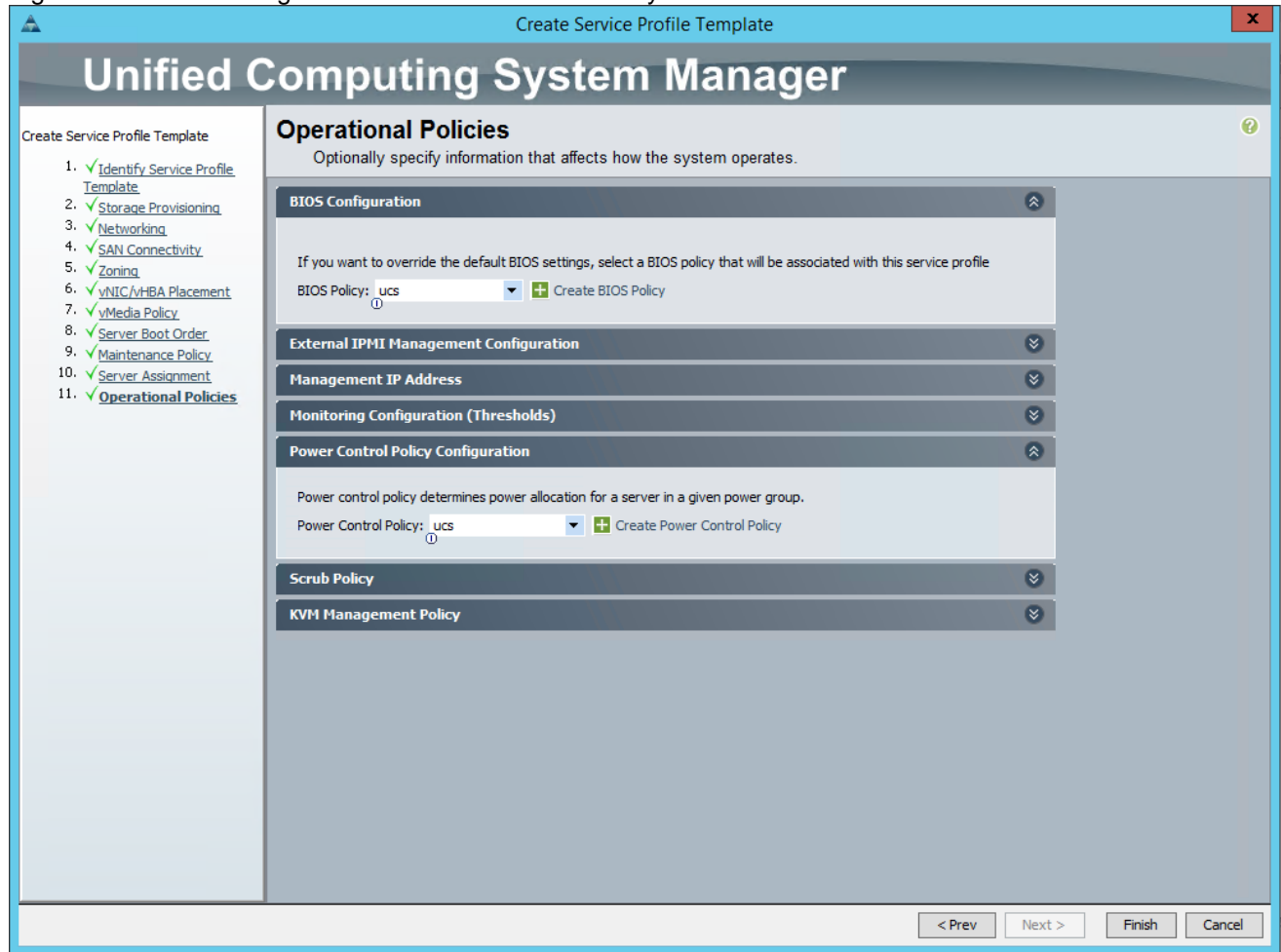


Configuring Operational Policies for the Template

In the Operational Policies Window, complete the following steps:

1. In the BIOS Configuration section, select `ucs` in the BIOS Policy field. (Figure 71)
2. Expand the Power Control Policy Configuration section. 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.

Figure 71 Selecting BIOS and Power Control Policy

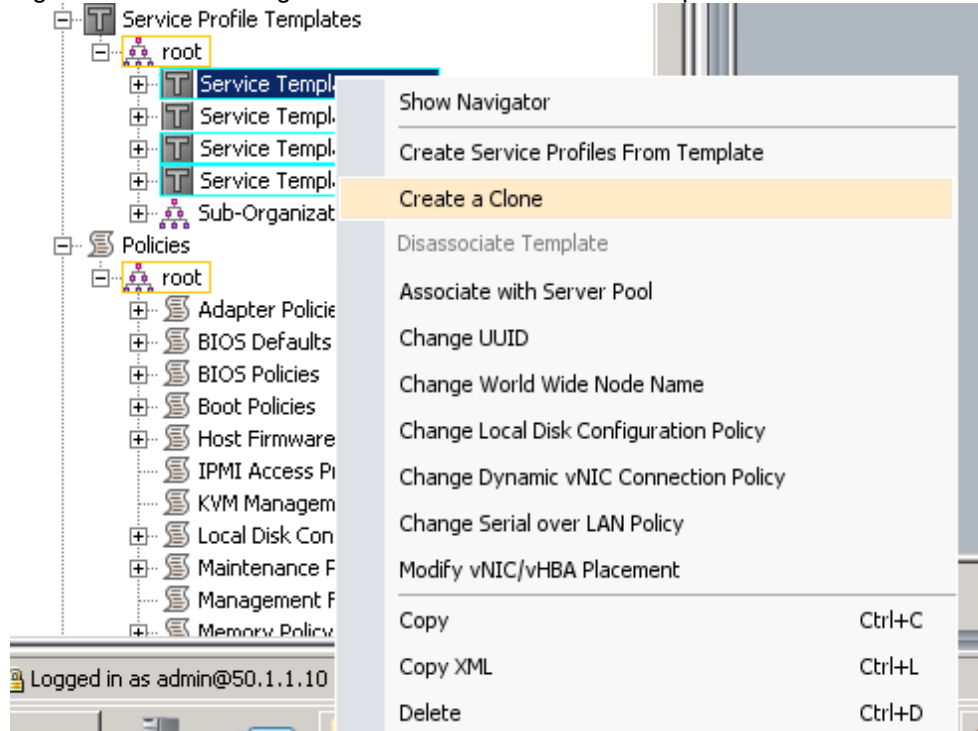


Creating a Service Profile Template for C3260

We need a separate service profile template for the server in C3260. We will start with a copy of our existing service profile and add a storage profile and boot policy.

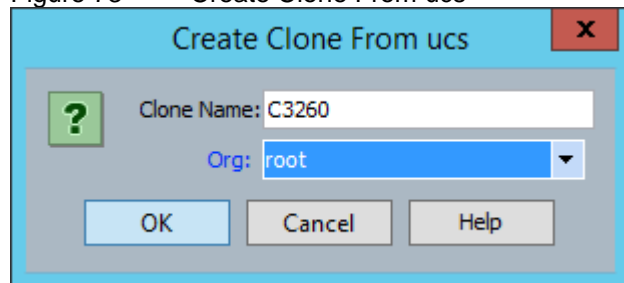
1. Select the `Servers` tab in the navigation pane.
2. Go to `Servers > Service Profile Templates > root`.
3. Right-click `Service Profile Templates ucs`.
4. Select `Create a Clone`. (Figure 72)

Figure 72 Creating a Clone of a Service Profile Template



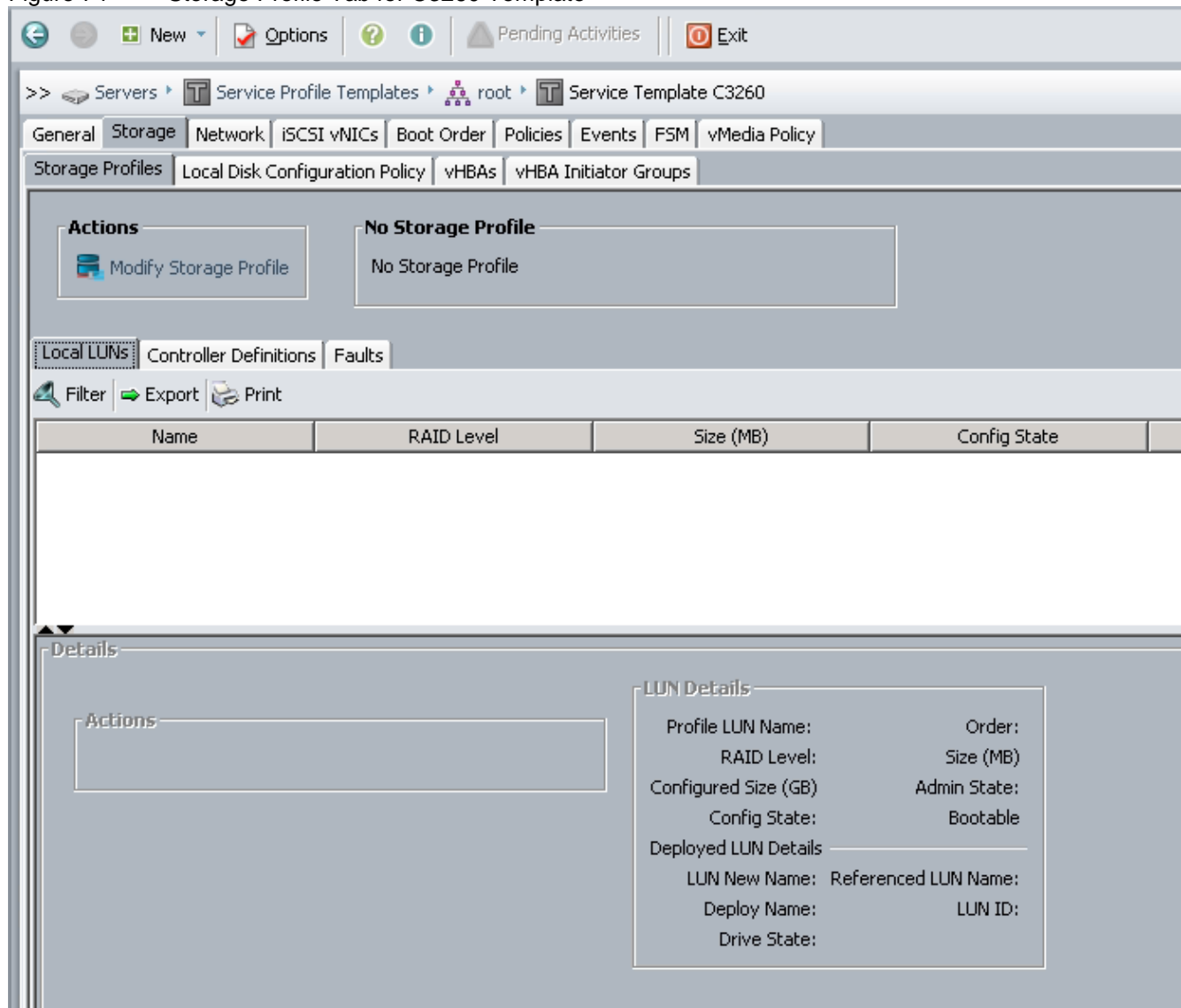
5. Enter C3260 for the Clone Name and select root for the organization. (Figure 73)

Figure 73 Create Clone From ucs



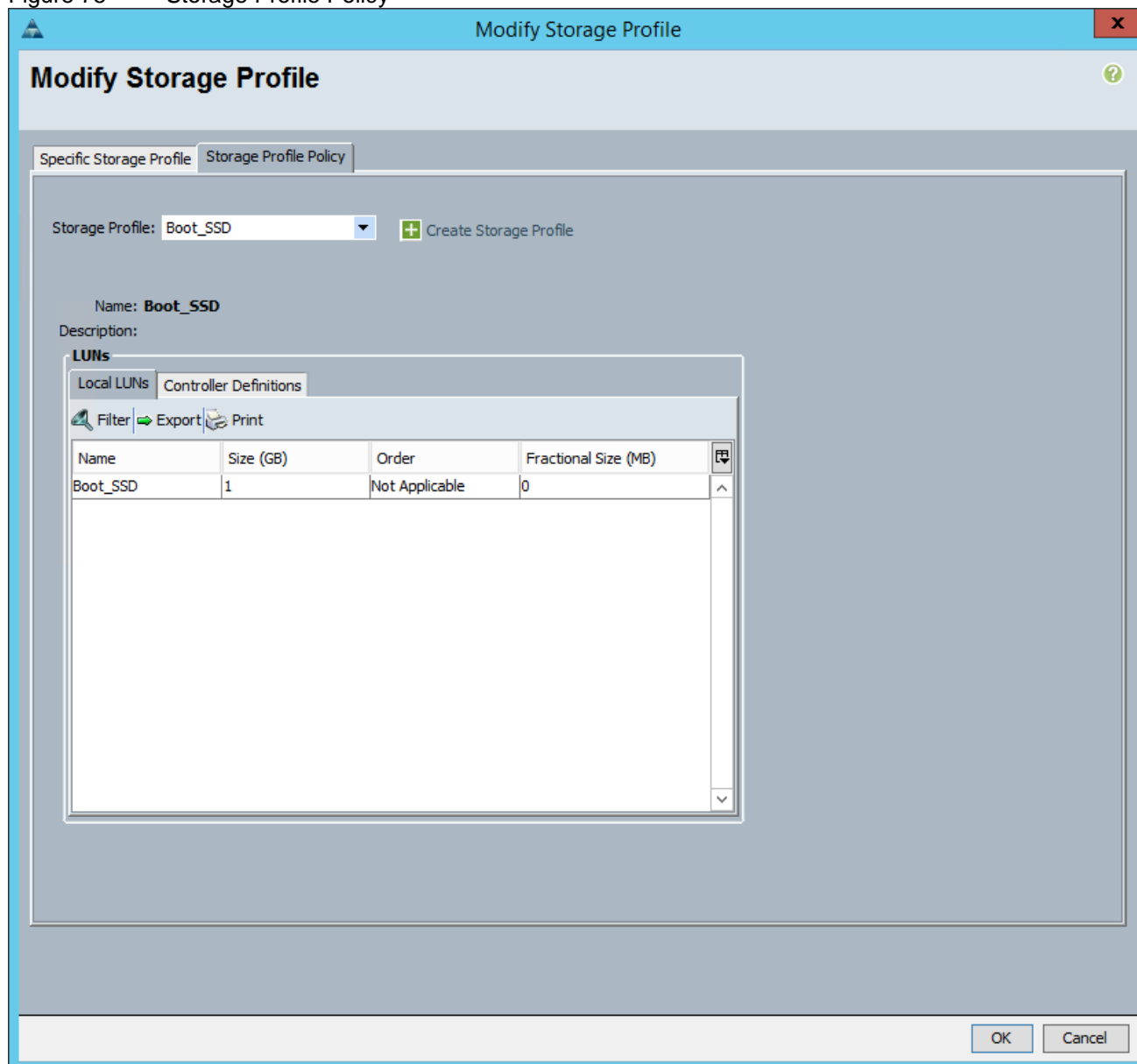
6. Select the new service template in the navigation pane.
7. Select the Storage and Storage Profiles tabs. (Figure 74)

Figure 74 Storage Profile Tab for C3260 Template



8. Click on Modify Storage Profile.
9. Click on the Storage Profile Policy tab.
10. Choose Boot_SSD for the Storage Profile. (Figure 75)

Figure 75 Storage Profile Policy



11. Click **OK** to finish modifying the storage profile. Click **OK** on the success dialog box.
12. Click on the **Boot Order** tab. (Figure 76)

Figure 76 Boot Order

Actions

Modify Boot Policy

Global 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
Local CD/DVD	1					
Local Disk	2					
LAN	3					
LAN eth0		eth0	Primary			

Create iSCSI vNIC Set iSCSI Boot Parameters Set Uefi Boot Parameters

13. The current boot policy is `ucs`. Click on `Modify Boot Policy` to change it.
14. Click `Create Boot Policy`.
15. Enter `LUN` for the name.
16. Expand `Local Devices` and select `Add Local LUN`. (Figure 77)

Figure 77 Add Local LUN Image Path

Add Local LUN Image Path

Type: Primary Secondary Any

LUN Name:

OK Cancel

17. In the Add Local LUN Image Path window, select Primary, enter a LUN Name, and click OK. (Figure 78)

Figure 78 Create Boot Policy

Create Boot Policy

Name:

Description:

Reboot on Boot Order Change:

Enforce vNIC/vHBA/iSCSI Name:

Boot Mode: Legacy Uefi

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

Local Devices

- Add Local Disk
 - Add Local LUN
 - Add Local JBOD
 - Add SD Card
 - Add Internal USB
 - Add External USB
 - Add Embedded Local LUN
 - Add Embedded Local Disk
- Add CD/DVD
 - Add Local CD/DVD
 - Add Remote CD/DVD
- Add Floppy
 - Add Local Floppy
 - Add Remote Floppy
- Add Remote Virtual Drive

Boot Order

Name	Order	vNIC/vHBA/iSCSI v...	Type	LUN Name	WWN	Slot Number	Boot Name	Boot Path	Description
Local LUN	1								
Local Lun Image			Primary	C3260					

Move Up Move Down Delete

Set Uefi Boot Parameters

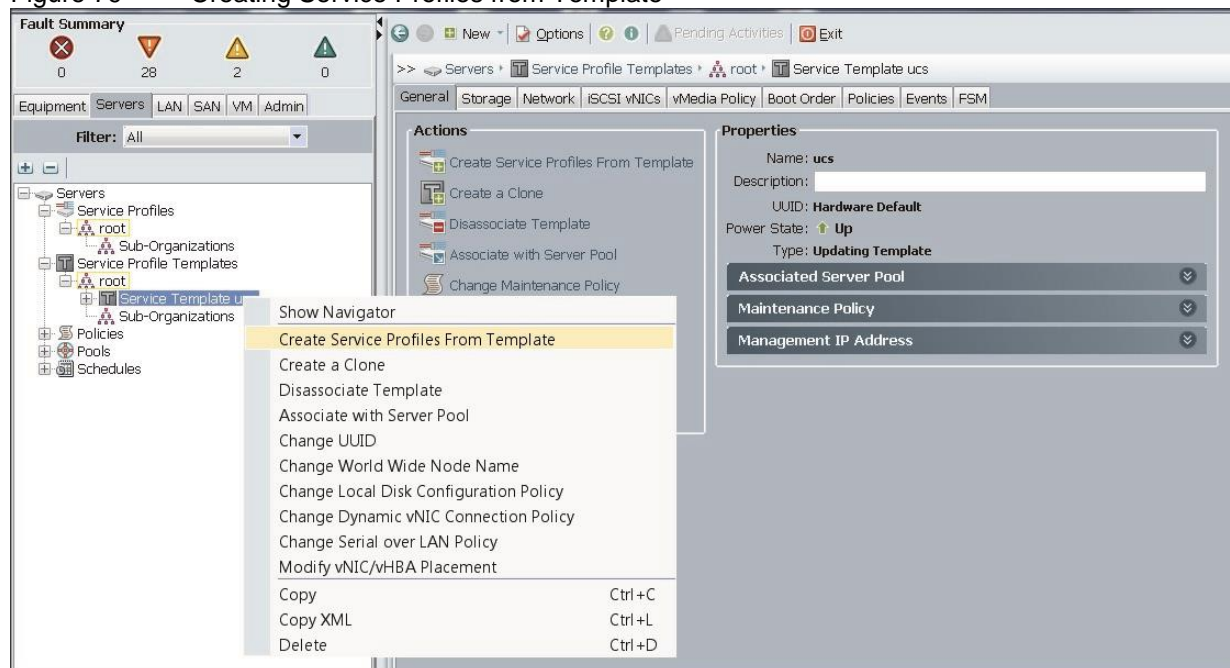
OK Cancel

18. After creating the new boot policy, select it in the Modify Boot Policy screen and click OK.

Creating Service Profiles from Template

1. Select the `Servers` tab in the left pane of the UCS Manager GUI.
2. Go to `Service Profile Templates > root`. (Figure 79)
3. Right-click `Service Profile Templates ucs`.
4. Select `Create Service Profiles From Template`.

Figure 79 Creating Service Profiles from Template



The Create Service Profile from Template window appears.

5. In the `Create Service Profile from Template` window enter the following:
 - a. In the field `Naming Prefix`, enter `ucs`. (Figure 80)
 - b. In the field `Enter Name Suffix Starting Number`, enter `1`.
 - c. In the field `Number of Instances`, enter `13`.

Figure 80 Selecting Name and Total number of Service Profiles

Create Service Profiles From Template

Naming Prefix:

Name Suffix Starting Number:

Number of Instances:

OK Cancel



Association of the Service Profiles will take place automatically.

- Repeat the above steps to create a service profile from the C3260 Service Profile Template. Use the parameters in Figure 81 .

Figure 81 Selecting Name and Total number of Service Profiles for C3260

Create Service Profiles From Template

Naming Prefix:

Name Suffix Starting Number:

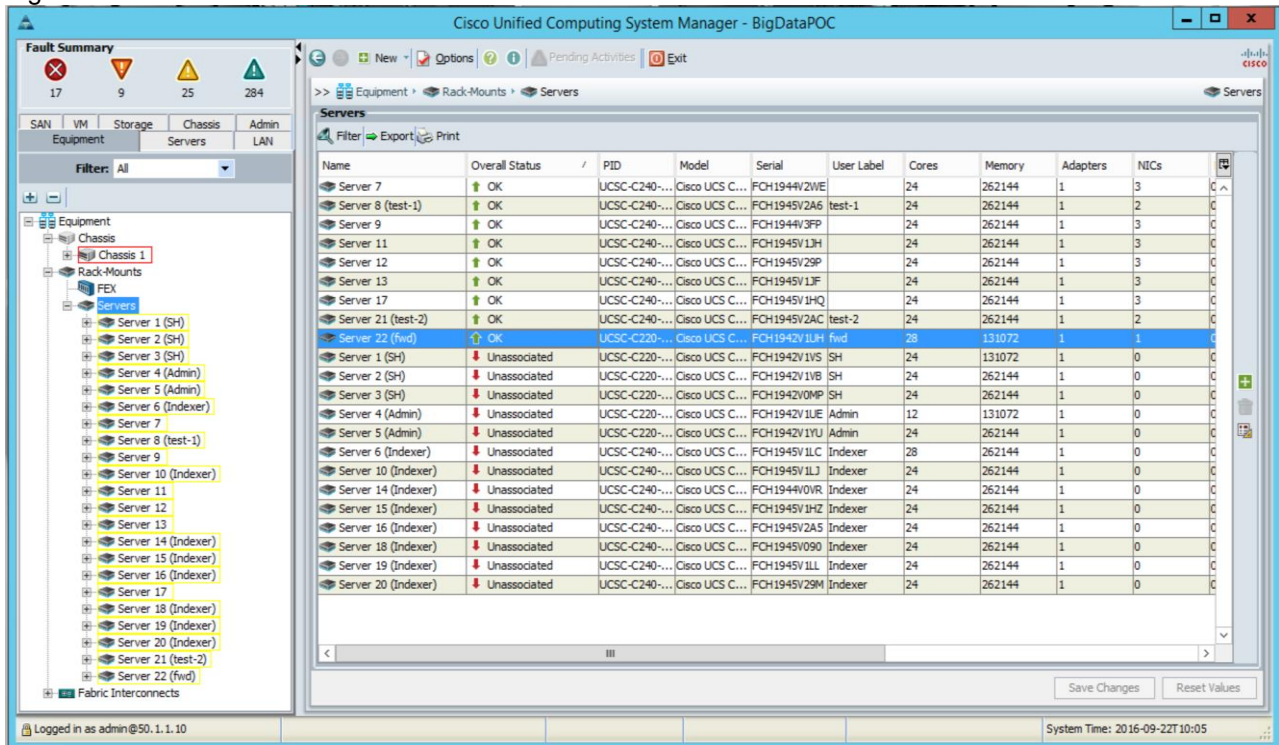
Number of Instances:

OK Cancel

Identifying the Servers

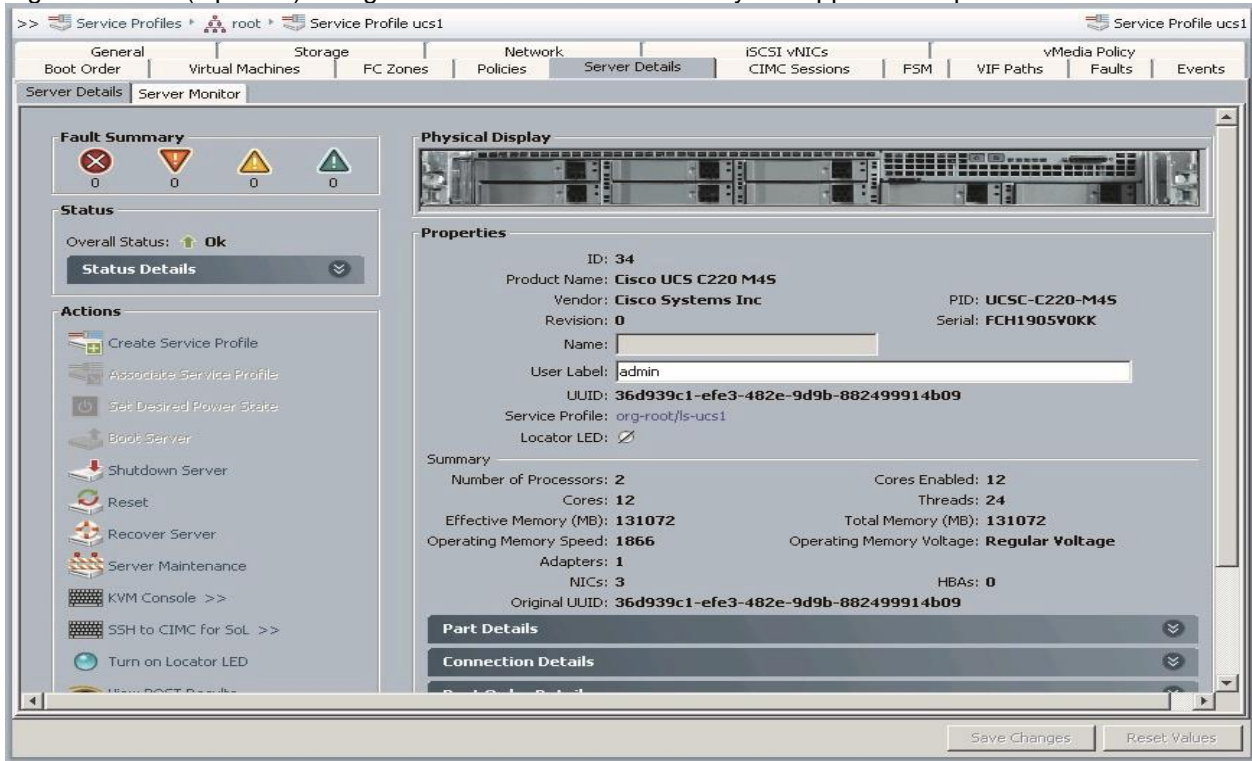
- In the **Equipment Tab**, select **Rack-Mounts** for the **Filter**, and click on **Servers**. In the right pane all thirteen servers are displayed along with their details. (Figure 82)

Figure 82 Cisco UCS Rack Servers Associated with Created Service Profiles



- (Optional) Double click on an individual server instance and enter an appropriate text string (name or role) in the User Label as shown below. (Figure 83) This could be helpful in identifying the server's application specific roles.

Figure 83 (Optional) Using the User Label Field to Identify the Application Specific Roles



Installing Red Hat Enterprise Linux 6.8 on C220 M4 Systems

The search heads and the admin nodes are C220 M4 servers. These servers should be populated with a minimum of 2 identical hard disk drives. The procedures documented in this section are applicable for all servers performing the admin and the search head functions.



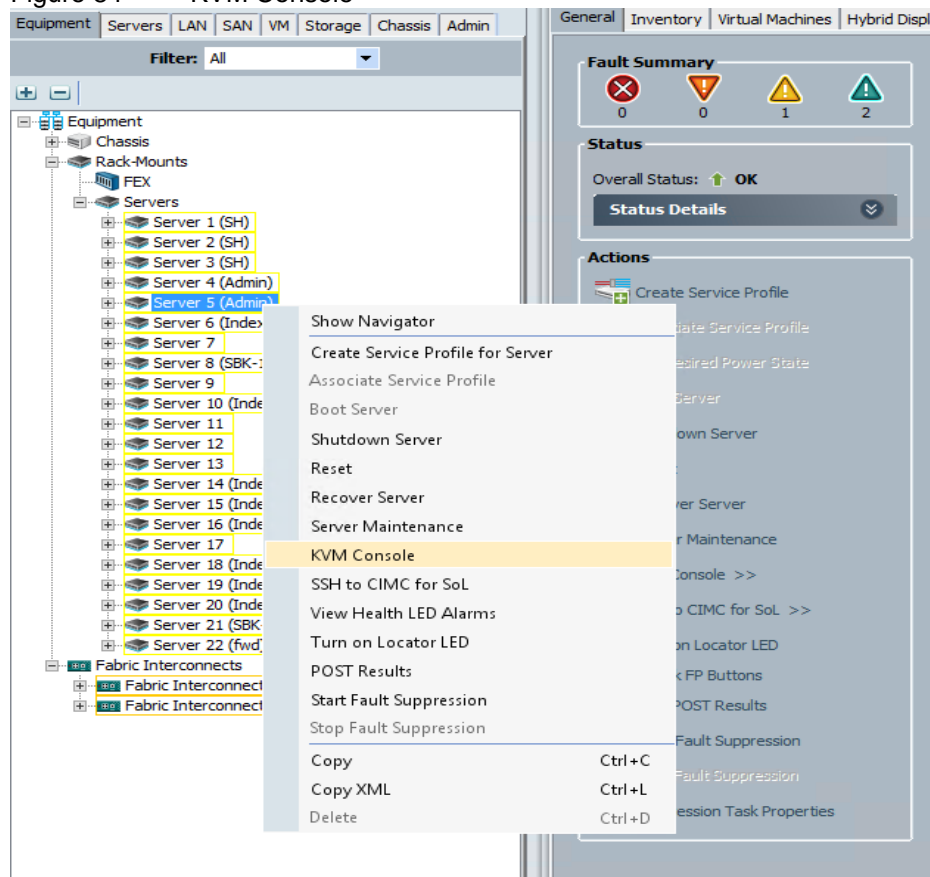
Note: This requires RHEL 6.8 DVD/ISO for the installation.

Creating a Virtual Drive Using Cisco 12G SAS RAID Controller Utility

To create a virtual drive using Cisco 12G SAS RAID Controller Utility, complete the following steps:

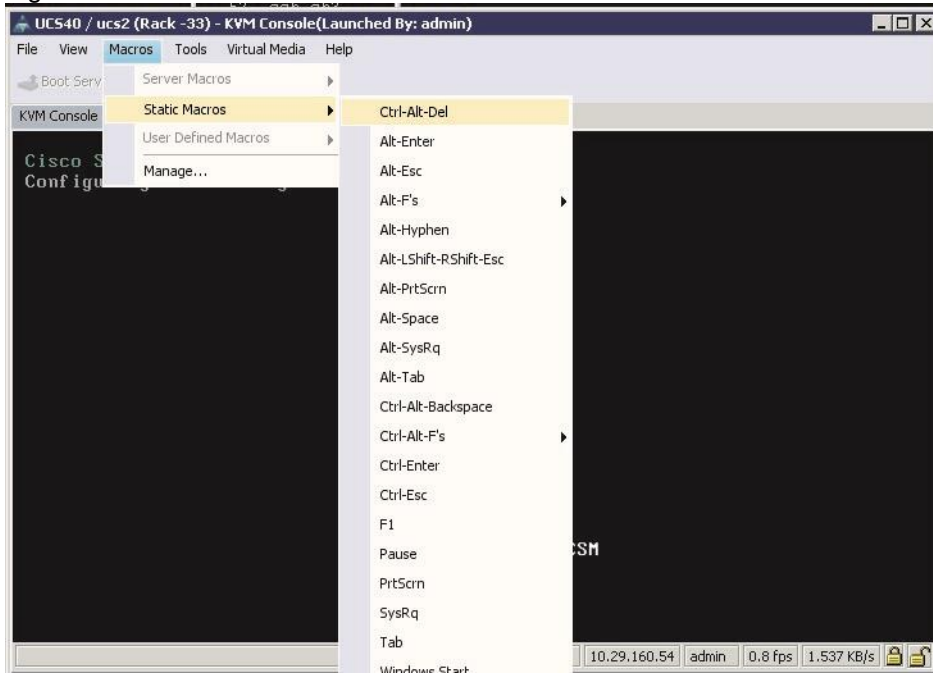
1. Log in to the Cisco UCS 6296 Fabric Interconnect and launch the Cisco UCS Manager application.
2. Select the `Equipment` tab.
3. In the navigation pane, expand `Rack-Mounts` and then `Servers`.
4. Right-click on the C220 server that will serve as the admin1 node and select `KVM Console`. (Figure 84)

Figure 84 KVM Console



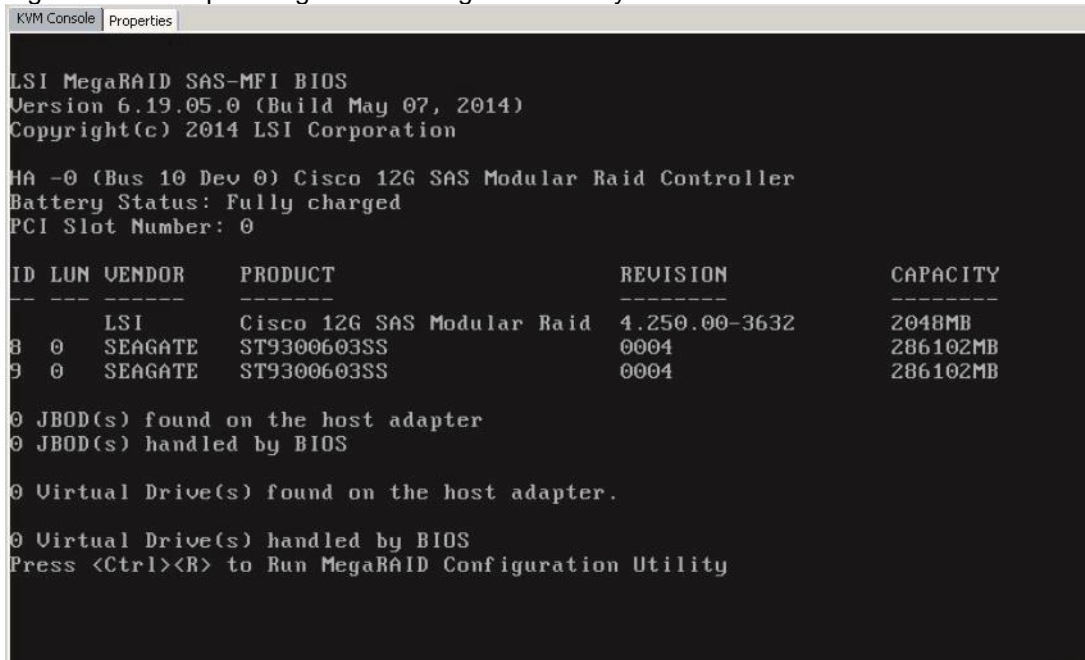
5. In the KVM window, select `Macros > Static Macros > Ctrl-Alt-Del`. (Figure 85)

Figure 85 Restart the Server



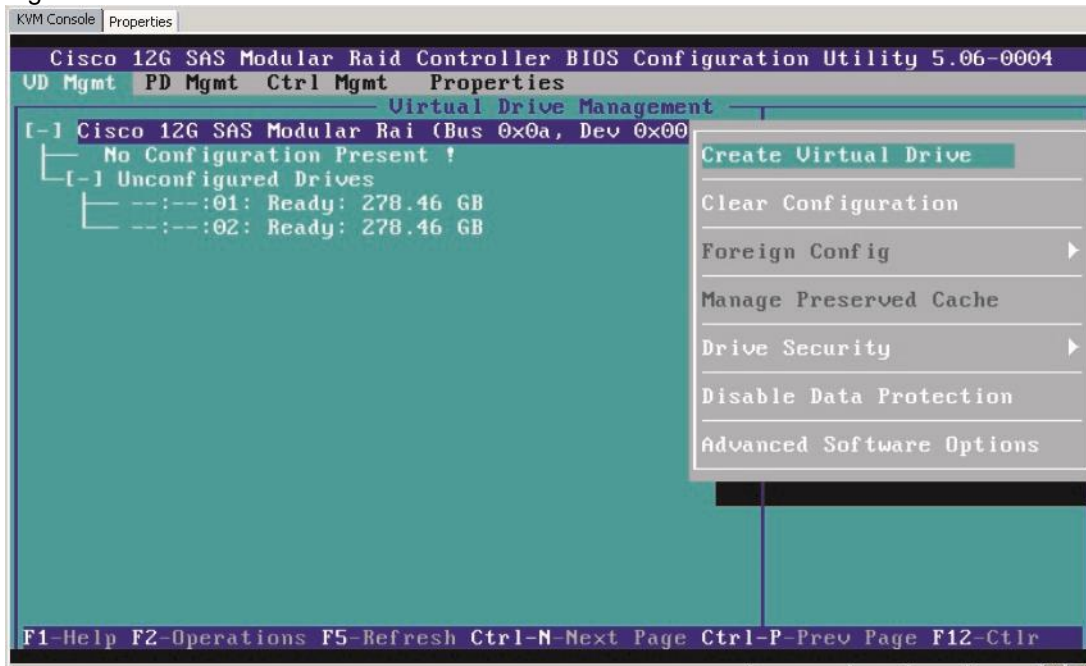
6. Wait for the initial server configurations and POST to complete. Press `Ctrl-R` when the next screen appears. (Figure 86) This will take you to the configuration utility for the controller.

Figure 86 Open MegaRAID Configuration Utility



7. In the Cisco 12G SAS Modular Raid Controller BIOS Configuration Utility, use the arrow keys to highlight the Cisco 12G SAS Modular Raid controller line item.
8. Press `F2` to open up the sub-menu. Select `Create Virtual Drive`. (Figure 87)

Figure 87 Create Virtual Drive



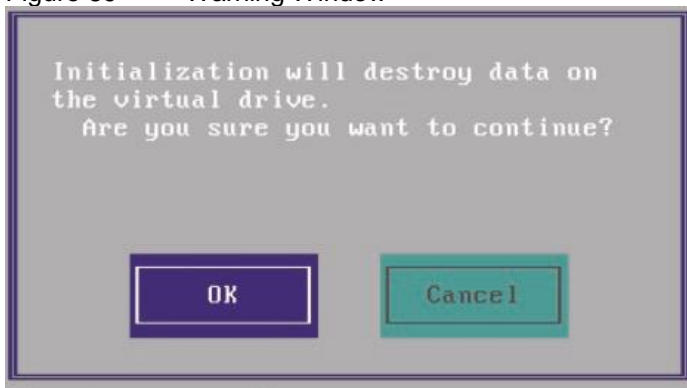
9. In the Create New VD dialog box, highlight the RAID Level field, and press Enter to select RAID-1.
10. Select the drives.
11. Press Enter on the Advanced button.
12. In the Create Virtual Drive-Advanced dialog box, select 128KB as the Strip Size. (Figure 88)
13. Select Read Ahead as the Read Policy.
14. Select Write Back with BBU as the Write Policy.
15. Select Direct as the I/O Policy.

Figure 88 Advanced Settings for Creating a Virtual Drive



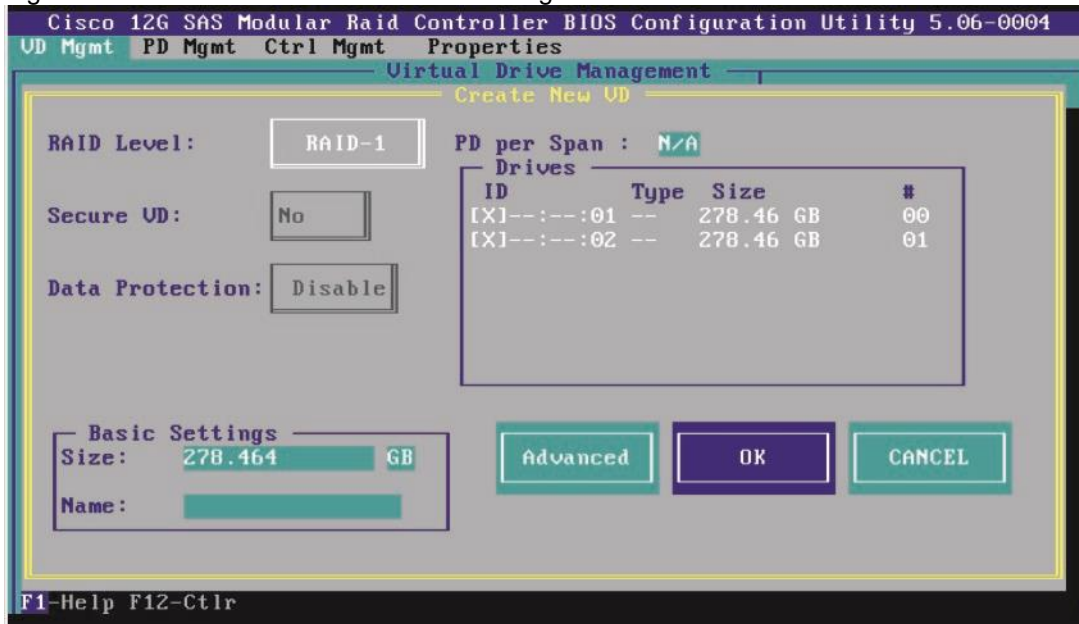
16. Check the Initialize check box.
17. Select OK. (Figure 89) and press the Enter key.

Figure 89 Warning Window



18. In the Create Virtual Drive-Advanced window, press OK to continue.
19. In the Create New VD dialog box, review the configuration and press OK to create the virtual drive. (Figure 90)

Figure 90 Review the Virtual Drive Configuration



20. Press Ctrl-N twice to reach the Ctrl Mgmt tab. (Figure 91)
21. Use the Tab key to navigate to the Boot device field. Press Enter to choose VD0 as the boot device.
22. Use the Tab key to select Apply to save the changes.

Figure 91 Save Created Configuration

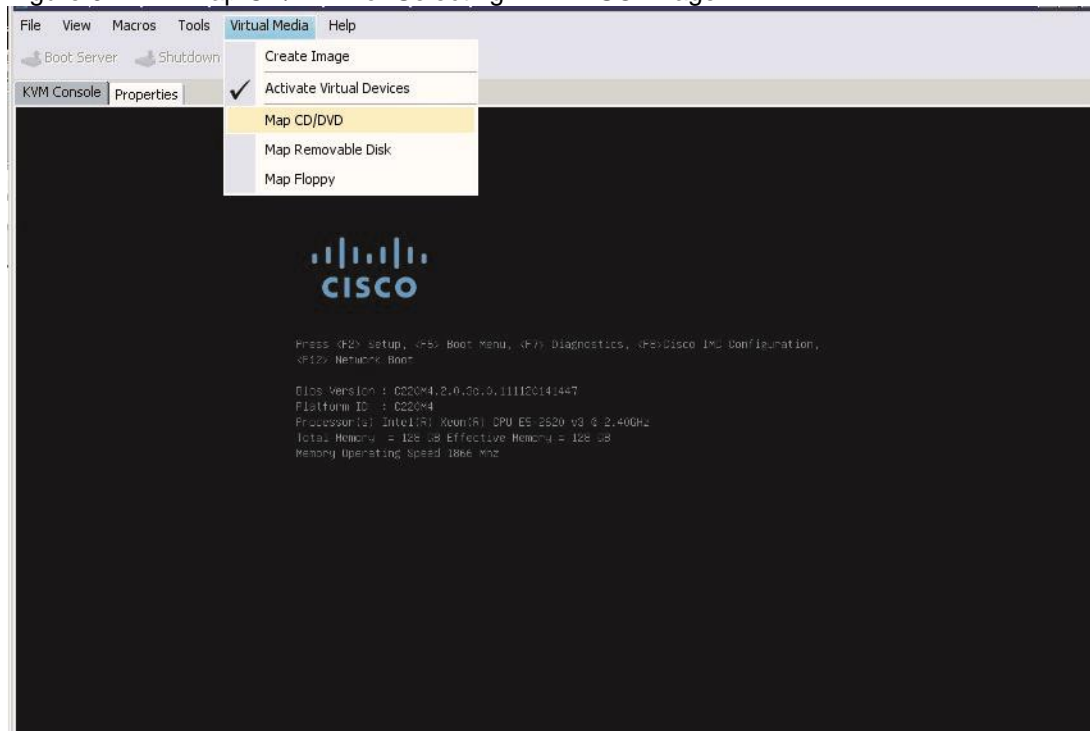


23. Press the Esc key and select OK to exit out of this utility.
24. Use the Macro menu to send a Ctrl-Alt-Del macro to reboot the server.

Installing the Operating System

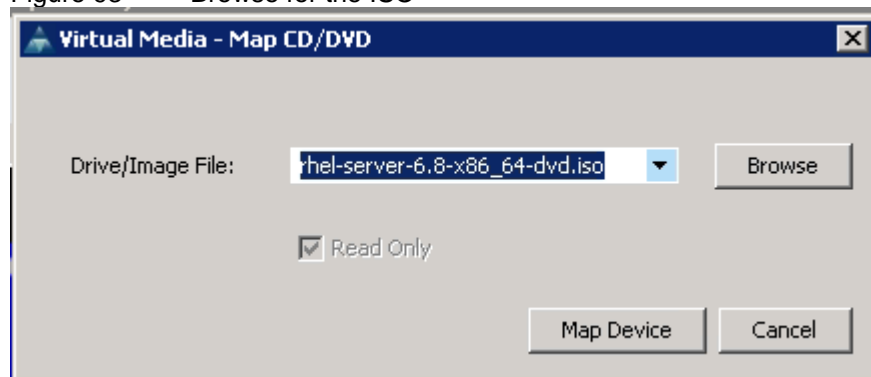
1. In the KVM console, select `Virtual Media > Activate Virtual Devices`. (Figure 92)
2. Select `Virtual Media > Map CD/DVD`.

Figure 92 Map CD/DVD for Selecting RHEL ISO Image



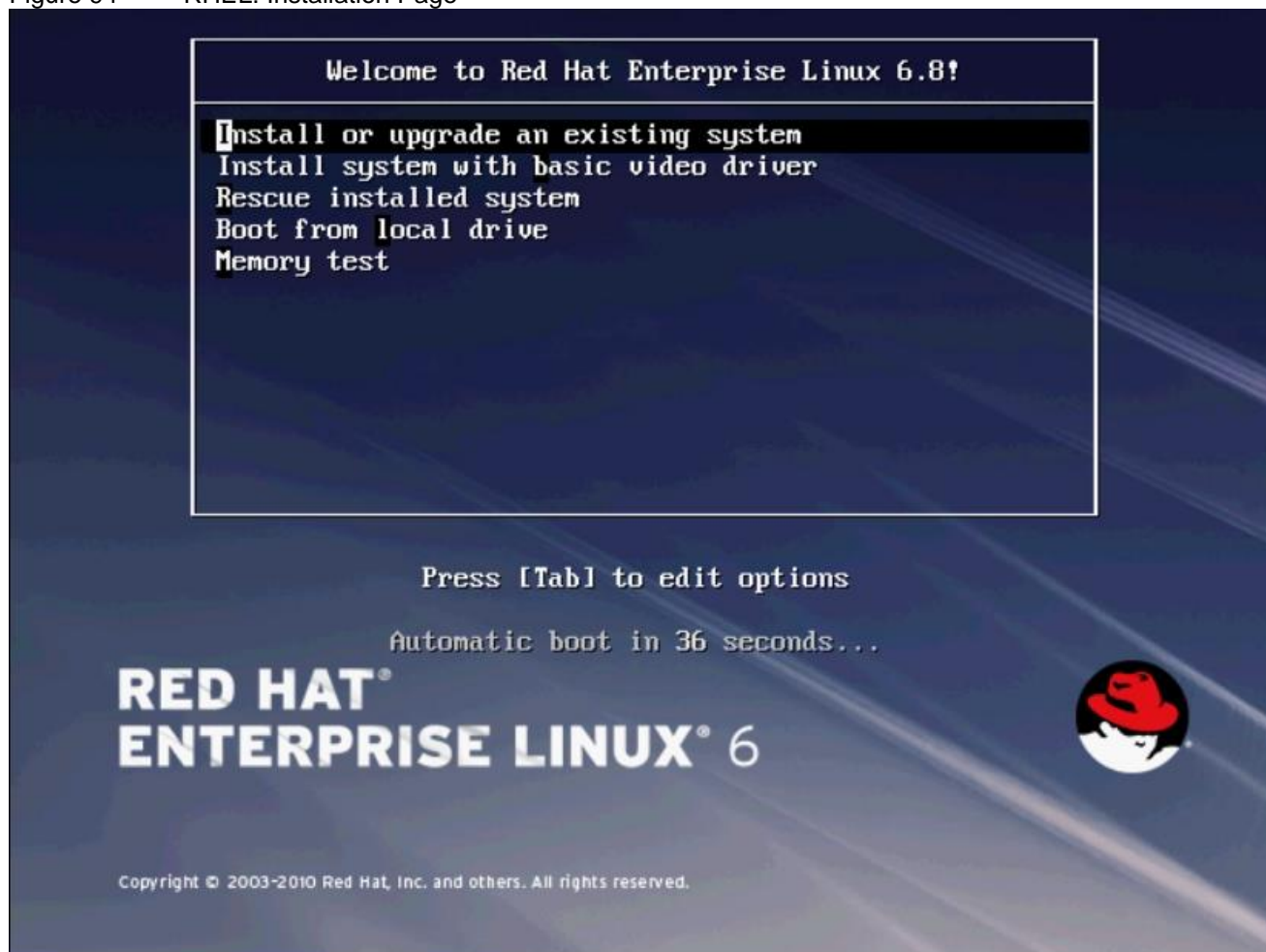
3. In the `Virtual Media - Map CD/DVD` dialog box, click on `Browse` button to choose the RHEL 6.8 Operating System ISO image file.
4. Click `Map Device`.

Figure 93 Browse for the ISO



5. Select `Macros > Static Macros > Ctrl-Alt-Del` option to reboot the server.
6. On reboot, the machine detects the presence of the Red Hat Enterprise Linux Server 6.8 install media.
7. Select `Install` or `upgrade an existing system`. (Figure 94)

Figure 94 RHEL: Installation Page



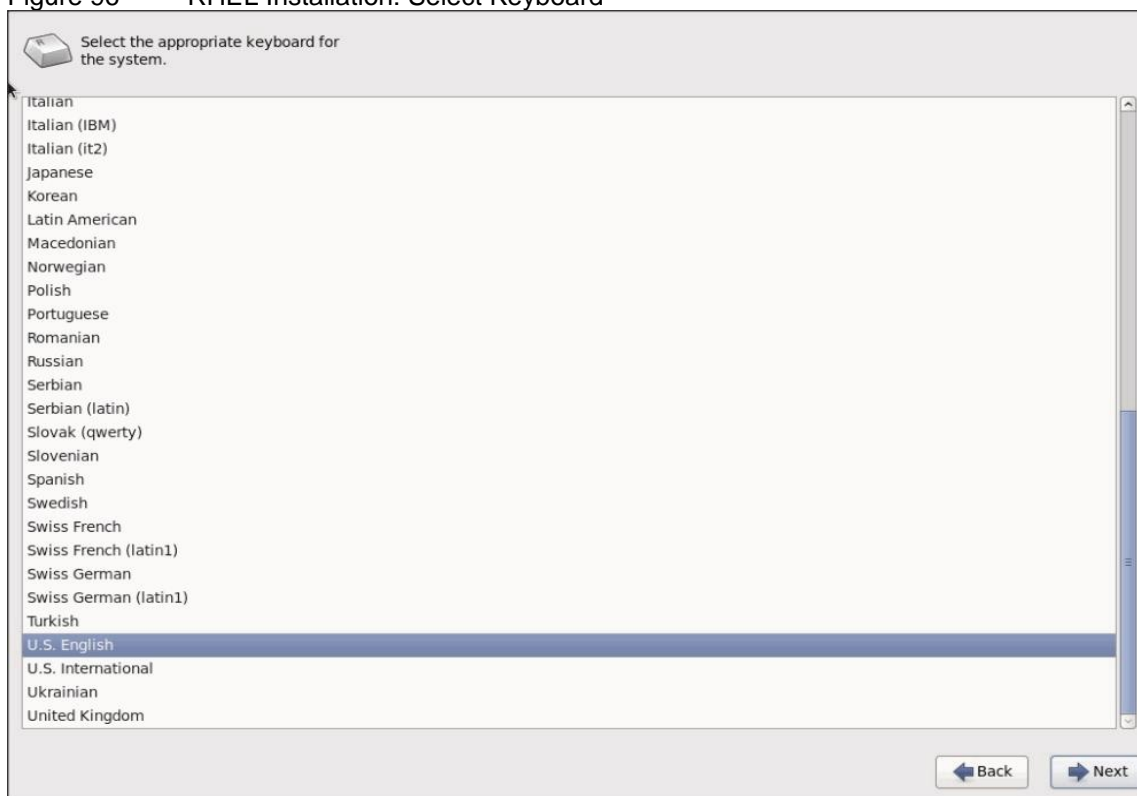
8. Skip the media test and start the installation.
9. Click `Next`.
10. Select the language of installation and click `Next`. (Figure 95)s

Figure 95 RHEL Installation: Select Language



11. Select the Keyboard for installation and click Next. (Figure 96)

Figure 96 RHEL Installation: Select Keyboard



12. Select Basic Storage Devices and click Next. (Figure 97)

Figure 97 RHEL Installation: Select Storage Devices



Figure 98 RHEL Installation: Storage Device Warning



13. Provide Hostname and click the Configure Network button on the bottom left to configure networking for the host (Figure 99).

Figure 99 RHEL Installation: Enter Host Name



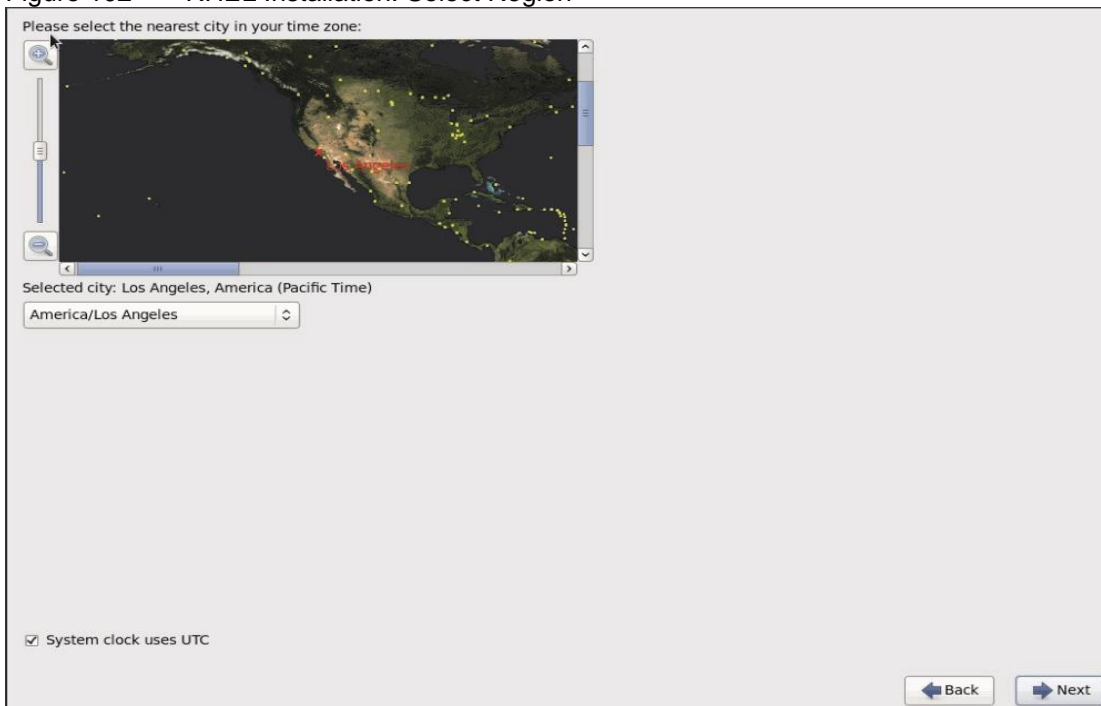
Figure 100 RHEL Installation: Configure Network Settings



Figure 101 RHEL Installation: Configure IPv4 Settings for eth 1

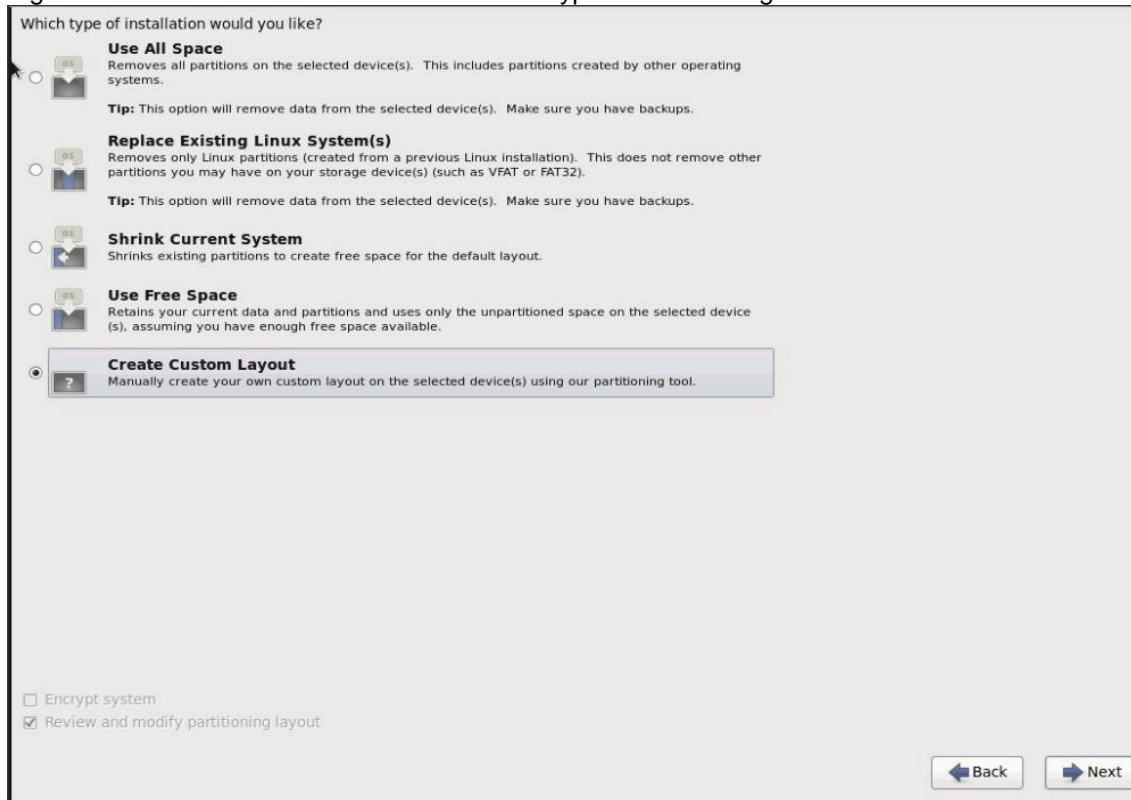


Figure 102 RHEL Installation: Select Region



14. In the type of Installation, select Create Custom Layout. (Figure 103)

Figure 103 RHEL Installation: Installation Type Selection Page

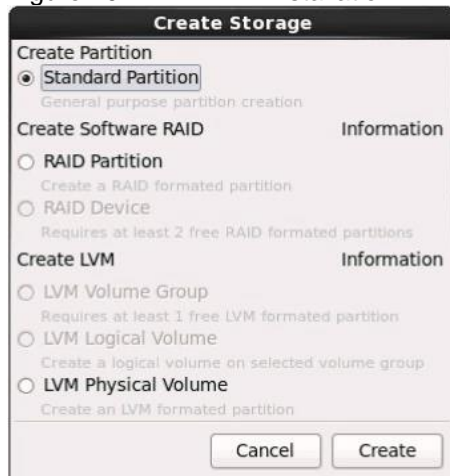


15. Click **Create**. Create three partitions as follows, to be assigned to `/boot`, `swap`, and `/` respectively.



Note: This partition layout customization is optional but highly recommended. As the Splunk software will be installed on the root partition under `/data/disk1`, it is recommended to allocate and ensure that sufficient storage is available to the `/` partition.

Figure 104 RHEL Installation: RAID Configuration



16. Set the Mount Point as `/boot`, specify the size to be 2048 MB, and select `Fixed` size under Additional Size Options. Click **OK**.

Figure 105 RHEL Installation: Add Partition Part 1

The screenshot shows the 'Add Partition' dialog box with the following settings:

- Mount Point: /boot
- File System Type: ext4
- Allowable Drives: A table with columns Drive, Size, and Model. The entry 'sdd' with size '285148 MB' and model 'Cisco UCSC-MRAID12G' is checked.
- Size (MB): 2048
- Additional Size Options:
 - Fixed size
 - Fill all space up to (MB): 1
 - Fill to maximum allowable size
- Force to be a primary partition
- Encrypt

Buttons: Cancel, OK

17. Create another standard partition with a fixed size of 512 MB for the swap partition by selecting File System Type as swap.

Figure 106 RHEL Installation: Add Partition Part 2

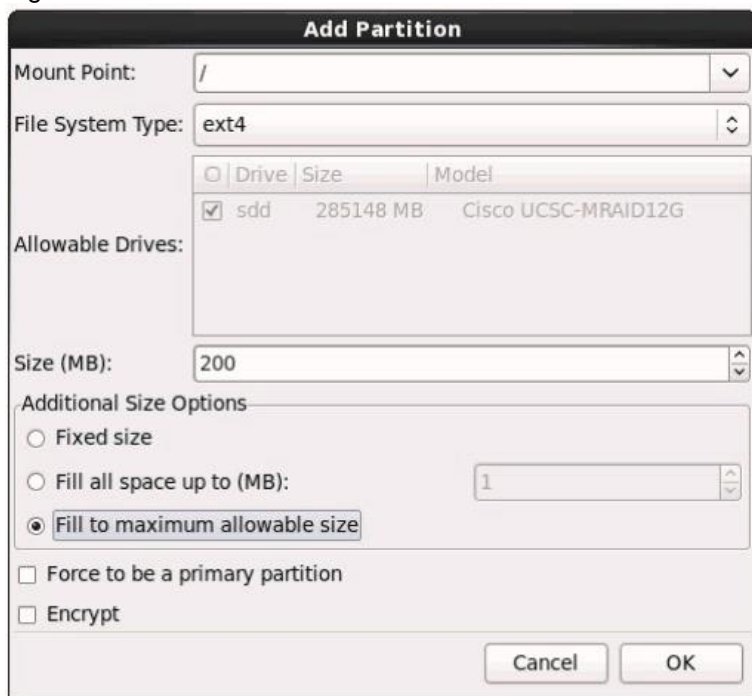
The screenshot shows the 'Add Partition' dialog box with the following settings:

- Mount Point: <Not Applicable>
- File System Type: swap
- Allowable Drives: A table with columns Drive, Size, and Model. The entry 'sdd' with size '285148 MB' and model 'Cisco UCSC-MRAID12G' is checked.
- Size (MB): 512
- Additional Size Options:
 - Fixed size
 - Fill all space up to (MB): 1
 - Fill to maximum allowable size
- Force to be a primary partition
- Encrypt

Buttons: Cancel, OK

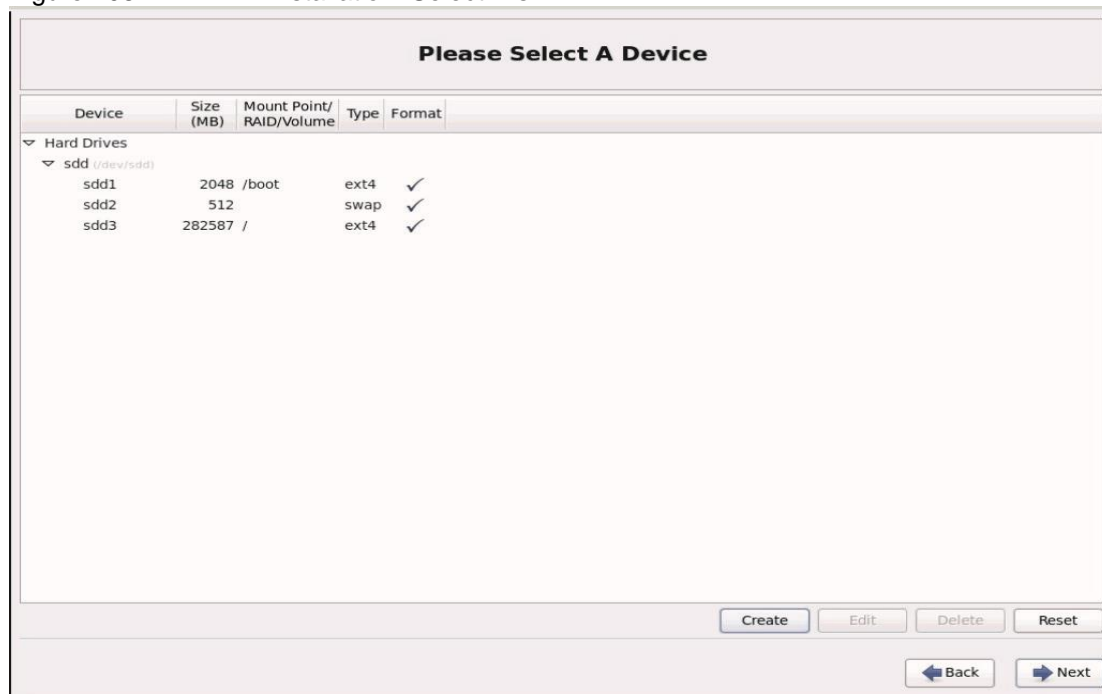
18. Create the third standard partition with Mount Point set to / and select the Fill to maximum allowable size in the Additional Size Options.

Figure 107 RHEL Installation: Add Partition Part 3



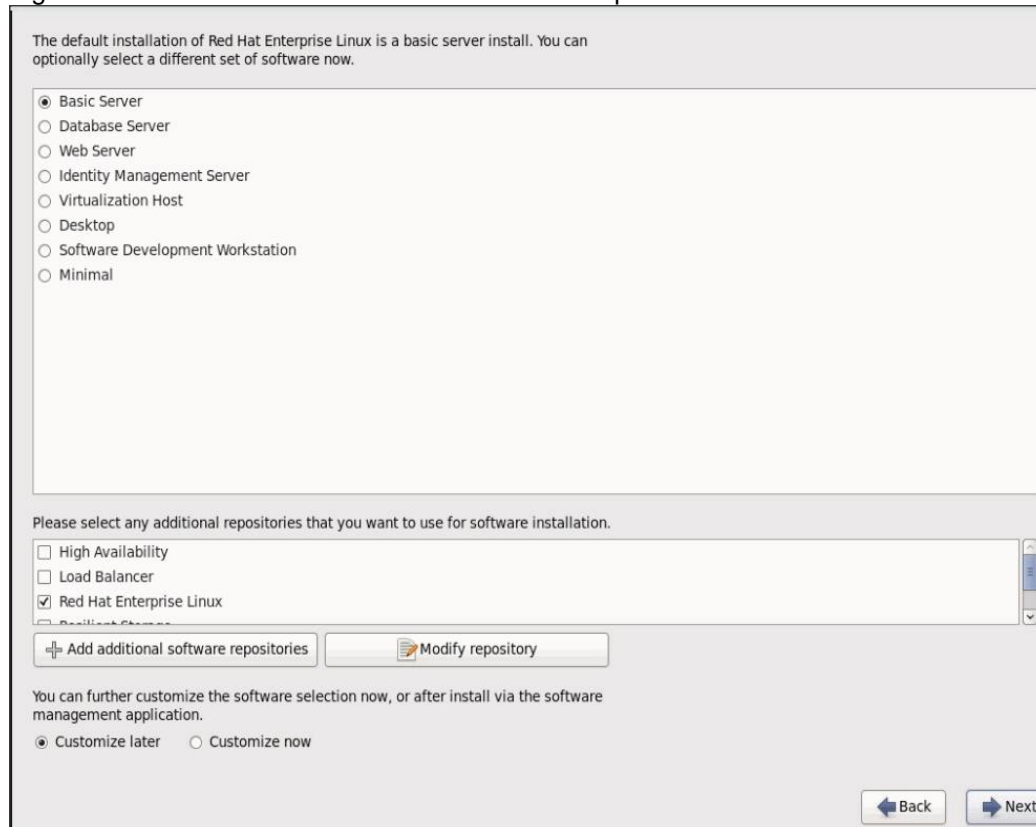
19. Click **Next** to continue.

Figure 108 RHEL Installation: Select Disk



20. Continue with the RHEL Installation as shown below.

Figure 109 RHEL Installation: Customization Option



21. Once the installation is complete, reboot the system.
22. Repeat steps 1 to 21 to install Red Hat Enterprise Linux 6.8 on the four other Cisco C220 M4 servers serving as search heads and admin nodes. Assign the host names as follows: admin2 for the other Cisco C220 M4 admin server, and sh1, sh2, sh3 for the three Cisco C220 M4 search head servers. Assign the respective IP addresses to these servers by referring to Table 6 .

Installing Red Hat Enterprise Linux 6.8 using Software RAID on C240 M4 Systems

To install Red Hat Enterprise Linux 6.8 using Software RAID (OS based Mirroring) on Cisco UCS C240 M4 servers complete the following steps:



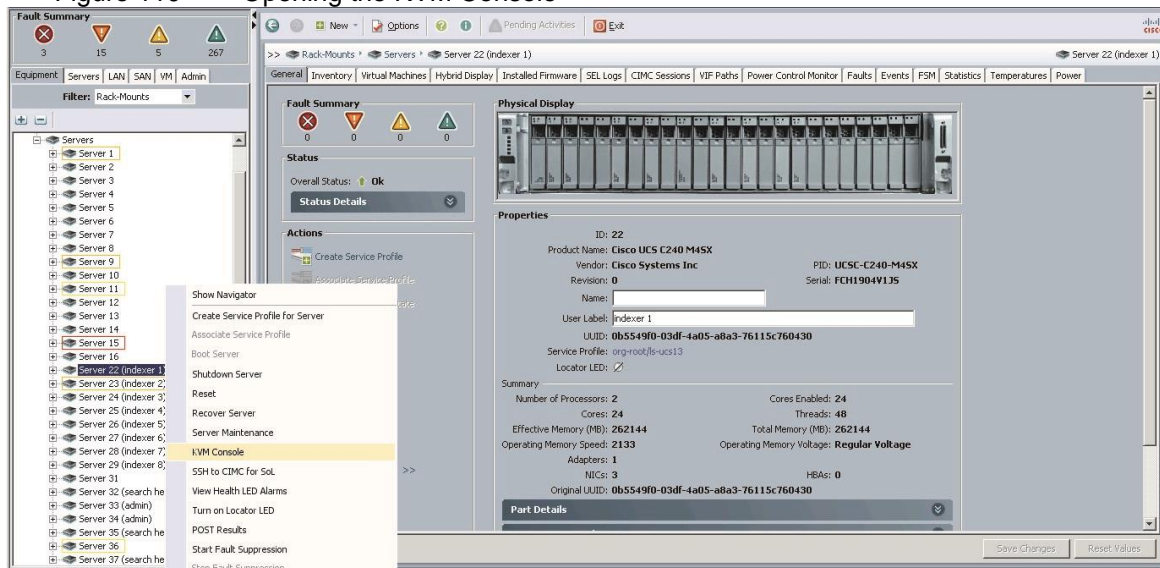
Note: The installation procedure described in this deployment guide uses KVM Console and virtual media from Cisco UCS Manager.



Note: RHEL 6.8 DVD/ISO is required for the installation

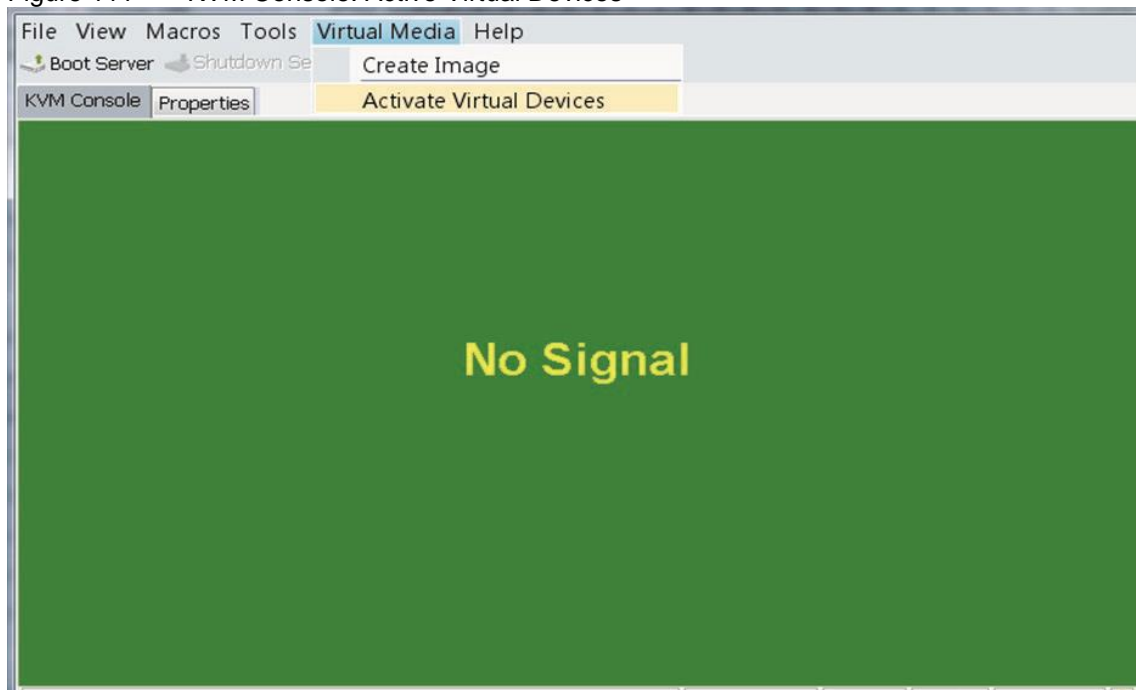
1. Log in to the Cisco UCS 6296 Fabric Interconnect and launch the Cisco UCS Manager application.
2. Select the `Equipment` tab.
3. In the navigation pane, expand `Rack-Mounts` and then `Servers`.
4. Right click on the server and select `KVM Console`. (Figure 110)

Figure 110 Opening the KVM Console



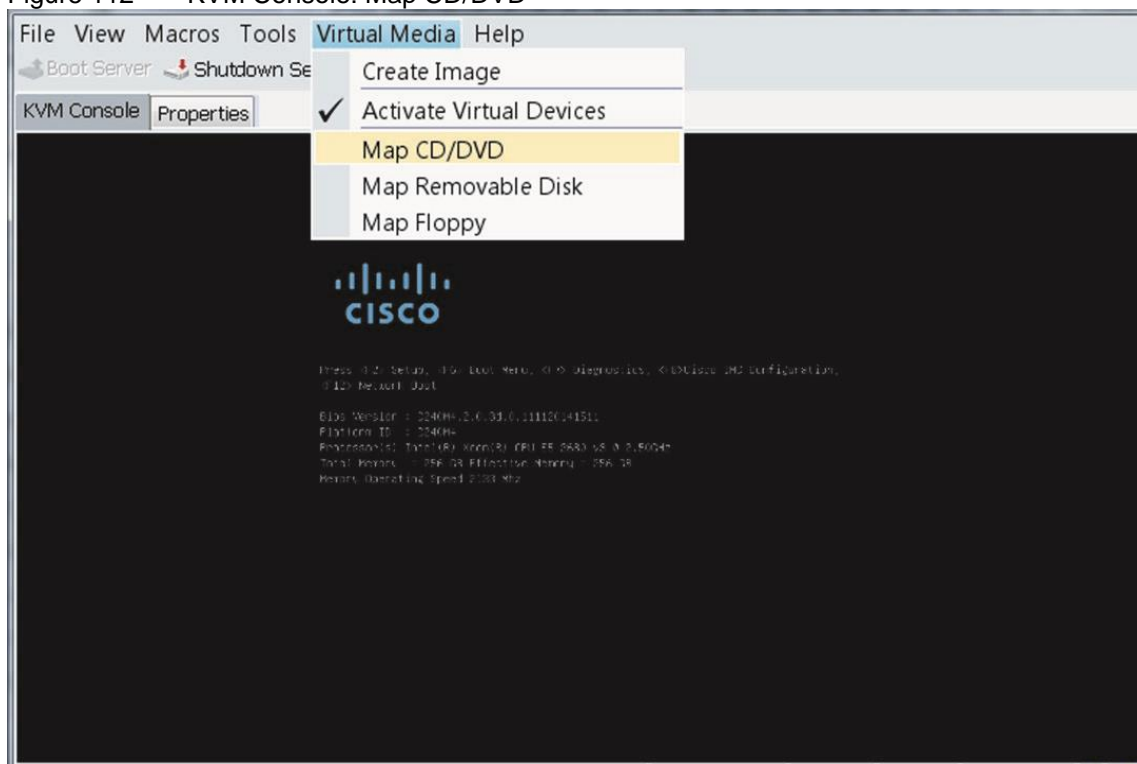
5. In the KVM window, select the Virtual Media tab. (Figure 111)
6. Click Activate Virtual Devices found in Virtual Media tab.

Figure 111 KVM Console: Active Virtual Devices



7. In the KVM window, select the Virtual Media tab and click Map CD/DVD. (Figure 112)

Figure 112 KVM Console: Map CD/DVD



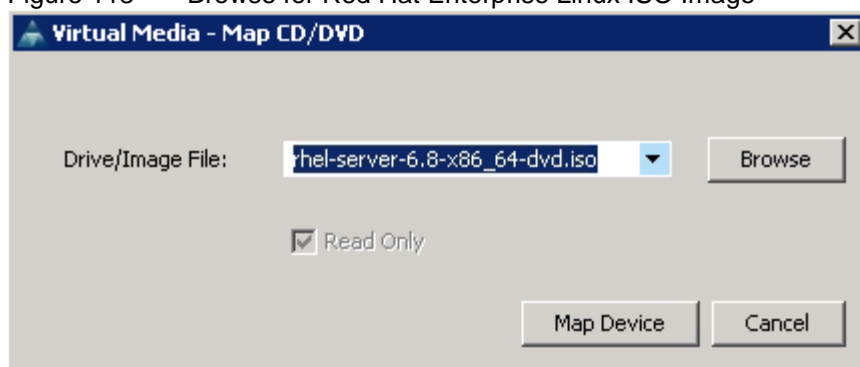
8. Browse to the Red Hat Enterprise Linux Server 6.8 installer ISO image file. (Figure 113)



The Red Hat Enterprise Linux 6.8 DVD is assumed to be on the client machine.

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

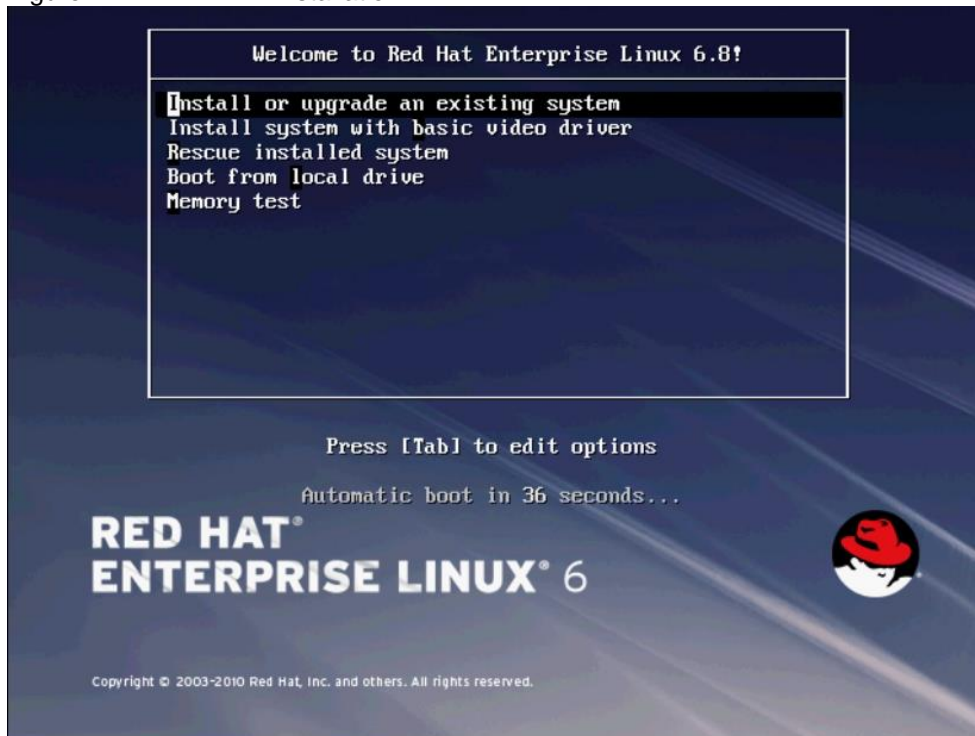
Figure 113 Browse for Red Hat Enterprise Linux ISO Image



10. In the KVM window, select the KVM tab to monitor during boot.
11. In the KVM window, select **Macros > Static Macros > Ctrl-Alt-Del** 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 6.8 install media.

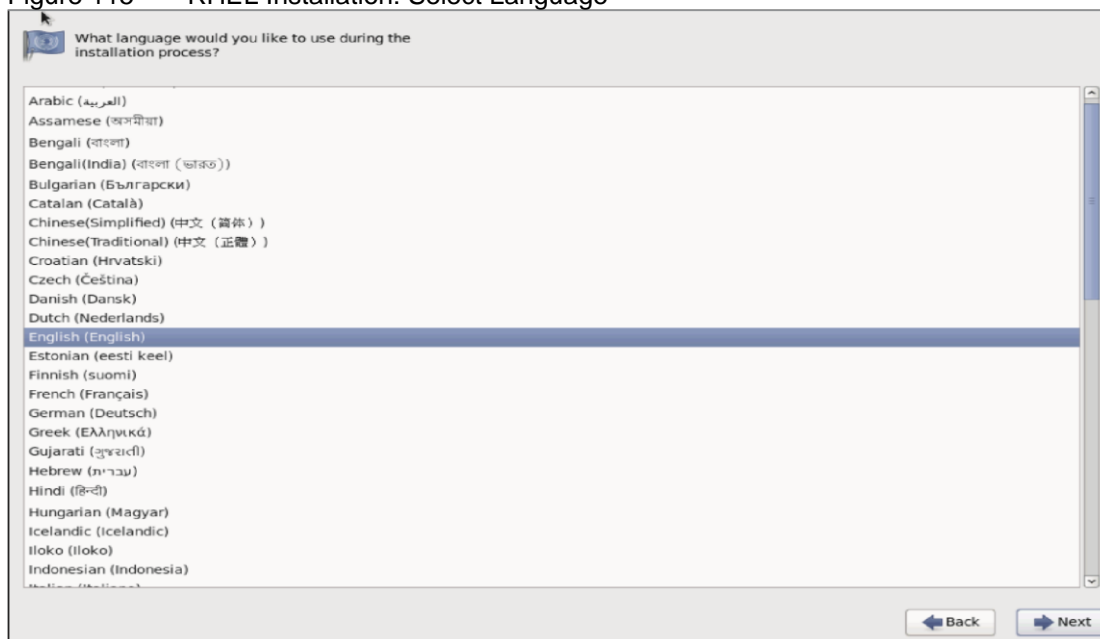
15. Select `Install or upgrade an existing system`. (Figure 114)

Figure 114 RHEL Installation



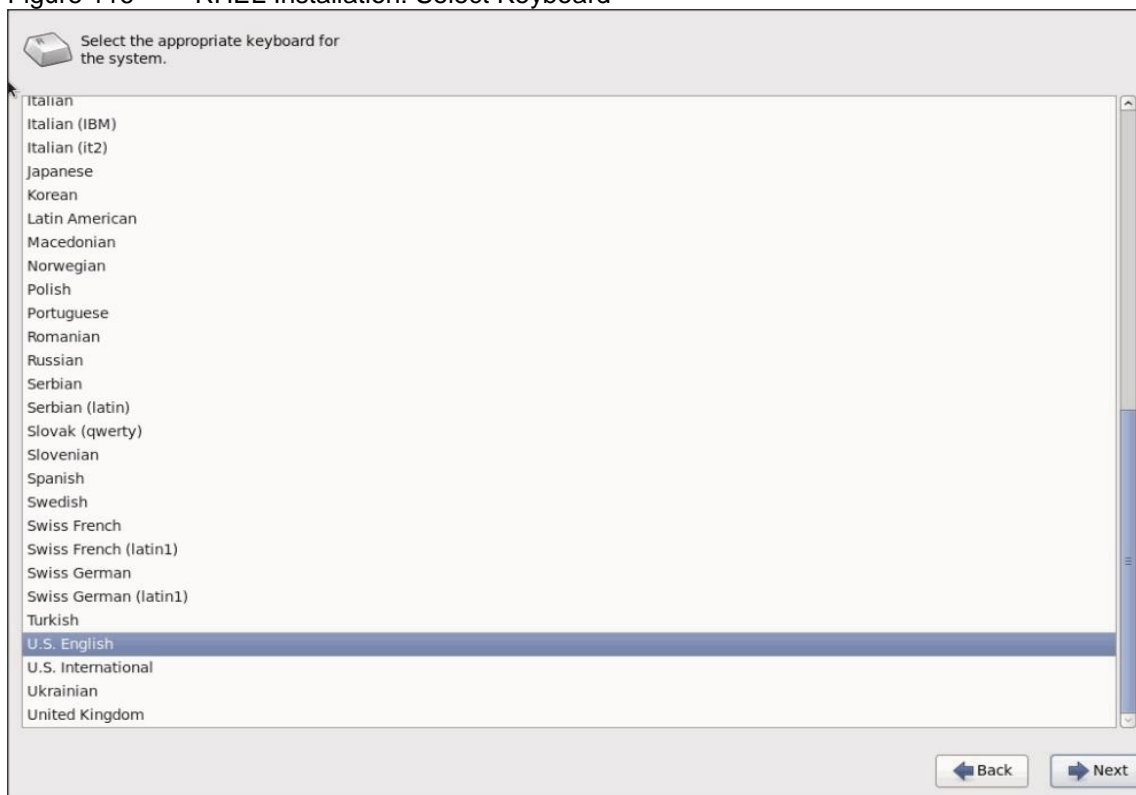
16. Skip the Media test and start the installation.
17. Click `Next`.
18. Select language of installation and click `Next`. (Figure 115)

Figure 115 RHEL Installation: Select Language



19. Select keyboard for the installation. (Figure 116)

Figure 116 RHEL Installation: Select Keyboard



20. Select Basic Storage Devices and click Next. (Figure 117)

Figure 117 RHEL Installation: Select Basic Storage Devices



Figure 118 RHEL Installation: Storage Device Warning



21. Provide Hostname. (Figure 119)

22. Click the `Configure Network` button on the bottom left to configure networking for the host (See Table 7).

Figure 119 RHEL Installation: Enter Host Name

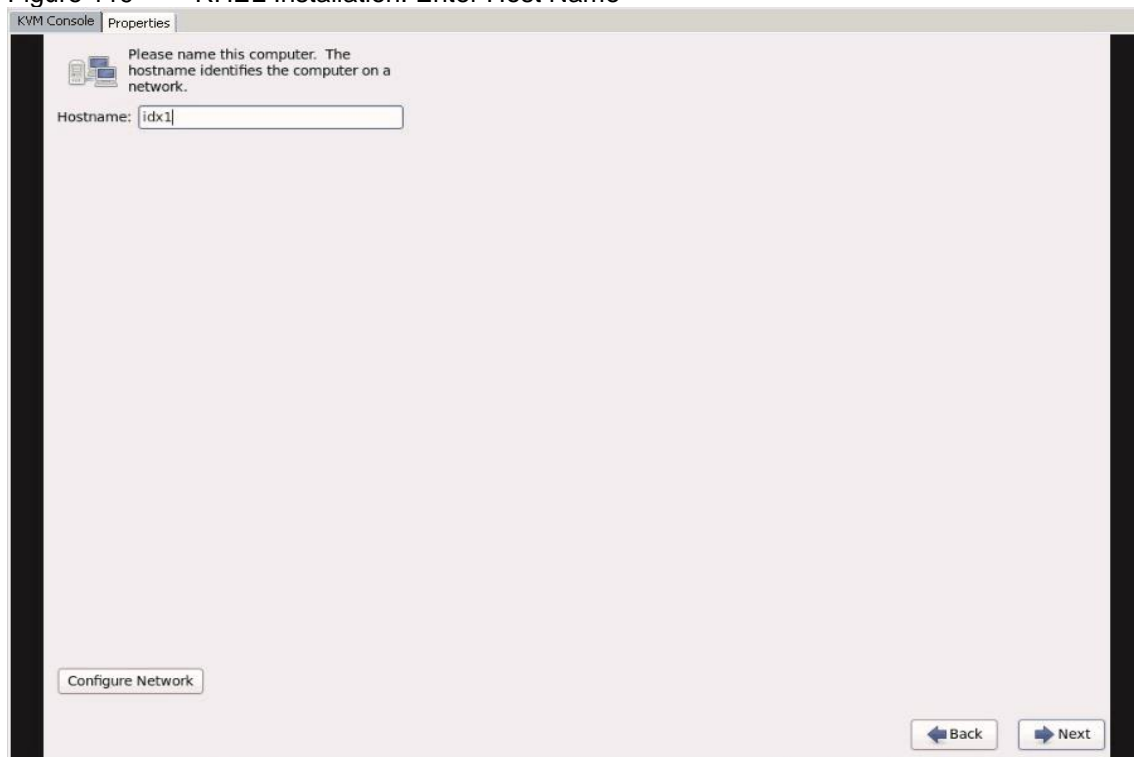


Figure 120 RHEL Installation: Configure Network Settings



Figure 121 RHEL Installation: Configure IPv4 Settings for eth 1



Figure 122 RHEL Installation: Select Region

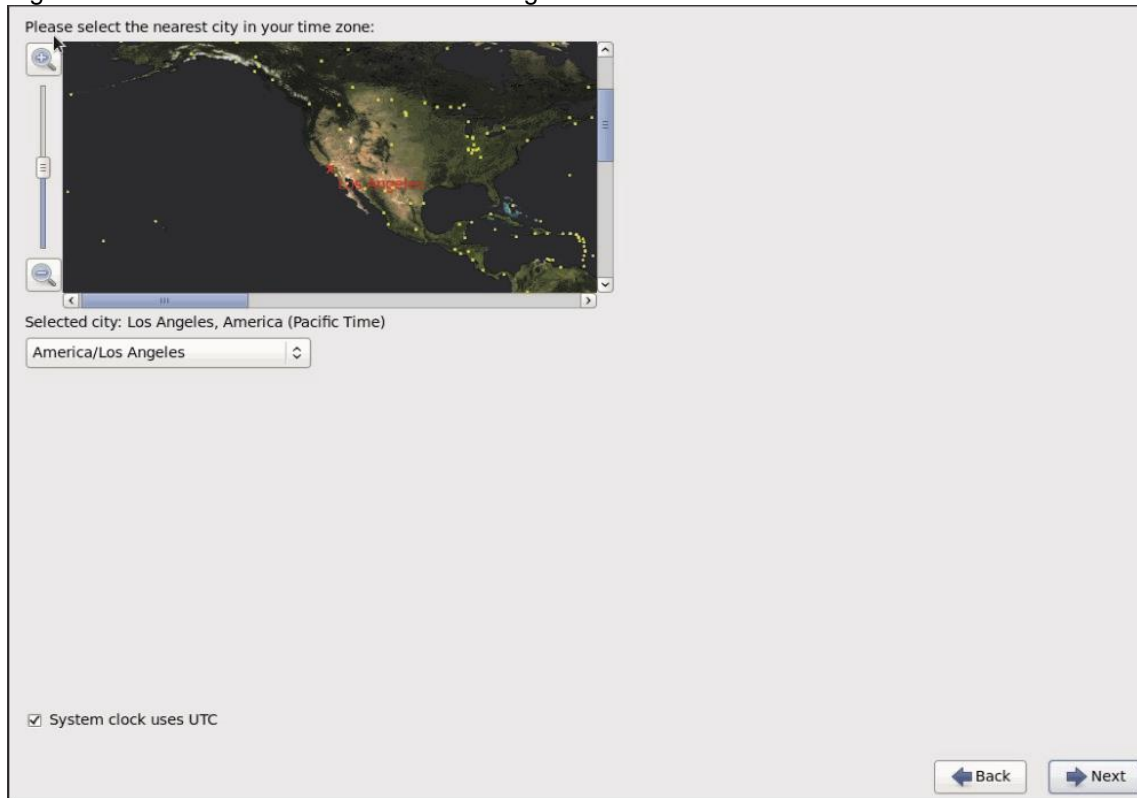
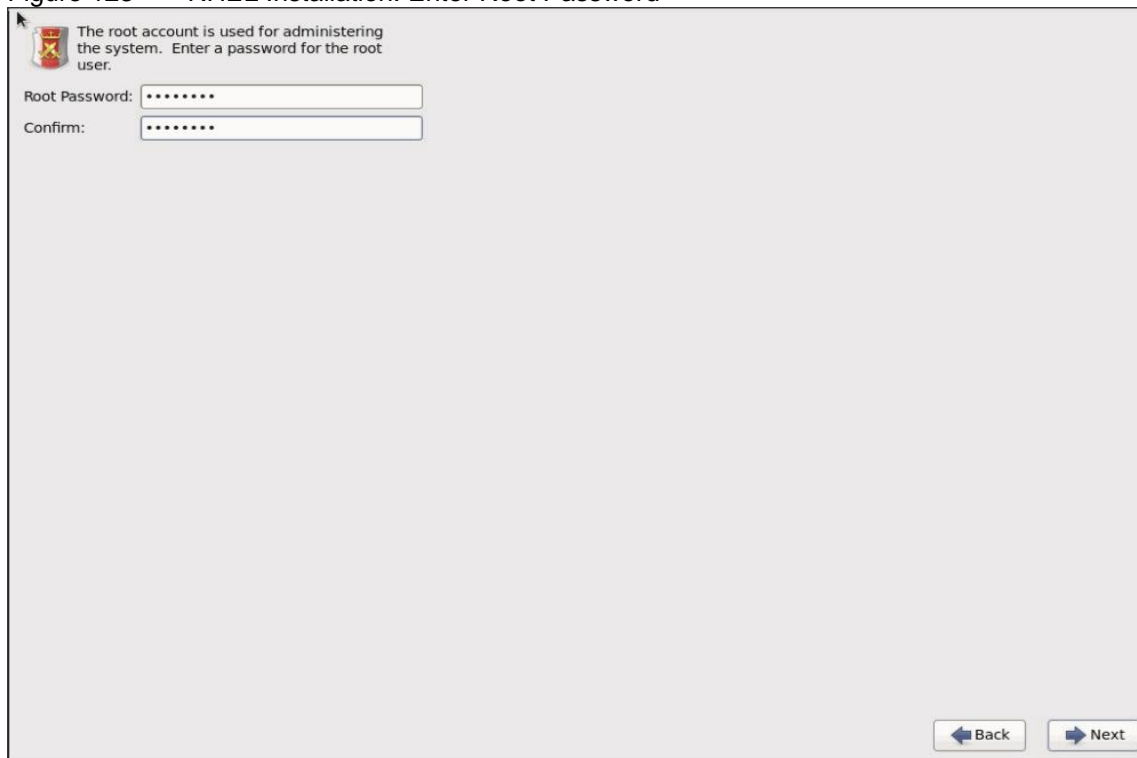
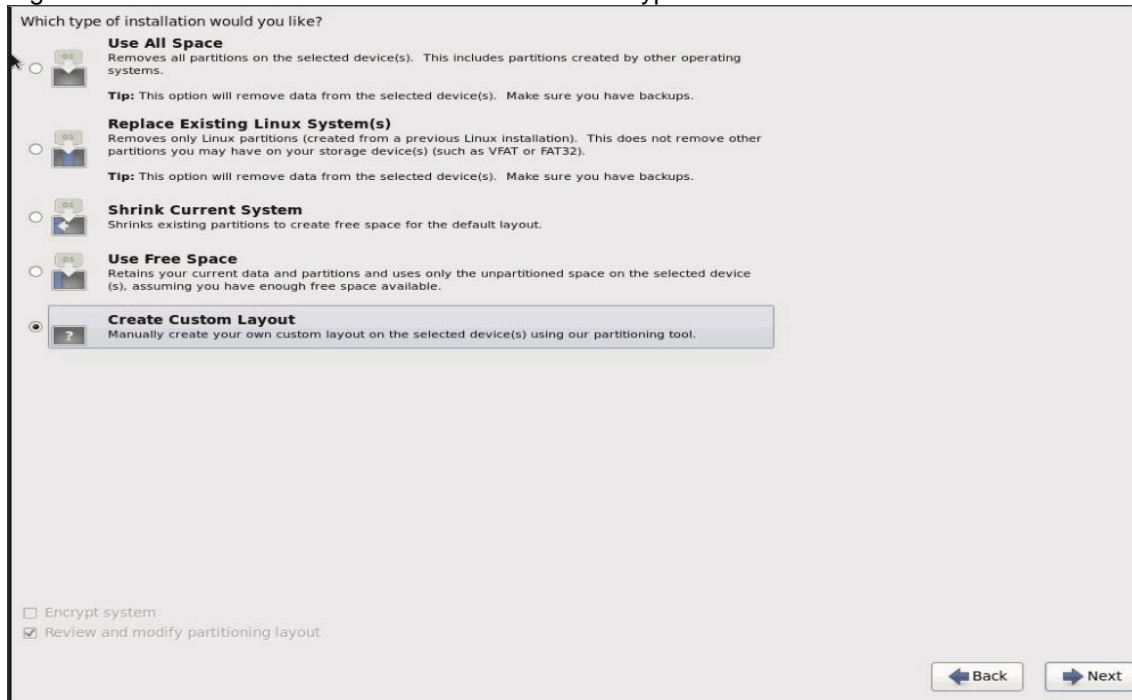


Figure 123 RHEL Installation: Enter Root Password



23. Choose Create Custom Layout for Installation type.

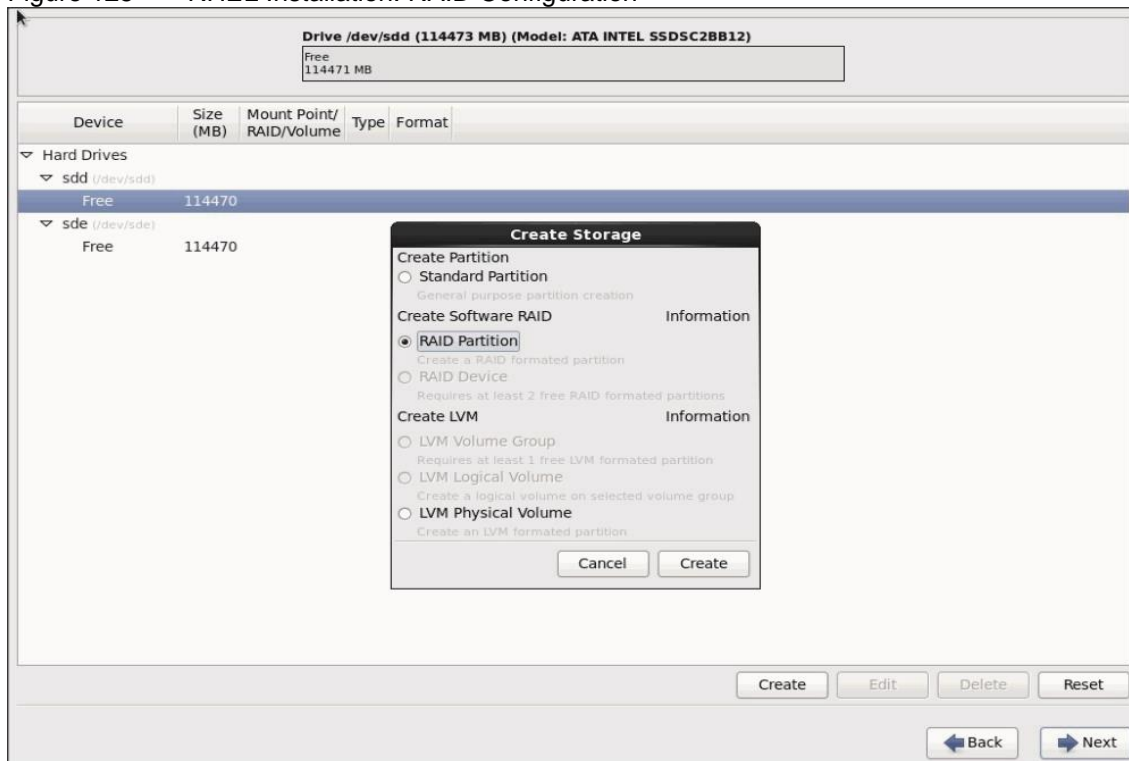
Figure 124 RHEL Installation: Select Installation Type



To create two software RAID 1 partitions for boot and / (root) partitions, complete the following steps:

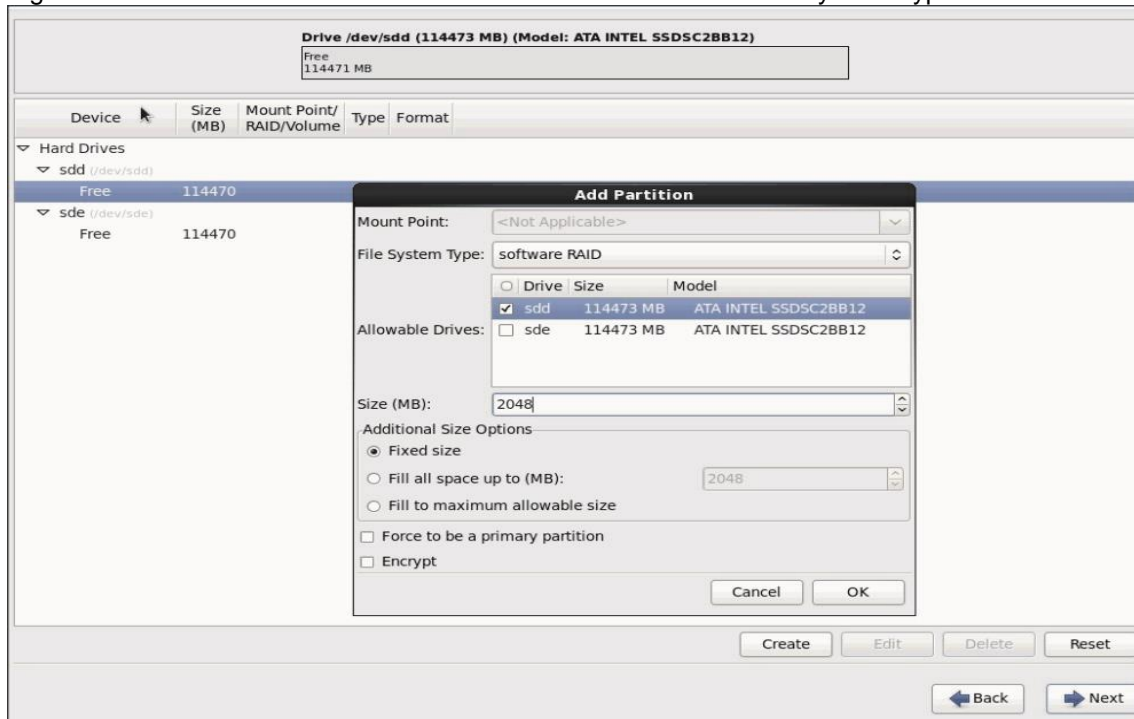
24. Choose free volume and click on **Create** and choose **RAID Partition**. (Figure 125)

Figure 125 RHEL Installation: RAID Configuration



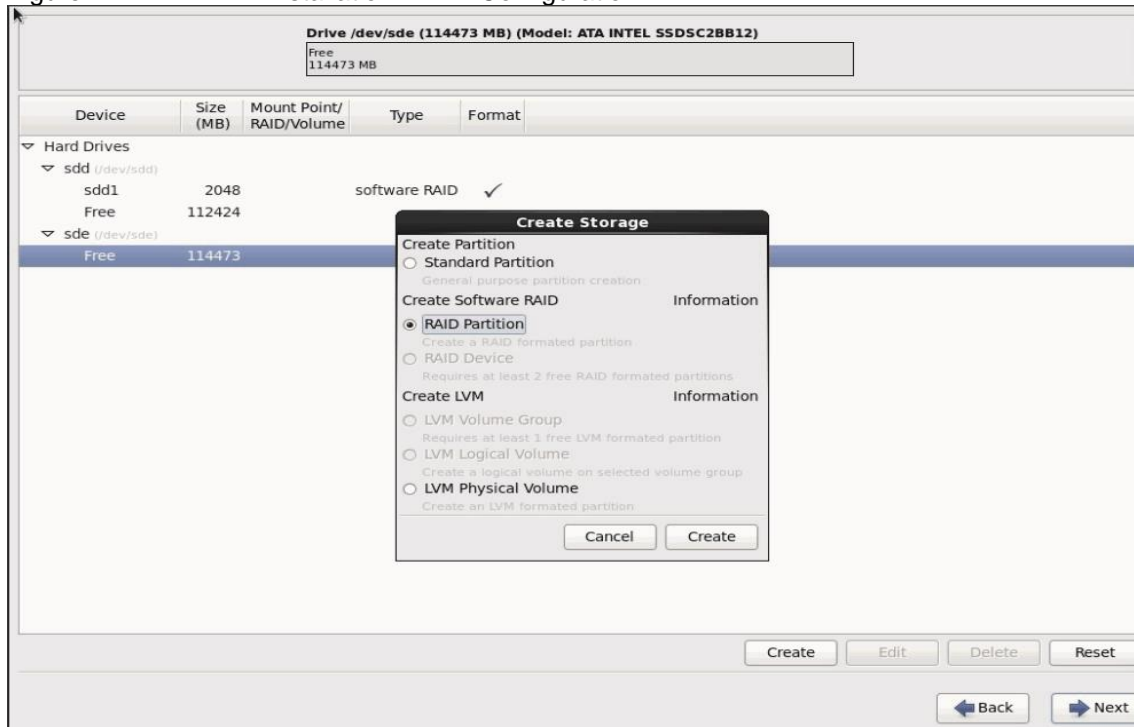
25. Choose **Software RAID** for File System Type and set Size for Boot volume.

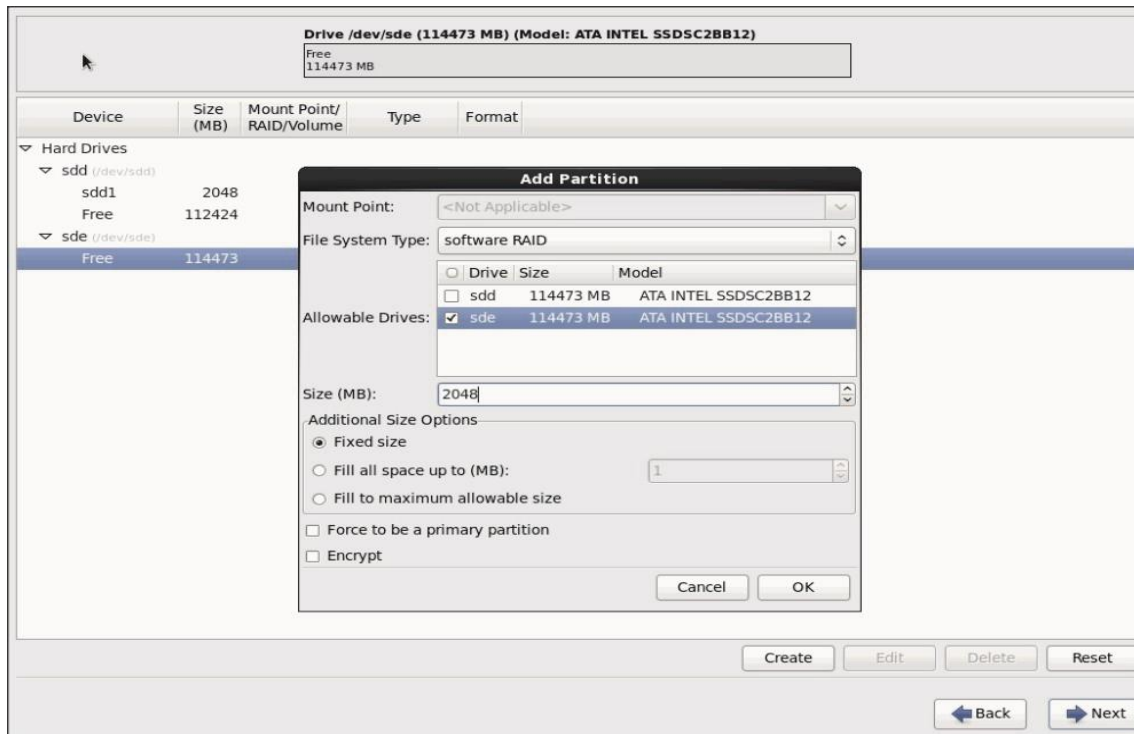
Figure 126 RHEL Installation: Set Boot Volume and Select File System Type



26. Do the same for the other free volume.

Figure 127 RHEL Installation: RAID Configuration





27. Now similarly create RAID partitions for root (/) partition on both the devices and use rest of the available space by selecting **Fill to maximum allowable size**.

Figure 128 RHEL Installation: RAID Configuration Add Partition

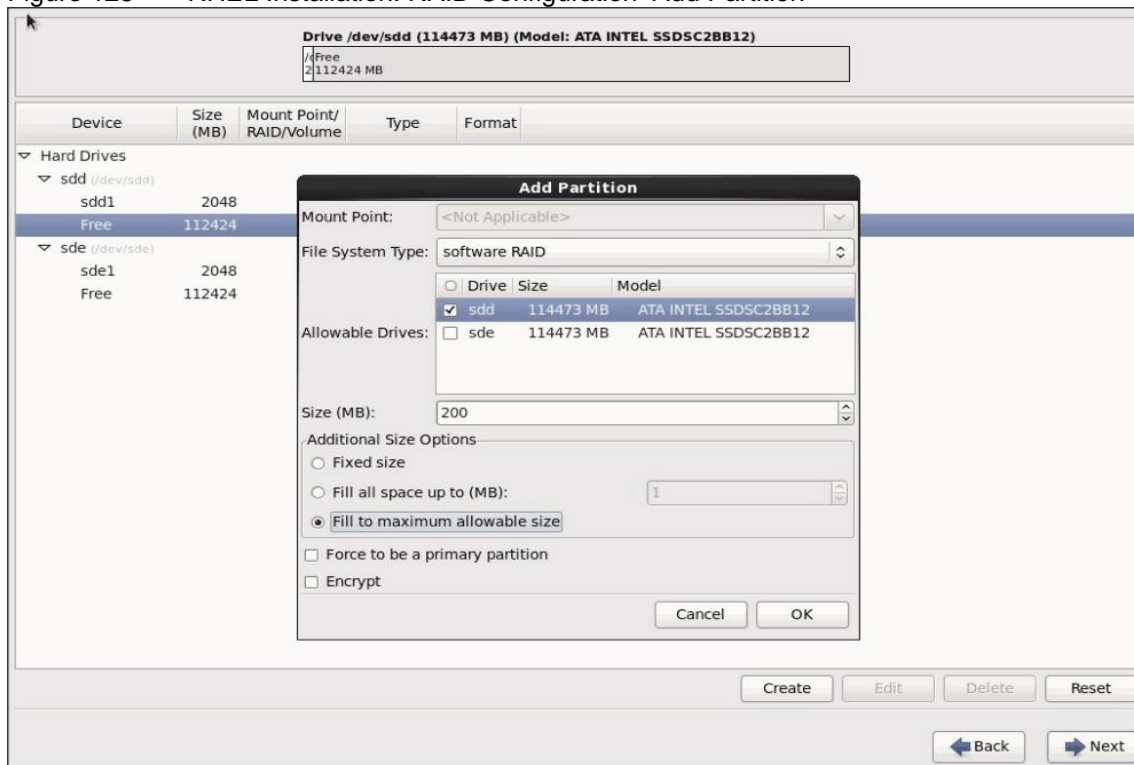


Figure 129 RHEL Installation: RAID Configuration: Create Storage

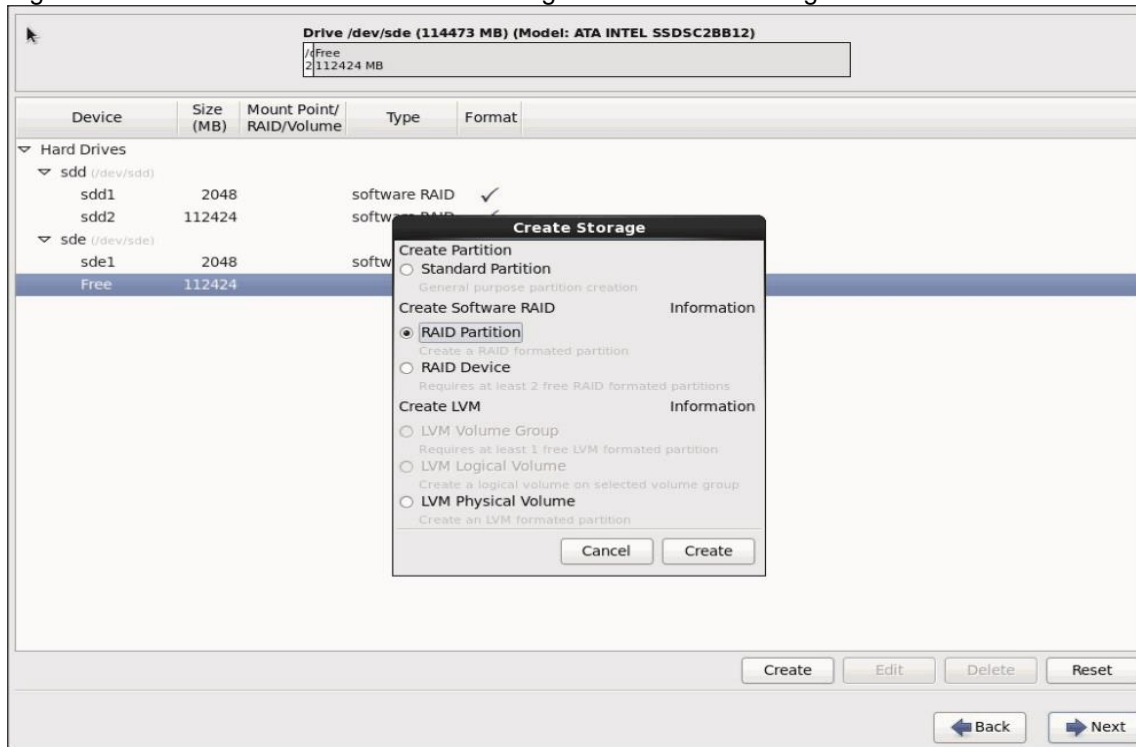
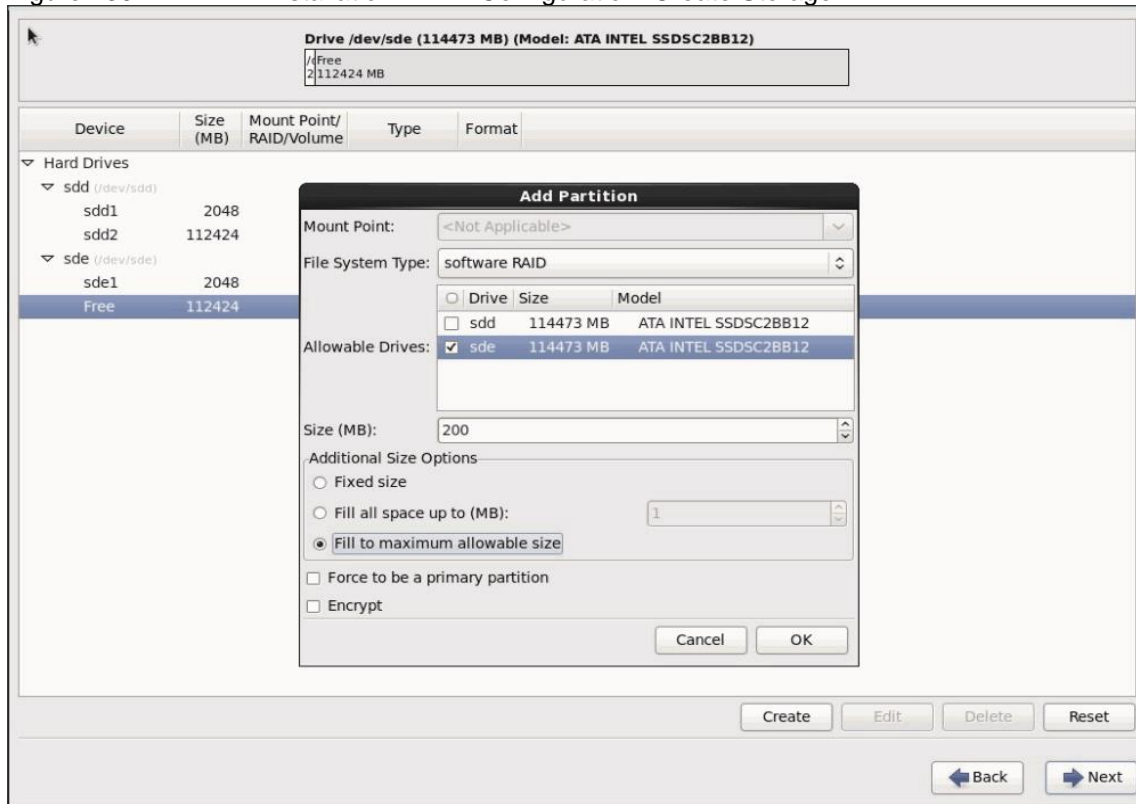


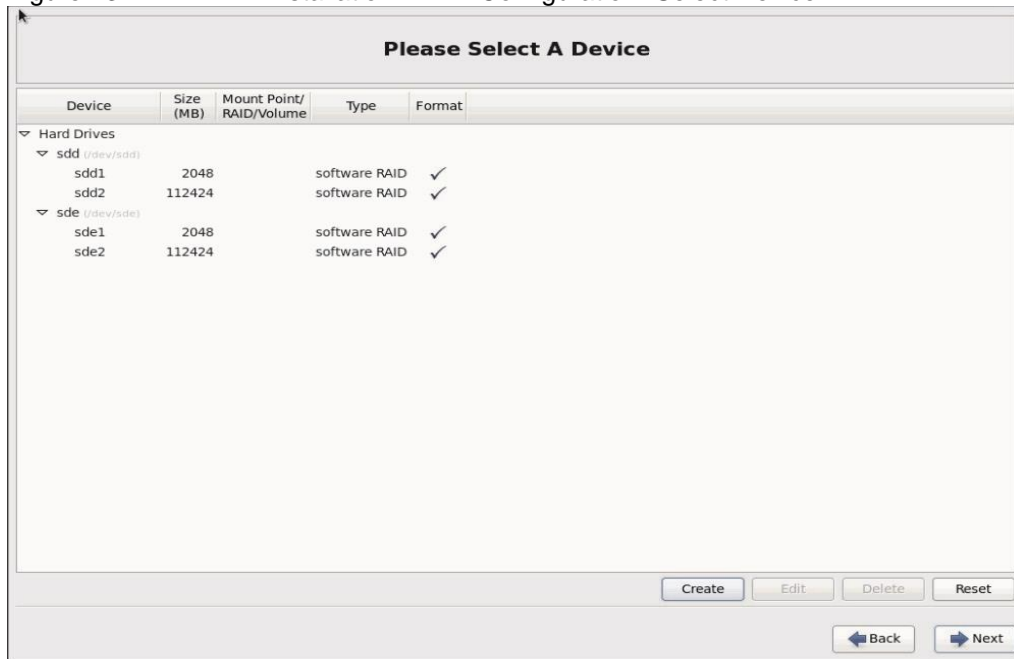
Figure 130 RHEL Installation: RAID Configuration: Create Storage



The steps above created 2 boot and 2 root (/) partitions. To create RAID1 Devices complete the following steps.

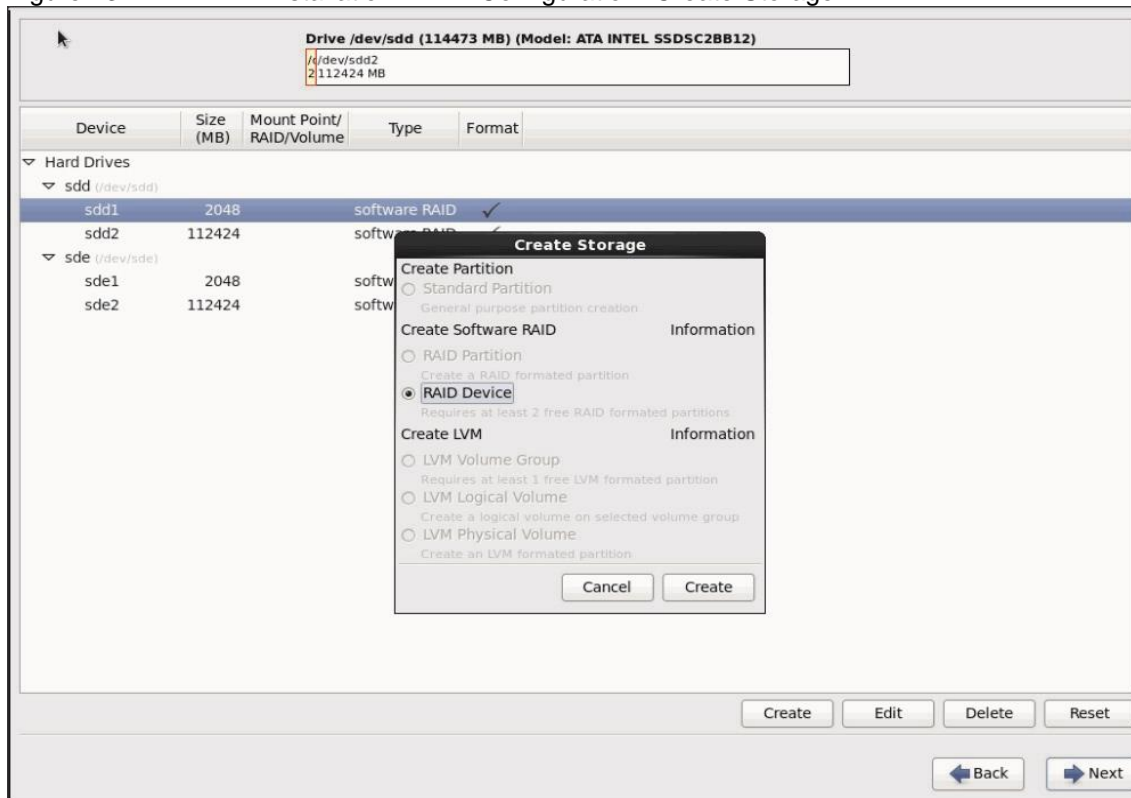
28. Choose one of the boot partitions and click on Create (Figure 131).

Figure 131 RHEL Installation: RAID Configuration- Select Device



29. Choose RAID Device. (Figure 132)

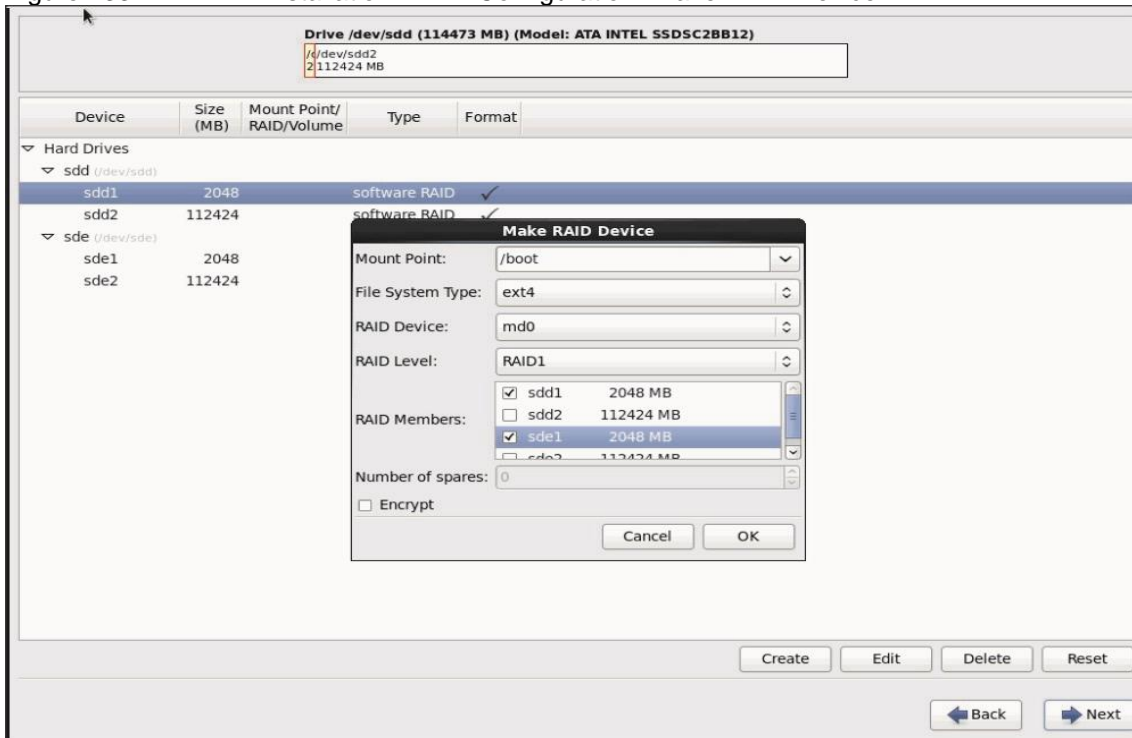
Figure 132 RHEL Installation: RAID Configuration- Create Storage



30. Choose this as the /boot (boot device).

- In RAID members, choose all the boot partitions created above to create a software RAID 1 for boot. (Figure 133)

Figure 133 RHEL Installation: RAID Configuration- Make RAID Device



- Repeat for / partitions created above choosing both members with mount point as "/".

Figure 134 RHEL Installation: RAID Configuration- Create Storage

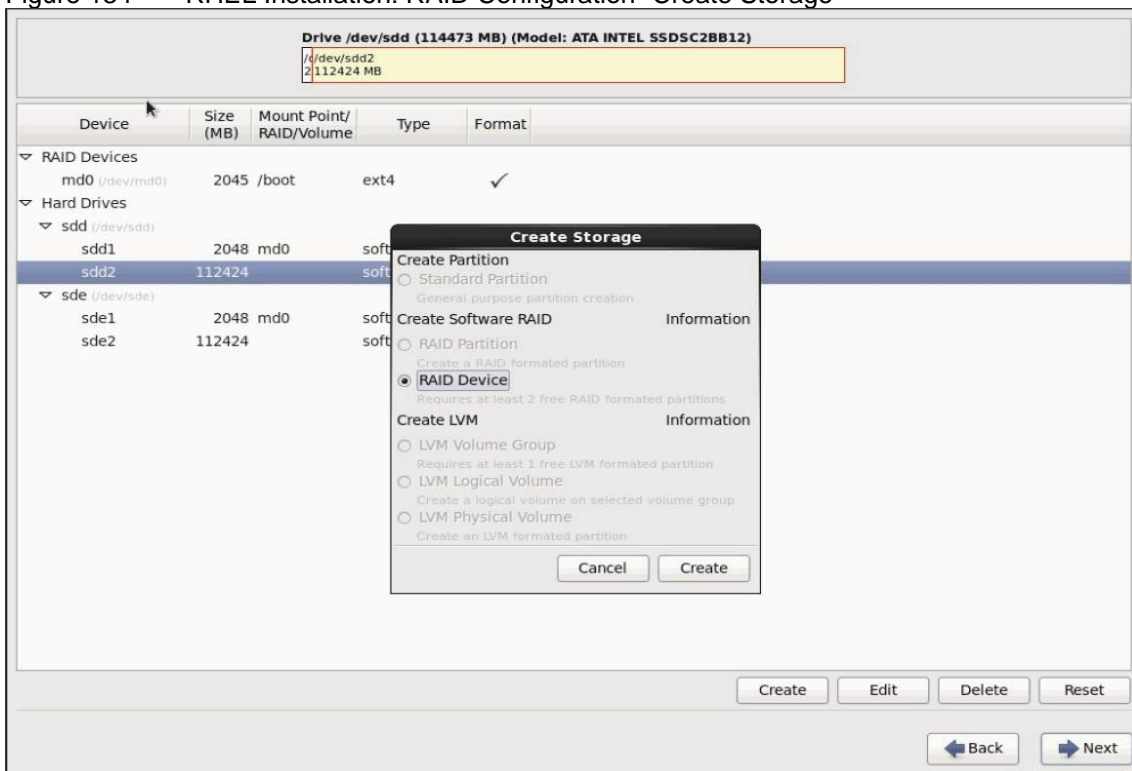


Figure 135 RHEL Installation: RAID Configuration- Make RAID Device

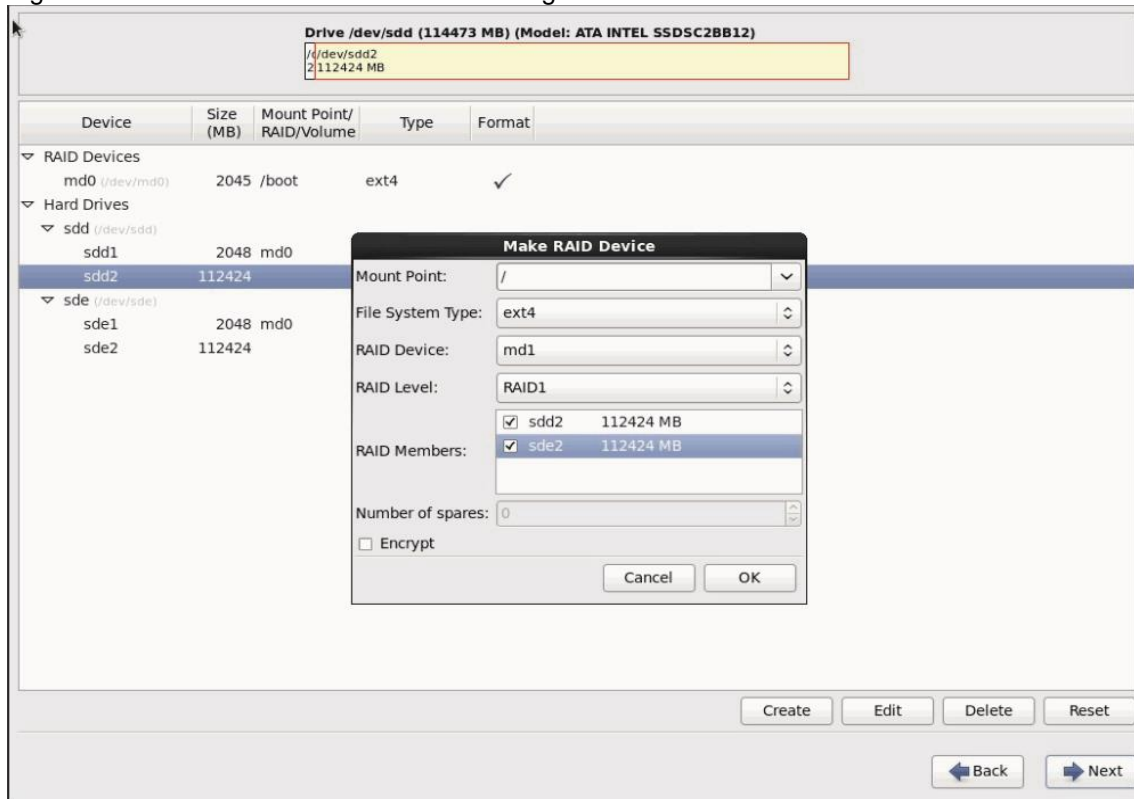
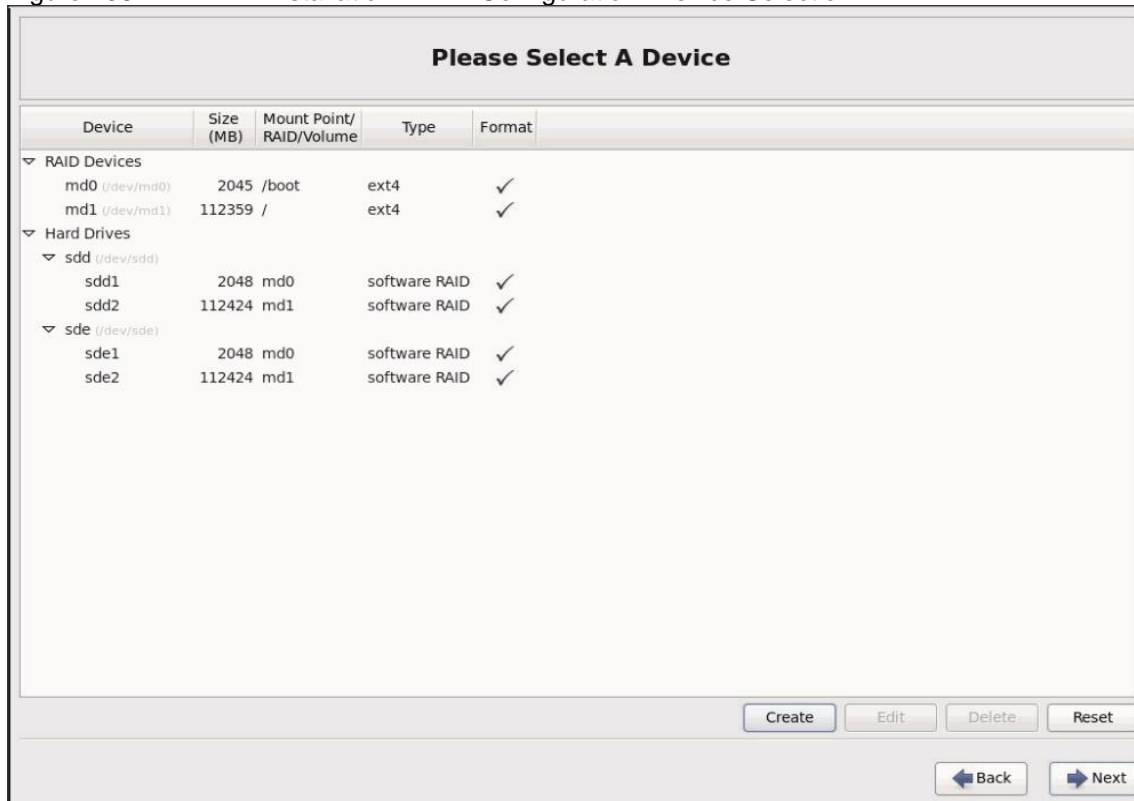


Figure 136 RHEL Installation: RAID Configuration- Device Selection



33. Click Next.

Figure 137 RHEL Installation: RAID Configuration- Partitioning Warnings

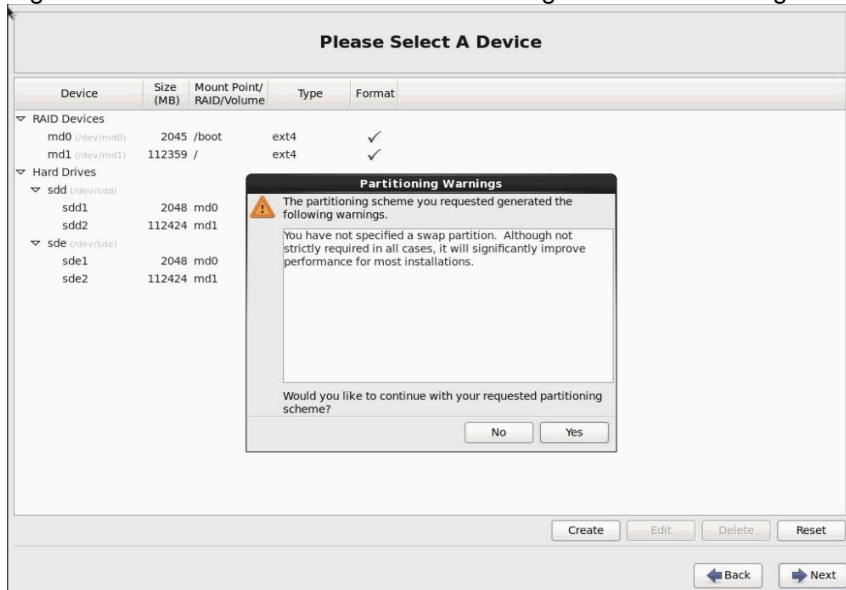
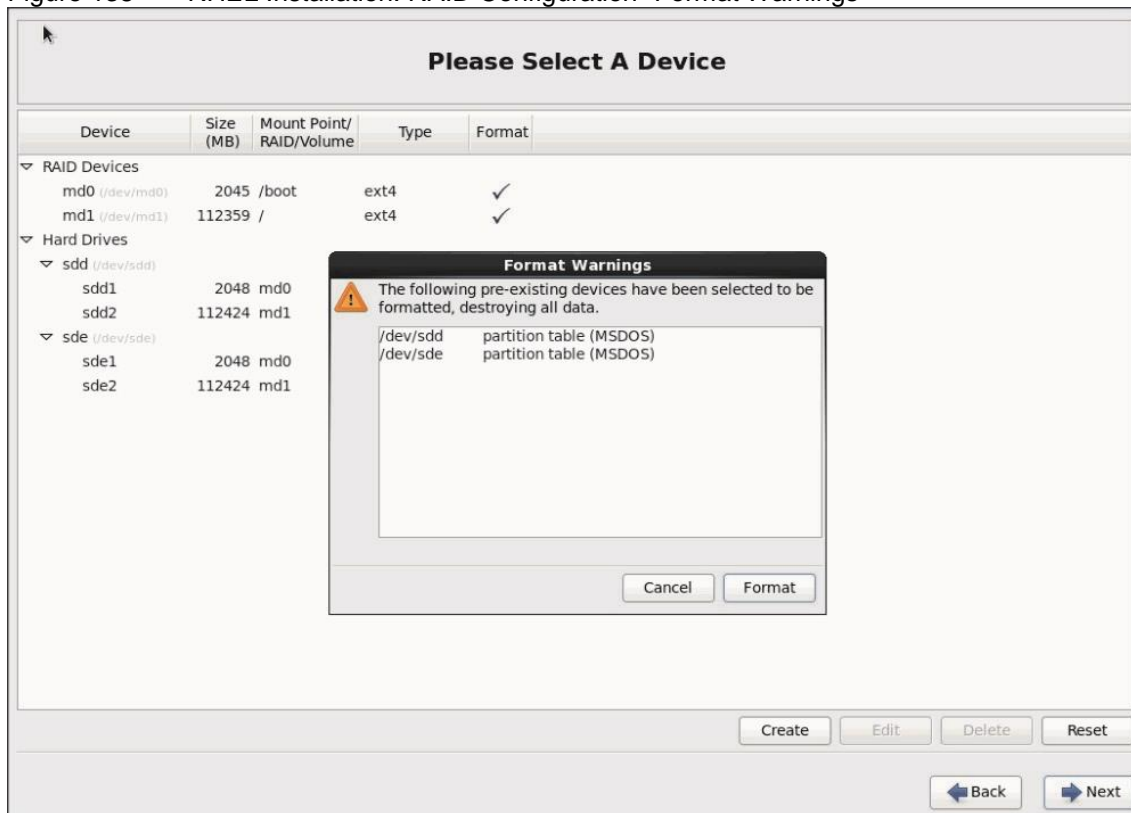
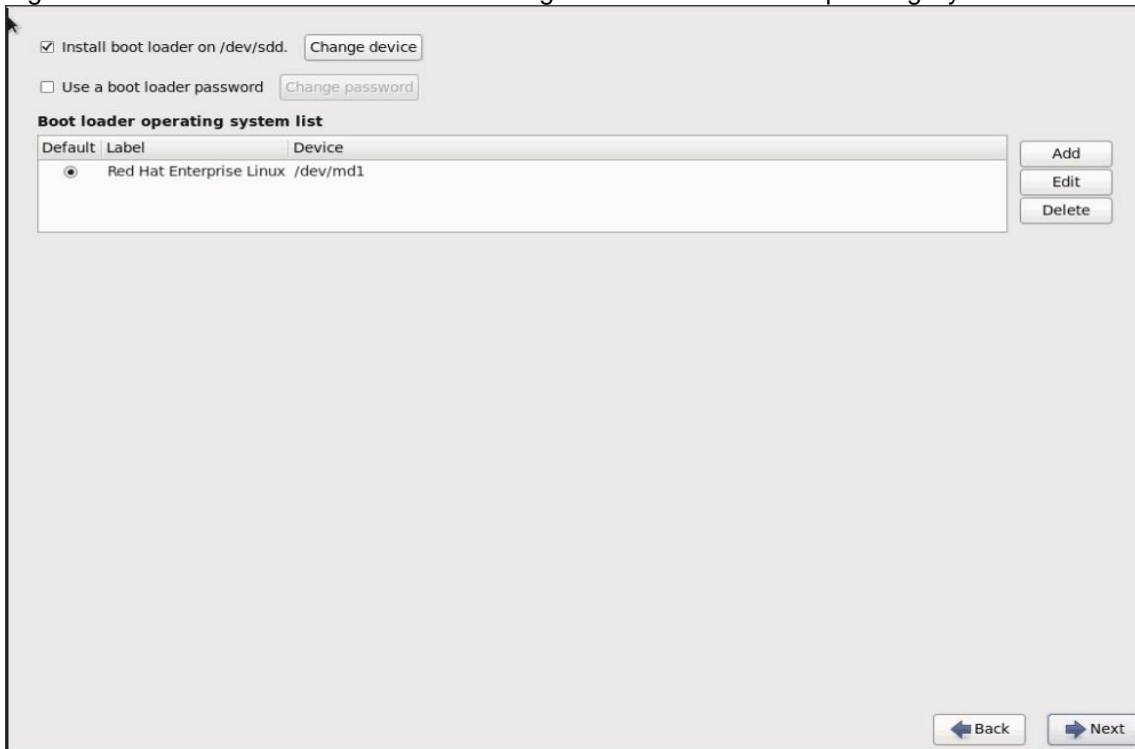


Figure 138 RHEL Installation: RAID Configuration- Format Warnings



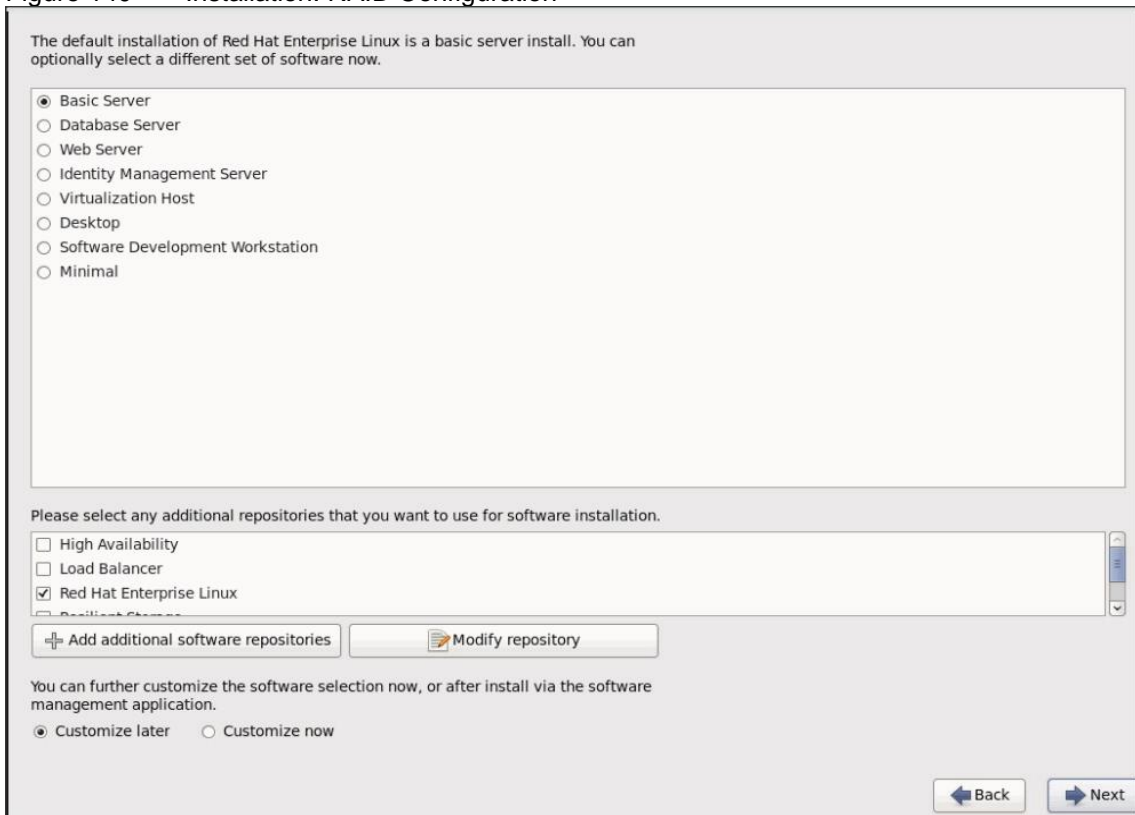
34. Select Default Settings and click Next.

Figure 139 RHEL Installation: RAID Configuration- Boot Loader Operating System



35. Continue with the RHEL Installation as shown below.

Figure 140 Installation: RAID Configuration



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

37. Repeat steps 1 to 36 to install Red Hat Enterprise Linux 6.8 on the other seven Cisco C240 M4 servers i.e. idx2 - idx8. The hostnames and their corresponding IP addresses are shown in Table 6 .

The table below shows how the hosts are assigned with their host names. Within the UCS domain, the eth0 network (i.e. 10.29.160.X subnet) over Fabric A is used as the primary network for all Splunk related data traffic except the replication traffic. The Splunk index replication-related data traffic will be configured to use eth1 over Fabric B.

For example, the host names associated with the various interfaces of an indexer i.e. idx1 are as follows:

- eth0:
 - Used to ingest the data streaming in from forwarders, and for traffic between the search head and indexers.
 - Used for management traffic such as SSH, Web UI, NTP sync.
 - Hostname is idx1.
 - Configured with Platinum QOS policy.
- eth1:
 - Used by the indexers to replicate indexes across each other.
 - Hostname is idx-rep.
 - Configured with Platinum QOS policy.

Table 6 Hostnames and IP Addresses

Hostname	eth0 Management Network, Data Ingestion	eth1 Data Replication Hostname: <hostname>-rep
admin1	10.29.160.101	192.168.11.101
admin2	10.29.160.102	192.168.11.102
idx1	10.29.160.103	192.168.11.103
idx2	10.29.160.104	192.168.11.104
idx3	10.29.160.105	192.168.11.105
idx4	10.29.160.106	192.168.11.106
idx5	10.29.160.107	192.168.11.107
idx6	10.29.160.108	192.168.11.108
idx7	10.29.160.109	192.168.11.109
idx8	10.29.150.110	192.168.11.110
sh1	10.29.150.111	192.168.11.111
sh2	10.29.150.112	192.168.11.112
sh3	10.29.150.113	192.168.11.113
storage1	10.29.150.114	192.168.11.114

Installing Red Hat Enterprise Linux 6.8 on the C3260

There are multiple methods 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 6.8 DVD/ISO for the installation

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 `Chassis` and then `Servers`.
4. Right-click on the server and select `KVM Console`.
5. Follow the directions in the section "Installing the Operation System" under "Installing Red Hat Enterprise Linux 6.8 on C220 M4 Systems."

Post OS Install Configuration

Choose one of the admin nodes of the cluster for management such as installation, cluster parallel shell, creating a local Red Hat repo, and others. In this document, we use admin1 for this purpose.

Configuring /etc/hosts

Setup /etc/hosts on the Admin node and other nodes as follows:

Follow the steps below to create the host file across all the nodes in the cluster:

1. Login to the Admin Node (admin1)

```
ssh 10.29.160.101
```

2. Populate the host file with IP addresses and corresponding hostnames. We will later copy this over to the other nodes.

On the Admin Node (admin1)

```
vi /etc/hosts
```

```
127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
10.29.160.101    admin1
192.168.11.101   admin1-rep
10.29.160.102    admin2
192.168.11.102   admin2-rep
10.29.160.103    idx1
192.168.11.103   idx1-rep
10.29.160.104    idx2
192.168.11.104   idx2-rep
10.29.160.105    idx3
192.168.11.105   idx3-rep
10.29.160.106    idx4
192.168.11.106   idx4-rep
10.29.160.107    idx5
192.168.11.107   idx5-rep
10.29.160.108    idx6
192.168.11.108   idx6-rep
```



```

10.29.160.109    idx7
192.168.11.109   idx7-rep
10.29.160.110    idx8
192.168.11.110   idx8-rep
10.29.160.111    sh1
192.168.11.111   sh1-rep
10.29.160.112    sh2
192.168.11.112   sh2-rep
10.29.160.113    sh3
192.168.11.113   sh3-rep
10.29.160.114    storage1

```

Setting Up Password-less Login

To manage all of the cluster's nodes from the admin node, setup password-less login. It assists in automating common tasks with Cluster-Shell - a cluster-wide parallel shell command utility, and shell-scripts without having to use passwords.

Once Red Hat Linux is installed across all the nodes in the cluster, to enable password-less login across all the nodes, complete the following steps:

1. Login to the Admin Node (admin1). For example: `ssh 10.29.160.101`
2. Run the `ssh-keygen` command to create both public and private keys on the admin node.

```

[root@admin1 ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
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:
42:4e:45:fb:0f:1d:a5:7b:4c:0a:59:13:12:3b:bc:93 root@admin1
The key's randomart image is:
+--[ RSA 2048 ]-----+
|      . o o . + .    |
|      . o = +        |
|      o . * o .     |
|      + . * *        |
|      o S E + o     |
|      . + .         |
|      .             |
+-----+
[root@admin1 ~]# █

```

3. Then run the following script 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 host in admin1 admin2 idx1 idx2 idx3 idx4 idx5 idx6 idx7 idx8 sh1 sh2 sh3
storagel; do echo -n "$host -> "; ssh-copy-id -i ~/.ssh/id_rsa.pub $host; done
```

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



Note: The admin node's /etc/hosts should be copied over to all thirteen other servers by using the cluster shell command after it is installed. See the next section, Setting up ClusterShell.

Setting up ClusterShell

ClusterShell (or clush) is a cluster-wide shell to run commands on several hosts in parallel.

1. From a system connected to the Internet, download ClusterShell (clush) and install it on admin1. ClusterShell is available from the EPEL (Extra Packages for Enterprise Linux) repository.

2. Download clustershell and copy it to the root folder of admin1.

```
wget http://dl.fedoraproject.org/pub/epel/6/x86_64/clustershell-1.7.2-1.el6.noarch.rpm
```

```
scp clustershell-1.7.2-1.el6.noarch.rpm admin1:/root/
```

3. Login to admin1 and install cluster shell.

```
yum -y install clustershell-1.7.2-1.el6.noarch.rpm
```

4. Edit the /etc/clustershell/groups file to include hostnames for all the nodes of the cluster. Create four special groups besides the group that takes all the hosts of the cluster. These groups help target the cluster wide commands to a specific set of nodes grouped by their role in the Splunk deployment.

```
vi /etc/clustershell/groups
```

5. Copy and paste the content below and save the groups file.

```
admins: admin[1-2]
```

```
indexers: idx[1-8]
```

```
searchheads: sh[1-3]
```

```
storage: storagel
```

```
all-splunk: admin[1-2],sh[1-3],idx[1-8]
```

```
all: admin[1-2],sh[1-3],idx[1-8],storagel
```

```
[root@admin1 ~]# cat /etc/clustershell/groups
admins: admin[1-2]
indexers: idx[1-8]
searchheads: sh[1-3]
storage: storagel
all-splunk: admin[1-2],sh[1-3],idx[1-8]
all: admin[1-2],sh[1-3],idx[1-8],storagel
```

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

When the IP address or different hostname i.e. idx1 or 10.29.160.103 is used to configure the `/etc/clustershell/groups` file, ClusterShell will not work until a manual SSH session is initiated to the machine by using that IP or hostname (as it needs to be in the `known_hosts` file), for instance, as in the case below for idx1 and 10.29.160.103.

```
[root@admin1 ~]# ssh 10.29.160.103
The authenticity of host '10.29.160.103 (10.29.160.103)' can't be established.
RSA key fingerprint is 94:db:4e:7c:8b:cb:3d:ee:1c:a7:fe:ae:1f:02:1a:de.
Are you sure you want to continue connecting (yes/no)? yes
```

6. From the admin node, i.e. admin1, copy over the `/etc/hosts` file to all the other servers.

```
clush -a -B -x admin1 -c /etc/hosts
```

```
[root@admin1 ~]# clush -a -B -x admin1 -c /etc/hosts
```

Creating Red Hat Enterprise Linux (RHEL) 6.8 Local Repo

To create a repository using RHEL DVD or ISO on the admin node (in this deployment, admin1 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 admin1. Create a directory that would contain the repository.

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

2. Copy the contents of the RedHat DVD to `/var/www/html/rhelrepo`

3. Alternatively, if you have access to a RedHat ISO Image, copy the ISO file to admin1.

```
scp rhel-server-6.8-x86_64-dvd.iso admin1:/root/
```



Note: Make sure the Red Hat ISO file is located in your present working directory. Use the IP address of the admin node instead of the hostname admin1 if hostnames have not been configured on this computer.

4. Mount the Red Hat ISO image.

```
mkdir -p /mnt/rheliso
```

```
mount -t iso9660 -o loop /root/rhel-server-6.8-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@admin1 ~]# mkdir -p /var/www/html/rhelrepo
[root@admin1 ~]# mkdir -p /mnt/rheliso
[root@admin1 ~]# mount -t iso9660 -o loop /root/rhel-server-6.8-x86_64-dvd.iso /mnt/rheliso/
[root@admin1 ~]# cp -r /mnt/rheliso/* /var/www/html/rhelrepo
```

6. On admin1, create a `.repo` file to enable the use of the yum command.

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

```
[rhel6.8]
```

```

name=Red Hat Enterprise Linux 6.8

baseurl=http://10.29.160.101/rhelrepo

gpgcheck=0

enabled=1

```

7. Now copy the `rheliso.repo` file from `/var/www/html/rhelrepo` to `/etc/yum.repos.d` on `admin1`

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



Based on this repo file, yum requires httpd to be running on `admin1` for other nodes to access the repository.

8. Copy the `rheliso.repo` to all the nodes of the cluster.

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

```
[root@admin1 ~]# clush -a -b -c /etc/yum.repos.d/rheliso.repo
```

9. To make use of repository files on `admin1` without httpd, edit the `baseurl` of repo file `/etc/yum.repos.d/rheliso.repo` to point to the repository location in the file system.



Note: This step is needed to install software on Admin Node (`admin1`) using the repo (such as `httpd`, `createrepo`, etc)

```

vi /etc/yum.repos.d/rheliso.repo
[rhel6.8]

name=Red Hat Enterprise Linux 6.8

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

gpgcheck=0

enabled=1

```

Creating the Red Hat Repository Database

1. Install the `createrepo` package on admin node (`admin1`). Use it to generate the repository database(s) for the local copy of the RHEL DVD contents.

```
yum -y install createrepo
```

```
[root@admin-1 ~]# yum -y install createrepo
Loaded plugins: product-id, search-disabled-repos, security, subscription-manager
This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package createrepo.noarch 0:0.9.9-24.el6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                Arch                Version              Size                Repository
=====
Installing:
createrepo             noarch              0.9.9-24.el6         96 k                RHEL6.8
=====
Transaction Summary
=====
Install      1 Package(s)
=====
```

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

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

createrepo .
```

```
[root@admin1 ~]# cd /var/www/html/rhelrepo/
[root@admin1 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
[root@admin1 rhelrepo]#
```

3. Finally, purge the yum caches after httpd is installed (steps outlined in the next section, "Installing httpd")

Installing httpd

Setting up RHEL repo on the admin node requires httpd. This section describes the process of setting one up:

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

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

```
yum -y install httpd
```

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

```
vi /etc/httpd/conf/httpd.conf
```

```
ServerName 10.29.160.101:80
```

```
[root@admin1 ~]# grep ServerName /etc/httpd/conf/httpd.conf
# ServerName gives the name and port that the server uses to identify itself.
#ServerName www.example.com:80
ServerName 10.29.160.101:80
```

3. Start httpd.

```
service httpd start
```

```
chkconfig httpd on
```

4. Purge the yum caches after httpd is installed (the last step in the previous section, “Creating the Red Hat Repository Database”).

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

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

```
[root@admin-1 ~]# clush -a -B yum clean all
admin-[1-3],index-[1-8],sh-[1-2] (12)
Loaded plugins: product-id, search-disabled-repos, security, subscription-
                : manager
This system is not registered to Red Hat Subscription Management. You can use subscription-manager to
register.
Cleaning repos: RHEL6.8
Cleaning up Everything
[root@admin-1 ~]# clush -a -B yum repolist
admin-[1-3],index-[1-8],sh-[1-2] (12)
Loaded plugins: product-id, search-disabled-repos, security, subscription-
                : manager
This system is not registered to Red Hat Subscription Management. You can use subscription-manager to
register.
repo id          repo name          status
RHEL6.8         RHEL6.8           3,997
repolist: 3,997
[root@admin-1 ~]#
```



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

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

Verify Cisco Network Driver for VIC1227

Ensure that the correct version of the kmod-enic driver is being used on all nodes:

```
clush -a -B "modinfo enic | head -5"
```

```
[root@admin1 ~]# clush -a -B "modinfo enic | head -5"
-----
admin[1-2],idx[1-8],sh[1-3],storaget1 (14)
-----
filename:      /lib/modules/2.6.32-431.el6.x86_64/extra/enic/enic.ko
version:      2.1.1.66
license:      GPL v2
author:       Scott Feldman <scofeldm@cisco.com>
description:  Cisco VIC Ethernet NIC Driver
```

Disabling SELinux

Security-Enhanced Linux (SELinux) must be disabled during the install procedure and cluster setup. SELinux can be enabled after installation and while the cluster is running.

1. To disable SELinux, edit **/etc/selinux/config** and change 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@admin1 ~]# clush -a -b "sed -i 's/SELINUX=enforcing/SELINUX=disabled/g' /e
tc/selinux/config"
[root@admin1 ~]#
[root@admin1 ~]# clush -a -b cat /etc/selinux/config
-----
admin[1-2],idx[1-8],sh[1-3],storaget1 (14)
-----
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
SELINUX=disabled
```

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



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

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

```
#clush -a -b sestatus
```

Disabling the Linux Firewall

The default Linux firewall settings are far too restrictive for any application deployment. Since the Cisco UCS Big Data deployment will be in its own isolated network, the firewall service is disabled.

```
clush -a -b "service iptables stop"
```

```
clush -a -b "chkconfig iptables off"
```

```
[root@admin1 ~]# clush -a -b service iptables stop
-----
admin[1-2],idx[1-8],sh[1-3],storage1 (14)
-----
iptables: Setting chains to policy ACCEPT: filter [ OK ]
iptables: Flushing firewall rules: [ OK ]
iptables: Unloading modules: [ OK ]
[root@admin1 ~]# clush -a -b chkconfig iptables off
```



Note: The user could re-configure the IP tables' settings in order to match the requirements of his/her particular deployment and turn the service back on. Consult Splunk documentation to determine the appropriate IP tables' settings.

Installing xfsprogs

1. From the admin node admin1, run the command below to install xfsprogs on all the nodes for xfs filesystem.

```
clush -a -B yum -y install xfsprogs
```

```
Loaded plugins: product-id, search-disabled-repos, security, subscription-
                : manager
This system is not registered to Red Hat Subscription Management. You can use subscription-manager to
register.
Setting up Install Process
Resolving Dependencies
--> Running transaction check
--> Package xfsprogs.x86_64 0:3.1.1-19.el6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                Arch          Version           Repository        Size
=====
Installing:
xfsprogs                x86_64        3.1.1-19.el6      RHEL6.8           725 k
=====
Transaction Summary
=====
Install      1 Package(s)

Total download size: 725 k
Installed size: 3.2 M
Downloading Packages:
Running rpm_check debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : xfsprogs-3.1.1-19.el6.x86_64                1/1
  Verifying  : xfsprogs-3.1.1-19.el6.x86_64                1/1

Installed:
  xfsprogs.x86_64 0:3.1.1-19.el6

Complete!
```

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 (admin1). Configuring NTP is critical for any clustered application.

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

1. Configure `/etc/ntp.conf` on the admin node 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 /tmp/ntp.conf on the admin node and copy it to all nodes.

```

vi /tmp/ntp.conf

server 10.29.160.101

driftfile /var/lib/ntp/drift

restrict 127.0.0.1

restrict -6 ::1

includefile /etc/ntp/crypto/pw

keys /etc/ntp/keys

```

3. Copy /tmp/ntp.conf from the admin node to /etc/ntp.conf of all the other nodes by executing the following command in the admin node (admin1)

```
clush -a -B -x admin1 -c /tmp/ntp.conf --dest=/etc/ntp.conf
```

```

[root@admin1 ~]# clush -a -B -x admin1 -c /tmp/ntp.conf --dest=/etc/ntp.conf
[root@admin1 ~]# clush -a -B cat /etc/ntp.conf
-----
admin2,idx[1-8],sh[1-3],storagel (13)
-----
server 10.29.160.101
driftfile /var/lib/ntp/drift
restrict 127.0.0.1
restrict -6 ::1
includefile /etc/ntp/crypto/pw
keys /etc/ntp/keys
-----
admin1
-----
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

```

4. Start the NTP service on the admin node (admin1).

```
service ntpd start
```

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

```
clush -a -B "yum install -y ntpdate"

clush -a -b -x admin1 "service ntpd stop"

clush -a -b -x admin1 "ntpdate 10.29.160.101"

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

6. Ensure restart of NTP daemon across reboots.

```
clush -a -b "chkconfig ntpd on"
```

Enabling Syslog

To preserve logs regarding killed processes or failed job, enable Syslog on each node. Versions such as syslog-ng and rsyslog are used, making it more difficult to be sure that a syslog daemon is present.

To confirm that the service is properly configured, run the following commands:

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

```
[root@admin1 ~]# clush -B -a rsyslogd -v
-----
admin[1-2],idx[1-8],sh[1-3],storage1 (14)
-----
rsyslogd 5.8.10, 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

See http://www.rsyslog.com for more information.
[root@admin1 ~]#
```

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

Setting Ulimit

In Linux, the 'nofile' property in /etc/security/limits.conf defines the number of i-nodes that can be opened simultaneously. With the default value of 1024, the system may appear to be out of disk space and would show no i-nodes are available. This value should be set to 64000 on every node for users root and splunk.



Note: When the Splunk Enterprise software is installed, a service user account by name "splunk" gets created automatically. Since, all Splunk related operations are performed as user "splunk", its ulimits need to be increased as well. Higher values are unlikely to result in an appreciable performance gain.

1. Set the "nofile" properties of root and splunk users to 64000 by editing the **/etc/security/limits.conf** on the admin node. Add the following lines to this file.

```
root soft nofile 64000

root hard nofile 64000

splunk soft nofile 64000

splunk hard nofile 64000
```

```
[root@admin1 ~]# grep 64000 /etc/security/limits.conf
root soft nofile 64000
root hard nofile 64000
splunk soft nofile 64000
splunk hard nofile 64000
```

2. Now, copy the **/etc/security/limits.conf** file from admin node (admin1) to all the nodes using the following command.

```
clush -a -B -c /etc/security/limits.conf

clush -a -B grep 64000 /etc/security/limits.conf
```

```
[root@admin1 ~]# clush -a -B -c /etc/security/limits.conf
[root@admin1 ~]# clush -a -B grep 64000 /etc/security/limits.conf

-----
admin[1-2],idx[1-8],sh[1-3],storage1 (14)
-----

root soft nofile 64000
root hard nofile 64000
splunk soft nofile 64000
splunk hard nofile 64000
```

3. Verify the ulimit settings by running the following command. The command should report 64000.

```
clush -B -a ulimit -n
```



ulimit values are applied only to a new shell, running the command on a node from 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, setting the number of TCP retries to 5 can help detect unreachable nodes with less latency.

1. Edit the file **/etc/sysctl.conf** on admin1 and add the following line:

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

2. Copy the **/etc/sysctl.conf** file from the admin node (admin1) to all the other nodes using the following command:

```
clush -a -b -c /etc/sysctl.conf
```

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

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

Configure VM Swapping

`vm.swappiness`, with a value from 0 to 100, controls the degree to which the system swaps. A high value prioritizes system performance, aggressively swapping processes out of physical memory when they are not active. A low value avoids swapping processes out of physical memory for as long as possible. In order to reduce swapping, run the following on all nodes. The default value is 60.

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

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 the 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.

1. From the admin node, run the following commands:

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

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

2. Run the commands above every time the Linux system starts up. Add these commands to `/etc/rc.local`, so they are executed automatically upon every reboot.

3. From the admin node, run the following commands:

```
rm -f /root/thp_disable
```

```
echo "echo never > /sys/kernel/mm/redhat_transparent_hugepage/enabled" > /root/thp_disable
```

```
echo "echo never > /sys/kernel/mm/redhat_transparent_hugepage/defrag " >> /root/thp_disable
```

4. Copy the file over to all the nodes.

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

5. Append the content of file `thp_disable` to `/etc/rc.local`.

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

Installing LSI StorCLI utility on all Indexers and Archival Nodes

This section describes steps to configure non-OS disk drives as RAID10 using StorCli command as described below. All the drives are going to be part of a single RAID10 volume. From the website below, download StorCLI: http://docs.avagotech.com/docs/1.19.04_StorCLI.zip

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

```
scp storcli-1.19.04-1.noarch.rpm admin1:/tmp/
```

3. Copy storcli rpm to all the Indexers and Storage node(s) using the following commands:

```
clush -a -b -X searchheads,admins -c /tmp/storcli-1.19.04-1.noarch.rpm
```

4. Run the below command to install storcli all the Indexers and Storage node(s).

```
clush -a -b -X searchheads,admins rpm -ivh /tmp/storcli-1.19.04-1.noarch.rpm
```

Configuring the Virtual Drive on the Indexers

1. Create a script by name `raid10.sh` on the admin node and copy it over to all indexers.

```
vi /root/raid10.sh
```

2. Paste the following contents into the file and save it.

```
/opt/MegaRAID/storcli/storcli64 /c0 add vd type=raid10 drives=$1:1-24
pdperarray=12 WB ra direct Strip=128
```



Note: Do not execute this script on the admin or search head nodes. This script is meant only for the indexers.



Note: This script needs to be executed on each of the indexer nodes manually. This is because the script takes the EnclosureID as Input, which would generally be different on different indexer servers.

3. Change the mode to include execution privileges.

```
chmod +x /root/raid10.sh
```

4. Copy the script over to all the indexers.

```
clush --group=indexers -c /root/raid10.sh
```

```
[root@admin1 ~]# cat raid10.sh
/opt/MegaRAID/storcli/storcli64 /c0 add vd type=raid10 drives=$1:1-24 pdperarray
=12 WB ra direct Strip=128

[root@admin1 ~]# clush --group=indexers -c /root/raid10.sh
```

5. The script above requires enclosure ID as a parameter. Run the following command to get EnclosureID on each indexer by launching an SSH session onto that indexer.

```
/opt/MegaRAID/storcli/storcli64 pdlist -a0 | grep Enc | grep -v 252 | awk
'{print $4}' | sort | uniq -c | awk '{print $2}'
```

6. Run the script to create a single RAID10 volume as follows:

```
./raid10.sh <EnclosureID> # obtained by running the command above
```



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

```
[root@admin1 ~]# ssh idx1
Last login: Thu Apr  9 04:04:38 2015 from admin1
[root@idx1 ~]#
[root@idx1 ~]# cat raid10.sh
/opt/MegaRAID/storcli/storcli64 /c0 add vd type=raid10 drives=$1:1-24 pdperarray
=12 WB ra direct Strip=128

[root@idx1 ~]# /opt/MegaRAID/storcli/storcli64 pdlist -a0 | grep Enc | grep -v 2
52 | awk '{print $4}' | sort | uniq -c | awk '{print $2}'
0
[root@idx1 ~]# ./raid10.sh 0
Controller = 0
Status = Success
Description = Add VD Succeeded
```



Note: The above figure shows the procedure for creating virtual drive on one indexer. This process needs to be performed on all eight indexers individually.

Configuring the XFS File System

The following script will format and mount the RAID10 volume (virtual drive) that was created in the previous section. It looks at all available volumes on the indexers, but will skip OS/boot related volumes. The RAID10 volume will be mounted based on its UUID as /data/disk1. This script assumes a strip size of 128K and one RAID10 volume with 12 physical drives in each span.

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

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



Note: The script assumes there are no partitions already existing on the data volumes. If there are partitions, then they have to be deleted first before running this script. This process is documented in the Note at the end of the section.

```
vi /root/driveconf-idx.sh

#!/bin/bash
[[ "-x" == "${1}" ]] && set -x && set -v && shift 1
count=1
for devX in /sys/class/scsi_host/host?/scan
do
```

```

echo '- - -' > ${devX}
done
for devD in $(lsblk | grep disk | cut -c1-3)
do
echo /dev/${devD}
devX=/dev/${devD}
if [[ -b ${devX} && ` /sbin/parted -s ${devX} print quit|/bin/grep -c boot` -
ne 0 ]]
then
echo "${devX} bootable - skipping."
continue
else
echo $devX
echo "Setting up Drive => ${devX}"
/sbin/mkfs.xfs -f ${devX}
(( $? )) && continue
#Identify UUID
UUID=`blkid ${devX} | cut -d " " -f2 | cut -d "=" -f2 | sed 's/"//g'`

echo "UUID of ${devX} = ${UUID}, mounting ${devX} as UUID on
/data/disk${count}"
/bin/mkdir -p /data/disk${count}
(( $? )) && continue
/bin/mount -t xfs -o allocsize=128m,inode64,noatime,nobarrier,nodiratime -U
${UUID} /data/disk${count}
(( $? )) && continue
echo "UUID=${UUID} /data/disk${count} xfs
allocsize=128m,inode64,noatime,nobarrier,nodiratime 0 0" >> /etc/fstab
((count++))
fi
done

```

2. Run the following command to copy driveconf-idx.sh to all the indexers.

```

chmod 755 /root/driveconf-idx.sh

clush --group=indexers -B -c /root/driveconf-idx.sh

```

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

```

clush --group=indexers -B /root/driveconf-idx.sh

```

4. Run the following from the admin node to list the partitions and mount points to ensure that the drive /data/disk1 is mounted properly.

```

clush --group=indexers -B df -h

clush --group=indexers -B mount

clush --group=indexers -B cat /etc/fstab

```



Note: In-case there is a need to delete any partitions; it can be done so using the following: Run command 'mount' to identify which drive is mounted to which device: /dev/sd<?>. Unmount the drive for the partition to be deleted and run fdisk to delete it as shown below. Be careful **not to delete the OS partition** as this will wipe out the installed OS.

```
mount

umount /data/disk1 # <- disk1 shown as example

(echo d; echo w;) | sudo fdisk /dev/sd<?>
```

Configuring Data Drives on Archival Nodes

This section describes steps to configure non-OS disk drives as 4 RAID6 volumes each with 15 drives using the StorCLI command as described below. These volumes shall be shared as NFS exports and will be used for archiving frozen data.

1. In the node admin1, create a shell script with the StorCLI commands to create four RAID6 volumes with 15 drives in each volume.

```
vi /root/raid6.sh

/opt/MegaRAID/storcli/storcli64 /c0 add vd type=raid6 drives=$1:1-15 WB ra
direct Strip=1024

/opt/MegaRAID/storcli/storcli64 /c0 add vd type=raid6 drives=$1:16-30 WB ra
direct Strip=1024

/opt/MegaRAID/storcli/storcli64 /c0 add vd type=raid6 drives=$1:31-45 WB ra
direct Strip=1024

/opt/MegaRAID/storcli/storcli64 /c0 add vd type=raid6 drives=$1:46-60 WB ra
direct Strip=1024
```

2. Copy over this script to all the storage nodes.

```
clush --group=storage -B -c /root/raid6.sh
```

3. Log onto each of the storage servers as user root.



Note: This document covers the procedure for creating the RAID volumes on only one storage server. If there are multiple storage nodes required in the solution, this process needs to be repeated on all the storage nodes individually.

4. Use the following command to add execute permissions to the shell script.

```
chmod +x ./raid6.sh
```

5. Execute the following command to get the enclosure ID of the controller.

```
/opt/MegaRAID/storcli/storcli64 pdlist -a0 | grep Enc | grep -v 252 | awk
'{print $4}'

| sort | uniq -c | awk '{print $2}'
```


- Execute the shell script with the EnclosureID obtained in the previous step.

```
/raid6.sh <EnclosureID>
```

- Verify the virtual drives created by using the following command.

```
lsblk
```

```
[root@storagel ~]# /opt/MegaRAID/storcli/storcli64 pdlist -a0 | grep Enc | gre
p -v 252 | awk '{print $4}' | sort | uniq -c | awk '{print $2}'
```

```
65
```

```
[root@storagel ~]# ./raid6.sh 65
Controller = 0
Status = Success
Description = Add VD Succeeded

Controller = 0
Status = Success
Description = Add VD Succeeded

Controller = 0
Status = Success
Description = Add VD Succeeded

Controller = 0
Status = Success
Description = Add VD Succeeded
```



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

Configuring the XFS File System

The following script will format and mount the available volumes on each archival node. OS boot partitions are skipped. All drives shall be mounted based on their UUID as /data/disk1, /data/disk2, and so on.

- On the admin node, create a file containing the following script.

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



Note: The script assumes there are no partitions already existing on the data volumes. If there are partitions, then they have to be deleted first before running this script. This process is documented in the note at the end of the section.

- Create a file named "driveconf-arch.sh" and copy-paste the following contents.

```
vi driveconf-arch.sh
```



Note: Copy and paste the content below into the script file.

```
#!/bin/bash
[[ "-x" == "${1}" ]] && set -x && set -v && shift 1
```

```

count=1
for devX in /sys/class/scsi_host/host?/scan
do
echo '- - -' > ${devX}
done
for devD in $(lsblk | grep disk | cut -c1-3)
do
echo /dev/${devD}
devX=/dev/${devD}
if [[ -b ${devX} && `sbin/parted -s ${devX} print quit|/bin/grep -c boot` -
ne 0 ]]
then
echo "$devX bootable - skipping."
continue
else
echo $devX
echo "Setting up Drive => ${devX}"
/sbin/mkfs.xfs -f ${devX}
(( $? )) && continue
#Identify UUID
UUID=`blkid ${devX} | cut -d " " -f2 | cut -d "=" -f2 | sed 's/"//g'`

echo "UUID of ${devX} = ${UUID}, mounting ${devX} as UUID on
/data/disk${count}"
/bin/mkdir -p /data/disk${count}
(( $? )) && continue
/bin/mount -t xfs -o allocsize=128m,inode64,noatime,nobarrier,nodiratime -U
${UUID} /data/disk${count}
(( $? )) && continue
echo "UUID=${UUID} /data/disk${count} xfs
allocsize=128m,inode64,noatime,nobarrier,nodiratime 0 0" >> /etc/fstab
((count++))
fi
done

```

3. Copy the driveconf-arch.sh script file to all the Storage nodes.

```
clush --group=storage -B -c /root/driveconf-arch.sh
```

4. Execute the script from the admin node targeting all the storage nodes.

```
clush --group=storage -B /root/driveconf-arch.sh
```

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

```
clush --group=storage -B df -h
```

```
clush --group=storage -B mount
```

```
clush --group=storage -B cat /etc/fstab
```

```
[root@admin1 ~]# clush --group=storage -B df -h
-----
storagel
-----
Filesystem      Size  Used Avail Use% Mounted on
/dev/md1        109G  2.0G  101G   2% /
tmpfs           127G   0    127G   0% /dev/shm
/dev/md0        2.0G   64M   1.9G   4% /boot
/dev/sdc1       48T   34M   48T   1% /data/disk1
/dev/sdd1       48T   34M   48T   1% /data/disk2
/dev/sde1       48T   34M   48T   1% /data/disk3
/dev/sdf1       48T   34M   48T   1% /data/disk4
[root@admin1 ~]#
[root@admin1 ~]# clush --group=storage -B mount
-----
storagel
-----
/dev/md1 on / type ext4 (rw)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
tmpfs on /dev/shm type tmpfs (rw,rootcontext="system_u:object_r:tmpfs_t:s0")
/dev/md0 on /boot type ext4 (rw)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
/dev/sdc1 on /data/disk1 type xfs (rw,noatime,nodiratime,allocsize=128m,nobarrier)
/dev/sdd1 on /data/disk2 type xfs (rw,noatime,nodiratime,allocsize=128m,nobarrier)
/dev/sde1 on /data/disk3 type xfs (rw,noatime,nodiratime,allocsize=128m,nobarrier)
/dev/sdf1 on /data/disk4 type xfs (rw,noatime,nodiratime,allocsize=128m,nobarrier)
[root@admin1 ~]#

[root@admin1 ~]# clush --group=storage -B cat /etc/fstab
-----
storagel
-----
#
# /etc/fstab
# Created by anaconda on Wed Apr  8 16:29:20 2015
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=da022d79-5a23-4743-b595-a06350489791 /                ext4    defaults
    1 1
UUID=3d452238-001e-41f4-8e94-d2299c483dbb /boot            ext4    defaults
    1 2
tmpfs                /dev/shm        tmpfs    defaults    0 0
devpts               /dev/pts        devpts   gid=5,mode=620 0 0
sysfs                /sys            sysfs    defaults    0 0
proc                 /proc           proc     defaults    0 0
UUID=ee0659e0-0e4d-44a2-aebc-560d24e41bd8 /data/disk1     xfs     allocsize=128m,noatime,no
barrier,nodiratime 0 0
UUID=76465cf8-c5cd-4819-ba09-9a07c74720c3 /data/disk2     xfs     allocsize=128m,noatime,no
barrier,nodiratime 0 0
UUID=9b4014cd-3b05-4bee-9d91-1af0c0d029a5 /data/disk3     xfs     allocsize=128m,noatime,no
barrier,nodiratime 0 0
UUID=11c8cfd9-6e56-4325-87c9-fladd1celbc0 /data/disk4     xfs     allocsize=128m,noatime,no
barrier,nodiratime 0 0
```

```
[root@admin1 ~]# clush --group=storage -B cat /etc/fstab
-----
storage1
-----
#
# /etc/fstab
# Created by anaconda on Wed Apr  8 16:29:20 2015
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=da022d79-5a23-4743-b595-a06350489791 / ext4 defaults
1 1
UUID=3d452238-001e-41f4-8e94-d2299c483dbb /boot ext4 defaults
1 2
tmpfs /dev/shm tmpfs defaults 0 0
devpts /dev/pts devpts gid=5,mode=620 0 0
sysfs /sys sysfs defaults 0 0
proc /proc proc defaults 0 0
UUID=ee0659e0-0e4d-44a2-aebc-560d24e41bd8 /data/disk1 xfs allocsize=128m,noatime,no
barrier,nodiratime 0 0
UUID=76465cf8-c5cd-4819-ba09-9a07c74720c3 /data/disk2 xfs allocsize=128m,noatime,no
barrier,nodiratime 0 0
UUID=9b4014cd-3b05-4bee-9d91-1af0c0d029a5 /data/disk3 xfs allocsize=128m,noatime,no
barrier,nodiratime 0 0
UUID=11c8cfd9-6e56-4325-87c9-f1add1ce1bc0 /data/disk4 xfs allocsize=128m,noatime,no
barrier,nodiratime 0 0
```



Note: To delete any of the partitions, run the 'mount' command to identify which drive is mounted to which device.

Unmount the drive with the Cluster Verification partition to be deleted and run **fdisk** to delete it as shown below.

Be careful **not to delete the OS partition** as this will wipe out the installed OS.

```
mount
```

```
umount /data/disk1 # <- disk1 shown as example
```

```
(echo d; echo w;) | sudo fdisk /dev/sd<?>
```

Cluster Verification

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

1. Create the `cluster_verification.sh` script as follows on the Admin node (admin1):

```
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 \
Cluster

Verification === ${NC}"

echo ""

echo ""

echo -e "${green} ==== System Information ==== ${NC}"

echo ""

echo ""

echo -e "${green}System ${NC}"

clush -a -B " `which dmidecode` |grep -A2 '^System Information'"

echo ""

echo ""

echo -e "${green}BIOS ${NC}"

clush -a -B " `which dmidecode` | grep -A3 '^BIOS I'"

echo ""

echo ""

echo -e "${green}Memory ${NC}"

clush -a -B "cat /proc/meminfo | grep -i ^memt | uniq"

echo ""

echo ""

echo -e "${green}Number of Dimms ${NC}"

clush -a -B "echo -n 'DIMM slots: '; `which dmidecode` |grep -c \
'^[[:space:]]*Locator:'"

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

clush -a -B " `which dmidecode` | awk '/Memory Device$/ ,/^$/ {print}' | grep -e \
'^Mem' -e Size: -e Speed: -e Part | sort -u | grep -v -e 'NO DIMM' -e 'No Module Installed' -e Unknown"

echo ""

```

```

echo ""

# probe for cpu info #

echo -e "${green}CPU ${NC}"

clush -a -B "grep '^model name' /proc/cpuinfo | sort -u"

echo ""

clush -a -B "`which lscpu` | grep -v -e op-mode -e ^Vendor -e family -e \
Model: -e Stepping: -e BogomIPS -e Virtual -e ^Byte -e '^NUMA node(s)'"

echo ""

echo ""

# probe for nic info #

echo -e "${green}NIC ${NC}"

clush -a -B "`which ifconfig` | egrep ' (^e|^p)' | awk '{print \$1}' | xargs -l \
\
`which ethtool` | grep -e ^Settings -e Speed"

echo ""

clush -a -B "`which lspci` | grep -i ether"

echo ""

echo ""

# probe for disk info #

echo -e "${green}Storage ${NC}"

clush -a -B "echo 'Storage Controller: '; `which lspci` | grep -i -e \
raid -e storage -e lsi"

echo ""

clush -a -B "dmesg | grep -i raid | grep -i scsi"

echo ""

clush -a -B "lsblk -id | awk '{print \$1,\$4}'|sort | nl"

echo ""

echo ""

echo -e "${green} ===== Software ===== ${NC}"

echo ""

```

```

echo ""

echo -e "${green}Linux Release ${NC}"

clush -a -B "cat /etc/*release | uniq"

echo ""

echo ""

echo -e "${green}Linux Version ${NC}"

clush -a -B "uname -srvm | fmt"

echo ""

echo ""

echo -e "${green}Date ${NC}"

clush -a -B date

echo ""

echo ""

echo -e "${green}NTP Status ${NC}"

clush -a -B "ntpstat 2>&1 | head -1"

echo ""

echo ""

echo -e "${green}SELINUX ${NC}"

clush -a -B "echo -n 'SElinux status: '; grep ^SELINUX= /etc/selinux/config
2>&1"

echo ""

echo ""

echo -e "${green}IPTables ${NC}"

clush -a -B "`which chkconfig` --list iptables 2>&1"

echo ""

clush -a -B "`which service` iptables status 2>&1 | head -10"

echo ""

echo ""

echo -e "${green}Transparent Huge Pages ${NC}"

clush -a -B " cat /sys/kernel/mm/*transparent_hugepage/enabled"

```

```

echo ""

echo ""

echo -e "${green}CPU Speed${NC}"

clush -a -B "echo -n 'CPUspeed Service: '; `which service` cpuspeed status
2>&1"

clush -a -B "echo -n 'CPUspeed Service: '; `which chkconfig` --list cpuspeed
2>&1"

echo ""

echo ""

echo -e "${green}Hostname Lookup${NC}"

clush -a -B " ip addr show"

echo ""

echo ""

echo -e "${green}Open File Limit${NC}"

clush -a -B 'echo -n "Open file limit(should be >32K): "; ulimit -n'

```

2. Change permissions to executable

```
chmod 755 cluster_verification.sh
```

3. Run the Cluster Verification tool from the admin node. This can be run before starting Splunk Enterprise software to identify any discrepancies in post-OS configuration among the servers.

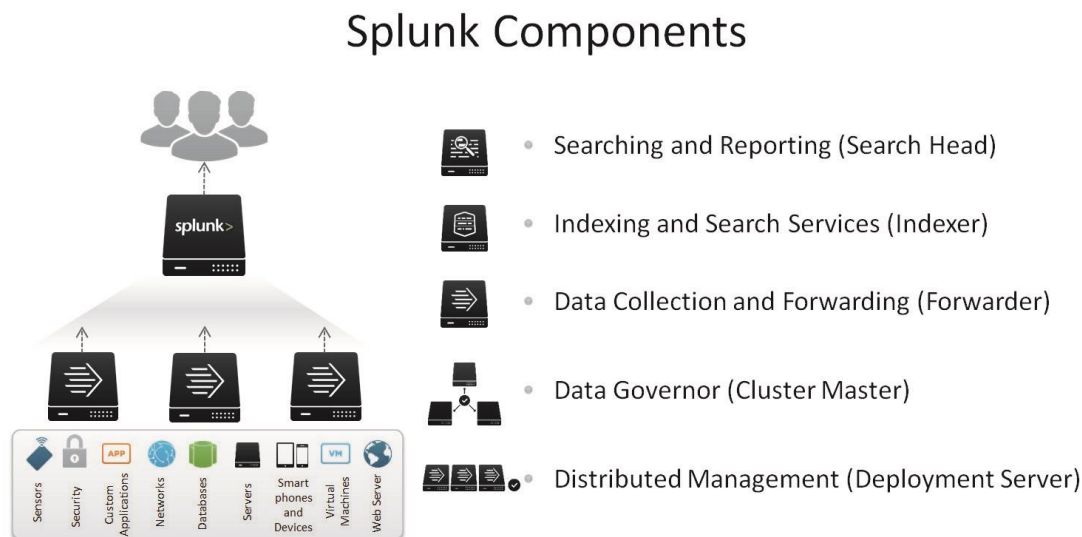
```
./cluster_verification.sh
```


Installing Splunk Enterprise 6.4

Splunk Architecture and Terminology

Splunk comes packaged as an 'all-in-one' distribution. The single file can be configured to function as one or all of the following components (Splunk's Universal Forwarder is a separate package). In a distributed deployment, installations follow a 3-tier approach. (Figure 141)

Figure 141 Splunk Components



- **Search Head:** A Splunk Enterprise instance that handles search management functions in a distributed search environment, directing search requests to a set of search peers and then merging the results back to the user. A Splunk Enterprise instance can function as both a search head and a search peer. If it does only searching (and not any indexing), it is usually referred to as a dedicated search head. Search head clusters are groups of search heads that coordinate their activities.
- **Indexer:** A Splunk Enterprise instance that indexes data, transforming raw data into events and placing the results into an index. It also searches the indexed data in response to search requests. The indexer also frequently performs the other fundamental Splunk Enterprise functions: data input and search management. In larger deployments, forwarders handle data input, and forward the data to the indexer for indexing. Similarly, although indexers always perform searches across their own data, in larger deployments, a specialized Splunk Enterprise instance, called a search head, handles search management and coordinates searches across multiple indexers.
- **Universal Forwarder:** A small-footprint version of a forwarder, a Splunk Enterprise instance that forwards data to another Splunk server or a third-party system without parsing.
- **Heavy Forwarder:** A fully functional Splunk instance that is configured to send data to the indexing tier. The heavy forwarder performs Splunk's parsing phase before forwarding the data.
- **Cluster Master (Master Node):** The indexer cluster node that regulates the functioning of an indexer cluster.
- **Deployment Server:** A Splunk Enterprise instance that acts as a centralized configuration manager, grouping together and collectively managing any number of Splunk Enterprise instances. Instances that are remotely configured by deployment servers are called deployment clients. The deployment server

downloads updated content, such as configuration files and apps, to deployment clients. Units of such content are known as deployment apps.

- **Deployer (not pictured):** A Splunk Enterprise instance that distributes apps and certain other configuration updates to search head cluster members
- **License Master (not pictured):** A license master controls one or more license slaves. From the license master, you can define stacks, pools, add licensing capacity, and manage license slaves.
- **Distributed Management Console (not pictured):** The distributed management console lets you view detailed performance information about your Splunk Enterprise deployment. The topics in this chapter describe the available dashboards and alerts.

In this Distributed Configuration, indexers and search heads are configured in a clustered mode. Splunk Enterprise supports clustering for both search heads and indexers.

- Search head cluster is a group of interchangeable and highly available search heads. By increasing concurrent user capacity and by eliminating single point of failure, search head clusters reduce the total cost of ownership.
- Indexer clusters are made up of groups of Splunk Enterprise indexers configured to replicate peer data so that the indexes of the system become highly available. By maintaining multiple, identical copies of indexes, clusters prevent data loss while promoting data availability for searching.
- An Archival node is configured to host the frozen data generated by the indexers. (See the sections NFS Configurations for Splunk Frozen Data Storage, and Configuring Archival of Data from Cold to Frozen)

For more information, please refer to [Splunk Documentation](#).

Splunk Services and Processes

A Splunk Enterprise server installs a process on your host, splunkd.

splunkd is a distributed C/C++ server that accesses, processes and indexes streaming IT data. It also handles search requests. splunkd processes and indexes your data by streaming it through a series of pipelines, each made up of a series of processors.

- Pipelines are single threads inside the splunkd process, each configured with a single snippet of XML.
- Processors are individual, reusable C or C++ functions that act on the stream of IT data passing through a pipeline. Pipelines can pass data to one another through queues. splunkd supports a command-line interface for searching and viewing results.
- Splunkd also provides the Splunk Web user interface. It allows users to search and navigate data stored by Splunk servers and to manage your Splunk deployment through a web interface. It communicates with your web browser through Representational State Transfer (REST).
- Splunkd runs administration and management services on port 8089 with SSL/HTTPS turned on by default.
- It also runs a web server on port 8000 with SSL/HTTPS turned off by default.

Planning the Installation

In this CVD, three (3) clustered Search Heads, eight (8) clustered indexers, a deployment server, a deployer, a distributed management console, a master node, and a license master are configured.

Installation order will be as follows:

- Splunk Installation

- Configure License Master
- Configure Master Node
- Configure Indexing Cluster
- Configure Deployer
- Configure Search Head Cluster
- Configure Distribution Management Console
- Configure Archival of frozen data
- Configure Deployment Server
- Install universal forwarder
- Verify Installation
- Post Install Clean up

It is highly recommended that assigned hostnames match their corresponding function, for example a search head may be 'splksrch1.domain.com' or an indexer may be idx1.domain.com. Throughout this document, instructions are provided and examples include the use of hostnames. Your deployment may or may not use the same hostnames. Use Table 7 to plan and track assigned roles and hostnames/IP addresses:

Table 7 Assigned Roles and IP Addresses

CVD Hostname	Function / Model	Hostname	IP
sh1	Search Head 1 C220 M4		
sh2	Search Head 2 C220 M4		
sh3	Search Head 3 C220 M4		
idx1	Indexer 1 C240 M4		
idx2	Indexer 2 C240 M4		
idx3	Indexer 3 C240 M4		
idx4	Indexer 4 C240 M4		
idx5	Indexer 4 C240 M4		
idx6	Indexer 4 C240 M4		
idx7	Indexer 4 C240 M4		
idx8	Indexer 4 C240 M4		

CVD Hostname	Function / Model	Hostname	IP
admin1	Admin Box 1 (Master Node, License Master, Distributed Management Console, Deployer) C220 M4		
admin2	Admin Box 2 Deployment Server C220 M4		
storage1	C3260		



Note: The IP addresses and hostnames used in this CVD can be found in Table 7.

Installing Splunk

The Splunk Enterprise software is a single software package that can be configured to function in a specific role. Installation of Splunk across all nodes will be the same, with no specific parameters required; configuration changes will be required for each respective component. As such, a simple installation across every server will be the base to build this architecture.

1. From a host connected to internet, download Splunk Enterprise software from the splunk.com website. Copy it over to the server admin1.

```
[root@admin1 ~]# ls splunk*
splunk-6.4.1-debde650d26e-linux-2.6-x86_64.rpm
```

2. Copy Splunk software over to all the nodes (2 admins, 3 search heads, and 8 indexers) but the storage nodes.

```
clush -a -X storage -c ./splunk-6.4.1-debde650d26e-linux-2.6-x86_64.rpm --
dest=/tmp
```

3. Modify the permissions on the Splunk Enterprise RPM file to include execution privileges.

```
clush -a -X storage chmod +x /tmp/splunk-6.4.1-debde650d26e-linux-2.6-
x86_64.rpm
```

4. Create a directory tree “/data/disk1” on the search heads and admin nodes.

```
clush --group=admins,searchheads mkdir -p /data/disk1
```



Note: The indexers already have a similar directory i.e. /data/disk1 which serves as the mount point for the RAID10 volume we created in the earlier sections. This step will make the directory structure uniform across all nodes where Splunk Enterprise is installed.

5. Install Splunk Enterprise in the directory /data/disk1 of the indexers, search heads and admin nodes.

```
clush -a -X storage -B rpm -ivh --prefix=/data/disk1 /tmp/splunk-6.4.1-
debde650d26e-linux-2.6-x86_64.rpm
```

```
[root@admin1 ~]# clush -a -X storage -c ./splunk-6.4.1-debde650d26e-linux-2.6-x86_64.rpm --dest=/tmp
[root@admin1 ~]# clush -a -X storage chmod +x /tmp/splunk-6.4.1-debde650d26e-linux-2.6-x86_64.rpm
[root@admin1 ~]# clush --group=admins,searchheads mkdir -p /data/disk1
[root@admin1 ~]# clush -a -X storage -B rpm -ivh --prefix=/data/disk1 /tmp/splunk-6.4.1-debde650d26e-
linux-2.6-x86_64.rpm
-----
admin[1-2],idx[1-2],sh[1-3] (7)
-----
warning: /tmp/splunk-6.4.1-debde650d26e-linux-2.6-x86_64.rpm: Header V4 DSA/SHA1 Signature, key ID 65
3fb112: NOKEY
Preparing...      #####
splunk            #####
complete
```

This step installs Splunk Enterprise and creates a user named splunk.



Note: When Splunk Enterprise is installed by means of the RPM package as mentioned above, the installation tool automatically creates a user splunk and group splunk.

6. Setup the Environment Variable:

```
clush --group=all-splunk "echo SPLUNK_HOME=/data/disk1/splunk >>
/etc/environment"
```

```
[root@admin1 ~]# clush --group=all-splunk -B "echo SPLUNK_HOME=
/data/disk1/splunk >> /etc/environment"
```

7. Log off and log back in to the server admin1.

8. Use the cluster-shell utility command to verify if the environment variable has been setup correctly.

```
clush --group=all-splunk -B echo $SPLUNK_HOME
```

```
[root@admin1 ~]# clush --group=all-splunk -B echo $SPLUNK_HOME
-----
admin[1-2],idx[1-8],sh[1-3] (13)
-----
/data/disk1/splunk
```

9. Verify the ownership of the SPLUNK_HOME directory and its contents. All of these files should belong to splunk user and splunk group.

```
clush --group=all-splunk -B ls -l $SPLUNK_HOME
```

```
clush --group=all-splunk -B ls -l $SPLUNK_HOME/bin/splunk
```

```
[root@admin1 ~]# clush --group=all-splunk -B ls -l $SPLUNK_HOME/bin/splunk
-----
admin2,idx[1,3,5-8],sh[1-3] (10)
-----
-r-xr-xr-x. 1 splunk splunk 356592 Feb 18 15:41 /data/disk1/splunk/bin/splunk
-----
admin1,idx[2,4] (3)
-----
-r-xr-xr-x 1 splunk splunk 356592 Feb 18 15:41 /data/disk1/splunk/bin/splunk
```

Setting Up Login for Splunk User

As mentioned above, the user 'splunk' is created without a password. This section describes the procedure to assign a password and configure the password-less login for that user account.

This facilitates the usage of Cluster-Shell commands.

1. From the admin node 'admin1', assign the password for the user 'splunk' on all the Splunk indexers, search heads and admin servers.

```
clush --group=all-splunk -B "echo cisco123 | passwd splunk --stdin"
```

```
[root@admin1 ~]# clush --group=all-splunk -B "echo cisco123 | p
passwd splunk --stdin"
-----
admin[1-2],idx[1-8],sh[1-3] (13)
-----
Changing password for user splunk.
passwd: all authentication tokens updated successfully.
```



Note: In this example, we are using a command line method with clear-text password for the sake of simplification. It is recommended to setup a strong password and set the password manually on each server individually to match the target datacenter's security practices.

2. Log onto the admin node as user splunk using the password selected in the above step.
3. Run the `ssh-keygen` command to create both public and private keys on the admin node for the user 'splunk'.

```
login as: splunk
splunk@10.29.160.101's password:
Last login: Mon Apr 13 12:20:14 2015 from 10.29.160.220
[splunk@admin1 ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/data/disk1/splunk/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /data/disk1/splunk/.ssh/id_rsa.
Your public key has been saved in /data/disk1/splunk/.ssh/id_rsa.pub.
The key fingerprint is:
f1:2d:02:5e:67:23:99:e7:c0:ab:be:7c:3f:f1:de:4a splunk@admin1
The key's randomart image is:
+--[ RSA 2048 ]-----+
|
| . o
| . O =
| . o @ o
| . S + .
| . . . .
| . . . . oE
| o . . . .
| +o ..oo..
+-----+
[splunk@admin1 ~]$
[splunk@admin1 ~]$
```

4. Then run the following script from the admin node to copy the public key `id_rsa.pub` to all the Splunk servers i.e. indexers, search heads and admins of the cluster. `ssh-copy-id` appends the keys to the remote-host's `.ssh/authorized_key`.

```
for host in admin1 admin2 idx1 idx2 idx3 idx4 idx5 idx6 idx7 idx8 sh1 sh2
sh3; do echo -n "$host -> "; ssh-copy-id -i ~/.ssh/id_rsa.pub $host; done
```

5. Enter yes at the prompt Are you sure you want to continue connecting (yes/no)? Enter the password of the remote host.
6. Verify the password-less login, by entering the following command. The output should display the hostnames of all splunk servers.

```
clush --group=all-splunk hostname
```

```
[splunk@admin1 ~]$ clush --group=all-splunk hostname
admin1: admin1
idx2: idx2
idx4: idx4
admin2: admin2
sh3: sh3
idx3: idx3
idx6: idx6
sh2: sh2
idx5: idx5
idx7: idx7
sh1: sh1
idx1: idx1
idx8: idx8
```

Starting Splunk Enterprise Cluster

1. Log onto the admin node as user splunk.
2. Start the Splunk Enterprise

```
clush --group=all-splunk $SPLUNK_HOME/bin/splunk start --accept-license
```

```
[splunk@admin1 ~]$ clush --group=all-splunk $SPLUNK_HOME/bin/splunk
start --accept-license
```

3. Verify the status of the Splunk Enterprise services.

```
clush --group=all-splunk $SPLUNK_HOME/bin/splunk status
```

```
[splunk@admin1 ~]$ clush --group=all-splunk $SPLUNK_HOME/bin/splunk
status
admin1: splunkd is running (PID: 13005).
admin1: splunk helpers are running (PIDs: 13006 13015 13067 13107).
idc4: splunkd is running (PID: 2738).
idc4: splunk helpers are running (PIDs: 2739 2750 2800 2847).
sh3: splunkd is running (PID: 4358).
sh3: splunk helpers are running (PIDs: 4359 4374 4464 4498).
idc2: splunkd is running (PID: 14682).
idc2: splunk helpers are running (PIDs: 14683 14694 14744 14791).
admin2: splunkd is running (PID: 7362).
admin2: splunk helpers are running (PIDs: 7363 7370 7421 7458).
idc3: splunkd is running (PID: 10697).
idc3: splunk helpers are running (PIDs: 10698 10709 10759 10805).
sh1: splunkd is running (PID: 30426).
sh1: splunk helpers are running (PIDs: 30427 30443 30540 30575).
sh2: splunkd is running (PID: 25615).
sh2: splunk helpers are running (PIDs: 25616 25632 25729 25764).
idc7: splunkd is running (PID: 11041).
idc7: splunk helpers are running (PIDs: 11042 11053 11103 11150).
idc5: splunkd is running (PID: 16593).
idc5: splunk helpers are running (PIDs: 16594 16605 16655 16702).
idc8: splunkd is running (PID: 11233).
idc8: splunk helpers are running (PIDs: 11234 11245 11295 11343).
idc1: splunkd is running (PID: 10453).
idc1: splunk helpers are running (PIDs: 10454 10465 10515 10562).
idc6: splunkd is running (PID: 13262).
idc6: splunk helpers are running (PIDs: 13263 13274 13324 13371).
```

Logging in for the First Time

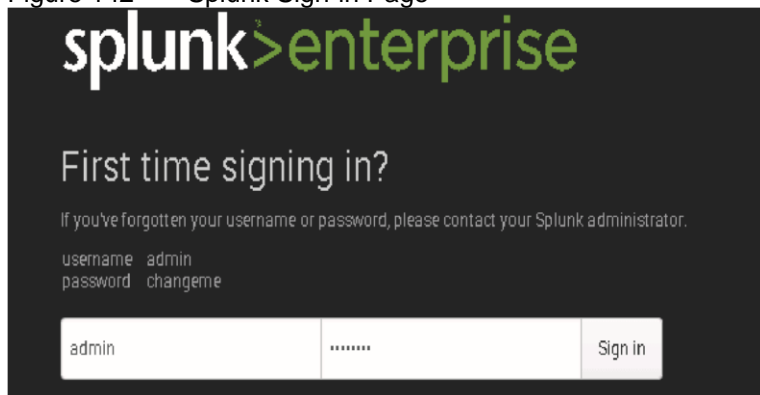
When logging in for the first time, the default password is 'changeme'. The GUI then prompts for the user to change the admin password. This can be completed by logging on to the GUI via every instance:

Log into the admin1 instance. The URL will point to the default port of '8000'. For example <http://admin1:8000>



Note: If you have not added these servers to DNS, you will need to use the IP address, e.g. <http://10.29.160.101:8000/>

Figure 142 Splunk Sign-in Page



In this CVD the password for the Splunk Administrator is set to 'cisco123' (the same as the OS 'splunk' user). You will need to perform this action once on every node via the GUI.

Creating User Accounts

Splunk RPM packages automatically create the user 'splunk' with the home directory of the original installation (for example: /data/disk1/splunk). If an alternative user is created, repeat instructions under one of the previous sections, "Setting Up Login for Splunk User".



Note: The splunk user is installed without a password. A password should be assigned to the user splunk across all the nodes.

Throughout this CVD, the user 'splunk' is utilized to run all Splunk processes. If there is a requirement to run Splunk as a different user, perform the following:

- Export /data/disk1/splunk as \$SPLUNK_HOME, add it to the PATH
- Home Dir. For new users should be Splunk installation directory (/data/disk1/splunk)
- Stop all splunk processes (\$SPLUNK_HOME/bin/splunk stop)
- Chown -R user:usergroup \$SPLUNK_HOME/*
- Change or sudo to new user
- Start all splunk processes (\$SPLUNK_HOME/bin/splunk start)
- When the CVD refers to the user 'splunk', substitute the alternate user.

Initializing Splunk on Boot

Log onto the admin server 'admin1' as root user.

From the command line, launch the following command:

```
clush --group=all-splunk $SPLUNK_HOME/bin/splunk enable boot-start -user splunk
```

This will initialize splunk running as user 'splunk' if any server is rebooted. If the splunk user account is not 'splunk', change the -user reference accordingly.

```
[root@admin1 ~]# clush --group=all-splunk -B $SPLUNK_HOME/bin/splunk
enable boot-start -user splunk
-----
admin[1-2],idx[1-8],sh[1-3] (13)
-----
Init script installed at /etc/init.d/splunk.
Init script is configured to run at boot.
```

Default Ports

The following are the default ports that are used by Splunk software on every node. For more information please refer to [\[Splunk Documentation\]](#).

Table 8 Default Ports used by Splunk

Function	Default Port
Management Port	8089
Web Interface	8000

NFS Configurations for Splunk Frozen Data Storage

Create the User Splunk in the Storage Servers

1. From the node admin1, execute the following command to check the user splunk's user and group identification info.

```
sudo -u splunk id
```

```
[root@admin1 ~]# sudo -u splunk id
uid=500(splunk) gid=500(splunk) groups=500(splunk)
```

2. Take a note of the uid and gid fields output from the command output.



In this case, the splunk user has been created with uid=500 and gid=500

3. Create a group named splunk and user named splunk with matching ids on all the storage nodes.

```
clush --group=splunk -B groupadd --gid=500 splunk
```

```
clush --group=splunk -B useradd --gid=500 --uid=500 splunk
```

```
[root@admin1 ~]# clush --group=splunk -B groupadd --gid=500 splunk
[root@admin1 ~]# clush --group=splunk -B useradd --gid=500 --uid=500 splunk
```



Note: The user splunk gets created without a password. If necessary use the passwd command to assign a password to this user.

NFS Server Setup on Archival Nodes

This section describes the procedure to configure NFS server on the storage servers. As described in section [Configuring Data Drives on Archival Nodes using CIMC](#), each archival node consists of four volumes. They are mounted locally on the archival node i.e. hostname storage1, as /data/disk1, /data/disk2, /data/disk3 and /data/disk4. In each of these volumes a directory tree is created as /splunk/frzn[<indexer1-hostname>, <indexer2-hostname>, ...]. The indexer specific directory under /data/disk[1-4]/splunk/frzn/ is then assigned to the respective indexer, resulting in one or more indexers getting assigned to a volume on the archival nodes depending on the number of indexers and number of storage nodes available in the particular Splunk deployment.

Use the following tables as a guideline to map a volume on a given storage node to one or more indexers. For more information about how Splunk stores and manages the frozen data, see [Configuring Archival of Data From Cold to Frozen](#).

Scenario A: 4 Indexers, 6 Indexers, 8 Indexers (or) 16 Indexers with only One Storage Node

In this scenario, since there is only a single storage node, a sub-directory is created in each of the volumes and assigned to an indexer. The location of the indexer specific sub-directory is determined by the ratio between the numbers of storage nodes and indexers.

Table 9 shows the sample scenarios. There is no hard and fast rule about how to perform this mapping; it can be changed as per the requirements of the particular deployment.

Table 9 Scenario A: Storage Volume Scalability and Mapping Options with One Storage Node

Volume #	Volume Name	1 Storage Node

		4 indexers	6 indexers	8 indexers	16 indexers
1	/data/disk1	idx1	idx1, idx5	idx1, idx2	idx1, idx2, idx3, idx4
2	/data/disk2	idx2	idx2, idx6	idx3, idx4	idx5, idx6, idx7, idx8
3	/data/disk3	idx3	idx3	idx5, idx6	idx9, idx10, idx11, idx12
4	/data/disk4	idx4	idx4	idx7, idx8	idx13, idx14, idx15, idx16

Scenario B: 8 Indexers (or) 16 Indexers with Two Storage Nodes

The following table shows the mapping of eight or sixteen indexers across two storage nodes.

Table 10 Scenario B: Storage Volume Scalability and Mapping Options with Two Storage Nodes

Volume #	Volume Name	8 Indexers		16 Indexers	
		Storage node #1	Storage node #2	Storage node #1	Storage node #2
1	/data/disk1	idx1	idx5	idx1, idx2	idx9, idx10
2	/data/disk2	idx2	idx6	idx3, idx4	idx11, idx12
3	/data/disk3	idx3	idx7	idx5, idx6	idx13, idx14
4	/data/disk4	idx4	idx8	idx7, idx8	idx15, idx16

The script provided in this section creates two sub-directories per disk volume thereby mapping two indexers to a disk volume.

1. Install the NFS tools on the storage servers.

```
clush --group=storage -B yum install -y nfs-*
```

```
[root@admin1 ~]# clush --group=storage -B yum install -y nfs-*
-----
storagel
-----
Loaded plugins: product-id, security, subscription-manager
This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.
Setting up Install Process
Package nfs-utils-lib-1.1.5-6.el6.x86_64 already installed and latest version
Package 1:nfs-utils-1.2.3-39.el6.x86_64 already installed and latest version
Nothing to do
```

2. Create a file by name nfs_server_setup.sh.

```
vi /root/nfs_server_setup.sh
```

3. Copy and paste the following contents and save the file.

```
#!/bin/bash
## Setup NFS Server for Splunk Frozen storage and setup exports
## Stop the NFS Service
service nfs stop
```

```

service rpcbind stop
indexer=1
## Create the NFS Export directories and prepare the /etc/exports file
for diskName in disk1 disk2 disk3 disk4
do
    echo /data/$diskName
    mkdir -p /data/$diskName/splunk/frzn
    ## Exported for Frozen Data from Splunk cluster
    ## Create the root directory for each indexer by their hostnames.
    ## Each volume is dedicated for two indexers.
    mkdir -p /data/$diskName/splunk/frzn/idx$indexer
    ((indexer++))
    mkdir -p /data/$diskName/splunk/frzn/idx$indexer
    ((indexer++))
    chown -R splunk:splunk /data/$diskName/*
    ## Add the directory paths to the NFS Exports file.
    echo "/data/$diskName/splunk/frzn 192.168.11.0/24(rw, sync)" >> /etc/exports
done
## Start the NFS Service in the proper order
service rpcbind start
service nfs start
exit 0

```



Note: This script creates the necessary directories to accommodate eight indexers in the four RAID6 volumes and configures them to be NFS exports. In a deployment scenario where the number of indexers and storage nodes are different from what is described in this CVD, the script will need to be modified accordingly.

4. Change the mode to make it into an executable script.

```
chmod +x /root/nfs_server_setup.sh
```

5. Copy over the script `nfs_server_setup.sh` to all the storage nodes.

```
clush --group=storage -B -c /root/nfs_server_setup.sh
```

6. Execute the `nfs_server_setup.sh` script on all the storage nodes.

```
clush --group=storage -B /root/nfs_server_setup.sh
```

```
[root@admin1 ~]# clush --group=storage -B /root/nfs_server_setup.sh
-----
storagel
-----
Shutting down NFS daemon: [ OK ]
Shutting down NFS mountd: [ OK ]
Shutting down NFS quotas: [ OK ]
Shutting down NFS services: [ OK ]
Shutting down RPC idmapd: [ OK ]
Stopping rpcbind: [ OK ]
/data/disk1
/data/disk2
/data/disk3
/data/disk4
Starting rpcbind: [ OK ]
Starting NFS services: [ OK ]
Starting NFS quotas: [ OK ]
Starting NFS mountd: [ OK ]
Starting NFS daemon: [ OK ]
Starting RPC idmapd: [ OK ]
```



Note: During the shutdown of NFS, daemons may show errors if they weren't previously running. It is normal.



Note: This document assumes that there is only one storage/archival node. The script can be easily modified to accommodate another storage node and to distribute frozen data generated by the indexers between the storage nodes.

7. Check the availability of the mount points from the server admin1.

```
showmount -e storagel
```

```
[root@admin1 ~]# showmount -e storagel
Export list for storagel:
/data/disk4/splunk/frzn 192.168.11.0/24
/data/disk3/splunk/frzn 192.168.11.0/24
/data/disk2/splunk/frzn 192.168.11.0/24
/data/disk1/splunk/frzn 192.168.11.0/24
```



Note: Even though eight partitions have been exported from the storage node, this solution makes use of only the exports named as /data/disk[1-4]/splunk/frzn. Refer to [Splunk](#)

For more information how Splunk indexer stores indexes and the aging policy go to: <http://docs.splunk.com/Documentation/Splunk/latest/Indexer/HowSplunkstoresindexes>.

NFS Client Configurations on the Indexers

This section describes the procedure to configure NFS clients on the indexer servers. One mount point on the server will be shared between two indexers. But, the indexers shall be configured to make use of their own respective root directory mounted in their own file system as /data/frzn_data. It is a recommended best practice to perform "hard" NFS mount without attribute caching.

1. Install the NFS tools on all the indexers.

```
clush --group=indexers -B yum install -y nfs-*
```

```
[root@admin1 ~]# clush --group=indexers -B yum install -y nfs-*
-----
idx[1-8] (8)
-----
Loaded plugins: product-id, security, subscription-manager
This system is not registered to Red Hat Subscription Management. You can use subscrip
tion-manager to register.
Setting up Install Process
Package nfs-utils-lib-1.1.5-6.el6.x86_64 already installed and latest version
Package 1:nfs-utils-1.2.3-39.el6.x86_64 already installed and latest version
Nothing to do
```

2. Create a file by name `nfs_client_setup.sh`.

```
vi /root/nfs_client_setup.sh
```

3. Copy and paste the following contents and save the file.

```
#!/bin/bash
## NFS Clientside configurations for the Indexers
## Create a temporary script directory for holding auto generated scripts.
CISCO_SCRIPT_DIR=/root/cisco/scripts
mkdir -p $CISCO_SCRIPT_DIR
rm -rf $CISCO_SCRIPT_DIR/*
clush --group=indexers -B mkdir -p $CISCO_SCRIPT_DIR
clush --group=indexers -B rm -rf $CISCO_SCRIPT_DIR/*
## Create the mount points on the indexers
clush --group=indexers -B mkdir -p /mnt/frzn
## Create the RAID6 volume to indexer-set map.
declare -A diskMap=( [disk1]=idx1,idx2 [disk2]=idx3,idx4 [disk3]=idx5,idx6
[disk4]=idx7,idx8 )
FRZN_LINK_NAME="frzn_data"
for K in "${!diskMap[@]}"
do
    echo $K --- ${diskMap[$K]}

optParam="nolock,tcp,rw,hard,intr,timeo=600,retrans=2,rsize=131072,wsiz=13107
2"
    fstabEntry="storage1:/data/$K/splunk/frzn /mnt/frzn nfs $optParam"
    mountParam="-t nfs storage1:/data/$K/splunk/frzn /mnt/frzn -o $optParam"
    echo "mount $mountParam" > $CISCO_SCRIPT_DIR/mount_script_$K.sh
    echo "echo $fstabEntry >> /etc/fstab" >> $CISCO_SCRIPT_DIR/mount_script_$K.sh
    chmod +x $CISCO_SCRIPT_DIR/mount_script_$K.sh
    clush -w ${diskMap[$K]} -B -c $CISCO_SCRIPT_DIR/mount_script_$K.sh
    clush -w ${diskMap[$K]} $CISCO_SCRIPT_DIR/mount_script_$K.sh
done
echo "echo removing old mapping /data/$FRZN_LINK_NAME" >
    $CISCO_SCRIPT_DIR/frzn_link_setup.sh
echo "rm -f /data/$FRZN_LINK_NAME" >> $CISCO_SCRIPT_DIR/frzn_link_setup.sh
echo "ln -s /mnt/frzn/\$(hostname) /data/$FRZN_LINK_NAME" >>
    $CISCO_SCRIPT_DIR/frzn_link_setup.sh
chmod +x $CISCO_SCRIPT_DIR/frzn_link_setup.sh
clush --group=indexers -B -c $CISCO_SCRIPT_DIR/frzn_link_setup.sh
clush --group=indexers -B $CISCO_SCRIPT_DIR/frzn_link_setup.sh
exit 0
```

4. Change the mode to make it into an executable script.

```
chmod +x /root/nfs_client_setup.sh
```

5. Execute the script:

```
./nfs_client_setup.sh
```

```
[root@admin1 ~]# ./nfs_client_setup.sh
disk4 --- idx7,idx8
disk1 --- idx1,idx2
disk3 --- idx5,idx6
disk2 --- idx3,idx4
-----
idx[1-8] (8)
-----
removing old mapping /data/frzn_data
```



Note: If the mount point on the indexers i.e. /mnt/frzn was previously used, it needs to be un-mounted prior to executing the above script. In such a case, use the command `clush --group=indexers -B umount /mnt/frzn`.

6. Verify the NFS setup in all the indexers.

```
clush --group=indexers -B "mount -l | grep splunk"
```

```
clush --group=indexers -B ls -l /data
```

```
[root@admin1 ~]# clush --group=indexers -B "mount -l | grep splunk"

-----
idx1
-----
storage1:/data/disk1/splunk/active on /mnt/frzn type nfs (rw,nolock,tcp,hard,intr,t
imeo=600,retrans=2,rsiz=131072,wsiz=131072,vers=4,addr=192.168.11.114,clientaddr=
192.168.11.103)
-----
idx2
-----
storage1:/data/disk1/splunk/active on /mnt/frzn type nfs (rw,nolock,tcp,hard,intr,t
imeo=600,retrans=2,rsiz=131072,wsiz=131072,vers=4,addr=192.168.11.114,clientaddr=
192.168.11.104)
-----
idx3
-----
storage1:/data/disk2/splunk/active on /mnt/frzn type nfs (rw,nolock,tcp,hard,intr,t
imeo=600,retrans=2,rsiz=131072,wsiz=131072,vers=4,addr=192.168.11.114,clientaddr=
192.168.11.105)
-----
idx4
-----
storage1:/data/disk2/splunk/active on /mnt/frzn type nfs (rw,nolock,tcp,hard,intr,t
imeo=600,retrans=2,rsiz=131072,wsiz=131072,vers=4,addr=192.168.11.114,clientaddr=
192.168.11.106)
-----
idx5
-----
storage1:/data/disk3/splunk/active on /mnt/frzn type nfs (rw,nolock,tcp,hard,intr,t
imeo=600,retrans=2,rsiz=131072,wsiz=131072,vers=4,addr=192.168.11.114,clientaddr=
192.168.11.107)
-----
idx6
-----
storage1:/data/disk3/splunk/active on /mnt/frzn type nfs (rw,nolock,tcp,hard,intr,t
imeo=600,retrans=2,rsiz=131072,wsiz=131072,vers=4,addr=192.168.11.114,clientaddr=
192.168.11.108)
-----
idx7
-----
storage1:/data/disk4/splunk/active on /mnt/frzn type nfs (rw,nolock,tcp,hard,intr,t
imeo=600,retrans=2,rsiz=131072,wsiz=131072,vers=4,addr=192.168.11.114,clientaddr=
192.168.11.109)
-----
idx8
-----
storage1:/data/disk4/splunk/active on /mnt/frzn type nfs (rw,nolock,tcp,hard,intr,t
imeo=600,retrans=2,rsiz=131072,wsiz=131072,vers=4,addr=192.168.11.114,clientaddr=
192.168.11.110)
```



```

[root@admin1 ~]# clush --group=indexers -B ls -l /data
-----
idx1
-----
total 0
drwxr-xr-x. 3 root root 19 Apr  9 10:33 disk1
lrwxrwxrwx. 1 root root 14 Apr  9 15:37 frzn_data -> /mnt/frzn/idx1
-----
idx2
-----
total 0
drwxr-xr-x 3 root root 19 Apr  9 10:33 disk1
lrwxrwxrwx 1 root root 14 Apr  9 15:37 frzn_data -> /mnt/frzn/idx2
-----
idx3
-----
total 0
drwxr-xr-x. 3 root root 19 Apr  9 10:33 disk1
lrwxrwxrwx. 1 root root 14 Apr  9 15:37 frzn_data -> /mnt/frzn/idx3
-----
idx4
-----
total 4
drwxr-xr-x 3 root root 4096 Apr  9 10:33 disk1
lrwxrwxrwx 1 root root  14 Apr  9 15:37 frzn_data -> /mnt/frzn/idx4
-----
idx5
-----
total 0
drwxr-xr-x. 3 root root 19 Apr  9 10:33 disk1
lrwxrwxrwx. 1 root root 14 Apr  9 15:37 frzn_data -> /mnt/frzn/idx5
-----
idx6
-----
total 0
drwxr-xr-x. 3 root root 19 Apr  9 10:33 disk1
lrwxrwxrwx. 1 root root 14 Apr  9 15:37 frzn_data -> /mnt/frzn/idx6
-----
idx7
-----
total 0
drwxr-xr-x. 3 root root 19 Apr  9 10:33 disk1
lrwxrwxrwx. 1 root root 14 Apr  9 15:37 frzn_data -> /mnt/frzn/idx7
-----
idx8
-----
total 0
drwxr-xr-x. 3 root root 19 Apr  9 10:33 disk1
lrwxrwxrwx. 1 root root 14 Apr  9 15:37 frzn_data -> /mnt/frzn/idx8

```

Configuring the Splunk Enterprise Cluster

Configuring Splunk Enterprise Licenses

The servers in the Splunk Enterprise infrastructure that performs indexing must be licensed. Any Splunk instance can be configured to perform the role of license master. In this CVD, the admin node (admin1) is configured to be the license master and all the other Splunk instances are configured as license slaves.

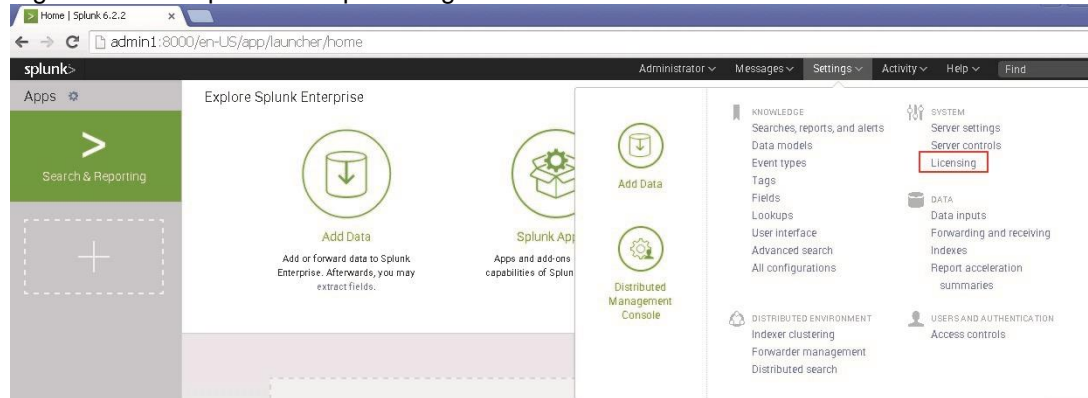
Setup License Master

Configure the server admin1 as the central license master by following the procedures detailed below.

1. Log onto the server admin1 as user admin.

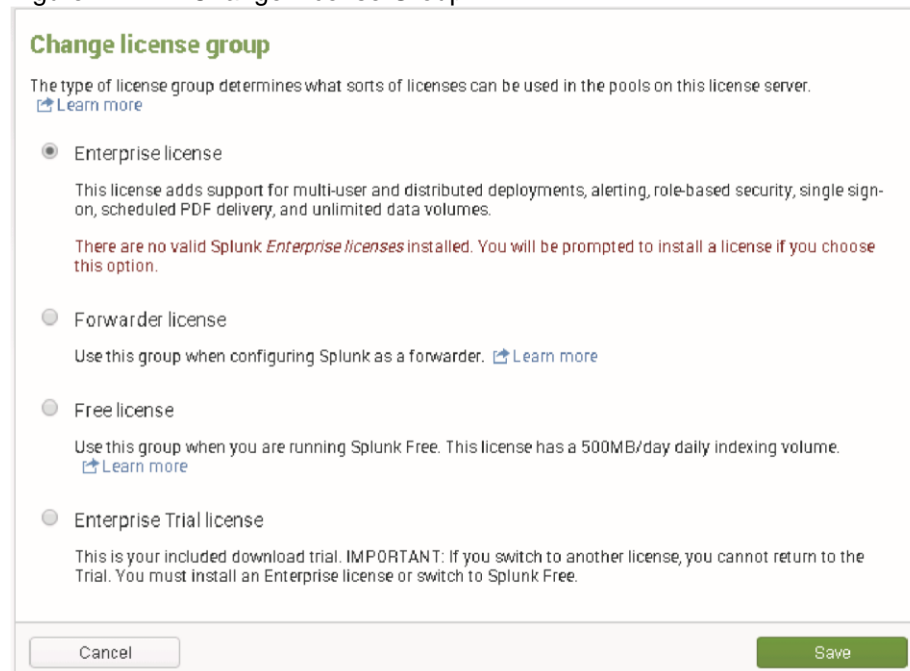
2. Navigate to the licensing screen by clicking on Settings → Licensing. (Figure 143)

Figure 143 Splunk Enterprise Page



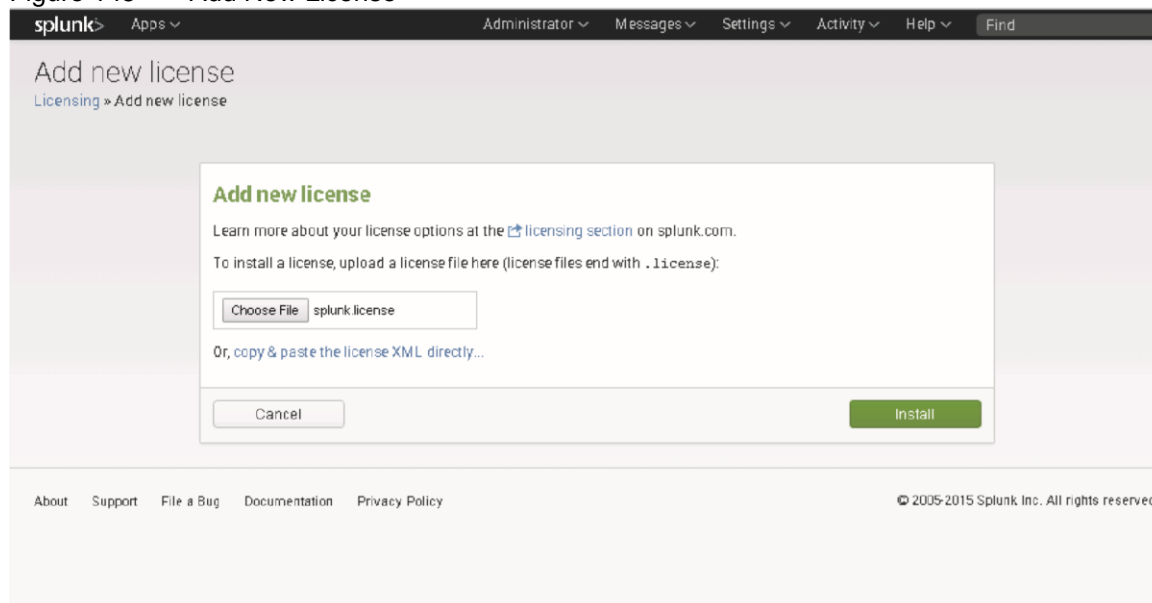
3. Click on Change License Group. (Figure 144)
4. Click on the Enterprise License radio button.

Figure 144 Change License Group



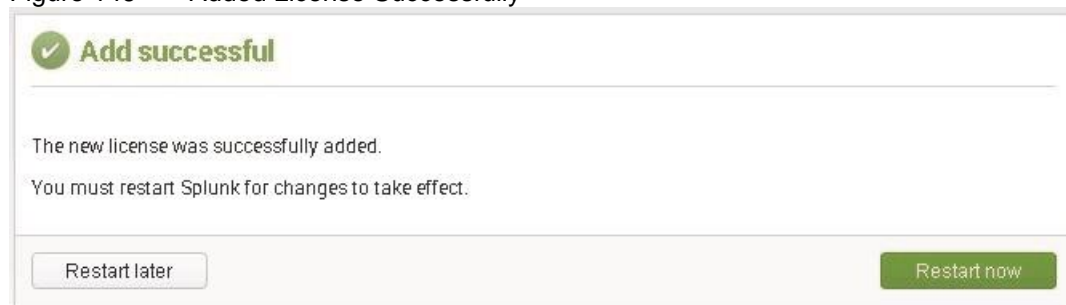
5. Click on Save.
6. In the Add new license dialog, click on Choose File to select your license file.
7. Click Install to install the license. (Figure 145)

Figure 145 Add New License



8. Click on `Restart now`.
9. Click `OK` to restart Splunk to complete the license installation.

Figure 146 Added License Successfully



10. Log back in to Splunk (if “are you sure you want to restart Splunk”) is still visible, click `Cancel`

For more information about Splunk Enterprise licensing, please refer to [Splunk Documentation](#).

Configure the Indexers, Search Heads, and Admin Nodes as License Slaves

Configure all the other Splunk instances to be the license slaves to the Splunk license master, i.e. `admin1`. This can be performed by following one of the two methods described below.

The first and preferred method is to use Cluster-Shell command (`clush`) to configure all the Splunk instances to be license slaves to the license master in `admin1`. The second (optional) method is to configure each node as a license slave individually by accessing the respective Web UI.

Configure all the License Slaves at Once Using CLI (`clush`)

1. From the admin node (`admin1`) as user ‘splunk’ execute the command:

```
clush --group=all-splunk -x admin1 -B $SPLUNK_HOME/bin/splunk edit licenser-
localslave -master_uri https://admin1:8089 -auth admin:cisco123
```

```
[splunk@admin1 ~]$ clush --group=all-splunk -x admin1 -B $SPLUNK_HOM
E/bin/splunk edit licenser-localslave -master_uri https://admin1:808
9 -auth admin:cisco123
-----
admin2,idx[1-8],sh[1-3] (12)
-----
The licenser-localslave object has been edited.
```

2. Next issue the command:

```
clush --group=all-splunk -x admin1 $SPLUNK_HOME/bin/splunk restart
```

```
[splunk@admin1 ~]$ clush --group=all-splunk -x admin1 $SPLUNK_HOME/b
in/splunk restart
```

This will restart all nodes except for admin1 and storage1 nodes.

3. During restart, you will receive confirmation that the instances are running as license-slaves.

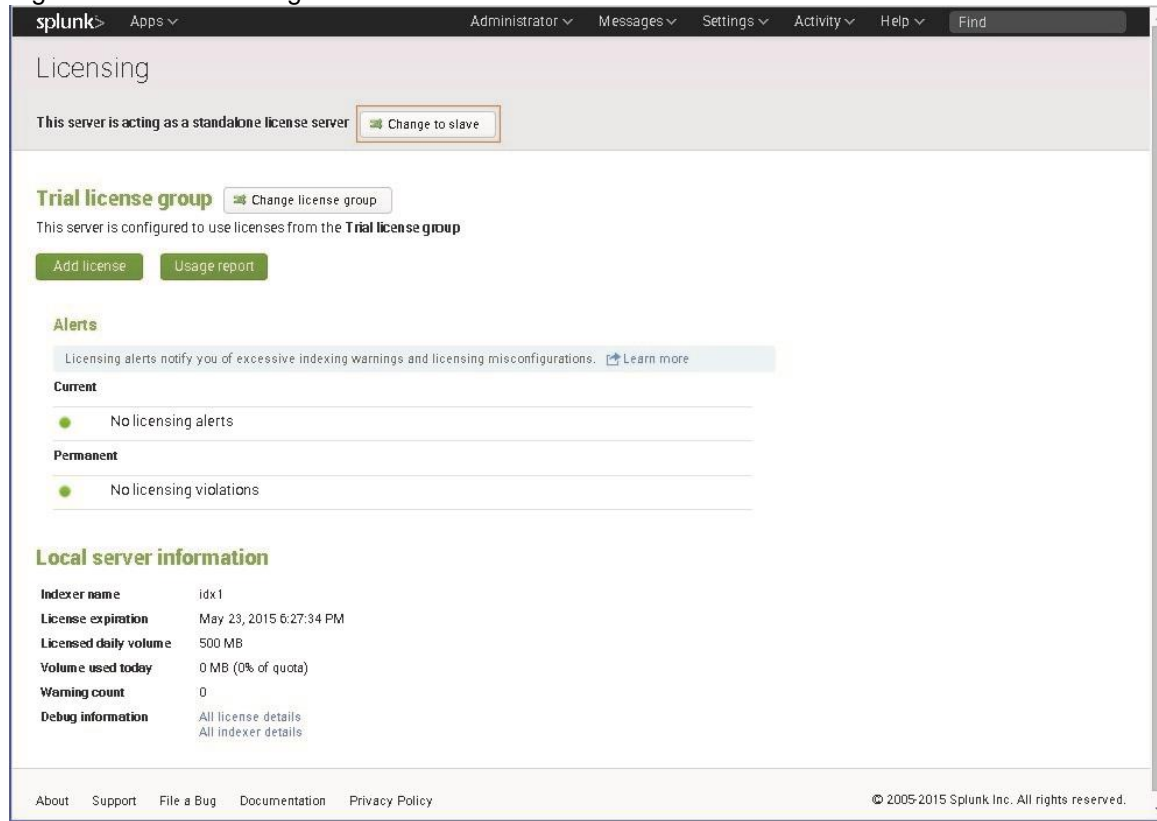
```
dx6: Checking kvstore port [8191]: open
dx7: Checking kvstore port [8191]: open
dx8: Checking kvstore port [8191]: open
dx2: Checking configuration... Done.
dx2: Checking critical directories... Done
dx2: Checking indexes...
dx2: Validated: _audit _blocksignature _internal _introspection _thefishbucket history
main summary
dx2: Done
dx2:
dx2: Bypassing local license checks since this instance is configured with a remote license maste
dx2:
dx2: Checking configuration... Done
```

4. Proceed to 'Verifying License-Slave Relationships'.

(Optional) Configure License Slaves Individually Using the Web Interface

1. Log onto an indexer server i.e. idx1 as user admin. (e.g. https://idx1:8000)
2. Navigate to the licensing screen by clicking on Settings → Licensing. (Figure 147)
3. Click on the button Change to slave.

Figure 147 Licensing Screen on an Indexer



4. In the Change master association dialog, click on Designate a different Splunk instance as the master license server radio button. (Figure 148)
5. Enter the Master license server URI in the format <https://<IP-or-hostname>:8089>. (e.g. <https://admin1:8089>)



Note: The port 8089 is the management port chosen while the server admin1 was provisioned as the master node.

6. Click Save.

Figure 148 Configure the Indexer to Choose Admin1 as its License Master

Change master association

This server, **idx1**, is currently acting as a master license server.

Designate this Splunk instance, **idx1**, as the master license server

Choosing this option will:

- Point the local indexer at the local master license server
- Disconnect the local indexer from any remote license server

Designate a different Splunk instance as the master license server

Choosing this option will:

- Deactivate the local master license server
- Point the local indexer at license server specified below
- Discontinue license services to remote indexers currently pointing to this server

Master license server URI

https://admin1:8089|

For example: https://splunk_license_server:8089
Use https and specify the management port.

Cancel
Save

7. Restart Splunk by clicking on `Restart now`.

Figure 149 Successfully Changed Master Association

✓

Change successful

The master server was successfully changed.

Restart later
Restart now

Repeat the steps above to configure all eight indexers, all three search heads, and the second admin node (admin2), to become license slaves to the license master on the server admin1.

Verifying License-Slave Relationships

Confirm the license configurations.

1. Go to the master node's Splunk GUI, and navigate to `Settings > Licensing`.
2. At the bottom of this screen, click `All indexer Details` to view the license slaves. (Figure 150)

Figure 150 Verifying Indexer Details

Local server information

Indexer name	admin1
Volume used today	0 MB
Warning count	0
Debug information	All license details All indexer details

There should be thirteen license slaves listed: eight indexers, three search heads, and two admin nodes.

Figure 151 Indexer Details

Indexers connected to: admin1 (13)

- 1. idx2**

active_pool_names	• auto_generated_pool_enterprise
added_usage_parsing_warnings	None
label	idx2
pool_names	<ul style="list-style-type: none"> • auto_generated_pool_download-trial • auto_generated_pool_enterprise • auto_generated_pool_forwarder • auto_generated_pool_free
stack_names	<ul style="list-style-type: none"> • download-trial • enterprise • forwarder • free
warning_count	0

- 2. idx8**

active_pool_names	• auto_generated_pool_enterprise
added_usage_parsing_warnings	None
label	idx8
pool_names	<ul style="list-style-type: none"> • auto_generated_pool_download-trial • auto_generated_pool_enterprise • auto_generated_pool_forwarder • auto_generated_pool_free
stack_names	<ul style="list-style-type: none"> • download-trial • enterprise • forwarder • free
warning_count	0

- 3. idx1**

active_pool_names	• auto_generated_pool_enterprise
--------------------------	----------------------------------



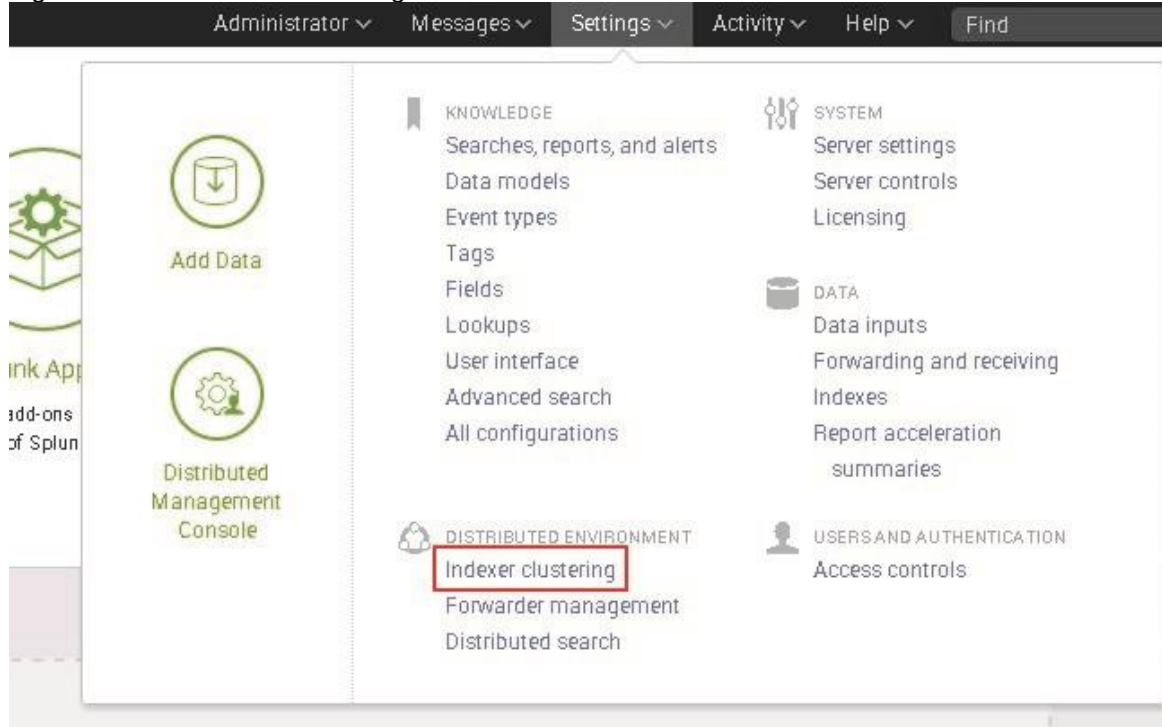
Note: The License Master counts all the license slaves as Splunk Indexer instances in spite of the actual roles the instances have been configured to perform.

Configuring the Master Node (aka: Cluster Master)

To start, configure the admin node admin1 as the Indexer Cluster master.

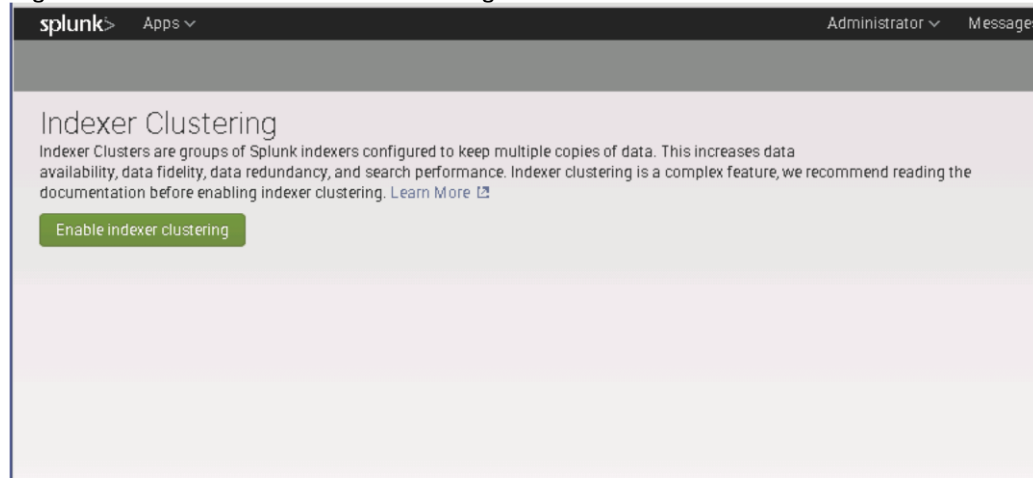
1. Using your browser, go to the master node (admin1) `http://hostname-or-IP:8000/` (for example, `https://admin1:8000/`)
2. Click on the `Settings > Indexer Clustering`. (Figure 152)

Figure 152 Indexer Clustering



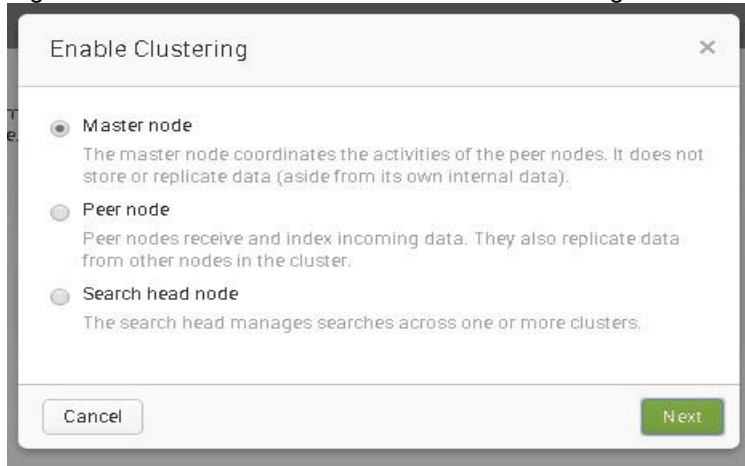
3. Click `Enable Indexer Clustering`. (Figure 153)

Figure 153 Enable Indexer Clustering



4. Click the `Master Node` radio button, then click `Next`. (Figure 154)

Figure 154 Select the Node to Enable Clustering



Enable Clustering

Master node
The master node coordinates the activities of the peer nodes. It does not store or replicate data (aside from its own internal data).

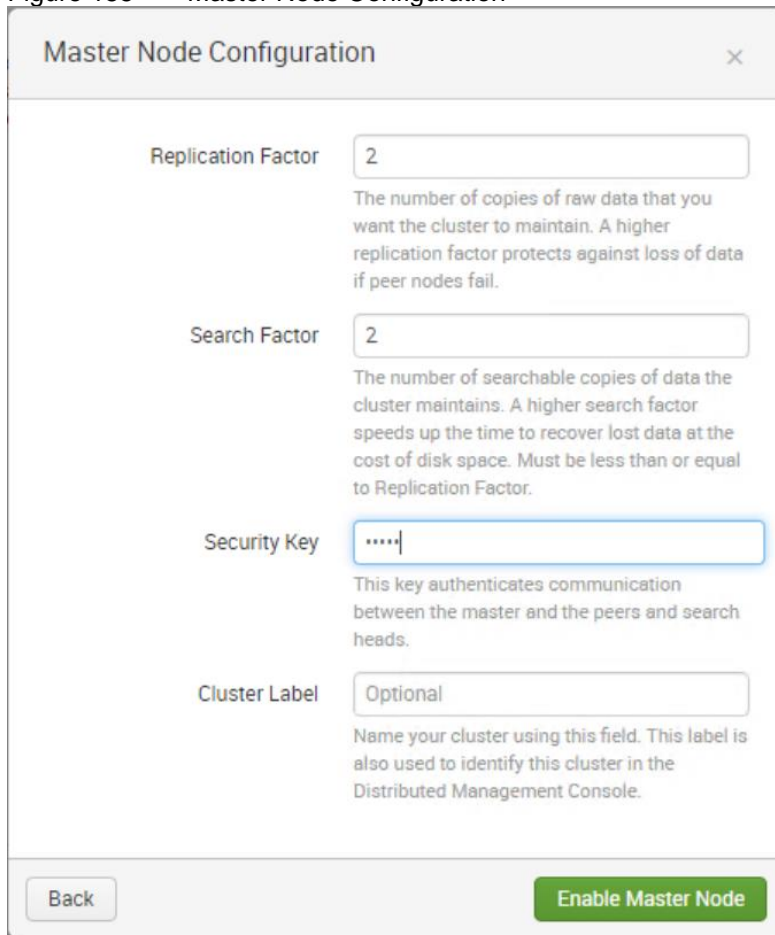
Peer node
Peer nodes receive and index incoming data. They also replicate data from other nodes in the cluster.

Search head node
The search head manages searches across one or more clusters.

Cancel Next

5. Set the Replication Factor field to 2, and the Search Factor field to 2.
6. Set up a Security Key; in this installation 'splunk+cisco' was used as the security key.
7. Click Enable Master Node.

Figure 155 Master Node Configuration



Master Node Configuration

Replication Factor
The number of copies of raw data that you want the cluster to maintain. A higher replication factor protects against loss of data if peer nodes fail.

Search Factor
The number of searchable copies of data the cluster maintains. A higher search factor speeds up the time to recover lost data at the cost of disk space. Must be less than or equal to Replication Factor.

Security Key
This key authenticates communication between the master and the peers and search heads.

Cluster Label
Name your cluster using this field. This label is also used to identify this cluster in the Distributed Management Console.

Back Enable Master Node



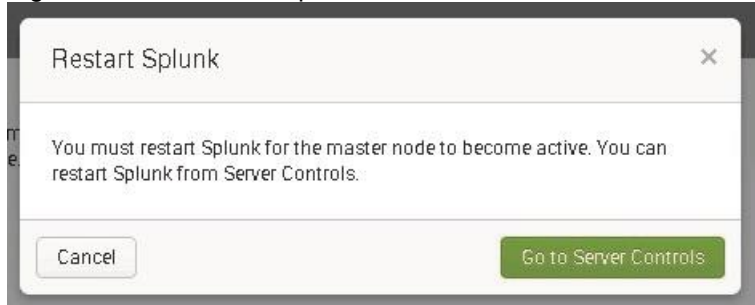
Note: Replication and search factors vary by deployment. The replication factor indicates the number of copies to be maintained on the indexers. The search factor indicates how many of those copies will return search results. In the configuration above, one indexer could be down and searches will still return all results. If the configuration needs to be more resilient, the replication factor may be increased, but will also increase disk consumption. Consult the documentation for more information.
<http://docs.splunk.com/Documentation/Splunk/6.4.1/Indexer/Thereplicationfactor>



Note: Make sure to apply a Security key.

- Click on `Go to Server Controls` to proceed with restarting Splunk as indicated. (Figure 156)

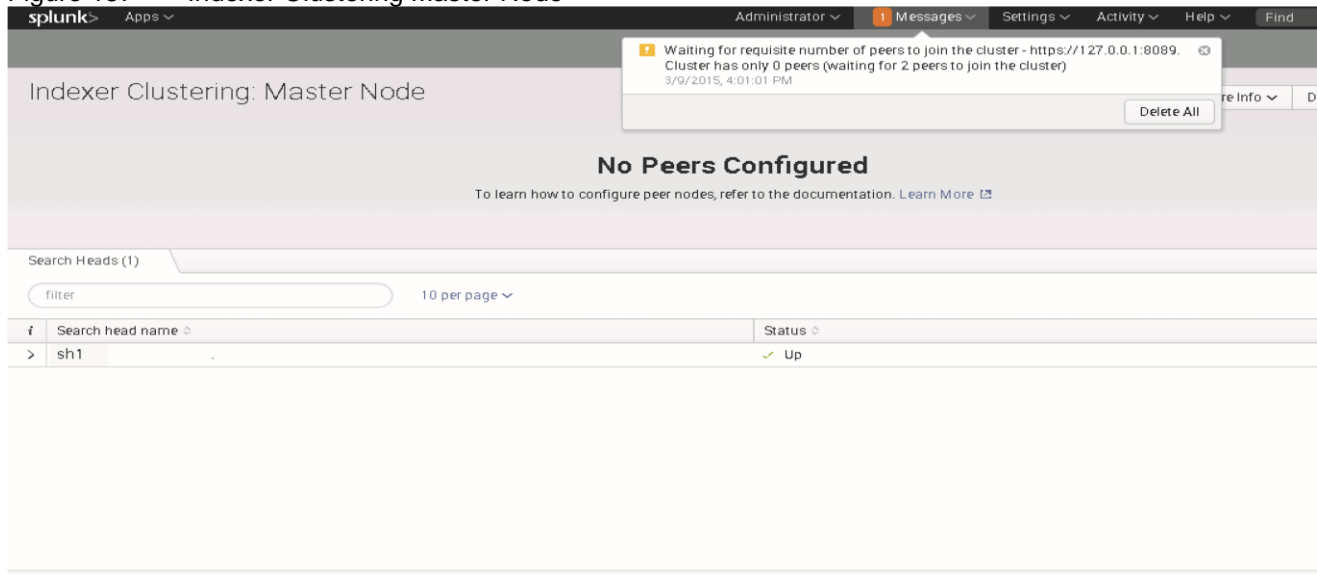
Figure 156 Restart Splunk to Make the Master Node Active



- Restart Successful message appears. Click `OK` to go back to the Login screen.
- Log back in as the “admin” user.
- Return to `Settings > Indexer Clustering`.

A message appears indicating that the necessary number of peers must join the cluster. For a replication factor of 2, Splunk Enterprise needs a minimum of 2 peers. (Figure 157)

Figure 157 Indexer Clustering Master Node



Configure Indexing Peers

Configure all the Splunk instances to be the Indexing Peers to the Master Node, admin1. This can be performed by following one of the two methods described below.

The first and preferred method is to use Cluster-Shell command (clush) to configure all the C240 M4 servers to be indexing peers to the cluster master in admin1. The second (optional) method is to configure each C240 M4 server as an indexing peer individually by accessing the respective Web UI.

Configuring Indexer Clusters

An indexer cluster is a group of Splunk Enterprise instances, or nodes, that, working in concert, provide a redundant indexing and searching capability. The parts of an indexer cluster are:

- A single master node to manage the cluster
- A number of peer nodes to index and maintain multiple copies of the data and to search the data.
- One or more search heads to coordinate searches across the set of peer nodes

The Splunk Enterprise indexers of an indexer cluster are configured to replicate each other's data, so that the system keeps multiple copies of all data. This process is known as index replication. The number of copies is controlled by a parameter known as replication factor. By maintaining multiple, identical copies of Splunk Enterprise data, clusters prevent data loss while promoting data availability for searching.

Indexer clusters feature automatic failover from one indexer to the next. This means that, if one or more indexers fail, incoming data continues to get indexed and indexed data continues to be searchable.

For more information please refer to [\[Splunk Documentation\]](#).

Configure All Indexing Peers Using CLI (clush)

1. From the admin1 box, as the 'splunk' user, issue the command:

```
clush --group=indexers $SPLUNK_HOME/bin/splunk edit cluster-config -mode slave
-master_uri https://admin1:8089 -replication_port 8080 -secret splunk+cisco -
auth admin:cisco123
```

```
[splunk@admin1 ~]$ clush --group=indexers $SPLUNK_HOME/bin/splunk ed
it cluster-config -mode slave -master_uri https://admin1:8089 -repli
cation_port 8080 -secret splunk+cisco -auth admin:cisco123
idx4: The cluster-config property has been edited.
idx2: The cluster-config property has been edited.
idx6: The cluster-config property has been edited.
idx8: The cluster-config property has been edited.
idx7: The cluster-config property has been edited.
idx5: The cluster-config property has been edited.
idx3: The cluster-config property has been edited.
idx1: The cluster-config property has been edited.
```

2. After editing the cluster configuration, the affected boxes must be restarted.

```
clush --group=indexers $SPLUNK_HOME/bin/splunk restart
```

3. After all the splunk process in peer nodes are restarted, check the Master node's (admin1) web UI. The Master node must report number of available peers. (Figure 158)

Figure 158 Available Peers in the Master Node

f	Peer Name	Fully Searchable	Status	Buckets
>	idx6	✓ Yes	Up	148
>	idx7	✓ Yes	Up	111
>	idx5	✓ Yes	Up	93
>	idx2	✓ Yes	Up	57
>	idx3	✓ Yes	Up	61
>	idx8	✓ Yes	Up	145
>	idx4	✓ Yes	Up	78
>	idx1	✓ Yes	Up	45

4. Proceed to Setting Dedicated Replication Address.



Note: Once the indexers are added to the cluster, it is not advised to use the command `'$SPLUNK_HOME/bin/splunk restart'` on individual indexers. For further information, see: <http://docs.splunk.com/Documentation/Splunk/latest/Indexer/Restartthecluster>

Configure Indexing Peers Individually Using the Web Interface (Optional)

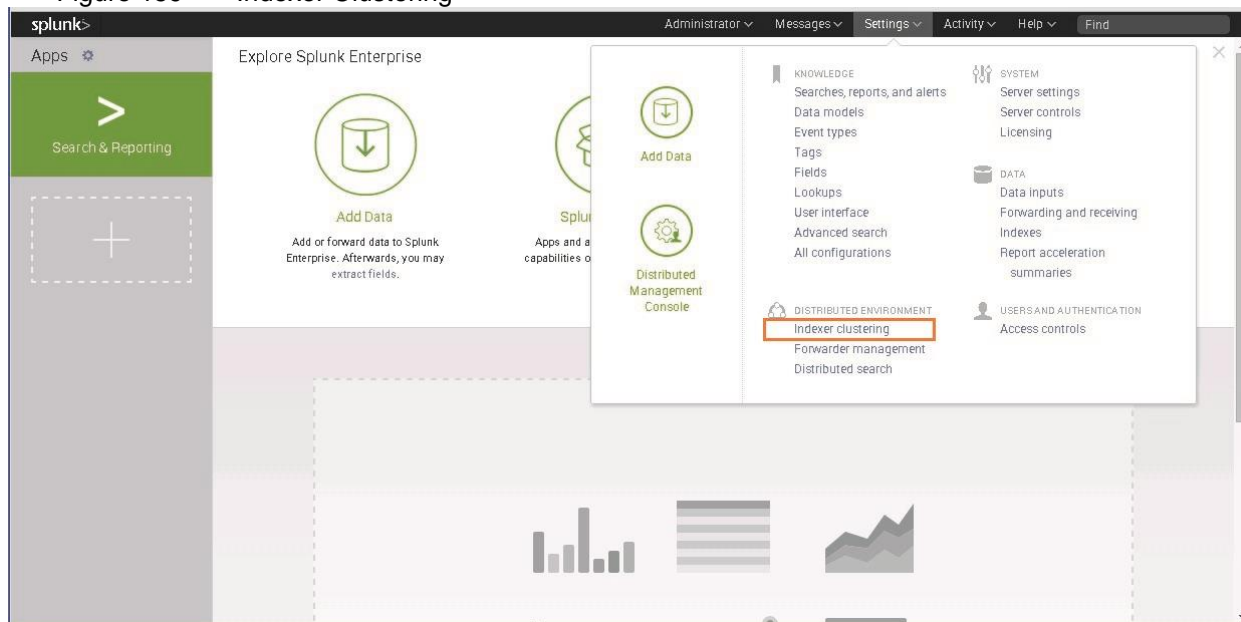


Note: This is an optional method that may be followed to configure each indexer manually through the Splunk Web-UI. The preferred method is to perform the configuration via CLI as shown in the previous section. See “Configure All Indexing Peers Using CLI (clush)” procedure on page 212.

To enable an indexer as a peer node, complete the following steps:

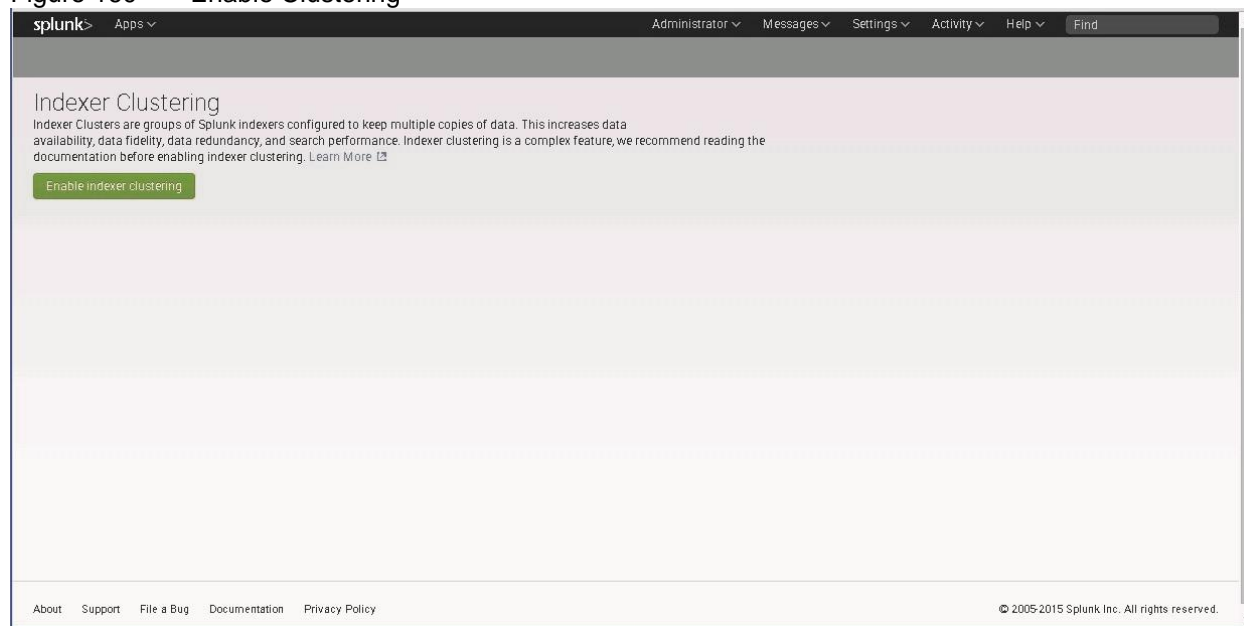
1. Go to an Indexer node's Splunk Web-UI – `http://idx1:8000/`
2. Login as “admin” user with password “cisco123”.
3. Click `Settings` in the upper right corner of Splunk Web.
4. In the `Distributed` environment group, click `Indexer Clustering`.

Figure 159 Indexer Clustering



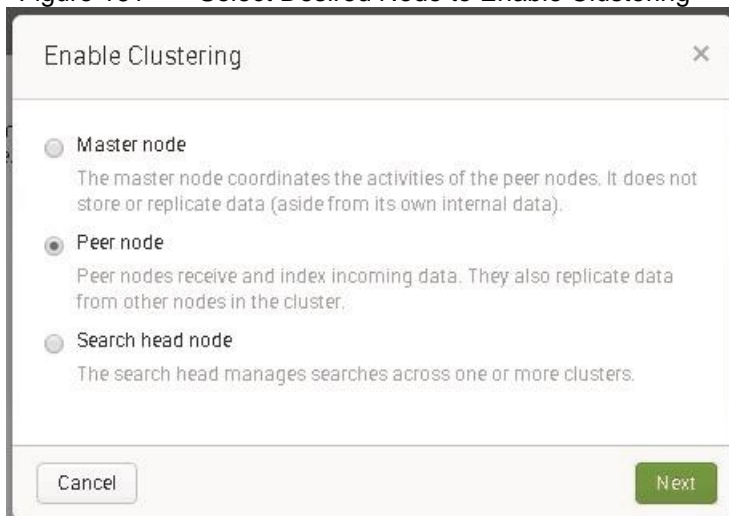
5. Select **Enable indexer clustering**.

Figure 160 Enable Clustering



6. Select **Peer** node and click **Next**. (Figure 161)

Figure 161 Select Desired Node to Enable Clustering



7. Complete entries for the following fields:

- a. Master IP address or Hostname. Enter the master's IP address or hostname. For example: `https://admin1`
- b. Master port. Enter the master's management port. For example: 8089.
- c. Peer replication port. This is the port on which the peer receives replicated data streamed from the other peers. You can specify any available, unused port for this purpose. This port must be different from the management or receiving ports.
- d. Security key. This is the key that authenticates communication between the master and the peers and search heads. The key must be the same across all cluster instances. If the master has a security key, you must enter it here.

8. Click `Enable peer node`. (Figure 162)

Figure 162 Enable Peer Node

Peer node configuration

Master IP address or Hostname:
E.g. https://10.152.31.202

Master port:
E.g. 8089

Peer replication port:
The port peer nodes use to stream data to each other (Eg: 8080).

Security key:
This key authenticates communication between the master and the peers and search heads.

Back Enable peer node

9. The message appears: "You must restart Splunk for the peer node to become active."

Figure 163 Restart Splunk for the Peer Node to get Active

Restart Splunk

You must restart Splunk for the peer node to become active.
Optional next steps after restart:

1. Configure the indexes for the peers.
The index file determines the peers set of indexes and the size and attributes of its buckets. This file must be identical across all peer nodes. Peer index files are edited and distributed from the Master Node. [Learn More](#)
2. Use forwarders to get data to this peer.
There are two reasons for using forwarders to send data to your cluster.
1. To ensure that all incoming data gets indexed. 2. To handle potential node failure. [Learn More](#)

You can restart Splunk from Server Controls.

Cancel Go to Server Controls

10. Click **Go to Server Controls**. This will take you to the **Settings** page where you can initiate the restart.



Note: The figures below show the Splunk restart process on indexer idx1 (i.e. 10.29.160.103).

Figure 164 Restart Splunk in Server Control Setting Page

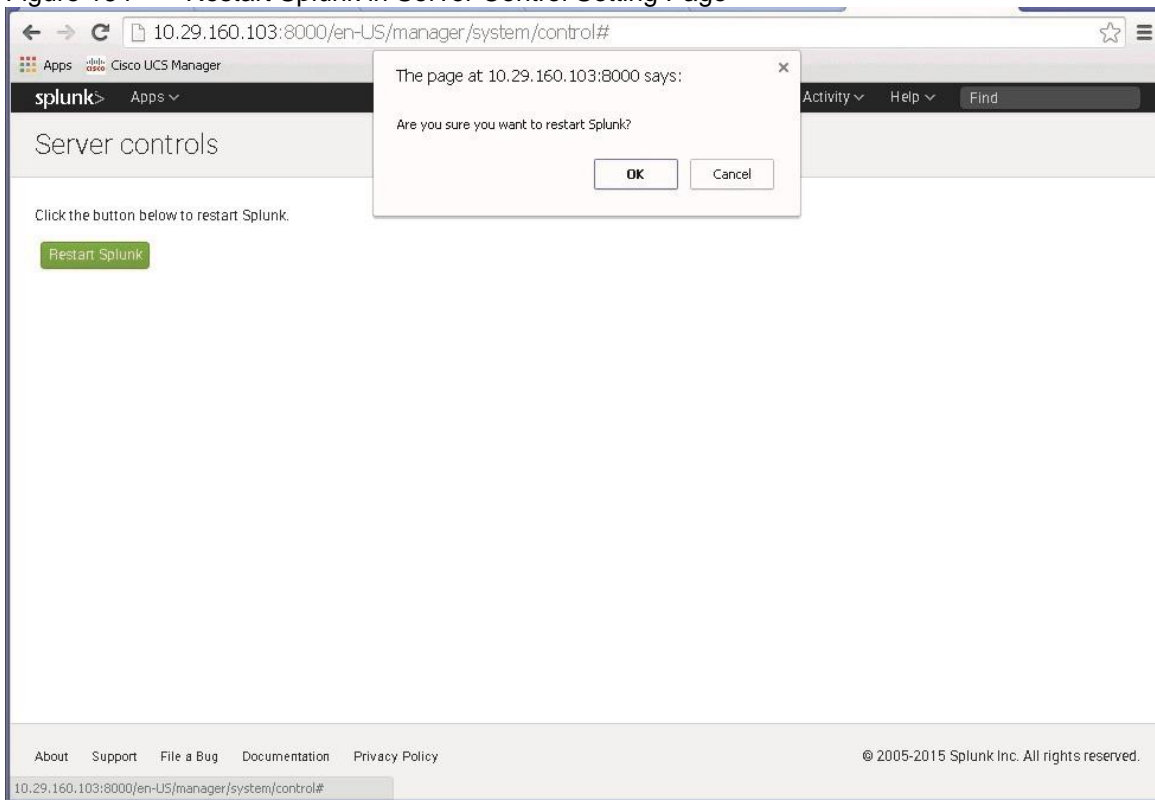
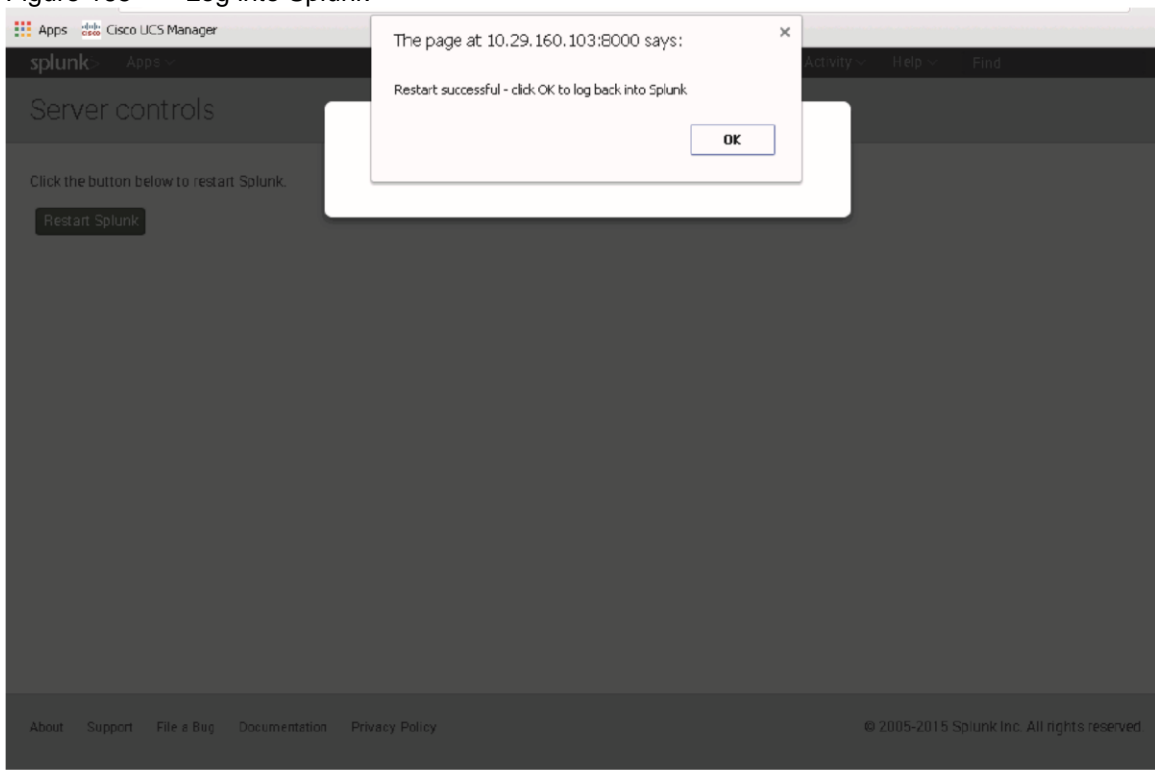


Figure 165 Log into Splunk



11. After the peer node restarts, check the Master node's web UI. The Master node must report number of available peers.

Figure 166 Master Node with Available Peers

i	Peer Name	Fully Searchable	Status	Buckets
>	idx6	✓ Yes	Up	148
>	idx7	✓ Yes	Up	111
>	idx5	✓ Yes	Up	93
>	idx2	✓ Yes	Up	57
>	idx3	✓ Yes	Up	51
>	idx8	✓ Yes	Up	145
>	idx4	✓ Yes	Up	78
>	idx1	✓ Yes	Up	45

- Repeat this process for all the cluster's peer nodes (indexers). When complete, the screen should report 8 indexers as reflected in figure above.



Note: Once the indexers are added to the cluster, it is not advised to use the command **'\$SPLUNK_HOME/bin/splunk restart'** on individual indexers. For further information, see: <http://docs.splunk.com/Documentation/Splunk/latest/Indexer/Restartthecluster>.

Setting Dedicated Replication Address

Splunk Enterprise provides a way to make use of a dedicated network interface for index replication data traffic that happens between the indexers in Splunk Enterprise Indexer cluster. In this CVD, the eth2 with an IP address in the range 192.168.12.0/24 is utilized for this purpose. This feature is configured in the server.conf file on each Splunk Enterprise indexer instance by setting the **register_replication_address** property. This property can be configured with an IP address or a resolvable hostname.

- SSH to idx1.
- As the splunk user, edit the file `$SPLUNK_HOME/etc/system/local/server.conf`
- Under the section `[clustering]`, include the line:

```
register_replication_address=idx1-rep
```

```
[clustering]
master_uri = https://admin1:8089
mode = slave
pass4SymmKey = $1$7E/toE/2WNC0yGwKeQ==
register replication address = idx1-rep
```



Note: It is important to make sure that the host name i.e. idx1-rep or IP address used when setting the `register_replication_address` field is local to the server on which the server.conf resides. The value entered must reflect the replication address of the local server i.e. idx1-rep.

- Save the file.

2. Repeat this across all indexers (idx1-8).
3. SSH to the master node, admin1.
4. As user 'splunk' issue the command:

```
$SPLUNK_HOME/bin/splunk rolling-restart cluster-peers
```

Verify Cluster Configuration

1. Navigate to the master node's web GUI (e.g. <https://admin1:8000>).
2. Select Settings -> Index Clustering.
3. All eight (8) indexers should appear as searchable.

Figure 167 Searchable Indexer Nodes

Indexer Clustering: Master Node

✓ All Data is Searchable ✓ Search Factor is Met

8 searchable 0 not searchable
PEERS

Peers (8) Indexes (2) Search Heads (1)

filter 10 per page ▾

i	Peer Name ↕	Fully Searchable ↕	Status ↕
>	idx2	✓ Yes	Up
>	idx8	✓ Yes	Up
>	idx1	✓ Yes	Up
>	idx4	✓ Yes	Up
>	idx6	✓ Yes	Up
>	idx5	✓ Yes	Up
>	idx7	✓ Yes	Up
>	idx3	✓ Yes	Up

Configure Receiving on the Peer Nodes

In order for the indexers (aka peer nodes) to receive data from the forwarders, the inputs.conf file of all the indexers needs to be configured with a stanza to enable the TCP port 9997. This is done by editing a special purpose app's inputs.conf file in the cluster master, i.e. admin1, as follows.

1. On the command line of the master node (admin1), navigate to

```
$SPLUNK_HOME/etc/master-apps/_cluster/local
```

2. Create and edit the file 'inputs.conf' with the following content:

```
[splunktcp://:9997]
```

```
connection_host = ip
```

```
[splunk@admin1 local]$ pwd
/data/disk1/splunk/etc/master-apps/_cluster/local
[splunk@admin1 local]$ vi inputs.conf
```

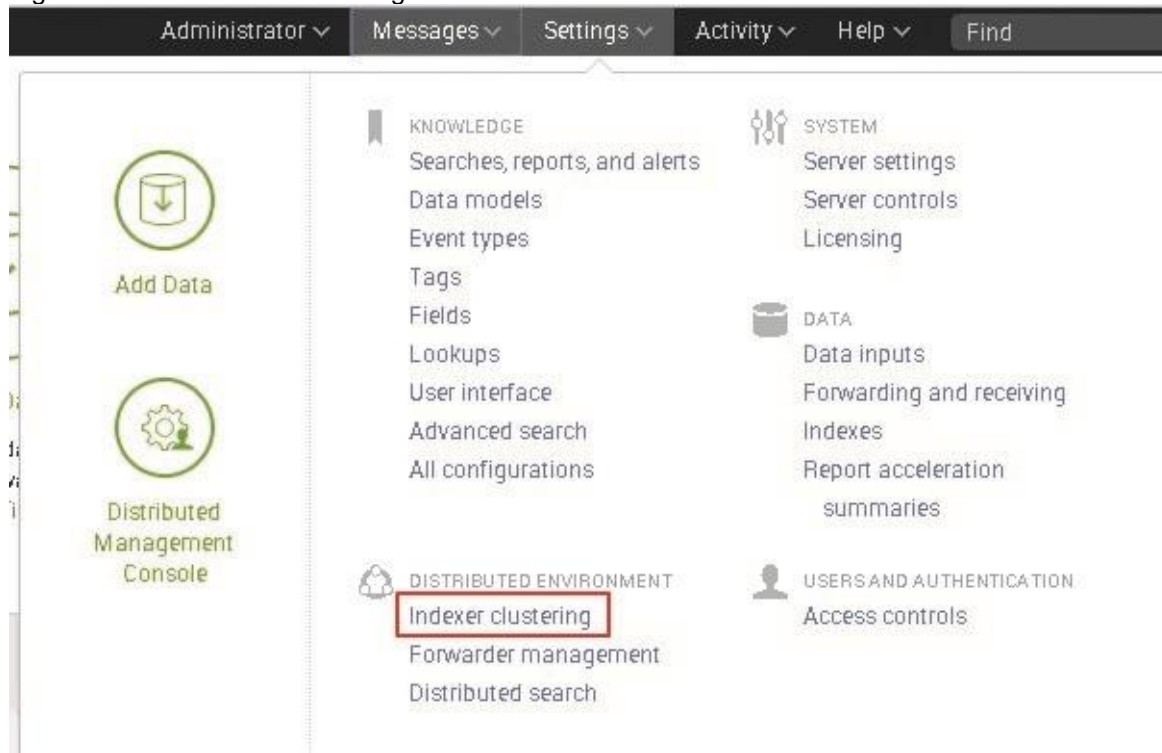
```
[splunktcp://9997]
connection_host = ip
```



Note: If this configuration uses DNS, edit 'connection_host = dns'.

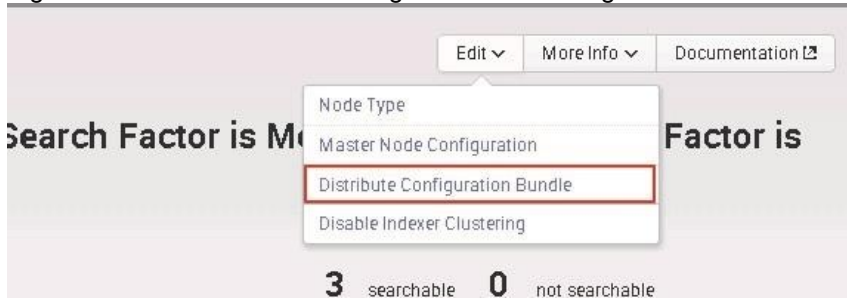
3. Navigate to the admin1 web interface via browser.
4. Navigate to Settings > Distributed Environment > Indexer Clustering.

Figure 168 Indexer Clustering



5. Select Edit > Distribute Configuration Bundle.

Figure 169 Indexer Clustering: Distribute Configuration Bundle



6. Select `Distribute Configuration Bundle`.

Figure 170 Distribute Configuration Bundle



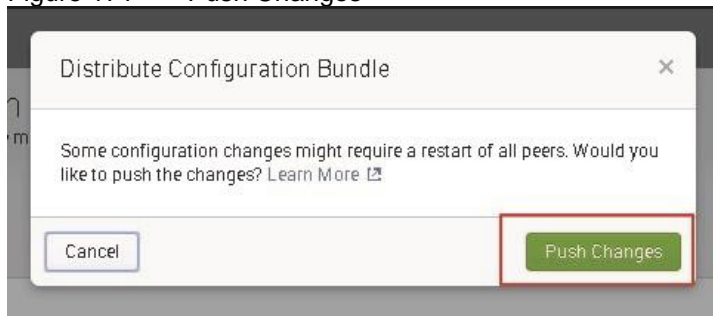
Last Push: ✓ Successful

Time 3/24/2015, 3:31:25 PM

Bundle ID ? 0DE75C59B77A4F1B3296FB0E75B1D750

7. Acknowledge the warning, and push changes.

Figure 171 Push Changes



8. Once Push is complete, the GUI should reflect a successful push.

Last Push: ✓ Successful

Time 4/6/2015, 2:38:25 PM

Bundle ID ? 7AD6211B37846DC07D746847C08597CE

Configure Master to Forward All its Data to the Indexer Layer

It is a best practice to forward all master node internal data to the indexer (peer node) layer. This has several advantages:

It enables diagnostics for the master node if it goes down. The data leading up to the failure is accumulated on the indexers, where a search head can later access it.

The preferred approach is to forward the data directly to the indexers, without indexing separately on the master. You do this by configuring the master as a forwarder. These are the main steps:

- Make sure that all necessary indexes exist on the indexers. This is normally the case, unless you have created custom indexes on the master node. Since `_audit` and `_internal` exist on indexers as well as the master, there is no need to create separate versions of those indexes to hold the corresponding master data.
- Configure the master as a forwarder. Create an `outputs.conf` file on the master node that configures it for load-balanced forwarding across the set of peer nodes. The indexing function on the master must also be turned off, so that the master does not retain the data locally as well as forward it to the peers.

In the cluster master node `admin1`, perform the following:

1. Create 'outputs.conf' file in the master node at `$SPLUNK_HOME/etc/system/local` directory

```
[splunk@admin1 root]$ cd $SPLUNK_HOME/etc/system/local/
[splunk@admin1 local]$ vi outputs.conf
```

2. Create an `outputs.conf` file with the following content:

```
#Turn off indexing on the master
[indexAndForward]
index = false

[tcput]
defaultGroup = search_peers
forwardedindex.filter.disable = true
indexAndForward = false

[tcput:search_peers]
server=idx1:9997,idx2:9997,idx3:9997,idx4:9997,idx5:9997,idx6:9997,idx7:9997,idx8:9997
autoLB = true
```

```
#Turn off indexing on the master

[indexAndForward]

index = false

[tcput]

defaultGroup = search_peers

forwardedindex.filter.disable = true

indexAndForward = false

[tcput:search_peers]
```

```
server=idx1:9997,idx2:9997,idx3:9997,idx4:9997,idx5:9997,idx6:9997,idx7:9997,idx8
:9997 autoLB = true
```



Note: It may be advantageous for scalability to use the "indexer discovery" feature instead of statically assigning indexers to forward to.

- Restart Splunk (\$SPLUNK_HOME/bin/splunk restart)

Configure Search Head Clustering

A search head cluster is a group of Splunk Enterprise search heads that serves as a central resource for searching. The members of a search head cluster are essentially interchangeable. You can run the same searches, view the same dashboards, and access the same search results from any member of the cluster.



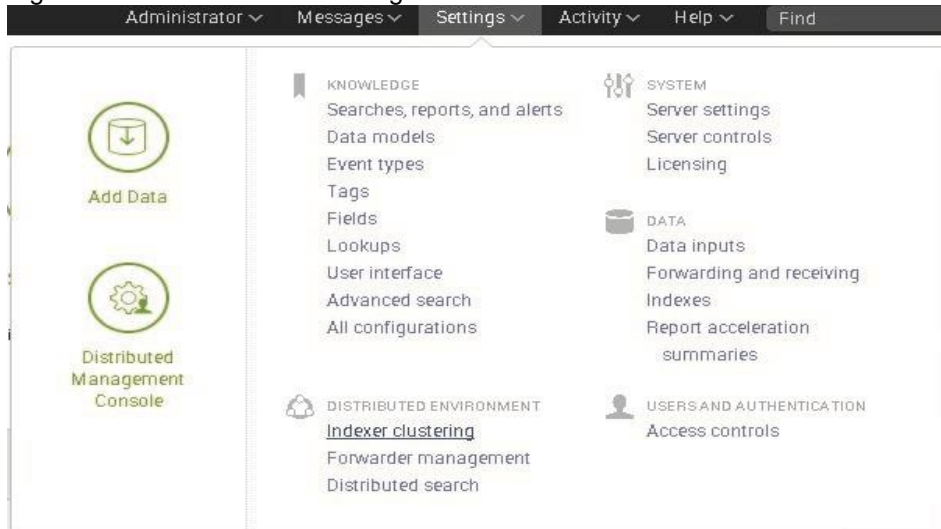
Note: In order to take full advantage of the search head cluster (also described in Splunk Architecture & Terminology), you must utilize a virtual or physical load balancer according to the enterprise's standards. Due to variability, the operator is suggested to use their own discretion in installing and configuring this. Further notes for configuration are provided under "Configuring Search Head Load-Balancing".

Add Search Heads to Master Node

A Splunk Enterprise instance can be configured as a search head via the Indexer clustering feature.

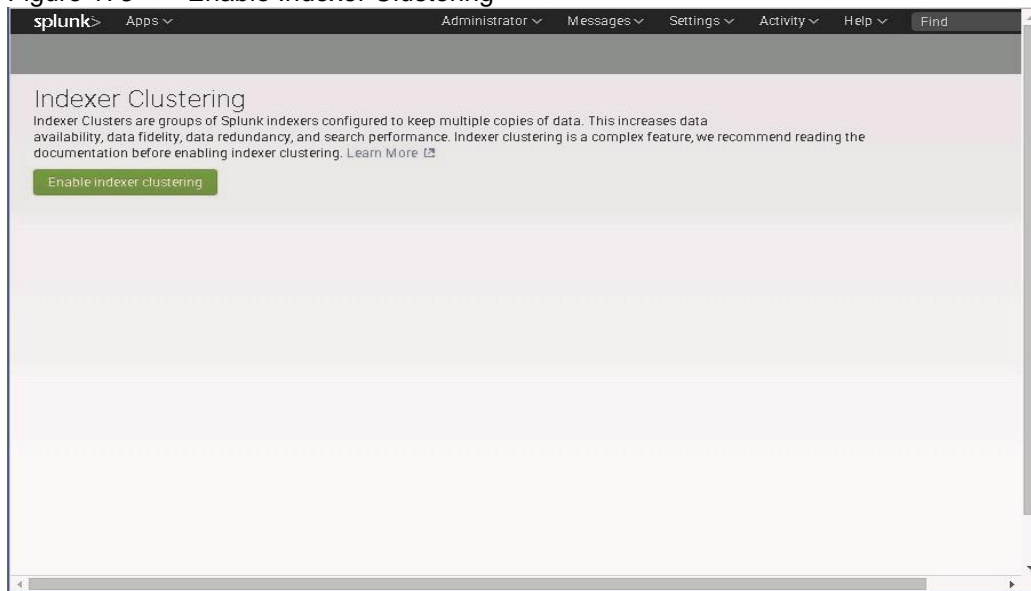
- Log onto one of the search heads as user admin.
- Navigate to Settings > Indexer Clustering.

Figure 172 Indexer Clustering



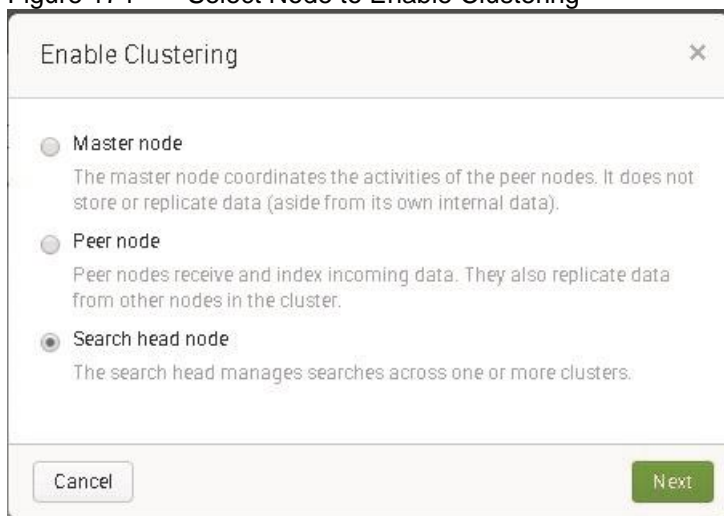
- Click Enable Indexer Clustering.

Figure 173 Enable Indexer Clustering



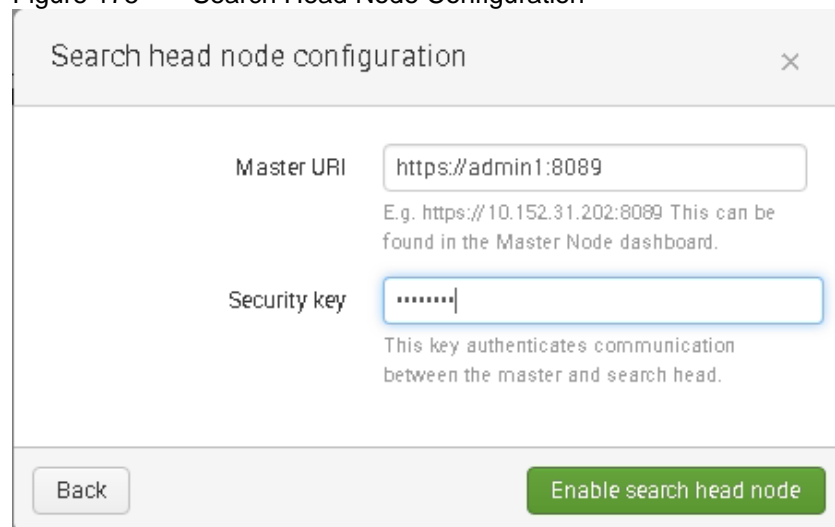
4. In the Enable Clustering dialog box, click on Search head node.
5. Click Next.

Figure 174 Select Node to Enable Clustering



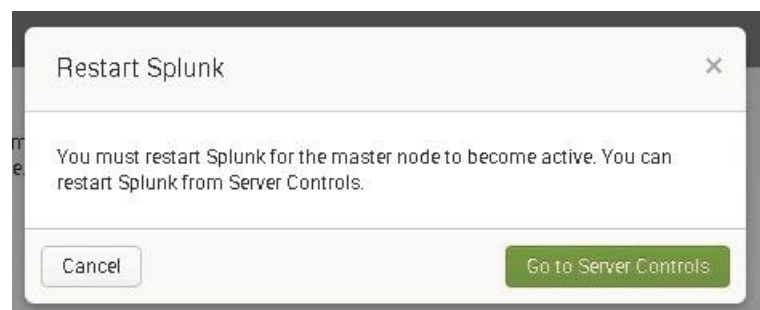
6. Enter the hostname of the master node including the the Master port number (default: 8089) in the format `https://<hostname_or_IP>`. (For example, <https://admin1:8089>) (Figure 175)
7. Enter the same security key that was used while configuring the master node e.g. `splunk+cisco`.
8. Click Enable search head node.

Figure 175 Search Head Node Configuration



The screenshot shows a dialog box titled "Search head node configuration" with a close button (X) in the top right corner. It contains two input fields: "Master URI" and "Security key". The "Master URI" field contains the text "https://admin1:8089". Below this field is a note: "E.g. https://10.152.31.202:8089 This can be found in the Master Node dashboard." The "Security key" field contains a series of dots ".....". Below this field is a note: "This key authenticates communication between the master and search head." At the bottom of the dialog, there are two buttons: "Back" on the left and "Enable search head node" on the right.

9. Click `Go to server controls` to view the Server controls screen and to restart Splunk.



The screenshot shows a dialog box titled "Restart Splunk" with a close button (X) in the top right corner. It contains a message: "You must restart Splunk for the master node to become active. You can restart Splunk from Server Controls." At the bottom of the dialog, there are two buttons: "Cancel" on the left and "Go to Server Controls" on the right.

10. Repeat the above steps to configure all three servers with hostnames sh1, sh2 and sh3 to be search heads.
11. Verify the search head cluster members in the master node, by navigating to `Setting > Indexer clustering`.
12. Click the `Search Heads` tab. (Figure 176)

Figure 176 Verify Search Head Cluster Members in the Master Node

Indexer Clustering: Master Node

✓ All Data is Searchable
✓ Search Factor is Met
✓ Replication Factor is Met

8 searchable **0** not searchable
PEERS

2 searchable **0** not searchable
INDEXES

Peers (8) | Indexes (2) | Search Heads (4)

filter 10 per page

Search head name	Status
> admin1	✓ Up
> sh3	✓ Up
> sh2	✓ Up
> sh1	✓ Up

About Support File a Bug Documentation Privacy Policy

© 2005-2015 Splunk Inc. All rights reserved.

Configure the Deployer

A Splunk Enterprise instance that distributes apps and certain other configuration updates to search head cluster members is referred to as a 'Deployer'. Any Splunk Enterprise instance can be configured to act as the Deployer. In this solution the admin1 is selected to serve this function as well.



Note: Do not locate deployer functionality on a search head cluster member. The deployer must be a separate instance from any cluster member, as it is used to manage the configurations for the cluster members.

1. Open an SSH session to admin1.
2. Navigate to `$SPLUNK_HOME/etc/system/local/`
3. As the user 'splunk', edit `server.conf` to include the following:

```
pass4SymmKey = your_secret_key (e.g. splunk+cisco)
```

```
[license]
active_group = Enterprise

[clustering]
access_logging_for_heartbeats = 1
max_peer_build_load = 5
mode = master
pass4SymmKey = $1$1+E/9zMEWeYo0yoxHQ==
replication_factor = 2

[shclustering]
pass4SymmKey = splunk+cisco
```

4. Restart the admin1 instance (\$SPLUNK_HOME/bin/splunk restart)

Configure Search Head Cluster Members

1. As the user 'splunk', Enter the following commands to make this search head join the search head cluster. Change the mgmt._uri per respective Search Head

```
$SPLUNK_HOME/bin/splunk init shcluster-config -auth admin:cisco123 -mgmt_uri
https://sh1:8089 -replication_port 18081 -replication_factor 2 -
conf_deploy_fetch_url https://admin1:8089 -secret splunk+cisco
```

2. Restart Splunk Search Head after the command is issued

```
$SPLUNK_HOME/bin/splunk restart
```

```

[splunk@sh1 ~]$ $SPLUNK_HOME/bin/splunk init shcluster-config -auth admin:cisco123 -mgmt uri
https://sh1:8089 -replication_port 18081 -replication_factor 2 -conf_deploy_fetch_url https
://admin1:8089 -secret splunk+cisco
Search head clustering has been initialized on this node.
You need to restart the Splunk Server (splunkd) for your changes to take effect.
[splunk@sh1 ~]$ $SPLUNK_HOME/bin/splunk restart
Stopping splunkd...
Shutting down. Please wait, as this may take a few minutes.
..
Stopping splunk helpers...
Done.

Splunk> Be an IT superhero. Go home early.

Checking prerequisites...
Checking http port [8000]: open
Checking mgmt port [8089]: open
Checking appserver port [127.0.0.1:8065]: open
Checking kvstore port [8191]: open
Checking configuration... Done.
Checking critical directories... Done
Checking indexes...
Validated: _audit_blocksignature_internal_introspection_thefishbucket hi
story main summary
Done

Bypassing local license checks since this instance is configured with a remote license master.

Checking filesystem compatibility... Done
Checking conf files for problems...
Done
Checking replication_port port [18081]: open
All preliminary checks passed.

Starting splunk server daemon (splunkd)...
Done

If you get stuck, we're here to help.
Look for answers here: http://docs.splunk.com
The Splunk web interface is at http://sh1:8000

```

3. Repeat the above steps for all search heads.

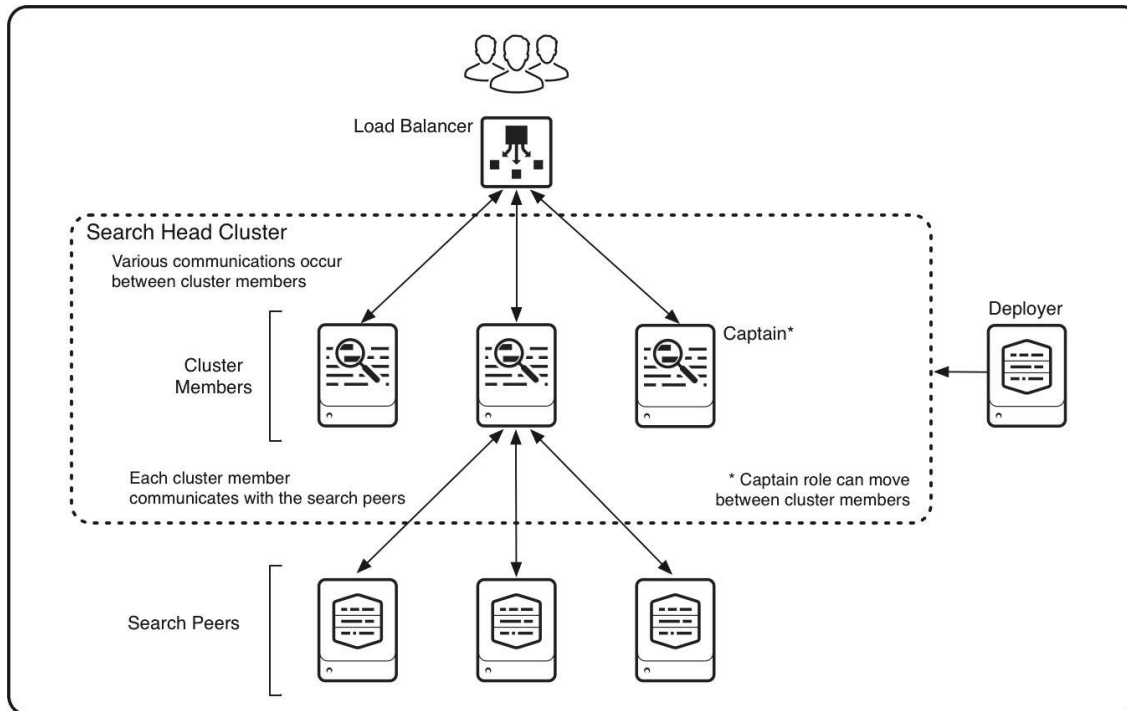
Elect a Search Head Captain

A search head cluster consists of a group of search heads that share configurations, job scheduling, and search artifacts. The search heads are known as the cluster members.

One cluster member has the role of captain, which means that it coordinates job scheduling and replication activities among all the members. It also serves as a search head like any other member, running search jobs, serving results, and so on. Over time, the role of captain can shift among the cluster members.

The following illustration shows a small search head cluster, consisting of three members:

Figure 177 Search Head Cluster with its Members



A search head cluster uses a dynamic captain. This means that the member serving as captain can change over the life of the cluster. Any member has the ability to function as captain. When necessary, the cluster holds an election, which can result in a new member taking over the role of captain.

The procedure described in this section helps bootstrap the election process.

1. Log into any search head as user `splunk`.
2. Start the search head captain election bootstrap process by using the following command as the `splunk` user.

```
$SPLUNK_HOME/bin/splunk bootstrap shcluster-captain -servers_list
"https://sh1:8089,https://sh2:8089,https://sh3:8089" -auth admin:cisco123
```

```
[splunk@sh1 ~]$ $SPLUNK_HOME/bin/splunk bootstrap shcluster-captain -servers_list "https://sh1:8089,
https://sh2:8089,https://sh3:8089" -auth admin:cisco123
Successfully bootstrapped this node as the captain with the given servers.
```



Note: The search head captain election process can be started from any of the search head cluster members.

Configure Search Heads to Forward their Data to the Indexer Layer

It is a best practice to forward all search head internal data to the search peer (indexer) layer. This has several advantages:

- It enables diagnostics for the search head if it goes down. The data leading up to the failure is accumulated on the indexers, where another search head can later access it.

- By forwarding the results of summary index searches to the indexer level, all search heads have access to them. Otherwise, they are only available to the search head that generates them.

The recommended approach is to forward the data directly to the indexers, without indexing separately on the search head. You do this by configuring the search head as a forwarder by creating an `outputs.conf` file on the search head that configures the search head for load-balanced forwarding across the set of search peers (indexers). The indexing on the search head, so that the search head does not both retain the data locally as well as forward it to the search peers.

1. Using the CLI, as the splunk user on admin1, navigate to `$(SPLUNK_HOME)/etc/shcluster/apps`.
2. Create the directory 'outputs' and 'outputs/local'.
3. Navigate to the newly created 'local' directory.
4. Within the `$(SPLUNK_HOME)/etc/shcluster/apps/outputs/local/` directory, create the file `outputs.conf` with the following content in step 5.

```
cd $(SPLUNK_HOME)/etc/shcluster/apps
```

```
mkdir -p outputs/local
```

```
vi outputs.conf
```

5. Copy and paste the following contents.

```
# Turn off indexing on the master
```

```
[indexAndForward]
```

```
index = false
```

```
[tcpout]
```

```
defaultGroup = search_peers
```

```
forwardedindex.filter.disable = true
```

```
indexAndForward = false [tcpout:search_peers]
```

```
server=idx1:9997,idx2:9997,idx3:9997,idx4:9997,idx5:9997,idx6:9997,idx7:9997,idx8:9997
```

```
autoLB = true
```



Note: It may be advantageous for scalability to use the "indexer discovery" feature instead of statically assigning indexers to forward to.

6. Execute the 'apply shcluster-bundle' command:

```
$(SPLUNK_HOME)/bin/splunk apply shcluster-bundle -target https://sh1:8089 -auth admin:cisco123
```

```
[splunk@admin1 local]$ $SPLUNK_HOME/bin/splunk apply shcluster-bundle -target
https://sh1:8089 -auth admin:cisco123
Warning: Depending on the configuration changes being pushed, this command m
ight initiate a rolling restart of the cluster members. Please refer to the
documentation for the details. Do you wish to continue? [y/n]: y
Bundle has been pushed successfully to all the cluster members.
[splunk@admin1 local]$
```

```
# Turn off indexing on the master
[indexAndForward]
index = false

[tcput]
defaultGroup = search_peers
forwardedindex.filter.disable = true
indexAndForward = false

[tcput:search_peers]
server=idx1:9997,idx2:9997,idx3:9997,idx4:9997,idx5:9997,idx6:9997,idx7:9997,
idx8:9997
autoLB = true
```

7. Acknowledge the warning. A message pop-up will notify that the bundle has been pushed successfully.

Configure Search Head Load-Balancing

As described above in the introductory note about search head clustering, it is useful to utilize a load balancer to take advantage of the search head cluster.

1. Designate a common URL for use throughout the enterprise (For example, <https://splunk.domain.com>)
2. The common URL should balance traffic between all three search heads and their respective ports. For example, <https://sh1:8000>, <https://sh2:8000>, <https://sh3:8000>.



Note: Explicit instructions for configuring the designated load balancer will differ by vendor, but the functionality and load balancing direction is the same.

Verify Search Head Clustering:

3. SSH to any search head.
4. As the 'splunk' user, issue the command '\$SPLUNK_HOME/bin/splunk show shcluster-status -auth <username>:<password>'.

```
$SPLUNK_HOME/bin/splunk show shcluster-status -auth admin:cisco123
```

```

[splunk@sh1 ~]$ $SPLUNK_HOME/bin/splunk show shcluster-status -auth admin:cisco123

Captain:
          elected_captain : Fri Apr 17 18:11:08 2015
          id              : 179BA91B-72DB-4E50-A33D-5EC143FCC
901
          initialized_flag : 1
          label            : sh1
          maintenance_mode : 0
          mgmt_uri         : https://sh1:8089
          min_peers_joined_flag : 1
          rolling_restart_flag : 0
          service_ready_flag : 1

Members:
  sh3
          label : sh3
          mgmt_uri : https://192.168.11.113:8089
          status : Up
  sh2
          label : sh2
          mgmt_uri : https://192.168.11.112:8089
          status : Up
  sh1
          label : sh1
          mgmt_uri : https://192.168.11.111:8089
          status : Up
[splunk@sh1 ~]$

```

- Alternatively, run '\$SPLUNK_HOME/bin/splunk list shcluster-members -auth <username>:<password>' to view the various members.

```
$SPLUNK_HOME/bin/splunk list shcluster-members -auth admin:cisco123
```

```
[splunk@sh1 ~]$ $SPLUNK_HOME/bin/splunk list shcluster-members -auth admin
cisco123
    37386839-F178-49CB-9BEB-BAA7277150E4
        adhoc_searchhead:0
        advertise_restart_required:0
        artifact_count:0
        delayed_artifacts_to_discard:
        fixup_set:
        host_port_pair:192.168.11.113:8089
        kv_store_host_port:192.168.11.113:8191
        label:sh3
        last_heartbeat:1429319835
        peer_scheme_host_port:https://192.168.11.113:8089
        pending_job_count:0
        replication_count:0
        replication_port:18081
        replication_use_ssl:0
        site:default
        status:Up

    4C26874E-A557-4D94-AD8C-5E6B6788AE04
        adhoc_searchhead:0
        advertise_restart_required:0
        artifact_count:0
        delayed_artifacts_to_discard:
        fixup_set:
        host_port_pair:192.168.11.112:8089
        kv_store_host_port:192.168.11.112:8191
        label:sh2
        last_heartbeat:1429319834
        peer_scheme_host_port:https://192.168.11.112:8089
        pending_job_count:0
        replication_count:0
        replication_port:18081
        replication_use_ssl:0
        site:default
        status:Up

    AD4CFCAB-B976-4EBB-A0AC-9CF1100F0547
        adhoc_searchhead:0
        advertise_restart_required:0
        artifact_count:0
        delayed_artifacts_to_discard:
        fixup_set:
        host_port_pair:192.168.11.111:8089
        kv_store_host_port:192.168.11.111:8191
        label:sh1
        last_heartbeat:1429319835
        peer_scheme_host_port:https://192.168.11.111:8089
        pending_job_count:0
        replication_count:0
        replication_port:18081
        replication_use_ssl:0
        site:default
        status:Up
```

6. Navigate to the directory `$SPLUNK_HOME/etc/apps/outputs/default/` on any search head. List the directory, `outputs.conf` will be listed, verifying that it has been pushed by the Deployer.

```
cd $SPLUNK_HOME/etc/apps/outputs/default
```

```
ls -l
```



```
[splunk@sh2 default]$ pwd
/data/disk1/splunk/etc/apps/outputs/default
[splunk@sh2 default]$ ls -l
total 8
-rw-----. 1 splunk splunk 77 Apr 10 10:13 app.conf
-rw-rw-r--. 1 splunk splunk 296 Apr 10 10:13 outputs.conf
```



Note: This app will not appear under 'apps' within the GUI, but will appear under 'Apps > Manage Apps'

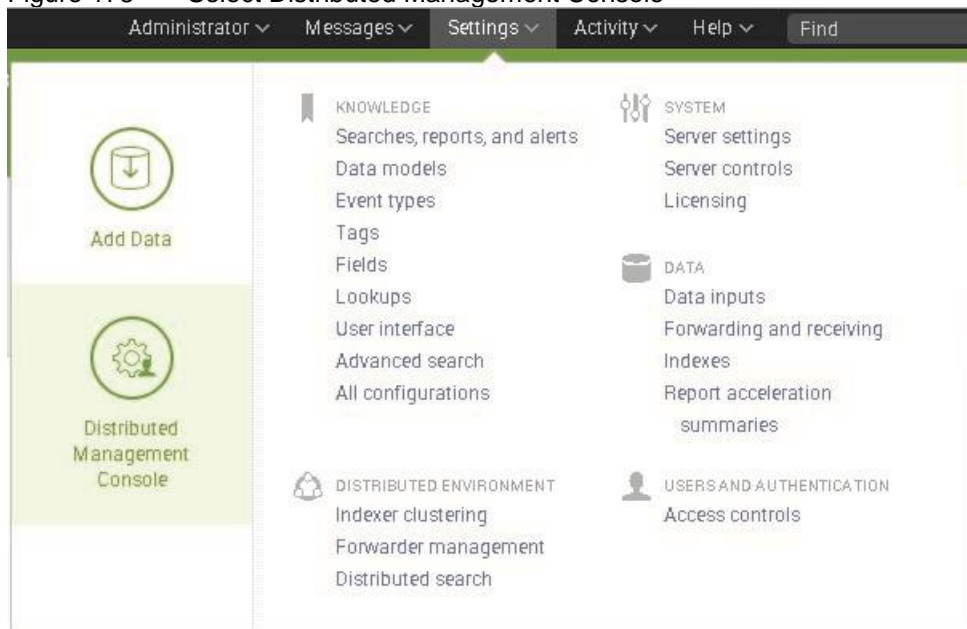
Configuring the Distributed Management Console

The distributed management console is a special purpose pre-packaged app that comes with Splunk Enterprise providing detailed performance information about the Splunk Enterprise deployment.

This section describes the procedure to configure the Distributed Management Console for this deployment. It is installed on the admin node, i.e. admin1. Please refer to [\[Splunk Documentation\]](#) for learning about other installation options.

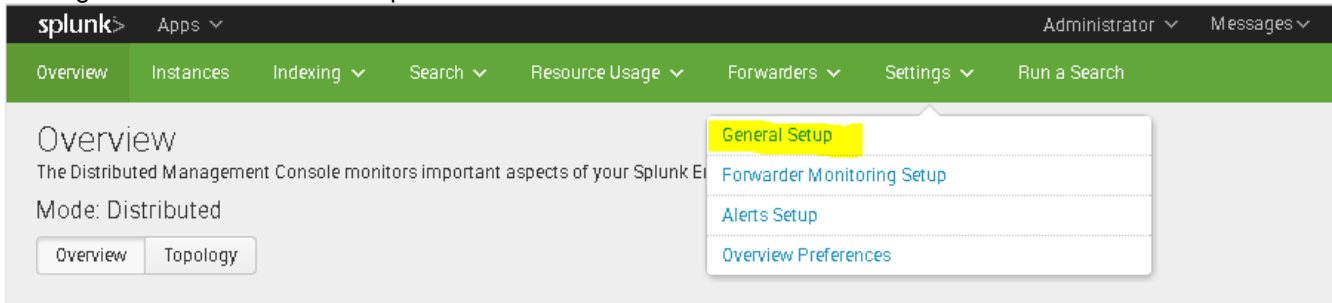
1. Navigate to the Splunk Web Interface on admin1 (<https://admin1:8000/>).
2. Click Settings > Distributed Management Console. (Figure 178)

Figure 178 Select Distributed Management Console



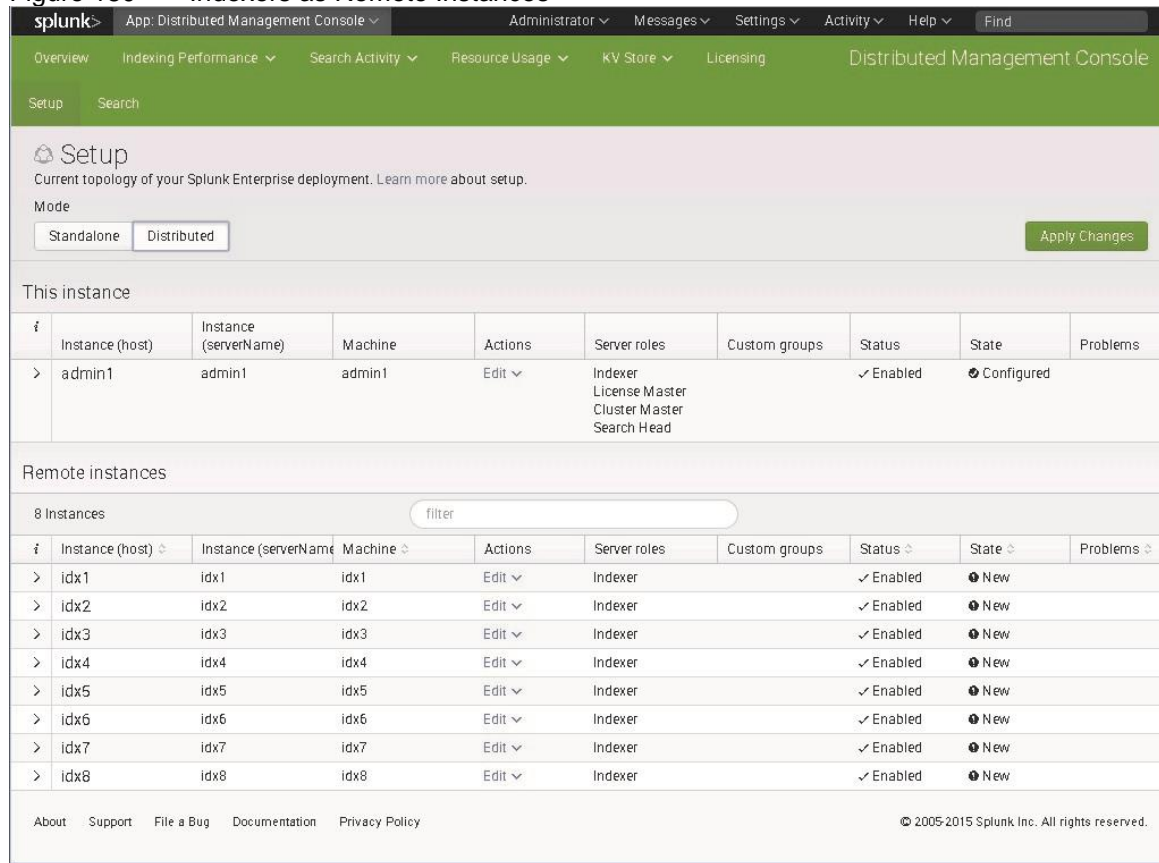
3. In the Distributed Management Console app, click Settings > General Setup. (Figure 179)

Figure 179 General Setup



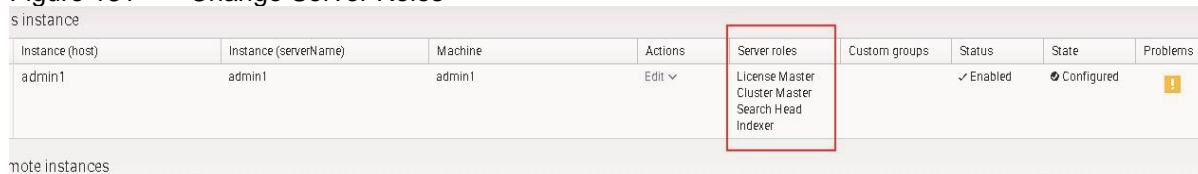
4. Click the `Distributed` mode. Click `Continue` through the warning (make sure this is on admin1). This should show all the eight indexers as remote instances. (Figure 180)

Figure 180 Indexers as Remote Instances



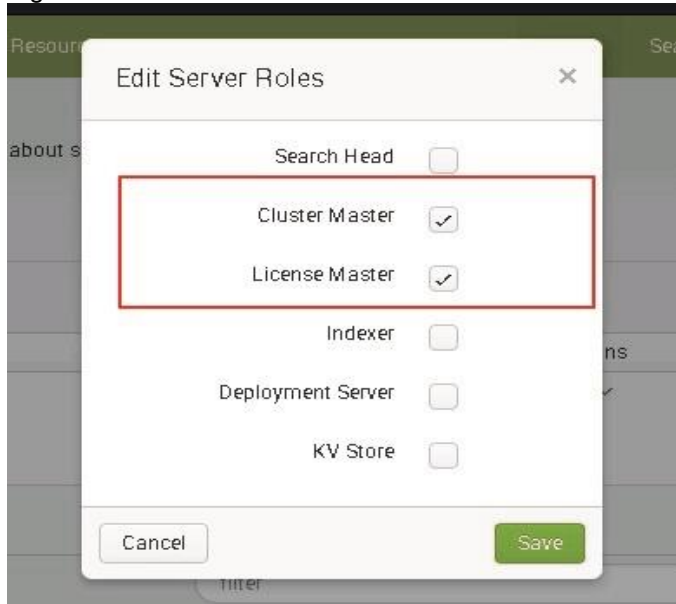
5. Select `Edit` on the admin1 box. The server must change roles to function properly. (Figure 181)

Figure 181 Change Server Roles



6. Select only 'License Master' and 'Cluster Master'.

Figure 182 Edit Server Roles



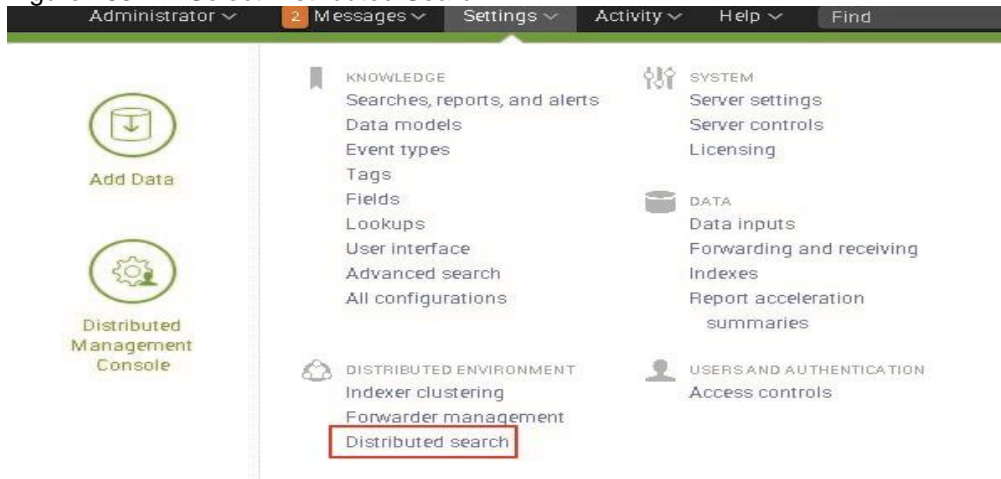
7. Click **Save**.
8. Click the **Apply Changes** button at the top right.
9. Confirm that changes have been saved successfully.

Configure Search Heads in Distributed Management Console

In the previous section the Distributed Management Console (DMC) was configured to manage the indexers and the master node. This section provides the procedure to configure DMC to monitor the search heads.

1. Navigate to the Master Node (admin1) via the GUI.
2. Open **Settings** → **Distributed Environment** → **Distributed search**.

Figure 183 Select Distributed Search



3. Select **Search Peers**. (Figure 184)

Figure 184 Search Peers

Distributed search

Distributed search

Perform a search across multiple Splunk indexes.

Distributed search setup

Search peers

4. Select New.

Figure 185 Selecting New Peers

New

Showing 1-11 of 11 items

Peer ↕	Splunk server name ↕
idx1.rep:8089	idx1
idx2.rep:8089	idx2
idx3.rep:8089	idx3
idx4.rep:8089	idx4
idx5.rep:8089	idx5

Figure 186 Add Search Peers

Add search peers

Use this page to explicitly add distributed search peers. Enable distributed search through the Distributed search setup page in Splunk Settings.

Peer *

sh2.8089

Search peer is either `servername:management_port` or `IP:management_port`. For example `'myhost:8089'`.

Distributed search authentication

To share a public key for distributed authentication, enter a username and password for an admin user on the remote search peer.

Remote username *

admin

Remote password *

Confirm password

Cancel

Save

5. Add a search Peer.

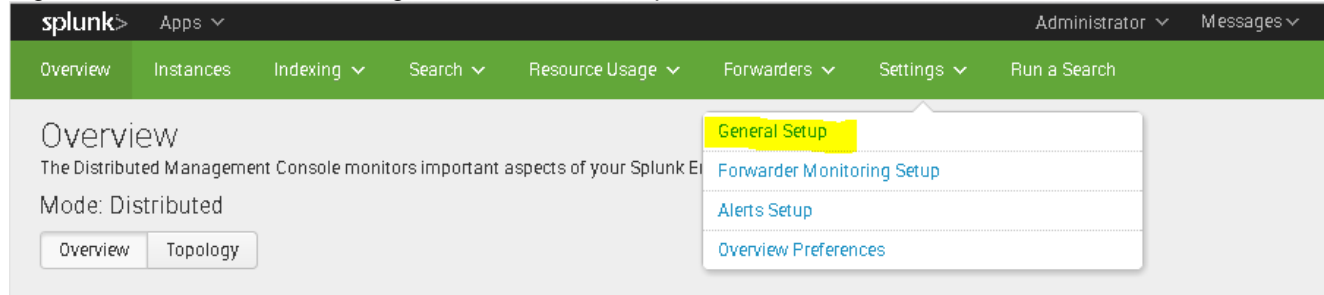
[Peer] - Enter the hostname or IP of one of your search heads

[Remote username] - use 'admin'

[Remote password] - the password to the Splunk admin account on the search head

6. Repeat this process for the other two search heads.
7. On the Master Node (admin1), navigate to Settings > Distributed Management Console.
8. In the Distributed Management Console app, click Settings > General Setup. (Figure 187)

Figure 187 Distributed Management Console: Setup



9. The three newly added search heads should be listed under 'remote instances' as 'new'.
10. Select Edit within the table next to the instance name, and ensure that the server roles are 'Search Head' and 'KV Store'. (Figure 188)

Figure 188 Verify Server Roles

>	sh1	sh1	sh1	Edit ▾	Search Head KV Store
>	sh2	sh2	sh2	Edit ▾	KV Store Search Head
>	sh3	sh3	sh3	Edit ▾	KV Store Search Head

11. Confirm changes and roles.
12. Ensure that the Master Node (admin1) does not have the role of 'search head'.
13. Click Apply Changes. (Figure 189)

Figure 189 Click on Apply Changes after Verifying



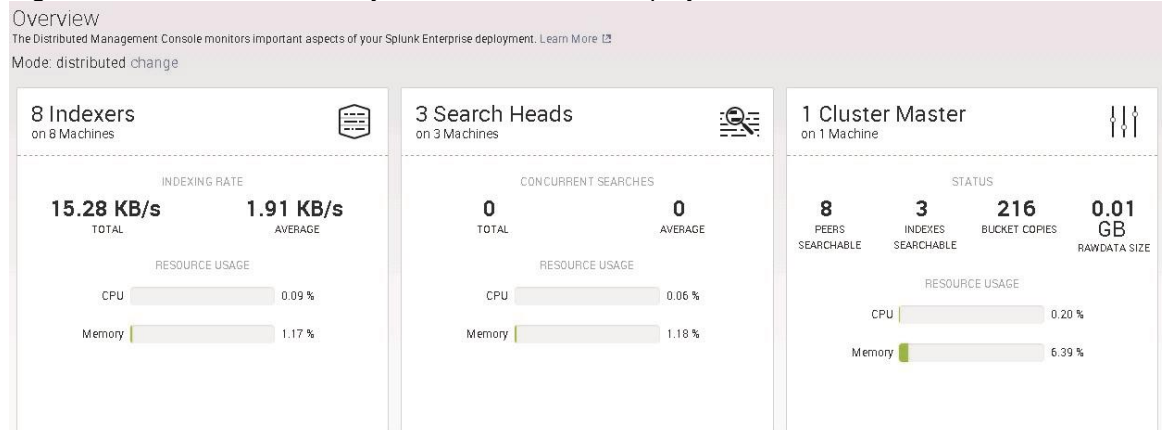
14. Click Overview.

Figure 190 Splunk Overview



15. DMC should now display “Search Heads” within the overview. (Figure 191)

Figure 191 Overview: Verify the Search Heads Displayed



Configuring Archive of Data from Cold to Frozen

As the indexer indexes the incoming data, it creates two types of files:

- The raw data in compressed form (rawdata files)
- Indexes that point to the raw data, plus some metadata (index files)

Together, these files constitute the Splunk Enterprise index. The files reside in sets of directories organized by age. Some directories contain newly indexed data; others contain previously indexed data. The number of such directories can grow quite large, depending on how much data you're indexing.

In short, each of the index directories is known as a **bucket**.

- An "index" contains compressed raw data and associated index files.
- An index resides across many age-designated index directories.
- An index directory is called a bucket.

A bucket moves through several stages as it ages:

- hot
- warm
- cold
- frozen
- thawed

As buckets age, they "roll" from one stage to the next. As data is indexed, it goes into a hot bucket. Hot buckets are both searchable and actively being written to. An index can have several hot buckets open at a time.

When certain conditions occur (for example, the hot bucket reaches a certain size or splunkd gets restarted), the hot bucket becomes a warm bucket ("rolls to warm"), and a new hot bucket is created in its place. Warm buckets are searchable, but are not actively written to. There are many warm buckets.

Once further conditions are met (for example, the index reaches some maximum number of warm buckets), the indexer begins to roll the warm buckets to cold, based on their age. It always selects the oldest warm

bucket to roll to cold. Buckets continue to roll to cold as they age in this manner. After a set period of time, cold buckets roll to frozen, at which point they are either archived or deleted. By editing attributes in `indexes.conf`, the bucket aging policy can be specified, which determines when a bucket moves from one stage to the next.

If the frozen data has been archived, it can later be thawed. Thawed data is available for searches. If archival of specific sets of data is required, each additional index that is added will require the stanza: `coldToFrozenDir = <directory of frozen data>`

Each index that is added will require this stanza to be appended. In the section `Verifying Master and Peer Replication`, an index will be created for the purposes of testing. Different configurations will apply to indexes as the Splunk installation matures.

For testing purposes only, an index will be pushed from the master node (`admin1`) in the verification stage of this CVD by applying the following stanza:

```
[archival]

coldToFrozenDir = /path/to/frozen
```

More information regarding archival can be found in the [documentation](#)

Configuring the Deployment Server

In this section, the server `admin2` is configured to function as the Deployment server, and procedure to push a sample “Splunk App” from the Deployment Server to a Universal Forwarder on a test server(not part of this CVD).

Any Splunk instance can act as a Deployment Server that assists in maintaining and deploying apps. In particular, the Deployment Server acts as a central manager for Universal Forwarders deployed throughout the enterprise.

Any configuration to be pushed to remote instances will be hosted under `$SPLUNK_HOME/etc/deployment-apps/`.

In the following section, a Universal Forwarder will be installed on a machine separate from the servers that make up the Splunk Enterprise platform of this CVD. The only requirement for this is it must be reachable via the same network to which the Indexers are connected to.

Once the machine is connected to the network with connectivity to the UCS platform, follow the steps below.



Note: In this documentation, it is assumed that the machine with Universal Forwarder is reachable via `192.168.11.0/24` network (in other words via NIC `eth1` of the Cisco UCS servers). This would require the respective VLANs configured appropriately to provide appropriate connectivity between the Cisco UCS infrastructure on which Splunk platform is built and the server with a universal forwarder.



Note: The Deployment Server is installed by default when Splunk Enterprise is deployed. In this CVD, the `admin2` box will function as the designated Deployment Server.

Installing a Universal Forwarder on a Test Server

1. Download Splunk Universal Forwarder: <http://www.splunk.com/download/universalfowarder>
2. Install the package as detailed in the [documentation](#) for the appropriate operating system of the Universal Forwarder host.

Register Universal Forwarder with the Deployment Server

1. Via the command line, access the system hosting the Universal Forwarder.
2. Navigate to the `$SPLUNK_HOME/etc/system/local` directory.
3. Create and edit the file 'deploymentclient.conf' with the following content

```
[deployment-client]
clientName = MyForwarder

[target-broker:deploymentServer]
targetUri = admin2:8089
```

`clientName` = the name or identifier of the host system

`targetUri` = the hostname/IP and port of the Deployment Server (For example, `admin2:8089`)

4. As the user 'splunk' restart the Universal Forwarder (`$SPLUNK_HOME/bin/splunk restart`)

Configure an App within the Deployment Server

1. In a browser, navigate to the Splunk instance's Web Interface of server `admin2`. (i.e. `https://admin2:8000/`)
2. Select `Settings -> Distributed Environment -> Forwarder Management`.
3. Notice the record of the Universal Forwarder communicating with the Deployment Server (this step may take up to five minutes due to polling cycle). (Figure 192)

Figure 192 Forwarder Management: Communication with the Deployment Server

The screenshot shows the 'Forwarder Management' page in the Splunk web interface. At the top, it displays '1 Client PHONED HOME IN THE LAST 24 HOURS' and '0 Clients DEPLOYMENT ERRORS'. Below this, there are tabs for 'Apps (0)', 'Server Classes (0)', and 'Clients (1)'. A search filter is present with the text 'filter'. Below the filter, there is a table with the following data:

i	Host Name	Client Name	IP Address	Actions	Machine Type
>	fwd1	MyForwarder	10.29.160.50	Delete Record	linux-x86_64

4. Using the command line, navigate to the Deployment Server, `admin2`.
5. Navigate to `$SPLUNK_HOME/etc/deployment-apps/`
6. Create the directory `appTest`.
7. Within `appTest` create the directory `local`.


```
[root@admin2 ~]# cd $SPLUNK_HOME/etc/deployment-apps
[root@admin2 deployment-apps]# mkdir appTest
[root@admin2 deployment-apps]# cd appTest
[root@admin2 appTest]# mkdir local
[root@admin2 appTest]# cd local
[root@admin2 local]# vi app.conf
```

8. Create the file 'app.conf' and include the following contents:

```
[tcpout]

defaultGroup = search_peers

[tcpout:search_peers]

autoLB = true

forceTimebasedAutoLB = true
server=idx1:9997,idx2:9997,idx3:9997,idx4:9997,idx5:9997,idx6:9997,idx7:9997,idx8:9997
```

```
[root@admin2 local]# cat app.conf
[tcpout]
defaultGroup = search_peers

[tcpout:search_peers]
autoLB = true
forceTimebasedAutoLB = true server=idx1:9997,idx2:9997,idx3:9997,idx4:9997,idx5:9997,idx6:9997,idx7:9997,idx8:9997
[root@admin2 local]#
```

9. As the splunk user, execute the command to reload the deployment server:

```
$SPLUNK_HOME/bin/splunk reload deploy-server
```

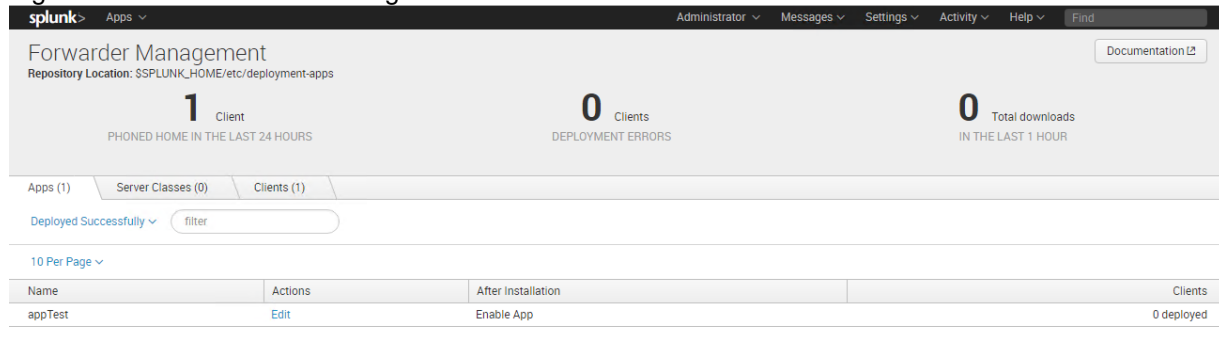
```
[splunk@admin2 local]$ $SPLUNK_HOME/bin/splunk reload deploy-server
Your session is invalid. Please login.
Splunk username: admin
Password:
Login successful, running command...
Reloading serverclass(es).
```



Note: The login step could be bypassed by appending "auth admin:cisco" to the command line.

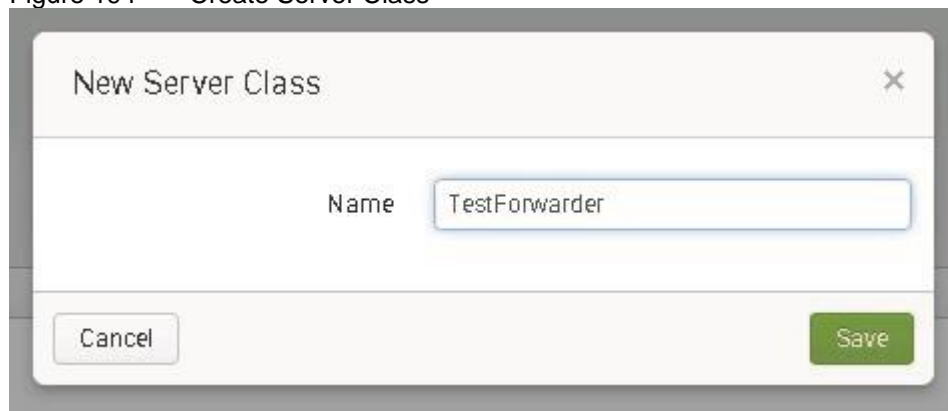
10. Navigate to the Web GUI on admin2 (<http://admin2:8000>) and navigate to Settings > Forwarder Management. Click Apps. (Figure 193)

Figure 193 Forwarder Management



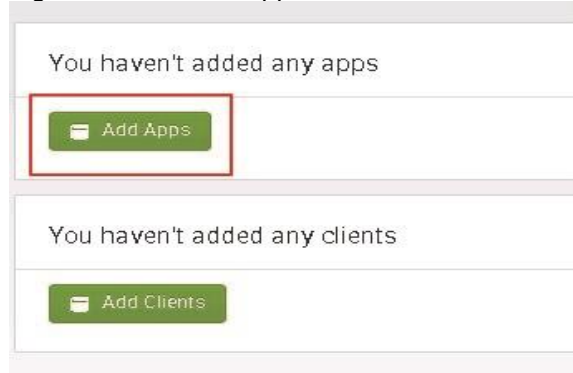
11. Zero apps have been deployed. Click `Server Class`.
12. Click `Create One` and give it the name `TestForwarder`.

Figure 194 Create Server Class



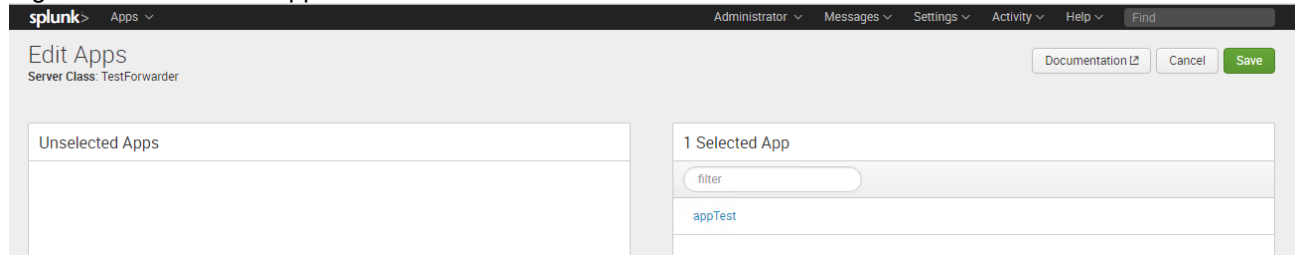
13. Figure 195 presents options for adding apps and clients. Click `Add Apps`.

Figure 195 Add Apps to New Server Class



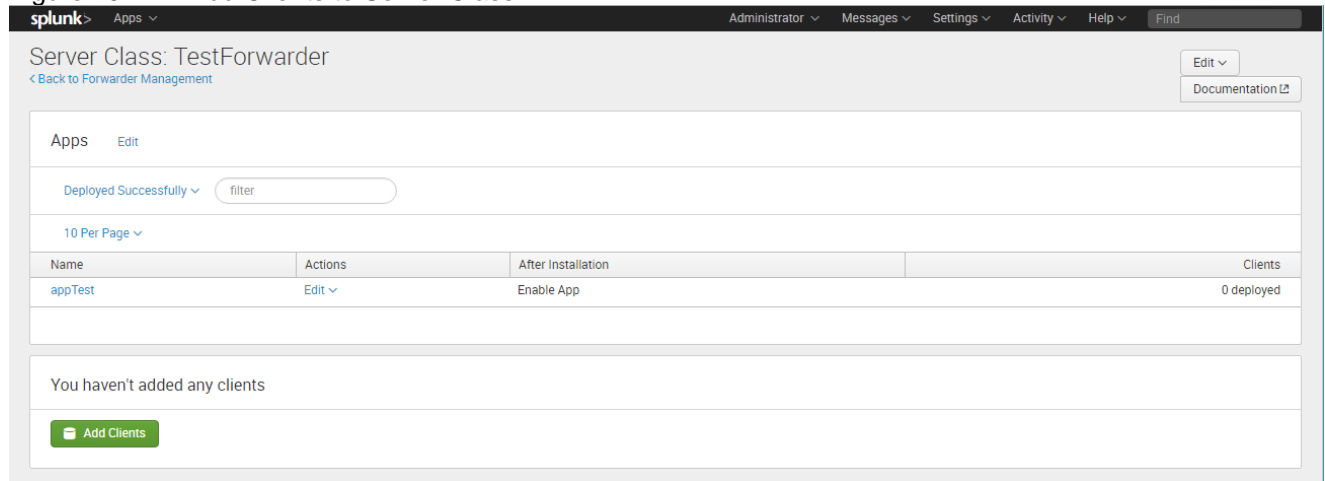
14. Click `appTest` in the `Unselected Apps` section to move it to `Selected Apps` section. (Figure 196)

Figure 196 Select Apps



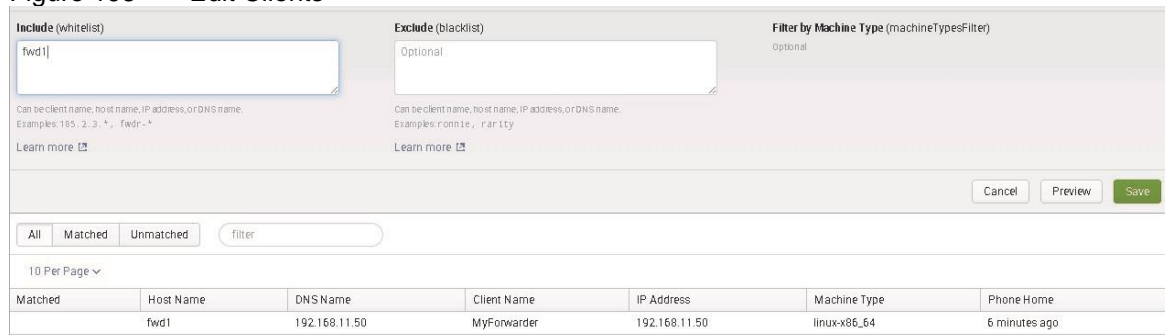
15. Click the `Save` button in the upper right.
16. The next screen will show the apps listed under **Apps**.
17. Click the `Add Clients` button. (Figure 197)

Figure 197 Add Clients to Server Class



18. Within the `Edit Clients` screen, add the hostname of the forwarder to the whitelist. In this instance, the forwarder used is named `fwd1` . (Figure 198)

Figure 198 Edit Clients



19. Click `Save`.
20. Go back to the `Forwarder Management` screen (`Select Settings -> Distributed Environment -> Forwarder Management`).

Figure 199 Forwarder Management

#	Host Name	Client Name	IP Address	Actions	Machine Type	Deployed Apps	Phone Home
>	fwd1	MyForwarder	192.168.11.50	Delete Record	linux-x86_64	1 deployed	6 minutes ago

21. On the forwarder box, navigate to `$SPLUNK_HOME/etc/apps`. List the directory to view the newly deployed app.

```
[root@fwd1 apps]# pwd
/opt/splunkforwarder/etc/apps
[root@fwd1 apps]# ls -la
total 4
drwxr-xr-x  7 splunk splunk 117 Apr 10 12:10 .
drwxr-xr-x 13 splunk splunk 4096 Mar 27 09:43 ..
drwxr-xr-x  4 splunk splunk  30 Mar 27 01:19 introspection_generator_addon
drwxr-xr-x  5 splunk splunk  47 Mar 27 09:43 learned
drwx----- 4 root  root   33 Apr 10 12:10 outputTest
drwxr-xr-x  5 splunk splunk  49 Mar 27 01:19 search
drwxr-xr-x  4 splunk splunk  35 Mar 27 01:19 SplunkUniversalForwarder
[root@fwd1 apps]#
```

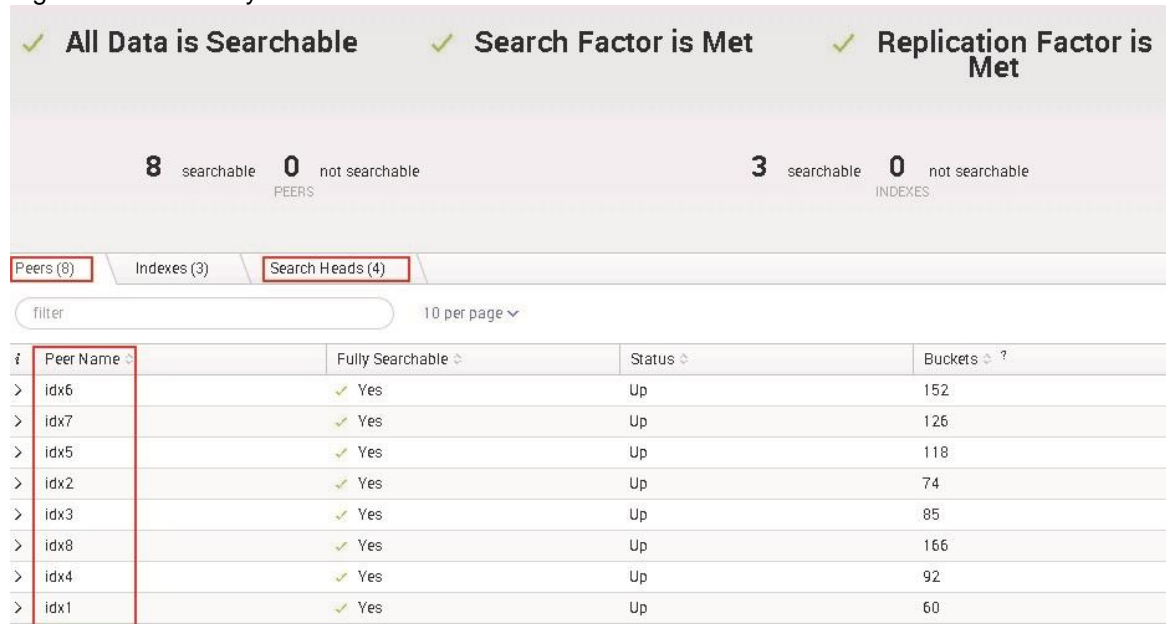
Installation Verification

The purpose of this verification is to ensure connectivity between the indexers, search heads, license master, master node, and the distributed management console.

Verifying from DMC

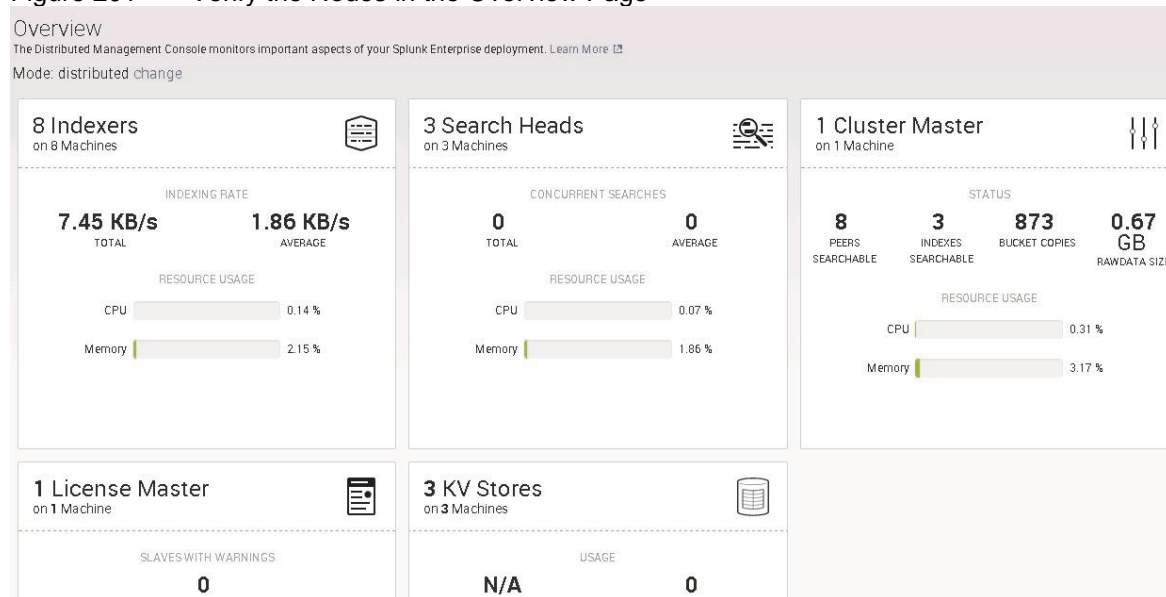
1. Log into the Master Node (admin1).
2. Navigate to Settings > Indexer Clustering.
3. Verify that all search heads and indexers are listed. (Figure 200)

Figure 200 Verify Search Heads and Indexers



4. Navigate to Settings > Distributed Management Console.
5. The overview page should display similar results (8 Indexers, 3 Search Heads, 1 Cluster Master, 1 License Master, 3 KV stores).

Figure 201 Verify the Nodes in the Overview Page



6. Navigate to Settings > General Setup.
7. Ensure that the Master Node (admin1) is not attributed with the role of 'search head'. If so, edit the role to only reflect License Master and Cluster Master. (Figure 202)

Figure 202 Verify the Nodes

This instance								
i	Instance (host)	Instance (serverName)	Machine	Actions	Server roles	Custom groups	Status	State
>	admin1	admin1	admin1	Edit ▾	Cluster Master License Master		✓ Enabled	⚙ Configured

Verifying Master and Peer Replication

The purpose of this test is to ensure that indexes are distributed across each peer. By creating an index for testing, as well as a small retention time-frame, we will force the instance to push data into the archival directory.

1. SSH into the master node (admin1) as the 'splunk' user.
2. Navigate to `$$SPLUNK_HOME/etc/master-apps/_cluster/local/`
3. Create and edit the file 'indexes.conf'

```
[splunk@admin1 ~]$ cd $$SPLUNK_HOME/etc/master-apps/_cluster/local
[splunk@admin1 local]$ vi indexes.conf
```

4. Add the following stanzas:

```
### TESTING PURPOSES ONLY ###

[archival]

repFactor = auto

homePath   = $$SPLUNK_DB/archival/db
coldPath   = $$SPLUNK_DB/archival/colddb
thawedPath = $$SPLUNK_DB/archival/thaweddb

maxDataSize = 1024

frozenTimePeriodInSecs = 3600

coldToFrozenDir = /data/frzn_data/archival/frzn
```

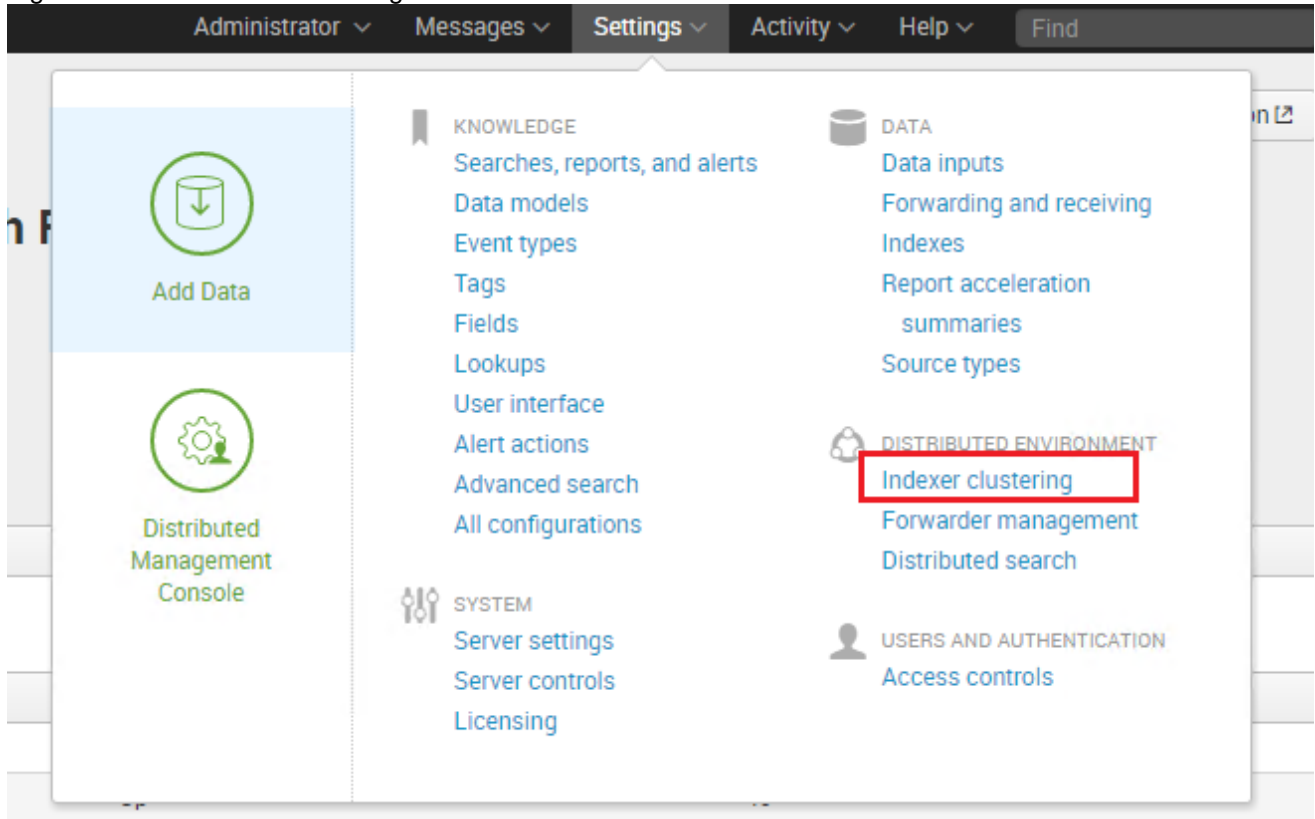
```
### TESTING PURPOSES ONLY ###
[archival]
repFactor = auto
homePath   = $$SPLUNK_DB/archival/db
coldPath   = $$SPLUNK_DB/archival/colddb
thawedPath = $$SPLUNK_DB/archival/thaweddb
maxDataSize = 1024
frozenTimePeriodInSecs = 3600
coldToFrozenDir = /data/frzn_data/archival/frzn
```



Note: This test index configuration make use of the frozen data path that was created in the earlier section. See NFS Configurations for the Splunk Frozen Data Storage. Save the file

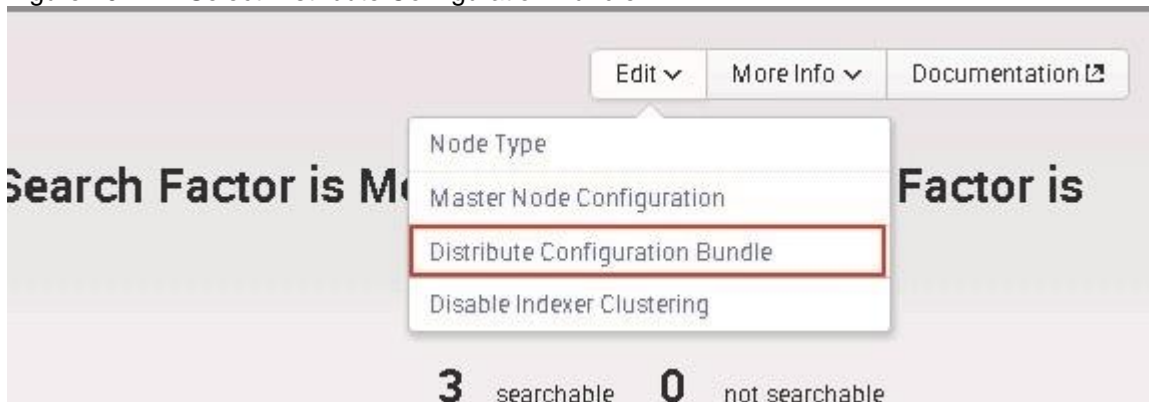
5. Log in to the Web Interface of the Master Node (<http://admin1:8000>).
6. Navigate to Settings -> Distributed Environment -> Indexer Clustering. (Figure 203)

Figure 203 Indexer Clustering



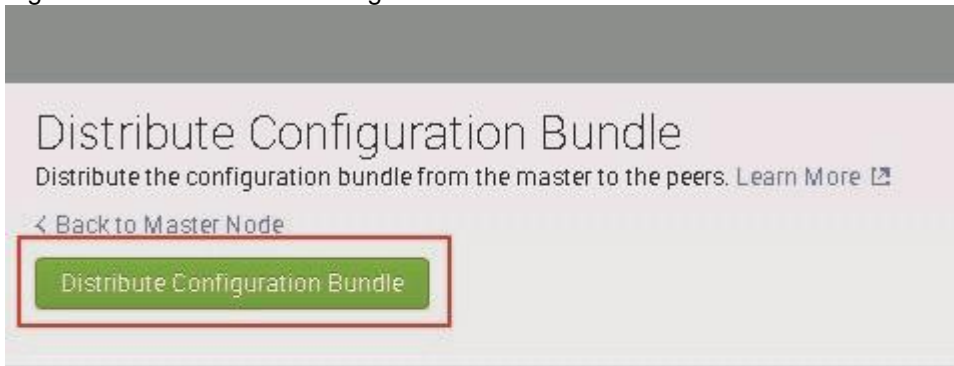
7. Click Edit > Distribute Configuration Bundle.

Figure 204 Select Distribute Configuration Bundle



8. Click Distribute Configuration Bundle.

Figure 205 Distribute Configuration Bundle



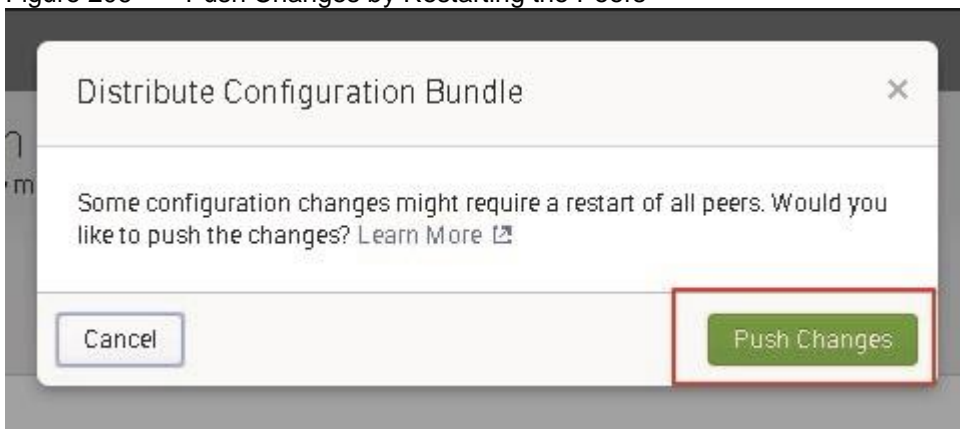
Last Push: ✓ Successful

Time 3/24/2015, 3:31:25 PM

Bundle ID ? 0DE75C59B77A4F1B3296FB0E75B1D750

9. A pop-up window will appear. Select **Push Changes**. (Figure 206)

Figure 206 Push Changes by Restarting the Peers



10. Verify that the push was successful.

Figure 207 Details of the Last Push



11. SSH to any indexer as the 'splunk' user.
12. Navigate to the directory '\$SPLUNK_HOME/etc/slave-apps/_cluster/local/'
13. Verify that the new 'indexes.conf' file exists.


```
[splunk@idx1 ~]$ cd $SPLUNK_HOME/etc/slave-apps/_cluster/local
[splunk@idx1 local]$ ls -l
total 12
-rw-rw-r-- 1 splunk splunk 274 Apr 18 04:19 indexes.conf
-rw-rw-r-- 1 splunk splunk  41 Apr 18 04:19 inputs.conf
-r--r--r-- 1 splunk splunk 231 Apr 18 04:19 README
[splunk@idx1 local]$ cat ./indexes.conf
### TESTING PURPOSES ONLY ###
[archival]
repFactor = auto
homePath   = $SPLUNK_DB/archival/db
coldPath   = $SPLUNK_DB/archival/colddb
thawedPath = $SPLUNK_DB/archival/thaweddb
maxDataSize = 1024
frozenTimePeriodInSecs = 3600
coldToFrozenDir = /data/frzn_data/archival/frzn
```

Verifying Data Replication

Next, verify that data is distributed across indexer nodes and is replicated across 'live' nodes when an indexer is down. In order to verify that the indexers are replicating data, the indexers must have a sample set of data to work with.

Any random syslog or file in which each line is a new event is acceptable. It is suggested that syslog data be used for verification due to the known and expected format. The recommended file size is at minimum ~250MB or 1million events. If a file is not available for testing, one may be found [here](#). Alternatively, you may use a random syslog generator.

Default throughput of universal forwarders is 256KBps. This may be increased by editing the "maxKBps" stanza in `$SPLUNK_HOME/etc/system/local/limits.conf`. Consult the [documentation](#) for more information. When testing, larger max KBps rates may be used (this configuration tested with 10240) but this may not be suitable for all environments depending upon network infrastructure.

Previously a universal forwarder was configured in the section [Configure the Universal Forwarder](#). It must be accessible from the same network which UCS is attached to.

Once the new system is available, follow the steps below:

1. From a command line interface on the universal forwarder system, enter the following command:

```
/opt/splunkforwarder/bin/splunk add oneshot -source ./your_test_file.log -
sourcetype syslog -index archival -auth admin:your_admin_password
```

```
[root@fwd1 data]# /opt/splunkforwarder/bin/splunk add oneshot -source ./test.log
-sourcetype syslog -index archival -auth admin:changeme
Warning: overriding $SPLUNK_HOME setting in environment ("/opt/splunk") with "/o
pt/splunkforwarder". If this is not correct, edit /opt/splunkforwarder/etc/splu
nk-launch.conf
Oneshot '/DATA/sdb/sbk/sbk/data/test.log' added
[root@fwd1 data]# █
```



Note: The `$SPLUNK_HOME` on universal forwarder may be set to `/opt/splunkforwarder` to avoid the warning seen in the screenshot above

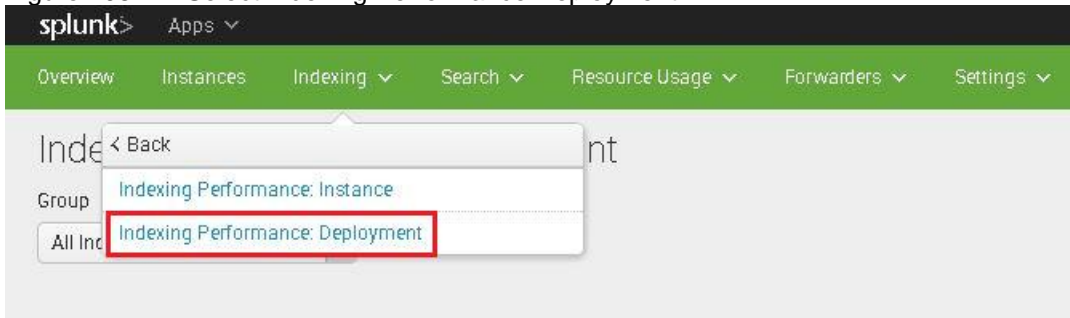
2. Screen will echo Oneshot '`your_test_file.log`' added.
3. Note the time that this step was executed.



Note: The time that this export was executed will be used to verify data transport to the archival node.

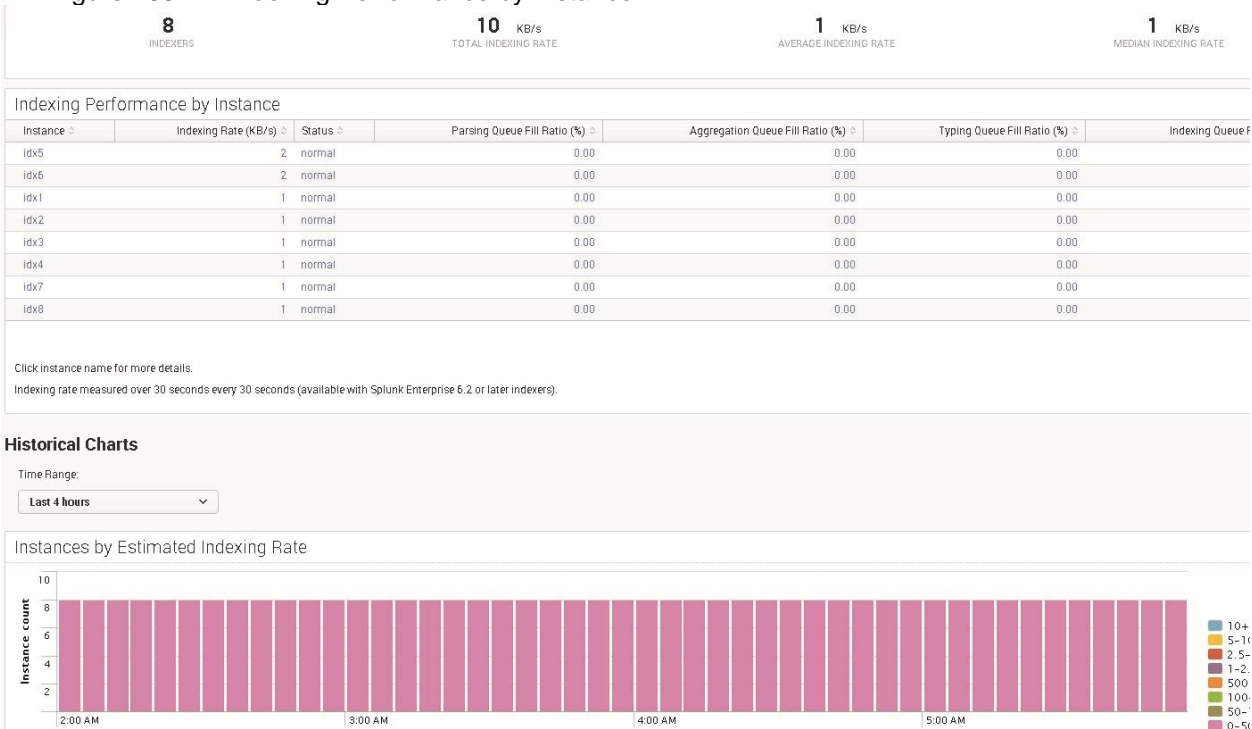
4. Issue the same command, but alter the stanza from “-index archival” to “-index main”. One dataset will be within two indexes.
5. Navigate to the DMC and your Master Node (admin1) in your browser by going to Settings -> Distributed Management Console.
6. Select Indexing > Performance > Indexing Performance Deployment.

Figure 208 Select Indexing Performance Deployment



Note that the DMC reflects indexing rates and data passing through the system.

Figure 209 Indexing Performance by Instance



7. Navigate to any of the search heads in your browser.
8. Click on Searching and Reporting.


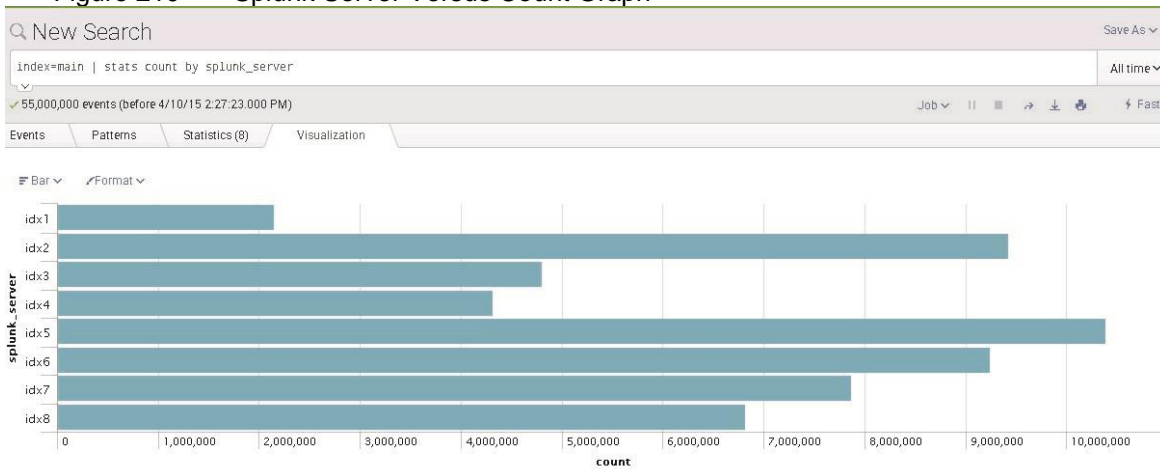
9. In the Splunk search bar, enter the following search: `index="archival" | stats count by splunk_server`
10. Note the indexer(s).
11. In the Splunk search bar, enter the following search: `index="main" | stats count by splunk_server`
12. Change the time range picker to 'All time'.
13. Change the search mode to 'fast mode'.
14. Click on 'search' (magnifying glass) .
15. Change view to 'Visualization' and set the chart type to 'Column'. Note the distribution of data across each of the indexers. Figure 210

Figure 210 Splunk Server Versus Count Graph



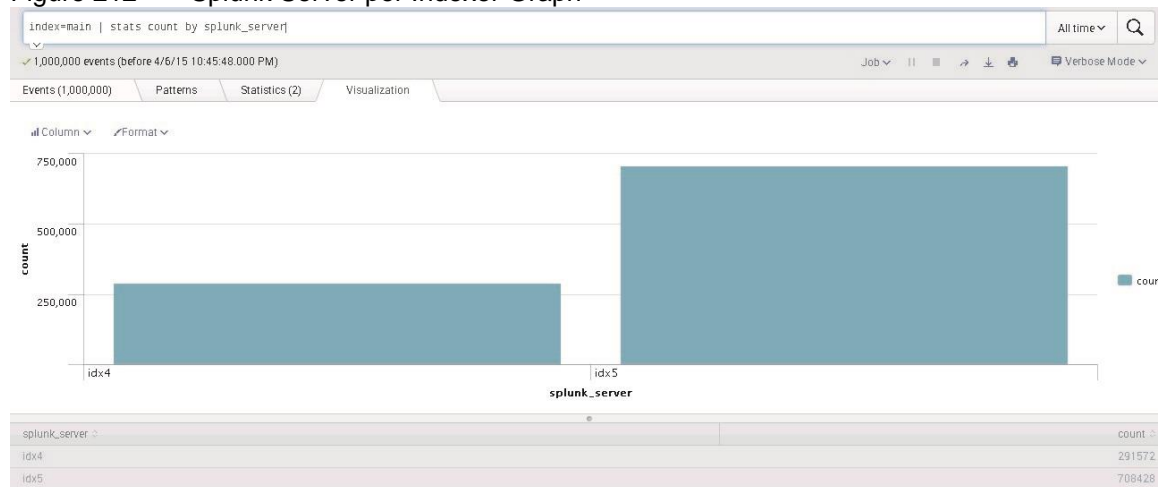
16. Change the view to 'Statistics'. Note the number of events per indexer, as well as the Total number of events, visible in the panel as well as the event summary under the search bar. Figure 211

Figure 211 Splunk Server Versus Count: Statistics

splunk_server	count	Total
idx1	629072	
idx3	59511	
idx4	311417	
	1000000	Total

17. Write down or take a screenshot of the totals per indexer for reference later.
18. Use shell access to navigate to one of the indexers which reported data.
19. Approximately 5 minutes after sending data to the indexers, proceed to step 24.
20. Issue the command `‘$SPLUNK_HOME/bin/splunk offline`.
21. Return to the browser and run the same search again. Figure 212

Figure 212 Splunk Server per Indexer Graph



Note: Jot down the distribution of events and the total. The event count from this search and the previous (step 18) should be the same. Bring your indexer back up (`$SPLUNK_HOME/bin/splunk start`).

While one indexer is down, this step has verified that the indexed data has been replicated and that the search results are consistent, even when an indexer is not functioning.

If the test does not present data across the indexers, the most common reasons for failure are listed below:

- The universal forwarder does not have the appropriate configurations listed in 'outputs.conf'
- There are network connectivity issues between the Universal Forwarder and the assigned receiving port on the indexers
- The dataset was large and has not finished replication to other indexers in the allotted amount of time

Verifying Transfer of Frozen Buckets to Archival Storage

In the previous test, the time that a 'oneshot' to the index 'archival' was performed was noted (Step 8). In this configuration, this setting is for one hour. For more information on frozen data, see [Configuring Archival of Data From Cold to Frozen](#).



Note: This is NOT a recommended setting, but simply for quick testing of archival transfer. You cannot verify the archival of frozen data until one hour has passed from the time of indexing.

1. As the 'splunk' user, SSH to the indexer which reported data ("verifying data replication: step 11")
2. Navigate to `/data/frzn_data/archival/frzn`

2. Issue the command:

```
ls -la
```

This displays the 'frozen' bucket(s) that have been moved.

```
[splunk@idx1 frzn]$ ls -la
total 12
drwxrwxrwx 4 splunk splunk 4096 Apr  6 21:19 .
drwxrwxrwx 4 splunk splunk  28 Feb  2 2010 ..
drwx--x--x 3 splunk splunk  20 Apr  6 19:16 db_1428343271_1428298175_0_ED614B15-7220-4
14
drwx--x--x 3 splunk splunk  20 Apr  6 21:19 db_1428354209_1428322606_1_ED614B15-7220-4
14
```



Note: If the selected indexer did not receive data, frozen buckets will not be present.

Post-Test Cleanup

Removing Test Data

To remove the data indexed during the test, complete the following steps:

1. Stop the Splunk service on the admin node. As the splunk user on admin1, issue the command:

```
$SPLUNK_HOME/bin/splunk stop
```

2. Stop all indexers:

```
clush --group=indexers $SPLUNK_HOME/bin/splunk stop
```

3. SSH to each indexer (idx1-idx8). As the splunk user, issue the command

```
$SPLUNK_HOME/bin/splunk clean eventdata -index main
```



Note: Alternatively, the clush command could be used to delete the indexes from all the peers at once by applying the force parameter '-f'. i.e. `clush --group=indexers $SPLUNK_HOME/bin/splunk clean eventdata -index main -f`. Use this command with **extreme caution** as this action can't be undone.

4. Confirm the cleaning of events *on every indexer!!*

5. Start the master node (admin1) as splunk user:

```
$SPLUNK_HOME/bin/splunk start
```

6. Start indexing peers as user 'splunk'

```
clush --group=indexers $SPLUNK_HOME/bin/splunk start
```

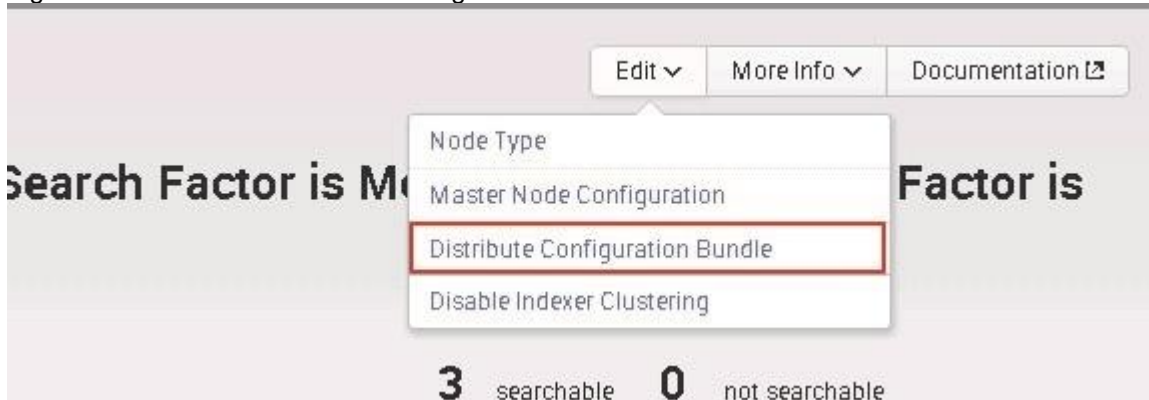
Removing Test Indexes

1. As user 'splunk', SSH into the master node (admin1)
2. Navigate to `$SPLUNK_HOME/etc/master-apps/_cluster/local/`
3. Remove the indexes.conf file. This file contains the 'archival' index and is not needed beyond testing.

```
[splunk@admin1 local]$ pwd
/data/disk1/splunk/etc/master-apps/_cluster/local
[splunk@admin1 local]$ ls -la
total 20
drwxr-xr-x 2 splunk splunk 4096 Apr 18 12:57 .
drwxr-xr-x 4 splunk splunk 4096 Apr  9 10:33 ..
-rw-rw-r-- 1 splunk splunk   9 Apr 18 12:57 indexes.conf
-rw-rw-r-- 1 splunk splunk  41 Apr 18 12:10 inputs.conf
-r--r--r-- 1 splunk splunk 231 Feb 18 15:01 README
[splunk@admin1 local]$ rm indexes.conf
[splunk@admin1 local]$ ls -la
total 16
drwxr-xr-x 2 splunk splunk 4096 Apr 18 12:58 .
drwxr-xr-x 4 splunk splunk 4096 Apr  9 10:33 ..
-rw-rw-r-- 1 splunk splunk  41 Apr 18 12:10 inputs.conf
-r--r--r-- 1 splunk splunk 231 Feb 18 15:01 README
[splunk@admin1 local]$
```

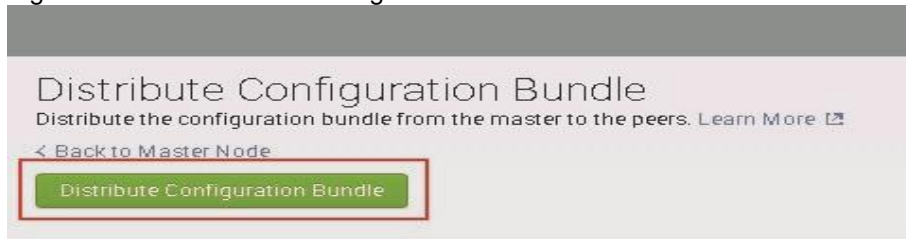
4. Using your browser, navigate to the master node web interface (admin1).
5. Select Settings > Indexer Clustering > Edit > Distribute Configuration Bundle. (Figure 213)

Figure 213 Select Distribute Configuration Bundle



6. Click Distribute Configuration Bundle. (Figure 214)

Figure 214 Distribute Configuration Bundle from Master to Peer



Last Push: ✓ Successful
 Time 3/24/2015, 3:31:25 PM
 Bundle ID ? 0DE75C59B77A4F1B3296FB0E75B1D750

7. Click `Push Changes`. This will remove the `indexes.conf` file which was created for testing purposes.

Removing the Universal Forwarder

1. Navigate to the host system of the Universal Forwarder
2. Stop the Universal Forwarder (`$SPLUNK_HOME/bin/splunk stop`)
3. Remove or uninstall the Universal Forwarder (actions will vary per OS)

Remove Deployment Server App

1. Via the CLI of the Deployment Server, navigate to `$SPLUNK_HOME/etc/deployment-apps`
2. Remove the directory 'outputTest'.

```
rm -r outputTest/
```

3. Reload the deployment Server.

```
$SPLUNK_HOME/bin/splunk reload deploy-server
```

4. Navigate to the URL of the Deployment Server.
5. Select `Settings > Forwarder Management`.
6. Under the tab `Server Classes`, click `Edit > Delete`.

The Splunk Enterprise installation is now in a 'clean' state, with no test data or forwarders.

Hardening the Splunk Installation

Taking the proper steps to secure Splunk Enterprise reduces its attack surface and mitigates the risk and impact of most vulnerabilities. We highly recommend you harden and tune the environment to your standards, so long as they do not overwrite configurations described within this document.

Turn Off Web Servers

Web connectivity should be limited to only those instances that require it. Web services should run on:

- Search Heads
- Distributed Management Console
- License Master

Web servers are not required to run on:

- Indexers

To disable web servers.

1. SSH into the instance of each indexer
2. As the 'splunk' user, issue the command `$SPLUNK_HOME/bin/splunk disable webserver --auth <username>:<password>`.

3. SSH into the master node (admin1)
4. Restart the indexing tier with the command:

```
$SPLUNK_HOME/bin/splunk rolling-restart cluster-peers
```

```
[splunk@admin1 local]$ $SPLUNK_HOME/bin/splunk rolling-restart cluster-peers
```

For up-to-date information regarding hardening the Splunk environment, visit [‘Securing Splunk Enterprise](#)

Best Practices for Onboarding Data

The Universal Forwarder

There are several methods to consider for collecting data into Splunk, otherwise referred to as 'onboarding'. Techniques include syslog forwarding, remote-polling techniques such as SNMP and WMI, writing of application logs to shared storage; batch uploads, and dedicated agents. Each of these approaches comes with limitations, and in many cases with additional costs for licensing or for computing overhead.

Optimal collection of data combines several factors: it has minimal overhead, supports myriad data sources including data that is not in log files, is securely transmitted, works with low bandwidth, sends in real time, and has scalable & robust delivery support. There should be centralized management of what's being collected.

To meet these goals, Splunk recommends the use of the Splunk Universal Forwarder (UF) on every server where this is possible. The Universal Forwarder is a centrally managed, lightweight tool for collecting and forwarding data to your Indexers, and it is available for installation on nearly all standard operating systems: Linux, Windows, OSX AIX, HP-UX, Solaris, FreeBSD, even Raspberry Pi.

Advantages of the Splunk Universal Forwarder

The use of the Universal Forwarder allows a platform-agnostic approach to managing data collection from your environment.

Here is how the Splunk UF meets each of the goals:

Minimal Overhead: The Splunk Universal Forwarder is a lightweight software package; its sole purpose is to collect and forward data. Unlike heavyweight agents, it does not analyze data locally for lowest local overhead.

Data sources: Like most other options, the UF can collect data from local syslog files (*NIX) and Event Logs (Windows). The Splunk UF can also read from virtually *any* local file source so long as it is in ASCII format. The UF can also collect data that does not exist on disk at all:

- For all data being forwarded, the UF provides meta-data for all data sources, including: hostname & time zone (per OS), source (typically the full file path), sourcetype, and routing information of destination indexes in Splunk.
- Each Universal forwarder can call shell, Python, or PowerShell scripts to monitor OS- and application-level usage; one example is to monitor and report on open network ports.
- Splunk Stream Forwarder can be configured to listen on network interfaces and collect protocol level data directly off the network stream. This is particularly useful when application logs lack the details necessary for your monitoring or analytics needs.
- The Splunk UF can listen on UDP or TCP ports directly, allowing applications to send application logs directly and avoid Disk I/O concerns. The UF routes this directly to the Indexers, removing the need to have compression / routing logic at the application layer.

Secure, low bandwidth: After collecting the raw data, Splunk uses data compression and optional SSL compression when sending the data to the Splunk indexers. SSL overhead is minimized by keeping TCP sessions open for set periods of time.

Real time: With the UF, Splunk can monitor and analyze data in near real time. As events are generated (for example, appended to a logfile), they are immediately forwarded to indexers, where they are typically available for analysis within a second or so of initial generation.

Data Delivery: The UF is designed with high availability and guaranteed delivery in mind. Delivery is over TCP rather than UDP, ensuring that the UF "knows" if the data was received or not. Every UF can be configured with one or more indexers as targets, automatically spreading the load of collected data across the indexers. When one or more indexers are off-line, the UF will automatically find an indexer that is available. If all

indexers are unavailable, The UF keeps track of last data sent – when an indexer becomes available, data transmission from where it left off.

Management: Splunk offers central management of the configuration of Universal Forwarders. Each UF connects to the Deployment server on a scheduled basis to check for new configurations, the Deployment server offers granular control over classes of systems that will collect any given data source. A Splunk administrator can change collection configurations and roll this out within minutes.

What About Systems Where the Splunk Universal Forwarder is Not Supported?

Networking and Storage gear, virtual appliances, and other “non-OS” devices are a vital part of any company’s environment, and should be monitored for performance and reliability. When the Splunk Universal Forwarder cannot be installed locally, here are a few recommended options to consider.

IPFIX/NetFlow: Most networking equipment (physical or virtual) supports either IPFIX, NetFlow, sflow or jflow for pushing out performance or security data. Systems sending on these protocols are called “exporters”. IPFIX and *Flow are binary protocols and cannot be sent directly to Splunk. Recommended approaches (select one):

- Have the exporters send their data to a system running the Splunk Universal Forwarder with the NetFlow or IPFIX TAs. These TAs translate the protocol from binary to ASCII.
- Have the exporters send their data to a 3rd party NetFlow/IPFIX parser, such as the NetFlow Integrator by NetFlow Logic. These systems accept binary data in, convert the data to syslog, and send out over the network. Install a Splunk UF on the same system, listening for network data streamed out of the middleware.

SNMP (polling): SNMP provides a valuable method for remotely collecting information from devices without a “normal” OS, such as network switches and routers, and on hardware management ports of physical server hardware. Recommended approaches (select one):

- Set up a Splunk heavy forwarder with the SNMP modular input app. The app will poll SNMP data and store it directly in Splunk. Details are in the app’s documentation. (Simply install on the Splunk search head for smaller deployments.)
- On any system where a Splunk UF could be installed, use an SNMP polling agent to collect data as necessary, and output the results to a log file. The UF can then collect the output files in the same manner as any other log file. The SNMP polling agent might be a commercial tool for this purpose, or something as simple as the ‘snmpwalk’ command running from a script.

SNMP (traps): SNMP traps are sent on alert conditions, typically by network devices. Recommended approaches (select one):

- Set SNMP devices to send their traps to a system running Splunk Universal Forwarder and the Splunk for Stream app. Configure Stream to listen for the SNMP protocol, forwarding whichever SNMP data is required.
- Set SNMP devices to send their traps to a system capable of running an SNMP daemon and a Splunk UF. Configure the SNMP daemon to log traps to a file, configure the UF to read the logs.
- **Syslog Forwarding:** Many devices, virtual appliances, and bare-metal hypervisors offer the ability to send critical information via Syslog. (Linux and UNIX family OSes do to – but those systems support UF installation.) Recommended approach:
 - Configure a system that runs a supported syslog server to listen for syslog data. (“syslog-ng” and “rsyslog” are excellent free options for Linux systems.)
 - Configure the log servers to store logs in host-specific folders.

- When possible, configure syslog senders to use TCP rather than UDP. This ensures that critical data will not be dropped.
- Install the Splunk Universal Forwarder or heavy forwarder on the system, and configure it to monitor the log files. Tell Splunk to extract the hostnames from the file paths. (Heavy forwarder is necessary for certain syslog streams, such as ESXi data.) Install additional TAs as recommended by documentation, depending on syslog data sources.
- Optionally, create two syslog collection systems – and put them behind a load balancer. Have the syslog sources send to the load balancer via TCP. This ensures that if a single syslog server is down, the data will still continue coming to Splunk in real time.

Proprietary APIs: There are a large number of computing infrastructure components that only provide the full set of information when polled through API calls. These include network, storage, power system controllers, and other devices. A few specific examples include VMware vCenter servers, NetApp OnTap filers, Checkpoint firewalls. Because these systems provide a piece of the overall infrastructure picture for performance and security, bringing this data into Splunk is important for many Splunk customers. There are many approaches available, here is a recommended methodology for getting this data in:



Note: Do not use the “find more apps” function within the Splunk UI.

- Check on splunkbase.splunk.com for an app that is designed to handle the technology. For example, search for “cisco”.
- If an app exists – read the documentation for that app.
- If an app does not exist on Splunkbase, simply perform an Internet search for “Splunk” and the technology.
- If all else fails, contact Splunk support to ask for suggestions.

Additional Terminology

When onboarding data, Splunk provides a number of apps and add-ons via splunkbase.splunk.com. It is imperative that the Splunk administrator is familiar with the following terms:

Apps: An application that runs on Splunk Enterprise and typically addresses several use cases. An app typically contains both components of a Technology add-on and a Search add-on. An app contains one or more views. An app can include various Splunk Enterprise knowledge objects such as reports, lookups, scripted inputs, and modular inputs. An app sometimes depends on one or more add-ons for specific functionality. Examples of apps are the Splunk Enterprise Search app, the Splunk on Splunk app, and the Splunk Enterprise Security app.

Technology Add-on (TA): A technology add-on is a Splunk app that extracts knowledge from IT data so that it can be processed by Splunk, as well as other apps that leverage the Common Information Model (CIM). The technology add-on may pull data into Splunk or simply map data that is already coming in. Technology add-ons may conflict with or duplicate other Splunk apps that are already pulling in the same sort of data if they disagree on the source type. The difference between a technology add-on and another Splunk app is compliance with the Common Information Model. Technology add-ons typically reside on the universal forwarder or on the indexing tier.

Search Add-on (SA): A search add-on is a Splunk app that contains pre-built dashboards, searches, lookups, forms, and various search components. The difference between a search add-on and a technology add-on is that SAs are primarily focused on visualizing data. Search add-ons exist on the search head(s).

Common Information Model (CIM): The Common Information Model Add-on is based on the idea that you can break down most log files into two components:

- fields

- event category tags

With these two components a knowledge manager can set up their log files in a way that makes them easy to process by Splunk and which normalizes noncompliant log files and forces them to follow a similar schema. The Common Information Model details the standard fields and event category tags that Splunk uses when it processes most IT data.

The Common Information Model is an overlay function and does not normalize or overwrite the raw data, it categorizes various fields into corresponding categories.

Recommended Apps and Add-ons for Data Collection

Here are the most commonly deployed add-ons, and what they collect.

You will find these add-ons at <https://splunkbase.splunk.com>; for each, you will also see a complete description as well as the documentation on how to install them. For some, these add-ons are installed to the forwarders, in some cases they are installed to the indexers, or to both.

Splunk Technical Add-on for Cisco UCS: Splunk's first (and only) supported integration for server environments provides real-time operational visibility across multiple Cisco UCS domains and enables our joint customers to identify & resolve problems faster, proactively monitor systems & infrastructure, track key performance indicators & understand trends & patterns of activity & behavior.

The app grabs UCS faults, events, performance statistics such as temperature, power and network throughput from one or more Cisco UCS Managers to:

- Deliver real time and historical visibility centrally across your entire UCS deployment
- Provide analytics such as available capacity, trending of faults over time, tracking of power & cooling costs.
- Correlate UCS performance, fault and events data with user, application, and hypervisor data to analyze, prevent and fix problems across broad infrastructure or application environments.

Splunk Add-on for Unix and Linux: This add-on includes predefined inputs to collect data from *NIX systems, and maps to normalize the data to the Common Information Model. It provides easy collection from standard system log directories (such as /var/log), and excludes collection of common binary files. Examples are provided for monitoring the contents of specific files, such as /etc/hosts. Scripted inputs are included to monitor a variety of OS performance and network data points.

Splunk App for Stream: This provides a scalable and easy-to-configure solution to capture real-time streaming wire data from anywhere in your datacenter through protocol-level inspection. Stream data is useful for IT Ops, DevOps, and Security use cases. The Stream forwarder can run directly on endpoint servers – no need for SPAN/TAP ports; this is particularly useful in public cloud environments where SPAN/TAP are not an option. Capture only the relevant wire data for analytics, through filters and aggregation rules. Manage wire data volumes with fine-grained precision by selecting or deselecting protocols and associated attributes within the App interface.

Splunk DB Connect 2: Enrich your data results by accessing the data stored in your database servers. Splunk can access your structured data on-demand, for providing supplemental information, or on a monitoring basis where Splunk indexes the new data in selected tables. Use the Outputs function to export Splunk results into your legacy database.

Splunk Support for Active Directory / Idapsearch: Enrich your data results by reading data stored in your LDAP directory servers, including Active Directory. Use cases include mapping host names to additional information, mapping user names to HR information, or accessing asset management information stored in LDAP.

Splunk add-on for Microsoft Windows: This add-on includes predefined inputs to collect data from Windows systems, and maps to normalize the data to the Common Information Model. Supported data includes performance data, event logs, commonly used log files, and Windows Registry content. Scripted inputs are included to monitor open Network ports and installed applications.

Splunk App for Windows Infrastructure: The Splunk App for Windows Infrastructure provides examples of pre-built data inputs, searches, reports, and dashboards for Windows server and desktop management. You can monitor, manage, and troubleshoot Windows operating systems, including Active Directory elements, all from one place. The App also contains dashboards needed to monitor your Active Directory environment and allows for correlation opportunities from the Active Directory data back to the Operating System.

For a complete list of all Splunk supported apps go to:

<https://splunkbase.splunk.com/apps/#/order/latest/author/splunk>.

Conclusion

Splunk Enterprise delivers operational visibility and digital intelligence by monitoring all machine generated data and making it accessible, usable and valuable across the organization. Cisco UCS Integrated Infrastructure for Big Data with its compute, storage, connectivity, and unified management features, streamlines the deployment and offers dependable, scalable integrated infrastructure that delivers predictable performance and high-availability for your Splunk Enterprise platform with a lower TCO.

The configuration detailed in the document can be extended to clusters of various sizes depending on application demands. Up to 80 servers (4 racks) can be supported with no additional switching in a single UCS domain without any network over-subscription. Scaling beyond 4 racks (80 servers) can be implemented by interconnecting multiple UCS domains using Nexus 6000/7000 Series switches, scalable to thousands of servers and to hundreds of petabytes storage, and managed from a single pane using [UCS Central](#).

Bill of Materials

Table 12 provides the BOM for a 13 node Splunk Enterprise cluster plus 1 Cisco UCS C3260 as an archival node. Table 11

Table 11 Bill of Materials

Part Number	Description	Quantity
Cisco UCS C220 M4 Rack Server configuration for Splunk search heads		
UCSC-C220-M4S	UCS C220 M4 SFF w/o CPU, mem, HD, PCIe, PSU, rail kit	3
CON-OSP-C220M4S	SNTC-24X7X4OS UCS C220 M4 SFF w/o CPU, mem, HD	3
UCSC-MLOM-CSC-02	Cisco UCS VIC1227 VIC MLOM - Dual Port 10Gb SFP+	3
UCSC-RAILB-M4	Ball Bearing Rail Kit for C220 M4 and C240 M4 rack servers	3
UCSC-PSU1-770W	770W AC Hot-Plug Power Supply for 1U C-Series Rack Server	6
CAB-N5K6A-NA	Power Cord, 200/240V 6A North America	6
UCSC-SCCBL220	Supercap cable 950mm	3
N20-BBLKD	UCS 2.5 inch HDD blanking panel	18
UCSC-HS-C220M4	Heat sink for UCS C220 M4 rack servers	6
UCSC-MRAID12G	Cisco 12G SAS Modular Raid Controller	3
UCSC-MRAID12G-2GB	Cisco 12Gbps SAS 2GB FBWC Cache module (Raid 0/1/5/6)	3
C1UCS-OPT-OUT	Cisco ONE Data Center Compute Opt Out Option	3
UCS-CPU-E52680E	2.40 GHz E5-2680 v4/120W 14C/35MB Cache/DDR4 2400MHz	6
UCS-MR-1X322RV-A	32GB DDR4-2400-MHz RDIMM/PC4-19200/dual rank/x4/1.2v	24
UCS-HD600G10K12G	600GB 12G SAS 10K RPM SFF HDD	6
UCS-M4-V4-LBL	Cisco M4 - v4 CPU asset tab ID label (Auto-Expand)	3
Cisco UCS C240 M4 Rack Server configuration for Splunk indexers		
UCSC-C240-M4SX	UCS C240 M4 SFF 24 HD w/o CPU,mem,HD,PCIe,PS,railkt w/expndr	8

Part Number	Description	Quantity
CON-OSP-C240M4SX	SNTC-24X7X4OS UCS C240 M4 SFF 24 HD w/o CPU,mem	8
UCSC-RAILB-M4	Ball Bearing Rail Kit for C220 M4 and C240 M4 rack servers	8
UCSC-HS-C240M4	Heat sink for UCS C240 M4 rack servers	16
UCSC-SCCBL240	Supercap cable 250mm	8
UCSC-MRAID12G	Cisco 12G SAS Modular Raid Controller	8
UCSC-MRAID12G-2GB	Cisco 12Gbps SAS 2GB FBWC Cache module (Raid 0/1/5/6)	8
UCSC-PCI-1C-240M4	Right PCI Riser Bd (Riser 1) 2onbd SATA bootdrvs+ 2PCI slts	8
UCSC-MLOM-CSC-02	Cisco UCS VIC1227 VIC MLOM - Dual Port 10Gb SFP+	8
CAB-N5K6A-NA	Power Cord, 200/240V 6A North America	16
C1UCS-OPT-OUT	Cisco ONE Data Center Compute Opt Out Option	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-SD120GBKS4-EB	120 GB 2.5 inch Enterprise Value 6G SATA SSD (boot)	16
UCSC-PSU2-1400W	1400W AC Power Supply (200 - 240V) 2U & 4U C Series Servers	16
UCS-M4-V4-LBL	Cisco M4 - v4 CPU asset tab ID label (Auto-Expand)	8
UCS-HD18TB10KS4K	1.8 TB 12G SAS 10K RPM SFF HDD (4K)	192
Cisco UCS C220 M4 Rack Server configuration for Splunk admin nodes		
UCSC-C220-M4S	UCS C220 M4 SFF w/o CPU, mem, HD, PCIe, PSU, rail kit	3
CON-OSP-C220M4S	SNTC-24X7X4OS UCS C220 M4 SFF w/o CPU, mem, HD	3
UCSC-HS-C220M4	Heat sink for UCS C220 M4 rack servers	6
UCSC-MRAID12G	Cisco 12G SAS Modular Raid Controller	3
UCSC-MRAID12G-2GB	Cisco 12Gbps SAS 2GB FBWC Cache module (Raid 0/1/5/6)	3
CAB-N5K6A-NA	Power Cord, 200/240V 6A North America	6
UCSC-SCCBL220	Supercap cable 950mm	3

Part Number	Description	Quantity
N20-BBLKD	UCS 2.5 inch HDD blanking panel	18
UCSC-MLOM-CSC-02	Cisco UCS VIC1227 VIC MLOM - Dual Port 10Gb SFP+	3
UCSC-RAILB-M4	Ball Bearing Rail Kit for C220 M4 and C240 M4 rack servers	3
UCSC-PSU1-770W	770W AC Hot-Plug Power Supply for 1U C-Series Rack Server	6
C1UCS-OPT-OUT	Cisco ONE Data Center Compute Opt Out Option	3
UCS-CPU-E52620E	2.10 GHz E5-2620 v4/85W 8C/20MB Cache/DDR4 2133MHz	6
UCS-MR-1X322RV-A	32GB DDR4-2400-MHz RDIMM/PC4-19200/dual rank/x4/1.2v	12
UCS-HD600G10K12G	600GB 12G SAS 10K RPM SFF HDD	6
UCS-M4-V4-LBL	Cisco M4 - v4 CPU asset tab ID label (Auto-Expand)	3
RACK-UCS2	Cisco R42610 standard rack, w/side panels	1
CON-SNT-R42610	SNTC-8X5XNBD Cisco R42610 expansion rack, no side pan	1
RP208-30-1P-U-2	Cisco RP208-30-U-2 Single Phase PDU 20x C13, 4x C19	2
CON-SNT-RPDUX	SNTC-8X5XNBD Cisco RP208-30-U-X Single Phase PDU	2
CVR-QSFP-SFP10G=	QSFP to SFP10G adapter	2
Cisco UCS C3260 Rack Server configuration for Splunk archival node		
UCSC-C3260	Cisco UCS C3260 Base Chassis w/4x PSU, SSD, Railkit	1
CON-OSP-C3260BSE	SNTC-24X7X4OS Cisco UCS C3260 Base Chassis w/4x PSU, 2x120GB	1
CAB-N5K6A-NA	Power Cord, 200/240V 6A North America	4
UCSC-C3160-BEZEL	Cisco UCS C3160 System Bezel	1
UCSC-C3X60-SBLKP	UCS C3x60 SIOC blanking plate	1
UCSC-C3X60-RAIL	UCS C3X60 Rack Rails Kit	1
UCSC-PSU1-1050W	UCS C3X60 1050W Power Supply Unit	4
N20-BBLKD-7MM	UCS 7MM SSD Blank Filler	3
UCSC-C3K-M4SRB	UCS C3000 M4 Server Node for Intel E5-2600 v4	1
UCS-CPU-E52620E	2.10 GHz E5-2620 v4/85W 8C/20MB Cache/DDR4 2133MHz	2

Part Number	Description	Quantity
UCS-MR-1X322RV-A	32GB DDR4-2400-MHz RDIMM/PC4-19200/dual rank/x4/1.2v	8
UCS-C3K-M4RAID	Cisco UCS C3000 RAID Controller M4 Server w 4G RAID Cache	1
UCSC-HS-C3X60	Cisco UCS C3X60 Server Node CPU Heatsink	2
UCSC-C3260-SIOC	Cisco UCS C3260 System IO Controller with VIC 1300 incl.	1
UCSC-C3X60-EX32T	UCS C3X60 Expander with 4x 8TB 7200RPM NL-SAS Drives	1
UCSC-C3X60-8TBRR	UCSC 3X60 8TB NL-SAS 7.2K Helium HDD rear with HDD Carrier	4
UCS-C3X60-G2SD12	UCSC C3X60 120GB Boot SSD (Gen 2)	1
UCSC-C3X60-56HD8	UCS C3X60 4 rows of 8TB NL-SAS7200 RPM SAS-3 (56Total) 448TB	1
UCSC-C3X60-HD8TB	UCSC 3X60 8TB NL-SAS 7.2KHelium HDD with HDD Carrier	56
RHEL-2S2V-3A=	Red Hat Enterprise Linux (1-2 CPU,1-2 VN); 3-Yr Support Req	1
CON-ISV1-EL2S2V3A	ISV 24X7 RHEL Server 2Socket-OR-2Virtual; ANNUAL List Price	1
UCS-RHEL-TERMS	Acceptance of Terms, Standalone RHEL License for UCS Servers	1
SFP-H10GB-CU5M=	10GBASE-CU SFP+ Cable 5 Meter	36

About Authors

Raghunath Nambiar, Distinguished Engineer, Data Center Business Group (Cisco Systems)

Raghunath Nambiar is a Distinguished Engineer at Cisco's Data Center Business Group. His current responsibilities include emerging technologies and big data strategy.

Karthik Karupasamy, Technical Marketing Engineer, Data Center Solutions Group (Cisco Systems)

Karthik Karupasamy is a Technical Marketing Engineer in Data Center Solutions Group at Cisco Systems. His main focus areas are architecture, solutions and emerging trends in big data related technologies and infrastructure in the Data Center.

Dennis Bourg, Senior Sales Engineer, Global Strategic Alliances Group, Splunk

Dennis Bourg is a Senior Sales Engineer in the Global Strategic Alliances group of Splunk. He focuses on technology alignment and innovation between Splunk and partners, in particular partners in IT Operations and Infrastructure.

Acknowledgements

The authors acknowledge contributions of Karthik Kulkarni, Manan Trivedi, Friea Berg (Splunk), Michael Donnelly (Splunk), Simeon Yep (Splunk) , Barbara Dixon, and Sindhu Sudhir in developing this document.

Appendix A

Provisioning a Splunk Cluster with UCSDE

An alternative to manually deploying a Splunk cluster as outlined in this CVD is to deploy it through Cisco UCS Director Express. Cisco UCS Director Express for Big Data provides a single-touch solution that automates Hadoop deployment on the Cisco UCS Common Platform Architecture (CPA) for Big Data infrastructure. It also provides a single management pane across both physical infrastructure and Hadoop software. All elements of the infrastructure are handled automatically with little user input.

The steps outlined in this section assume that you have the Fabric Interconnects configured according to the Fabric Configuration section and have enabled the server and uplink ports. The servers have undergone the discovery process and all necessary devices have assigned IP addresses on the management network.

Requirements

- Two virtual machines (VM) on an ESXi server to use for UCS Director Express (UCSDE) and the Baremetal Agent (BMA)
- UCSDE minimum requirements: 4 vCPUs, 12GB RAM, 100GB thick-provisioned storage
- BMA minimum requirements: 2 vCPUs, 3GB RAM, 40GB thick-provisioned storage



Note: Reserve 4000 MHz and 12GB RAM for the UCSDE VM and 2000 MHz and 3GB RAM for the Baremetal Agent VM.

- Three separate networks for Management, PXE, and Splunk index replication traffic. The PXE network should not have any other PXE/DHCP server present.
- Red Hat 6.x ISO file
- UCS Director Express software, available from the Cisco.com download site for UCS Director
 - CUCSD_Express_3_0_0_0_GA.zip
 - CUCSD_BMA_6_0_0_0_VMWARE_GA.zip
- Any necessary patches
- clustershell-1.7-1.el6.noarch.rpm (available at http://dl.fedoraproject.org/pub/epel/6/x86_64/clustershell-1.7-1.el6.noarch.rpm)
- Splunk software and license

Creating MAC Address Pools

Create three MAC address pools for the three server roles. To create MAC address pools, complete the following steps:

1. Select the LAN tab of the navigation pane (the left pane in the UCS Manager GUI).
2. Select Pools > root. If you created an organization, you may select that under Sub-Organizations instead of root.
3. Right-click MAC Pools under the root organization.

4. Select `Create MAC Pool` to create the MAC address pool.
5. Enter the MAC Pool name in the `Name` field. Here we will name them `Mgmt`, `Data1`, and `Data2`.
6. (Optional) Enter a description of the MAC pool.
7. Select `Assignment Order` to be `Sequential`.

Creating MAC Pool Window

Figure 215 Add MAC Address

Create MAC Pool

Unified Computing System Manager

Create MAC Pool

1. **Define Name and Description**
2. Add MAC Addresses

Define Name and Description

Name:

Description:

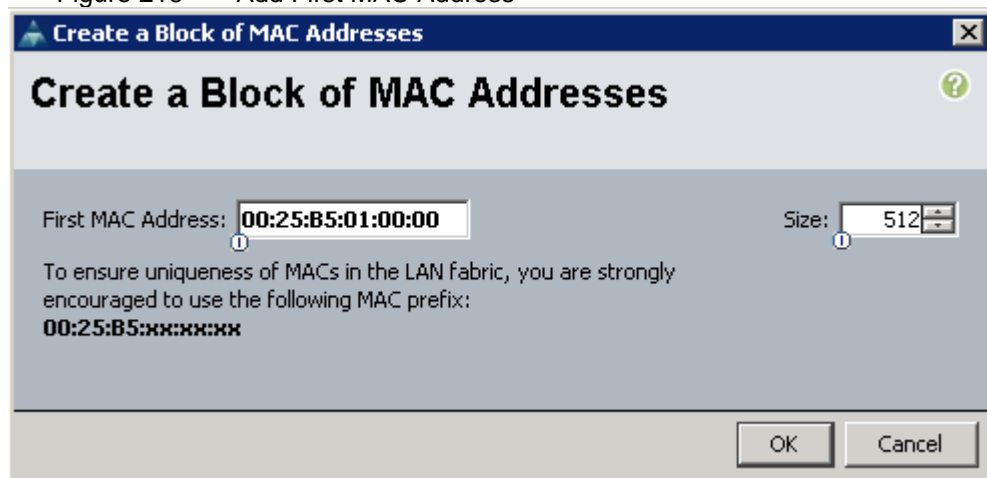
Assignment Order: Default Sequential

< Prev Next > Finish Cancel

8. Click `Next`.
9. Click `Add`.
10. Specify a starting MAC address, such that the three MAC address pools will not overlap.
11. Specify a size of the MAC address pool, which is sufficient to support the available server resources. (Figure 216)

Specifying First MAC Address and Size

Figure 216 Add First MAC Address



12. Click **OK**.
14. Click **Finish**.
14. When the message box displays, click **OK**.
15. Repeat the above steps to create the other two MAC address pools.

Figure 217 All Three MAC Pools

Name	Size	Assigned
MAC Pool Data1 [00:25:B5:02:00:00 - 00:25:B5:02:01:FF]	512	0
MAC Pool Data2 [00:25:B5:03:00:00 - 00:25:B5:03:01:FF]	512	0
MAC Pool Mgmt [00:25:B5:01:00:00 - 00:25:B5:01:01:FF]	512	0
MAC Pool default	0	0

Creating Server Pools

Create server pools for the indexers, search heads, and administrative nodes. Follow these steps to configure the server pool within the Cisco UCS Manager GUI:

1. In the navigation pane of UCS Manager, click the **Servers** tab.
2. Expand **Pools > root**. If you created an organization, you may select that under **Sub-Organizations** instead of **root**.
3. Right-click **Server Pools**.
4. Select **Create Server Pool**.
5. Enter the server pool name, which is **indexer** (the other server pools will be named **search** and **admin**).
6. (Optional) Enter a description for the server pool.

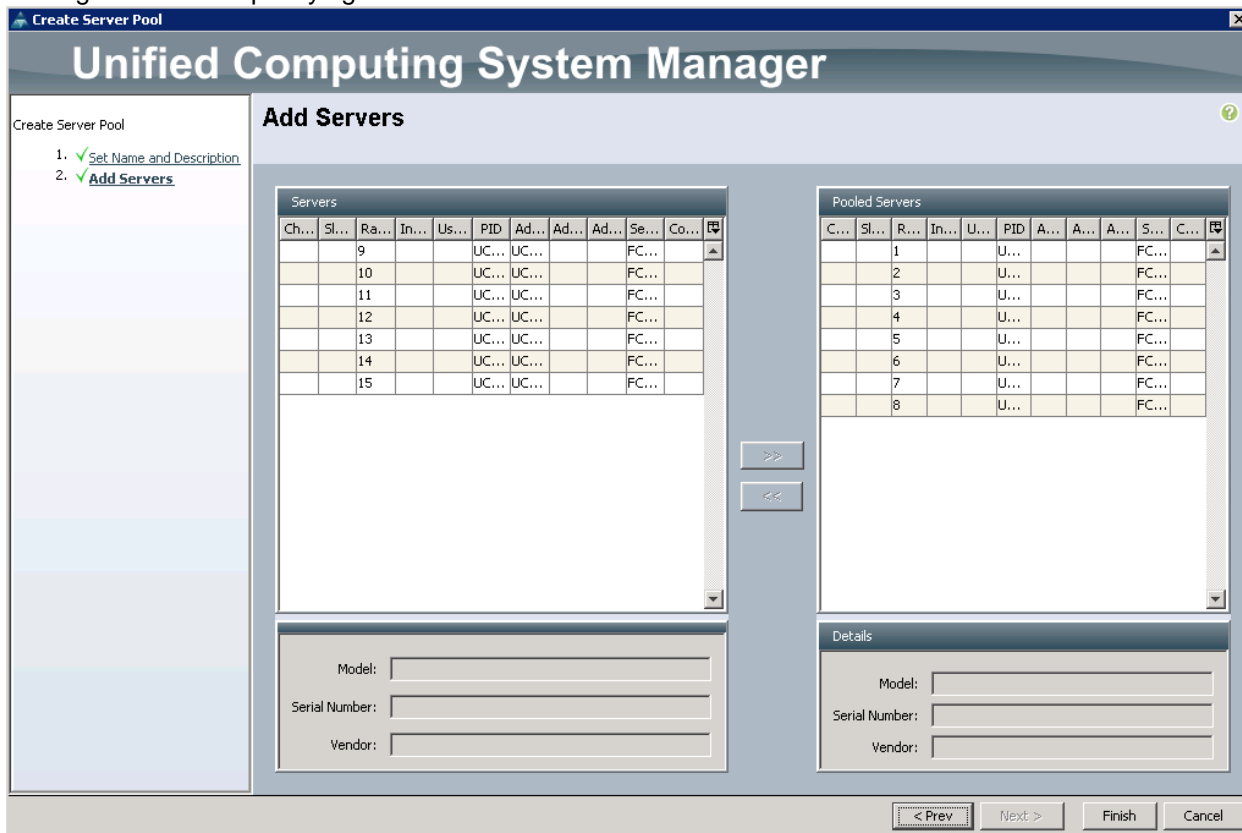
7. Click **Next** to add the servers.

Figure 218 Naming a Server Pool

The screenshot shows a window titled "Create Server Pool" from the "Unified Computing System Manager". The main heading is "Set Name and Description". On the left, a sidebar shows a progress list: "1. ✓ Set Name and Description" and "2. Add Servers". The main area contains two text input fields: "Name:" with the value "Indexer" and "Description:" with the value "8 Splunk indexers". At the bottom right, there are four buttons: "< Prev", "Next >", "Finish", and "Cancel".

8. For the indexer pool, select the 8 Cisco UCS C240 M4 servers to be added to the server pool. For the search pool, select 3 Cisco UCS C200 M4 servers. For the admin pool, select 2 Cisco UCS C200 M4 servers. Click **>>** to add them to the pool.
9. Click **Finish**.

Figure 219 Specifying Servers to be Added to the Server Pool



10. The `Create Server Pools` dialog box confirms that the server pool was successfully created. Click `OK`.

11. Repeat the steps above two more times to create the search and admin server pools.



Note: Though we support any type of servers, the 1RU servers are the appropriate ones for the search-head and admin pools.

Table 12 shows a summary of the server pool names and their corresponding server types.

Table 12 Server Pool Names and Server Types

Server Pool Name	Role	Server Type
Indexer	Indexer	Cisco C240 M4
Search	Search Head	Cisco C220 M4
Admin	Administration Node	Cisco C220 M4

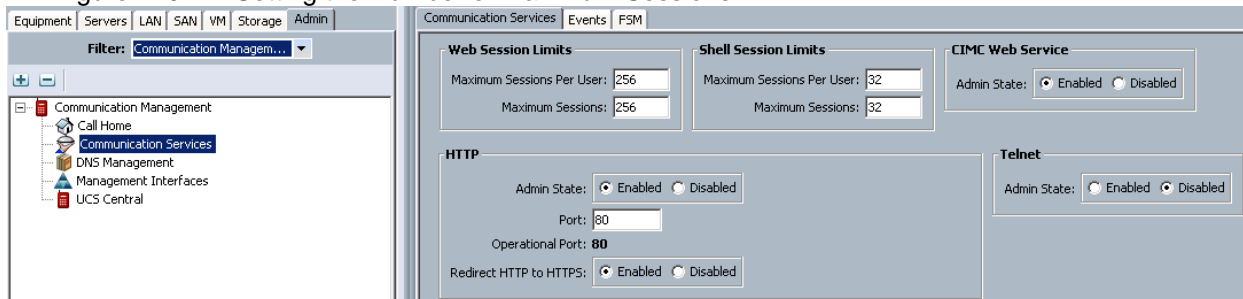
Communication Settings

Web session limits are used by Cisco UCS Manager to restrict the number of web sessions (both GUI and XML) that are permitted access to the system at any one time. We will set the possible number of concurrent HTTP and HTTPS sessions allowed for all users within the system to the maximum of 256.

1. In the navigation pane, click the `Admin` tab.
2. Select `Communication Management` in the `Filter` drop-down menu.

3. Expand Communication Management > Communication Services.
4. In the Web Session Limits section, change the Maximum Sessions to 256. (Figure 220)

Figure 220 Setting the Number of Maximum Sessions

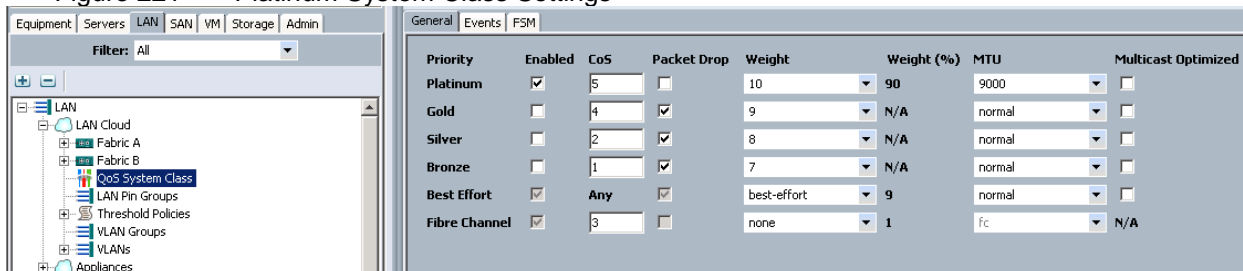


QOS System Class

We will make the Platinum system class available for assigning custom settings and policies.

1. In the navigation pane, click the LAN tab.
2. Expand LAN > LAN Cloud.
3. Click the QOS System Class node.
4. In the General tab, check the Enabled box for Platinum and set its MTU to 9000.
5. Click Save Changes.

Figure 221 Platinum System Class Settings



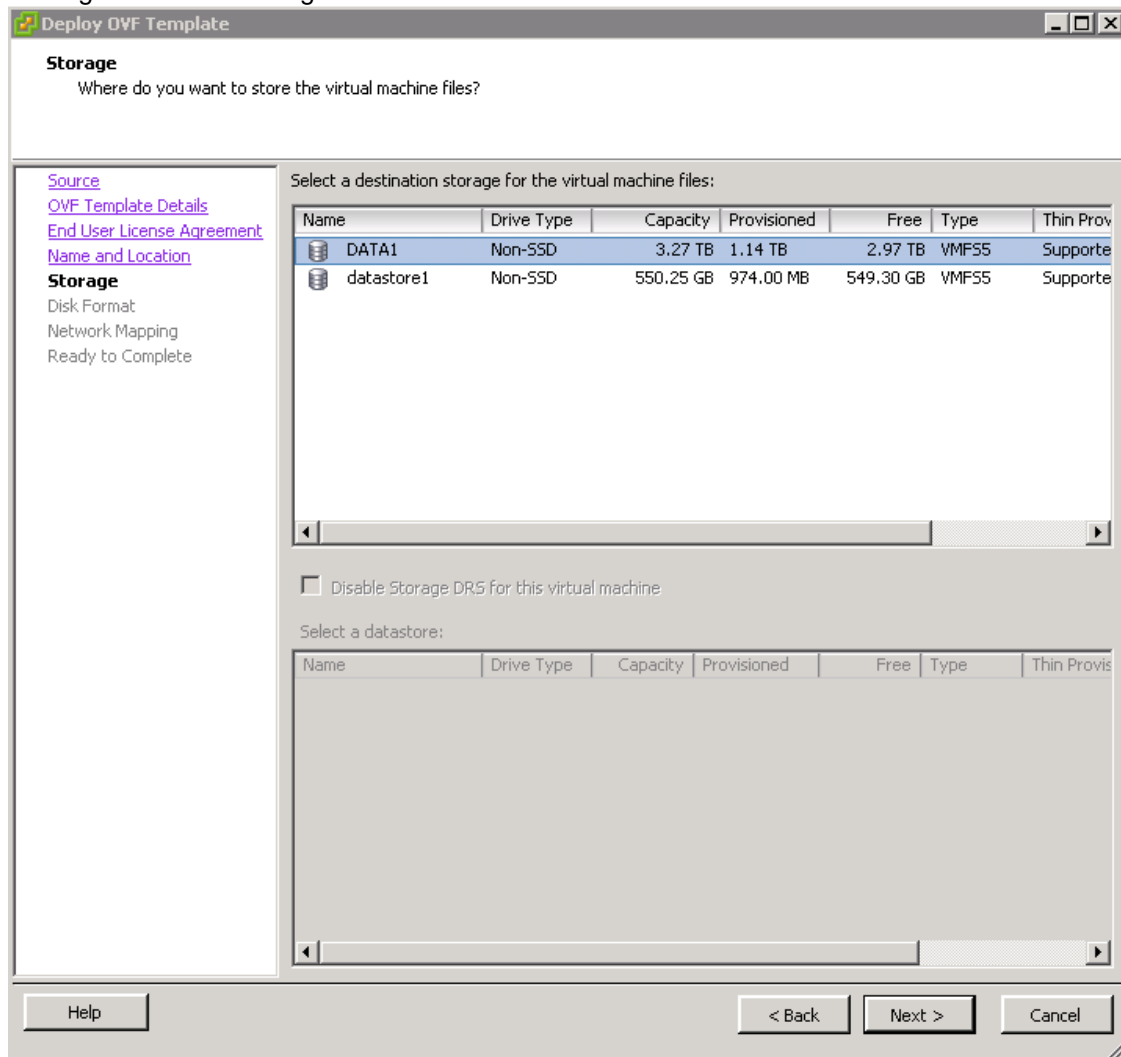
UCS Director Express Deployment and Configuration

These are instructions for installing UCS Director Express with the vSphere Client.

1. Unzip CUCSD_Express_3_0_0_0_GA.zip.
2. In the vSphere Client, select the desired host machine and then select File > Deploy OVF Template...
3. Browse to the location of the folder created in step 1.
4. Select the ovf file and click Open.
5. Click Next to go to OVF Template Details.

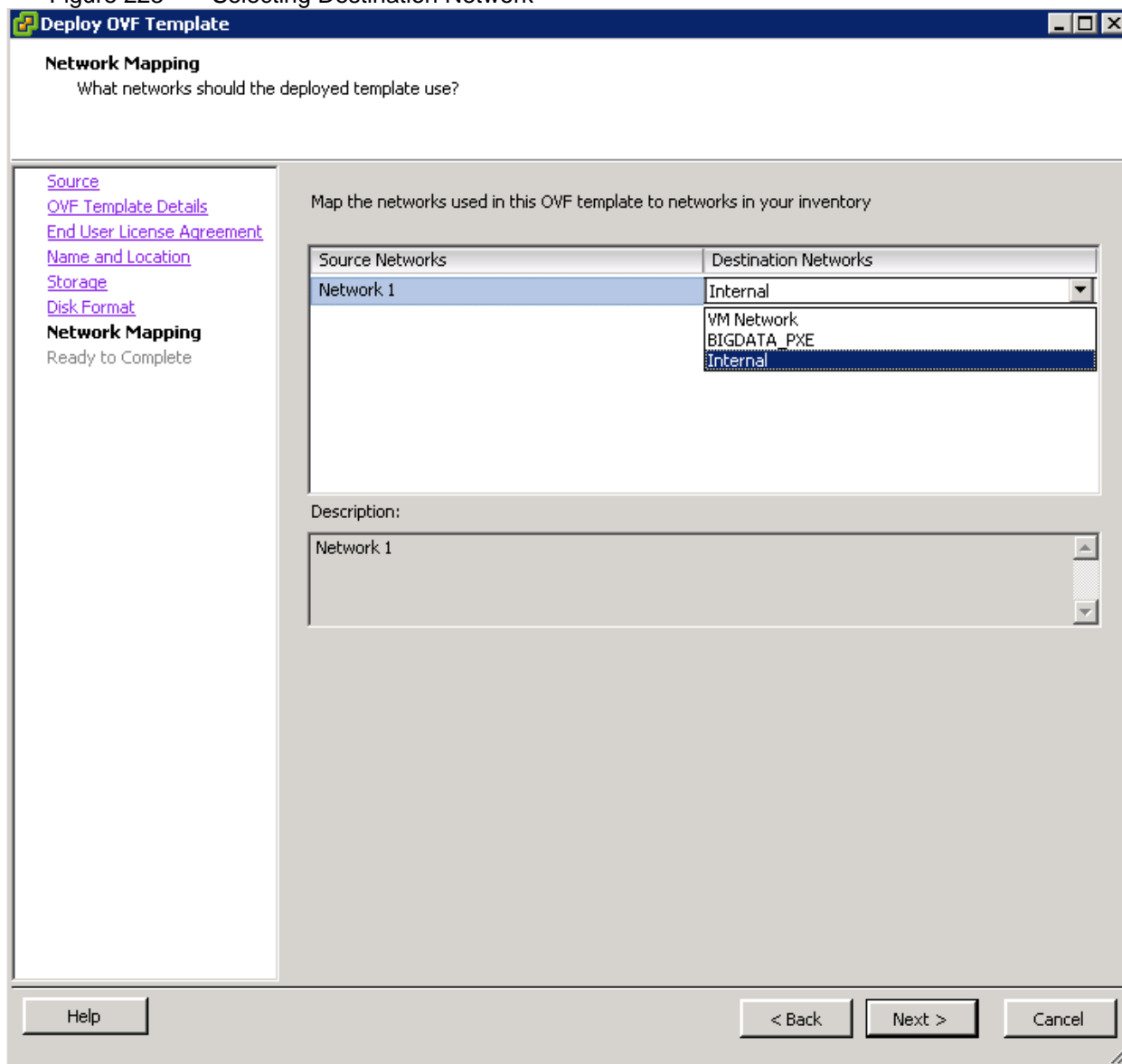
6. Confirm the details and click **Next to go to End User License Agreement**.
7. Read and accept the agreement. Then click **Next to go to Name and Location**.
8. Specify a unique name for the virtual machine. Click **Next to go to Storage**.
9. Select a storage location for the virtual machine files. Click **Next to go to Disk Format**.

Figure 222 Storage for Virtual Machine Files



10. Keep the default option of **Thick Provision Lazy Zeroed**. Click **Next to go to Network Mapping**.
11. Select the management network from the **Destination Networks drop-down menu** and click **Next**.

Figure 223 Selecting Destination Network



12. Confirm the settings, check `Power on after deployment`, and click `Finish`.

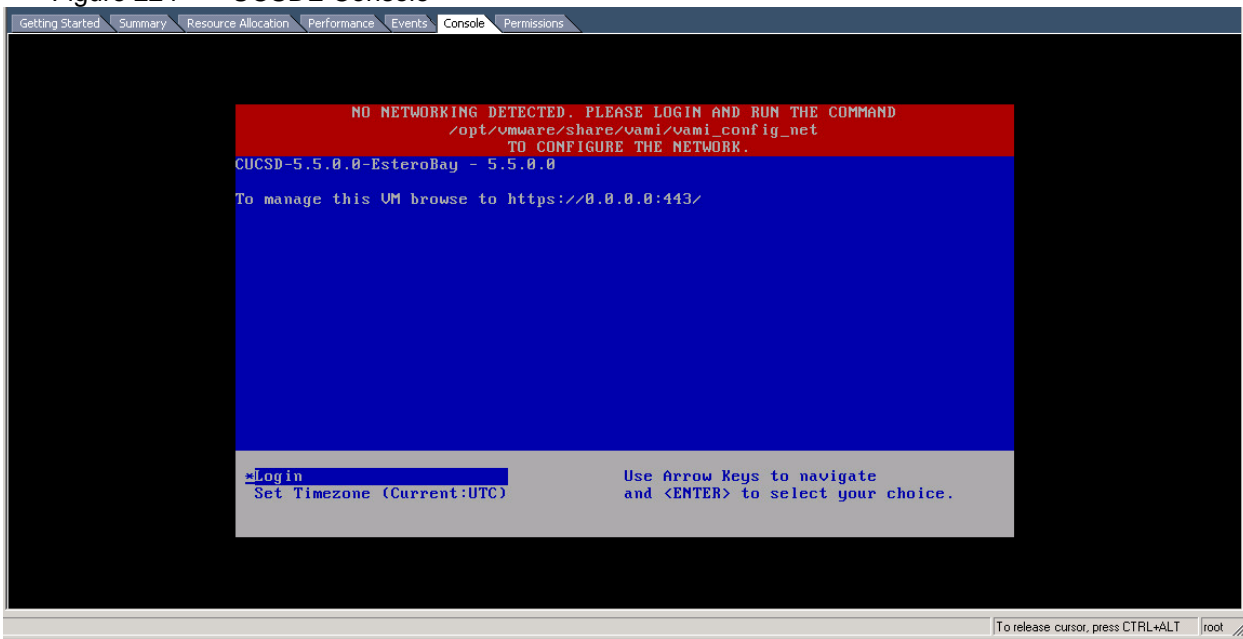
These are instructions for configuring UCS Director Express after installing and powering it on.

13. In the left navigation pane, select the UCSDE VM.

14. Click on the `Console` tab.

15. Read the End User License Agreement and enter yes to agree to the terms. When the installation is complete, you will see this screen:

Figure 224 UCSDE Console



16. Use the arrow keys to navigate to `Login` and press `Enter`.
17. Login as "shelladmin" with the password "changeme"
18. From the shell menu, select `1` to change the default password. Enter the new password and confirm it.
19. From the shell menu, select `14` to configure the network interface.
20. Enter `s` to configure a static IP address.
21. Enter `eth0`.
22. Enter `a` to use IPv4.
23. Enter `y` to confirm that you would like to configure the `eth0` interface with a static IP address.

Figure 225 Configuring Network Interface

```

Performance Events Console Permissions
18) Tail Inframgr Logs
19) Apply Patch
20) Shutdown Appliance
21) Reboot Appliance
22) Manage Root Access
23) Login as Root
24) Configure Multi Node Setup (Advanced Deployment)
25) Clean-up Patch Files
26) Collect logs from a Node
27) Collect Diagnostics
28) Quit

SELECT> 14

After configuring the network interface, you must restart the Cisco UCS Director
services for the updated network configuration to be used.

Do you want to Configure DHCP/STATIC IP [D/S] ? : S
Configuring STA TIC configuration..

Enter the ethernet interface that you want to configure E.g. eth0 or eth1: eth0
Select the IP version you want to configure [a) IPv4, b) IPv6] a/b : a
Do you want to configure IPv4 STA TIC IP for eth0 [y/n]? y
Configuring STA TIC IP for eth0...
IP Address: _

```

24. Enter the desired networking details on the management network for UCSDE. If you will not be using a DNS server, you can leave them blank. You will be prompted to confirm your choice.
25. Press `Return` to return to the menu.
26. From the shell menu, select `10` to ping an IP address. We will check to see if we can ping the gateway IP address.
27. Enter `v4` to indicate that you will be pinging an IPv4 address.
28. Enter the gateway IP address.

Figure 226 Pinging Default Gateway

```

Performance Events Console Permissions

20) Shutdown Appliance
21) Reboot Appliance
22) Manage Root Access
23) Login as Root
24) Configure Multi Node Setup (Advanced Deployment)
25) Clean-up Patch Files
26) Collect logs from a Node
27) Collect Diagnostics
28) Quit

SELECT> 10

Do you want to run ping/ping6 [v4/v6] ? : v4
Enter IP Address : 10.1.1.1
PING 10.1.1.1 (10.1.1.1) 56(84) bytes of data.
64 bytes from 10.1.1.1: icmp_seq=1 ttl=255 time=4.54 ms
64 bytes from 10.1.1.1: icmp_seq=2 ttl=255 time=1.71 ms
64 bytes from 10.1.1.1: icmp_seq=3 ttl=255 time=0.443 ms
64 bytes from 10.1.1.1: icmp_seq=4 ttl=255 time=0.490 ms
64 bytes from 10.1.1.1: icmp_seq=5 ttl=255 time=0.456 ms

--- 10.1.1.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4003ms
rtt min/avg/max/mdev = 0.443/1.528/4.542/1.583 ms
Press return to continue..._

```

29. Press `Return` to return to the menu.
30. Enter 28 to quit the shell menu. You will see a message saying that your VM is now accessible at the IP address you specified.

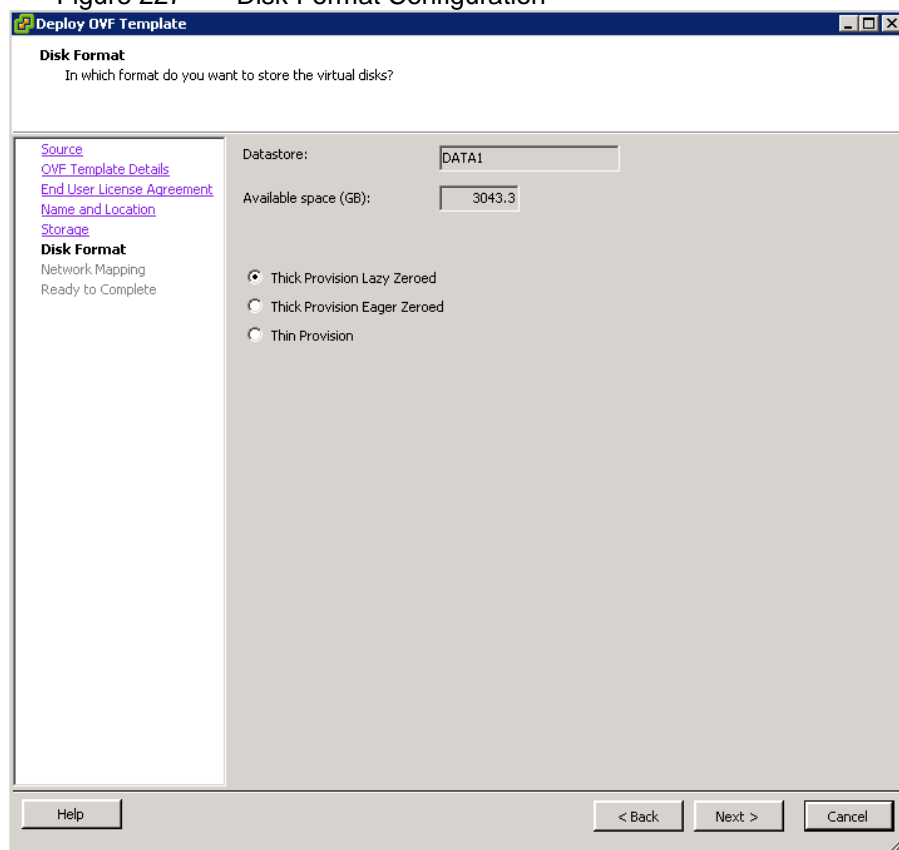
Baremetal Agent Deployment and Configuration

To install the baremetal agent with the vSphere Client, complete the following steps:

1. Unzip `CUCSD_BMA_6_0_0_0_VMWARE_GA.zip`.
2. In the vSphere Client, select the host machine that contains the UCSDE VM.
3. Select `File > Deploy OVF Template...`

4. Browse to the location of the unzipped files in step 1.
5. Select the ovf file and click `Open`. Click `Next to go to OVF Template Details`.
6. Confirm the details and click `Next to go to End User License Agreement`.
7. Read and accept the agreement. Click `Next to go to Name and Location`.
8. Specify a unique name and location for the virtual machine. Click `Next to go to Storage`.
9. Select a storage location for the virtual machine files. Click `Next to go to Disk Format`.
10. Keep the default option of `Thick Provision Lazy Zeroed`. Click `Next to go to Network Mapping`.

Figure 227 Disk Format Configuration



11. `Destination Networks` is a drop-down menu. For `Network 1`, select the management network. For `Network 2`, select the PXE network. Click `Next`.
12. Confirm the settings, check `Power on after deployment`, and click `Finish`.
13. Click on the `Console` tab.
14. After loading, there will be an `End User License Agreement` to read and accept.
15. Use the arrow keys to navigate to `Login` and press `Enter`.
16. Login as the user "root" with the default password "pxeboot".

17. Type this command to configure the BMA network interface:

```
/opt/vmware/share/vami/vami_config_net
```

Figure 228 Network Configuration for BMA

```

Performance Events Console Permissions
localhost.localdom login: root
Password:
[root@localhost ~]# /opt/vmware/share/vami/vami_config_net

Main Menu

0) Show Current Configuration (scroll with Shift-PgUp/PgDown)
1) Exit this program
2) Default Gateway
3) Hostname
4) DNS
5) Proxy Server
6) IP Address Allocation for eth0
7) IP Address Allocation for eth1
Enter a menu number [0]: 6_

```

18. In the main menu, enter 6 to configure the IP address for eth0.
19. Enter `y` to confirm you want to configure an IPv4 address for eth0.
20. Enter `n` to use a static IP address instead of DHCP.
21. Enter the IP address and netmask when prompted.
22. Enter `y` to confirm. You will be returned to the main menu after the network parameters are successfully changed.
23. Enter 7 to configure the IP address for eth1
24. Enter `y` to confirm.
25. Enter `n` to use a static IP address.
26. Enter the IP address and netmask when prompted.
27. Enter `y` to confirm.

Figure 229 Configuring IP address for eth1

```

Configure an IPv4 address for eth1? y/n [n]: y
Use a DHCPv4 Server instead of a static IPv4 address? y/n [n]: n
IPv4 Address [ ]: 192.168.12.42
Netmask [ ]: 255.255.255.0
IPv4 Address: 192.168.12.42
Netmask: 255.255.255.0

Is this correct? y/n [y]: y

Reconfiguring eth1...
Determining if ip address 192.168.12.42 is already in use for device eth1...
vami_login: no process killed
Network parameters successfully changed to requested values

Main Menu
0) Show Current Configuration (scroll with Shift-PgUp/PgDown)
1) Exit this program
2) Default Gateway
3) Hostname
4) DNS
5) Proxy Server
6) IP Address Allocation for eth0
7) IP Address Allocation for eth1
Enter a menu number [0]: _

```

License Administration

To add licenses to UCSDE, complete the following steps:

1. In a web browser, go to the UCSDE IP address.
2. Login as `admin` with a password of `admin`.

Figure 230 Login Screen for UCSD Express for Big Data



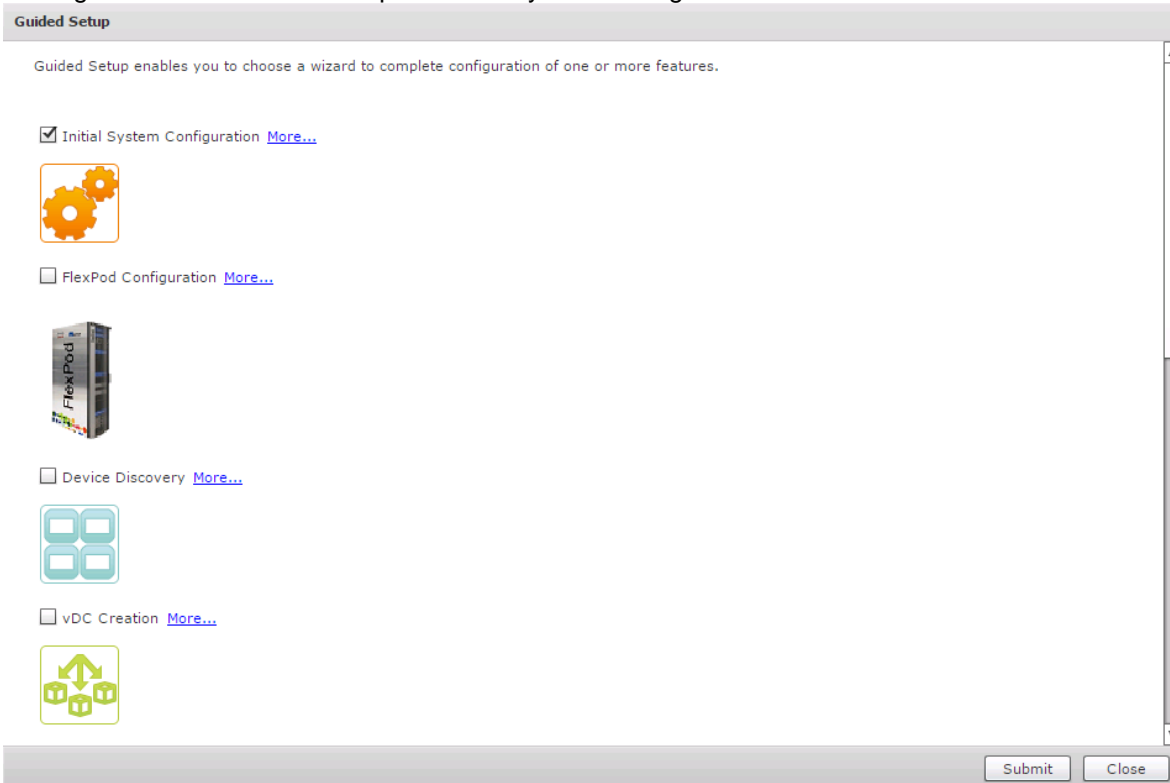
3. Change the password. (Figure 231)

Figure 231 Changing the Default Password

The screenshot shows a web interface for changing a password. At the top is a header bar labeled "User Information". Below it is a tabbed interface with the "Password" tab selected. Under the "Change Password" heading, there are three input fields: "Old Password" containing "*****", "New Password" containing "*****", and "Confirm New Password" containing "*****". Each field has a red asterisk icon to its right. A "Save" button is positioned at the bottom right of the form area.

4. Use the Guided Setup for Initial System Configuration to upload the license, select a locale, configure an SMTP mail server, configure your email address, and/or set up NTP and DNS servers. (Figure 232)

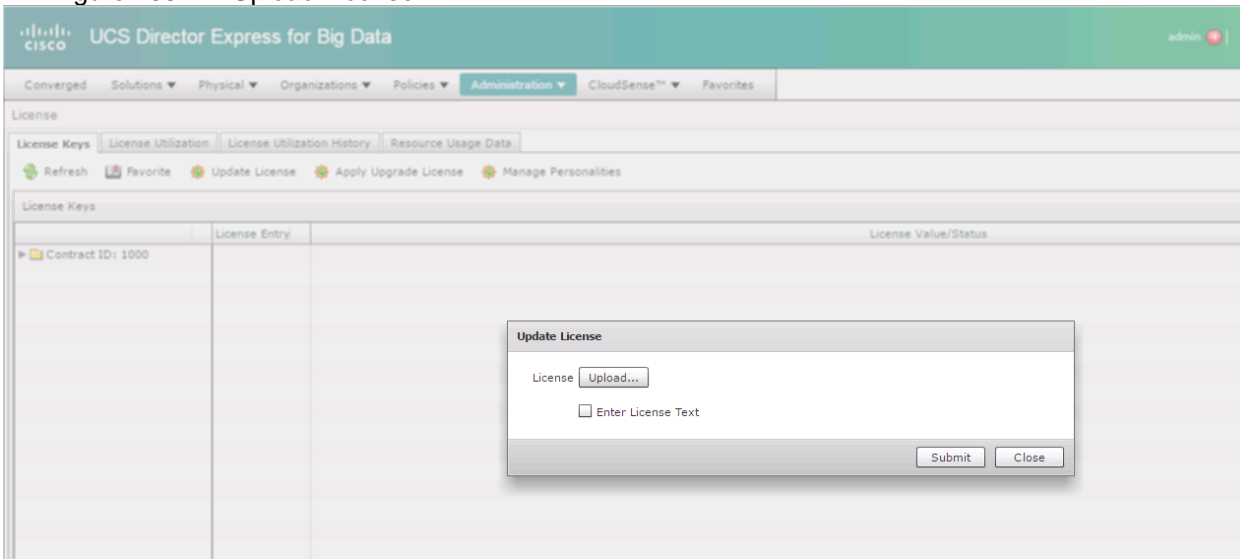
Figure 232 Guided Setup for Initial System Configuration



Alternatively, you can upload and configure the license with the following steps.

5. Navigate to Administration > License.
6. Click Upload and browse to the license file.

Figure 233 Upload License



7. Click Submit.

After uploading the license, enable the Big Data personality to unlock certain menu options.

8. Navigate to Administration > License.
9. Select the license and click on Manage Personalities.
10. In the Personality Configuration window, select Big Data only.
11. Click Submit.
12. Restart UCSD Services.
 - a. Login to the UCSDE console as 'shelladmin'
 - b. In the Shell Menu, select 3 to stop services.
 - c. Select 2 to confirm that the services have stopped.
 - d. Select 4 to start services.
 - e. Select 2 to confirm that the services have restarted successfully.
13. Re-login to UCSDE.

License Administration

To add licenses to UCSDE, complete the following steps:

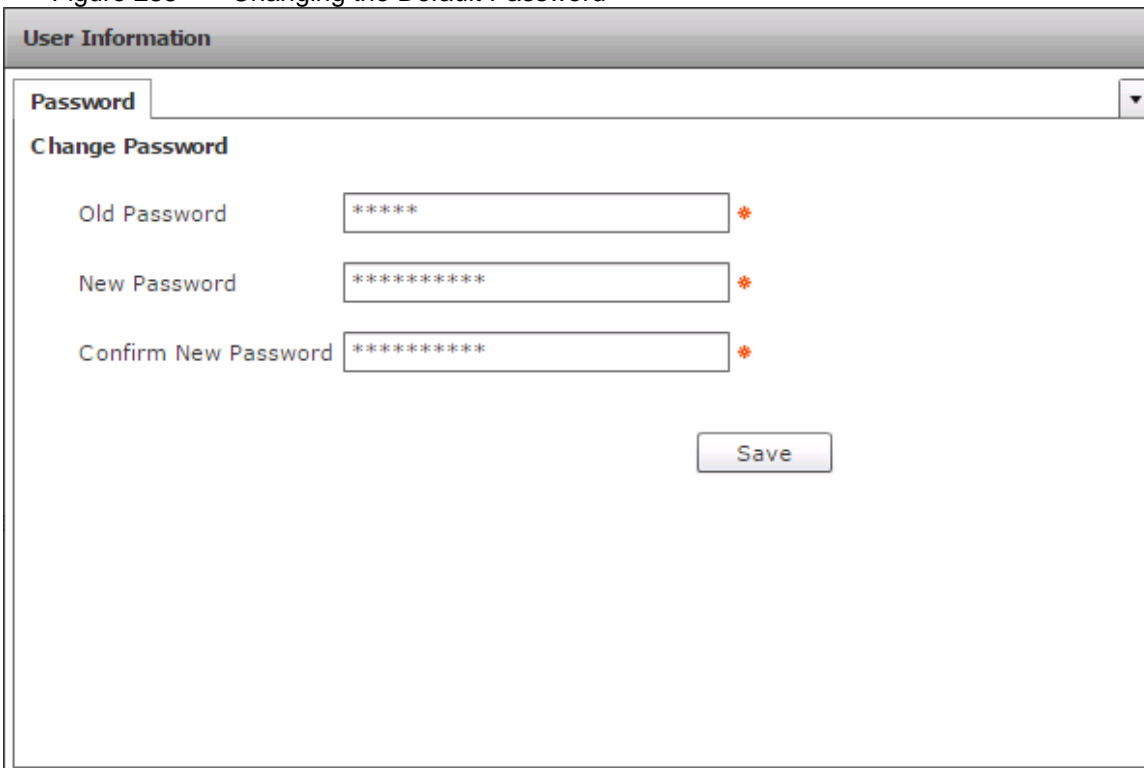
1. In a web browser, go to the UCSDE IP address.
2. Login as `admin` with a password of `admin`. (Figure 234)

Figure 234 Login Screen for UCSD Express for Big Data



3. Change the password. (Figure 235)

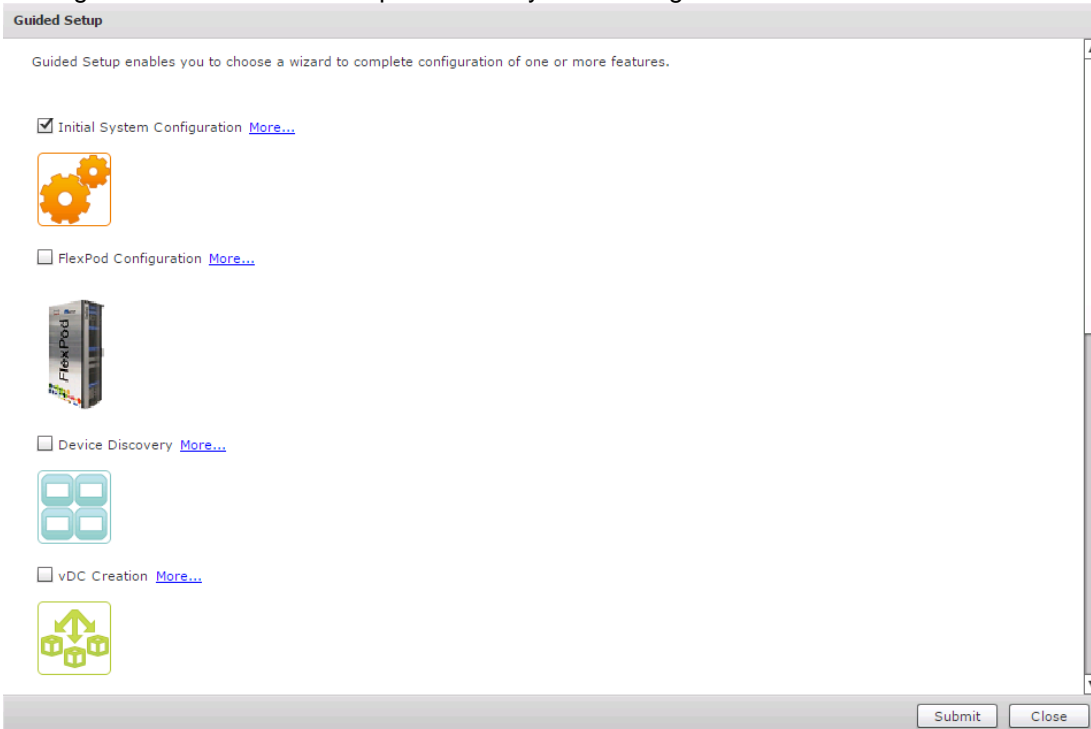
Figure 235 Changing the Default Password



The screenshot shows a web interface for changing a password. It is titled "User Information" and has a "Password" tab selected. Under the "Change Password" section, there are three input fields: "Old Password" containing "*****", "New Password" containing "*****", and "Confirm New Password" containing "*****". Each field has a red asterisk icon to its right. A "Save" button is located at the bottom right of the form.

4. Use the Guided Setup for Initial System Configuration to upload the license, select a locale, configure an SMTP mail server, configure your email address, and/or set up NTP and DNS servers, Figure 236

Figure 236 Guided Setup for Initial System Configuration



The screenshot shows the "Guided Setup" dialog box. It contains the text: "Guided Setup enables you to choose a wizard to complete configuration of one or more features." Below this, there are four options, each with a checkbox and a "More..." link:

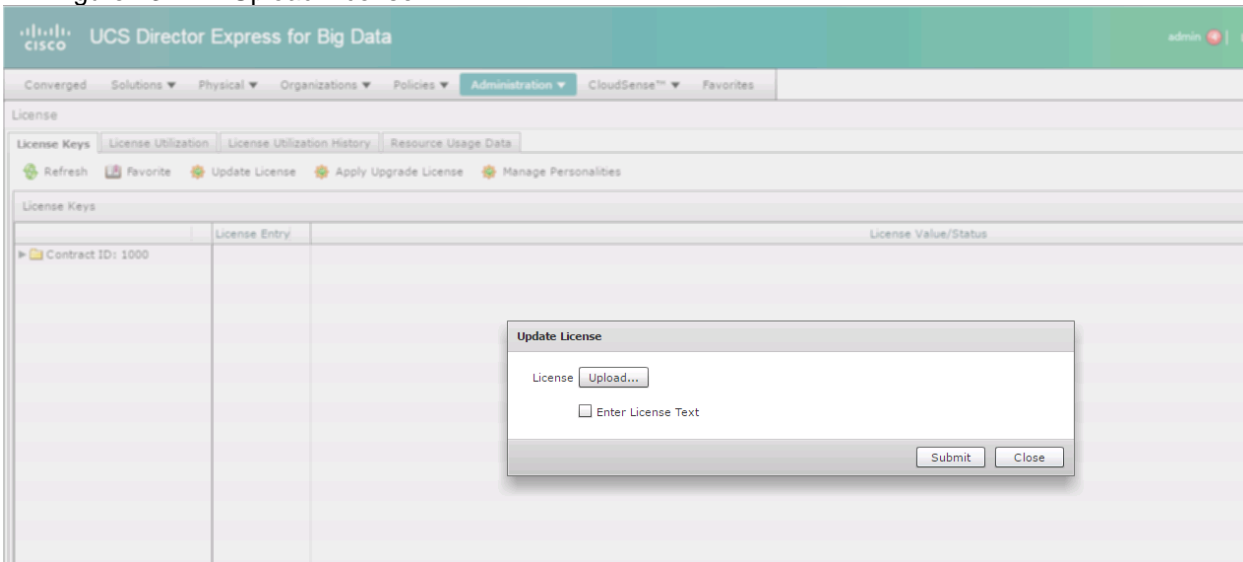
- Initial System Configuration [More...](#)
- FlexPod Configuration [More...](#)
- Device Discovery [More...](#)
- vDC Creation [More...](#)

At the bottom right, there are "Submit" and "Close" buttons.

Alternatively, upload and configure the license with the following steps.

5. Navigate to Administration > License.
6. Click Upload... and browse to the license file.

Figure 237 Upload License



7. Click Submit.

After uploading the license, enable the Big Data personality to unlock certain menu options.

8. Navigate to Administration > License.
9. Select the license and click on Manage Personalities.
10. In the Personality Configuration window, select Big Data only.
11. Click Submit.
12. Restart UCSDE services.
 - a. Login to the UCSDE console as 'shelladmin'
 - b. In the Shell Menu, select 3 to stop services.
 - c. Select 2 to confirm that the services have stopped.
 - d. Select 4 to start services.
 - e. Select 2 to confirm that the services have restarted successfully.
13. Re-login to UCSDE.

License Administration

To add licenses to UCSDE, complete the following steps:

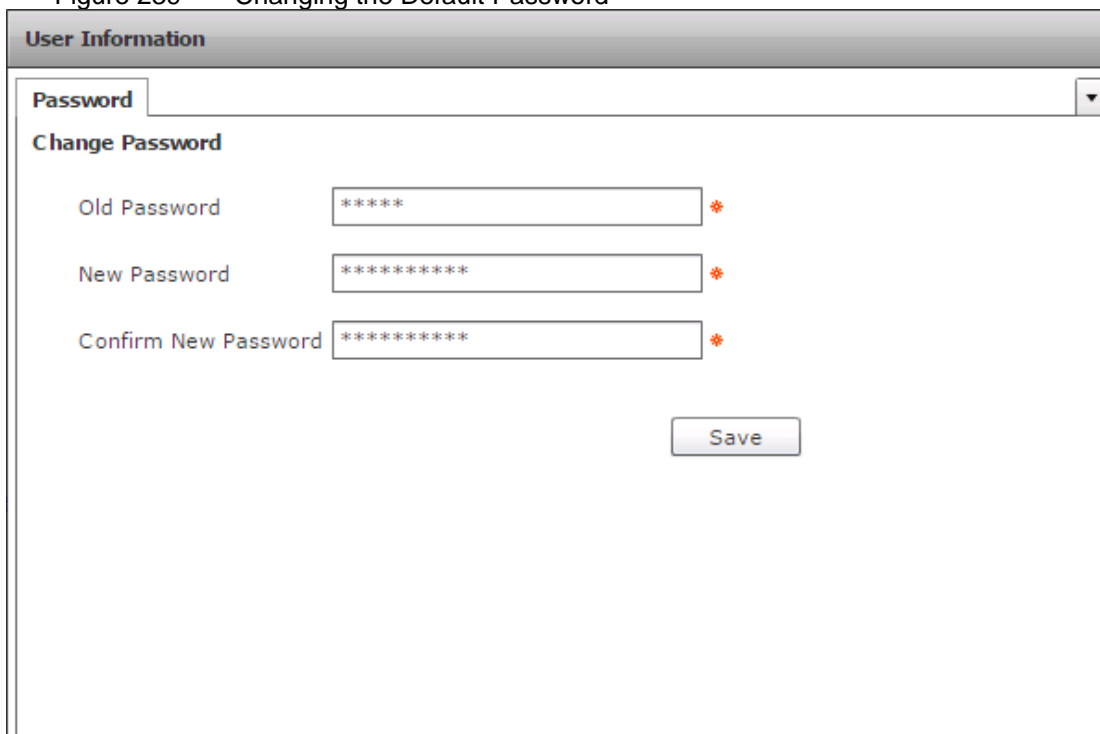
1. In a web browser, go to the UCSDE IP address.
2. Login as `admin` with a password of `admin`, Figure 238

Figure 238 Login Screen for UCSD Express for Big Data



3. Change the password

Figure 239 Changing the Default Password



User Information

Password

Change Password

Old Password *

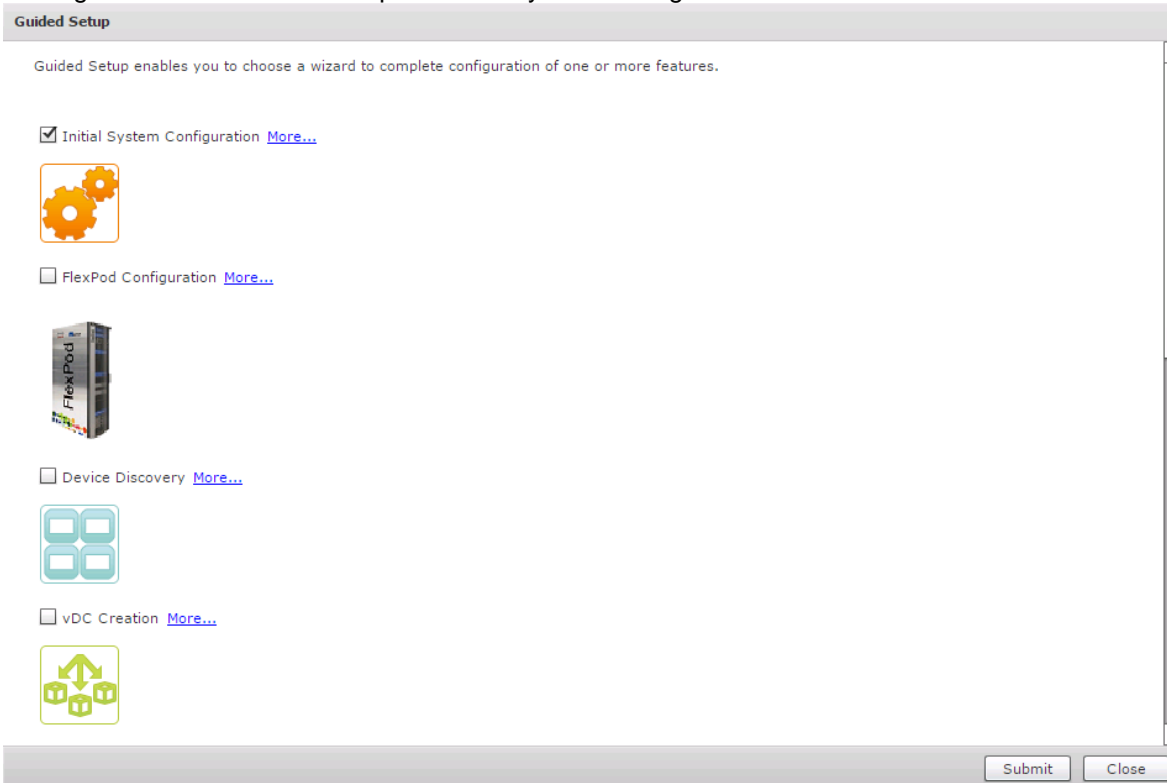
New Password *

Confirm New Password *

Save

4. Use the Guided Setup for Initial System Configuration to upload the license, select a locale, configure an SMTP mail server, configure your email address, and/or set up NTP and DNS servers.

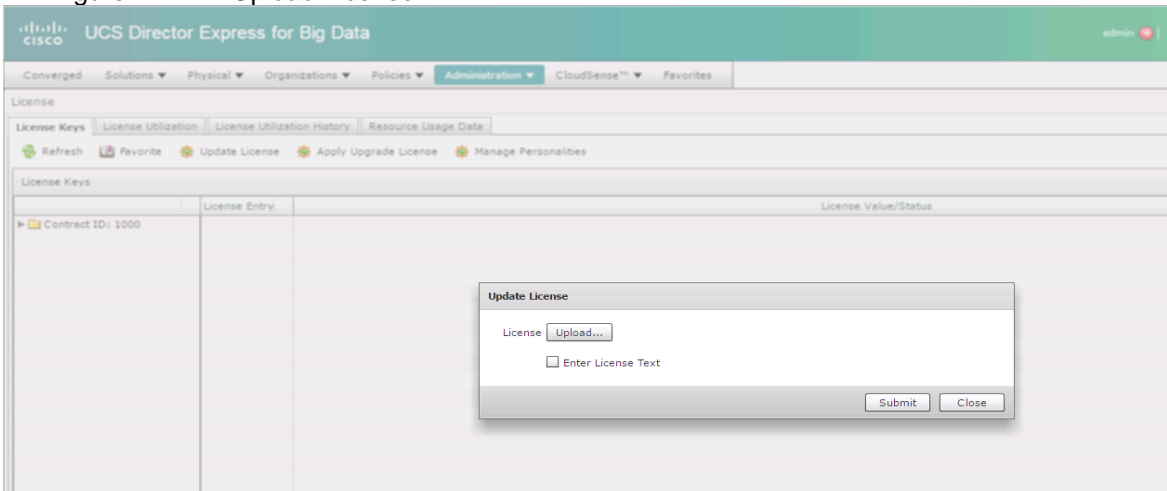
Figure 240 Guided Setup for Initial System Configuration



Alternatively, upload and configure the license with the following steps.

5. Navigate to Administration > License.
6. Click Upload and browse to the license file, Figure 241

Figure 241 Upload License



7. Click Submit.

After uploading the license, enable the Big Data personality to unlock certain menu options.

8. Navigate to Administration > License.

9. Select the license and click on Manage Personalities.
10. In the Personality Configuration window, select Big Data only.
11. Click Submit.
12. Restart UCSD services.
 - a. Login to the UCSDE console as 'shelladmin'
 - b. In the Shell Menu, select 3 to stop services.
 - c. Select 2 to confirm that the services have stopped.
 - d. Select 4 to start services.
 - e. Select 2 to confirm that the services have restarted successfully.
13. Re-login to UCSDE.

Building the Software Catalog

Cisco UCSDE uses a catalog of installed software when it deploys a cluster. For this Splunk cluster, Redhat Enterprise Linux 6.8 and Splunk are installed, other Hadoop distributions and versions can also be used. Have the RHEL ISO in the BMA VM.

1. Go to Solutions > Big Data > Settings.
2. Select the Software Catalogs tab.
3. Click Add.
4. In the Create Software Catalogs dialog box, select the target BMA.
5. Check the Restart BMA Services checkbox.
6. In the Linux OS Upload section, enter RHEL6.8 for the Catalog Name (the name can not have any spaces).
7. From the Upload Type drop-down menu, select Path to ISO in BMA.
8. Specify the location of the ISO file.
9. Click Submit.

Figure 242 Uploading to Software Catalogs

Create Software Catalogs

Upload all your software files to UCSD

[Supported file name formats:
 Linux OS -> rhel-x.x.iso
 Hadoop software: MapR-x.y.z.zip/gz/tgz/tar
 Common software: bd-sw-rep.zip/gz/tgz/tar
 JDK software: jdk-7u76-linux-x64.rpm
 splunk software: splunk-x.y.z.zip/gz/tgz/tar]

*

Upload...

Target BMA:

Restart BMA Services

Linux OS Upload:

Catalog Name:

Upload Type:

Location:

Big Data Software Upload:

Catalog Name:

Upload Type:

Common Software Upload:

Upload Type:

After some time, click Refresh and see RHEL6.8 in the software catalog. Figure 243 below, shows the software catalog sorted by the Missing Packages column so that RHEL6.8 will be the first entry.

Figure 243 RHEL 6.8 in the Software Catalogs

UCS Director Express for Big Data							admin
Converged Solutions Physical Organizations Policies Administration CloudSense™ Favorites							
Settings							
Big Data IP Pools External Database Hadoop Config Parameters QoS System Class Management Software Catalogs Configuration Check Rules							
Refresh Favorite Add Modify							
Software Catalogs							
BMA Server IP	Catalog Name	Distribution	Last Updated	Required Packages	Available Packages	Missing Packages	
10.1.1.42[DEFAULT]	RHEL6.8		08/19/2016 07:24:2				
10.1.1.42[DEFAULT]	Hortonworks-2.1	Hortonworks	08/19/2016 07:24:2	HDP-UTILS-1.1.0.17-centos6.tar.gz ambari-1.6.1-centos6.tar.gz openssl-1.0.1e-30.el6.x86_64.rpm HDP-2.1.5.0-centos6-rpm.tar.gz	catalog.properties userrpmlist.txt	HDP-UTILS-1.1.0.17-centos6.tar.gz ambari-1.6.1-centos6.tar.gz openssl-1.0.1e-30.el6.x86_64.rpm HDP-2.1.5.0-centos6-rpm.tar.gz ,pssh-2.3.1.tar.gz,clustershell-1.7-1.el6.noarch.rpm,libyaml-0.1.3-4.el6_6.x86_64.rpm	
10.1.1.42[DEFAULT]	Hortonworks-2.2	Hortonworks	08/19/2016 07:24:2	HDP-UTILS-1.1.0.20-centos6.tar.gz ambari-1.7.0-centos6.tar.gz openssl-1.0.1e-30.el6.x86_64.rpm HDP-2.2.0.0-centos6-rpm.tar.gz	catalog.properties userrpmlist.txt	HDP-UTILS-1.1.0.20-centos6.tar.gz ambari-1.7.0-centos6.tar.gz openssl-1.0.1e-30.el6.x86_64.rpm HDP-2.2.0.0-centos6-rpm.tar.gz ,pssh-2.3.1.tar.gz,clustershell-1.7-1.el6.noarch.rpm,libyaml-0.1.3-4.el6_6.x86_64.rpm	
10.1.1.42[DEFAULT]	Hortonworks-2.3	Hortonworks	08/19/2016 07:24:2	HDP-UTILS-1.1.0.20-centos6.tar.gz ambari-2.1.1-centos6.tar.gz openssl-1.0.1e-30.el6.x86_64.rpm HDP-2.3.0.0-centos6-rpm.tar.gz	catalog.properties userrpmlist.txt	HDP-UTILS-1.1.0.20-centos6.tar.gz ambari-2.1.1-centos6.tar.gz openssl-1.0.1e-30.el6.x86_64.rpm HDP-2.3.0.0-centos6-rpm.tar.gz ,pssh-2.3.1.tar.gz,clustershell-1.7-1.el6.noarch.rpm,libyaml-0.1.3-4.el6_6.x86_64.rpm	
10.1.1.42[DEFAULT]	cloudera-5.0.1	cloudera	08/19/2016 07:24:2	cm5.0.1-centos6.tar.gz CDH-5.0.1-1.cdhs.0.1.p0.47-el6.parcel.sha1 CDH-5.0.1-1.cdhs.0.1.p0.47-el6.parcel.sha1 manifest.json mysql-connector-java-5.0.8-bin.jar	catalog.properties userrpmlist.txt	cm5.0.1-centos6.tar.gz CDH-5.0.1-1.cdhs.0.1.p0.47-el6.parcel CDH-5.0.1-1.cdhs.0.1.p0.47-el6.parcel.sha1 manifest.json mysql-connector-java-5.1.26.tar.gz ,pssh-2.3.1.tar.gz,clustershell-1.7-1.el6.noarch.rpm,libyaml-0.1.3-4.el6_6.x86_64.rpm	

Put software files in specific folders for UCSDE to find.

- In the Software Catalogs tab, see the files for a particular Hadoop distribution in the Required Packages column.
- The Available Packages column shows which files UCSDE already has.

10. The Missing Packages column shows which files it still needs.

- The Required Packages list is maintained in the HadoopDistributionRPM.txt file located in the BMA's /opt/cnsaroot/bigdata_templates/common_templates directory. It may be necessary sometimes to change these requirements (for example, if you are using a newer version of a required utility file). Please note that the file names are case-sensitive.

```
[root@localhost ~]# cd /opt/cnsaroot/bigdata_templates/common_templates/
[root@localhost common_templates]# vi HadoopDistributionRPM.txt
```

For Splunk 6.4.0, there are several missing files. Some files, such as clustershell, are used by many distributions and can be placed in the /opt/cnsaroot/bd-sw-rep directory in the BMA VM. Files that are specific to a particular distribution should be placed in the appropriate subfolder in the /opt/cnsaroot/bd-sw-rep directory.

1. Copy the following files to /opt/cnsaroot/bd-sw-rep:

```
clustershell-1.7-1.el6.noarch.rpm
```

```
libyaml-0.1.3-4.el6_6.x86_64.rpm
```

```
PyYAML-3.10-3.1.el6.x86_64.rpm
```

2. Copy these Splunk files to /opt/cnsaroot/bd-sw-rep/splunk-6.4.0:

```
splunk-6.4.0-f2c836328108-linux-2.6-x86_64.rpm
```

```
splunk.license
```

3. In the Software Catalogs tab in UCSDE, click Refresh and confirm that there are no longer any files in the Missing Packages column of the Splunk distribution.

To use a version of Splunk that does not already have a software catalog, complete the following steps, (Splunk 6.4.3 is used in this example, but this can be adapted for any version):

4. Copy the following files to /opt/cnsaroot/bd-sw-rep:

```
clustershell-1.7-1.el6.noarch.rpm
```

```
libyaml-0.1.3-4.el6_6.x86_64.rpm
```

```
PyYAML-3.10-3.1.el6.x86_64.rpm
```

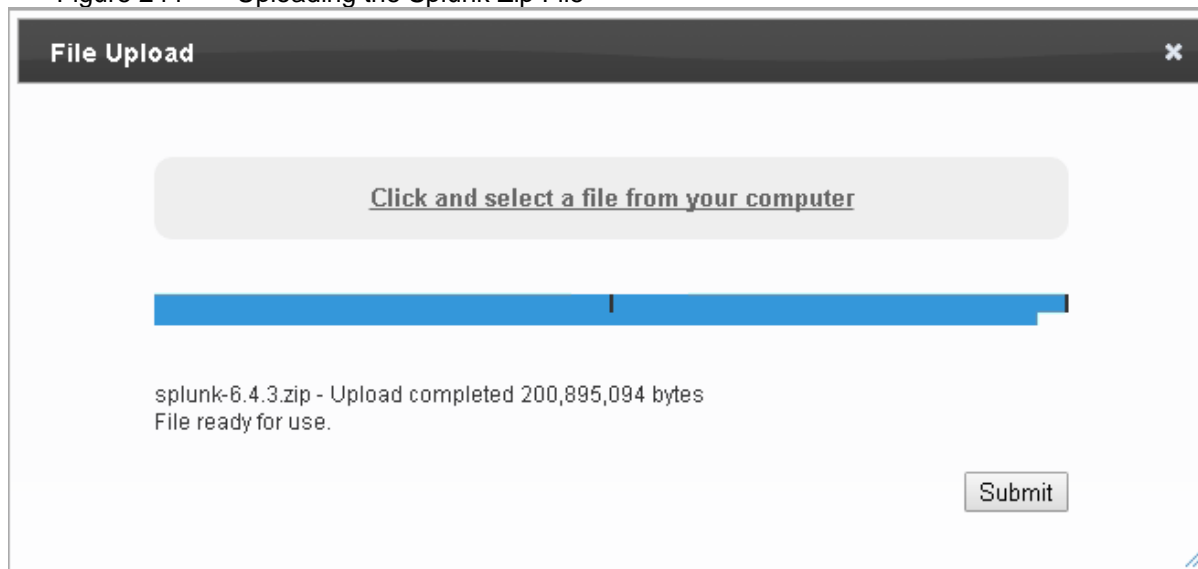
5. Create a folder called splunk-6.4.3 containing the files splunk-6.4.3-b03109c2bad4-linux-2.6-x86_64.rpm and splunk.license.

6. Zip that folder in a file called splunk-6.4.3.zip.

7. In UCSDE, go to Solutions > Big Data > Settings.

8. Click the Software Catalogs tab.
9. Click Add.
10. In the Create Software Catalogs window, click the Upload... button.
11. In the File Upload window, click the link to upload the zip file you created in Step 2.
12. When the upload completes, click Submit. Figure 244

Figure 244 Uploading the Splunk Zip File



13. In the Big Data Software Upload section, enter splunk-6.4.3 for the Catalog Name.
14. Keep the Upload Type set to Desktop File and note that your zip file shows up as the uploaded file.

Figure 245 Create Software Catalogs Window

Upload all your software files to UCSD

[Supported file name formats:
 Linux OS -> rhel-x.x.iso
 Hadoop software: MapR-x.y.z.zip/gz/tgz/tar
 Common software: bd-sw-rep.zip/gz/tgz/tar
 JDK software: jdk-7u76-linux-x64.rpm
 splunk software: splunk-x.y.z.zip/gz/tgz/tar]

Upload... Uploaded File: splunk-6.4.3.zip *

Target BMA BMA91(50.1.1.91)-default ▼

Restart BMA Services

Linux OS Upload:

Catalog Name

Upload Type Desktop File ▼

Big Data Software Upload:

Catalog Name

Upload Type Desktop File ▼

Uploaded file:splunk-6.4.3.zip

Common Software Upload:

Upload Type Desktop File ▼

JDK Upload:

Submit Close

15. Click Submit.
16. Click OK.
17. After some time, the splunk-6.4.3 software catalog will be created. There will be a splunk-6.4.3 subdirectory in /opt/cnsaroot/bd-sw-rep and splunk-6.4.3 will show up in the Software Catalogs list in UCSDE. Confirm that there are no missing packages.

Creating IP Pools

Create IP pools for the Splunk cluster. While provisioning the cluster, these pools will be associated with the NICs. UCSD Express workflow will assign IP addresses to the NICs from their respective IP pools. There will be one pool for management and two pools for data.

1. In UCS Director, navigate to Solutions > Big Data > Settings.
2. Click on the Big Data IP Pools tab.
3. Click the Add button.

4. Enter the IP Pool Name. We will be creating three pools (one for each vNIC interface): MGMT, DATA1, and DATA2.
5. (Optional) Enter a description for the IP pool.
6. Keep the Assignment Order as Default.

Figure 246 Creating and Naming an IP Pool

The screenshot shows a web-based configuration interface for creating an IP pool. The window title is "Create an IP Pool". On the left, there is a sidebar with two items: "IP Pool" (highlighted) and "IPv4 Addresses". The main content area is titled "IP Pool Management Specification" and contains the following fields:

- IP Pool Name:** A text input field containing "MGMT". A small red asterisk is visible to the right of the field.
- Description:** An empty text input field.
- Assignment Order:** A dropdown menu currently set to "Default".

At the bottom right of the window, there are two buttons: "Next" and "Close".

7. Click Next.
8. Click the Add button (designated with the + symbol) to add an entry to the list of IPv4 blocks.
9. Enter the IP address range, subnet mask, and default gateway.
10. Click Submit.

Figure 247 Configuring IP Pools

Add Entry to IPv4 Blocks

Static IP Pool *

Static IP Pool. Example (IPV4): 192.168.0.1 - 192.168.0.50,192.168.0.100,192.168.1.20-192.168.1.70

Subnet Mask *

Subnet Mask, ex (IPV4): 255.255.255.0

Default Gateway

Primary DNS

Secondary DNS

11. Verify the details in the table and click Submit to create the IP pool.
12. Repeat the above steps to create two more pools.

Creating an Instant Splunk Cluster

When creating a Splunk cluster, UCSDE will automatically associate the servers with a service profile, install Red Hat Enterprise Linux and Splunk, and assign roles to the servers.

To create a Splunk cluster, complete the following steps:

1. Go to Solutions > Big Data > Containers.
2. Select the Cluster Deploy Templates tab.
3. Click on Instant Splunk Cluster Creation.
4. In the Instant Splunk Cluster Creation dialog box, complete the fields as shown in Table 13 :

Table 13 Splunk Cluster Fields

Field Name	Description
Big Data Account Name	The name of the Big Data account
UCSM Policy Name Prefix	The UCSM Policy Name prefix is used in naming the service profile template
SSH (root) Password Confirm SSH Password	The SSH root password. The SSH username pertains to the root user.
Splunk Manager Password Confirm Splunk Manager Password	The management console password
OS Version	Choose the operating system to be installed on the servers in this cluster.
Splunk Distribution Version	Choose the Splunk Enterprise version to be used for this cluster.
Multi-UCSM	Check the Multi-UCSM check box if you use multiple UCSM accounts. In this design, we are leaving this unchecked.
UCS Manager Account	Choose the Cisco UCS Manager account for this cluster.

Field Name	Description
Organization	Choose the organization in which the servers for this cluster are located. If you did not create an organization, this will be <code>root</code> .
UCS SP Template	This is not a mandatory field. Leave this blank; UCSDE will create a service profile template for you.
PXE VLAN ID	Enter the PXE VLAN ID.

Figure 248 Instant Splunk Cluster Creation Window

The screenshot shows the 'Instant Splunk Cluster Creation' window with the following fields and values:

- Big Data Account Name: *
- UCSM Policy Name Prefix: *
- SSH (root) Password: *
- Confirm SSH Password: *
- Splunk Manager Password: *
- Confirm Splunk Manager Password: *
- OS Version: *
- Splunk Distribution Version: *
- Multi UCSM
- UCS Manager Account: *
- Organization: *
- UCS SP Template:

5. Enter 2 for the Replication Factor.
6. Enter 2 for the Search Factor.
7. In the Splunk Server Roles section, edit each role (click the role and then click the pencil icon) to specify the number of nodes for that role, the host name prefix, and the server pool. Recall that we created three separate server pools for these roles. There will be 8 indexers, 3 search heads, and 2 admin nodes, as shown in Figure 249



If using Cisco C240 M4 servers for the indexers, check the box for “SSD Boot Drives Available for OS” when editing the server roles for indexers.

Figure 249 Indexer Values

Edit Entry

Node Type: Indexer

Node Count: *

Host Name Prefix: *

SSD Boot Drives Available for OS

Server Pool

Search in Column: Text: [X] [X]

	ID	Server Pool	Server Pool f	Assigned	Size
<input checked="" type="checkbox"/>	UCSM10;org-ro	Indexer		0	8
<input type="checkbox"/>	UCSM10;org-ro	Search		0	3
<input type="checkbox"/>	UCSM10;org-ro	Admin		0	2

Total 9 items, shown 3 (filtered)

Submit Close

Figure 250

Instant Splunk Cluster Creation

Total 2 items

PXE VLAN ID: 102 [4048-4093],[1-3967] *

Replication Factor: 2 *

Search Factor: 2 *

Splunk Server Roles

Node Type	Node Count	Host Name	SSD Boot I	Search Hei	Server Pod
Indexer	8	idx	true	true	UCSM10;org-
Search Head	3	sh	true	true	UCSM10;org-
Admin	2	admin	true	true	UCSM10;org-

Total 3 items

8. In the vNIC Template section, edit each vNIC to configure it with the appropriate IP pool. If no service profile template was specified in step 4, specify the MAC address pool and VLAN for each vNIC.

Figure 251 vNIC Template

Edit vNIC Template Entry

vNIC Name: eth0 *

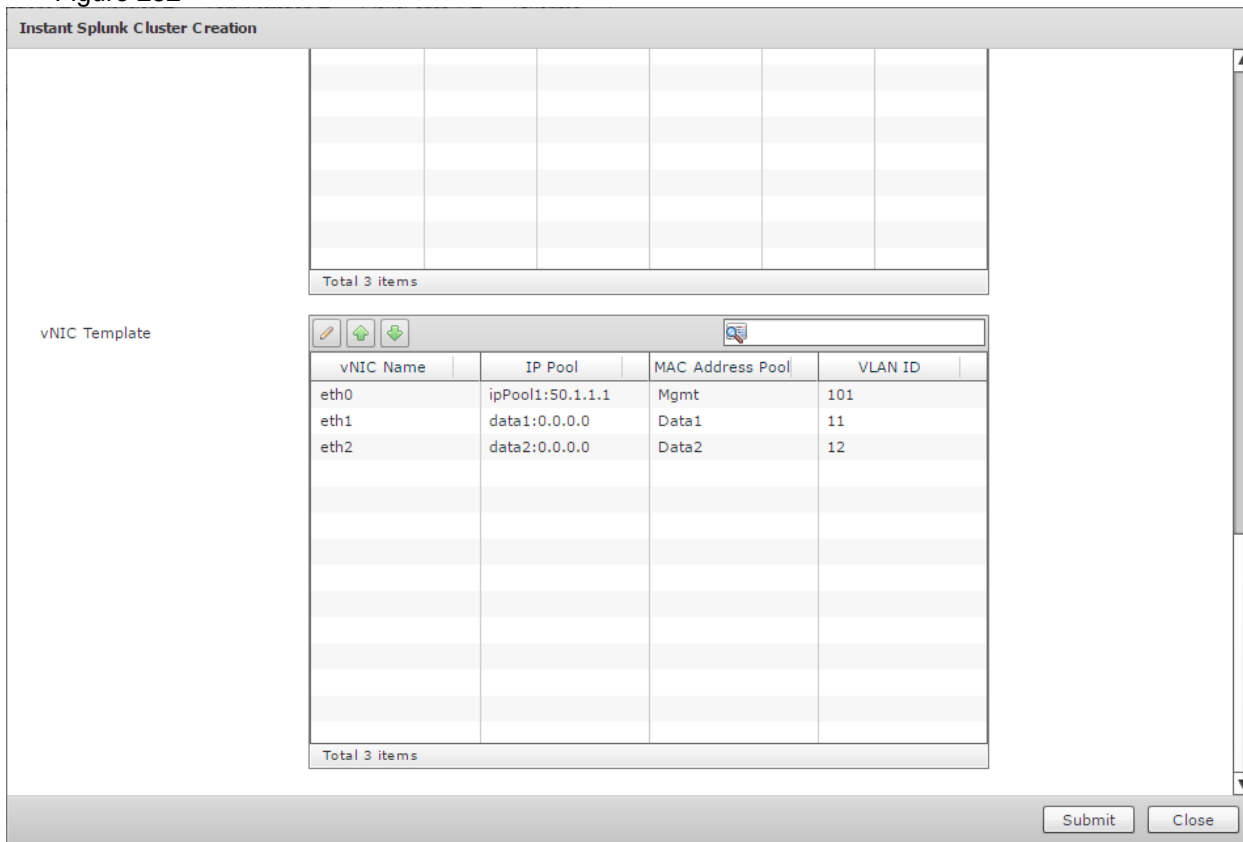
IP Pool: ipPool1(50.1.1.101 - 50.1.1.200) *

MAC Address Pool: Mgmt (512) *

VLAN ID: 101 [4048-4093],[1-3967] (MGMT VLAN) *

Submit Close

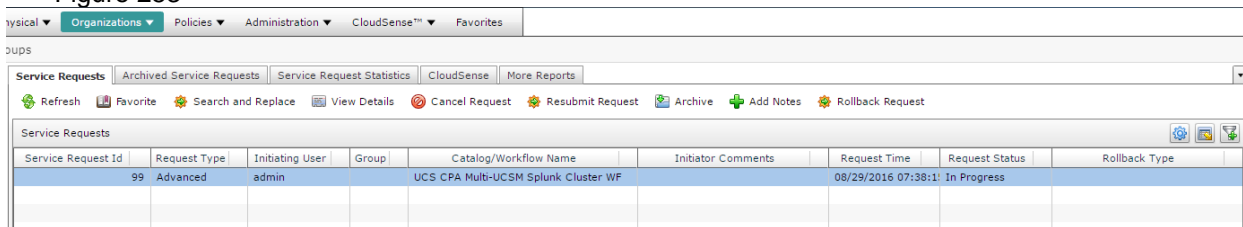
Figure 252



9. Click Submit.

10. Monitor the workflow by going to Organization > Service Requests. Click on the workflow and then click View Details to bring up the Service Request window.

Figure 253



11. Certain steps may require approval, Figure 254 Go to Organizations > My Approvals to approve.

Figure 254

Service Request

Status Refresh

▼ Overview

Request ID	99
Request Type	Advanced
Workflow Name	UCS CPA Multi-UCSM Splunk Cluster WF
Workflow Version Label	3.0
Request Time	08/29/2016 07:38:15 GMT-0700
Request Status	In Progress
Comments	

▼ Ownership

Initiating User	admin
-----------------	-------

Current status for the service request.

- 1 Initiated by admin 08/29/2016 07:38:22
- 2 Multi-UCSM Splunk Cluster Profile 08/29/2016 07:39:11
- 3 Setup Big Data Cluster Env 08/29/2016 07:39:51
- 4 User (Compute) Approval Required Condition is False 08/29/2016 07:39:52
- 5 User (OS) Approval Required Condition is True 08/29/2016 07:39:55
- 6 Approval By admin
- 7 Trigger multiple UCSM specific wfs
- 8 Wait for all single UCSM wfs to finish
- 9 Synchronized Command Execution
- 10 User (Splunk) Approval Required
- 11 Complete

Close

The Multi-UCSM Splunk Cluster workflow should trigger the Single UCSM Server Configuration workflow. In turn, the Single UCSM Server Configuration workflow should trigger multiple baremetal node workflows. These workflows will appear in the service requests list like in the screenshot as shown in Figure 255

Figure 255 Service Requests List

physical ▼ Organizations ▼ Policies ▼ Administration ▼ CloudSense™ ▼ Favorites

oops

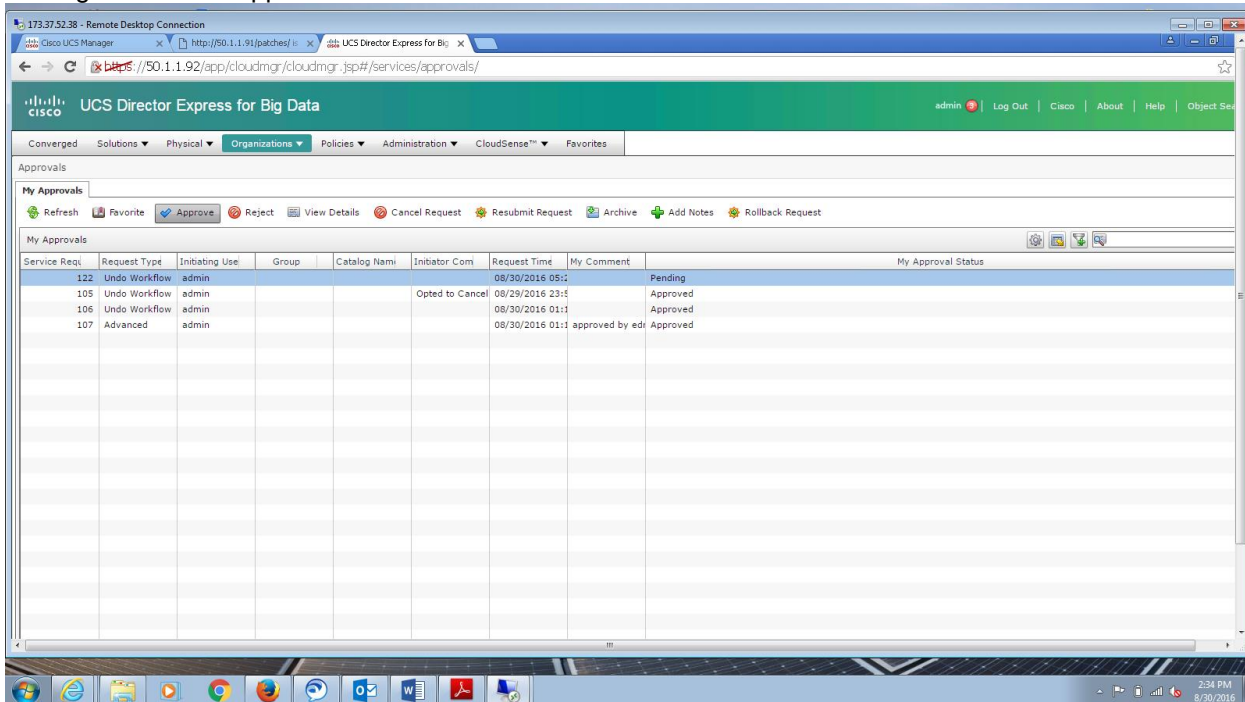
Service Requests Archived Service Requests Service Request Statistics CloudSense More Reports

Refresh Favorite Search and Replace

Service Request Id	Request Type	Initiating User	Group	Catalog/Workflow Name	Initiator Comments	Request Time	Request Status	Rollback Type
121	Advanced	admin		UCS CPA Node BareMetal	UCSM10;sys/rack-unit-4:Monitor PX	08/30/2016 01:26:3	In Progress	
120	Advanced	admin		UCS CPA Node BareMetal	UCSM10;sys/rack-unit-5:Monitor PX	08/30/2016 01:26:3	In Progress	
119	Advanced	admin		UCS CPA Node BareMetal	UCSM10;sys/rack-unit-2:Monitor PX	08/30/2016 01:26:3	In Progress	
118	Advanced	admin		UCS CPA Node BareMetal	UCSM10;sys/rack-unit-3:Monitor PX	08/30/2016 01:26:3	In Progress	
117	Advanced	admin		UCS CPA Node BareMetal	UCSM10;sys/rack-unit-1:Monitor PX	08/30/2016 01:26:3	In Progress	
116	Advanced	admin		UCS CPA Node BareMetal	UCSM10;sys/rack-unit-14:Monitor P	08/30/2016 01:26:3	In Progress	
115	Advanced	admin		UCS CPA Node BareMetal	UCSM10;sys/rack-unit-15:Monitor P	08/30/2016 01:26:2	In Progress	
114	Advanced	admin		UCS CPA Node BareMetal	UCSM10;sys/rack-unit-6:Monitor PX	08/30/2016 01:26:2	In Progress	
113	Advanced	admin		UCS CPA Node BareMetal	UCSM10;sys/rack-unit-18:Monitor P	08/30/2016 01:26:2	In Progress	
112	Advanced	admin		UCS CPA Node BareMetal	UCSM10;sys/rack-unit-20:Monitor P	08/30/2016 01:26:2	In Progress	
111	Advanced	admin		UCS CPA Node BareMetal	UCSM10;sys/rack-unit-16:Monitor P	08/30/2016 01:26:2	In Progress	
110	Advanced	admin		UCS CPA Node BareMetal	UCSM10;sys/rack-unit-19:Monitor P	08/30/2016 01:26:2	In Progress	
109	Advanced	admin		UCS CPA Node BareMetal	UCSM10;sys/rack-unit-10:Monitor P	08/30/2016 01:26:2	In Progress	
108	Advanced	admin		UCS CPA Single UCSM Server Configuration WF	Multi BareMetal WF Monitor[In Progr	08/30/2016 01:23:0	In Progress	
107	Advanced	admin		UCS CPA Multi-UCSM Splunk Cluster WF		08/30/2016 01:19:5	In Progress	

If a service request fails, retry the request. Or rollback the multi-node request to restore the system to the state before the workflow was executed. This will allow changes to be made before reattempting the request. To do this, select the UCS CPA Multi-UCSM Splunk Cluster WF and then click Rollback Request. The rollback may have to be approved.

Figure 256 Approvals



12. Deployment is finished when all workflows have a status of Complete.

Figure 257 Workflow Status

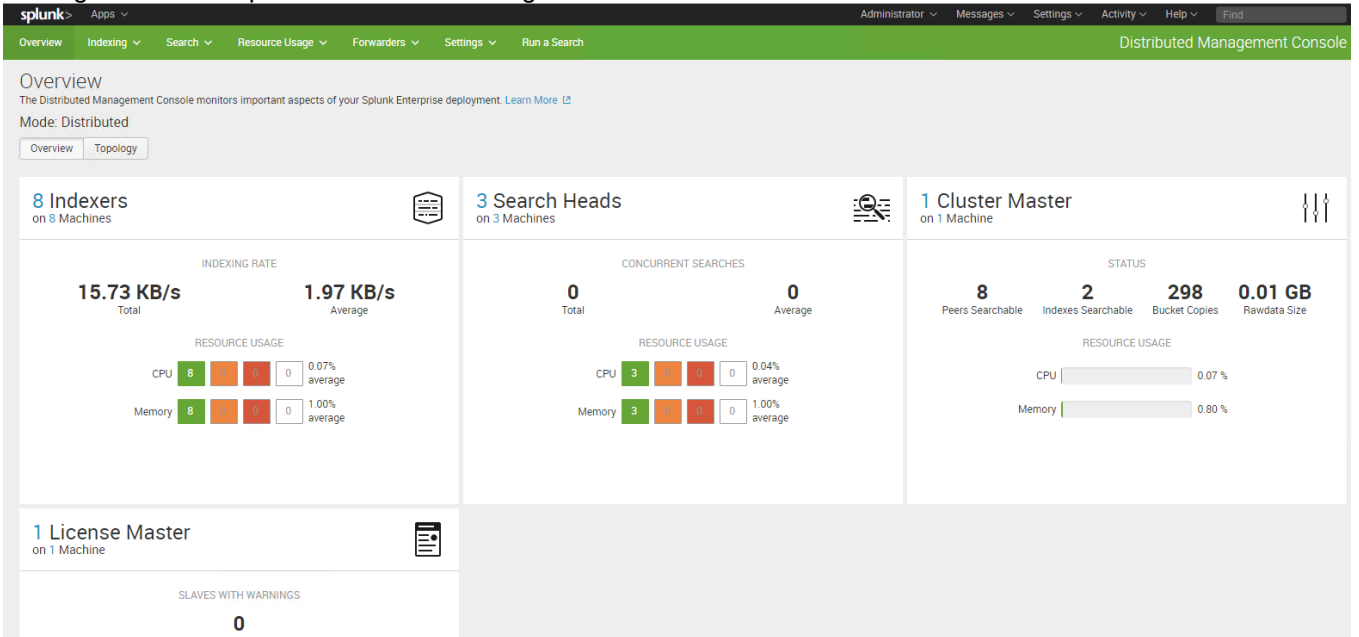
Service Request Id	Request Type	Initiating User	Group	Catalog/Workflow Name	Initiator Comments	Request Time	Request Status
278	Advanced	admin		UCS CPA Node Bare Metal	UCSM10;sys/rack-unit-4:	09/07/2016 03:39:07 GMT-0700	Complete
277	Advanced	admin		UCS CPA Node Bare Metal	UCSM10;sys/rack-unit-5:	09/07/2016 03:39:07 GMT-0700	Complete
276	Advanced	admin		UCS CPA Node Bare Metal	UCSM10;sys/rack-unit-2:	09/07/2016 03:39:06 GMT-0700	Complete
275	Advanced	admin		UCS CPA Node Bare Metal	UCSM10;sys/rack-unit-3:	09/07/2016 03:39:06 GMT-0700	Complete
274	Advanced	admin		UCS CPA Node Bare Metal	UCSM10;sys/rack-unit-1:	09/07/2016 03:39:05 GMT-0700	Complete
273	Advanced	admin		UCS CPA Node Bare Metal	UCSM10;sys/rack-unit-14:	09/07/2016 03:39:05 GMT-0700	Complete
272	Advanced	admin		UCS CPA Node Bare Metal	UCSM10;sys/rack-unit-15:	09/07/2016 03:39:04 GMT-0700	Complete
271	Advanced	admin		UCS CPA Node Bare Metal	UCSM10;sys/rack-unit-6:	09/07/2016 03:39:04 GMT-0700	Complete
270	Advanced	admin		UCS CPA Node Bare Metal	UCSM10;sys/rack-unit-18:	09/07/2016 03:39:03 GMT-0700	Complete
269	Advanced	admin		UCS CPA Node Bare Metal	UCSM10;sys/rack-unit-20:	09/07/2016 03:39:03 GMT-0700	Complete
268	Advanced	admin		UCS CPA Node Bare Metal	UCSM10;sys/rack-unit-16:	09/07/2016 03:39:02 GMT-0700	Complete
267	Advanced	admin		UCS CPA Node Bare Metal	UCSM10;sys/rack-unit-19:	09/07/2016 03:39:02 GMT-0700	Complete
266	Advanced	admin		UCS CPA Node Bare Metal	UCSM10;sys/rack-unit-10:	09/07/2016 03:39:01 GMT-0700	Complete
265	Advanced	admin		UCS CPA Single UCSM Server Configuration WF		09/07/2016 03:35:33 GMT-0700	Complete
264	Advanced	admin		UCS CPA Multi-UCSM Splunk Cluster WF		09/07/2016 03:33:49 GMT-0700	Complete

13. View information about this cluster by going to Solution > Big Data > Accounts. Select the Splunk Accounts tab, then select the account (as specified in the field Big Data Account Name), and click View Details.

14. Open the Splunk Distributed Management Console on the admin1 node by selecting the account in the Splunk Accounts tab and clicking Launch Splunk DMC.

15. Login with the user admin and password specified in Splunk Manager Password created for the cluster.

Figure 258 Splunk Distributed Management Console



Perform further verification by following the steps in the sections Configuring the Deployment Server and onward.