cisco.



Cisco UCS Integrated Infrastructure for Big Data and Analytics with Hortonworks Data Platform 3.0

Design and Deployment Guide of Cisco Integrated Infrastructure for Big Data with Hortonworks Data Platform 3.0 and Cisco UCS C480 M5L Platform

Last Updated: October 21, 2019



About the Cisco Validated Design Program

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

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

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

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

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

All other trademarks mentioned in this document or website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (o8o9R)

© 2019 Cisco Systems, Inc. All rights reserved.

Table of Contents

Executive Summary	7
Solution Overview	8
Introduction	8
Caveats and Limitations	8
Audience	9
Purpose of this Document	9
Solution Summary	9
Scaling the Solution	
Technology Overview	13
Cisco UCS Integrated Infrastructure for Big Data and Analytics	13
Cisco Unified Computing System	13
Cisco UCS 6300 Series Fabric Interconnects	13
Cisco UCS C-Series Rack-Mount Servers	13
Cisco UCS Virtual Interface Cards	
Cisco UCS Manager	
NVIDIA GPU	
NVIDIA CUDA	
Hortonworks Data Platform	
Apache Ambari	19
HDP for Data Access	
Docker Containerization	
YARN Support For Docker	
NVIDIA Docker	
GPU Pooling and Isolation	
Red Hat Ansible Automation	
Solution Design	24
Requirements	24
Rack and PDU Configuration	
Cabling for Cisco UCS C240 M5	
Software Distributions and Versions	
Hortonworks Data Platform (HDP 3.0.1)	
Red Hat Enterprise Linux (RHEL)	
Software Versions	
Fabric Configuration	
Perform Initial Setup of Cisco UCS 6332 Fabric Interconnects	27
Configure Fabric Interconnect A	27
Configure Fabric Interconnect B	
Log Into Cisco UCS Manager	

Upgrade Cisco UCS Manager Software to Version 4.0(2a)	
Add a Block of IP Addresses for KVM Access	
Enable Uplink Ports	
Configure VLANs	
Enable Server Ports	33
Create Pools for Service Profile Templates	
Create an Organization	
Create MAC Address Pools	35
Create a Server Pool	37
Create Policies for Service Profile Templates	
Create Host Firmware Package Policy	
Create QoS Policies	
Create the Local Disk Configuration Policy	
Create the Server BIOS Policy	
Create the Boot Policy	
Create Power Control Policy	
Create Server BIOS Policy	
Create Service Profile Template	51
Configure the Storage Provisioning for the Template	53
Configure Network Settings for the Template	53
Configure the vMedia Policy for the Template	58
Configure the Server Boot Order for the Template	59
Configure the Server Assignment for the Template	60
Configure the Operational Policies for the Template	
Install Red Hat Enterprise Linux 7.5	63
Post OS Install Configuration	
Configure /etc/hosts	
Set Up the Passwordless Login	
Create the Red Hat Enterprise Linux (RHEL) 7.5 Local Repository	89
Create the Red Hat Repository Database	
Set Up Ansible	
Install httpd	
Set Up All Nodes to Use the RHEL Repository	
Upgrade the Cisco Network Driver for VIC1387	94
Install xfsprogs	
Set Up JAVA	
Configure NTP	96
Enable Syslog	
Set ulimit	
Disable SELinux	

Set TCP Retries	
Disable the Linux Firewall	
Disable Swapping	100
Disable Transparent Huge Pages	100
Disable IPv6 Defaults	
Configure Data Drives on Name Node and Other Management Nodes	
Configure Data Drives on Data Nodes	
Configure the Filesystem for NameNodes and Datanodes	105
Cluster Verification	106
Install HDP 3.0.1	
Prerequisites for HDP Installation	
Hortonworks Repository	
Downgrade Snappy on All Nodes	
HDP Installation	
Install and Setup Ambari Server on rhel1	
Install PostgreSQL in rhel2	
Setup Ambari Server On Admin Node(Rhel1)	
Launch the Ambari Server	
Create the Cluster	
Select Version	
Select Hosts	
Hostname Pattern Expressions	
Confirm Hosts	125
Choose Services	
Assign Masters	
Assign Slaves and Clients	
Customize Services	
HDFS	132
MapReduce2	134
YARN	
HBase	135
Zookeeper	135
Storm	136
Ambari Metrics	136
Accumulo	
Atlas	
Kafka	
Knox	
SmartSense	
Spark	

Review	139
Deploy	139
Summary of the Installation Process	
High Availability for HDFS NameNode and YARN ResourceManager	141
Configure the HDFS NameNode High Availability	141
Configure the YARN ResourceManger HA	
HDP Post OS Deployment – Enable GPU Isolation and Scheduling	153
Install the Prerequisites for CUDA	154
Install CUDA	159
Download and Setup NVIDIA CUDA Deep Neural Network library (cuDNN)	
Install and Configure Docker	
Prerequisites	
Install Docker	163
Install nvidia-docker in GPU Nodes	163
Configure Docker	165
Configure YARN to Running Docker Containers	
Enable Cgroups	167
Run Docker on YARN Using the YARN Service API	
Run Docker Container with GPU on YARN	
Prerequisites	
Enable GPU through Ambari	
Verify YARN Configurations	
Setting Up Docker Registry	
Apache Hadoop YARN Distributed Shell	
Run TensorFlow Container Using YARN Distributed Shell	
Bill of Materials	195
About the Authors	
Acknowledgements	



Executive Summary

Years ago, most enterprises relied on batch processing to unlock the value of big data and achieved intelligence. As data size and data sources grew, it also drove a demand for real-time analysis which was catered by Spark and Spark streaming for business-critical decisions. In today's digital world where sensors, Internet of Things (IoT) devices, social media, and online transactions are generating enormous amount of data and companies are struggling to find ways how to better process this data to generate insights and innovation. This drove the next generation of Hadoop which enables Artificial Intelligence and Machine Learning (AI/ML) with deep learning allowing deep learning frameworks and faster compute with CPUs and GPUs to solve problems driving the next generation of platforms.

The emerging trend of Artificial Intelligence requires large amounts of data for its training and Hadoop is a natural fit for storing and retrieving these large amounts of data. Many machine learning tasks especially, deep learning requires the use of GPUs, a specialized, very high-performance processor that is massively parallel in nature. This solution focuses on Hadoop accelerating AI natively where GPUs are part of a Hadoop cluster and are natively scheduled by Hadoop schedulers to process massive amounts of data stored in the same cluster.

Building next-generation big data architecture requires simplified and centralized management, high performance, and a linearly-scaling infrastructure and software platform. Cisco UCS Integrated Infrastructure for Big Data and Analytics with Hortonworks Data Platform (HDP) is an optimal choice where world class performance and reliability are base requirements. It is the strong foundation upon which solutions are built. The Cisco UCS reference architecture has been designed to scale from a small starting solution to thousands of servers and hundreds of petabytes of storage with ease, and all managed from a single pane of glass. Cisco UCS reference architecture also provides the customer with the flexibility an AI workload demands proving choice of server with 2,4,6 or 8 GPUs in a single server catering to different AI compute requirements.

The Hortonworks Data Platform (HDP) is the industry's enterprise-ready open-source Apache Hadoop framework for distributed storage and processing of large datasets. HDP 3.0 bring the capability of managing and scheduling GPUs, Docker containers into Hadoop enabling AI workloads natively in Hadoop. HDP also enables agile application deployment in a containerized micro-services architecture. Containerization makes it possible to run multiple versions of applications for AI/ML/DL workloads. Furthermore, HDP also supports third-party applications in Docker containers and native YARN containers.

Solution Overview

Introduction

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

This solution implements the Cisco UCS Integrated Infrastructure for Big Data and Analytics, a world-class platform specifically designed for demanding workloads that is both easy to scale and easy to manage, even as the requirements grow to thousands of servers and petabytes of storage; and Hortonworks Data Platform, an integrated set of tools designed to enable flexible, fast access to the entire data store, while enabling AI workloads on servers with GPUs.

This CVD implements the following:

- Hortonworks Data Platform 3.0 on Cisco UCS Integrated Infrastructure for Big Data and Analytics
- Install and Enable Docker to be used by YARN 2.0
- Enable CUDA for the GPUs
- Enable GPU as a resource to the Docker Containers through NVIDIA-docker v1
- Enable GPU isolation and scheduling (with Docker Containers) through YARN 2.0
- Downloading a TensorFlow image from NVIDIA Cloud (NGC)
- Adding trusted registries for Docker for YARN 2.0
- Execute a sample TensorFlow job accessing data from Hadoop and running on a Docker container with GPU as a resource scheduled by YARN 2.0
- Installation and setup of the above through Apache Ambari

Caveats and Limitations

The following is beyond the scope of this CVD and therefore is not addressed.

Docker Networking

YARN does not manage Docker networks or Docker multi-host networking. In the implementation of this CVD, container(s) in one host cannot communicate with container(s) in another host.

- Docker Swarm and other docker container orchestration such as Kubernetes are not supported by HDP as they
 might be competing with YARN
- This document considers TensorFlow or other AI applications in only standalone container spawned and scheduled through YARN 2.0. Distributed TensorFlow and other framework support is expected in later release and will be added as part of addendum

NVIDIA-Docker

• As of this release of Hortonworks and this CVD, only nvidia-docker v1 is supported with HDP 3.0

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

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

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

Audience

The intended audience of this document includes, but not limited to, sales engineers, field consultants, professional services, IT managers, partner engineering and customers who want to deploy the Hortonworks Data Platform (HDP 3.0) on Cisco UCS Integrated Infrastructure for Big Data and Analytics. You are assumed to have intermediate level of knowledge for Apache Hadoop and Cisco UCS based scale-out infrastructure.

Purpose of this Document

This document describes the architecture and step by step guidelines of deployment procedures for Hortonworks Data Platform (HDP) 3.0.1 on a 28-node Cisco UCS C240 M5 cluster based on Cisco UCS Integrated Infrastructure for Big Data and Analytics.

Solution Summary

This CVD describes in detail the process for installing Hortonworks 3.0.1 with Apache Spark and Docker containers including the configuration details of the cluster. The current version of Cisco UCS Integrated Infrastructure for Big Data and Analytics offers the following configurations depending on the compute and storage requirements as shown in Table 1.

	Performance		High Capacity
_	(UCS-SP-C240IVI5-A2)	(0C3-5PC240M5L-51)	(UCS-SP-53200-BV)
Servers	16 x Cisco UCS C240 M5 Rack	16 x Cisco UCS C240 M5 Rack	8 x Cisco UCS S3260 Storage
	Servers with SFF drives	Servers with LFF drives	Servers
CPU	2 x Intel Xeon Processor Scalable	2 x Intel Xeon Processor Scalable	2 x Intel Xeon Processor Scalable
	Family 6132 (2 x 14 cores, 2.6 GHz)	Family 4110 (2 x 8 cores, 2.1 GHz)	Family 6132 (2 x 14 cores, 2.6 GHz)

Table 1	Cisco UCS Integrated	Infrastructure for Big	Data and Analy	tics Configuration Option	S
				/ / /	

	Performance (UCS-SP-C240M5-A2)	Capacity (UCS-SPC240M5L-S1)	High Capacity (UCS-SP-S3260-BV)
Memory	6 x 32 GB 2666 MHz (192 GB)	6 x 32 GB 2666 MHz (192 GB)	6 x 32 GB 2666 MHz (192 GB)
Boot	M.2 with 2 x 240-GB SSDs	M.2 with 2 x 240-GB SSDs	M.2 with 2 x 240-GB SSDs
Storage	24 X 2.4 TB 10K rpm SFF SAS HDDs or 12 X 1.6 TB Enterprise Value SATA SSDs	12 x 8 TB 7.2K rpm LFF SAS HDDs + 2 SFF rear hot-swappable 1.6 TB Enterprise Value SATA SSDs	24 x 6 TB 7.2K rpm LFF SAS HDDs
VIC	40 Gigabit Ethernet (Cisco UCS VIC 1387)	40 Gigabit Ethernet (Cisco UCS VIC 1387)	40 Gigabit Ethernet (Cisco UCS VIC 1387)
Storage Controller	Cisco 12-Gbps SAS Modular RAID Controller with 4-GB flash-based write cache (FBWC) or Cisco 12-Gbps Modular SAS Host Bus Adapter (HBA)	Cisco 12-Gbps SAS Modular RAID Controller with 2-GB flash-based write cache (FBWC) or Cisco 12-Gbps Modular SAS Host Bus Adapter (HBA)	Cisco 12-Gbps SAS Modular RAID Controller with 4-GB flash- based write cache (FBWC)
Network Connectivity	Cisco UCS 6332 Fabric Interconnect	Cisco UCS 6332 Fabric Interconnect	Cisco UCS 6332 Fabric Interconnect
GPU (Optional)	2 x NVIDIA TESLA V100 with 32G memory each	2 x NVIDIA TESLA V100 with 32G memory each	

Table 2 High Density GPU Nodes for Data Nodes

	Starter	High Performance
Servers	4 x Cisco UCS C480 M5 Rack Servers	4 x Cisco UCS C480 ML M5 Rack Servers
CPU	2 x Intel Xeon Processor Scalable Family 6142 (2 x 16 cores, 2.6 GHz)	2 x Intel Xeon Processor Scalable Family 6142 (2 x 16 cores, 2.6 GHz)
Memory	12 x 32 GB DDR4 (384 GB)	12 x 32 GB DDR4 (384 GB)
Boot	M.2 with 2 x 960-GB SSDs	M.2 with 2 x 960-GB SSDs
Storage	24 x 1.8 TB 10K rpm SFF SAS HDDs or 12 x 1.6 TB Enterprise Value SATA SSDs	24 x 1.8 TB 10K rpm SFF SAS HDDs or 12 x 1.6 TB Enterprise Value SATA SSDs
VIC	40 Gigabit Ethernet (Cisco UCS VIC 1387)	40 Gigabit Ethernet (Cisco UCS VIC 1387)
Storage Controller	Cisco 12-Gbps SAS Modular RAID Controller with 4-GB flash-based write cache (FBWC) or Cisco 12-Gbps Modular SAS Host Bus Adapter (HBA)	Cisco 12-Gbps SAS Modular RAID Controller with 4-GB flash-based write cache (FBWC) or Cisco 12-Gbps Modular SAS Host Bus Adapter (HBA)
Network Connectivity	Cisco UCS 6332 Fabric Interconnect	Cisco UCS 6332 Fabric Interconnect
GPU	4 x NVIDIA TESLA V100 with 32G memory each	8 x NVIDIA TESLA V100 with 32G memory each and with NVlink

Figure 1 illustrates a 16-node starter cluster. The first rack (left) has 16 Cisco UCS C240 M5 servers. Each link in the figure represents a 40 Gigabit Ethernet link from each of the 16 servers directly connected to a Fabric Interconnect. The second

rack (right) has 8 x Cisco UCS C240 M5 servers and 4 x Cisco UCS C480 ML M5 Servers. Every server is connected to both Fabric Interconnects.



Each Cisco UCS C480 ML M5 has 8 x NVIDIA SXM2 V100 32GB modules with NVLink interconnect. Each Cisco UCS C240 M5 supports up to two PCIe GPU adapters with NVIDIA Tesla V100. For more information about Cisco UCS C240 M5 Sever installation and GPU card configuration rules, go to https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/c/hw/C240M5/install/C240M5/C240M5 appendix 0101. https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/c/hw/C240M5/install/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C240M5/C



Power requirements per rack must be calculated since the exact values will change based on the power needs of the GPUs.

∕∕

2 x Cisco UCS 6454 Fabric Interconnects can also be used in this reference design. For more information about Cisco UCS 6454 Fl, go to <u>https://www.cisco.com/c/en/us/products/collateral/servers-unified-computing/datasheet-c78-</u>

<u>741116.html</u>. Cisco UCS 6332 series FI supports 40 Gb end-to-end and is a good choice for higher bandwidth and faster connections. Cisco UCS 6454 can be considered, if you prefer to use 10/25Gb connections and get faster 40/100 Gb uplinks or move to 25Gb in the future.

Scaling the Solution

Figure 2 illustrates how to scale the solution. Each pair of Cisco UCS 6332 Fabric Interconnects has 28 Cisco UCS C240 M5 servers connected to it. This allows for four uplinks from each Fabric Interconnect to the Cisco Nexus 9332 switch. Six pairs of 6332 FI's can connect to a single switch with four uplink ports each. With 28 servers per FI, a total of 168 servers can be supported. Additionally, the can scale to thousands of nodes with the Nexus 9500 series family of switches.



Figure 2 Scaling the Solution

Technology Overview

Cisco UCS Integrated Infrastructure for Big Data and Analytics

The Cisco UCS Integrated Infrastructure for Big Data and Analytics solution for Hortonworks Data Platform on <u>Cisco UCS</u> <u>Integrated Infrastructure for Big Data and Analytics</u>, is a highly scalable architecture designed to meet a variety of scale-out application demands with seamless data integration and management integration capabilities built using the components described in this section.

Cisco Unified Computing System

Cisco Unified Computing System is a next-generation solution for blade and rack server computing. Cisco UCS integrates a low-latency; lossless 10 and 40 Gigabit Ethernet unified network fabric with enterprise-class, x86-architecture servers. Cisco UCS is an integrated, scalable, multi-chassis platform in which all resources participate in a unified management domain. Cisco UCS accelerates the delivery of new services simply, reliably, and securely through end-to-end provisioning and migration support for both virtualized and non-virtualized systems. Cisco UCS fuses access layer networking and servers. This high-performance, next-generation server system provides a data center with a high degree of workload agility and scalability.

Cisco UCS 6300 Series Fabric Interconnects

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

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

Figure 3 Cisco UCS 6332 UP 32 -Port Fabric Interconnect



Cisco UCS C-Series Rack-Mount Servers

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

It is designed to operate both in standalone environments and as part of Cisco UCS managed configuration, these servers enable organizations to deploy systems incrementally—using as many or as few servers as needed—on a schedule that best meets the organization's timing and budget. Cisco UCS C-Series servers offer investment protection through the capability

to deploy them either as standalone servers or as part of Cisco UCS. One compelling reason that many organizations prefer rack-mount servers is the wide range of I/O options available in the form of PCIe adapters. C-Series servers support a broad range of I/O options, including interfaces supported by Cisco and adapters from third parties.

Cisco UCS C240 M5 Rack-Mount Server

The Cisco UCS C240 M5 Rack-Mount Server (Figure 4) is a 2-socket, 2-Rack-Unit (2RU) rack server offering industry-leading performance and expandability. It supports a wide range of storage and I/O-intensive infrastructure workloads, from big data and analytics to collaboration. Cisco UCS C-Series Rack Servers can be deployed as standalone servers or as part of a Cisco Unified Computing System managed environment to take advantage of Cisco's standards-based unified computing innovations that help reduce customers' Total Cost of Ownership (TCO) and increase their business agility.

In response to ever-increasing computing and data-intensive real-time workloads, the enterprise-class Cisco UCS C240 M5 server extends the capabilities of the Cisco UCS portfolio in a 2RU form factor. It incorporates the Intel Xeon Scalable processors, supporting up to 20 percent more cores per socket, twice the memory capacity, and five times more

Non-Volatile Memory Express (NVMe) PCI Express (PCle) Solid-State Disks (SSDs) compared to the previous generation of servers. These improvements deliver significant performance and efficiency gains that will improve your application performance. The Cisco UCS C240 M5 delivers outstanding levels of storage expandability with exceptional performance, along with the following:

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

Figure 4 Cisco UCS C240 M5 Rack-Mount Server – Front View



Figure 5 Cisco UCS C240 M5 Rack-Mount Server – Rear View



Cisco UCS C480 M5 Rack-Mount Server

The Cisco UCS C480 M5 Rack-Mount Server is a storage and I/O-optimized enterprise-class rack-mount server that delivers industry-leading performance for in-memory databases, big data analytics, virtualization, Virtual Desktop Infrastructure (VDI), and bare-metal applications. The Cisco UCS C480 M5 (Figure 6) delivers outstanding levels of expandability and performance for standalone or Cisco Unified Computing System managed environments in a 4RU form-factor. Because of its modular design, you pay for only what you need. It offers these capabilities:

- Latest Intel Xeon Scalable processors with up to 28 cores per socket and support for two-or four-processor configurations
- 2666-MHz DDR4 memory and 48 DIMM slots for up to 6 Terabytes (TB) of total memory
- 12 PCI Express (PCIe) 3.0 slots
 - Six x 8 full-height, full length slots
 - Six x16 full-height, full length slots
- Flexible storage options with support up to 32 Small-Form-Factor (SFF) 2.5-inch, SAS, SATA, and PCIe NVMe disk drives
- Cisco 12-Gbps SAS Modular RAID Controller in a dedicated slot
- Internal Secure Digital (SD) and M.2 boot options
- Dual embedded 10 Gigabit Ethernet LAN-On-Motherboard (LOM) ports

Figure 6 Cisco UCS C480 M5 Rack-Mount Server – Front View



Figure 7 Cisco UCS C480 M5 Rack-Mount Server – Rear View



For more information about Cisco UCS C480 M5 Rack Server, go to: https://www.cisco.com/c/en/us/products/collateral/servers-unified-computing/ucs-c-series-rack-servers/datasheet-c78-739291.html

Cisco UCS C480 ML M5 Rack Server

The Cisco UCS C480 ML M5 Rack Server is a purpose-built server for Deep Learning. It is storage and I/O optimized to deliver an industry-leading performance for training Models. The Cisco UCS C480 ML M5 delivers outstanding levels of storage expandability and performance options for standalone or Cisco Unified Computing System managed environments in a 4RU form factor. Because of its modular design, you pay for only what you need. It offers these capabilities:

- 8 NVIDIA SXM2 V100 32G modules with NVLink interconnect
- Latest Intel Xeon Scalable processors with up to 28 cores per socket and support for two processor configurations
- 2666-MHz DDR4 memory and 24 DIMM slots for up to 3 terabytes (TB) of total memory
- 4 PCI Express (PCIe) 3.0 slots for 100G UCS VIC 1495
- Flexible storage options with support for up to 24 Small-Form-Factor (SFF) 2.5-inch, SAS/SATA Solid-State Disks (SSDs) and Hard-Disk Drives (HDDs)
- Up to 6 PCIe NVMe disk drives
- Cisco 12-Gbps SAS Modular RAID Controller in a dedicated slot
- M.2 boot options
- Dual embedded 10 Gigabit Ethernet LAN-On-Motherboard (LOM) ports



Figure 8 Cisco UCS C480 ML M5 Purpose Built Deep Learning Server – Front View

Figure 9 Cisco UCS C480 ML M5 Purpose Built Deep Learning Server – Rear View



For more information about Cisco UCS C480 ML M5 Server, go to: <u>https://www.cisco.com/c/dam/en/us/products/collateral/servers-unified-computing/ucs-c-series-rack-servers/c480m5-specsheet-ml-m5-server.pdf</u>

Table 3 lists the features and benefits of Cisco UCS C480 ML M5 Server.

|--|

Feature	Benefits
8 x NVIDIA SXM2 V100 32GB modules with NVLink interconnect	Fast Deep Learning model training
Modular storage support with up to 24 front accessible hot- swappable Hard Disk Drives (HDDs) and Solid-State Disks (SSDs)	Modularity to right-size storage options to match training requirements Flexibility to expand as storage needs increase
High-capacity memory support of up to 3 TB using 128-GB DIMMs	Large memory footprint to deliver performance and capacity for large model training
Up to 6 PCle NVMe drives	Up to 6 Gen3 x4 lanes NVMe drives for extreme I/O performance for faster model training
Support for up to 4 PCIe Generation 3.0 slots	Support for up to four 10/25 or 40/100G Cisco VICs
Hot-swappable, redundant power supplies	Increased high availability

Feature	Benefits
Integrated dual 10-Gbps Ethernet	Increased network I/O performance and additional network options

Cisco UCS Virtual Interface Cards

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

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



Figure 10 Cisco UCS VIC 1387

For more information about Cisco UCS Adapters, go to: <u>https://www.cisco.com/c/en/us/products/interfaces-modules/unified-computing-system-adapters/index.html</u>

Cisco UCS Manager

Cisco UCS Manager (UCSM) resides within the Cisco UCS 6300 Series Fabric Interconnect. It makes the system self-aware and self-integrating, managing all of the system components as a single logical entity. Cisco UCS Manager can be accessed through an intuitive GUI, a CLI, or an XML 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.

For more information about Cisco UCS Manger, go to: <u>https://www.cisco.com/c/en/us/products/servers-unified-computing/ucs-manager/index.html</u>

NVIDIA GPU

Graphics Processing Units or GPUs are specialized processors designed to render images, animation and video for computer displays. They perform this task by running many operations simultaneously. While the number and kinds of operations they can do are limited, they make up for it by being able run many thousands in parallel. As the graphics capabilities of GPUs increased, it soon became apparent that the massive parallelism of GPUs could be put to other uses beside rendering graphics.

NVIDIA GPU used in this document, NVIDIA Tesla V100, is advanced data center GPU built to accelerate AI, HPC, and graphics. It is powered by NVIDIA Volta architecture, comes in 16 and 32 GB configurations.

NVIDIA GPUs bring two key advantages to the table. First, they make possible solutions that were simply not computationally possible before. Second, by providing the same processing power as scores of traditional CPUs they reduce the requirements for rack space, power, networking and cooling in the data center.

NVIDIA CUDA

GPUs are very good at running the same operation on different data simultaneously. This is often referred to as single instruction, multiple data, or SIMD. This is exactly what's needed to render graphics but many other computing problems can benefit from this approach. As a result, NVIDIA created CUDA. CUDA is a parallel computing platform and programming model that makes it possible to use a GPU for many general-purpose computing tasks via commonly used programming languages like C and C++.

In addition to the general-purpose computing capabilities that CUDA enables there is also a special CUDA library for deep learning called the CUDA Deep Neural Network library, or cuDNN. cuDNN makes it easier to implement deep machine learning architectures that take full advantage of the GPU's capabilities.

Hortonworks Data Platform

The Hortonworks Data Platform (HDP 3.0.1) delivers essential capabilities in a completely open, integrated and tested platform that is ready for enterprise usage. With Hadoop YARN at its core, HDP provides flexible enterprise data processing across a range of data processing engines, paired with comprehensive enterprise capabilities for governance, security and operations.

All the integration of the entire solution is thoroughly tested and fully documented. By taking the guesswork out of building out a Hadoop deployment, HDP gives a streamlined path to success in solving real business problems.

Hortonworks Data Platform (HDP) 3.0 delivers significant new features, including the ability to launch apps in a matter of minutes and address new use cases for high-performance deep learning and machine learning apps. In addition, this new version of HDP enables enterprises to gain value from their data faster, smarter, in a hybrid environment.

Apache Ambari

Apache Ambari is a completely open source management platform. It performs provisioning, managing, securing, and monitoring Apache Hadoop clusters. Apache Ambari is a part of Hortonworks Data Platform and it allows enterprises to plan and deploy HDP cluster. It also provides ongoing cluster maintenance and management.

Ambari provides an intuitive Web UI as well as an extensive REST API framework which is very useful for automating cluster operations.

Below are the core benefits that Hadoop operators get with Ambari:

• Simplified Installation, Configuration and Management. Easily and efficiently create, manage and monitor clusters at scale. Takes the guesswork out of configuration with <u>Smart Configs</u> and Cluster Recommendations. Enables repeatable, automated cluster creation with <u>Ambari Blueprints</u>.

- Centralized Security Setup. Reduce the complexity to administer and configure cluster security across the entire platform. Helps automate the setup and configuration of advanced cluster security capabilities such as Kerberos and <u>Apache Ranger</u>.
- Full Visibility into Cluster Health. Ensure your cluster is healthy and available with a holistic approach to monitoring. Configures predefined alerts — based on operational best practices — for cluster monitoring. Captures and visualizes critical operational metrics — using <u>Grafana</u> — for analysis and troubleshooting. Integrated with <u>Hortonworks</u> <u>SmartSense</u> for proactive issue prevention and resolution.
- Highly Extensible and Customizable. Fit Hadoop seamlessly into your enterprise environment. Highly extensible with <u>Ambari Stacks</u> for bringing custom services under management, and with <u>Ambari Views</u> for customizing the Ambari Web UI.

HDP for Data Access

With YARN at its foundation, HDP provides a range of processing engines that allow users to interact with data in multiple and parallel ways, without the need to stand up individual clusters for each data set/application. Some applications require batch while others require interactive SQL or low-latency access with NoSQL. Other applications require search, streaming or in-memory analytics. Apache Solr, Storm and Spark fulfill those needs respectively.

To function as a true data platform, the YARN-based architecture of HDP enables the widest possible range of access methods to coexist within the same cluster avoiding unnecessary and costly data silos.

As shown in Figure 11, HDP Enterprise natively provides for the following data access types:

- Batch Apache MapReduce has served as the default Hadoop processing engine for years. It is tested and relied upon by many existing applications.
- Interactive SQL Query Apache Hive is the de facto standard for SQL interactions at petabyte scale within Hadoop. Hive delivers interactive and batch SQL querying across the broadest set of SQL semantics.
- Search HDP integrates Apache Solr to provide high-speed indexing and sub-second search times across all your HDFS data.
- Scripting Apache Pig is a scripting language for Hadoop that can run on MapReduce or Apache Tez, allowing you to aggregate, join and sort data.
- Low-latency access via NoSQL Apache HBase provides extremely fast access to data as a columnar format, NoSQL database. Apache Accumulo also provides high-performance storage and retrieval, but with fine-grained access control to the data.
- Streaming Apache Storm processes streams of data in real time and can analyze and take action on data as it flows into HDFS.



Docker Containerization

Hortonworks Data Platform (HDP 3.0) makes use of container technology. Containers are conceptually similar to virtual machines, but instead of virtualizing the hardware, a container virtualizes the operating system. With a VM there is an entire operating system sitting on top of the hypervisor. Containers dispense with this time-consuming and resource hungry requirement by sharing the host system's kernel. As a result, a container is far smaller, and its lightweight nature means they can be instantiated quickly. In fact, they can be instantiated so quickly that new application architectures are possible.

Docker is an open-source project that performs operating-system-level virtualization, also known as "containerization." It uses Linux kernel features like namespaces and control groups to create containers. These features are not new, but Docker has taken these concepts and improved them in the following ways:

- Ease of use: Docker makes easier for anyone—developers, systems admins, architects and others—to take advantage of containers in order to quickly build and test portable applications. It allows anyone to package an application on their development system, which can then run *unmodified* on any cloud or bare metal server. The basic idea is to create a "build once, run anywhere" system.
- **Speed:** Docker containers are very fast with a small footprint. Ultimately, containers are just sandboxed environments running on the kernel, so they take up few resources. You can create and run a Docker container in seconds. Compare this to a VM which takes much longer because it has to boot up a full virtual operating system every time.
- **Modularity:** Docker makes it easy to take an application and breaks its functionality into separate individual containers. These containers can then be spun up and run as needed. This is particularly useful for cases where an application needs to hold and lock a particular resource, like a GPU, and then release it once it's done using it. Modularity also enables each component, i.e., container to be updated independently.
- **Scalability:** modularity enables scalability. With different parts of the system running in different containers it becomes possible, and with Docker, it becomes easy to connect these containers together to create an application, which can then be scaled out as needed.

YARN Support For Docker

Containerization provides YARN support for Docker containers, which makes it easier to bundle libraries and dependencies along with their application, allowing third-party applications to run on Apache Hadoop (for example, containerized applications), enabling:

• Faster time to deployment by enabling third-party apps.

- The ability to run multiple versions of an application, enabling users to rapidly create features by developing and testing new versions of services without disrupting old ones.
- Improved resource utilization and increased task throughput for containers, yielding faster time to market for services.
- Orchestration of stateless distributed applications.
- Packaging libraries for Spark application, eliminating the need for operations to deploy those libraries cluster wide.

Figure 12 Containerized Application on Apache Hadoop YARN 3.1



As shown in Figure 12, YARN Services Framework in addition with Docker containerization, it is now possible to run both existing Hadoop frameworks, such as Hive, Spark, etc., and new containerized workloads on the same underlying infrastructure. Apache Hadoop 3.1 further improved these capabilities to enable advanced use cases such as TensorFlow and HBase.

NVIDIA Docker

Docker containers are platform-agnostic, but also hardware-agnostic. This presents a problem when using specialized hardware such as NVIDIA GPUs which require kernel modules and user-level libraries to operate. As a result, Docker does not natively support NVIDIA GPUs within containers.

One of the early workarounds to this problem was to fully install the NVIDIA drivers inside the container and map in the character devices corresponding to the NVIDIA GPUs (for example, /dev/nvidiao) on launch. This solution is brittle because the version of the host driver must exactly match the version of the driver installed in the container. This requirement drastically reduced the portability of these early containers, undermining one of Docker's more important features.

To enable portability in Docker images that leverage NVIDIA GPUs, NVIDIA developed nvidia-docker, an open-source project hosted on GitHub that provides the two critical components needed for portable GPU-based containers:

driver-agnostic CUDA images; and a Docker command line wrapper that mounts the user mode components of the driver and the GPUs (character devices) into the container at launch.

nvidia-docker is essentially a wrapper around the docker command that transparently provisions a container with the necessary components to execute code on the GPU.



As of the publishing of this CVD, Hortonworks only supports nvidia-docker version 1.

GPU Pooling and Isolation

GPU pooling and isolation allows GPU to be a first-class resource type in Hadoop, making it easier for customers to run machine learning and deep learning workloads.

- Compute-intensive analytics require not only a large compute pool, but also a fast and expensive processing pool with GPUs in tandem
- Customers can share cluster-wide GPU resources without having to dedicate a GPU node to a single tenant or workload
- GPU isolation dedicates a GPU to an application so that no other application has access to that GPU

When it comes to resource scheduling, it is important to recognize GPU as a resource. YARN extends the resource model to more flexible mode which makes it easier to add new countable resource-types. When GPU is added as resource type, YARN can schedule applications on GPU machines. Furthermore, by specifying the number of requested GPU to containers, YARN can find machines with available GPUs to satisfy container requests.

Figure 13 YARN Scheduling for GPU/Non-GPU Applications



么

When GPU scheduling is enabled, YARN can schedule non-GPU applications such as LLAP, Tez, and etc. to servers without GPU. Moreover, YARN can allocate GPU applications such as TensorFlow, Caffe, MXNet, and so on, to servers with GPU.

Red Hat Ansible Automation

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



For more information about Ansible, go to: <u>https://www.redhat.com/en/technologies/management/ansible</u>

Solution Design

Requirements

This CVD describes the architecture and deployment procedures for Hortonworks Data Platform (HDP) 3.0.1 on a 28 Cisco UCS C240 M5 node cluster based on Cisco UCS Integrated Infrastructure for Big Data and Analytics. The solution goes into detail configuring HDP 3.0.1 on the Cisco UCS Integrated infrastructure for Big Data. In addition, it also details the configuration for Hortonworks Dataflow for various use cases.

The cluster configuration consists of the following:

- 2 Cisco UCS 6332UP Fabric Interconnects
- 24 Cisco UCS C240 M5 Rack-Mount servers
- 4 Cisco UCS C480 ML M5 Rack-Mount server
- 8 NVIDIA GPU in each Cisco UCS C480 ML M5
- 2 Cisco R42610 standard racks
- 4 Vertical Power distribution units (PDUs) (Country Specific) per rack

Rack and PDU Configuration

Each rack consists of two vertical PDUs. The first rack consists of two Cisco UCS 6332UP Fabric Interconnects, 16 Cisco UCS C240 M5 Rack Servers connected to each of the vertical PDUs for redundancy; thereby ensuring availability during power source failure. The second rack consists of 8 Cisco UCS C240 M5 Servers and 4 Cisco UCS C480 ML M5 connected to each of the vertical PDUs for redundancy; thereby ensuring availability during power source failure, similar to the first rack.

Please contact your Cisco representative for country specific information.

Table 4 Port Configuration on Fabric Interconnects

Port Type	Port Number
Network	29-32
Server	1-28

Cabling for Cisco UCS C240 M5

The Cisco UCS C240 M5 rack server is equipped with 2 x Intel Xeon Processor Scalable Family 6132 (2 x 14 cores, 2.6 GHz), 192 GB of memory, Cisco UCS Virtual Interface Card 1387 Cisco 12-Gbps SAS Modular Raid Controller with 4-GB FBWC, 26 x 1.8 TB 10K rpm SFF SAS HDDs or 12 x 1.6 TB Enterprise Value SATA SSDs, M.2 with 2 x 240-GB SSDs for Boot.

Figure 14 illustrates the port connectivity between the Fabric Interconnect, and Cisco UCS C240 M5 server. Sixteen Cisco UCS C240 M5 servers are used in Master rack configurations.

Figure 14 Cisco UCS C240 M5 and 6300 Series Fabric Interconnect Port Connectivity



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

https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/c-series_integration/ucsm3-2/b_C-Series-Integration_UCSM3-2/b_C-Series-Integration_UCSM3-2_chapter_o10.html?bookSearch=true

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

https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/c-series_integration/ucsm3-2/b_C-Series-Integration_UCSM3-2/b_C-Series-Integration_UCSM3-2_chapter_o10.html?bookSearch=true

Software Distributions and Versions

The software distributions required versions are listed below.

Hortonworks Data Platform (HDP 3.0.1)

The Hortonworks Data Platform supported is HDP 3.0.1. For more information, go to: http://www.hortonworks.com.

Red Hat Enterprise Linux (RHEL)

The operating system supported is Red Hat Enterprise Linux 7.5. For more information, go to: http://www.redhat.com.

Software Versions

The software versions tested and validated in this document are shown in Table 5.

Layer	Component	Version or Release
Compute	Cisco UCS C240 M5	C240M5.4.0.2a
	Cisco UCS C480 ML M5	
Network	Cisco UCS 6332	UCS 4.0(2a)
	Cisco UCS VIC1387 Firmware	4.3(2a)

Table 5Software Versions

Layer	Component	Version or Release
	Cisco UCS VIC1387 Driver	3.1.137.5
Storage	SAS Expander	65.02.15.00
	Cisco 12G Modular Raid controller	50.6.0-1952
	Red Hat Enterprise Linux Server	7.5
	Cisco UCS Manager	4.0(2a)
Software	HDP	3.0.1
	Docker	1.13.1
	Ansible	2.4.6.0
	Nvidia-docker	1.0.1
GPU	CUDA	9.2
	NVIDIA GPU Driver	396.44

4

The latest drivers can be downloaded from this link:

https://software.cisco.com/download/home/283862063/type/283853158/release/3.1%25283%2529.

The latest supported RAID controller driver is already included with the RHEL 7.5 operating system.

Cisco UCS C240 M5 Rack Servers with Intel Scalable Processor Family CPUs are supported from Cisco UCS firmware 3.2 onwards.

Fabric Configuration

This section provides the details to configure a fully redundant, highly available Cisco UCS 6332 fabric configuration. The following is the high-level workflow to setup Cisco UCS:

- Initial setup of the Fabric Interconnect A and B
- Connect to Cisco UCS Manager using virtual IP address of using the web browser
- Launch Cisco UCS Manager
- Enable server and uplink ports
- Start discovery process
- Create pools and polices for service profile template

- Create Service Profile template
- Create service profile for each server from service profile template
- Associate Service Profiles to servers5

Perform Initial Setup of Cisco UCS 6332 Fabric Interconnects

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

Configure Fabric Interconnect A

To configure Fabric Interconnect A, follow these steps:

- 1. Connect to the console port on the first Cisco UCS 6332 Fabric Interconnect.
- 2. At the prompt to enter the configuration method, enter console to continue.
- 3. If asked to either perform a new setup or restore from backup, enter setup to continue.
- 4. Enter y to continue to set up a new Fabric Interconnect.
- 5. Enter 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 Mgmto IPv4 address.
- 12. Enter the Mgmto IPv4 netmask.
- 13. Enter the IPv4 address of the default gateway.
- 14. Enter the cluster IPv4 address.
- 15. To configure DNS, answer y.
- 16. Enter the DNS IPv4 address.
- 17. Answer y to set up the default domain name.
- 18. Enter the default domain name.
- 19. Review the settings that were printed to the console, and if they are correct, answer yes to save the configuration.
- 20. Wait for the login prompt to make sure the configuration has been saved.

Configure Fabric Interconnect B

To configure Fabric Interconnect B, follow these steps:

- 1. Connect to the console port on the second Cisco UCS 6332 Fabric Interconnect.
- 2. When prompted to enter the configuration method, enter console to continue.
- 3. The installer detects the presence of the partner Fabric Interconnect and adds this fabric interconnect to the cluster. Enter y to continue the installation.
- 4. Enter the admin password that was configured for the first Fabric Interconnect.
- 5. Enter the Mgmto 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 about configuring Cisco UCS 6332 Series Fabric Interconnect, go to:

https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/ucs-manager/GUI-User-Guides/Getting-Started/3-2/b_UCSM_Getting_Started_Guide_3_2/b_UCSM_Getting_Started_Guide_3_2_chapter_o1oo.html

Log Into Cisco UCS Manager

To log into Cisco UCS Manager, follow these steps:

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

Upgrade Cisco UCS Manager Software to Version 4.0(2a)

This document assumes the use of UCS 4.0(2a). Refer to the <u>Cisco UCS 4.0 Release</u> (upgrade Cisco UCS Manager software and UCS 6332 Fabric Interconnect software to version 4.0(2a). Also, make sure the Cisco UCS C-Series version 4.0(2a) software bundles are installed on the Fabric Interconnects.

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

Add a Block of IP Addresses for KVM Access

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

1. Select the LAN tab at the top of the left window.

- 2. Select Pools > root > IpPools > Ip Pool ext-mgmt.
- 3. Right-click IP Pool ext-mgmt.
- 4. Select Create Block of IPv4 Addresses.

```
Figure 15 Adding a Block of IPv4 Addresses for KVM Access Part 1
```

Æ	All	LAN / Pools / root / IP Poo
	 Threshold Policies VMQ Connection Policies 	General IP Addresses
器	 usNIC Connection Policies vNIC Templates 	Delete
≞	 Sub-Organizations Pools 	Create Block of IPv4 Address Create Block of IPv6 Address
	 root ♥ IP Pools 	Create DNS Suffix Create IPV4 WINS Server
	IP Pool ext-mgmt	Show Pool Usage Create Block of IPv4 Addresses
	 IP Pool iscsi-initiator-pool MAC Pools Sub-Organizations 	Create Block of IPv6 Addresses Copy Copy XML
	 Traffic Monitoring Sessions Fabric A Fabric B 	Delete

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

Figure 16 Adding Block of IPv4 Addresses for KVM Access Part 2

Create Block of IPv4 Addresses					
From : 10.13.1.11	Size : 28 🗘				
Subnet Mask : 255.255.255.0	Default Gateway : 10.13.1.1				
Primary DNS : 0.0.0.0	Secondary DNS : 0.0.0.0				

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

Enable Uplink Ports

To enable uplinks ports, follow these steps:

1. Select the Equipment tab on the top left of the window.

- Select Equipment > Fabric Interconnects > Fabric Interconnect A (primary) > Fixed Module.
- 3. Expand the Unconfigured Ethernet Ports section.
- 4. Select port 29-32 that is connected to the uplink switch, right-click, then select Reconfigure > Configure as Uplink Port.
- 5. Select Show Interface and select 40GB for Uplink Connection.
- 6. A pop-up window appears to confirm your selection. Click Yes then OK to continue.
- 7. Select Equipment > Fabric Interconnects > Fabric Interconnect B (subordinate) > Fixed Module.
- 8. Expand the Unconfigured Ethernet Ports section.
- 9. Select port number 29-32, which is connected to the uplink switch, right-click, then select Reconfigure > Configure as Uplink Port.
- 10. Select Show Interface and select 40GB for Uplink Connection.
- **11**. A pop-up window appears to confirm your selection. Click Yes then OK to continue.

All Fouinment / Fahric Interc.../ Fahr Æ Disable Port 22 8 Port 23 몲 Port 24 Configure as FCoE Uplink Port Configure as FCoE Storage Port Port 25 Configure as Appliance Port Port 26 Unconfigure Port 27 Port 28 Port 29 Port 30 Disable Port Port 32 Reconfigure v -0 Unconfigure FC Ports Show Interface PSUs Fabric Interconnect B (subordinate)

Figure 17 Enabling Uplink Ports Part1

Figure 17 Enabling Uplink Ports Part2

Port 29	Actions
Port 30	
Port 31	Disable Port
Port 32	Reconfigure 🔻
FC Ports	Unconfigure
PSUs	Show Interface
 Fabric Interconnect B (subordinate) 	

Figure 18 Enabling Uplink Ports Part 3

General Faults Event	16	
Actions	Properties	
Enable Interface Disable Interface	ID : 31 Slot ID : 1 Fabric ID : A	
	Transport Type : Ether Port : sys/switch-A/slot-1/switch-ether/port-31	
	Link Profile : default Admin Speed : 1 Gbps 10 Gbps 40 Gbps	

Configure VLANs

VLANs are configured as in shown in Table 6.

Table 6	VLAN Configurations	
VLAN	NIC Port	Function
VLAN13	etho	Data

The NIC will carry the data traffic from VLAN13. A single vNIC is used in this configuration and the Fabric Failover feature in Fabric Interconnects will take care of any physical port down issues. It will be a seamless transition from an application perspective.

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

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



- 5. Enter vlan13 for the VLAN Name.
- 6. Keep multicast policy as <not set>.
- 7. Select Common/Global for vlan16.
- 8. Enter 13 in the VLAN IDs field for the Create VLAN IDs.
- 9. Click OK and then, click Finish.
- 10. Click OK in the success message box.

Figure 20	Creating VLAN for Data
-----------	------------------------

Create VLANs	? ×
VLAN Name/Prefix : vlan13	
Multicast Policy Name : <pre></pre> <pre></pre> <pre></pre> <pre>Create Multicast Policy</pre>	
Common/Global Fabric A Fabric B Both Fabrics Configured Differently	
You are creating global VLANs that map to the same VLAN IDs in all available fabrics. Enter the range of VLAN IDs.(e.g. " 2009-2019", " 29,35,40-45", " 23", " 23,34-45") VLAN IDs : 13	
Sharing Type : None Primary Isolated Community	

11. Click OK and then click Finish.

Enable Server Ports

To enable server ports, follow these steps:

- 1. Select the Equipment tab on the top left of the window.
- Select Equipment > Fabric Interconnects > Fabric Interconnect A (primary) > Fixed Module.
- 3. Expand the Unconfigured Ethernet Ports section.
- 4. Select all the ports that are connected to the Servers 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 Unconfigured Ethernet Ports section.
- 8. Select all the ports that are connected to the Servers right-click them, and select Reconfigure > Configure as a Server Port.
- 9. A pop-up window appears to confirm your selection. Click Yes, then OK to continue.



Figure 21 Enabling Server Ports

After the Server Discovery, Port 29-32 will be a Network Port and 1-28 will be Server Ports.

cisco.	UCS Manager		8 🔽	3 41				• = •	00	9 C
.	All	Equipment / Fat	ric Interconne /	Fabric Interconne	ec / Fixed Module	/ Ethernet	Ports			
	 Fabric Interconnect A (primary) 	Ty Advanced Filter	+ Export Print	VAI VUnce	onfigured 🗸 Network	Server	V FCoE Uplink	Inified Uplink	>>	¢
<u>.e.</u>	 Fixed Module 	Slot	Aggr. Port ID	Port ID	MAC	If Role	If Type	Overall Status	Admin State	
	Ethernet Ports	1	0	1	70:7D:89:F3:60	Server	Physical	1 Up	* Enabled	
	Port 1	1	0	2	70:7D:B9:F3:60	Server	Physical	1 Up	t Enabled	
	Port 2	3	0	3	70:7D:89:F3:60	Server	Physical	t Up	t Enabled	
9	Port 3	1	0	4	70:7D:B9:F3:60	Server	Physical	t Up	t Enabled	
-	Port 4	1	0	5	70:7D:89:F3:60	Server	Physical	t Up	Enabled	
	Port 5	1	0	6	70:7D:B9:F3:60	Server	Physical	1 Up	Enabled	
	Port 6	1	0	7	70.7D.89.F3.60	Server	Physical	1 Up	t Enabled	
	Port 7	1	0	8	70:7D:89:F3:60	Server	Physical	1 Up	1 Enabled	
30	Port 8	1	0	9	70:7D:B9:F3:60	Server	Physical	1 Up	* Enabled	
	Port 9	1	0	10	70:7D:B9:F3:60	Server	Physical	1 Up	* Enabled	

Figure 22 Ports Status after the Server Discover

Create Pools for Service Profile Templates

Create 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.

To configure an organization within the Cisco UCS Manager GUI, follow these steps:

- 1. Click Quick Action icon on the top right corner in the right pane in the Cisco 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.

cisco.	UCS Manager	8 😳 🕗 🐼 1 6 3 41	Creste Service Profile
-	All	* Servers	Create Service Profile (expert)
			Create Organization
	▶ ucs-11	Service Profiles	Create VLANs
	▶ ucs-12	All Failed Active Passive Disassociated Pending Hierarchical Pending Activities	Create VSAN
윪	▶ ucs-13	Ty Advanced Filter + Export - ⊕ Print	Create User
_	▶ ucs-14	Name User Label Overall Status Assoc State	Server
=	▶ ucs-15	Service Profile ucs-1 OK Associated	sys/rack-unit-1
	► ucs-16	Service Profile ucs-10 CK Associated	sys/rack-unit-10
	▶ ucs-17	Service Profile ucs-11 TOK Associated	sys/rack-unit-11

Create (rganization	? ×
Name :	ucs	
Description :		
		OK Cancel

Create MAC Address Pools

To create MAC address pools, follow these steps:

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

æ	All	LAN / Pools / root / MAC Pools
	 vNIC Templates Sub-Organizations 	+ - Ty Advanced Filter + Export
윪	✓ Pools	MAC Pool default
	 IP Pools 	MAC Pool ucs
Q	 IP Pool ext-mgmt IP Pool iscsi-initiator-pool 	
=	MAC Pools Sub-Omanizations	Create MAC Pool
	 Traffic Monitoring Sessions ▶ Fabric A 	
20	Fabric B	
0	Define Name and Description Name :	
-	Description :	

0	Define Name and Description	Name : ucs
2	Add MAC Addresses	Description :
		Next> Cancel

Figure 23 Specifying first MAC Address and Size

010000 0 010		101 000 00		
First MAC Address:	00:25:85:00:00:00	Size : 512	\$	
To ensure uniqueness MAC prefix: 00:25:85:xx:xx:xx	of MACs in the LAN fab	ric, you are strong	gly encouraged to u	se the following
				(

12. Click Finish.
| | | Create MAC F | Pool | ? × |
|---|-----------------------------|--------------------|-----------------------|-------------------|
| | Define Name and Description | + - Ty Advanced Fi | tter 🛧 Export 🚔 Print | \$ |
| - | | Name | From | То |
| 3 | AUD MAU AUDIESSES | [00:25:85:00:00: | 00:25:85:00:00:00 | 00:25:85:00:01:FF |
| | | | (+) Add (1) Delete | |
| | | < Prev | (liste) Fin | ish Cancel |

13. When the message box displays, click OK.



Create a Server Pool

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

To configure the server pool within the Cisco UCS Manager GUI, follow these steps:

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

		Create Server Pool	? ×	2
0	Set Name and Description	Name : ucs		
2	Add Servers	Description :		
			Next> Cancel	

8. Select all the Cisco UCS C240M5 servers to be added to the server pool that was previously created (ucs), then Click >> to add them to the pool.

		Create Serv	ver Poo	ol			: ×
•	Set Name and Description		Servers			Pooled Servers	
0	Add Servere			A C	•		*
~		G. SL. R.	U., PID	A 5		G., SI., Na., US., PID Ad., Se.,	
		1	u.,	U W.		No data available	
		2	u	U W	. I		
		3	U	U W.		<<	
		4	U	U. W.			
		5	U	U. W.			
		0	u	U W.	201 - E		
		7	u	U., W.			
			1 1 1 1				_
		Model: Serial Number; Vendor:				Model: Serial Number Vendor:	
						<prev cance<="" finish="" td=""><td>el</td></prev>	el

- 9. Click Finish.
- 10. Click OK and then click Finish.

Create Policies for Service Profile Templates

Create Host Firmware Package Policy

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

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

- 1. Select the Servers tab in the left pane in the UCS Manager GUI.
- 2. Select Policies > root.
- 3. Right-click Host Firmware Packages.
- 4. Select Create Host Firmware Package.
- 5. Enter the required Host Firmware package name (ucs).
- 6. Select Simple radio button to configure the Host Firmware package.
- 7. Select the appropriate Rack package that has been installed.
- 8. Click OK to complete creating the management firmware package
- 9. Click OK.

Create Host Firmware Package	? ×
Name : ucs	-
Description :	
How would you like to configure the Host Firmware Package?	
● Simple ◯ Advanced	
Blade Package : <not set=""></not>	
Rack Package : 3.2(2b)C	
Service Pack : www.service.example.com	
The images from Service Pack will take precedence over the images from Blade or Rack Package Excluded Components:	_
GPUs	
HBA Option ROM	
Host NIC	
✓ Local Disk	
PSU PSU	
SAS Expander	
SAS Expander Regular Firmware	•
OK (Cancel

Create QoS Policies

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

Platinum Policy

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



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

Create Qo	S Policy	? >
Name : Platinu Egress	1	
Priority :	Platinum 👻	
Burst(Bytes) : Rate(Kbps) :	line-rate	
Host Control :	None Full	
		OK Cancel

Set Jumbo Frames

To set Jumbo frames and enable QoS, follow these steps:

- 1. Select the LAN tab in the left pane in the Cisco UCS Manager GUI.
- 2. Select LAN Cloud > QoS System Class.
- 3. In the right pane, select the General tab
- 4. In the Platinum row, enter 9216 for MTU.
- 5. Check the Enabled Check box next to Platinum.
- 6. In the Best Effort row, select none for weight.
- 7. In the Fiber Channel row, select none for weight.
- 8. Click Save Changes.
- 9. Click OK.

OoS System Class								
LAN Pin Groups	Deleviter	Feebl	-4.0-6	Deskat	Walaha		Walaht	MTH
 Threshold Policies 	Priority	Chabl	ed Co3	Drop	weight		(%)	MIU
 VLAN Groups 					-		0.010.000	
 VLANs 	Platinum	v	5		10		N/A	9216
Appliances	Gold	- m	S.	7			N/A	
Fabric A	GUIG		4	61	9	*	inter	normal
 Fabric B 	Silver		2	V	A	٣	N/A	normal
 VLANs 			12					
Internal LAN	Bronze		1	\checkmark	7	٣	N/A	normal
 Internal Fabric A 	14.792							
 Internal Fabric B 	Effort	\mathbb{X}	Any	×	none	۲	50	normal
Threshold Policies	Fibre	\square	3		none	٠	50	

Create the Local Disk Configuration Policy

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

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

Create Local Dis	k Configuration Policy	? ×
Name	: ucs	
Description	я.	
Mode	: Any Configuration	
Protect Configuration	. 💌	
disassociated with the server. In that case, that server if the local disk co FlexFlash	a configuration error will be raised when a new serv nfiguration in that profile is different.	ice profile is associated with
FlexFlash State	: Disable Enable	
lf FlexFlash State is disabled Please en sure SD cards are n	, SD cards will become unavailable immediately. ct in use before disabling the FlexFlash State.	
FlexFlash RAID Reporting St	ate : 💿 Disable 🔿 Enable	
		OK Cancel

Create the Server BIOS Policy

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

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, follow these 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 your preferred BIOS policy name (ucs).
- 6. Change the BIOS settings as shown in the following figures.
- 7. Only changes that need to be made are in the Processor and RAS Memory settings.

uluilu cisco.	UCS Manager	🛞 👽 🛆 🕥 3 5 5 37		8 9 0 8 C
₼	All	Servers / Policies / root / BIOS Policies / ucs		
8	BIOS Defaults BIOS Policies OPICY	Main Advanced Boot Options Server Management Events Processor Intel Directed IO RAS Memory Serial Port USB PC	I QPI LOM and PCIe Slots Trusted Platform	Graphics Configuration
蔬 =		V Advanced Filter	Value	\$
	usNIC	Altitude	Platform Default	× /
<u> </u>	 Boot Policies Diagnostics Policies 	CPU Hardware Power Management Boot Performance Mode	Platform Default Platform Default	
	Graphics Card Policies Host Firmware Packages	CPU Performance Core Multi Processing	Enterprise All	▼ ▼
=	detault ucs	DRAM Clock Throttling	Performance	· · · · · · · · · · · · · · · · · · ·
•	ucs-3.22b IPMI Access Profiles	Direct Cache Access Energy Performance Tuning	Enabled Platform Default	▼ ▼
	KVM Management Policies Local Disk Config Policies Maintenance Policies	Enhanced Intel SpeedStep Tech Execute Disable Bit	Disabled Platform Default	▼ / /

æ	All	Servers / Policies / noot / BIOS Policies / ucs		
8	 > BIOS Defaults ▲ ✓ BIOS Policies 	Main Advanced Boot Options Server Management Events Processor Intel Directed IO RAS Memory Serial Port USB PCI	OPI LOM and PCIe Slots Trusted Platform	Graphics Configuration
윪	SRIOV	Ty Advanced Filter		\$
=	UCS1	BIOS Setting	Value	
-	usNIC	Frequency Floor Override	Platform Default	¥.
Q	 Boot Policies 	Intel HyperThreading Tech	Enabled	v
	 Diagnostics Policies 	Intel Turbo Boost Tech	Enabled	V
	 Graphics Card Policies 	Intel Virtualization Technology	Disabled	· ·
	 Host Firmware Packages default 	Channel Interleaving	Auto	2
	ucs	IMC Inteleave	Platform Default	V
20	ucs-3.22b	Memory Interleaving	Platform Default	v
	 IPMI Access Profiles 	Rank Interleaving	Platform Default	y
	 KVM Management Policies 	Sub NUMA Clustering	Platform Default	· · ·
	 Local Disk Config Policies Maintenance Policies 	Local X2 Apic	Platform Default	×

-m	All	Servers / Policies / root / BIOS Policies / ucs	
=	 ▶ BIOS Defaults ▲ BIOS Policies 	Main Advanced Boot Options Server Management Events Processor Intel Directed IO RAS Memory Serial Port USB PCI OPI LOM and PCIe Slots Trusted Platform	Graphics Configuration
뮮	SRIOV	T _p Advanced Filter ↑ Export	\$
	UCS1	BIOS Setting Value	
	usNIC	Max Variable MTRR Setting Platform Default	T
▣	 Boot Policies 	P STATE Coordination HW ALL	v
	 Diagnostics Policies 	Package C State Limit Platform Default	V
	 Graphics Card Policies 		(
	▼ Host Firmware Packages	Processor C State Disabled	· · · · · · · · · · · · · · · · · · ·
	default	Processor C1E Disabled	
	ucs	Processor C3 Report Disabled	T
20	ucs-3.22b	Processor C6 Report Disabled	v
	 IPMI Access Profiles 	Processor C7 Report Disabled	•
	 KVM Management Policies 	Breaster CMCI	<i>i</i>
	Local Disk Config Policies	Plocessui Cwici Platform Default	· · · · · · · · · · · · · · · · · · ·
	Maintenance Policies	Power Technology Performance	7, 🔽

æ	All 👻	Servers / Policies / root / BIOS Policies / ucs	
8	 BIOS Defaults BIOS Policies 	Main Advanced Boot Options Server Management Events Processor Intel Directed IO RAS Memory Serial Port USB PCI QPI LOM and PCIe Slots Trusted Platform	Graphics Configuration
윪	SRIOV	Ty Advanced Filter	¢
≣	UCS1	BIOS Setting Value	
	usNIC	Energy Performance Performance	₹
	 Boot Policies 	Adjacent Cache Line Prefetcher Enabled	V
	 Diagnostics Policies 	DCU IP Prefetcher Enabled	V
	 Graphics Card Policies 	DCU Streamer Prefetch Enabled	Ţ
_	 Host Firmware Packages 	Hardware Prefetcher Enabled	T
	detault	UPI Prefetch Enabled	
20	ucs-3.22b	LLC Prefetch Enabled	Ţ
	 IPMI Access Profiles 	XPT Prefetch Enabled	7
	 KVM Management Policies 	Demand Scrub Enabled	V
	 Local Disk Config Policies 	Patrol Scrub Enabled	· ·

₩	All	Servers / Policies / root / BIOS Policies / ucs	
8	 ✓ Service Profile Templates ▲ ✓ root ♥ 	Main Advanced Boot Options Server Management Events Processor Intel Directed IO RAS Memory Serial Port LISB PCI OPI LOM and PCIe Stats Trusted Platform Graphics Configuration	
윪	 Service Template ucs Sub-Organizations 	Year Advanced Filter ↑ Export ⊕ Print	۵
	▼ Policies	BIOS Setting Value	
	🔻 root 💎	Adjacent Cache Line Prefetcher Enabled	
	 Adapter Policies 	DCU IP Prefetcher Enabled	
	BIOS Defaults BIOS Belicies	DCU Streamer Prefetch Enabled	
	* BIOS Policies	Hardware Prefetcher Enabled	
	ucs	UPI Prefetch Enabled V	
-	UCS1	LLC Prefetch Enabled	
20	usNIC	XPT Prefetch Enabled	
	 Boot Policies 	Demand Scrub Enabled	
	 Diagnostics Policies 	Patrol Scrub	
	 Graphics Card Policies 		
	▼ Host Firmware Packages	Workload Conngulation	•
alialia cisco.	UCS Manager		۲
æ	All	Servers / Policies / root / BIOS Policies / ucs	
8	 Service Profile Templates 	Main Advanced Boot Options Server Management Events	
	🔻 root 👽	Processor Intel Directed IO RAS Memory Serial Port USB PCI QPI LOM and PCIe Slots Trusted Platform Graphics Configurat	ion
윪	Service Template ucs	Ty Advanced Filter 🔶 Export 👼 Print	
	 Sub-Organizations 		-

	Service Template ups								
ठॅठ	 Sub-Organizations 	🏹 Advanced Filter 🔶 Export 🚔 Print	Ty Advanced Filter 🔶 Export 🎂 Print						
-	▼ Policies	BIOS Setting	Value						
	🔻 root 👽	DDR3 Voltage Selection	Platform Default	V					
Q	 Adapter Policies 	DRAM Refresh Rate	1x	V					
	 BIOS Defaults 	LV DDR Mode	Platform Default	▼					
	 BIOS Policies SDIOV 	Mirroring Mode	Platform Default						
	ucs	NUMA optimized	Platform Default	y					
-	UCS1	Memory RAS configuration	Maximum Performance	V					
9	usNIC								

Create the Boot Policy

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

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



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

Create Boot Policy											? ×
Name :	ucs										
Description :											
Reboot on Boot Order Change :											
Enforce vNIC/vHBA/iSCSI Name : 1	v										
Boot Mode :	● Legacy 🔵 Uefi										
WARNINGS: The type (primary/secondary) does n The effective order of boot devices w If Enforce vNIC/vHBA/iSCSIName i If it is not selected, the vNICs/vHBAs	not indicate a boot order presence. within the same device class (LAN/Stora s selected and the vNIC/vHBA/iSCSId are selected if they exist, otherwise the	nge/iSCSI) is bes not exist, ∋ vNIC/vHBA	determined a config ei with the lo	l by PCIe I ror will be west PCIe	bus scan (reported. bus scan	order. order is use	ed.				
 Local Devices 	Boot Order										
Add Local Disk	+ - Ty Advanced Filter	🕈 Export	🖶 Print							₽	▲
Add Local LUN	Name	Or 🔺	vNIC/	Туре	WWN	LUN	Slot	Boot	Boot	Descr	
Add Local JBOD	Local CD/DVD	1									
Add SD Card	Local Disk	2									
Add Internal USB	🔻 LAN	3									
Add External USB	LAN ethD		ethO	Prima							
Add Embedded Local LUN									ок	Cance	el
Add LAN Boot	? ×										

Create Power Control Policy

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

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

Æ	All	Servers / Policies / root / Power Control Policies
	 Dour Folicies Diagnostics Policies 	Power Control Policies Events
	 Graphics Card Policies 	+ - Ty Advanced Filter ↑ Export ⊕ Print
ਨੱਨ	 Host Firmware Packages 	Name
-	 IPMI Access Profiles 	default
-	 KVM Management Policies 	ucs
	 Local Disk Config Policies 	
-	 Maintenance Policies 	
=	 Management Firmware Packages 	
	Memory Policy	
	Power Control Policies	to Device Control Ballon
	 Power Sync Policies 	te Power Control Policy
-0	Scrub Policies	
	 Serial over LAN Policies 	

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

Name	: UCS	
Description	:	
Fan Speed Pol	icy: Performance *	
Power Capp	ing	
lisco UCS Man nore powertha egardless of th	ager only enforces power capping when the serv n is currently available. With sufficient power, all eir priority.	ers in a power group require servers run at full capacity
Cisco UCS Man nore power tha egardless of th	ager only enforces power capping when the serv n is currently available. With sufficient power, all eir priority.	ers in a power group require servers run at full capacity

Create Server BIOS Policy

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

- In Cisco UCS Manager, click the Servers tab in the navigation pane. 1.
- Select Policies > root > Sub-Organization > UCS-HDP > BIOS Policies. 2.
- Right-click BIOS Policies. 3.
- Select Create BIOS Policy. 4.
- Enter C240M5-BIOS as the BIOS policy name. 5.

٩.	Policies 👻	Policies / root / S	Sub-Organizat	ions / UCS-HDP / BIOS Policies		
	 Threshold Policies iSCSI Authentication Profiles vMedia Policies vNIC/vH8A Placement Policies Sub-Organizations 	BIOS Policies + - + Export Name > root	Events		_	_
	 Adapter Policies BIOS Policies Boot Policies Diagnostics Policies Graphics Card Policies Graphics Card Policies Host Firmware Packages IPMI Access Profiles KVM Management Policies Local Disk Config Policies Maintenance Policies Management Firmware Packages Power Control Policies 			Create BIOS Policy Name : Description : Reboot on BIOS Settings Change :	UCS-HDP-BIOS BIOS for Cisco UCS Cluster	3
	Power Sync Policies Scrub Policies					OK Cancel

 \times

Processor Intel Directed IO RAS Memory Serial Port USB PC	QPI LOM and PCIe Slots Trusted Platform Graphics Configuration	
T₂ Advanced Filter 🔶 Export 🚔 Print		
BIOS Setting	Value	
Altitude	Platform Default	
CPU Hardware Power Management	Platform Default	Ψ.
Boot Performance Mode	Platform Default	Ψ.
CPU Performance	Enterprise	Ψ.
Core Multi Processing	All	Ŧ
DCPMM Firmware Downgrade	Platform Default	
DRAM Clock Throttling	Performance	v
Direct Cache Access	Enabled	*
Energy Performance Tuning	Platform Default	Ŧ
Enhanced Intel SpeedStep Tech	Enabled	Ŧ
Execute Disable Bit	Platform Default	*
Frequency Floor Override	Platform Default	*
Intel HyperThreading Tech	Enabled	*
Energy Efficient Turbo	Platform Default	Ŧ
Intel Turbo Boost Tech	Enabled	*
Intel Virtualization Technology	Disabled	*
Intel Speed Select	Platform Default	*
hannel Interleaving	Auto	Ŧ
IC Inteleave	Platform Default	Ψ.
emory Interleaving	Platform Default	Ψ.
ank Interleaving	Platform Default	Ψ.
ub NUMA Clustering	Platform Default	Ψ.
cal X2 Apic	Platform Default	Ψ.
ax Variable MTRR Setting	Platform Default	Ψ.
STATE Coordination	HW ALL	Ψ.
ickage C State Limit	Platform Default	Ψ.
itonomous Core C-state	Platform Default	Ψ.
ocessor C State	Disabled	T.
ocessor C1E	Disabled	₹.
ocessor C3 Report	Disabled	Υ.
acessor C6 Report	Disabled	¥.
ocessor C7 Report	Disabled	Ψ.
pcessor CMCI	Platform Default	Ψ.
ower Technology	Performance	Ŧ

Ty Advanced Filter 🔶 Export 🚔 Print		\$
BIOS Setting	Value	
Energy Performance	Performance	Ψ.
ProcessorEppProfile	Performance	Ψ.
Adjacent Cache Line Prefetcher	Enabled	Υ.
DCU IP Prefetcher	Enabled	v
DCU Streamer Prefetch	Enabled	Υ.
Hardware Prefetcher	Enabled	*
UPI Prefetch	Enabled	Ψ.
LLC Prefetch	Enabled	Υ.
XPT Prefetch	Enabled	Υ.
Core Performance Boost	Platform Default	Y.
Downcore control	Platform Default	Υ.
Global C-state Control	Platform Default	Υ.
L1 Stream HW Prefetcher	Platform Default	Ψ.
L2 Stream HW Prefetcher	Platform Default	Υ.
Determinism Slider	Platform Default	Ψ.
IOMMU	Platform Default	Υ.
Bank Group Swap	Platform Default	T
Bank Group Swap	Platform Default	v .
Chipselect Interleaving	Platform Default	Ψ.
Configurable TDP Control	Platform Default	Ψ.
AMD Memory Interleaving	Platform Default	Ψ.
AMD Memory Interleaving Size	Platform Default	Ψ.
SMEE	Platform Default	Ψ.
SMT Mode	Platform Default	Ψ.
SVM Mode	Platform Default	Ψ.
Demand Scrub	Enabled	Ψ.
Patrol Scrub	Enabled	*
Workload Configuration	Platform Default	Ψ.

Policies / root / Sub-Organizations / TPC-BDA / BIOS Policies / BDA-BIOS		
Main Advanced Boot Options Server Management Events		
Processor Intel Directed IO RAS Memory Serial Port USB PCI QPI LOM and PCIe Slot	s Trusted Platform Graphics Configuration	
▼ _F Advanced Filter 🔶 Export 🚔 Print		\$
BIOS Setting	Value	
DDR3 Voltage Selection	Platform Default	
DRAM Refresh Rate	Platform Default	
LV DDR Mode	Platform Default	
Mirraring Made	Platform Default	
NUMA optimized	Platform Default	
Memory RAS configuration	Maximum Performance	

Create Service Profile Template

To create the Service Profile Template, follow these steps:

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

#	All Servers Service Profiles	Servers / Service Profile Templates Service Profile Templates + - Ty Advanced Filter + Export + Print
쁆	Service Profile Templates	Name
	 Policies Pools Schedules default exp-bkup-outdate fi-reboot infra-fw 	Create Service Profile Template

The Create Service Profile Template window appears.

To identify the service profile template, follow these steps:

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

		Create Service Profile Template	? ×
0	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 template and enter a description.	o this
	Storage Provisioning	Name : UCS	- í
3	Networking	The template will be created in the following organization. Its name must be unique within this organization. Where : org-root	
(4)	SAN Connectivity	Type : Initial Template (Updating Template	
5	Zoning	Specify how the UUID will be assigned to the server associated with the service generated by this template. UUID	
6	vNIC/vHBA Placement	UUID Assignment: Hardware Default	
0	vMedia Policy	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.	
8	Server Boot Order	Optionally enter a description for the profile. The description can contain information about when and where the service profile should be up	sed.
9	Maintenance Policy		1
10	Server Assignment	Next> Finish Can	el

Configure the Storage Provisioning for the Template

To configure storage policies, follow these steps:

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

		Create Service Prof	le Templat	te			
ŋ	Identify Service Profile	Optionally specify or create a Storage Profile, and select a local disk configuration policy.					
	Template	Specific Storage Profile Stora	age Profile Policy	Local Disk Configuration Policy]		
2	Storage Provisioning		gor formor only				
3	Networking	Local Storage: ucs 🔻					
		Create Local Disk Configuration F	olicy	Modie Protect Configuration	: Any Configuration		
4	SAN Connectivity			If Protect Configuration is se	et, the local disk configuration is		
5	Zoning			with the server. In that case, a raised when a new service pro-	a configuration error will be office is associated with		
6	vNIC/vHBA Placement			that server if the local disk co different.	nfiguration in that profile is		
	vMedia Doliov			FlexFlash FlexFlash State	: Disable		
	Unicale Folicy			l f FlexFlash State is disabled, unavailable immediately	, SD cards will become		
8	Server Boot Order			Please ensure SD cards are n FlexFlash State.	ot in use before disabling the		
9	Maintenance Policy			FlexFlash RAID Reporting St	ate : Disable		

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

Configure Network Settings for the Template

To configure the network settings for the templates, follow these steps:

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

1		Create Servic	e Profile Template			? ×
	Identify Service Profile	Optionally specify LAN	I configuration information.			
	Template	Dynamic vNIC Connecti	on Policy: Select a Policy to use (no D	ynamic vNIC Policy by default) 🔻		1
0	Storage Provisioning		Create Dynamic vNIC Connection	n Policy		
0						
0	SAN Connectivity	How would you like to o	onfigure LAN connectivity? No vNICs () Use Connectivity Policy			
	Zoning	Click Add to specify an	e or more vNICs that the server should	use to connect to the LAN.		
-		Name	MAC Address	Fabric ID	Native VLAN	
0	vNIC/vHBA Placement			No data available		
	vMedia Policy					
8	Server Boot Order					1
9	Maintenance Policy		0 D	elete 🕀 🗚 🦷 Maddu		
10	Server Assignment			< Prev	Next> Finish	Cancel

- 4. The Create vNIC window displays. Name the vNIC as eth0.
- 5. Select ucs in the Mac Address Assignment pool.
- 6. Select the Fabric A radio button and check the Enable failover check box for the Fabric ID.
- 7. Check the VLAN13 check box for VLANs and select the Native VLAN radio button.
- 8. Select MTU size as 9000.
- 9. Select adapter policy as Linux.
- 10. Select QoS Policy as Platinum.
- 11. Keep the Network Control Policy as Default.
- 12. Click OK.

Create vNIC			? ×
Name : eth0 MAC Address			
MAC Address Assignment: ucs(512/512) 🔻			
Create MAC Pool The MAC address will be automatically assig	gned from the selected po	ool.	
Use vNIC Template : 🗖 Fabric ID : 💿 Fabric A	○ Fabric B	Z Enable Failover	_
VLAN in LAN cloud will take the precedence ov	er the Appliance Cloud wh	hen there is a name clash.	
VLANs VLAN Groups			
	default	\circ	
\checkmark	vlan13_data	\circ	
	vlan14	\bigcirc	
			Cancel

Create vNIC		(?) >
CDN Source : • vNIC Name User Defined		
MTU : 9000 Warning		
Make sure that the MTU has the same value in the QoS corresponding to the Egress priority of the selected Qo	System Class S Policy.	
Pin Group : <pre><not set=""> </not></pre>	Create LAN Pin Group	
 Operational Parameters 		
Adapter Performance Profile		
Adapter Policy : Linux 🔻	Greate Ethernet Adapter Policy	
QoS Policy : platinum ¥	Create QoS Policy	
Network Control Policy : default 🔻	Create Network Control Policy	
Connection Policies		
	(OK Cancel

		Create Service	Create Service Profile Template					
1	Identify Service Profile	Optionally specify LAN configuration information.						
2	Template Storage Provisioning	Dynamic vNIC Connection	Policy: Select a Policy to use (no D	ynamic vNIC Policy by default) 🔻		^		
3	Networking		Create Dynamic vNIC Connection	on Policy				
4	SAN Connectivity	How would you like to con	figure LAN connectivity?					
5	Zoning	Simple Expert	ovNICs Use Connectivity Policy or more vNICs that the server should	use to connect to the LAN.	Notice VI AN			
6	vNIC/vHBA Placement	► vNIC eth0	Derived	A B	INSIDE VLAN			
7	vMedia Policy							
8	Server Boot Order							
9	Maintenance Policy							
10	Server Assignment		i De	lete 🕀 Add 🕕 Modify		••		
				< Prev	Next > Finish	Cancel		

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

- 13. Click ${\tt Next}$ to continue with SAN Connectivity.
- 14. Select no vHBAs for How would you like to configure SAN Connectivity?

		Create Service Profile Template	? ×
1	Identify Service Profile	Optionally specify disk policies and SAN configuration information.	
	Template	How would you like to configure SAN connectivity?	
2	Storage Provisioning	◯ Simple ◯ Expert ④ No vHBAs ◯ Use Connectivity Policy	
3	Networking	This server associated with this service profile will not be connected to a storage area network.	
•	SAN Connectivity		
5	Zoning		
6	vNIC/vHBA Placement		
7	vMedia Policy		
8	Server Boot Order		
9	Maintenance Policy		
10	Server Assignment	< Prev Next > Finish Cance	*

15. Click ${\tt Next}$ to continue with Zoning .

		Create Service Pr	ofile Template			? ×
	Identify Service Profile	Specify zoning information				
	Template	Zoning configuration involves the 1. Select vHBA Initiator(s) (vf	e following steps : HBAs are created on storage page)			
2	Storage Provisioning	 Select vHBA Initiator Grou Add selected Initiator(s) to 	p(s) selected Initiator Group(s)			
3	Networking	Select vHBA Initiators		Select vHBA Initiato	r Groups	
4	SAN Connectivity	Name	-	Name	Storage Connection Poli	cy Na
6	Zoning	No data available	>> Add To >>		No data available	
6	vNIC/vHBA Placement					
	vMedia Policy					
8	Server Boot Order				Delete 🕀 Add 🙃 Modify	
9	Maintenance Policy					
10	Server Assignment			< Prev	Next > Finish	Cancel

16. Click ${\tt Next}$ to continue with vNIC/vHBA placement.

		Create Ser	vice Profile Template		? ×			
0	Identify Service Profile Template	Specify how vNIC	Specify how vNICs and vHBAs are placed on physical network adapters					
		vNIC/vHBA Placeme in a server hardware	ent specifies how vNICs and vHBAs are place configuration independent way.	ed on physical network adapters (mezzanine)				
2	Storage Provisioning	Select Placement:	Let System Perform Placement 🔹	Create Placement Policy				
3	Networking	System will perfor	m automatic placement of vNICs and vHBAs	based on PCI order.				
		Name	Address	Order	-			
4	SAN Connectivity	vNIC eth0	Derived	1				
6	Zoning							
0	vNIC/vHBA Placement							
0	vMedia Policy							
8	Server Boot Order		T Move µp ♦ Move Dow	n B Delete C Reorder: O Modity				
9	Maintenance Policy							
10	Server Assignment			< Prev Next>	Finish Cancel			

17. Click Next to configure vMedia Policy.

Configure the vMedia Policy for the Template

To configure the vMedia policy for the template, follow these steps:

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

		Create Service Profile Template	? ×
1	Identify Service Profile Template	Optionally specify the Scriptable vMedia policy for this service profile template.	
2	Storage Provisioning	vMedia Policy: Select vMedia Policy to use 🔻	
3	Networking	Create vMedia Policy	
4	SAN Connectivity	The default boot policy will be used for this service profile.	
6	Zoning		
6	vNIC/vHBA Placement		
7	vMedia Policy		
8	Server Boot Order		
9	Maintenance Policy		
10	Server Assignment	< Prev Next > Finish Ca	ncel

Configure the Server Boot Order for the Template

To set the boot order for the servers, follow these steps:

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

		Create Serv	ice Profile T	emplate						?	\times
0	Identify Service Profile	Optionally specify th	e boot policy for this se	rvice profile ter	mplate.						
	Template	Post Policy									
2	Storage Provisioning	Name	: ucs		Create	Boot Policy					
3	Networking	Description	:						~		
	SAN Connectivity	Enforce vNIC/vHBA	viscsi Name : Yes								
		Boot Mode WARNINGS:	: Legacy	r							
5	Zoning	The type (primary/se The effective order of	econdary) does not indi of boot devices within th	cate a boot ord ne same device	er presence. class (LAN/Si	torage/iSCSI) is c	letermined b	y PCIe bus so	an order.		
6	vNIC/vHBA Placement	If Enforce vNIC/vHE If it is not selected, t	A /iSCSIName is selec he vNICs/vHBAs are se	ted and the vN lected if they e	IC/vHBA/iSCS xist, otherwise	l does not exist, a the vNIC/vHBA t	a config erro with the lowe	rwill be repor st PCIe bus s	ted. can order is u	ised.	
7	vMedia Policy	Boot Order	ed Eilter 🔺 Evport	Drint						ж	
		+ - to Advan				LUN Norse	Slot Num	Post No.	Root Dath		
8	Server Boot Order			n Type	40.4014	LON Name	SIOL NUTL	BOOLINA	BOOL Path	Descripti	
	Maintenance Policy	Local Disk	2								
	Server Assignment	🖝 LAN	3								•
10 3	ooso Assignment					< ا	Prev	Next >	Finish	Cancel)

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

		Create Service Profile Template ? X
1	Identify Service Profile Template	Specify how disruptive changes such as reboots, network interruptions, and firmware upgrades should be applied to the server associated with this service profile.
2	Storage Provisioning	⊖ Maintenance Policy
3	Networking	Select a maintenance policy to include with this service profile or create a new maintenance policy that will be accessible to all service profiles.
4	SAN Connectivity	
5	Zoning	
6	vNIC/vHBA Placement	No maintenance policy is selected by default. The service profile will immediately reboot when disruptive changes are applied.
0	vMedia Policy	
8	Server Boot Order	
9	Maintenance Policy	
10	Server Assignment	< Prev Next > Finish Cancel

Configure the Server Assignment for the Template

To assign the servers to the pool, In the Server Assignment window, follow these steps:

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

		Create Service Profile Template				
0	Identify Service Profile	Optionally specify a server pool for this service profile template.				
	Template Storage Devisioning	Pool Assignment: ucs V Create Server Pool	_			
	Networking	Select the power state to be applied when this profile is associated with the server.				
4	- SAN Connectivity	Outprovide Down				
5	Zoning	The service profile template will be associated with one of the servers in the selected pool. If desired, you can energy an additional server nool policy qualification that the selected server must meet. To do so, select the qualification				
6	vNIC/vHBA Placement	from the list. Server Pool Qualification : <a href="https://www.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.server.pool.org/list.ser</th> <th></th>				
•	vMedia Policy	Restrict Migration : 🗹				
8	Server Boot Order	⊖ Firmware Management (BIOS, Disk Controller, Adapter)				
9	Maintenance Policy	If you select a host firmware policy for this service profile, the profile will update the firmware on the server that it is associated with. Otherwise the system uses the firmware already installed on the associated server.				
10	Server Assignment	<pre></pre>	• ncel			

Configure the Operational Policies for the Template

To configure the operational policies for the template, in the Operational Policies Window, follow these steps:

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

		Create Service Profile Template	? ×						
	Identify Service Profile Template	Optionally specify information that affects how the system operates.							
2	Storage Provisioning	 BIOS Configuration If you want to override the default BIOS settings, select a BIOS policy that will be associated with this service profile 	-						
3	Networking	BIOS Policy: ucs 🔻							
4	SAN Connectivity		▶						
6	Zoning	External IPMI Management Configuration	_						
6	vNIC/vHBA Placement	Management IP Address) Management IP Address						
	vMedia Policy	Monitoring Configuration (Thresholds)							
8	Server Boot Order	Power Control Policy Configuration Power control policy determines power allocation for a server in a given power group.							
9	Maintenance Policy	Power Control Policy : ucs V							
10	Server Assignment	< Prev Next> Finish Cance							
-	Operational Policies								

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



The Create Service Profiles from Template window appears.

laming Prefix : ucs			
Name Suffix Starting Number:	1		
lumber of Instances :	28		

Association of the Service Profiles will take place automatically.

Install Red Hat Enterprise Linux 7.5

This section provides detailed procedures to install Red Hat Enterprise Linux 7.5 using Software RAID (OS based Mirroring) on Cisco UCS C240 M5 servers. There are multiple ways to install the Red Hat Linux operating system. The installation procedure described in this deployment guide uses the KVM console and virtual media from Cisco UCS Manager.



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

1. Log into the Cisco UCS 6332 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. In the right pane, click the KVM Console >>.



5. Click OK on the KVM Console – Select IP address pop-up window.

KVM Console-Select IP Address	×
Equipment derived: 10.16.1.82 (Outband) 	
Launch Java KVM Console OK Cancel	\supset

6. Click the link to launch the KVM console.

```
7redirect_url=https://10.16.1.10/app/4_0_2_80a/kvm.html%3F%26kvmlpAddr%3D10.16.1.82
```

KVM server certificate has been accepted. Click this link to continue loading the KVM client application: https://10.16.1.10/app/4_0_2_80a/kvm.html?&kvmIpAddr=10.16.1.82

7. Point the cursor over the top right corner, select the Virtual Media tab.



- 8. Click the Activate Virtual Devices found in Virtual Media tab.
- 9. Click the Virtual Media tab again to select CD/DVD.

	8002
	o 🕁 📀
	8 ≡ ≎ 6 % © ? ⊖
	Create Image Deactivate
	Removable Disk
	Floppy Disk
No Signal	

6

Virtual [Disk Management	×
CD/DVD	Choose File No file chosen Read Only Map Drive	
To share file area.	es/folders you can drag and drop them in the area below or in the video dis	play
	Drop files/folders here	

10. Select Choose File in the Virtual Disk Management windows.

11. Browse to the Red Hat Enterprise Linux Server 7.5 installer ISO image File.

The Red Hat Enterprise Linux 7.5 DVD is assumed to be on the client machine.

0		Oper	ı				x
🔄 🕘 🔻 🚺	▶ Thi	is PC → Local Disk (C:) → ISOs		~ ¢	Search IS(Ds	م
Organize 🔻 New	folder						
🔆 Favorites	^	Name	Date modified	Туре		Size	
🛄 Desktop		intel-server-7.5-x86_64-dvd	10/10/2018 9:44 AM	Disc Imag	je File	4,509,696 KB	
Downloads							
mecent places							
🌉 This PC							
📄 Desktop	=						
Documents							
🚺 Music							
Pictures							
Videos							
DVD Drive (D:)	IR						
👽 Network	~						
	File na	me: rhel-server-7.5-x86_64-dvd		~	Disc Ima	ge File	~
					Open	- (ancel

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

13. Click Map Drive after selecting the .iso file.

Virtual D	isk Management	×
CD/DVD	Choose File rhel-server-76_64-dvd.iso Read Only Map Drive	
To share files area.	/folders you can drag and drop them in the area below or in the video display	
	Drop files/folders here	
4		

- 14. In the KVM window, select the KVM tab to monitor during boot.
- **15**. In the KVM window, select the Macros > Static Macros > Ctrl-Alt-Del button in the upper left corner.
- 16. Click OK.
- **17.** Click OK to reboot the system.
- 18. Press F6 key on the keyboard to select install media.



Press F6 on your keyboard as soon as possible when the screen below appears to avoid the server reboot again.



19. On reboot, the machine detects the presence of the Red Hat Enterprise Linux Server 7.5 install media.



20. Select the Install Red Hat Enterprise Linux 7.5.



21. Skip the Media test and start the installation. Select language of installation and click Continue.

e to use during the installation	process?
e to use during the installation	process?
English N En	
English > Eng	glish (United States)
Afrikaans Eng	glish (United Kingdom)
Amharic	glish (India)
Arabic	glish (Australia)
Assamese _	glish (Canada)
Asturian	glish (Denmark)
Belarusian	gush (Ireland)
Bulgarian	glish (New Zealand)
Bengali Eng	alish (Hong Kong SAR China)
Bosnian En	glish (Philippines)
Catalan Eng	glish (Singapore)
Czech Eng	glish (South Africa)
Welch	glish (Zambia)
Dapish	glish (Zimbabwe)
Eng	glish (Botswana)
German End	olish (Antiqua & Barbuda)
	Afrikaans En Amharic En Arabic En Assamese En Asturian En Belarusian En Bulgarian En Bosnian En Catalan En Catalan En Uvelsh En Danish En German En

22. Select Date and time, which pops up another window as shown below:



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



24. Click INSTALLATION DESTINATION.


25. This opens a new window with the boot disks. Make the selection and choose I will configure partitioning. Click Done.

INSTALLA	TION DESTINATION			PRISE LINUX 7.5 INSTALLATION
Device Select	t ion device(s) you'd like to ins	tall to. They will be left untouched until	you click on the main menu's "Beg	in Installation" button.
Local Stand	1676.72 GIB	1675 72 G/B	223 57 GB	223 57 GIB
		1010.12 0.0		13.37 010
1M0008 B free	SEAGATE ST1800MM sdf / 1676.72 GiB	10008 SEAGATE ST1800MM0008 free sdg / 1676.72 GiB free	ATA Mircon_5100_MTFD sdh / 1592.5 KiB free	ATA Mircon_5100_MTFD sdl / 1592.5 KiB free
Specialized i	k Network Disks		Disks le	ft unselected here will not be touched
Add a di	sk			
_			Disks le	ft unselected here will not be touche
ther Stora	ge Options			
Partitioning				
Automati	cally configure partitioning.	I will configure partitioning.		
I would b	ke to make additional space a	vailable.		
Encryption	an data. You'll out a monolour			
_ encypt i	ny una roun ser a passinana			

- 26. This opens the new window for creating the partitions. Click the + sign to add a new partition as shown below, boot partition of size 2048 MB.
- 27. Click Add MountPoint to add the partition.

MANUAL PARTITIONING			RED HAT ENTERPRISE	LINUX 7.5 INSTALLATION
			圈 us	Halpi
New Red Hat Enterprise Linux 7.5 Inst You haven't created any mount points for your 1 7.5 installation yet. You can Click here to create them externatically Create new mount points by clicking the 41 b New mount points will use the following partition	allation Red Hat Enterprise Linux utton. aning scheme:			
LVM				
•	ADD A NEW M More custon after creatin	IOUNT POINT nization options are availabl g the mount point below.	e	
	Mount Point:	/boot	nts for your Red Hat Enter etails here	prise Linux 7.5 installation,
	Desired Capacity:	2048MB		
	[Cancel Add mount po	sint	
+ - C				
AVAILABLE SPACE 447.14 GIB 447.14 GIB				
2 storage devices selected				Reset All

28. Change the Device type to RAID and make sure the RAID Level is RAID1 (Redundancy) and click Update Settings to save the changes.

MANUAL PARTITIONING	RED HAT ENTERPRISE LINUX 7.5 INSTALLATION 편한 us
New Red Hat Enterprise Linux 7.5 Installation	sdll
SYSTEM	Mount Point: Device(s):
7000t 1953 M sdl1	/boot
	Desired Capacity: ATA Mircon_5100_MTFD (sdl) and 1 1953 M/B
	Modify
	Device Type: BAID Lond
	RAID T Encrypt RAID (Redundance)
	File System
*	xfs 🔹 🐨 Reformat
	Label: Name:
	boot
	Update Settings
+ - 0	Note: The settings you make on this screen will not be applied until you click on the main menu's Begin
Available space 445.23 GIB 47.14 GIB	initialation button.
2 storage devices selected	Reset All

29. Click the + sign to create the swap partition of size 2048 MB as shown below.

MANUAL PARTITIONING	ation	boot	ľ	RED HAT	ENTERPRISE LINUX 7.5 INSTALLATIO
SYSTEM /boot boot	1953 MiB 义	Mount Point: /boot Desired Capacity: 1953 MiB			Device(s): ATA Mircon.5100_MTFD (sdl) and 1 other
	ADD A NEW M More custom after creating	OUNT POINT nization options are availa g the mount point below.	ble		Modify
	Mount Point: Desired Capacity:	swap 2048MB Cancel Add mount	Point] Encrypt	RAID1 (Redundancy). 👻
					Name: boot
+ - C AVAILABLE SPACE 443.33 GIB TOTAL SPACE 447.14 GIB				Note T be app	Update Settings The settings you make on this screen will not led until you click on the main menu's 'Begin Installation' button.
2 storage devices selected					Reset All

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

New Red Hat Enterprise Linux 7.5 Installa	ition	rhel-swap	
SYSTEM /boot boot	1953 MiB	Mount Point:	Device(s):
swap thel-swap	1952 MiB >	Desired Capacity: 1952 MiB	ATA Mircon_5100_MTFD (sdh) and 1 other
*		Device Type: RAID Encry File System:	Modify RAID Level: RAID1 (Redundancy)
		swap 🔹 Reformat	Name: swap
		No	Update Settings
+ - C		Na	Update Set de. The settings you make on this screen e applied until you click on the main menu installation

31. Click + to add the / partition. The size can be left empty, so it uses the remaining capacity and click Add Mountpoint.

MANUAL PARTITIONING			RED HAT	ENTERPRISE LINUX 7.5 INSTALLATION
New Red Hat Enterprise Linux 7.9 SYSTEM /boot	5 Installation 1953 MiB	swap Mount Point:	100 us	Devke(s):
boot swap swap	1952 MiB 🗲	Desired Capacity: 1952 MiB		ATA Mircon_5100_MTFD (sdh) and 1 other
	ADD A NEW MOU More customiza after creating th Mount Point: / Desired Capacity:	UNT POINT tion options are available le mount point below.	•] Encrypt	Modify RAID Level: RAID1 (Redundancy)
	c	ancel Add mount poin	t d	Name: swap
+ - C			Note be sp;	Update Settings. The settings you make on this screen will not sled until you click on the main menu's 'Begin
AVAILABLE SPACE 439.51 GIB TOTAL SPACE 447.14 GIB				Prisent Al

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

	RED HAT	ENTERPRISE LINUX 7.5 INSTALLATION
▼ New Red Hat Enterprise Linux 7.5 Installation	rhel-root	
SYSTEM /boot 1953 MiB boot	Mount Point:	Device(s):
/ 439.5 GiB >	Desired Capacity:	ATA Mircon_5100_MTFD (sdh) and 1
swap 1952 MiB	439.5 GB Device Type: RAID Encrypt File System: Sfs Reformat Label:	Modify RAID Level: RAID1 (Redundancy)
+ - C AVAILABLE SPACE 3185 KIB 2 storage devices selected	Note: be app	root Update Settings The settings you make on this screen will not pled until you click on the main menu's 'Begin Installation' button. Reset All

- 33. Click Done to return to the main screen and continue the Installation.
- 34. Click SOFTWARE SELECTION.



35. Select Infrastructure Server and select the Add-Ons as noted below. Click Done.

Base Environment	Add-Ons for Selected Environment
 Minimal Install Basic functionality. Infrastructure Server Server for operating network infrastructure services. File and Print Server File, print, and storage server for enterprises. Basic Web Server Server for serving static and dynamic internet content. Virtualization Host Minimal virtualization host. Server with GUI Server for operating network infrastructure services, with a GUI. 	Java support for the Red Hat Enterprise Linux Server and Desktop Platfor Large Systems Performance Performance support tools for large systems. Load Balancer Load Balancing support for network traffic. MariaDB Database Server The MariaDB SQL database server, and associated packages. Network File System Client Enables the system to attach to network storage. Performance Tools Tools for diagnosing system and application-level performance problems. PostgreSQL Database Server The PostgreSQL Database Server The PostgreSQL SQL database server, and associated packages. Print Server Allows the system to act as a print server. Remote Management for Linux Remote management interface for Red Hat Enterprise Linux, including OpenLMI and SNMP. Virtualization Hypervisor Smallest possible virtualization host installation. Compatibility Libraries Compatibility Libraries Compati

36. Click **NETWORK & HOSTNAME** and configure Hostname and Networking for the Host.

🥮 redhat.	INSTALLATION S	UMMARY	RED	D HAT ENTERPRISE LINUX 7.5 IN	ISTALLATION Helpl
	LOCALIZA	TION			
	Θ	DATE & TIME Americas/Los Angeles timezone		KEYBOARD English (US)	
H/L	á	LANGUAGE SUPPORT English (United States)			
	SOFTWAR	E			
(A = /	0	INSTALLATION SOURCE	6	SOFTWARE SELECTION Infrastructure Server	
	SYSTEM				
	2	INSTALLATION DESTINATION Custom partitioning selected	Q	KDUMP Kdump is enabled	
	÷	NETWORK & HOST NAME Not connected		SECURITY POLICY No profile selected	
		k			
			W	Ouit Po Pe won't touch your disks until you click	gin Installation 'Begin Installation'.

37. Type in the hostname as shown below.

NETWORK & HOST NAME	RED HAT	ENTERPRISE LINUX	7.5 INSTALLATION
Ethernet (enp65s0) Cisco Systems Inc VIC Ethernet NIC (VIC 1385 PCIe Ethernet NIC)	Ethernet (enp65s0) Disconnected		OFF
	Hardware Address 00:25:B5:00:00:13		
	Subnet Mask 127.0.0.1		
			Configure
+ -			
Host name: rhell.hdp3.cisco.com	·	Current host name:	rhel1.hdp3.cisco.com

 $_{\rm 3}8.~$ Click Configure to open the Network Connectivity window. Click IPV4Settings.

TWORK & HOST NAME			RED HAT EN	ITERPRISE LINUX 7.	5 INSTALLATION
Ethernet (enn65s0) Cisco Systems I		Editing enp65s0			OFF
Connection	name: enp65s0				
Genera	l Ethernet 80	2.1X Security DCB P	roxy IPv4 Settings	IPv6 Settings	
Method:	Automatic (DHCP)			•	
Addition	al static addresses	Netmask	Gateway	Add	
			,	Delete	
Addition	al DNS servers:				
Additior	al search domains:				
DHCP c	ient ID:				
- Red	uire IPV4 addressing for	this connection to complete		Routes	
			Can	cel Save	Configure
).hdp3.cisco.co	om	Apply	c	urrent host name: rhe	el20.hdp3.cisco.com

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

Systems	enn65s0)	Ed	iting enp65s0	
	Connection name:	enp65s0		
	General Ethe	ernet 802.1X Security	DCB Proxy IPv4 Settings	IPv6 Settings
	Method: Manual			-
	Addresses			
	Address	Netmask	Gateway	Add
	10.16.1.31	255.255.255.0	10.16.1.1	Delete
	DNS servers:			
	Search domains:			
	DHCP client ID:			
	🗌 Require IPv4 a	ddressing for this connection to	o complete	
				Routes
				Config
			6	

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

NETWORK & HOST NAME		RED HAT	ENTERPRISE LINU	x 7.5 INSTALLATION Help!
Ethernet (enp65s0) Gisco Systems Inc VIC Ethernet NIC (VIC 1385 PCIe Ethernet NIC)	J	Ethernet (enp65s0) Connected		ON
	Hardware Address	00:25:B5:00:00:13		
	Speed	40000 Mb/s		
	IP Address	10.16.1.50		
	Subnet Mask	255.255.255.0		
	Default Route	10.16.1.1		
	DNS			
				Configure
+ -				connigure
Host name: rhel20.hdp3.cisco.com Apply	/		Current host name:	rhel20.hdp3.cisco.com

41. Click Begin Installation on the Main menu.



42. Select Root Password in the User Settings.



43. Enter the Root Password and click Done.

ROOT PASSWORD		RE	D HAT ENTERPRISE LINUX 7.5 INSTALLATION
	The root account is used for admini	stering the system. Enter a password for	the root user.
	Root Password:	•••••	N 5
			Strong
	Confirm:	•••••	

44. When the installation is complete reboot the system.

🧶 redhat	CONFIGURATION			R	ED HAT ENTERPRISE LINUX 7.5	
	USER SE	TTINGS				
	C	ROOT PASSWORD Root password is set		SER CREATION		•
/ # / / //						
	Completel		Red Hat E	nterprise Linux is now successfully Go	installed and ready for you to use ahead and reboot to start using it!	
					Rihoot	
	A Une of this product is subject to	the license agreement found at /usr/share/redbath elease/EULA				

45. Repeat the steps to install Red Hat Enterprise Linux 7.5 on the remaining servers.

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

The hostnames and their corresponding IP addresses are shown in Table 7.

Tuble / Tiostilumes u	
Hostname	etho
rhelı	10.16.1.31
rhel2	10.16.1.32
rhel3	10.16.1.33
rhel4	10.16.1.34
rhelı	10.16.1.35
rhel6	10.16.1.36

Table 7	Hostnames and IP Addresses
---------	----------------------------

么

Hostname	etho
rhel7	10.16.1.37
rhel8	10.16.1.38
rhelg	10.16.1.39
rhel10	10.16.1.40
rhel11	10.16.1.41
rhel12	10.16.1.42
rhel13	10.16.1.43
rhel14	10.16.1.44
rhel15	10.16.1.45
rhel16	10.16.1.46
rhel24	10.16.1.54



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

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

Post OS Install Configuration

Choose one of the nodes of the cluster or a separate node as the Admin Node for management such as HDP installation, Ansible, creating a local Red Hat repo and so on. In this document, rhel1 has been used for this purpose.

Configure /etc/hosts

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

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

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

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

1. Log into the Admin Node (rhel1).

#ssh 10.16.1.31

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

```
# cat /etc/hosts
127.0.0.1
          localhost localhost.localdomain localhost4 localhost4.localdomain4
           localhost localhost.localdomain localhost6 localhost6.localdomain6
::1
10.16.1.31 rhel1 rhel1.hdp3.cisco.com
10.16.1.32 rhel2 rhel2.hdp3.cisco.com
10.16.1.33 rhel3 rhel3.hdp3.cisco.com
10.16.1.34 rhel4 rhel4.hdp3.cisco.com
10.16.1.35 rhel5 rhel5.hdp3.cisco.com
10.16.1.36 rhel6 rhel6.hdp3.cisco.com
10.16.1.37 rhel7 rhel7.hdp3.cisco.com
10.16.1.38 rhel8 rhel8.hdp3.cisco.com
10.16.1.39 rhel9 rhel9.hdp3.cisco.com
10.16.1.40 rhel10 rhel10.hdp3.cisco.com
10.16.1.41 rhel11 rhel11.hdp3.cisco.com
10.16.1.42 rhel12 rhel12.hdp3.cisco.com
10.16.1.43 rhel13 rhel13.hdp3.cisco.com
10.16.1.44 rhel14 rhel14.hdp3.cisco.com
10.16.1.45 rhel15 rhel15.hdp3.cisco.com
10.16.1.46 rhel16 rhel16.hdp3.cisco.com
10.16.1.47 rhel17 rhel17.hdp3.cisco.com
```

Set Up the Passwordless Login

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

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

1. Log into the Admin Node (rhel1).

#ssh 10.13.1.31

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

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

```
ssh-keygen
Figure 25
[root@rhel1 ansible] # ssh-keygen -N '' -f ~/.ssh/id rsa
Generating public/private rsa key pair.
/root/.ssh/id rsa already exists.
Overwrite (y/n)? y
Your identification has been saved in /root/.ssh/id rsa.
Your public key has been saved in /root/.ssh/id rsa.pub.
The key fingerprint is:
SHA256:j+IdDaaxUBH2ciy/c4M0YcDPhgOoRWrsb8NGnaaj28s root@rhel1
The key's randomart image is:
   -[RSA 2048]----+
     . . . =
  ...
      ..+*+.
    ..+.os
     ο.
         * 0
        + * =
    0
     0.0+.
  o.E.
      [SHA256]-
```

3. Run the following command from the admin node to copy the public key id_rsa.pub to all the nodes of the cluster. ssh-copy-id appends the keys to the remote-host's .ssh/authorized keys.

for i in {1..17}; do echo "copying rhel\$i.hdp3.cisco.com"; ssh-copy-id -i ~/.ssh/id_rsa.pub root@rhel\$i.hdp3.cisco.com; done;

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

Create the Red Hat Enterprise Linux (RHEL) 7.5 Local Repository

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

1. Log into rhel1. Create a directory that would contain the repository.

mkdir -p /var/www/html/rhelrepo

- 2. Copy the contents of the Red Hat DVD to /var/www/html/rhelrepo
- 3. Alternatively, if you have access to a Red Hat ISO Image, Copy the ISO file to rhel1.
- 4. Log back into rhel1 and create the mount directory.

```
# scp rhel-server-7.5-x86_64-dvd.iso rhel1:/root/
# mkdir -p /mnt/rheliso
# mount -t iso9660 -o loop /root/rhel-server-7.5-x86 64-dvd.iso /mnt/rheliso/
```

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

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

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

```
# vi /var/www/html/rhelrepo/rheliso.repo
[rhe17.5]
name=Red Hat Enterprise Linux 7.5
baseurl=http://10.16.1.31/rhelrepo
gpgcheck=0
enabled=1
```

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

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



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

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



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

```
# vi /etc/yum.repos.d/rheliso.repo
[rhel7.5]
name=Red Hat Enterprise Linux 7.5
baseurl=file:///var/www/html/rhelrepo
gpgcheck=0
enabled=1
```

Create the Red Hat Repository Database

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

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

yum -y install createrepo

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

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

```
Figure 26 createrepo
```

```
[root@rhel1 rhelrepo]# createrepo .
Spawning worker 0 with 3763 pkgs
Workers Finished
Gathering worker results
Saving Primary metadata
Saving file lists metadata
Saving other metadata
Generating sqlite DBs
Sqlite DBs complete
```

Set Up Ansible

To set up Ansible, follow these steps:

1. Download Ansible rpm from the following link:

https://releases.ansible.com/ansible/rpm/release/epel-7-x86_64/ansible-2.4.6.0-1.el7.ans.noarch.rpm

```
# wget https://releases.ansible.com/ansible/rpm/release/epel-7-x86 64/ansible-2.4.6.0-
1.el7.ans.noarch.rpm
```



For more information about to download and install Ansible engine, please follow the link <u>https://access.redhat.com/articles/3174981</u>

2. Run the following command to install ansible.

```
# yum localinstall -y ansible-2.4.6.0-1.el7.ans.noarch.rpm
```

3. Verify Ansible installation by running the following commands.

```
# ansible -version
# ansible localhost -m ping
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost
does not match 'all'
localhost | SUCCESS => {
    "changed": false,
    "failed": false,
    "ping": "pong"
}
```

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

```
[root@rhel1 ~]# cat /etc/ansible/hosts
[admin]
rhel1.hdp3.cisco.com
[namenodes]
rhell.hdp3.cisco.com
rhel2.hdp3.cisco.com
rhel3.hdp3.cisco.com
[datanodes]
rhel4.hdp3.cisco.com
rhel5.hdp3.cisco.com
rhel6.hdp3.cisco.com
rhel7.hdp3.cisco.com
rhel8.hdp3.cisco.com
rhel9.hdp3.cisco.com
rhel10.hdp3.cisco.com
rhel11.hdp3.cisco.com
rhel12.hdp3.cisco.com
rhel13.hdp3.cisco.com
rhel14.hdp3.cisco.com
rhel15.hdp3.cisco.com
rhel16.hdp3.cisco.com
rhel17.hdp3.cisco.com
rhel18.hdp3.cisco.com
rhel19.hdp3.cisco.com
rhel20.hdp3.cisco.com
```

rhel21.hdp3.cisco.com rhel22.hdp3.cisco.com rhel23.hdp3.cisco.com rhel24.hdp3.cisco.com rhel25.hdp3.cisco.com rhel26.hdp3.cisco.com rhel27.hdp3.cisco.com rhel28.hdp3.cisco.com [nodeswithgpu] rhel25.hdp3.cisco.com rhel26.hdp3.cisco.com rhel27.hdp3.cisco.com rhel28.hdp3.cisco.com [nodes] rhell.hdp3.cisco.com rhel2.hdp3.cisco.com rhel3.hdp3.cisco.com rhel4.hdp3.cisco.com rhel5.hdp3.cisco.com rhel6.hdp3.cisco.com rhel7.hdp3.cisco.com rhel8.hdp3.cisco.com rhel9.hdp3.cisco.com rhel10.hdp3.cisco.com rhel11.hdp3.cisco.com rhel12.hdp3.cisco.com rhel13.hdp3.cisco.com rhel14.hdp3.cisco.com rhel15.hdp3.cisco.com rhel16.hdp3.cisco.com rhel17.hdp3.cisco.com rhel18.hdp3.cisco.com rhel19.hdp3.cisco.com rhel20.hdp3.cisco.com rhel21.hdp3.cisco.com rhel22.hdp3.cisco.com rhel23.hdp3.cisco.com rhel24.hdp3.cisco.com rhel25.hdp3.cisco.com rhel26.hdp3.cisco.com rhel27.hdp3.cisco.com rhel28.hdp3.cisco.com

5. Verify host group by running the following commands. Figure 27 shows the outcome of the ping command.

ansible nodeswithgpu -m ping

Figure 27 Ansible – Ping Hosts

```
[root@rhel1 ~]# ansible nodeswithgpu -m ping
rhel17.hdp3.cisco.com | SUCCESS => {
    "changed": false,
    "failed": false,
    "ping": "pong"
}
rhel15.hdp3.cisco.com | SUCCESS => {
    "changed": false,
    "failed": false,
    "ping": "pong"
}
rhel16.hdp3.cisco.com | SUCCESS => {
    "changed": false,
    "failed": false,
    "ping": "pong"
}
rhel14.hdp3.cisco.com | SUCCESS => {
    "changed": false,
    "failed": false,
    "failed": false,
    "failed": false,
    "failed": false,
    "failed": false,
    "ping": "pong"
}
```

Install httpd

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

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



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

yum -y install httpd

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

```
# vi /etc/httpd/conf/httpd.conf
ServerName 10.16.1.31:80
```

3. Start httpd:

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

Set Up All Nodes to Use the RHEL Repository

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



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

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

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

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

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

3. Purge the yum caches:

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



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

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

Upgrade the Cisco Network Driver for VIC1387

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

https://software.cisco.com/download/home/283862063/type/283853158/release/4.0.2

In the ISO image, the required driver kmod-enic-.....rpm can be located at Network\Cisco\VIC\RHEL\RHEL7.5.

- 1. From a node connected to the Internet, download, extract and transfer kmod-enic-.rpm to rhel1 (admin node).
- Copy the rpm on all nodes of the cluster using the following Ansible commands. For this example, the rpm is assumed to be in present working directory of rhel1:

```
[root@rhel1 ~]# ansible all -m copy -a "src=/root/kmod-enic-3.1.137.5-700.16.rhel7u5.x86_64.rpm
dest=/root/."
```

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

```
[root@rhel1 ~]# ansible all -m yum -a "name=/root/kmod-enic-3.1.137.5-700.16.rhel7u5.x86_64.rpm
state=present"
```

 Make sure that the above installed version of kmod-enic driver is being used on all nodes by running the command "modinfo enic" on all nodes:

[root@rhel1 ~]# ansible all -m command -a "modinfo enic"

5. It is recommended to download the kmod-megaraid driver for higher performance. The RPM can be found in the same package at: \Linux\Storage\LSI\Cisco Storage 12G SAS RAID controller\RHEL\RHEL7.5

Install xfsprogs

From the admin node rhel1 run the command shown below to Install xfsprogs on all the nodes for xfs filesystem:

ansible all -m yum -a "name=xfsprogs state=present"

Set Up JAVA

To setup JAVA, follow these steps:



HDP 3.01 requires JAVA 8.

- Download jdk-7u75-linux-x64.rpm and scp the rpm to admin node (rhel1) from the link (<u>http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html</u>)
- 2. Copy JDK rpm to all nodes:

ansible nodes -m copy -a "src=/root/jdk-8u181-linux-x64.rpm dest=/root/."

3. Extract and Install JDK on all nodes:

ansible all -m command -a "rpm -ivh jdk-8u181-linux-x64.rpm"

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

```
vi java-set-alternatives.sh
#!/bin/bash
for item in java javac javaws jar jps javah javap jcontrol jconsole jdb; do
rm -f /var/lib/alternatives/$item
alternatives --install /usr/bin/$item $item /usr/java/jdk1.8.0_181-amd64/bin/$item 9
alternatives --set $item /usr/java/jdk1.8.0_181-amd64/bin/$item
done
vi java-home.sh
export JAVA_HOME=/usr/java/jdk1.8.0_181-amd64
```

5. Make the two java scripts created above executable:

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

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

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

7. Setup Java Alternatives

[root@rhel1 ~]# ansible all -m shell -a "/root/java-set-alternatives.sh"

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

ansible all -m shell -a "alternatives --display java | head -2"

9. Setup JAVA_HOME on all nodes.

ansible all -m shell -a "source /root/java-home.sh"

10. Display JAVA_HOME on all nodes.

ansible all -m command -a "echo \$JAVA_HOME"

11. Display current java -version.

ansible all -m command -a "java -version"

```
[root@rhel1 ~]# ansible all -m command -a "java -version"
rhel3.hdp3.cisco.com | SUCCESS | rc=0 >>
java version "1.8.0_181"
Java(TM) SE Runtime Environment (build 1.8.0_181-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.181-b13, mixed mode)
```

Configure NTP

The Network Time Protocol (NTP) is used to synchronize the time of all the nodes within the cluster. The Network Time Protocol daemon (ntpd) sets and maintains the system time of day in synchronism with the timeserver located in the admin node (rhel1). Configuring NTP is critical for any Hadoop Cluster. If server clocks in the cluster drift out of sync, serious problems will occur with HBase and other services.

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

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

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

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

```
includefile /etc/ntp/crypto/pw
keys /etc/ntp/keys
```

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

```
# vi /root/ntp.conf
server 10.16.1.31
driftfile /var/lib/ntp/drift
restrict 127.0.0.1
restrict -6 ::1
includefile /etc/ntp/crypto/pw
keys /etc/ntp/keys
```

3. Copy ntp.conf file from the admin node to /etc of all the nodes by executing the following commands in the admin node (rhel1):

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

Run the following to syncronize the time and restart NTP daemon on all nodes:

```
# ansible all -m service -a "name=ntpd state=stopped"
# ansible all -m command -a "ntpdate rhell.hdp3.cisco.com"
# ansible all -m service -a "name=ntpd state=started"
```

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

ansible all -a "systemctl enable ntpd"

- 6. Verify NTP is up and running in all nodes by running the following commands:
- # ansible all -a "systemctl status ntpd"

```
[root@rhel1 ~]# ansible all -m command -a "systemctl status ntpd"
rhel5.hdp3.cisco.com | SUCCESS | rc=0 >>
• ntpd.service - Network Time Service
Loaded: loaded (/usr/lib/systemd/system/ntpd.service; enabled; vendor preset: disabled)
Active: active (running) since Tue 2018-10-23 10:50:25 PDT; 1 months 2 days ago
Main PID: 1401 (ntpd)
Tasks: 1
Memory: 4.0K
CGroup: /system.slice/ntpd.service
_____1401 /usr/sbin/ntpd -u ntp:ntp -q
```

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

7. Install the Chrony service:

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

8. Activate the Chrony service at boot:

ansible all -a "systemctl enable chronyd"

9. Start the Chrony service:

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

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

Enable Syslog

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

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

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

Set ulimit

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

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

To set ulimit, follow these steps:

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

root soft nofile 64000 root hard nofile 64000

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

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

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

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

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



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

Disable SELinux

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

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

To disable SELinux, follow these steps:

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

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

ansible nodes -m shell -a "setenforce 0"

- The command (above) may fail if SELinux is already disabled. This require reboot to take effect.
- Reboot the machine, if needed for SELinux to be disabled in case it does not take effect. It can be checked using the fol-2. lowing command:

ansible nodes -a "sestatus"



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).

To set TCP retries, follow these steps:

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

Edit the file /etc/sysctl.conf and on admin node rhel1 and add the following lines: 1.

net.ipv4.tcp retries2=5

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

ansible nodes -m copy -a "src=/etc/sysctl.conf dest=/etc/sysctl.conf"

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

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

Disable the Linux Firewall

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

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

Disable Swapping

To disable swapping, follow these steps:

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

ansible all -m shell -a "echo 'vm.swappiness=1' >> /etc/sysctl.conf"

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

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



Disable Transparent Huge Pages

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

To disable Transparent Huge Pages, follow these steps:

1. The following commands must be run for every reboot, so copy this command to /etc/rc.local so they are executed automatically for every reboot:

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

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

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

3. Copy file to each node:

ansible nodes -m copy -a "src=/root/thp disable dest=/root/thp disable"

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

ansible nodes -m shell -a "cat /root/thp disable >> /etc/rc.local"

Disable IPv6 Defaults

To disable IPv6 defaults, follow these steps:

1. Disable IPv6 as the addresses used are IPv4:

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

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

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

Configure Data Drives on Name Node and Other Management Nodes

This section describes the steps to configure non-OS disk drives as RAID1 using the StorCli command. All drives are part of a single RAID1 volume. This volume can be used for staging any client data to be loaded to HDFS. This volume will not be used for HDFS data.

To configure data drives on Name node and other nodes, follow these steps:

- 1. From the website download storcli <u>https://www.broadcom.com/support/download-search/?pg=&pf=&pn=&po=&pa=&dk=storcli.</u>
- 2. Extract the zip file and copy storcli-1.14.12-1.noarch.rpm from the linux directory.
- 3. Download storcli and its dependencies and transfer to Admin node:

#scp storcli-1.14.12-1.noarch.rpm rhel1:/root/

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

ansible all -m copy -a "src=/root/storcli-1.14.12-1.noarch.rpm dest=/root/."

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

ansible all -m shell -a "rpm -ivh storcli-1.14.12-1.noarch.rpm"

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

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

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

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



If the drive state shows up as JBOD, creating RAID in the subsequent steps will fail with the error "*The specified physical disk does not have the appropriate attributes to complete the requested command.*"

- 8. If the drive state shows up as JBOD, it can be converted into Unconfigured Good using Cisco UCSM or storcli64 command. Following steps should be performed if the state is JBOD.
- 9. Get the enclosure id as follows:

```
ansible all -m shell -a "./storcli64 pdlist -a0 | grep Enc | grep -v 252 | awk '{print $4}' | sort | uniq
-c | awk '{print $2}'"
[root@rhel1 ~]# ansible all -m shell -a "./storcli64 pdlist -a0 | grep Enc | grep -v 252 | awk '{print $4}' | sort | uniq -c | awk '{print $2}'"
rhel1.hdp3.cisco.com | SUCCESS | rc=0 >>
    8 Enclosure Device ID: 66
    8 Enclosure position: 0
rhel2.hdp3.cisco.com | SUCCESS | rc=0 >>
    8 Enclosure Device ID: 66
    8 Enclosure Device ID: 66
    8 Enclosure position: 0
```

么

It is observed that some earlier versions of storcli64 complains about above mentioned command as if it is deprecated. In this case, please ./storcli64 /co show all| awk '{print \$1}'| sed -n '/[0-9]:[0-9]/p'|awk '{print sub-str(\$1,1,2)}'|sort -u command to determine enclosure id.

10. Convert to unconfigured good:

```
ansible datanodes -m command -a "./storcli64 /c0 /e66 /sall set good force"
[root@rhel1 ~]# ansible datanodes -m command -a "./storcli64 /c0 /e66 /sall set good force"
rhel8.hdp3.cisco.com | SUCCESS | rc=0 >>
Controller = 0
Status = Success
Description = Set Drive Good Succeeded.
rhel6.hdp3.cisco.com | SUCCESS | rc=0 >>
Controller = 0
Status = Success
Description = Set Drive Good Succeeded.
```

11. Verify status by running the following command:

ansible datanodes -m command -a "./storcli64 /c0 /e66 /sall show"

12. Run this script as root user on rhel1 to rhel3 to create the virtual drives for the management nodes:

```
#vi /root/raid1.sh
./storcli64 -cfgldadd
r1[$1:1,$1:2,$1:3,$1:4,$1:5,$1:6,$1:7,$1:8,$1:9,$1:10,$1:11,$1:12,$1:13,$1:14,$1:15,$1:16,$1:17,$1:18,$1:
19,$1:20,$1:21,$1:22,$1:23,$1:24] wb ra nocachedbadbbu strpsz1024 -a0
```



The script (above) requires enclosure ID as a parameter.

13. Run the following command to get enclosure id:

```
#./storcli64 pdlist -a0 | grep Enc | grep -v 252 | awk '{print $4}' | sort | uniq -c | awk '{print $2}'
#chmod 755 raidl.sh
```

14. Run MegaCli script:

```
#./raid1.sh <EnclosureID> obtained by running the command above
WB: Write back
RA: Read Ahead
NoCachedBadBBU: Do not write cache when the BBU is bad.
Strpsz1024: Strip Size of 1024K
```



The command (above) will not override any existing configuration. To clear and reconfigure existing configurations refer to Embedded MegaRAID Software Users Guide available: <u>www.broadcom.com.</u>

15. Run the following command. State should change to Online:

ansible namenodes -m command -a "./storcli64 /c0 /e66 /sall show"

[root@r] rhel2.hd Control] Status = Descript	hel1 dp3.d ler = = Sud tion	~]# an cisco.(= 0 ccess = Show	nsik com W Di	ole nam SUCC cive I:	meno CES: nfo:	odes – 5 ro rmatio	-m co c=0) on Si	ommai >>	nd - edeo	-a	" - /	/storcli64 /	c0 /€	∍66 ,	/sall	show"
Drive I	nforr	nation														
EID:Slt	DID	State	DG		ize	Intf		SED	PI			Model				
66.1	43	 Onln		1 635	 тв	SAS	 аан				KB	ST1800MM012				
66:2	42	Onln		1.635	TB	SAS	HDD				KB	ST1800MM012				
66:3	45	Onln		1.635	TB	SAS	HDD				KB	ST1800MM012				
66:4	44	Onln		1.635	TΒ	SAS	HDD				KB	ST1800MM012				
66:5	41	Onln		1.635	TB	SAS	HDD				KB	ST1800MM012				
66:6		Onln		1.635	$^{\mathrm{TB}}$	SAS	HDD				KB	ST1800MM012				
66:7	40	Onln		1.635	TB	SAS	HDD				KB	ST1800MM012				
66:8	39	Onln		1.635	TB	SAS	HDD				KB	ST1800MM012				

16. State can also be verified in UCSM as show below in Equipment>Rack-Mounts>Servers>Server # under Inventory/Storage/Disk tab:

Control IIIVC	entory Virtual	Machines	Hybrid Display	Installed Firm	ware SEL	Logs CIM	C Sessions	VIF Paths	Power Contro
Votherboard C	IMC CPUs	GPUs 1	Memory Ad	apters HBAs	NICs	iSCSI vNICs	Storage		
Controller LUNs	a Disks	SAS Expander	Security						
+ — Ty Advance	ed Filter 🔶 Exp	ort 📥 Print							
Name	Size (M	B)	Serial		Operability		Drive State		Presence
Storage Controlle	er PC								
	er SA								
✓ Storage Controlle Disk 1	ar SA 171565	5	WBN05W	/HC0000E8236	Operable		Online		Equipped
Storage Controlle Disk 1 Disk 2	er SA 171565 171565	5	WBN05W	/HC0000E8236	Operable Operable		Online Online		Equipped Equipped
Storage Controlle Disk 1 Disk 2 Disk 3	er SA 171565 171565 171565	5 5	WBN05W WBN06Q WBN047	/HC0000E8236 TS0000E826M DN0000E8280	Operable Operable Operable		Online Online Online		Equipped Equipped Equipped

Configure Data Drives on Data Nodes

To configure non-OS disk drives as individual RAIDo volumes using StorCli command, follow these steps. These volumes will be used for HDFS Data.

1. Issue the following command from the admin node to create the virtual drives with individual RAID o configurations on all the data nodes:

```
[root@rhel1 ~]# ansible datanodes -m command -a "./storcli64 -cfgeachdskraid0 WB RA direct NoCachedBadBBU
strpsz1024 -a0"
rhel7.hdp3.cisco.com | SUCCESS | rc=0 >>
Adapter 0: Created VD 0
Configured physical device at Encl-66:Slot-7.
Adapter 0: Created VD 1
Configured physical device at Encl-66:Slot-6.
Adapter 0: Created VD 2
Configured physical device at Encl-66:Slot-8.
Adapter 0: Created VD 3
Configured physical device at Encl-66:Slot-5.
Adapter 0: Created VD 4
Configured physical device at Encl-66:Slot-3.
Adapter 0: Created VD 5
Configured physical device at Encl-66:Slot-4.
```

```
Adapter 0: Created VD 6
Configured physical device at Encl-66:Slot-1.
Adapter 0: Created VD 7
Configured physical device at Encl-66:Slot-2.
..... Omitted Ouput
24 physical devices are Configured on adapter 0.
Exit Code: 0x00
```

A

The command (above) will not override existing configurations. To clear and reconfigure existing configurations, refer to the Embedded MegaRAID Software Users Guide available at <u>www.broadcom.com</u>.

Configure the Filesystem for NameNodes and Datanodes

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

1. From the Admin node, 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:



The script assumes there are no partitions already existing on the data volumes. If there are partitions, delete them before running the script. This process is documented in section Delete Partitions.

```
#vi /root/driveconf.sh
#!/bin/bash
[[ "-x" == "${1}" ]] && set -x && set -v && shift 1
count=1
for X in /sys/class/scsi host/host?/scan
do
echo '- - -' >  {X}
done
for X in /dev/sd?
do
list+=$(echo $X " ")
done
for X in /dev/sd??
do
list+=$(echo $X " ")
done
for X in $list
do
echo "======"
echo $X
echo "======"
if [[ -b ${X} && `/sbin/parted -s ${X} print quit|/bin/grep -c boot` -
ne O
11
then
echo "$X bootable - skipping."
continue
else
Y=${X##*/}1
echo "Formatting and Mounting Drive => ${X}"
166
/sbin/mkfs.xfs -f ${X}
(( $? )) && continue
#Identify UUID
UUID=`blkid ${X} | cut -d " " -f2 | cut -d "=" -f2 | sed 's/"//q'`
/bin/mkdir -p /data/disk${count}
(( $? )) && continue
echo "UUID of {X} = {UUID}, mounting {X} using UUID on
/data/disk${count}"
```

```
/bin/mount -t xfs -o inode64,noatime,nobarrier -U ${UUID}
/data/disk${count}
(( $? )) && continue
echo "UUID=${UUID} /data/disk${count} xfs inode64,noatime,nobarrier 0
0" >> /etc/fstab
((count++))
fi
done
```

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

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

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

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

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

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

Delete Partitions

To delete a partition, follow these steps:

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



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

Cluster Verification

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

To verify a cluster, follow these steps:

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

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

```
Solution Design
```

```
red='\e[0;31m'
NC='\e[Om' # No Color
echo -e "${green} === Cisco UCS Integrated Infrastructure for Big Data and Analytics \ Cluster
Verification === ${NC}"
echo ""
echo ""
echo -e "${green} ==== System Information ==== ${NC}"
echo ""
echo ""
echo -e "${green}System ${NC}"
ansible all -m shell -a "dmidecode |grep -A2 'System Information'"
echo ""
echo ""
echo -e "${green}BIOS ${NC}"
ansible all -m shell -a "dmidecode | grep -A3 'BIOS Information'"
echo ""
echo ""
echo -e "${green}Memory ${NC}"
ansible all -m shell -a "cat /proc/meminfo | grep -i ^memt | unig"
echo ""
echo ""
echo -e "${green}Number of Dimms ${NC}"
ansible all -m shell -a "echo 'DIMM Slots:'; dmidecode |qrep -c \ '^[[:space:]]*Locator:'"
ansible all -m shell -a "echo 'DIMM Count is:'; dmidecode |grep -c \ '^[[:space:]]*Locator:'"
ansible all -m shell -a "dmidecode | awk '/Memory Device$/,/^$/ {print}' | grep -e '^Mem' -e Size: -e
Speed: -e Part | sort -u | grep -v -e 'NO \ DIMM' -e 'No Module Installed' -e Unknown"
echo ""
echo ""
# probe for cpu info #
echo -e "${green}CPU ${NC}"
ansible all -m shell -a "grep '^model name' /proc/cpuinfo | sort -u"
echo ""
ansible all -m shell -a "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 ${NC}"
ansible all -m shell -a "ifconfig | egrep '(^{p})' | awk '{print \$ | \ xargs -l `which ethtool` |
grep -e ^Settings -e Speed"
echo ""
ansible all -m shell -a "lspci | grep -i ether"
echo ""
echo ""
# probe for disk info #
echo -e "${green}Storage ${NC}"
ansible all -m shell -a "echo 'Storage Controller: '; `which lspci` | grep -i -e \ raid -e storage -e
lsi"
echo ""
ansible all -m shell -a "dmesg | grep -i raid | grep -i scsi"
echo ""
ansible all -m shell -a "lsblk -id | awk '{print \$1,\$4}'|sort | nl"
echo ""
echo ""
echo -e "${green} ======== Software ========== ${NC}"
echo ""
echo ""
echo -e "${green}Linux Release ${NC}"
ansible all -m shell -a "cat /etc/*release | uniq"
echo ""
echo ""
echo -e "${green}Linux Version ${NC}"
ansible all -m shell -a "uname -srvm | fmt"
echo ""
echo ""
echo -e "${green}Date ${NC}"
ansible all -m shell -a "date"
echo ""
echo ""
echo -e "${green}NTP Status ${NC}"
ansible all -m shell -a "ntpstat 2>&1 | head -1"
echo ""
```
```
echo ""
echo -e "${green}SELINUX ${NC}"
ansible all -m shell -a "echo -n 'SElinux status: '; grep ^SELINUX= \ /etc/selinux/config 2>&1"
echo ""
echo ""
ansible all -m shell -a "echo 'CPUspeed Service: '; service cpuspeed status 2>&1"
ansible all -m shell -a "echo 'CPUspeed Service: '; chkconfig --list cpuspeed 2>&1"
echo ""
echo ""
echo -e "${green}Java Version${NC}"
ansible all -m shell -a "java -version 2>&1; echo JAVA_HOME is ${JAVA_HOME}"
echo ""
echo ""
echo -e "${green}Hostname LoOKup${NC}"
ansible all -m shell -a "ip addr show"
echo ""
echo ""
echo -e "${green}Open File Limit${NC}"
ansible all -m shell -a "echo 'Open file limit(should be >32K): '; ulimit -n"
```

2. Change permissions to executable:

chmod 755 cluster_verification.sh

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

#./cluster_verification.sh

Install HDP 3.0.1

HDP is an enterprise grade, hardened Hadoop distribution. HDP combines Apache Hadoop and its related projects into a single tested and certified package. HPD 3.0.1 components are depicted in below. This section details how to install HDP 3.0.1 on the cluster.

												Or	ngo	oin	g	Inr	10	va	tic	n	in	A	pa	ch	e																Add Sk	on u
HDP 3.1 Q4 2018	3.1.1		4.3.1	0.	.16.0	3.:	1.0	0.12.:		0.9.1	1.16.	0	.4.7	2.:	3.2	0.8.0		2.0.2	5.	0.0	1.7	.0	1.0.0		1.2.0	1	.1.0	1.	2.1	2.	0	2.7.	3	3.4.6		9	6	-(9	-0	7.4	į[4]
HDP 3.0.0 Q3 2018	3.1.0	H	4.3.1	0.	.16.0	3.(0.0	0.12.	0	0.9.1	1.16.	0	L.4.7	2	.3	0.8.	•	2.0.0	5.	0.0	1.7	.0	1.0.0		1.1.0	1	.0.0	1.	2.1	1.0).1	2.7.	0	3.4.6](9	6	H	9	-0	7.4	j[4]
HDP 2.6.5 Q2 2018	2.7.3		4.2.0]-[0.	.16.0	1.2	1+ 1 ^[3]	0.10.:	ı]-[(0.7.0	1.2.0 1.10 [[]	न्त ग - :	L.4.6	1.6	.3+ .3	0.7.3	3-[1.1.2	-[4.]	7.0	- 1.7	.0	0.12.	₀}-[0.7.0	}[0	.8.0	- 1.	1.0	1.0	.0	2.6.	2	3.4.6	-[1.	.5.2	0.10.	0-0-0).90	-0.92.0	6.6.	2 ^[4]
HDP 2.6.4 ^[1]	2.7.3		4.2.0]-[0	.16.0	- <mark>1.2</mark> 2.:	1+ 1 ^[3]	0.10.:	ŀ]-[0.7.0	1.2.0	ण ७१ - ि	L.4.6	1.6	.3+ [5]	0.7.3	3	1.1.2	-[4.	7.0	- 1.7	.0	0.12.	0-[0.7.0	}[0	.8.0	- 1.	1.0	0.10	0.1	2.6.	1	3.4.6	-[1	.5.2	-0.10.	0-0	0.90	-0.92.0	5.5.	1 ^[4]
HDP 2.5 Aug 2016	2.7.3	(4.2.0]_[0.	.16.0	1.2 2.1	1+ 1 ^[3]		_[0.7.0	1.2.0 1.6 ^{[7}	ויי 1-[:	L.4.6	1.6 2.(.2+) ^[2]	0.6.0	2	1.1.2	-4.	7.0	1.7	.0	0.9.0)-[0.6.0	}-[•	0.7.0	- 1.	0.1	0.10	0.0	2.4.	• -[3.4.6	-[1	.5.2	0.10.	0-0	0.90	-0.91.0	5.5	.1
	Hadoop & YARN		Oozie		Pig	Line	ыле	Druid		Tez	Calcite		Sqoop	Curat	opark	Zeppeli		HBase		Phoenix	1. minor		Knox		Ranger		Atlas		Storm	Kafka		Ambari		Zookeep		Flume	Falcon		Mahout	Slider	Solr	100
	HD	P Core		I		E	Interp	orise D	ata \	Wareh	ouse				Da Scie	ta nce	I	Opera	ationa	l Dat	ta Sto	re	Sec	urity	Gove	rnar	nce	Ľ	Stro Proc	eam essing	g	Op	erat	tions	R	emov	/ed/M	oved	Comp	oonents	HD Sea	P rch
													н	ort	٥n	vor	ks	Da	ata	Pla	atfo	orn	n																			
[1] HDP 2.6 - [2] Spark 1.6 [3] Hive 2.1 i	- Shows (.3+ Spark	curren 2.1 -	t Apac HDP	:he b 2.6 s	ranche uppor	es be ts bo	eing u oth Sp	sed. F ark 1.	inal (6.3 a	compo nd Sp	nent v ark 2.1	ersio as (n subj GA.	iect to	char	ige ba	ised	on Ap	bache	relea	ase pr	oces	s.																			
[4] Apache S [5] Spark 2.2	olr is ava is GA	ilable	r 2.0. as an	add-	on pro	duct	HDP	Searc	h.																																	

Prerequisites for HDP Installation

This section details the prerequisites for the HDP installation, such as setting up HDP repositories.

Hortonworks Repository

From a host connected to the Internet, create a Hortonworks folder and download the Hortonworks repositories as 1. shown below, then transfer it to the admin node.



If the admin node is connected to the internet via outbound NAT, repositories can be downloaded directly into the admin node.

mkdir -p /tmp/Hortonworks/ # cd /tmp/Hortonworks

Download Hortonworks HDP repo: 2.

```
# wget http://public-repo-1.hortonworks.com/HDP/centos7/3.x/updates/3.0.1.0/HDP-3.0.1.0-centos7-
rpm.tar.gz
--2018-10-13 11:02:02-- http://public-repo-1.hortonworks.com/HDP/centos7/3.x/updates/3.0.1.0/HDP-
3.0.1.0-centos7-rpm.tar.gz
Resolving public-repo-1.hortonworks.com (public-repo-1.hortonworks.com) ... 13.35.121.86, 13.35.121.14,
13.35.121.127, ..
Connecting to public-repo-1.hortonworks.com (public-repo-1.hortonworks.com) |13.35.121.86|:80...
connected.
HTTP request sent, awaiting response... 200 OK
ength: 8964079720 (8.3G) [application/x-tar]
Saving to: 'HDP-3.0.1.0-centos7-rpm.tar.gz'
in 2m 42s
2018-10-13 11:04:44 (52.9 MB/s) - 'HDP-3.0.1.0-centos7-rpm.tar.gz' saved [8964079720/8964079720]
```

Download Hortonworks HDP-Utils repo: 3.

4. Download HDP-GPL repo:

```
# wget http://public-repo-1.hortonworks.com/HDP-GPL/centos7/3.x/updates/3.0.1.0/HDP-GPL-3.0.1.0-centos7-
gpl.tar.gz
--2018-10-13 12:10:45-- http://public-repo-1.hortonworks.com/HDP-GPL/centos7/3.x/updates/3.0.1.0/HDP-
GPL-3.0.1.0-centos7-gpl.tar.gz
Resolving public-repo-1.hortonworks.com (public-repo-1.hortonworks.com)... 13.35.121.120, 13.35.121.127,
13.35.121.86, ...
Connecting to public-repo-1.hortonworks.com (public-repo-1.hortonworks.com)|13.35.121.120|:80...
connected.
HTTP request sent, awaiting response... 200 OK
Length: 162054 (158K) [application/x-tar]
Saving to: 'HDP-GPL-3.0.1.0-centos7-gpl.tar.gz'
100%[=========>] 162,054 --.-K/s in 0.1s
2018-10-13 12:10:45 (1.27 MB/s) - 'HDP-GPL-3.0.1.0-centos7-gpl.tar.gz' saved [162054/162054]
```

5. Download the Ambari repo:

```
# wget http://public-repo-1.hortonworks.com/ambari/centos7/2.x/updates/2.7.1.0/ambari-2.7.1.0-
centos7.tar.gz
```

6. Copy the repository directory to the admin node (rhel1):

scp -r /tmp/Hortonworks/ rhel1:/var/www/html/

7. Extract the tar ball:

```
[root@rhell Hortonworks]# tar -zxvf HDP-3.0.1.0-centos7-rpm.tar.gz
[root@rhell Hortonworks]# tar -zxvf HDP-UTILS-1.1.0.22-centos7.tar.gz
[root@rhell Hortonworks]# tar -zxvf HDP-GPL-3.0.1.0-centos7-gpl.tar.gz
[root@rhell Hortonworks]# tar -zxvf ambari-2.7.1.0-centos7.tar.gz
```

8. Create HDP repo with the following contents:

```
[root@rhel1]# cat /etc/yum.repos.d/hdp.repo
[HDP-3.0.1.0]
name= Hortonworks Data Platform Version - HDP-3.0.1.0
baseurl= http://rhel1.hdp3.cisco.com/Hortonworks/HDP/centos7/3.0.1.0-187
gpgcheck=0
enabled=1
priority=1
[HDP-GPL-3.0.1.0]
```

```
name=Hortonworks GPL Version - HDP-GPL-3.0.1.0
baseurl= http://rhel1.hdp3.cisco.com/Hortonworks/HDP-GPL/centos7/3.0.1.0-187
gpgcheck=0
enabled=1
priority=1
[HDP-UTILS-1.1.0.22]
name=Hortonworks Data Platform Utils Version - HDP-UTILS-1.1.0.22
baseurl= http://rhel1.hdp3.cisco.com/Hortonworks/HDP-UTILS/centos7/1.1.0.22
gpgcheck=0
enabled=1
priority=1
```



To verify the files, go to: http://rhel1.hdp3.cisco.com/Hortonworks.

9. Create the Ambari repo:

```
vi /etc/yum.repos.d/ambari.repo
[Updates-ambari-2.7.1.0]
name=ambari-2.7.1.0 - Updates
baseurl=http://rhel1.hdp3.cisco.com/Hortonworks/ambari/centos7/2.7.1.0-169
gpgcheck=0
enabled=1
priority=1
```

10. From the admin node copy the repo files to /etc/yum.repos.d/ of all the nodes of the cluster:

```
# ansible nodes -m copy -a "src=/etc/yum.repos.d/hdp.repo dest=/etc/yum.repos.d/."
# ansible nodes -m copy -a "src=/etc/yum.repos.d/ambari.repo dest=/etc/yum.repos.d/."
```

Downgrade Snappy on All Nodes

Downgrade snappy on all data nodes by running this command from admin node:

ansible all -m command -a "yum -y downgrade snappy"

HDP Installation

To install HDP, complete the following the steps:

Install and Setup Ambari Server on rhel1

1. Run the following command in rhel1 to install ambary-server:

```
#yum -y install ambari-server
Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use subscription-manager to register.
Resolving Dependencies
--> Running transaction check
--> Package ambari-server.x86_64 0:2.7.1.0-169 will be installed
--> Processing Dependency: postgresql-server >= 8.1 for package: ambari-server-2.7.1.0-169.x86_64
--> Running transaction check
---> Package postgresql-server.x86_64 0:9.2.23-3.el7_4 will be installed
--> Processing Dependency: postgresql-libs(x86-64) = 9.2.23-3.el7_4 for package: postgresql-server-
9.2.23-3.el7_4.x86_64
--> Processing Dependency: postgresql(x86-64) = 9.2.23-3.el7_4 for package: postgresql-server-
9.2.23-3.el7_4.x86_64
--> Processing Dependency: libpq.so.5()(64bit) for package: postgresql-server-9.2.23-3.el7_4.x86_64
--> Running transaction check
```

---> Package postgresql.x86 64 0:9.2.23-3.el7 4 will be installed ---> Package postgresql-libs.x86 64 0:9.2.23-3.el7 4 will be installed --> Finished Dependency Resolution Dependencies Resolved _____ _____ Package Arch Version Repositorv Size _____ _____ Installing: ambari-server x86 64 2.7.1.0-169 Updates-ambari-2.7.1.0 _____353 M Installing for dependencies: x86 64 9.2.23-3.el7 4 postgresql 3.0 M rhel7.5 postgresql-libs x86 64 9.2.23-3.el7 4 rhel7.5 _____234 k postgresql-server x86_64 9.2.23-3.el7 4 rhel7.5 3.8 M Transaction Summary _____ _____ Install 1 Package (+3 Dependent packages) Total download size: 360 M Installed size: 452 M Downloading packages: (1/4): postgresql-libs-9.2.23-3.el7_4.x86_64.rpm | 234 kB 00:00:00 (2/4): postgresql-9.2.23-3.el7 4.x86 64.rpm | 3.0 MB 00:00:00 (3/4): postgresgl-server-9.2.23-3.el7 4.x86 64.rpm | 3.8 MB 00:00:00 (4/4): ambari-server-2.7.1.0-169.x86 64.rpm 1 353 MB 00:00:03 _____ _____ Total 109 MB/s | 360 MB 00:00:03 Running transaction check Running transaction test Transaction test succeeded Running transaction Installing : postgresql-libs-9.2.23-3.el7 4.x86 64 1/4 Installing : postgresql-9.2.23-3.el7 4.x86 64 2/4 Installing : postgresql-server-9.2.23-3.el7 4.x86 64 3/4 Installing : ambari-server-2.7.1.0-169.x86 64 4/4 Verifying : postgresql-9.2.23-3.el7_4.x86_64 1/4 Verifying : postgresql-libs-9.2.23-3.el7 4.x86 64 2/4 Verifying : postgresql-server-9.2.23-3.el7_4.x86_64 3/4 Verifying : ambari-server-2.7.1.0-169.x86 64 4/4Installed: ambari-server.x86 64 0:2.7.1.0-169 Dependency Installed: postgresql.x86 64 0:9.2.23-3.el7 4 postgresql-libs.x86 64 0:9.2.23-3.el7 4 postgresql-server.x86 64 0:9.2.23-3.el7 4

Complete!

Install PostgreSQL in rhel2

The PostgreSQL database is used by Ambari, Hive, and Oozie services.

The rhelp hosts the Hive and Oozie services and Ambari Server is installed on rhelp. To install, follow these steps:

- 1. Log into rhel2.
- 2. Install Red Hat Package Manager (RPM) according to the requirements of your operating system:

yum install https://yum.postgresql.org/9.6/redhat/rhel-7-x86_64/pgdg-redhat96-9.6-3.noarch.rpm

3. Install PostgreSQL version 9.5 or later:

yum install postgresql96-server postgresql96-contrib postgresql96

4. Initialize the database as shown in the below figure by running the following command:

/usr/pgsql-9.6/bin/postgresql96-setup initdb

```
[root@rhel2 ~]#
[root@rhel2 ~]#
[root@rhel2 ~]# /usr/pgsql-9.6/bin/postgresql96-setup initdb
Initializing database ... OK
[root@rhel2 ~]#
```

5. Start PostgreSQL:

```
# systemctl enable postgresql-9.6.service
# systemctl start postgresql-9.6.service
```

6. Open /var/lib/pgsql/9.6/data/postgresql.conf and update to the following:

listen_addresses = '*'

7. Update these files on rhel2 in the location chosen to install the databases for Hive, Oozie and Ambari, using the host ip addresses:

```
[root@rhel2 ~]# cat /var/lib/pgsql/9.6/data/pg hba.conf|tail -n 20
# TYPE DATABASE
                       USER
                                      ADDRESS
                                                              METHOD
# "local" is for Unix domain socket connections only
#local all
                       all
                                                               peer
# IPv4 local connections:
#host
       all
                       all
                                       127.0.0.1/32
                                                               ident
# IPv6 local connections:
                                      ::1/128
host
      all
                      all
                                                              ident
# Allow replication connections from localhost, by a user with the
# replication privilege.
#local replication
                                                               peer
                       postgres
                                       127.0.0.1/32
#host
       replication
                      postgres
                                                               ident
      replication
                      postgres
                                       ::1/128
                                                               ident
#host
local
      all postgres
                             peer
```

local all all md5 host all postgres,hive,oozie 10.16.1.32/24 md5 host all ambari 10.16.1.31/24 md5 [root@rhel2 ~]#



Before adding new entries, comment the old entries as mentioned above.

8. Restart PostgreSQL:

```
# systemctl stop postgresql-9.6.service
# systemctl start postgresql-9.6.service
```

9. Run the following:



For more information about setting up PostgreSQL, go to: <u>https://docs.hortonworks.com/HDPDocuments/Ambari-</u> 2.7.1.0/bk_ambari-installation/content/install-postgres.html

Create Database for Ambari

To create the database for Ambari, follow these steps:

1. Run the following commands mentioned below in bold to create and prepare database for Ambari:

```
[root@rhel2 ~]# sudo -u postgres psql
could not change directory to "/root": Permission denied
psql (9.6.10)
Type "help" for help.
postgres=# \dt
No relations found.
postgres=#
postgres=#
postgres=# create database ambari;
CREATE DATABASE
postgres=# create user ambari with password 'bigdata';
CREATE ROLE
postgres=# grant all privileges on database ambari to ambari;
GRANT
postgres=# \connect ambari;
You are now connected to database "ambari" as user "postgres".
ambari=# create schema ambari authorization ambari;
CREATE SCHEMA
ambari=# alter schema ambari owner to ambari;
ALTER SCHEMA
ambari=# alter role ambari set search_path to 'ambari', 'public';
ALTER ROLE
ambari=# \q
```

2. Restart PostgreSQL:

[[root@rhel2 ~] # systemctl restart postgresql-9.6.service

3. Verify the ambari user by logging into psql:



4. Load the Ambari Server database schema:



Pre-load the Ambari database schema into your PostgreSQL database using the schema script.

- 5. Find the Ambari-DDL-Postgres-CREATE.sql file in the /var/lib/ambari-server/resources/ directory of the Ambari Server host after you have installed Ambari Server.
- 6. Copy /var/lib/ambari-server/resources/ from rhel1 to rhel2:/tmp/.

[root@rhel1 ~]# scp -r /var/lib/ambari-server/resources/* rhel2:/tmp/

7. Run the following command to launch the Ambari-DDL-Postgres-CREATE.sql script:

```
[root@rhel2 tmp]# cd /tmp
[root@rhel2 tmp]# psql -U ambari -d ambari
Password for user ambari:
psql (9.6.10)
Type "help" for help.
ambari=> \i Ambari-DDL-Postgres-CREATE.sql
CREATE TABLE
CREATE TABLE
CREATE TABLE
..... OUTPUT OMITTED ----
```

8. Check the table is created by running \dt command:

[root@rhe] psql (9.6 Type "hel]	12 ~]# psql -U ambari -d ambari .10) p" for help.		
ambari=>			
ambari=>			
ambari=> '	\dt		
	List of relations		
Schema	Name	Type	Owner
+	+		+
ambari	adminpermission	table	ambari
ambari	adminprincipal	table	ambari
ambari	adminprincipaltype	table	ambari
ambari	adminprivilege	table	ambari
ambari	adminresource	table	ambari
ambari	adminresourcetype	table	ambari
ambari	alert_current	table	ambari
ambari	alert definition	table	ambari
ambari	alert_group	table	ambari
ambari	alert_group_target	table	ambari
ambari	alert grouping	table	ambari
ambari	alert_history	table	ambari

9. Restart PostgreSQL:

[root@rhel2 ~]# systemctl restart postgresql-9.6.service

Create Database for Hive

Run the following command as shown in bold to create and prepare database for Hive:

```
[root@rhel2 ~]# sudo -u postgres psql
could not change directory to "/root": Permission denied
psql (9.6.10)
Type "help" for help.
postgres=# create database hive;
CREATE DATABASE
postgres=# create user hive with password 'bigdata';
CREATE ROLE
postgres=# grant all privileges on database hive to hive;
GRANT
postgres=# \q
[root@rhel2 ~]#
```

Create Database for Oozie

Run the following command to create and prepare database for Oozie:

```
[root@rhel2 ~]# sudo -u postgres psql
could not change directory to "/root": Permission denied
psql (9.6.10)
Type "help" for help.
postgres=# create database oozie;
CREATE DATABASE
postgres=# create user oozie with password 'bigdata';
CREATE ROLE
postgres=# grant all privileges on database oozie to oozie;
GRANT
postgres=# \q
[root@rhel2 ~]#
```

Setup Ambari Server On Admin Node(Rhel1)

To setup the Ambari server, follow these steps:

1. Install the PostgreSQL JDBC driver:

```
[root@rhell 2.7.1.0-169]# yum -y install postgresql-jdbc*
Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use subscription-manager to register.
Resolving Dependencies
--> Running transaction check
---> Package postgresql-jdbc.noarch 0:9.2.1002-5.el7 will be installed
--> Processing Dependency: jpackage-utils for package: postgresql-jdbc-9.2.1002-5.el7.noarch
--> Processing Dependency: java for package: postgresql-jdbc-9.2.1002-5.el7.noarch
--> Running transaction check
---> Package java-1.8.0-openjdk.x86 64 1:1.8.0.161-2.b14.el7 will be installed
```

2. Configure Ambari server to use the JDBC driver for connectivity to Ambari database in PostgreSQL:

```
[root@rhel1 2.7.1.0-169]# ambari-server setup --jdbc-db=postgres --jdbc-
driver=/usr/share/java/postgresql-jdbc.jar
Using python /usr/bin/python
Setup ambari-server
Copying /usr/share/java/postgresql-jdbc.jar to /var/lib/ambari-server/resources/postgresql-jdbc.jar
If you are updating existing jdbc driver jar for postgres with postgresql-jdbc.jar. Please remove the old
driver jar, from all hosts. Restarting services that need the driver, will automatically copy the new jar
to the hosts.
JDBC driver was successfully initialized.
Ambari Server 'setup' completed successfully.
```

3. Setup Ambari Server by running the following command:

```
[root@rhel1 ~] # ambari-server setup -j $JAVA HOME
Using python /usr/bin/python
Setup ambari-server
Checking SELinux...
SELinux status is 'disabled'
Customize user account for ambari-server daemon [y/n] (n)? n
Adjusting ambari-server permissions and ownership...
Checking firewall status...
Checking JDK...
WARNING: JAVA HOME /usr/java/jdk1.8.0 181-amd64 must be valid on ALL hosts
WARNING: JCE Policy files are required for configuring Kerberos security. If you plan to use
Kerberos, please make sure JCE Unlimited Strength Jurisdiction Policy Files are valid on all hosts.
Check JDK version for Ambari Server...
JDK version found: 8
Minimum JDK version is 8 for Ambari. Skipping to setup different JDK for Ambari Server.
Checking GPL software agreement...
GPL License for LZO: https://www.gnu.org/licenses/old-licenses/gpl-2.0.en.html
Enable Ambari Server to download and install GPL Licensed LZO packages [y/n] (n)? y
Completing setup...
Configuring database...
Enter advanced database configuration [y/n] (n)? y
Configuring database...
_____
Choose one of the following options:
[1] - PostgreSQL (Embedded)
[2] - Oracle
[3] - MySQL / MariaDB
[4] - PostgreSQL
[5] - Microsoft SQL Server (Tech Preview)
[6] - SQL Anywhere
[7] - BDB
                    ______
Enter choice (4):
Hostname (rhel2.hdp3.cisco.com):
Port (5432):
Database name (ambari):
```

```
Postgres schema (ambari):

Username (ambari):

Enter Database Password (bigdata):

Configuring ambari database...

Configuring remote database connection properties...

WARNING: Before starting Ambari Server, you must run the following DDL against the database to create the

schema: /var/lib/ambari-server/resources/Ambari-DDL-Postgres-CREATE.sql

Proceed with configuring remote database connection properties [y/n] (y)? y

Extracting system views...

.....

Adjusting ambari-server permissions and ownership...

Ambari Server 'setup' completed successfully.
```

4. Start the Ambari Server:

[root@rhel1 ~]# ambari-server start

```
[root@rhel1 ~]# ambari-server start
Using python /usr/bin/python
Starting ambari-server
Ambari Server running with administrator privileges.
Organizing resource files at /var/lib/ambari-server/resources...
Ambari database consistency check started...
Server PID at: /var/run/ambari-server/ambari-server.pid
Server out at: /var/log/ambari-server/ambari-server.out
Server log at: /var/log/ambari-server/ambari-server.log
Waiting for server start.....
Server started listening on 8080
DB configs consistency check: no errors and warnings were found.
Ambari Server 'start' completed successfully.
[root@rhel1 ~]# ^C
[root@rhel1 ~]# ________
```

5. To check status of Ambari Server, run the following command:

ambari-server status

```
[root@rhel1 ~]# ambari-server status
Using python /usr/bin/python
Ambari-server status
Ambari Server running
Found Ambari Server PID: 65658 at: /var/run/ambari-server/ambari-server.pid
[root@rhel1 ~]# _____
```

Launch the Ambari Server

When the Ambari service starts, access the Ambari Install Wizard through the browser. To launch the Ambari server, follow these steps:

1. Point the browser to http://<ip address for rhel1>:8080 or http://rhel1.hdp3.cisco.com:8080

The Ambari Login screen opens.

← → ♂ ③ Not secure rhel1.hdp3.cisco.com:8080/#/login		Q 🛠 🖯
ᢙ Ambari		
	Sign in Username Password	
	SIGN IN	

- 2. Log into the Ambari Server using the default username/password: **admin/admin**. This can be changed at a later period of time.
- 3. When logged in the "Welcome to Apache Ambari" window opens.

← → C	hdp3.cisco.com/8080/views/ADMIN_VIEW/2.7.1.0/INSTANCE/#/clusterInformation	• @ # O :
ᢙ Ambari	Admin / Cluster Information	🚢 admin 👻
🚔 Cluster Management 🗸 🗸		
Cluster Information	Welcome to Apache Ambari	
Remote Clusters	Provision a cluster, manage who can access the cluster, and customize views for Ambari users.	
🖀 Users		
₩ Views	Create a Cluster Use the Install Wizard to select services and configure your cluster	

Create the Cluster

To create the cluster, follow these steps:

- 1. To create a cluster click "LAUNCH INSTALL WIZARD."
- 2. From the Get started page type "Cisco_HDP" for the name for the cluster.
- 3. Click Next.



Select Version

To select the version, follow these steps:

1. In the Select Version section, choose the HDP 3.0.1.0 version.

← → C ① Not secure rhel1.hdp	o3.cisco.com:8080/#/installer/step1	
Installer		
Get Started	Select Version Select the software version and method of delivery	for your cluster.
Select Version Install Options	HDP-3.0	
(3) Confirm Hosts	Accumulo	1.7.0
(4) Choose Services	Infra Solr	0.1.0
(5) Assign Masters	Ambari Metrics	0.1.0
U	Atlas	1.0.0
6 Assign Slaves and Clients	Druid	0.12.1
Customize Services	HBase	2.0.0
8 Review	Repositories Using a Public Repository requires Internet con	nectivity. Using a Local Repository requires you have configured the software in a repository available in your network.
(9) Install, Start and Test	Use Public Repository	O Use Local Repository
D Summary	Provide Base URLs for the Operating	Systems you are configuring.

- 2. Under Repositories, select "Use Local Repository."
- 3. Update the Redhat 7 HDP-3.0 URL to http://rhel1.hdp3.cisco.com/Hortonworks/HDP/centos7/3.0.1.0-187/
- 4. Update the Redhat 7 HDP-3.0-GPL URL to http://rhel1.hdp3.cisco.com/Hortonworks/HDP-GPL/centos7/3.0.1.0-187/
- 5. Update the Redhat 7 HDP-UTILS-1.1.0.22 to http://rhel1.hdp3.cisco.com/Hortonworks/HDP-UTILS/centos7/1.1.0.22/

	HDP-3.0	http://rhel1.hdp3.cisco.com/Hortonworks/HDP/centos7/3.0.1.0-187/	5
redhat7	HDP-3.0-GPL	http://rhel1.hdp3.cisco.com/Hortonworks/HDP-GPL/centos7/3.0.1.0-187/	0
	HDP-UTILS-1.1.0.22	http://rhell.hdp3.cisco.com/Hortonworks/HDP-UTILS/centos7/1.1.0.22/	C



Select Hosts

To build up the cluster, you need to provide the general information about how you want to set up the cluster. This requires providing the Fully Qualified Domain Name (FQDN) of each of the hosts. You also need to provide access to the private key file that was created in Set Up Password-less SSH; this is used to locate all the hosts in the system and to access and interact with them securely.

- 1. Use the **Target Hosts** text box to enter the list of host names, one per line. Ranges inside brackets can also be used to indicate larger sets of hosts.
- 2. Select the option **Provide your SSH Private Key** in the Ambari cluster install wizard.

3. Copy the contents of the file /root/.ssh/id_rsa on rhel1 and paste it in the text area provided by the Ambari cluster install wizard.



Make sure there is no extra white space after the text-----END RSA PRIVATE KEY-----

[root@rhel1 ~]# cat /root/.ssh/id_rsa

----BEGIN RSA PRIVATE KEY----

MIIE0QIBAAKCAQEAyD0IRbk4mBZrizc0/g0M2iYT2h4vxkIxA/uvQVPthFreUdgT Zehw/Qtdk7meeqhqqsHmb1CriF0m6SxvPEXW2cGoAx75hZwTuDIR3Qlvk6oYUmDW BKq5TMfUMKfD7tknkGkq5N+YHsPCoNILlz/Wqc01hZZ0tiCmrxeRnPGSlJY74/Db A0BewMuNajAoVppPD6cLGF6/NKORpEDUnCuwe5pCRV5tko+qzBeBF5oeCS6Ya6I7 nS0HplJXV0Mv23SNUwl3cswbqLdrr3atG6YRieVrmmr/PlrKMp192tzQ1mHZMBqG w1RJTILjygW0gp5g7NQBGeM7sX4V60mzv4vmzwIBIwKCAQEAg4+UEI+o2PjKVCuX 2h+XEwMUXCJ3KoNEyBpr2nj7KxckYas/8oLN6B1pYROUB3X2YZVc6hBwuLI+JDMk hrGNMALqwDjtHUl0yX/9HDlmlDyTo9k8LvPY2q8zqvHnJ+3Jisi92Dspc01xRRxQ wnpofjAmlCDx5WXp4MZYX9HynCcKmheFefobLys6qloxd84eHW1y6b0xU1dh7hsQ pcK+xpdFWlsHYFbvckTuCHUAezF4+uBT5F0PMiD7PwzrvbXKA65ABuezv9gg2/I1 PekIkRvbosniFbBUi220S1uN/gsaZgmSQ9gTarJ1V8zMy6K31LETcOckl2LZHRX2 5sEx6wKBqQD9CiKc0HFiulrQWW5cLTDJU8wzTiNK4M91Qb2L0hfFuzfluiAl3Ref yiL9MjE3A5Mnn9pcRxMmXXPF4t9iuLh3+3tCsr1TzPm14WT+Fipa9sh+3JZ2HKqm pCquAEdoFRK4oP3/yYQq95qie2SC9sB0z6zVohdyNUvnkiMb9vwi3wKBqQDKiyTi Yu4210wsYKfZ7YjomjRKUFaH4CKtnyJy1SM3wFPRnZJd4BUaMq0DaTxr2tW4si+4 t88M8XS6FHGHymSqRtL0tYzMlmmwUtjCLNZQfqSeq1NovekXxXL0iUzel8PL3Z0H AeBj0/GLQ3SF/PGWMokCwNtaJoV/xldBdIsqEQKBqEERPBmx8UVF3NZ9ZYVqtMY0 09KtsU3Ex52x0ad1VpHt5TsSmo1kv06TEE+8cw41f2x5j+vXwxh+bjozBj30/Dwc GGGbrQbrkKscs5HLL3Z5+QqtwEpB4hiQnUKvnVVHP1QMJA6S53YxCdz7KHlypnqq bkWQFKhW2QEIUivDKuRlAoGASzr/EkIAtUfFb5Gdbj0n4V3Y6Gb7kY3DvNS1BhSm rk7ADAdTnzX5NZ3L08gAf9TwS+ppfx+zTfNIn0MFmNY1Y9EpyJs0S/1adLE0roWu sC8J8bu/5RNWk8z+z9s5zwUrd5txT2cY1J8t1KQGtWyUPxoVoe/ccfENA5LP872s xnsCgYAFRE4SbB416p9miiR1+gNCiihM9N+FmHMmcP/y80QL/MoAYoHB1Tn8cwVu l+sju4bWGUZvnGMWXwpEU5zVBra+yShh309IwjP/lkpCNWz7CX+/uI6FY+sl2xTr t5P/Avh0vUKMhRFjXFQoY5yqNUkasvIu6S8Q1unl8N2IhEgw1g== ----END RSA PRIVATE KEY---

4. Click the **Register and Confirm** to continue.

uur SSH key: n Name (FQDN), one per line. Or use Pattern Expressions y register Perform manual registration on hosts and do not use SSH
eur SSH key n Name (FQDN), one per line. Or use Pattern Expressions y register Perform manual registration on hosts and do not use SSH
n Name (FQDN), one per line. Or use Pattern Expressions
y register O Perform manual registration on hosts and do not use SSH
y register O Perform manual registration on hosts and do not use SSH
XXMqZNFU9#XXDDq3NC5MUgKte 73wktzxyxT/oMA3waXhjGNY bgr91BRST750EndwnlttanXX R3Tsd52ED6fJT00aljwR22vd qY119Gxqm305eETkilkgKML x
root
22
g¥4

Hostname Pattern Expressions

1. Click OK in the Host Name Pattern Expressions popup.



Confirm Hosts

Confirm Hosts helps ensure that Ambari has located the correct hosts for the cluster and checks those hosts to make sure they have the correct directories, packages, and processes to continue the install.

To confirm host, follow these steps:

- 1. If any host was selected in error, remove it by selecting the appropriate checkboxes and clicking the grey **Remove Selected** button.
- 2. To remove a single host, click the small white **Remove** button in the Action column.
- 3. When the list of hosts is confirmed, click **Next**.

O O HOLSECULE III	er understensenten uns 2001 « Luistene Laceba			Y H
er				1 ac
Started	Confirm Hosts Registering your hosts. Preserve confirm the host list and remove any hosts that you do not want to include in the cluster.			
all Options				Show: All (17) installing.(0) Registering.(0) Success.(17) Fail.(
firm Hosts	Host	Progress	Status	Action
ose Services	 rhel1.hdp3.cisco.com 		Success	ŵ
gn Masters	 rhel2.hdp3.cisco.com 		Success	ŵ
an Slaves and Clients	 rhel3 hdp3 cisco com 		Success	ŵ
	 rhel4.hdp3.cisco.com 		Success	ŵ
omize Services	 rhel5.hdp3.cisco.com 		Success	Ê
W	 rhel6.hdp3.cisco.com 		Success	8
Start and Test	 rhel7.hdp3.cisco.com 		Success	10
, start and rest	rhel8.hdp3.cisco.com		Success	10
sary	 rhel9.hdp3.cisco.com 		Success	10
	 rhei10.hdp3.cisco.com 		Success	Û
	rhei11.hdp3.cisco.com		Success	Û
	16 Other Registered Hosts			
	Please wait while the hosts are being checked for potential problems	0		

Choose Services

HDP is made up of a number of components. Go to Hortonworks Understand the Basics for more information.

To choose services, follow these steps:

- 1. Select **all** to preselect all items.
- 2. When you have made your selections, click **Next**.

← → C (1) Not secure rhel1.hd	p3.cisco.com:8080/#/installer/step4		Q \$	Θ:
Installer			1 admir	1.
C Get Started	Choose File System Choose which file system you want to install on your cli	luster.		
Select Version	Service	Version	Description	
Install Options	MDFS	3.1.1	Apache Hadoop Distributed File System	
Confirm Hosts				
4 Choose Services	Choose Services Choose which services you want to install on your clust	ter.		
Assign Masters	Service	Version	Description	
6 Assign Slaves and Clients	YARN + MapReduce2	3.1.1	Apache Hadoop NextGen MapReduce (YARN)	
⑦ Customize Services	🗹 Tez	0.9.1	Tez is the next generation Hadoop Query Processing framework written on top of YARN.	
(®) Review	🗹 Hive	3.1.0	Data warehouse system for ad-hoc queries & analysis of large datasets and table & storage management service	
	HBase	2.0.0	Non-relational distributed database and centralized service for configuration management & synchronization	
(9) Install, Start and Test	C Pig	0.16.0	Scripting platform for analyzing large datasets	
D Summary	Sqoop	1.4.7	Toot for transferring buik data between Apache Hadoop and structured data stores such as relational databases	
	Oozie	4.3.1	System for workflow coordination and execution of Apache Hadoop jobs. This also includes the installation of the optional Oozie Web Console which relies on and will install the ExtJS Library.	

Assign Masters

The Ambari installation wizard attempts to assign the master nodes for various services that have been selected to appropriate hosts in the cluster, as listed in Table 8. The right column shows the current service assignments by host, with the hostname and its number of CPU cores and amount of RAM indicated.

1. Reconfigure the service assignments to match Table 8 shown below.

Service Name	Host
NameNode	rhel1, rhel3 (HA)
SNameNode	rhel2
History Server	rhel2
App Timeline Server	rhel2
Resource Manager	rhel2, rhel3 (HA)
Hive Metastore	rhel2
WebHCat Server	rhel2
HiveServer2	rhel2
HBase Master	rhel2
Oozie Server	rhelı
Zookeeper	rhel1, rhel2, rhel3
Spark History Server	rhel2
SmartSense HST Server	rhelı
Grafana	rhelı
Atlas Metadata Server	rhel2
Metrics Collector	rhelı

Table 8Reconfigure the service assignments

2. Click Next.

Assign Slaves and Clients

The Ambari install wizard attempts to assign the slave components (DataNodes, NFSGateway, NodeManager, RegionServers, Supervisor, and Client) to appropriate hosts in the cluster.

To assign slaves and clients, follow these steps:

- 1. Reconfigure the service assignment to match the values shown in Table 9.
- 2. Assign DataNode, NodeManager, RegionServer, and Supervisor on nodes rhel3- rhel28.
- 3. Assign Client to all nodes.
- 4. Click Next.

Table 9 Services and Hostnames

Client Service Name	Host
DataNode	rhel4-rhel28
NFSGateway	rhelı
NodeManager	rhel4-rhel28
RegionServer	rhel4-rhel28
Supervisor	rhel4-rhel28
Client	All nodes, rhel1-rhel28

staller										adn
Get Started	Assign Slaves and Clients Assign slave and client components of Horist that are assigned master comp "Client" will install HDFS Client, YARN Client	io hosta you want to r ionents are shown wi Client, MapReduce2	un them on. In . Client, Tez Client, Hi	ve Client, HBase Clier	ıt, Pig Client, Sqoop	Chent, Dozie Client, ZooKeep	er Client, Accumu	lo Client, Infra Solr Client,	Atlas Metadata Client	and S
Install Options	Host	all none	ali none	all none	all none	all none	all none	all none	all none	all
Onfirm Hosts	rhel1.hdp3.cisco.com •	DataNode	NFSGateway	NodeManager	RegionServer	Phoenix Query Server	Supervisor	Accumulo TServer	C Ranger Tagsync	
Choose Seruree	rhel2.hdp3.cisco.com e	DataNode	NFSGateway	NodeManager	RegionServer	Phoenix Query Server	Supervisor	Accumulo TServer	C Ranger Tagsync	
Choose services	rhel3.hdp3.cisco.com.e	DataNode	NFSGateway	O NodeManager	RegionServer	Phoenix Query Server	Supervisor	Accumulo TServer	Ranger Tagsync	
Assign Masters	rhel4.hdp3.cisco.com	DataNode	NFSGateway	NodeManager	RegionServer	Phoenix Query Server	Supervisor	Accumulo TServer	Ranger Tagsync	
Assign Slaves and Clients	rhel5.hdp3.cisco.com	DataNode	NFSGateway	NodeManager	RegionServer	Phoenix Query Server	Supervisor	Accumulo TServer	Ranger Tagsync	
	rhel6.hdp3.cisco.com	Ø DataNode	NFSGateway	NodeManager	RegionServer	Phoenix Query Server	Supervisor	Accumulo TServer	Ranger Tagsync	
Customize Services	rhel7.hdp3.cisco.com	DataNode	NFSGateway	NodeManager	RegionServer	Phoenix Query Server	Supervisor	Accumulo TServer	Ranger Tagsync	
Review	rhel8.hdp3.cisco.com	DataNode	NFSGateway	NodeManager	RegionServer	Phoenix Query Server	Supervisor	Accumulo TServer	Ranger Tagsync	
No	rhel9.hdp3.cisco.com	DataNode	NFSGateway	NodeManager	RegionServer	Phoenix Query Server	Supervisor	Accumulo TServer	Ranger Tagsync	
/ install, start and rest	rhel10.hdp3.cisco.com	DataNode	NFSGateway	NodeManager	RegionServer	Phoenix Query Server	Supervisor	Accumulo TServer	🖸 Ranger Tagsync	
) Summary	rhel11.hdp3.cisco.com	DataNode	NFSGateway	NodeManager	RegionServer	Phoenix Query Server	Supervisor	Accumulo TServer	C Ranger Tagsync	
	the state of the s	Detablada	In MICCO etermine		· Daninatanan	Dhanie Ouno Camer	· Promanulane	Annunda Transe	Denner Tanmine	

Customize Services

This section shows the tabs that manage configuration settings for Hadoop components. The wizard attempts to set reasonable defaults for each of the options here, but this can be modified to meet specific requirements. The following sections provide configuration guidance that should be refined to meet specific use case requirements.

The following changes need to be made:

- Memory and service level settings for each component and service level tuning.
- Customize the log locations of all the components to make sure growing logs do not cause the SSDs to run out of space.

Credentials

Specify the credentials as per your organizational policies for services in CREDENTIALS tab as shown below.

C (i) Not secu	re rhel1.hdp3.cisco.com:8080/#/installer/step7				or Q ☆
					1
tarted		CCOUNTS FALL CONFIGURATIONS			
t Version	Please provide credentials for these services				
Options		Ubername*	Password*	Confirm Pasaword*	
m Hosts	Accumulo Root	N/A		******	
e Services	Accumulo instance Secret	N/A	******		
Masters	Grafana Admin	admin			
nize Services	Atlas Admin User	admin	******	******	
	Druid Metadata User	druid	******	******	
Start and Test	Hive Database	hive	******	******	
ey.	Knox Mester Secret	N/A	******	******	
	Gozie Detabase	oozie			
	Ranger Admin User Credentials	admin			
	Ranger Admin Credentials For Ambari User	amb_ranger_admin			
	Ranger Admin DB Credentials	rangeradmin			
	Ranger Usersync User's Password	N/A			
	Ranger Tagsync User's Password	N/A			

Databases

In the DATABASES tab, to configure the database for DRUID, HIVE, OOZIE, and RANGER, follow these steps:

1. Configure DRUID as shown below.

DRUID META DATA STORAGE Druid Metadata storage database name	
druid	
PostGRESQL +	
Vetadata storage user	
druid	
Metadata storane hostname	
rhel2.hdp3.cisco.com	
vletadata storage port	
5432	
vletadata storage connector url	
idbc:postgresgl://rhel2.hdp3.cisco.com:5432/druid	

2. Change the default log location by finding the Log Dir property and modifying the /var prefix to /data/disk1.

druid_log_dir	/data/disk1/log/druid
Druid PID dir	/var/run/druid

- 3. Configure HIVE:
 - a. Select Existing PostgreSQL database in the Hive Database drop-down list.
 - b. Enter Database Name as hive
 - c. Enter Database Username as hive
 - d. Enter Database URL as jdbc:postgresql://rhel2.hdp3.cisco.com/hive
 - e. Enter Database password (use the password created during hive database setup in earlier steps; for example, bigdata)
 - f. Click TEST CONNECTION to verify the connectivity
 - g. In Advanced tab, Change the default log location by filtering the Log Dir property and modifying the /var prefix to /data/disk1.
 - h. Change the WebHCat log directory by filtering the Log Dir property and modifying the /var prefix to /data/disk1.

Installer		👤 admin 🗸
Set Started		
Select Version	Please choose and configure the appropriate databases for these services	
Install Options	DRUID HIVE OOZIE RANGER	
Confirm Hosts		
Choose Services	Hive Database Hive Database Type	
Assign Masters	Existing PostgreSQL • PostgreS	
Assign Slaves and Clients	To use PostgreSQL with Hive, you must download the https://gbbc.postgresql.org/ from PostgreSQL. Once downloaded to the Ambari Server host. un: andbari-server stebu - glabe.dhep-ostgres.gl.Driver (Sess org.postgres.gl.Driver Cless org.postgres.gl.D	
(3) Review	Database Name Database Password	
Install, Start and Test	hive G man man	
(i) Summary	Database Username Inive C TEST CONNECTION Connection OK C	
	Database URL jdbc:postgresql://mel2.hdp3.cisco.com:5432/hive	
← BACK CANCEL		NEXT \rightarrow

4. Configure OOZIE:

- a. Select Existing PostgreSQL database in the Hive Database drop-down list.
- b. Enter Database Name as oozie.
- c. Enter Database Username as oozie.
- d. Enter Database URL as jdbc:postgresql://rhel2.hdp3.cisco.com/oozie.
- e. Enter Database password (use the password created during hive database setup in earlier steps; for example, bigdata).

- f. Click TEST CONNECTION to verify the connectivity.
- g. In Advanced tab, Change the default log location by filtering the Log Dir property and modifying the /var prefix to /data/disk1.

\leftrightarrow \rightarrow C (i) Not secure rhe	1.hdp3.cisco.com:8080/#/installer/step7			॰ ९ ☆ 🖰 :
Installer				💄 admin 🛩
Set Started		INTS FALL CONFIGURATION	łS	
Select Version	Please choose and configure the appropriate databases for these services			
Install Options	DRUID HIVE OOZIE RANGER			
Confirm Hosts				
Choose Services	Cozie Database		JDBC Driver Class	
Assign Masters	Existing PostgreSQL -		org.postgresql.Driver	
Assign Slaves and Clients	Database Name		Database Password	
Customize Services	oozie	C m		
(B) Review	Database Username			
Install, Start and Test	oozie	C 🗎	TEST CONNECTION Connection OK	0
10 Summary	Database URL			
	dbc:postgresql://rhel2.hdp3.cisco.com.5432/oozie			
				NDXT →

Advanced oozie-env

Oozie Log Dir	/data/disk1/log/oozie	(

- 5. Configure RANGER:
 - a. Select POSTGRES from DB FLAVOR drop-down list.
 - b. Provide a Ranger DB name. For example, ranger.
 - c. Provide Ranger DB Username such as rangeradmin.
 - d. Enter JDBC connect string for a Ranger Database as jdbc:postgresql//rhel2.dhp3.cisco.com:54323/ranger.
 - e. Ranger DB Host as rhel2.hdp3.cisco.com.
 - f. Enter Ranger DB password. (Ranger database is not previously created. provide password string that would be configured in the DB. For example, bigdata).
 - g. Select "Yes" for Setup Database and Database User.
 - h. Enter "postgres" in Database Administrator (DBA) username.
 - i. Enter Database Administrator (DBA) password.
 - j. Enter jdbc:postgresql://rhel2.hdp3.cisco.com:5432/postgres in JDBC connect string for root user.
 - k. Update Ranger Admin Log Dir from /var to /data/disk1.

$\leftrightarrow \rightarrow C$ (i) Not secure	e rhel1.hdp3.cisco.com:8080/#/installer/step7		
Installer			
🥑 Get Started			
Select Version	Please choose and configure the appropriate databases for these services		
🥏 Install Options	DRUID HIVE OOZIE RANGER		
Confirm Hosts			
Choose Services	DB FLAVOR	Ranger DB host	
Assign Masters	POSTGRES +	rhel2.hdp3.cisco.com	
Assign Slaves and Clients		Driver class name for a JDBC Ranger database	
Custamiza Sanicaa	To use PostgreSQL with Ranger, you must download the https://jdbc.postgreagl.org/ from PostgreSQL. Once downloaded to the Ambari Server host, run: ambari-server setupjdbc-db-postgresjdbc-dhiver=/path/to/postgres/org.postgresgl.Driver	org.postgresql.Driver	
Customize services	Ranger DB name		
B Hevew	ranger C	Renger DB peasword	
(9) Install, Start and Test			
Summary	ranger ub username C		
	JDBC connect string for a Ranger distabase		
	jdbc.postgresql:///hel2.hdp3.cisco.com:5432/ranger		
	Setup Database and Database User		
	Database Administrator (DBA) username	Database Administrator (DBA) password	
	postgres		
	JDBC connect string for root user		
	jdbc:postgresql://rhel2.hdp3.cisco.com.5432/postgres		
	TEST CONNECTION Connection OK		

HDFS

- 1. In Ambari, select the HDFS Service tab and use the "Search" box to filter for the properties mentioned in Table 10 and update their values.
- 2. Update the following HDFS configurations in Ambari.

Table 10 HDFS Configurations in Ambari

Property Name	Value
NameNode Java Heap Size	4096
Hadoop maximum Java heap size	4096
DataNode maximum Java heap size	4096
Datanode failed disk toleration	5

3. Change the default log location by filtering the Log Dir property and modifying the /var prefix to /data/disk1.



Advanced hadoop-env	
Hadoop PID Dir Prefix	/var/run/hadoop
Hadoop Root Logger	INFO,RFA
Hadoop Log Dir Prefix	/data/disk1/log/hadoop

MapReduce₂

- 1. In Ambari, choose the MapReduce Service tab and update the values as shown below.
- 2. Under the MapReduce2 tab, change the default log location by finding the Log Dir property and modifying the /var prefix to /data/disk1.

SETTINGS ADVANCED				
MapReduce Framework Map Memory	B B B B B B B B B B B B B B	2017/MB 2047 MB	10	0
Advanced mapred-env				
Mapreduce Log Dir Prefix	/data/disk1/log/hadoop-mapreduce		c	
Mapreduce PID Dir Prefix	/var/run/hadoop-mapreduce			

YARN

1. In Ambari, select the YARN Service, and update the following as shown in Table 11.

Table 11YARN Configuration

Property Name	Value
ResourceManager Java heap size	4096
NodeManager Java heap size	2048
YARN Java heap size	4096

Solution Design

Resource Manager		
ResourceManager Java heap size	4096	MB
Node Manager		
NodeManager Java heap size	2048	MB
Application Timeline Server		
General		
YARN Java heap size	4096	MB

2. Under YARN tab, change the default log location by filtering the Log Dir property and modifying the /var prefix to /data/disk1.



YARN requires other configurations such as config group, node labeling, enabling docker runtime, CPU/GPU scheduling and isolation, and so on, which can be found in section High Availability for HDFS NameNode and YARN ResourceManager.

High availability for NameNode and YARN Resource Manager can be configured using Ambari or also on non-Ambari clusters. This deployment guide covers the configuration of high availability using Ambari – Use the Ambari wizard interface to configure HA of the components.

HBase

Under the HBase tab, change the default log location by finding the Log Dir property and modifying the /var prefix to /data/disk1.

HBase Log Dir Prefix

/data/disk1/log/hbase

Zookeeper

Under the Zookeeper tab, change the default log location by filtering the Log Dir property and modifying the /var prefix to /data/disk1.

Advanced zookeeper-env	
ZooKeeper Log Dir	/data/ <u>disk1</u> /log/zookeeper
ZooKeeper PID Dir	/var/run/zookeeper

Storm

Under the Storm tab, change the default log location by finding the Log Dir property and modifying the /var prefix to /data/disk1.

Advanced storm-env	
Storm Log directory	/data/disk1/log/storm
Storm PID directory	/var/run/storm

Ambari Metrics

- 1. Choose the Ambari Metrics Service and expand the General tab and make the changes shown below.
- 2. Enter the Grafana Admin password as per organizational policy.
- 3. Change the default log location for Metrics Collector, Metrics Monitor and Metrics Grafana by finding the Log Dir property and modifying the /var prefix to /data/disk1.
- 4. Change the default data dir location for Metrics Grafana by finding the data Dir property and modifying the /var prefix to /data/disk1.

General

Metrics Service operation mode	embedded	
Metrics Collector log dir	/data/disk1/log/ambari-metrics-collector	
Metrics Collector pid dir	/var/run/ambari-metrics-collector	
Metrics Monitor log dir	/data/disk1/log/ambari-metrics-monitor	
Metrics Monitor pid dir	/var/run/ambari-metrics-monitor	
Grafana Admin Username	admin	
Grafana Admin Password		
Advanced ams-grafana-env		
Advanced ams-grafana-env Metrics Grafana data dir	/data/disk1/lib/ambari-metrics-grafana	
Advanced ams-grafana-env Metrics Grafana data dir Metrics Grafana log dir	/data/disk1/lib/ambari-metrics-grafana /data/disk1/log/ambari-metrics-grafana	
Advanced ams-grafana-env Metrics Grafana data dir Metrics Grafana log dir Metrics Grafana pid dir	/data/disk1/lib/ambari-metrics-grafana /data/disk1/log/ambari-metrics-grafana /var/run/ambari-metrics-grafana	

Advanced ams-hbase-env

HBase Log Dir Prefix

/data/disk1/log/ambari-metrics-collector

Accumulo

Select Accumulo Service and change the default log location by finding the Log Dir property and modifying the /var prefix to /data/disk1.

Advanced accumulo-env

Accumulo Log Dir

/data/disk1/log/accumulo

Atlas

Under the Atlas tab, change the default log location by finding the Log Dir property and modifying the /var prefix to /data/disk1.

Advanced atlas-env	
Metadata Data directory	/var/lih/atlac/data
metadata bata ancetory	/Yur/nb/ utus/ utu
Metadata Log directory	/data/disk1/log/atlas
Metadata PID directory	/var/run/atlas

Kafka

Under the Kafka tab, change the default log location by finding the Log Dir property and modifying the /var prefix to /data/disk1.

Advanced kafka-env	

Kafka Log directory

/data/disk1/log/kafka

Knox

- 1. Select the Knox Service tab and expand the Knox gateway tab and make the changes shown below.
- 2. Enter the Knox Master Secret password as per organizational policy.
- 3. For Knox, change the gateway port to 8444 to ensure no conflicts with local HTTP server.

Knox Gateway		
Knox Gateway host	rhel1.hdp3.cisco.com	
Knox Master Secret		
Advanced gateway-site		
gateway.port	8444	

SmartSense

The SmartSense account requires the Hortonworks support subscription. Subscribers can populate the properties as shown below.

SmartSense Account		Local Storage
Customer account name		Bundle storage directory
unspecified	C O	/var/llb/smartsense/hst-server/data
SmartSense ID unspecified	C •	Server temporary data directory /var/lib/smartsense/hst-server/tmp
Notification Email unspecified	c o	Agent temporary data directory /var/lib/smartsense/hst-agent/data/tmp
Enable Flex Subscription	2 •	

Spark

Select the Spark tab, change the default log location by finding the Log Dir property and modifying the /var prefix to /data/disk1.

Advanced livy2-env	
Livy2 Log directory	/data/disk1/log/livy2
Livy2 PID directory	/var/run/livy2
Advanced spark2-env	
Spark Log directory	/data/disk1/log/spark2
Spark PID directory	/var/run/spark2

Review

The assignments that have been made are displayed. Check to make sure all is correct before clicking the Deploy button. If any changes are necessary, use the left navigation bar to return to the appropriate screen.

\leftrightarrow \rightarrow C (i) Not secure rhe	1.hdp3.cisco.com:8080/#/installer/step8	⊶ ☆ ⊖ :
Installer		🚨 admin 🗸
Get Started	Review Please review the configuration before installation	
Select Version		
🤣 Install Options	Admin Name : admin Cluster Name : Clsco, HDP	
🕑 Confirm Hosts	Total Hosts : 17 (17 new) Repositories:	
Choose Services	redhat7 (HDP-3.0): http://thell.hdp3.cisco.com/Hortonworks/HDP/centos7/3.0.1.0-187/	
Assign Masters	redhat7 (HDP-3.0-GPL): http://thel1.hdp3.cisco.com/Hortonworks/HDP-GPL/centos7/3.0.1.0-187/	
Assign Slaves and Clients	redhat7 (HDP-UTILS-1.1.0.22): http://rhel1.hdp3.cisco.com/Hortonworks/HDP-UTILS/centos7/1.1.0.22/	
🕗 Customize Services	Services: HDFS	
Review	DataNode : 15 hosts NameNode : rhell hdp3.cisco.com	
Install, Start and Test	NFSGateway: 1 host SNameNode: rhel2.hdp3.cisco.com	
10 Summary		
← BACK CANCEL	[GENERATE BLUEPRINT PRINT DEPLOY →

Deploy

When the review is complete, click the **DEPLOY** button.

← → C ① Not secure m	el1.hdp3.cisco.com/8080/#/installer/step8				• ☆ 0 :
installer					L admin +
e on state	Review Please tenter the configuration before justification	Initializing Tasks Propading to Diploy 27 of 90 tasks completed			
	Admin Name . admin	•			
	Cluster Name : Cisco, HDP Total Hosts : 17 (17 new) Repositories				4
	redhart? (HDP-3.0) http://mei3.hdp3.cisco.com/Hort	inworks/HDP/centos7/3.0.1.0-187/			
	redhat7 (HDP-3.0-GPL). http://rfiel1.hdp3.cscn.com/Hurb	anworks/HDP-GPL/centros7/3.0.1.0-187/			
	redhar(7 (HDP-UTILS-1 1.0.22) http://md1.hdp3.cisco.com/Hort	anworks/HDP-UTILS/centos7/1.1.0.22/			
	Services: HDFS				
	DataNode : 15 hosts NameNode : thei1 hdp3 cisco co	m			
	NF5Gateway 1 host SNameNode rhel2.hdp3.clsco.c	om			
+ BACK CANCEL			L GENERATE BLUEPRINT	PRUKT	DEPLOY

Follow the installation process. Watch for warnings and failure by clicking on the link as shown below.

C O Not secure rhel1.hdp3.cis	sco.com:8080/#/installer/step9				• @ \$ E
Installer				Cisco_HDP	💄 admin 🗸
Get Started	Install, Start and Test Please wait while the selected services are installed and s	started.			
Select Version					32 % overall
Install Options	Show: All (17) In Progress (17) Warning (0)	<u>Success.(0)</u> <u>Fail.(0)</u>			
Confirm Hosts	Host	Status		Message	
Choose Services	rhel1.hdp3.cisco.com		19%	Installing Ranger Admin	
	rhel2.hdp3.cisco.com		33%	Install complete (Waiting to start)	
Assign Masters	rhel3.hdp3.cisco.com		33%	Install complete (Waiting to start)	
Assign Slaves and Clients	rhel4.hdp3.cisco.com		33%	Install complete (Waiting to start)	
Customize Services	rhel5.hdp3.cisco.com		33%	Install complete (Waiting to start)	
I	rhel6.hdp3.cisco.com		33%	Install complete (Waiting to start)	
🧭 Review	rhel7.hdp3.cisco.com		33%	Install complete (Waiting to start)	
Install, Start and Test	rhel8.hdp3.cisco.com		33%	Install complete (Waiting to start)	
	rhel9.hdp3.cisco.com		33%	Install complete (Waiting to start)	
Journality	rhel10.hdp3.cisco.com		33%	Install complete (Waiting to start)	

Summary of the Installation Process

On the Summary page click "COMPLETE."

← → C A Not secure rhel1.hdp3.cisco.com:8080/#/main/dashboard/metrics					
ᢙ Ambari	♠ / Dashboard / Metrics			🔅 🕕 💧 🗰 ք admin -	
↑ Dashboard				Cisco_HDP	
🚔 Services 🛛 🔐 🗸	METRICS HEATMAPS CONFIG HISTORY			METRIC ACTIONS * LAST 1 HOUR *	
• HDFS O					
• YARN					
MapReduce2	NameNode Heap	HDFS Disk Usage	NameNode CPU WIO	DataNodes Live	
Tez				15/15	
Hive	6%	0%	0.0%	15/15	
HBase					
Pig					
Sqoop	NameNode RPC	Memory Usage	Network Usage	CPU Usage	
• Oozie	0.15	92.1.69		100%	
 ZooKeeper 	0.15 ms	33.100	How here berees	50%	
 Storm 					
Accumulo					
 Infra Solr 	Cluster Load	NameNode Uptime	HBase Master Heap	HBase Ave Load	
Ambari Metrics					
 Atlas 	40	14h 45m 7s	13%	0.31	
 Kafka 	20				
 Knox 					
Ranger	Region In Transition	HBase Master Uptime	ResourceManager Heap	NodeManagers Live	
• SmartSense 🛛 🕕					
Spark2	0	14h 26m 43s	17%	15/15	
Zeppelin Note					
Druid					

High Availability for HDFS NameNode and YARN ResourceManager

High availability for NameNode and YARN Resource Manager can be configured using Ambari or also on non-Ambari clusters. This deployment guide covers the configuration of high availability using Ambari – Use the Ambari wizard interface to configure HA of the components.

The Ambari web UI provides a wizard-driven user experience that allows to configure high availability of the components in many Hortonworks Data Platform (HDP) stack services. The high availability of the components are achieved by setting up primary and secondary components. In the event that the primary component fails or becomes unresponsive, services failover to secondary component. After configuring the high availability for a service, Ambari enables you to manage and disable (roll back) the high availability of components within that service.

Configure the HDFS NameNode High Availability

The HDFS NameNode high availability feature enables you to run redundant NameNodes in the same cluster in an Active/Passive configuration with a hot standby. This eliminates the NameNode as a potential single point of failure (SPOF) in an HDFS cluster. With the release of Hadoop 3.0, you can configure more than one backup NameNode.

Prior to the release of Hadoop 2.0, the NameNode represented a single point of failure (SPOF) in an HDFS cluster. Each cluster had a single NameNode, and if that machine or process became unavailable, the cluster as a whole would be unavailable until the NameNode was either restarted or brought up on a separate machine. This situation impacted the total availability of the HDFS cluster in two major ways:

- In the case of an unplanned event such as a machine crash, the cluster would be unavailable until an operator restarted the NameNode.
- Planned maintenance events such as software or hardware upgrades on the NameNode machine would result in periods of cluster downtime.

HDFS NameNode HA avoids this by facilitating either a fast failover to one or more backup NameNodes during machine crash, or a graceful administrator-initiated failover during planned maintenance.



Secondary NameNode is not required in high availability configuration because the Standby node also performs the tasks of the Secondary NameNode.

Active NameNode honors all the client requests and the Standby NameNode acts as a backup. The Standby NameNode keeps its state synchronized with Active NameNode through a group of JournalNodes(JNs). When the Active node performs any namespace modification, the Active node durably logs a modification record to a majority of these JNs. The Standby node reads the edits from the JNs and continuously watches the JNs for changes to the edit log.

Prerequisites for NameNode High Availability

The following are the prerequisites for NameNode high availability:

- NameNode Machine: Hardware for Active and Standby node should be exactly identical.
- JournalNodes Machine: JournalNode daemon is relatively lightweight, therefore it can be co-located on machines with other Hadoop daemons; it is typically located on the management nodes.
- There MUST be at least three JournalNodes, because the edit log modifications must be written to a majority of JNs. This allows the system tolerate failure of a single machine. You may also run more than three JournalNodes, but in order to increase the number of failures that the system can tolerate, you must run an odd number of JNs (3, 5, 7, and so on).

- ZooKeeper Machines: For automatic failover capability, an existing Zookeeper cluster must exist. The Zookeeper service can also co-exist with other Hadoop daemons.
- In HA Cluster, the Standby NameNode also performs the checkpoints of the namespace state, therefore do not deploy a Secondary NameNode, CheckpointNode, or BackupNode in an high availability cluster.

Deploy the NameNode High Availability Cluster

To deploy the NameNode high availability on a Ambari managed cluster, follow these steps:



1. Log into Ambari. Click HDFS > CONFIGS. Click the ACTIONS drop-down list and click Enable NameNode HA to launch the wizard.

Figure 29 Enable NameNode HA

↑ / Services / HDFS / Configs	Cisco_HDP
SUMMARY HEATMAPS CONFIGS METRICS	ACTIONS
Version: 9 - SETTINGS ADVANCED	Config Group Default (17) Config Group Default (17) © Restart All © Restart DataNodes © Restart NFSGateways
No properties to display.	Move NameNode Move SNameNode Enable NameNode HA
	للله Add New HDFS Namespa الله Run Service Check الله Turn On Maintenance Mo
	

2. Step 1 launches the Enable NameNode HA wizard. On the Get Started page, specify the Nameservice ID as shown below. Click Next.

Figure 30 Enable NameNode HA Wizard – Get Started

Enable NameNode HA Wizard

(1) Get Started	Get Started
	This wizard will walk you through enabling NameNode HA on your cluster Once enabled, you will be running a Standby NameNode in addition to your Active NameNode. This allows for an Active Standby NameNode configuration that automatically performs failover. The process to enable HA involves a combination of automated steps (that will be handled by the wizard) and manual steps (that you must perform in sequence as instructed by the wizard). You should plan a cluster maintenance window and prepare for cluster downtime when enabling NameNode HA.
	If you have HBase running, please exit this wizard and stop HBase first.
	Nameservice ID: CiscoHDP

3. On the Select Hosts page, select the Additional NameNode and JournalNode. Click Next.

Figure 31 Enable NameNode HA Wizard – Select Hosts

Enable NameNode HA Wizard

Get Started Select Select a In addition 2 Select Hosts	Hosts host that will be running the additional NameNo on, select the hosts to run JournalNodes, which	ode store NameNode edit logs in a fault tolerant manner.	
3 Freezew	Current NameNode:	rhel1 hdp3 cisco com (125 4 GB, 56 cares)	
Greate Checkpoint	Additional NameNode:	rhei3.hdp3.cisco.com (125.4 GB, 56 cores)	•
3 Configure Components	JournalNode:	rhell hdn3 cisco com (125.4 GR-55 cores)	
Trittalize JournalNodes		mannaps.obtectmin(rea.4.ob) or corea)	
3 Start Components	JournalNode:	rhel2.hdp3.cisco.com (125.4 GB, 56 cores)	•
③ Initialize Metadata	JournalNode:	rhel3.hdp3.cisco.com (125.4 GB, 56 cores)	•
③ Finalize HA Setup			
9) Finalize HA Setup			

4. On the Review page, confirm the selection. To change any values, click Back, or to continue click Next.

Get Started	Confirm your host selections.			
Select Hosts				
martin	Current NameNode:	rhel1.hdp3.cisco.com		
) Review	Secondary NameNode:	rhel2.hdp3.cisco.com	- TO BE DELETED	
Create Checkpoint	Additional NameNode:	rhel3.hdp3.cisco.com	+ TO BE INSTALLED	
y oreare oncerpoint	JournalNode:	rhel1.hdp3.cisco.com	TO BE INSTALLED	
) Configure Components		rhel2.hdp3.cisco.com rhel3.hdp3.cisco.com	 TO BE INSTALLED TO BE INSTALLED 	
) Initialize JournalNodes				
) Start Components	Review Configuration Char The following lists the con review only and is not edita	nges. figuration changes tha able except for the dfs .	will be made by the Wizard to enable NameNode HA.	This information is for
) Initialize Metadata				
Prinaties (1), Parilie	HDES			

Figure 32 Enable NameNode HA Wizard – Review

5. Create a checkpoint on the NameNode on the linux server (rhel1.hdp3.cisco.com) as shown below.

Figure 33 Enable NameNode HA Wizard – Create Checkpoint

Enable NameNode HA Wi	zard	×
🥑 Get Started	Manual Steps Required: Create Checkpoint on NameNode	
Select Hosts	Login to the NameNode host rhel1.hdp3.cisco.com . Put the NameNode in Safe Mode (read-only mode):	
Peview	sudo su hdfs -l -c 'hdfs dfsadmin -safemode enter' 3. Once in Safe Mode, create a Checkpoint:	
(4) Create Checkpoint	sudo su hdfs -l -c 'hdfs dfsadmin -saveNamespace'	
	 You will be able to proceed once Ambari detects that the NameNode is in Safe Mode and the Checkpoint has been created successfully. 	
	If the Next button is enabled before you run the "Step 4: Create a Checkpoint" command, it means there is a recent Checkpoint already and you may proceed without running the "Step 4: Create a Checkpoint" command.	
() Finalize HA Setup		

6. SSH to current NameNode, rhel1.hdp3.cisco.com and run the following commands:




- 7. Return to the Ambari web UI, verify that the Checkpoint was created. Click Next.
- 8. See the progress bar on the Configure Components page. When the configuration steps are completed, click Next.

Figure 35 Enable NameNode HA Wizard – Configure Components

Enable NameNode HA Wiza	rd	×
C Get Started	Please wait while NameNode HA is being deployed.	
🗢 Review	Stop All Services 60%	
Create Checkpoint	🏠 Install Additional NameNode	
5 Configure Components	🏟 Install JournalNodes	
6 Initialize JournalNodes	Reconfigure HDFS	
(7) Start Components	Start JournalNodes	
(8) Initialize Metadata	Disable Secondary NameNode	
() Finalize HA Setup		
	NE	хт

9. Initialize the JournalNodes as shown below.

Enable NameNode HA Wiz	ard	×
🥥 Get Started	Manual Steps Required: Initialize JournalNodes	
Select Hosts	Login to the NameNode host rhel1.hdp3.cisco.com . Initialize the JournalNodes by running: Initialize the JournalNodes by running:	
Review	3. You will be able to proceed once Ambari detects that the JournalNodes have been initialized successfully.	
Create Checkpoint		
Configure Components		
6 Initialize JournalNodes		
(7) Start Components		
(8) Initialize Metadata		
③ Finalize HA Setup		

Figure 36 Enable NameNode HA Wizard – Initialize JournalNodes

- 10. SSH to the current NameNode, for example rhel1.hdp3.cisco.com.
- 11. Run the following command:

[root@rhel1 ~]# sudo su hdfs -l -c 'hdfs namenode -initializeSharedEdits'



- 12. Return to the Ambari UI, when Ambari detects success, click Next.
- 13. On the Start Components page, when completed, click Next.

Enable NameNode HA Wiza	rd	×
Get Started	Please proceed to the next step.	
Review	✓ Start ZooKeeper Servers	
Create Checkpoint	✔ Start Ambari Infra	
Configure Components	V Start Ranger	
Initialize JournalNodes	V Start NameNode	
T Start Components		
Initialize Metadata		
() Finalize HA Setup		
	NEXT	

Figure 38 Enable NameNode HA Wizard – Start Components

14. On the Initialize Metadata page, add the information as shown below.





15. SSH to rhel1.hdp3.cisco.com and run the following command:

[root@rhell ~]# sudo su hdfs -l -c 'hdfs zkfc -formatZK'

Figure 40	Initialize the Metadata for NameNode
[root@rhel1 ·	√] #
[root@rhel1	<pre>~]# sudo su hdfs -l -c 'hdfs zkfc -formatZK'</pre>
19/01/15 12:	49:24 INFO tools.DFSZKFailoverController: STARTUP MSG:
/*******	***************************************
STARTUP MSG:	Starting DFSZKFailoverController
STARTUP MSG:	host = rhel1/10.16.1.31
STARTUP MSG:	args = [-formatZK]
STARTUP MSG:	version = 3.1.1.3.0.1.0-187
STARTUP MSG:	classpath = /usr/hdp/3.0.1.0-187/hadoop/conf:/usr/hdp/3.0.1.0-187/hadoop/lib/nimbus-jose-jwt-4.41.1.jar:/usr
/hdp/3.0.1.0-	-187/hadoop/lib/ranger-hdfs-plugin-shim-1.1.0.3.0.1.0-187.jar:/usr/hdp/3.0.1.0-187/hadoop/lib/jersey-server-1.1
9.jar:/usr/h	dp/3.0.1.0-187/hadoop/lib/ranger-plugin-classloader-1.1.0.3.0.1.0-187.jar:/usr/hdp/3.0.1.0-187/hadoop/lib/jerse
y-servlet-1.	l9.jar:/usr/hdp/3.0.1.0-187/hadoop/lib/ranger-yarn-plugin-shim-1.1.0.3.0.1.0-187.jar:/usr/hdp/3.0.1.0-187/hadoo
p/lib/jetty-	111-9.3.19.v20170502.jar:/usr/hdp/3.0.1.0-187/hadoop/lib/accessors-smart-1.2.jar:/usr/hdp/3.0.1.0-187/hadoop/l

16. SSH to an additional NameNode, for example, rhel3.hdp3.cisco.com and run the following command:

[root@rhel3 ~]# sudo su hdfs -l -c 'hdfs namenode -bootstrapStandby'



17. Return to the Ambari web UI, click NEXT. Click OK on the confirmation pop-up window. Make sure the initialization of metadata was performed in NameNode and an additional NameNode as mentioned in step 15 and 16.

Enable NameNode HA Wizard	Confirmation	¢.
Select Hosts	Please confirm that you have run the manual steps before continuing.	
📀 Réview	CANCEL	
Create Checkpoint	Important! Be sure to login to the Additional NameNode host.	
🥑 Contigure Components	Initialize the metadata for the Additional NameNode by running	
🤣 Initialize JournalNodes	oudo su hdfs -1 -c 'hdfs namenodo -bootstrapStandby'	
Start Components	ase proceed once you have completed the steps above.	
Initialize Metadata		
③ Finalize HA Setup		
	NEXT→	

Figure 42 Enable NameNode HA Wizard – Confirmation for Initialize the Metadata

18. On the Finalize HA Setup page, you can see the progress bar as the high availability completes.

Cet Started		
Select Hosts	Please wait while NameNode HA is being deployed.	
🥏 Review	Start Additional NameNode	
Create Checkpoint	Controllers 35%	
Configure Components	Start Failover Controllers	
Initialize JournalNodes	🕸 Reconfigure Ranger	
Start Componento	🖨 Reconfigure HBase	
Start Components	🛱 Reconfigure AMS	
🥑 Initialize Metadata	🕸 Reconfigure Accumulo	
9 Finalize HA Setup	Delete Secondary NameNode	
	Stop HDFS	
	Start All Services	

Figure 43 Enable NameNode HA Wizard - Finalize High Availability Setup

- 19. Click COMPLETE when done.
- 20. Click HDFS > SUMMARY tab, verify the Active and Standby NameNode. The Quick Links pane also shows that rhel1.hdp3.cisco.com is running the Active NameNode and rhel3.hdp3.cisco.com is running in Standby NameNode.

Figure 44 Ambari – HDFS – Summary Information

/ Services / HD	FS / Summary				¢0	4 admir Cisco_HDP
SUMMARY HEATM	IAPS CONFIGS METRICS					ACTIONS
Summary					40	Quick Links
Components	Started ACTIVE NAMENODE 2m 42s NAMENODE UPTIME	 Started ZKFAILOVERCONTROLLER 10.8% 435.0 MB / 4.0 GB NAMENODE HEAP 	Started STANDBY NAMENODE	Started ZKFAILOVERCONTROLLER		rhell.hdp3.cisco.com (Active) NameNode UI NameNode Logs NameNode JMX Thread Stacks rhel3.hdp3.cisco.com (Standby)
	15/15 Started DATANODES	3/3 Live JOURNALNODES	1/1 Started NFSGATEWAYS			NameNode UI NameNode Logs NameNode JMX Thread Stacks
	15 Live	O Dead	0 Decommissioning			

Configure the YARN ResourceManger HA

This section provides instructions on setting up the ResourceManager (RM) HA feature in a HDFS cluster. The Active and Standby ResourceManagers embed the ZooKeeper based ActiveStandbyElector to determine which RM should be active.

Prerequisites for ResourceManager HA

The servers where Active and Standby RMs are run should have identical hardware. ٠

For automated failover configurations, there must be an existing Zookeeper cluster available. The ZooKeeper . service nodes can be co-located with the other Hadoop daemons.



At least three ZooKeeper servers must be running.

Deploy the ResourceManager HA

ResourceManager HA can be configured manually or through Ambari. These instructions are based on configuring ResourceManager HA using Ambari.

To setup ResourceManager HA, follow these steps:

1. From the Ambari web UI, click Services > YARN. Click the ACTIONS drop-down list and select Enable ResourceManager HA.

Figure 45 Services/YARN - Enable ResourceManger F	Figure 45	Services/YARN - Enable ResourceManger HA
---------------------------------------------------	-----------	------------------------------------------

/ Services / YAI	RN / Summary				<	to the second s
SUMMARY HEATM	IAPS CONFIGS METRICS					ACTIONS - > Start Stop
Summary						 Refresh YARN Capacity Scheduler Restart All Restart NodeManagers Move Timeline Service V1.5 Move ResourceManager
Components	 Started TIMELINE SERVICE V1.5 16/16 Started NODEMANAGERS 	Started RESOURCEMANAGER	Started TimeLine Service v2.0 READER	▲ Stopped yarn registry DNS		Move Timeline Service V2.0 Reader Move YARN Registry DNS Enable ResourceManager HA Kon Service Check Turn On Maintenance Mode
	6m 16s RESOURCEMANAGER UPTIMI	8				L Download Client Configs

This launches the ResourceManager HA wizard as shown below. Click NEXT. 2.

Enable ResourceManager HA Wizard – Get Started Figure 46

Get Started Select Host Review	Get Started This wizard will walk you through enabling BesourceManager HA on your cluster. Once enabled, you will be running a Standby ResourceManager in addition to your Active ResourceManager. This allows for an Active-Standby ResourceManager configuration that automatically performs failover. You should plan a cluster maintenance window and prepare for cluster downtime when enabling ResourceManager HA.	

From the Select Host page, select the host for Additional Resource Manager. Click NEXT. 3.

Figure 47 Enable ResourceManger HA Wizard – Select Host

Enable ResourceManager HA Wizard

Get Started	Select Host elect a host that will be running the additional ResourceManager	
Select Host Review Configure Components	Current ResourceManager: rhel2.hdp3 cisco.com (125.4 GB, 56 cores) • Additional ResourceManager: rhel3.hdp3.cisco.com (125.4 GB, 56 cores) • Additional ResourceManager: rhel3.hdp3.cisco.com (125.4 GB, 56 cores) •	Attastore HBase Master Server ccumulo OC Grafana
	Metrica Collector Kalifus Broker Krox Panger Jumppic Ranger Admin Activ Activity Analyzer Zoppelin Notobox Topical tecker Druka Coordinator Oreid tecker Oreid	atoway ty Explorer HST Server ruid Overlord Router
	mel2.htg3 cisco.com (125.4 GB, 56 cores Timeline Sarvice V1.5 Resource Manager Teneline Sarvice V2.5 Reader Zoofkeger Server Nenhau Storm U1 Infra Solr Instance Atlas Metadata Server	er HiveServer2 erver Accumado Master Spark2 History Server
	rhel3.hdp3.cisco.com (125.4 GB, 56 cores KamsNode ResourceManager ZooKe	per Server

4. Proceed to the Review page.

Figure 48 Enable ResourceManager HA Wizard - Review

Enable ResourceManager HA Wizard

Durrent ResourceManager: rhel2.hdp3.cisc Additional ResourceManager: rhel3.hdp3.cisc	to com 🛧 TO BE INSTALLED	
Review Configuration Changes. The following lists the configuration changes the	at will be made by the Wizard to enable ResourceManager HA. This information is for review only and is not editable.	
YARN		
yam resourcemanager ha enabled	8	
yam resourcemanager.ha.m-ids	im1,m2	
yam.resourcemanager.hostname.rm1	ihel2.hdp3.cisco.com	
	rhel2.hdp3.cisco.com.9088	
yam resourcemanager webapp address.m1		
yam resourcemanager webapp address rm1 yam resourcemanager webapp address rm2	rhel3.hdp3.cisco.com:8088	
yarn resourcemanager webapp address.m1 yarn resourcemanager webapp address.m2 yarn resourcemanager webapp https address	rhel3 hdp3 cisco com:8098 m1 rhel2.hdp3.cisco.com:8090	

5. The Configure Components page shows the progress bar as the Additional ResourceManager is being deployed.



Enable ResourceManager	HA Wizard	×
Get Started	Configure Components Please wait while ResourceManager HA is being deployed.	
 Review Configure Components 	 Stop Required Services Install Additional 35% ResourceManager Reconfigure YARN Reconfigure HDFS Start All Services 	
	COMPLETE	

6. Click COMPLETE when done.

2

It was observed that in certain circumstances, services might fail to restart. Click COMPLETE and restart the services in Ambari dashboard.

7. Verify the ResourceManger HA setup by clicking Services > YARN > SUMMARY tab. The Quick Links pane identifies Active and Standby ResourceManager.

♠ / Services / YARN / S	Summary				¢ 🛛	♣ 100 ■ ▲ admin • Cisco_HDP
SUMMARY HEATMAPS	CONFIGS METRICS					ACTIONS -
Summary					40	Quick Links
Components	 Started TIMELINE SERVICE V1 5 Started 	Started ACTIVE RESOURCEMANAGER	STANDBY RESOURCEMANAGER	Started TIMELINE SERVICE V2.0 READER		rhel2.hdp3.cisco.com (Active) ResourceManager UI ResourceManager JOS ResourceManager JMX
	YARN REGISTRY DNS 16/16 Started NODEMANAGERS	18 Installed YARN CLIENTS				rheið huðus rheið huðuð sisso com (Standby) ResourceManager Ul ResourceManager Jogs ResourceManager, JAX
	13m RESOURCEMANAGER UPTIME					Thread Stacks
	NODEMANAGERS STATUS 16 Active	0 Lost	O Unhealthy	0 Rebooted		

Figure 50 Services/YARN/Summary – Verify ResourceManger HA

HDP Post OS Deployment – Enable GPU Isolation and Scheduling

One of the prerequisites for HDP to enable GPU isolation and scheduling, is CUDA. CUDA itself also has prerequisites. The order of installation is as follows:

- Install CUDA prerequisites
- CUDA

This CVD incorporates Cisco UCS C480 ML M5 with 8 V100 GPUs; the following steps detail how to enable GPU as a Hadoop resource.

• Cisco UCS C480 ML M5 resource inventory as seen through Cisco UCS Manager

eneral Inventory Virtual Machines Hybrid Display	Installed Firmware SEL	ogs CIMC Sessio	ons VIF Paths	Power Contro	ol Monitor Health	Diagnostics Fa	aults Events FSM	Statistics Ter
fotherboard CIMC CPUs Coprocessor Cards	GPUs PCI Switch Mem	ry Adapters	HBAs NICs	ISCSI VNICs	Storage			
ry Advanced Filter + Export ⊕ Print								
lame	Ð				Model			Serial
Graphics Card 1	1				nVidia Volta V100-SX	/2 32GB		0323618047
Graphics Card 2	2				nVidia Volta V100-SX	//2 32GB		0323618047
Graphics Card 3	3				nVidia Volta V100-SX?	//2 32GB		0323618047
Graphics Card 4	4				nVidia Volta V100-SX	42 32GB		0323618047
Graphics Card 5	5				nVidia Volta V100-SX?	M2 32GB		0323618046
Graphics Card 6	6				nVidia Volta V100-SX	//2 32GB		0323618047
Graphics Card 7	7				nVidia Volta V100-SX	M2 32GB		0323618046
Graphics Card 8	8				nVidia Volta V100-SX	//2 32GB		0323618046
Dataila						****		
ID : 1						PCI Slot	GPU-3	
Expander Slot ID: NA						PID	UCSC-GPU-V100-SX	M2-32G
Is Supported : Yes						Vendor	: nVidia	
MadelVidia Valla V100, CVH2 2200						Secol	0323518047152	

Activate Status : Ready

Install the Prerequisites for CUDA

Running Version : 88.00.43.00.03[G503.0203.00.04

To install the prerequisites for CUDA, follow these steps:





These commands are run as root or sudo.

1. List GPUs and CPUs installed:

[root@rhel1 ~]# ansible nodeswithgpu -m shell -a "lspci | grep -i nvidia"

```
[root@rhel1 ~]# ansible nodeswithgpu -m shell -a "lspci | grep -i nvidia"
rhel16.hdp3.cisco.com | SUCCESS | rc=0 >>
5e:00.0 3D controller: NVIDIA Corporation GP100GL [Tesla P100 PCIe 16GB] (rev a1)
af:00.0 3D controller: NVIDIA Corporation GP100GL [Tesla P100 PCIe 16GB] (rev a1)
rhel15.hdp3.cisco.com | SUCCESS | rc=0 >>
5e:00.0 3D controller: NVIDIA Corporation GV100GL [Tesla V100 PCIe] (rev a1)
rhel14.hdp3.cisco.com | SUCCESS | rc=0 >>
5e:00.0 3D controller: NVIDIA Corporation GV100GL [Tesla V100 PCIe] (rev a1)
rhel17.hdp3.cisco.com | SUCCESS | rc=0 >>
5e:00.0 3D controller: NVIDIA Corporation GP100GL [Tesla P100 PCIe] (rev a1)
af:00.0 3D controller: NVIDIA Corporation GP100GL [Tesla P100 PCIe 16GB] (rev a1)
```

[root@rhel1 ~]# ansible nodeswithgpu -m shell -a "lscpu"

<pre>[root@rhel1 ~]# ansibl</pre>	e nodeswithgpu -m shell -a "lscpu"
rhel14.hdp3.cisco.com	SUCCESS rc=0 >>
Architecture:	x86_64
CPU op-mode(s):	32-bit, 64-bit
Byte Order:	Little Endian
CPU(s):	56
On-line CPU(s) list:	0-55
Thread(s) per core:	
Core(s) per socket:	14
Socket(s):	
NUMA node(s):	
Vendor ID:	GenuineIntel
CPU family:	
Model:	85
Model name:	Intel(R) Xeon(R) Gold 6132 CPU @ 2.60GHz
Stepping:	
CPU MHz:	2599.841
CPU max MHz:	2600.0000
CPU min MHz:	1000.0000
BogoMIPS:	5200.00
Virtualization:	VT-x
Lld cache:	32K
Lli cache:	32K
L2 cache:	1024K
L3 cache:	19712K
NUMA node0 CPU(s):	0-13,28-41
NUMA nodel CPU(s) ·	14-27.42-55

Install GCC

To install GCC, follow these steps:

1. Make sure gcc is installed in the system:

```
[root@rhel1 ~]# ansible nodeswithgpu -m shell -a "gcc --version"
```

[root@rhel1 ~]#
[root@rhel1 ~]# ansible nodeswithgpu -m shell -a "gccversion"
rhel16.hdp3.cisco.com SUCCESS rc=0 >>
gcc (GCC) 4.8.5 20150623 (Red Hat 4.8.5-28)
Copyright (C) 2015 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
rhel17.hdp3.cisco.com SUCCESS rc=0 >>
gcc (GCC) 4.8.5 20150623 (Red Hat 4.8.5-28)
Copyright (C) 2015 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
arrandy, not even for incommunity of frinces for a factored in the

Install Kernel Headers and Installation Packages

ansible nodeswithgpu -m shell -a "uname -r"

The CUDA Driver requires that the kernel headers and development packages for the running version of the kernel are installed at the time of the driver installation, as well as whenever the driver is rebuilt. For example, if your system is running kernel version 3.17.4-301, the 3.17.4-301 kernel headers and development packages must also be installed.

```
[root@rhel1 ~]# ansible nodeswithgpu -m shell -a "uname -r"
rhel16.hdp3.cisco.com | SUCCESS | rc=0 >>
3.10.0-862.14.4.el7.x86_64
rhel14.hdp3.cisco.com | SUCCESS | rc=0 >>
3.10.0-862.14.4.el7.x86_64
rhel17.hdp3.cisco.com | SUCCESS | rc=0 >>
3.10.0-862.14.4.el7.x86_64
rhel15.hdp3.cisco.com | SUCCESS | rc=0 >>
3.10.0-862.14.4.el7.x86_64
```

[root@rhell ~]# ansible nodeswithgpu -m yum -a "name=kernel-devel-\$(uname -r) state=present"



[root@rhel1 ~]# ansible nodeswithgpu -m yum -a "name=kernel-headers-\$(uname -r) state=present"

rhel20.hdp3.cisco.com SUCCESS => {
"changed": true,
"failed": false,
"msg": "",
"rc": 0,
"results": [
"Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-\n : manager\nResolving Dependenc
ies\n> Running transaction check\n> Package kernel-headers.x86_64 0:3.10.0-862.e17 will be updated\n> Package kernel-heade
rs.x86_64 0:3.10.0-862.14.4.e17 will be an update\n> Finished Dependency Resolution\n\nDependencies Resolved\n\n==================================
======================================
Size\n====================================
86_64 3.10.0-862.14.4.el7 rhel-7-server-rpms 7.1 M\n\nTransaction Summary\n====================================
======================================
ecause /usr/bin/applydeltarpm not installed.\nRunning transaction check\nRunning transaction test\nTransaction test succeeded\nRun
ning transaction\n Updating : kernel-headers-3.10.0-862.14.4.el7.x86_64 1/2 \n Cleanup : kernel-headers-
3.10.0-862.el7.x86_64 2/2 \n Verifying : kernel-headers-3.10.0-862.14.4.el7.x86_64 1/
2 \n Verifying : kernel-headers-3.10.0-862.el7.x86_64 2/2 \n\nUpdated:\n kernel-headers.x86_64 0:3.10.0
-862.14.4.el7 \n\nComplete!\n"

Install DKMS

The NVIDIA driver RPM packages depend on other external packages, such as DKMS. Those packages are only available on third-party repositories, such as <u>EPEL</u>.

To install DKMS, follow these steps:

http://rpmfind.net/linux/rpm2html/search.php?query=dkms for RHEL 7.x

RHEL 7.x http://rpmfind.net/linux/epel/7/x86_64/Packages/d/dkms-2.6.1-1.el7.noarch.rpm

#wget http://rpmfind.net/linux/epel/7/x86 64/Packages/d/dkms-2.6.1-1.el7.noarch.rpm

1. Copy dkms rpm to all the GPU servers:

[root@rhell ~]# ansible nodeswithgpu -m copy -a "src=/root/dkms-2.6.1-1.el7.noarch.rpm dest=/root/."

2. Install dkms with yum install:

[root@rhell ~]# ansible nodeswithgpu -m command -a "yum install -y /root/dkms-2.6.1-1.el7.noarch.rpm"

Install NVIDIA GPU Drivers

To install the NVIDIA GPU drivers, follow these steps:

- 1. Download this NVIDIA GPU driver from http://www.nvidia.com/Download/index.asp?lang=en-us
- 2. For the NVIDIA driver download, select the product type, Series, Product, OS, and CUDA toolkit.



For this deployment, select 9.2 for CUDA Toolkit.

- > C 🌲 https://www.nvidia.com/E)ownload/index.aspx?lang=en-us				_		_
Apps D Taplt! D Online Certificatio	N.J. Department o V Hydroponic,	Farm 🥋 HDP 🤱 A	Amazon.com: Doli.	🗋 Guides	ee car lift pdf T	N Using PXE Boot T	FREE SHIPPING
© NVIDIA,∨							
	PLATFORMS > DEVELOPER	COMMUNITY >	SHOP	DRIVERS >	SUPPORT	ABOUT NVIDIA >	
	DOWNLOAD DRIVERS						
	NVIDIA > Download Drivers						
	GEFORCE	NVIDIA Drive	r Downloa	ads			
	RTX [°] 2070	Option 1: Manually find	d drivers for my N	VIDIA products.			Help
	GRAPHICS	Product Type:	Tesla		+		
	REINVENTED	Product Series:	V-Series		¢)		
		Product:	Tesla V100		\$		
	GEFORNE RTX	Operating System:	Linux 64-bit RHE	17	\$		
		CUDA Toolkit:	9.2		\$		
		Language:	English (US)		\$		SEARCH
		Ontion 2: Automaticall	u find driver for	my NVIDIA produc	**		

3. Click SEARCH. The selected driver is shown below.

← → C 🔒 https://www.nvidia.com/Do	ownload/driverResults.aspx/136948/er	1-us			
👯 Apps 🗋 Taplt! 🗋 Online Certificatio 🗧	🚸 N.J. Department o 🔽 Hydroponic,	Farm 🏦 HDP 🤱 Amazon.com: Do	li 🗋 Guides 🚥 car lift pd	f TN Using PXE Boot T	FREE SHIPPING
	PLATFORMS > DEVELOPERS	G COMMUNITY > SHOP	DRIVERS > SUPPOR	T ABOUT NVIDIA >	
	DOWNLOAD DRIVERS				
	NVIDIA > DRIVERS > TESLA DRIVER FOR LINUX	RHEL 7			
	THE NEW NVIDIA QUADRO	TESLA DRIVER FOR LI	NUX RHEL 7		
	Unmatched Creative Freedom.	Version: 396.44 Release Date: 2018.8.6 Operating System: Linux 64-bit RHEL7 CUDA Toolkit: 9.2 Language: English (US) File Size: 94.25 MB			
	NVIDIA.	DOWNLOAD			

4. Click DOWNLOAD. The download link can be captured by right-clicking AGREE & DOWNLOAD as shown below.



[root@rhel1 ~]# wget http://us.download.nvidia.com/tesla/396.44/nvidia-diag-driver-local-repo-rhel7-396.44-1.0-1.x86 64.rpm

5. Copy the .rpm file in all the GPU nodes as shown below.

```
[root@rhel1 ~]# ansible nodeswithgpu -m copy -a "src=/root/nvidia-diag-driver-local-repo-rhel7-396.44-
1.0-1.x86 64.rpm dest=/root/."
```

6. Install the driver by running the following command:

Install CUDA

To install CUDA, follow these steps:

1. Download CUDA 9.2.

TensorFlow needs CUDA; make sure that the version of CUDA is supported by TensorFlow before installing CUDA. Earlier versions of CUDA are here: <u>https://developer.nvidia.com/cuda-toolkit-archive.</u>

The latest version of CUDA is available here:

<u>https://developer.nvidia.com/cuda-</u> <u>downloads?target_os=Linux&target_arch=x86_64&target_distro=RHEL&target_version=7&target_type=rpmlocal</u>



[root@rhel1 ~]# wget https://developer.nvidia.com/compute/cuda/9.2/Prod2/local_installers/cuda-reporhel7-9-2-local-9.2.148-1.x86 64 2. Copy .rpm file to all GPU nodes as follows:

```
[root@rhell ~]# ansible nodeswithgpu -m copy -a "src=/root/cuda-repo-rhel7-9-2-local-9.2.148-1.x86_64.rpm
dest=/root/."
```

3. Install CUDA in all GPU nodes by running the following set of commands:

```
[root@rhel1 ~]# ansible nodeswithgpu -m shell -a "rpm -i cuda-repo-rhel7-9-2-local-9.2.148-1.x86_64.rpm"
[root@rhel1 ~]# ansible nodeswithgpu -m shell -a "yum clean all"
[root@rhel1 ~]# ansible nodeswithgpu -m shell -a "yum -y install cuda"
```

Download and Setup NVIDIA CUDA Deep Neural Network library (cuDNN)

Download cuDNN 9.2

- 1. Download cuDNN from https://developer.nvidia.com/cudnn for the same CUDA version.
- 2. Copy cuNN into all the GPU servers.

https://developer.nvidia.com/rdp/cudnn-archive

This step may require to join the NVIDIA developer community.

3. Run the following Ansible commands in all GPU nodes to setup cuDNN:

```
[root@rhell ~]# ansible nodeswithgpu -m copy -a "src=/root/cudnn-9.2-linux-x64-v7.1.tgz dest=/root/."
[root@rhell ~]# ansible nodeswithgpu -m shell -a "tar -xzvf cudnn-9.2-linux-x64-v7.1.tgz"
[root@rhell ~]# ansible nodeswithgpu -m shell -a "cp /root/cuda/include/cudnn.h /usr/local/cuda-
9.2/include"
[root@rhell ~]# ansible nodeswithgpu -m shell -a "cp /root/cuda/lib64/libcudnn* /usr/local/cuda-
9.2/lib64"
[root@rhell ~]# ansible nodeswithgpu -m shell -a "cp /root/cuda/lib64/libcudnn* /usr/local/cuda-
9.2/lib64"
```

Post Installation Steps

- 1. Add CUDA in PATH and LD_LIBRARY_PATH variable in all the GPU nodes.
- 2. The PATH and LD_LIBRARY_PATH variable need to include /usr/local/<cuda>/bin:

```
export PATH=/usr/local/cuda-9.2/bin${PATH:+:${PATH}}
export LD_LIBRARY_PATH=/usr/local/cuda-9.2/lib64${LD_LIBRARY_PATH:+:${LD_LIBRARY_PATH}}
```

3. Reboot the GPU nodes:

ansible nodeswithgpu -a "/sbin/reboot"



The SSH and Ansible connection will disconnect with this command.



For more information, go to: <u>https://docs.nvidia.com/cuda/cuda-installation-guide-linux/index.html#post-</u>installation-actions.

Verify Drivers

To verify the drivers have been installed, run the following commands in all GPU nodes as shown below:

```
[root@rhel1 ~]# ansible nodeswithgpu -a "nvidia-smi"
Or on specific node
[root@rhel1 ~]# ansible rhel20.hdp3.cisco.com -a "nvidia-smi"
```

<pre>[root@rhel1 ~]# [root@rhel1 ~]# ansi rhel20.hdp3.cisco.co Mon Jan 7 20:33:44</pre>	ble rhel20.hd om SUCCESS 2019	hp3.cisco.com −a "nvidia rc=0 >>	a-smi"	
NVIDIA-SMI 396.44		Driver Version: 396.	. 4 4	
GPU Name I Fan Temp Perf I	Persistence-M Pwr:Usage/Cap	Bus-Id Disp.A Memory-Usage	Volatile GPU-Util	Uncorr. ECC Compute M.
0 Tesla V100-SX N/A 32C P0	M2 Off 42W / 300W	00000000:1B:00.0 off 11MiB / 32510MiB	0%	0 Default
	M2 Off 43W / 300W	00000000:1C:00.0 off 11MiB / 32510MiB	0%	0 Default
	M2 Off 42W / 300W	00000000:42:00.0 off 11MiB / 32510MiB	0%	0 Default
	M2 off 44W / 300W	00000000:43:00.0 off 11MiB / 32510MiB	0%	0 Default
	M2 off 44w / 300w	00000000:89:00.0 off 11MiB / 32510MiB	0%	0 Default
' 5 Tesla V100-SX N/A 36C P0	M2 off 44w / 300w	00000000:8A:00.0 off 11MiB / 32510MiB	0%	0 Default
' 6 Tesla V100-S⊁ N/A 34C P0	M2 off 42w / 300w	00000000:B2:00.0 off 11MiB / 32510MiB	0%	0 Default
' 7 Tesla V100-SX N/A 35C P0	M2 off 43W / 300W	0000000:B3:00.0 off 11MiB / 32510MiB	0%	0 Default
· +				
Processes: GPU PID 1	Type Process	name		GPU Memory Usage
No running proces	ses found			

[root@rhell ~]# ansible nodeswithgpu -m shell -a "/usr/local/cuda-9.2/bin/nvcc --version"

[root@rhell ~]# ansible nodeswithgpu -m shell -a "/usr/local/cuda-9.2/bin/nvcc --version"
rhel16.hdp3.cisco.com | SUCCESS | rc=0 >>
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2018 NVIDIA Corporation
Built on Tue_Jun_12_23:07:04_CDT_2018
Cuda compilation tools, release 9.2, V9.2.148
rhel14.hdp3.cisco.com | SUCCESS | rc=0 >>
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2018 NVIDIA Corporation
Built on Tue_Jun_12_23:07:04_CDT_2018
Cuda compilet driver
Copyright (c) 2005-2018 NVIDIA Corporation
Built on Tue_Jun_12_23:07:04_CDT_2018
Cuda compilation tools, release 9.2, V9.2.148

[root@rhell ~]# ansible nodeswithgpu -m shell -a "export NVIDIA DRIVER VERSION=396.44"

Install and Configure Docker

Prerequisites

Docker Version

Docker only supports systems with modern kernel as it is highly dependent on Linux OS kernel version and its capabilities.

YARN requires Docker should be setup and installed in all NodeManger hosts where Docker containers will run. It is important to consider these recommendations before installing Docker to be used by YARN.

Docker 1.12.5 is required at a minimum. However, it is recommended to use the latest version.

Storage Driver

It was observed through testing that device mapper using LVM is generally stable. However, high writes to the container root file system, device mapper exhibit panics. Overlay and Overlay2 perform significantly better than device mapper and recommended. In this reference architecture Overlay2 has been used.

Networking

YARN has support for running Docker containers on a user specified network, however, it does not manage the Docker networks. Administrators are expected to create the networks prior to running the containers. Node labels can be used to isolate particular networks. It is vital to read and understand the Docker networking documentation. Swarm based options are not recommended, however, overlay networks can be used if setup using an external store, such as etcd.

YARN will ask Docker for the networking details, such as IP address and hostname. As a result, all networking types are supported. Set the environment variable YARN_CONTAINER_RUNTIME_DOCKER_CONTAINER_NETWORK to specify the network to use.

Host networking is only recommended for testing. If the network where the NodeManagers are running has a sufficient number of IP addresses. The bridge networking with --fixed-cidr option works well. Each NodeManager is allocated a small portion of the larger IP space, and then allocates those IP addresses to containers.

To use an administrator defined network, add the network to docker.allowed.networks in containerexecutor.cfg and yarn.nodemanger.runtime.linux.docker.allowed-container-networks in yarn-site.xml .



Setting up Docker Networking is beyond the scope of this CVD. This CVD focuses on standalone container for Tensor-Flow on YARN. Hadoop YARN is adding support for distributed TensorFlow applications and will be added as addendum when it is supported.

Image Management

Images can be preloaded on all NodeManager hosts or they can be implicitly pulled at runtime if they are available in a public Docker registry, such as Docker hub. If the image does not exist on the NodeManager and cannot be pulled, the container will fail.



For AI framework specific images, it is recommend using Docker images from NG Cloud, which is described in subsequent sections of this document.

It is also recommended to have a private Docker image repository. This CVD details the setup of a private registry for simplicity but doesn't go into details on a registry setup with high availability and is provided only as an example.

Install Docker

To install Docker, follow these steps:

1. Uninstall any previous version of Docker completely by running the following command:

yum remove docker docker-client docker-client-latest docker-common docker-latest docker-latestlogrotate docker-logrotate docker-selinux docker-engine-selinux docker-engine

2. Remove any existing Docker repository:

rm /etc/yum.repos.d/docker*.repo

3. Docker has been released as part of Extra channel in Red Hat Enterprise Linux. When the extra channel has been enabled, docker packages can be installed. To register and subscribe a system to Red Hat customer portal, attached the required pool, and enable the extras channel, run the following commands:

```
# subscription-manager register --username <userid> --password <password>
# subscription-manager list --available
# subscription-manager attach --pool=<pool id>
# subscription-manager repos --disable='*'
# subscription-manager repos --enable="rhel-7-server-extras-rpms"
```

4. Install Docker by running the following command:

ansible datanodes -m yum -a "name=docker state=present"

5. Start the Docker service:

```
# ansible datanodes -a ``systemctl start docker"
# anaible datanodes -a ``systemctl enable docker"
```

6. Verify Docker by running the following command:

ansible datanodes -a "docker info"

Install nvidia-docker in GPU Nodes

To install nvidia-docker in GPU nodes, follow these steps:

- 1. In order for the Docker container to see GPU as a resource, you need to install nvidia-docker v1 plugin.
- 2. Add the package repositories:

```
# distribution=$(. /etc/os-release;echo $ID$VERSION_ID)
# ansible nodeswithgpu -m shell -a "curl -s -L https://nvidia.github.io/nvidia-
docker/$distribution/nvidia-docker.repo | sudo tee /etc/yum.repos.d/nvidia-docker.repo"
```

3. Install nvidia-docker v1:

```
# ansible nodeswithgpu -m command -a "yum install -y nvidia-docker"
```

nvidia-docker2 is not supported by HDP 3.0. Make sure you install nvidia-docker.

4. Start the nvidia-docker service:

```
# ansible nodeswithgpu -m shell -a "systemctl start nvidia-docker"
# ansible nodeswithgpu -m shell -a "systemctl enable nvidia-docker"
```

5. Test nvidia-smi with the latest CUDA image. SSH to the node with GPU and run the following command:

nvidia-docker run --rm nvidia/cuda:9.2-base nvidia-smi

[root@rhel16 ~]# [root@rhel16 ~]# nvidia-docker Mon Dec 17 23:05:38 2018	runrm nvidia/cuda:9.2-base	nvidia-smi
NVIDIA-SMI 396.44	Driver Version: 396.44	
GPU Name Persistence Fan Temp Perf Pwr:Usage/C	-M Bus-Id Disp.A Vol ap Memory-Usage GPU	atile Uncorr. ECC J-Util Compute M.
0 Tesla P100-PCIE Off N/A 28C P0 26W / 250	00000000:5E:00.0 Off W 10MiB / 16280MiB	0 0% Default
1 Tesla P100-PCIE Off N/A 26C P0 24W / 250	00000000:AF:00.0 Off W 10MiB / 16280MiB	0 0% Default
·		
Processes: GPU PID Type Proc	ess name	GPU Memory Usage
No running processes found		
<pre>froot@rhel16 ~]#</pre>		+

ø

For more information, go to: https://github.com/NVIDIA/nvidia-docker

6. Verify that the /var/lib/nvidia-docker/volumes/nvidia_driver/\$NVIDIA_DRIVER_VERSION/ directory was created:

ls /var/lib/nvidia-docker/volumes/nvidia_driver

```
[root@rhel15 ~]#
[root@rhel15 ~]# ls /var/lib/nvidia-docker/volumes/nvidia_driver
396.44
[root@rhel15 ~]#
```

7. Run the following command to verify the nvidia-docker plugin:

```
[root@rhel15 lib64]# docker volume ls
[root@rhel15 lib64]# docker inspect volume nvidia_driver_396.44
```

```
t@rhe115
             11064
[root@rhel15 lib64]# docker volume ls
DRIVER
                     VOLUME NAME
nvidia-docker
                     nvidia_driver_396.44
[root@rhel15 lib64]#
[root@rhel15 lib64]#
[root@rhel15 lib64]#
root@rhel15 lib64]# docker volume inspect nvidia driver 396.44
        "CreatedAt": "0001-01-01T00:00:002",
        "Driver": "nvidia-docker",
        "Labels": null,
"Mountpoint": "/var/lib/nvidia-docker/volumes/nvidia_driver/396.44",
        "Name": "nvidia driver 396.44",
        "Options": null,
        "Scope": "local"
```

8. Use the following Docker command to verify that HDP can access the GPU:

```
# NVIDIA_DRIVER_VERSION=396.44
# docker run --device=/dev/nvidiactl --device=/dev/nvidia-uvm --device=/dev/nvidia0 -v /var/lib/nvidia-
docker/volumes/nvidia_driver/$NVIDIA_DRIVER_VERSION/:/usr/local/nvidia/ -it nvidia/cuda:9.2-base
/usr/local/nvidia/bin/nvidia-smi
```

[root@rhel15 bin]# docker run dia_driver/\$NVIDIA_DRIVER_VERSIO Mon Dec 17 23:39:23 2018	-device=/dev/nvidiactl N/:/usr/local/nvidia/ -i	-device=/dev/nvidia-uv t nvidia/cuda:9.2-base	ndevice=/dev/nvidia0 -v /var/lib /usr/local/nvidia/bin/nvidia-smi
NVIDIA-SMI 396.44	Driver Version: 396.	44	
GPU Name Persistence-M Fan Temp Perf Pwr:Usage/Cap	Bus-Id Disp.A Memory-Usage	Volatile Uncorr. ECC GPU-Util Compute M.	
0 Tesla V100-PCIE Off N/A 33C P0 26W / 250W	00000000:5E:00.0 Off 11MiB / 16160MiB	Off 0% Default	
+		GDII Memory	-
GPU PID Type Process	5 name	Usage	
No running processes found			
[root@rhel15 bin]#			

ø

On a multi-GPU server, the output of this command will show exactly one GPU. This is because we have run this sample docker container with only one device (/dev/nvidia0).

Configure Docker

After Docker and nvidia-docker are installed, the following process is the minimum recommended configuration. To configure Docker, follow these steps:

Configuring the storage driver is not included.

1. Edit /etc/docker/daemon.json and add the following options:

```
"live-restore" : true,
"debug" : true
}
```



Live-restore keeps the container alive during daemon downtime.

2. Restart Docker service:

```
ansible datanodes -m command -a "systemctl daemon-reload" ansible datanodes -m command -a "systemctl restart docker"
```

Configure YARN to Running Docker Containers

YARN provides isolation through the use of cgroups. Docker also has cgroup management built in. If isolation through cgroups if desired, the only recommended solution is to use YARN's cgroup management at this time. YARN will create the cgroup hierarchy and set the --cgroup-parent flag when launching the container.

The cgroupdriver must be set to cgroupfs in all datanodes where Docker is running. Ensure that Docker is running using the --exec-opt native.cgroupdriver=cgroupfs docker daemon option.

To configure YARN to run Docker containers, follow these steps:

vi /usr/lib/systemd/system/docker.service

1. Find and fix the cgroupdriver. If it is not there, it should be added.

```
--exec-opt native.cgroupdriver=cgroupfs \
```

```
[root@rhel16 ~]#
[root@rhel16 ~]# cat /usr/lib/systemd/system/docker.service | grep cgroupfs
--exec-opt native.cgroupdriver=cgroupfs \
[root@rhel16 ~]#
```

The commands (above) should be configured on all the nodes where Docker is installed, on the datanodes.

2. The version of Docker may include oci-hooks that expect to use the systemd cgroup driver. Search for oci on your system and remove these files. For example:

```
[root@rhell ansible]# ansible datanodes -m shell -a " [ -f /usr/libexec/oci/hooks.d/oci-systemd-hook ] &&
echo 'File exist' || echo 'File does not exist'"
```

3. If it exists, run the following:

[root@rhel1 ansible]# ansible datanodes -m shell -a "rm -f /usr/libexec/oci/hooks.d/oci-systemd-hook"

```
ansible datanodes -m shell -a " [ -f /usr/libexec/oci/hooks.d/oci-register-machine] && echo 'File exist' || echo 'File does not exist'"
```

4. If it exists, remove the file:

ansible datanodes -m shell -a "rm -f /usr/libexec/oci/hooks.d/oci-register-machine"

Enable Cgroups

To set up the CGroup hierarchy, run the following command:

1. Create the following script in admin node:

```
[root@rhell ~]# vi hadoop-yarn.sh
mkdir -p /sys/fs/cgroup/cpu/hadoop-yarn
chown -R yarn /sys/fs/cgroup/cpu/hadoop-yarn
mkdir -p /sys/fs/cgroup/memory/hadoop-yarn
chown -R yarn /sys/fs/cgroup/blkio/hadoop-yarn
chown -R yarn /sys/fs/cgroup/blkio/hadoop-yarn
mkdir -p /sys/fs/cgroup/net_cls/hadoop-yarn
mkdir -p /sys/fs/cgroup/net_cls/hadoop-yarn
chown -R yarn /sys/fs/cgroup/devices/hadoop-yarn
chown -R yarn /sys/fs/cgroup/devices/hadoop-yarn
chown -R yarn /sys/fs/cgroup/devices/hadoop-yarn
```

2. Copy the script in all datanodes and run the script as shown below:

```
# ansible datanodes -m copy -a "src=/root/hadoop-yarn.sh dest=/root/."
# ansible datanodes -m file -a "dest=/root/hadoop-yarn.sh mode=755"
# ansible datanodes -m shell -a "/root/hadoop-yarn.sh"
```



It is important to note that CGroup hierarchy MUST be created on every reboot of NodeManager node(s), otherwise the YARN NodeManager service will not start. It is recommend to be auto run on every system reboot and nodemanager reboot.

- 3. To enable cgroups on an Ambari cluster, select YARN > CONFIGS on the Ambari dashboard, then click CPU Isolation under CPU. Click Save.
- Enable Docker Runtime by selecting YARN > CONFIGS > SETTINGS on the Ambari dashboard as shown below.

c O
c 0
/ •

5. Set the following properties for Advanced yarn-site in Ambari.

Table 12	Advanced	yarn-site Properties	
----------	----------	----------------------	--

Properties	Value
yarn.nodemanager.runtime.linux.allowed-runtimes	default,docker
yarn.nodemanager.runtime.linux.docker.capabilities	CHOWN,DAC_OVERRIDE,FSETID,FOWNER,MKNOD,NET_RAW,SETGID,S ETUID,
	SETFCAP,SETPCAP,NET_BIND_SERVICE,SYS_CHROOT,KILL,AUDIT_WRIT E
yarn.nodemanager.runtime.linux.docker.privilegedcontai ners.	false
allowed	
yarn.nodemanager.runtime.linux.docker.allowed- containernetworks	host,bridge
yarn.nodemanager.runtime.linux.docker.default- containernetwork	bridge

yarn.nodemanager.runtime.linux.allowed- runtimes	default,docker
yarn.nodemanager.runtime.linux.docker. allowed-container-networks	host,bridge
yarn.nodemanager.runtime.linux.docker. capabilities	CHOWN,DAC_OVERRIDE,FSETID,FOWNER,MKNOD,NET_RAW,SETGID,SETUID,SETFCAP, SETPCAP,NET_BIND_SERVICE,SYS_CHROOT,KILL,AUDIT_WRITE
	▲ ○ C
yarn.nodemanager.runtime.linux.docker. default-container-network	bridge

6. Configure the following isolation properties in YARN. In Ambari, click YARN > CONFIGS > ADVANCED. Properties can also be configured in the /etc/hadoop/conf/yarn-site.xml on the ResourceManager and NodeManager nodes. Configure and verify properties listed in Table 13 in Ambari so that changes get applied to all the respective nodes.

Property	Value
yarn.nodemanager.container-executor.class	org.apache.hadoop.yarn.server.nodemanager.LinuxContainerExecutor
yarn.nodemanager.linux-container- executor.cgroups.hierarchy	/hadoop-yarn
yarn.nodemanager.linux-container- executor.cgroups.mount	false
yarn.nodemanager.linux-container- executor.cgroups.mount-path	/sys/fs/cgroup
yarn.nodemanager.linux-container- executor.cgroups.strict-resource-usage	false
yarn.nodemanager.linux-container-executor.group	hadoop
yarn.nodemanager.linux-container- executor.resources-handler.class	org.apache.hadoop.yarn.server.nodemanager.util.CgroupsLCEResourcesHandler

Table 13 YARN – Isolation Properties

solation	
yarn.nodemanager.container-executor.cla	org.apache.hadoop.yarn.server.nodemanager.LinuxContainerExecutor
	org.apache.hadoop.yarn.server.nodemanager.LinuxContainerExecutor
yarn.nodemanager.linux-container-execu cgroups.hierarchy	tor. /hadoop-yarn
yarn.nodemanager.linux-container-execu cgroups.mount	tor. false
	false
yarn.nodemanager.linux-container-execu cgroups.mount-path 🔒	tor. /sys/fs/cgroup
	/sys/fs/cgroup
yarn.nodemanager.linux-container-execu cgroups.strict-resource-usage	tor. false
yarn.nodemanager.linux-container-execu group	tor. hadoop
	hadoop
yarn.nodemanager.linux-container-execu resources-handler.class	org.apache.hadoop.yarn.server.nodemanager.util.CgroupsLCEResourcesHandler
	org.apache.hadoop.yarn.server.nodemanager.util.CgroupsLCEResourcesHandler

To leverage YARN cgroup support, the NodeManager must be configured to use LinuxContainerExecutor. The Docker YARN integration also requires this container executor.

7. Configure the following properties for Container Executor in YARN. Some properties are pre-set as per YARN recommendation.

Properties	Values
Docker Allowed Read-only Mounts	/sys/fs/cgroup
Docker Binary	/usr/bin/docker
Minimum user ID for submitting job	50
Yarn CGroup Hierarchy	/hadoop-yarn

Table 14 YARN – Container Executor Properties

∕≏⊾

8. The administrator must define the volume whitelist in container-executor.cfg by setting docker.allowed.romounts and docker.allowed.rw-mounts to the list of parent directories that are allowed to be mounted. The application submitter requests the required volumes at application submission time using the YARN CONTAINER RUNTIME DOCKER MOUNTS environment variable.

Be careful when enabling this feature. Enabling access to directories such as, but not limited to, /, /etc, /run, or /home is not advisable and can result in containers negatively impacting the host or leaking sensitive information.

Container Executor		
CGroup Root Path		
	/sys/fs/cgroup	s
Docker Allowed Devices		
	regex:^/dev/nvidia.*\$	s
Docker Allowed Read-only Mounts	/sys/fs/cgroup	
	/sys/fs/cgroup,regex:^nvidia_driver*\$	s
Docker Allowed Read-write Mounts		
Docker Allowed Volume-drivers		
	nvidia-docker	s
Docker Binary	/usr/bin/docker	
Enable Launching Privileged Containers	0 C	
Minimum user ID for submitting job	50	
Yarn CGroup Hierarchy	/hadoop-yarn	

/hadoop/yarn/local is a default Docker ro-mount. However, these are not mounted by default but are allowed to be mounted

CPU jobs are constrained with CPU scheduling and cgroups enabled, but by default these are flexible limits. If spare CPU cycles are available, containers are allowed to exceed the CPU limits set for them. With flexible limits, the amount of CPU resources available for containers to use can vary based on cluster usage; the amount of CPU available in the cluster at any given time.

You can use cgroups to set strict limits on CPU usage. When strict limits are enabled, each process receives only the amount of CPU resources it requests. With strict limits, a CPU process will receive the same amount of cluster resources every time it runs.

Strict limits are not enabled (set to false) by default for property yarn.nodemanager.linux-containerexecutor.cgroups.strict-resourceusage as shown above.



Regardless of whether this property is true or false, at no point will the total container CPU usage exceed the limit set in yarn.nodemanager.resource.percentage-physical-cpu-limit.

9. Set the Percentage of CPU used by YARN. Set the percentage of CPU that can be allocated for YARN containers. In most cases, the default value of 100 percent should be used. If you have another process that needs to run on a node that also requires CPU resources, you can lower the percentage of CPU allocated to YARN to free up resources for the other process.



10. Click SAVE. Restart All Affected Services as shown below.

UMMARY HEATMAPS CONFIGS METRICS			ACTIONS
$\mathcal Z$ Restart Required: 36 Components on 17 Hosts			RESTART -
			Restart All Affected
Version: 104 -	Config Group	Default (13)	nestart NoueManager

6

These steps (above) configure the YARN Node Manager to run LinuxContainerExecutor in non-secure mode, just for demonstration purposes, so that all Docker containers scheduled by YARN will run as a 'nobody' user. Kerberized cluster with cgroups enabled is recommended for production.

Run Docker on YARN Using the YARN Service API

To deploy the web server using YARN service API, follow these steps:

1. Create the following Yarn file and save it to /tmp/httpdserver.json.

```
[root@rhel1 tmp]# cat httpdserver.json
{
    "name":"httpd-service",
    "version":"1.0.0",
    "lifetime":"3600",
    "configuration":{
        "properties":{
            "docker.network":"bridge"
        }
}
```

Solution Design

```
},
   "components":[
      {
         "name":"httpd",
         "number_of_containers":1,
         "artifact":{
            "id":"centos/httpd-24-centos7:latest",
            "type":"DOCKER"
         },
         "launch command":"/usr/bin/run-httpd",
         "resource":{
            "cpus":1,
            "memory":"1024"
         },
         "configuration":{
            "files":[
               {
                   "type":"TEMPLATE",
                  "dest_file":"/var/www/html/index.html",
                   "properties":{
                      "content":"<html><header><title>Title</title></header><body>Hello from
${COMPONENT INSTANCE NAME}!</body></html>"
                  }
               }
            ]
         }
      }
  ]
}
[root@rhel1 tmp]#
```

2. Submit the HTTP application using the YARN CLI:

yarn app -launch my-httpd httpdserver.json

```
[root@rhel1 tmp]# yarn app -launch my-httpd httpdserver.json
18/11/07 20:09:52 INFO client.RMProxy: Connecting to ResourceManager at rhel1.hdp3.cisco.com/10.16.1.
31:8050
18/11/07 20:09:52 INFO client.AHSProxy: Connecting to Application History server at rhel2.hdp3.cisco.
com/10.16.1.32:10200
18/11/07 20:09:52 INFO client.RMProxy: Connecting to ResourceManager at rhel1.hdp3.cisco.com/10.16.1.
31:8050
18/11/07 20:09:52 INFO client.AHSProxy: Connecting to Application History server at rhel2.hdp3.cisco.
com/10.16.1.32:10200
18/11/07 20:09:52 INFO client.AHSProxy: Connecting to Application History server at rhel2.hdp3.cisco.
com/10.16.1.32:10200
18/11/07 20:09:52 INFO client.ApjServiceClient: Loading service definition from local FS: /root/tmp/h
ttpdserver.json
18/11/07 20:09:52 INFO util.log: Logging initialized @1351ms
18/11/07 20:09:53 INFO client.ApjServiceClient: Application ID: application_1541650078908_0002
[root@rhel1 tmp]#
[root@rhel1 tmp]#
```

3. Get the IP address httpd container and the node where it is scheduled by running the following command:

[root@rhel1 tmp]# yarn app -status my-httpd



4. Verify the application through YARN ResourceManager web UI:

Пере	Cluster Ov	erview	Queues	Applications	Se	rvices Flow Activit	ty Nodes Tool	5						L	ogged in a	S:
Home / Applicatio	ns															
Day of Count				Ct												
Reg V Search				Search											-	
User (3)		Applica	tion ID		Å Ŧ	Application Type $\frac{A}{V}$	Application Name	User	Å Ŧ	State	Å V	Queue	Å	Progress	5	A.
✓ root✓ hdfs	11 11	applica	tion_154165	0078908_0002		yam-service	my-httpd	root		• Runni	ng	default			00%	

5. SSH to rhel7 and using curl, perform a GET request to the HTTP server on port 8080 using the IP address obtained in previous step:

curl <u>http://172.17.0.2:8080</u>

```
[root@rhel7 ~]#
[root@rhel7 ~]#
[root@rhel7 ~]# curl http://172.17.0.2:8080
<html><header><title>Title</title></header><body>Hello from httpd-0!</body></html>[root@rhel7 ~]#
[root@rhel7 ~]#
[root@rhel7 ~]#
```

6. Run the following command to verify if Docker container is provisioned:

docker ps -a



Run Docker Container with GPU on YARN

A GPU is a specialized processor that can be used to accelerate highly parallelized computationally-intensive workloads. Because of their computational power, GPUs have been found to be particularly well-suited to <u>deep learning workloads</u>. Ideally, CPUs and GPUs should be used in tandem for data engineering and data science workloads. A typical machine learning workflow involves data preparation, model training, model scoring, and model fitting. You can use existing general-purpose CPUs for each stage of the workflow, and optionally accelerate the math-intensive steps with the selective application of special-purpose GPUs. For example, GPUs allow you to accelerate model fitting using frameworks such as <u>TensorFlow</u>, Caffe, <u>PyTorch</u>, <u>Keras</u>, <u>MXNet</u>, and <u>Microsoft Cognitive Toolkit (CNTK)</u>.

By enabling GPU support, data scientists can share GPU resources available on HDP nodes. Users can request a specific number of GPU instances, up to the total number available on a node, which are then allocated to the running session or job for the duration of the run.

Prerequisites

- Currently, only NVIDIA GPUs are supported by YARN
- NVIDIA drivers must be pre-installed in YARN node managers

When Docker is used as container runtime context, nvidia-docker 1.0 needs to be installed (the current supported version is YARN for nvidia-docker).

Enable GPU through Ambari

This section provides instructions about how to enable GPU support in HDP through Ambari.

If you have HDP cluster where datanodes with GPU installed and datanodes without GPU installed, you must create host groups. Enabling GPU in Ambari for non-GPU nodes will complain on restarting NodeManager service. Therefore, it is important to separate out the GPU nodes configuration by creating host group for GPU nodes.

Ambari initially assigns all hosts in your cluster to one default configuration group for each service you install. For example, after deploying a three-node cluster with default configuration settings, each host belongs to one configuration group that has default configuration settings for the HDFS service.

Create a New Host Configuration Group

To create host group, follow these steps:

1. Click a service name, for example, here we need to create host group for YARN. Click YARN > CONFIGS.

Config Group	Default (17) -	Filter	•
	Manage Config Grou	ips	
	Default (17)		

2. In Configs, click Manage Config Groups.

Default (17)	rhel1.hdp3.cisco.com						
	rhel10.hdp3.cisco.com						
	rhel11.hdp3.cisco.com						
	rhel12.hdp3.cisco.com						
	rhel13.hdp3.cisco.com						
	rhel14.hdp3.cisco.com						
	rhel15.hdp3.cisco.com						
	rhel16.hdp3.cisco.com						
	rhel17.hdp3.cisco.com						
	rhel2.hdp3.cisco.com						
	rhel3.hdp3.cisco.com						
	rhel4.hdp3.cisco.com						
+ - 0 -	+						
	Overrides 0 properties						
	Description Default cluster level MapReduce2 configuration						

3. Click + icon on the left pane to create new configuration group. Create New Configuration Group window will pop-up as shown below. Provide a name to the configuration group. For example, GPU-Nodes

CANCEL

Manage Spark	Create New	Configuration Grou	р	×	×
You can apply differe belonging to a Spark	Name: Description:	GPU-Nodes			embership. Hosts ration Group.
Default (17)			CAN	CEL OK	
		rhel16.hdp3.cisco. rhel17.hdp3.cisco.c rhel2.hdp3.cisco.c rhel3.hdp3.cisco.cc rhel3.hdp3.cisco.cc	com com om om		
+	- 0 -	Overrides Description	0 properties Default cluster level Spark2 o	configuration	
				CAN	ICEL SAVE

4. Click the newly created configuration group in the left pane and click + icon to add hosts to this group as shown below. Select hosts that has GPU and leave the non-GPU nodes in default configuration group.

Default (17) GPU-Nodes (0)		
		nor
		Add bosts to selected Configuration Group
+ - \$.		
	Overrides 0 properties	
	Description	

5. Click the check box on the hosts that have GPU installed as shown below. Click OK when done.

Sel	ect Configuration Group Hosts	×			
Select hosts that should belong to this GPU-Nodes Configuration Group. All hosts belonging to this group will have the same set of configurations.					
4 out	of 17 hosts selected	Filter 🔹 COMPONENTS 🕶			
	Host	IP Address			
	rhel1.hdp3.cisco.com	10.16.1.31			
	rhel10.hdp3.cisco.com	10.16.1.40			
	rhel11.hdp3.cisco.com	10.16.1.41			
	rhel12.hdp3.cisco.com	10.16.1.42			
	rhel13.hdp3.cisco.com	10.16.1.43			
۵	rhel14.hdp3.cisco.com	10.16.1.44			
۵	rhel15.hdp3.cisco.com	10.16.1.45			
۲	rhel16.hdp3.cisco.com	10.16.1.46			
۵	rhel17.hdp3.cisco.com	10.16.1.47			
	rhel2.hdp3.cisco.com	10.16.1.32			
		Items per page: 10 🔻 1 - 10 of 17 <>			
		CANCEL			

6. Click SAVE to save the configuration group as shown below.

Default (13) GPU-Nodes (4)	rhel14.hdp3.cisco.com rhel15.hdp3.cisco.com rhel16.hdp3.cisco.com rhel17.hdp3.cisco.com	
+ - \$ -	Overrides 17 properties Description	+ -

CANCEL

Enable GPU

To enable GPU, follow these steps:

1. Under default config group. In the GPU section, provide absolute path for nvidia-smi as shown below.

Absolute path of nvidia-smi on NodeManagers



2. Select the GPU Nodes configuration group from Config group drop-down list created in the earlier steps as shown below. Select YARN > CONFIGS.
| Config Group | Default (13) 🔹 | Filter |
|--------------|-------------------|--------|
| | Manage Config Gro | ups |
| | GPU-Nodes (4) | |
| | Default (13) | |
| YARN Fe | alures | |

3. Click Enabled for GPU.

GPU	
GPU Scheduling and Isolation Disabled	
Enabled	2 O C
Absolute path of nvidia-smi on NodeManagers	
/usr/bin/nvidia-smi	

4. Verify Advanced yarn-site with the following:

Properties	Values
GPU docker plugin	nvidia-docker-v1 (default)
GPU docker plugin endpoint for Nvidia Docker Version 1	http://localhost:3476/v1.o/docker/cli

5. Click ADVANCED tab and verify the config for yarn.resource-types in Resource Types section as shown below.

Resource Types

yarn.resource-types	yam.io/gpu
	yam.io/gpu
For non-GPU nodes conf yarn.io/gpu	ig group which is default in this case, make sure yarn.resource-types should also be

6. Verify and/or apply the following settings for Container Executor in YARN config for GPU-nodes configuration group.

Properties	Values
CGroup Root Path	/sys/fs/cgroup
Docker Allowed Devices	regex:^/dev/nvidia.*\$
Docker Allowed Read-only Mounts	/sys/fs/cgroup,regex:^nvidia_driver*\$
Docker Allowed Volume-drivers	nvidia-docker
Docker Binary	/usr/bin/nvidia-docker
Yarn CGroup Hierarchy	/hadoop-yarn

Table 15 YARN – Container Executor for GPU Properties

Container Executor		
CGroup Root Path		
	/sys/fs/cgroup	•
Docker Allowed Devices		
	regex:*/dev/nvidia.*\$	•
Docker Allowed Read-only Mounts	/sys/fs/cgroup	
	/sys/fs/cgroup,regex:*nvidia_driver_*\$	•
Docker Allowed Read-write Mounts		0
Docker Allowed Volume-drivers		
	nvidia-docker	•
Docker Binary	/usr/bin/docker	
	/usr/bin/nvidia-docker	•
Enable Launching Privileged Containers	= 0	
Minimum user ID for submitting job	50	0

7. Click SAVE and RESTSRT for all affected services in Ambari.

Verify YARN Configurations

All of the configurations make changes in /etc/hadoop/conf/yarn-site.xml and /etc/hadoop/conf/container-executor.cfg file. To verify the YARN configurations, follow these steps:

1. For Non-GPU nodes, contents of /etc/hadoop/conf/container-executor.cfg file are in this reference design.

```
[root@rhel4 ~]# cat /etc/hadoop/conf/container-executor.cfg
#/*
# * Licensed to the Apache Software Foundation (ASF) under one
# * or more contributor license agreements. See the NOTICE file
# * distributed with this work for additional information
# * regarding copyright ownership. The ASF licenses this file
# * to you under the Apache License, Version 2.0 (the
# * "License"); you may not use this file except in compliance
# * with the License. You may obtain a copy of the License at
# *
# * http://www.apache.org/licenses/LICENSE-2.0
```

```
# *
# * Unless required by applicable law or agreed to in writing, software
# * distributed under the License is distributed on an "AS IS" BASIS,
# * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# * See the License for the specific language governing permissions and
# * limitations under the License.
# */
yarn.nodemanager.local-dirs=/hadoop/yarn/local,/data/disk1/hadoop/yarn/local
yarn.nodemanager.log-dirs=/hadoop/yarn/log,/data/disk1/hadoop/yarn/log
yarn.nodemanager.linux-container-executor.group=hadoop
banned.users=hdfs,yarn,mapred,bin
min.user.id=50
[docker]
 module.enabled=true
 docker.binary=/usr/bin/docker
docker.allowed.capabilities=CHOWN, DAC OVERRIDE, FSETID, FOWNER, MKNOD, NET RAW, SETGID, SETUID, SETFCAP, SETPCAP,
NET BIND SERVICE, SYS CHROOT, KILL, AUDIT WRITE
 docker.allowed.devices=
  docker.allowed.networks=host,bridge
  docker.allowed.ro-mounts=/hadoop/yarn/local,/data/disk1/hadoop/yarn/local,/sys/fs/cgroup
  docker.allowed.rw-
mounts=/hadoop/yarn/local,/data/disk1/hadoop/yarn/local,/hadoop/yarn/log,/data/disk1/hadoop/yarn/log,
 docker.privileged-containers.enabled=false
  docker.trusted.registries=local,centos,hortonworks
 docker.allowed.volume-drivers=
[gpu]
 module.enabled=false
[cgroups]
 root=
 yarn-hierarchy=/hadoop-yarn
[root@rhel4 ~]#
```

2. For GPU nodes, contents of /etc/hadoop/conf/container-executor.cfg file are in this reference design.

```
[root@rhel16 ~]# cat /etc/hadoop/conf/container-executor.cfg
#/*
# * Licensed to the Apache Software Foundation (ASF) under one
# * or more contributor license agreements. See the NOTICE file
# * distributed with this work for additional information
# * regarding copyright ownership. The ASF licenses this file
# * to you under the Apache License, Version 2.0 (the
#
 * "License"); you may not use this file except in compliance
 ^{\star} with the License. You may obtain a copy of the License at
#
# *
# *
        http://www.apache.org/licenses/LICENSE-2.0
# *
# * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
#
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
\# * See the License for the specific language governing permissions and
# * limitations under the License.
# */
yarn.nodemanager.local-dirs=/hadoop/yarn/local,/data/disk1/hadoop/yarn/local
yarn.nodemanager.log-dirs=/hadoop/yarn/log,/data/disk1/hadoop/yarn/log
yarn.nodemanager.linux-container-executor.group=hadoop
banned.users=hdfs,yarn,mapred,bin
min.user.id=50
[docker]
 module.enabled=true
  docker.binary=/usr/bin/nvidia-docker
docker.allowed.capabilities=CHOWN,DAC OVERRIDE,FSETID,FOWNER,MKNOD,NET RAW,SETGID,SETUID,SETFCAP,SETPCAP,
NET BIND SERVICE, SYS CHROOT, KILL, AUDIT WRITE
```

```
docker.allowed.devices=regex:^/dev/nvidia.*$
docker.allowed.networks=host,bridge
docker.allowed.ro-
mounts=/hadoop/yarn/local,/data/disk1/hadoop/yarn/local,/sys/fs/cgroup,regex:^nvidia_driver_.*$
docker.allowed.rw-
mounts=/hadoop/yarn/local,/data/disk1/hadoop/yarn/local,/hadoop/yarn/log,/data/disk1/hadoop/yarn/log,
docker.privileged-containers.enabled=false
docker.trusted.registries=local,centos,hortonworks
docker.allowed.volume-drivers=nvidia-docker
[gpu]
module.enabled=true
[cgroups]
root=/sys/fs/cgroup
yarn-hierarchy=/hadoop-yarn
[root@rhel16 ~]#
```

3. Launch the YARN Web UI and verify cluster overview as shown below.



4. Click Nodes and select a node as shown below.



Setting Up Docker Registry

Private trusted registry is required to provision YARN container. This topic provides basic information about deploying and configuring a registry.



This is a sample registry to showcase the use-case and not recommended for Production grade setup.

Designate a Server for Docker and Start the Registry

To designate a server for Docker and start the registry, follow these steps:

- 1. Designate a server in the cluster for the Docker registry. Minimal resources are required, but sufficient disk space is needed to store the images and metadata. Docker must be installed and running.
- 2. Optional: By default, data will only be persisted within the container. If you would like to persist the data on the host, you can customize the bind mounts using the -v option.
- 3. Create /var/lib/registry folder:

mkdir /var/lib/registry

4. Configure Docker to allow pulling from this insecure registry. Modify /etc/docker/daemon.json on all nodes in the cluster to include the following configuration options:

```
# cat /etc/docker/daemon.json
{
    "live-restore" : true,
    "debug" : true,
    "insecure-registries" : ["linuxjh.hdp3.cisco.com:5000"]
}
```

- 5. Restart Docker on all nodes.
- 6. Provision the registry container by running the following command:

docker run -d -p 5000:5000 --restart=always --name registry -v /mnt/registry:/var/lib/registry registry:2

7. Verify the registry container is provisioned by running docker ps command:

[root@LinuxJB ~]# d	locker ps -a			
CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
PORTS	NAMES			
037d176d2576	registry:2	"/entrypoint.sh /etc"	14 minutes ago	Up 4 minutes
0.0.0:5000->5000/	tcp registry			
[root@LinuxJB ~]#				

Test the Docker Registry

To test the Docker registry, follow these steps:

1. Pull the Docker image as shown in below:

```
[root@linuxjh dockerfile]# docker pull nvidia/cuda:9.2-base
9.2-base: Pulling from nvidia/cuda
18d680d61657: Already exists
0addb6fece63: Already exists
78e58219b215: Already exists
eb6959a66df2: Already exists
c7f8a5420911: Pull complete
```

```
f20e55ec643a: Pull complete
964bd7cf2ca3: Pull complete
Digest: sha256:de51e247fe0ecfde0fca7ef2d11285db79bd8afc0ef5adf7aade6aee4a9f5f72
Status: Downloaded newer image for nvidia/cuda:9.2-base
```

2. Tag the image:

```
[root@linuxjh dockerfile]# docker tag nvidia/cuda:9.2-base linuxjh.hdp3.cisco.com:5000/nvidia-cuda-cisco-
demo
```

3. Push the image into private registry:

```
[root@LinuxJB ~]# docker push linuxjh.linuxjh.hdp3.cisco.com:5000/nvidia-cuda-cisco-demo
The push refers to repository [linuxjh.hdp3.cisco.com:5000/nvidia-cuda-cisco-demo]
ece4f9fdef59: Pushed
ad5345cbb119: Pushed
ef68f6734aa4: Pushed
latest: digest: sha256:87e9b6904b4286b8d41bba4461c0b736835fcc218f7ecbe5544b53fdd467189f size: 1778
[root@LinuxJB ~]#
```

Apache Hadoop YARN Distributed Shell

To run the YARN distributed shell with docker container having GPU as a resource, follow these steps:

1. Use the following command to run the distributed shell and GPU without a Docker container:

```
# export DJAR="/usr/hdp/current/hadoop-yarn-client/hadoop-yarn-applications-distributedshell.jar"
# yarn jar $DJAR -jar $DJAR -shell_command "/usr/bin/nvidia-smi;sleep 120" -container_resources memory-
mb=1024,vcores=1,yarn.io/gpu=1 -num containers 1 -node label expression "gpu"
```



Node labels can be created and assign to nodes containing GPU, so that YARN schedule the container where the GPU is installed. For configuring node labels, go to: <u>https://docs.hortonworks.com/HDPDocuments/HDP3/HDP-3.0.1/data-operating-</u> <u>system/content/configuring_node_labels.html</u>

2. Use the following command to run the distributed shell and GPU with a Docker container:

```
yarn jar $DJAR -jar $DJAR -shell_env YARN_CONTAINER_RUNTIME_TYPE=docker -shell_env
YARN_CONTAINER_RUNTIME_DOCKER_IMAGE=linuxjh.hdp3.cisco.com:5000/nvidia-cuda-cisco-demo -shell_command
"nvidia-smi;sleep 120" -container_resources memory-mb=1024,vcores=1,yarn.io/gpu=1 -num_containers 1 -
node_label_expression "gpu"
```

3. On the YARN web UI, find the application and the node where it is provisioned as shown below and click prelaunch.out for _ooooo2 container

Node Manager	Show 10 • entries			
i i i i i i i i i i i i i i i i i i i	Container ID	Container State	User	Logs
Node Information	container_e32_1542339065950_0039_01_000002	RUNNING	yarn	prelaunch.out,prelaunch.err,launch_containe
List of Applications on this Node	container_e32_1542339065950_0039_01_000001	RUNNING	yarn	container-localizer- syslog,prelaunch.out,prelaunch.err,launch_co
List of Containers on this Node				
GPU Information				
GFO Information				
ntainer: container	32 1542339865958 8839 81 888882 on r	hel14:45454	1	
ntainer: container_ gAggregationType: L(232_1542339065950_0039_01_000002 on r	hel14:45454]	
ntainer: container_ gAggregationType: L(232_1542339065950_0039_01_000002 on r	hel14:45454]	
ntainer: container_ gAggregationType: L(≥32_1542339065950_0039_01_000002 on r DCAL	hel14:45454		
ntainer: container_ gAggregationType: Lu gType:prelaunch.out gLastModifiedTime:Fi	232_1542339065950_0039_01_000002 on r DCAL ri Nov 16 09:08:15 -0800 2018	hel14:45454		
ntainer: container_ gAggregationType: Lo gType:prelaunch.out gLastModifiedTime:Fo gLength:1412	232_1542339065950_0039_01_000002 on r DCAL ri Nov 16 09:08:15 -0800 2018	hel14:45454]	
ntainer: container_ gAggregationType: Lo gType:prelaunch.out gLastModifiedTime:Fo gLength:1412 gContents:	232_1542339065950_0039_01_000002 on r DCAL ri Nov 16 09:08:15 -0800 2018	•hel14:45454]	
ntainer: container_ gAggregationType: L(gType:prelaunch.out gLastModifiedTime:Fr gLength:1412 gContents: tting up env variab	232_1542339065950_0039_01_000002 on r DCAL ri Nov 16 09:08:15 -0800 2018 les	hel14:45454]	
ntainer: container_ gAggregationType: Lu gType:prelaunch.out gLastModifiedTime:Fu gLength:1412 gContents: tting up env variab tting up job resource	232_1542339065950_0039_01_000002 on r DCAL ri Nov 16 09:08:15 -0800 2018 Les	hel14:45454]	
ntainer: container_ gAggregationType: L(gType:prelaunch.out gLastModifiedTime:Fr gLength:1412 gContents: tting up env variab tting up job resour- pying debugging info	232_1542339065950_0039_01_000002 on r DCAL ri Nov 16 09:08:15 -0800 2018 les ces	hel14:45454]	
ntainer: container_ gAggregationType: L(gType:prelaunch.out gLastModifiedTime:Fr gLength:1412 gContents: tting up env variab tting up job resource pying debugging info unching container	232_1542339065950_0039_01_000002 on r DCAL TNOV 16 09:08:15 -0800 2018 Les tes	hel14:45454]	
ntainer: container_ gAggregationType: L(gType:prelaunch.out gLastModifiedTime:Fr gLength:1412 gContents: tting up env variab tting up job resour- pying debugging info unching container i Nov 16 09:08:15 20	232_1542339065950_0039_01_000002 on r DCAL ri Nov 16 09:08:15 -0800 2018 les ces prmation	hel14:45454]	
ntainer: container_ gAggregationType: L(gType:prelaunch.out gLastModifiedTime:Fr gLength:1412 gContents: tting up env variab tting up job resourc pying debugging info unching container i Nov 16 09:08:15 20 NVIDIA-SMI 396.44	232_1542339065950_0039_01_000002 on r DCAL ri Nov 16 09:08:15 -0800 2018 les tes primation 218 Driver Version: 396.44	hel14:45454]	+
ntainer: container_ gAggregationType: L0 gType:prelaunch.out gLastModifiedTime:Fr gLength:1412 gContents: tting up env variab tting up job resour- pying debugging info unching container i Nov 16 09:08:15 20 NVIDIA-SMI 396.44	232_1542339065950_0039_01_000002 on r DCAL ri Nov 16 09:08:15 -0800 2018 les ces prmation 218 Driver Version: 396.44	hel14:45454]	+
ntainer: container_ gAggregationType: L(gType:prelaunch.out gLastModifiedTime:Fr gLength:1412 gContents: tting up env variab tting up job resource pying debugging info unching container i Nov 16 09:08:15 20 NVIDIA-SMI 396.44 GPU Name Per	232_1542339065950_0039_01_000002 on r DCAL ri Nov 16 09:08:15 -0800 2018 les ces prmation 218 Driver Version: 396.44 rsistence-M Bus-Id Disp.A Vo	hel14:45454	orr. EC	+ I .c
ntainer: container_ gAggregationType: L(gType:prelaunch.out gLastModifiedTime:Fr gLength:1412 gContents: tting up env variab tting up job resource pying debugging info unching container i Nov 16 09:08:15 20 NVIDIA-SMI 396.44 GPU Name Per Fan Temp Perf Pw	232_1542339065950_0039_01_000002 on r DCAL ri Nov 16 09:08:15 -0800 2018 les ces prmation 018 Driver Version: 396.44 rsistence-M Bus-Id Disp.A Vo r:Usage/Cap Memory-Usage Gi	hel14:45454	orr. EC	+ + +
ontainer: container_ ogAggregationType: L(ogType:prelaunch.out ogLastModifiedTime:Fo ogContents: etting up env variab etting up job resour- opying debugging info unching container ei Nov 16 09:08:15 20 NVIDIA-SMI 396.44 GPU Name Per Fan Temp Perf Pwi	232_1542339065950_0039_01_000002 on r DCAL ri Nov 16 09:08:15 -0800 2018 les tes tes Driver Version: 396.44 rsistence-M Bus-Id Disp.A Vo r:Usage/Cap Memory-Usage Gi	hel14:45454	orr. EC	+ + + + +

4. As shown in the figure above, the container is running in rhel14 node. ssh to rhel14 and run the following command:

[root@rhel14 ~]# docker ps CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES 37a3ba06cfe0 local/nvidia-cuda "bash /data/disk1/ha..." 58 seconds ago Up 58 seconds container_e32_1542339065950_0039_01_000002

Run TensorFlow Container Using YARN Distributed Shell

In this example, Nvidia GPU Cloud (NGC) is used as the repository for AI frameworks Docker image. The TensorFlow Docker container image is pulled from NGC which is aligned to the CUDA installed on the server pulling the image.

To deploy TensorFlow container with YARN distributed shell, follow these steps:

1. Pull the tensorflow docker image as shown below:

docker pull nvcr.io/nvidia/tensorflow:18.07-py3

2. Tag the image:

docker tag nvcr.io/nvidia/tensorflow:18.07-py3 linuxjh.hdp3.cisco.com:5000/nvcr-tf18.07-demo

3. Push the image to private registry:

```
1. [root@linuxjh dockerfile]# docker push linuxjh.hdp3.cisco.com:5000/nvcr-tf18.07-demo
2. The push refers to repository [linuxjh.hdp3.cisco.com:5000/nvcr-tf18.07-demo]
3. d36fbb9466ff: Preparing
4. 7101bec62098: Preparing
5. 7359bbd5e7f6: Preparing
6. ba881f735df8: Preparing
7.
  df07f5454848: Preparing
8. d36fbb9466ff: Pushed
9. 3211ba387e89: Mounted from nvcr-tf-demo
10. fc45365a529d: Mounted from nvcr-tf-demo
11. 0191fba0370f: Mounted from nvcr-tf-demo
12. f77fe3a5de0b: Mounted from nvcr-tf-demo
13.
14. [root@linuxjh dockerfile]# docker image ls
15. REPOSITORY
                                                         TAG
                                                                                        IMAGE ID
   CREATED
                       SIZE
16. linuxjh.hdp3.cisco.com:5000/nvcr-tf18.07-demo
                                                         latest
                                                                                        a9950e6bf1b5
   4 minutes ago
                       4.49GB
```

4. Run YARN distributed shell to provision tensorflow docker container in GPU nodes:

export DJAR="/usr/hdp/current/hadoop-yarn-client/hadoop-yarn-applications-distributedshell.jar"

[root@rhel1 tmp]# yarn jar \$DJAR -jar \$DJAR -shell_env YARN_CONTAINER_RUNTIME_TYPE=docker -shell_env YARN_CONTAINER_RUNTIME_DOCKER_IMAGE=linuxjh.hdp3.cisco.com:5000/nvcr-tf18.07-demo -shell_command "python -c 'import tensorflow as tf; sess=tf.Session();print(sess.run(tf.constant(65)*tf.constant(445)))';sleep 300" -container_resources memory-mb=4096,vcores=2,yarn.io/gpu=6 -num_containers 1 -node_label_expression "c480ml"

The node label was used in the command (above) so that YARN provisions the docker container in the node with GPU. In this reference example, rhel20.hdp3.cisco.com was assigned a node label named "c48oml"

Below is the output of this command. Output has been truncated for simplicity:

```
[root@rhel1 tmp]# yarn jar $DJAR -jar $DJAR -shell_env YARN_CONTAINER_RUNTIME_TYPE=docker -shell_env
YARN_CONTAINER_RUNTIME_DOCKER_IMAGE=linuxjh.hdp3.cisco.com:5000/nvcr-tf18.07-demo -shell_command "python
-c 'import tensorflow as tf; sess=tf.Session();print(sess.run(tf.constant(65)*tf.constant(445)))';sleep
300" -container_resources memory-mb=4096,vcores=2,yarn.io/gpu=6 -num_containers 1 -node_label_expression
"c480ml"
19/01/16 11:56:48 INFO distributedshell.Client: Initializing Client
19/01/16 11:56:48 INFO distributedshell.Client: Running Client
```

19/01/16 11:56:48 INFO client.RMProxy: Connecting to ResourceManager at rhel1.hdp3.cisco.com/10.16.1.31:8050 19/01/16 11:56:49 INFO client.AHSProxy: Connecting to Application History server at rhel2.hdp3.cisco.com/10.16.1.32:10200 19/01/16 11:56:49 INFO distributedshell.Client: Got Cluster metric info from ASM, numNodeManagers=16 19/01/16 11:56:49 INFO distributedshell.Client: Got Cluster node info from ASM 19/01/16 11:56:49 INFO distributedshell.Client: Got node report from ASM for, nodeId=rhel14:45454, nodeAddress=rhel14:8042, nodeRackName=/default-rack, nodeNumContainers=0 19/01/16 11:56:49 INFO distributedshell.Client: Got node report from ASM for, nodeId=rhel9:45454, nodeAddress=rhel9:8042, nodeRackName=/default-rack, nodeNumContainers=0 19/01/16 11:56:49 INFO distributedshell.Client: Got node report from ASM for, nodeId=rhel5:45454, nodeAddress=rhel5:8042, nodeRackName=/default-rack, nodeNumContainers=0 19/01/16 11:56:49 INFO distributedshell.Client: Got node report from ASM for, nodeId=rhel15:45454, nodeAddress=rhel15:8042, nodeRackName=/default-rack, nodeNumContainers=0 19/01/16 11:56:49 INFO distributedshell.Client: Got node report from ASM for, nodeId=rhel8:45454, nodeAddress=rhel8:8042, nodeRackName=/default-rack, nodeNumContainers=0 ... Output truncated for readability ••• 19/01/16 12:02:06 INFO distributedshell.Client: Got application report from ASM for, appId=11, clientToAMToken=null, appDiagnostics=, appMasterHost=rhel20.hdp3.cisco.com/10.16.1.50, appQueue=default, appMasterRpcPort=-1, appStartTime=1547668610305, yarnAppState=RUNNING, distributedFinalState=UNDEFINED, appTrackingUrl=http://rhel1.hdp3.cisco.com:8088/proxy/application 1547666438624 0011/, appUser=root 19/01/16 12:02:07 INFO distributedshell.Client: Got application report from ASM for, appId=11, clientToAMToken=null, appDiagnostics=, appMasterHost=rhel20.hdp3.cisco.com/10.16.1.50, appQueue=default, appMasterRpcPort=-1, appStartTime=1547668610305, yarnAppState=RUNNING, distributedFinalState=UNDEFINED, appTrackingUrl=http://rhel1.hdp3.cisco.com:8088/proxy/application 1547666438624 0011/, appUser=root 19/01/16 12:02:08 INFO distributedshell.Client: Got application report from ASM for, appId=11, clientToAMToken=null, appDiagnostics=, appMasterHost=rhel20.hdp3.cisco.com/10.16.1.50, appQueue=default, appMasterRpcPort=-1, appStartTime=1547668610305, yarnAppState=RUNNING, distributedFinalState=UNDEFINED, appTrackingUrl=http://rhel1.hdp3.cisco.com:8088/proxy/application 1547666438624 0011/, appUser=root 19/01/16 12:02:09 INFO distributedshell.Client: Got application report from ASM for, appId=11, clientToAMToken=null, appDiagnostics=, appMasterHost=rhel20.hdp3.cisco.com/10.16.1.50, appQueue=default, appMasterRpcPort=-1, appStartTime=1547668610305, yarnAppState=RUNNING, distributedFinalState=UNDEFINED, appTrackingUrl=http://rhel1.hdp3.cisco.com:8088/proxy/application 1547666438624 0011/, appUser=root 19/01/16 12:02:10 INFO distributedshell.Client: Got application report from ASM for, appId=11, clientToAMToken=null, appDiagnostics=, appMasterHost=rhel20.hdp3.cisco.com/10.16.1.50, appQueue=default, appMasterRpcPort=-1, appStartTime=1547668610305, yarnAppState=RUNNING, distributedFinalState=UNDEFINED, appTrackingUrl=http://rhel1.hdp3.cisco.com:8088/proxy/application 1547666438624 0011/, appUser=root 19/01/16 12:02:11 INFO distributedshell.Client: Got application report from ASM for, appId=11, clientToAMToken=null, appDiagnostics=, appMasterHost=rhel20.hdp3.cisco.com/10.16.1.50, appQueue=default, appMasterRpcPort=-1, appStartTime=1547668610305, yarnAppState=FINISHED, distributedFinalState=SUCCEEDED, appTrackingUrl=http://rhel1.hdp3.cisco.com:8088/proxy/application 1547666438624 0011/, appUser=root 19/01/16 12:02:11 INFO distributedshell.Client: Application has completed successfully. Breaking monitoring loop 19/01/16 12:02:11 INFO distributedshell.Client: Application completed successfully

5. Log into YARN Web UI and click the Applications tab as shown below.

	Cluster Ov	verview Queues	Applications	Services Flo	ow Activity N	odes Tools							Logged in as: UN	IKNOV	NN_USER
Home / Application	ons														Refresh
Reç • Search			Search								1 2	3 4	5 Last - 27	2	15 Rows •
User (7)		Application ID		\$ Application	Type ‡ Appli	cation Name $\stackrel{\scriptscriptstyle A}{\tau}$	User 🗍	State	Queue	÷	Progress	¢	Start Time	÷	Elapsed Tin
Filter		application_15476664	438624_0011	YARN	Distri	butedShell	root	• Finished	default		100	6	2019/01/16 11:5	k	5m 21s 481
 in root in varn-ats 	538 52	application_15476664	438624_0010	YARN	Distri	butedShell	root	Finished	default		100		2019/01/16 11:5		10s 457ms
🗷 yam	20	application_15476664	438624_0009	YARN	Distri	butedShell	root	• Finished	default		100		2019/01/16 11:4	É.	9s 902ms
☑ hdfs☑ cisco	13 12	application_15476664	438624_0008	YARN	Distri	butedShell	root	Finished	default		100	<i>k</i> / /	2019/01/16 11:4	k	10s 818ms

6. Click the application_1547666438624_0011 for details about the application, such as where it is provisioned by YARN. As shown below, this particular container is provisioned in rhel20.hdp3.cisco.com by YARN.

C C NOTS							
hadoop	Cluster Overview	Queues Applications	Services Flow Act	ivity Nodes	Tools		Logged in as: UNKNOW!
Iome / Applicatio	ons / App [application_154	7666438624_0011] / Attemp	pts				
istributedS							=
plication_1547666	6438624_0011						P
Finished	-1 1547000010005						н
root 🔘 Started a	at 1547668610305						
the second second							
THE PROPERTY AND A DEST	Resource Usage Diag	inostics Logs					
ttempts List	Resource Usage Diag	inostics Logs					
Application Att	Resource Usage Diag	inostics Logs					
Application Atte	Resource Usage Diag empts Grid View	inostics Logs					
Application Atte	Resource Usage Diag empts Grid View	nostics Logs			2010 app	hattempt_1547666438624_0011_00	00001
Application Att	Resource Usage Diag tempts Grid View	nostics Logs			2014 app	nattempt_1547666438624_0011_00	00001 appattempt_1547666438624_0011
Application Att Graph View	Resource Usage Diag tempts Grid View	nostics Logs			2010 app App Star	nattempt_1547666438624_0011_00 Nication Attempt Id rted Time	appattempt_1547666438624_0011 2019/01/16 11:56:50
Application Att	Resource Usage Dia tempts Grid View	nostics Logs			2016 app App Stai Fini	Nattempt_1547666438624_0011_00 Nication Attempt Id ted Time shed Time	appattempt_1547666438624_0011 2019/01/16 11:56:50 2019/01/16 12:02:11
Application Atto	Resource Usage Dia tempts Grid View 20100110	nostics Logs			2011 app App Sta Fini Elaj	hattempt_1547668438624_0011_00 Mication Attempt Id rted Time shed Time psed Time	00001 appattempt_1547666438624_0011 2019/01/16 11:56:50 2019/01/16 12:02:11 23 Hrs : 59 Mins : 24 Secs
Application Atto	Resource Usage Dia tempts Grid View 20160105	nostics Logs			2015 app App Sta Fini Elaj AM	attempt_1547666438624_0011_00 Nication Attempt Id ted Time shed Time psed Time Container Id	20001 appattempt_1547666438624_0011 2019/01/16 11:56:50 2019/01/16 12:02:11 23 Hrs : 59 Mins : 24 Secs container_e77_1547666438624_001.
Application Atta	Resource Usage Dia pempts Grid View 20100110	nostics Logs			2015 app App Star Fini Elaj AM	attempt_1547666438624_0011_00 Nication Attempt Id rted Time shed Time psed Time Container Id Node Id	20001 appattempt_1547666438624_0011 2019/01/16 11:56:50 2019/01/16 12:02:11 23 Hrs : 59 Mins : 24 Secs container_e77_1547666438624_001. rhel20.hdp3.cisco.com:45454
Application Att	Resource Usage Diag empts Grid View	nostics Logs			2015 app App Sta Fini Elay AM AM	attempt_1547666438624_0011_00 Nication Attempt Id ted Time shed Time osed Time Container Id Node Id Node Web UI	00001 appattempt_1547666438624_0011 2019/01/16 11:56:50 2019/01/16 12:02:11 23 Hrs : 59 Mins : 24 Secs container_e77_1547666438624_001. rhet20 hdp3.cisco.com:45454 rhet20 hdp3.cisco.com:8042

7. Click rhel2o.hdp3.cisco.com. Detailed node information is displayed as shown below. Under Node Information tab, panel Resource yarn.io/gpu shows 6 GPUs is used and 2 GPUs are available. In the previous YARN distributed shell command mentioned in step 4, 6 GPUs were requested for the container.



8. Click List of Containers on this Node. This displays the containers provisioned in this node, as shown below.

ime Nodes Node [rhel20 ht	dp3.cisco.com.45454] Containers			
ada Managar	Show 10 • entries			Search:
ode Manager	Container ID	Container	liser	1005
Node Information	container_e77_1547666438624_0011_01_000002	RUNNING	root	stdout.bit, stderr.bd prelaunch.out.prelaunch.err, launch_container.sh, directory info, stdout, stderr
List of Applications on this Node	container_e77_1547666438624_0011_01_000001	RUNNING	root	container-localizer- syslog.prelaunch out,prelaunch err, launch_container sh.directory into AppMaster stdout, AppMaster s
List of Containers on this				Pre

9. Click prelaunch.out to see the output of the container, as shown below

ome / Nodes / Node [rhel20.hd	p3.cisco.com:45454] / Container [container_e77_1547666438624 / Log
Node Manager	prelaunch.out for container_e77_1547666438624_0011_01_000002
Node Information List of Applications on this Node	Container: container_e77_1547666438624_0011_01_000002 on rhel20.hdp3.cisco.com:45454 LogAggregationType: LOCAL LogType:prelaunch.out LogLastNodifiedTime:Wed Jan 16 11:56:22 -0800 2019 LogLength:106
List of Containers on this Node	LogContents: Setting up pob resources Copying debugging information Launching container 28925 End of LogType:prelaunch.out.This log file belongs to a running container (container_e77_1547666438624_0011_01_000002) and so may not be complete

6

Bill of Materials

This section provides the BOM for the 24 Nodes Hadoop Base Rack and 8 Nodes Hadoop Expansion Rack. See Table 16 for the BOM for the Hadoop Base rack, Table 17 for BOM for Hadoop Expansion Rack, Table 18 for BOM for Hadoop GPU Rack, Table 19 for Red Hat Enterprise Linux License, and Table 20 lists Cloudera SKUs available from Cisco.

If UCS-SP-CPA₄-P₂ is added to the BOM all the required components for 16 servers only are automatically added. If not customers can pick each of the individual components that are specified after this and build the BOM manually.

Part Number	Description	Qty
UCS-SP-C240M5-A2	SP C240 M5SX w/2x6132,6x32GB mem,VIC1387	24
CON-OSP-C240M5A2	SNTC 24X7X4OS UCS C240 M5 A2	24
UCS-CPU-6132	2.6 GHz 6132/140W 14C/19.25MB Cache/DDR4 2666MHz	48
UCS-MR-X32G2RS-H	32GB DDR4-2666-MHz RDIMM/PC4-21300/dual rank/x4/1.2v	144
UCSC-PCI-1-C240M5	Riser 1 including 3 PCIe slots (x8, x16, x8); slot 3 required CPU2	24
UCSC-MLOM-C40Q-03	Cisco VIC 1387 Dual Port 40Gb QSFP CNA MLOM	24
UCSC-PSU1-1600W	Cisco UCS 1600W AC Power Supply for Rack Server	48
CAB-9K12A-NA	Power Cord, 125VAC 13A NEMA 5-15 Plug, North America	48
UCSC-RAILB-M4	Ball Bearing Rail Kit for C220 & C240 M4 & M5 rack servers	24
CIMC-LATEST	IMC SW (Recommended) latest release for C-Series Servers.	24
UCSC-HS-C240M5	Heat sink for UCS C240 M5 rack servers 150W CPUs & below	48
UCSC-BBLKD-S2	UCS C-Series M ₅ SFF drive blanking panel	624
UCSC-PCIF-240M5	C240 M5 PCIe Riser Blanking Panel	24
CBL-SC-MR12GM5P	Super Cap cable for UCSC-RAID-M5HD	24
UCSC-SCAP-M5	Super Cap for UCSC-RAID-M5, UCSC-MRAID1GB-KIT	24
UCSC-RAID-M5HD	Cisco 12G Modular RAID controller with 4GB cache	24
UCS-SP-FI6332-2X	UCS SP Select 2 x 6332 FI	1
UCS-SP-FI6332	(Not sold standalone) UCS 6332 1RU FI/12 QSFP+	2
CON-OSP-SPFI6332	ONSITE 24X7X4 (Not sold standalone) UCS 6332 1RU FI/No PSU/3	2
UCS-PSU-6332-AC	UCS 6332 Power Supply/100-240VAC	4
САВ-9К12А-NA	Power Cord, 125VAC 13A NEMA 5-15 Plug, North America	4
QSFP-H40G-CU3M	4oGBASE-CR4 Passive Copper Cable, 3m	16
QSFP-40G-SR-BD	QSFP40G BiDi Short-reach Transceiver	8

Table 16 Bill of Materials for Cisco UCS C240 M5 SX Hadoop Nodes Base Rack

Part Number	Description	Qty
N10-MGT015	UCS Manager v3.2(1)	2
UCS-ACC-6332	UCS 6332 Chassis Accessory Kit	2
UCS-FAN-6332	UCS 6332 Fan Module	8
QSFP-H40G-CU3M=	4oGBASE-CR4 Passive Copper Cable, 3m	48
UCS-SP-H1P8TB-4X	UCS SP 1.8 TB 12G SAS 10K RPM SFF HDD (4K) 4Pk	96
UCS-SP-H1P8TB	1.8 TB 12G SAS 10K RPM SFF HDD (4K)	384
UCS-SP-HD-1P8T-2	1.8TB 12G SAS 10K RPM SFF HDD (4K) 2 Pack	16
UCS-SP-HD-1P8T	SP 1.8TB 12G SAS 10K RPM SFF HDD (4K)	32

Table 17	Bill of Materials for Hadoon Nodes Expansion Rack
I dDIE 1/	

Part Number	Description	Qty
UCS-SP-C240M5-A2	SP C240 M5SX w/2x6132,6x32GB mem,VIC1387	8
CON-OSP-C240M5A2	SNTC 24X7X4OS UCS C240 M5 A2	8
UCS-CPU-6132	2.6 GHz 6132/140W 14C/19.25MB Cache/DDR4 2666MHz	16
UCS-MR-X32G2RS-H	32GB DDR4-2666-MHz RDIMM/PC4-21300/dual rank/x4/1.2v	48
UCSC-PCI-1-C240M5	Riser 1 including 3 PCIe slots (x8, x16, x8); slot 3 required CPU2	8
UCSC-MLOM-C40Q-03	Cisco VIC 1387 Dual Port 40Gb QSFP CNA MLOM	8
UCSC-PSU1-1600W	Cisco UCS 1600W AC Power Supply for Rack Server	16
CAB-9K12A-NA	Power Cord, 125VAC 13A NEMA 5-15 Plug, North America	16
UCSC-RAILB-M4	Ball Bearing Rail Kit for C220 & C240 M4 & M5 rack servers	8
CIMC-LATEST	IMC SW (Recommended) latest release for C-Series Servers.	8
UCSC-HS-C240M5	Heat sink for UCS C240 M5 rack servers 150W CPUs and below	16
UCSC-BBLKD-S2	UCS C-Series M5 SFF drive blanking panel	208
UCSC-PCIF-240M5	C240 M5 PCIe Riser Blanking Panel	8
CBL-SC-MR12GM5P	Super Cap cable for UCSC-RAID-M5HD	8
UCSC-SCAP-M5	Super Cap for UCSC-RAID-M5, UCSC-MRAID1GB-KIT	8
UCSC-RAID-M5HD	Cisco 12G Modular RAID controller with 4GB cache	8
UCS-SP-H1P8TB-4X	UCS SP 1.8 TB 12G SAS 10K RPM SFF HDD (4K) 4Pk	48
UCS-SP-H1P8TB	1.8 TB 12G SAS 10K RPM SFF HDD (4K)	192
UCS-SP-HD-1P8T-2	1.8TB 12G SAS 10K RPM SFF HDD (4K) 2 Pack	8
UCS-SP-HD-1P8T	SP 1.8TB 12G SAS 10K RPM SFF HDD (4K)	16

Table 18Bill of Materials for Cisco UCS C480 ML Nodes RackPart NumberDescription

Qty

UCSC-C480-M5ML8	Chassis w/8GPU, NoPSU, NoRAID/cable, NoHDDmod, NoCPUmod	
CON-OSP-480M5ML8	SNTC-24X7X4OS Chassis w/8GPU, NoPSU, NoRAID/cable, NoHDDmod,	1
UCSC-C480-CM	UCS C480 M5 CPU Module w/o CPU, mem	1
UCS-CPU-6142	2.6 GHz 6142/150W 16C/22MB Cache/DDR4 2666MHz	2
UCS-MR-X32G2RS-H	32GB DDR4-2666-MHz RDIMM/PC4-21300/dual rank/x4/1.2v	12
UCS-M2-960GB	960GB SATA M.2	2
UCSC-PSU1-1600W	Cisco UCS 1600W AC Power Supply for Rack Server	4
CIMC-LATEST	IMC SW (Recommended) latest release for C-Series Servers.	1
UCS-SID-INFR-BD	Big Data and Analytics Platform (Hadoop/IoT/ITOA/AI/ML)	1
UCS-SID-WKL-BD	Big Data and Analytics (Hadoop/IoT/ITOA)	1
UCSC-RAIL-4U-M5	Rail Kit for UCS C480 M5	1
UCS-DIMM-BLK	UCS DIMM Blanks	12
UCS-MSTOR-M2	Mini Storage carrier for M.2 SATA/NVME (holds up to 2)	1
UCSC-SCAP-M5	Super Cap for UCSC-RAID-M5, UCSC-MRAID1GB-KIT	1
CBL-SC-MR12GM5P	Super Cap cable for UCSC-RAID-M5HD	1
UCSC-C480-CM-FLR	UCS C480 M5 CPU Module Filler	1
UCSC-HS-02-EX	CPU Heat Sink for UCS C480 M5 Rack Server	2
UCSC-BZL-EX-M5ML	Optional Bezel for UCS C480 M5ML rack server	1
UCSC-C480-8HDD	UCS C480 M5 Drive Module for 8x HDD	1
UCS-HD18TB10K4KN	1.8TB 12G SAS 10K RPM SFF HDD (4K)	8
UCSC-C480-8HDD	UCS C480 M5 Drive Module for 8x HDD	1
UCS-HD18TB10K4KN	1.8TB 12G SAS 10K RPM SFF HDD (4K)	8
UCSC-C480-8HDD	UCS C480 M5 Drive Module for 8x HDD	1
UCS-HD18TB10K4KN	1.8TB 12G SAS 10K RPM SFF HDD (4K)	8
UCSC-RAID-M5HD	Cisco 12G Modular RAID controller with 4GB cache	1
CAB-250V-10A-CN	AC Power Cord - 250V, 10A - PRC	4

Table 19 Red Hat Enterprise Linux License

Red Hat Enterprise Linux		
RHEL-2S2V-3A	Red Hat Enterprise Linux	28
CON-ISV1-EL2S2V3A	3 year Support for Red Hat Enterprise Linux	28
_		

Cisco SKU	Cisco PID with Duration	Product Name
UCS-BD-HDP-JSS=	UCS-BD-HDP-JSS-6M	HDP Data Platform Jumpstart Subscription - Up to 16 Nodes – 1 Business Day Response - 6 Months - sold to new customers only - max. Qty. to buy 1 SKU per customer
UCS-BD-HDP-ENT-ND=	UCS-BD-ENT-ND-1Y	HDP Enterprise Subscription - 4 Nodes - 24x7 Sev 1 Response - 1 Year - min of 3 SKUs required for new customers
UCS-BD-HDP-ENT-ND=	UCS-BD-ENT-ND-2Y	HDP Enterprise Subscription - 4 Nodes - 24x7 Sev 1 Response - 2 Year - min of 3 SKUs required for new customers
UCS-BD-HDP-ENT-ND=	UCS-BD-ENT-ND-3Y	HDP Enterprise Subscription - 4 Nodes - 24x7 Sev 1 Response - 3 Year - min of 3 SKUs required for new customers
UCS-BD-HDP-EPL-ND=	UCS-BD-EPL-ND-1Y	HDP Enterprise Plus Subscription - 4 Nodes - 24x7 Sev 1 Response - 1 Year - min of 3 SKUs required for new customers
UCS-BD-HDP-EPL-ND=	UCS-BD-EPL-ND-2Y	HDP Enterprise Plus Subscription - 4 Nodes - 24x7 Sev 1 Response - 2 Year - min of 3 SKUs required for new customers
UCS-BD-HDP-EPL-ND=	UCS-BD-EPL-ND-3Y	HDP Enterprise Plus Subscription - 4 Nodes - 24x7 Sev 1 Response - 3 Year - min of 3 SKUs required for new customers
UCS-BD-HDP-J2E-ND=	UCS-BD-J2E-ND-U-1Y	HDP Jumpstart to Enterprise Subscription Upgrade - 4 Nodes - 24x7 Sev 1 Response - 1 Year - min of 3 SKUs required
UCS-BD-HDP-J2E-ND=	UCS-BD-J2E-ND-U-2Y	HDP Jumpstart to Enterprise Subscription Upgrade - 4 Nodes - 24x7 Sev 1 Response - 2 Year - min of 3 SKUs required
UCS-BD-HDP-J2E-ND=	UCS-BD-J2E-ND-U-3Y	HDP Jumpstart to Enterprise Subscription Upgrade - 4 Nodes - 24x7 Sev 1 Response - 3 Year - min of 3 SKUs required
UCS-BD-HDP-J2P-ND=	UCS-BD-J2P-ND-U-1Y	HDP Jumpstart to Enterprise Plus Subscription Upgrade - 4 Nodes - 24x7 Sev 1 Response - 1 Year - min of 3 SKUs required

Table 20 Cisco SKUs with Cisco PID and Product Names

Cisco SKU	Cisco PID with Duration	Product Name
UCS-BD-HDP-J2P-ND=	UCS-BD-J2P-ND-U-2Y	HDP Jumpstart to Enterprise Plus Subscription Upgrade - 4 Nodes - 24x7 Sev 1 Response - 2 Year - min of 3 SKUs required
UCS-BD-HDP-J2P-ND=	UCS-BD-J2P-ND-U-3Y	HDP Jumpstart to Enterprise Plus Subscription Upgrade - 4 Nodes - 24x7 Sev 1 Response - 3 Year - min of 3 SKUs required
UCS-BD-HDP-E2P-ND=	UCS-BD-E2P-ND-U-1Y	HDP Enterprise to Enterprise Plus Subscription Upgrade - 4 Nodes - 24x7 Sev 1 Response - 1 Year
UCS-BD-HDP-E2P-ND=	UCS-BD-E2P-ND-U-2Y	HDP Enterprise to Enterprise Plus Subscription Upgrade - 4 Nodes - 24x7 Sev 1 Response - 2 Year
UCS-BD-HDP-E2P-ND=	UCS-BD-E2P-ND-U-3Y	HDP Enterprise to Enterprise Plus Subscription Upgrade - 4 Nodes - 24x7 Sev 1 Response - 3 Year

About the Authors

Muhammad Afzal, Engineering Architect, Computing Systems Product Group, Cisco Systems, Inc.

Muhammad Afzal is an Engineering Architect and Technical Marketing Engineer in Cisco UCS Product Management and Datacenter Solutions Engineering. He is currently responsible for designing, developing, and producing validated architectures for Big Data and analytics while working collaboratively with product partners. Previously, Afzal had been a lead architect for various cloud and data center solutions in Solution Development Unit at Cisco. Prior to this, Afzal has been a Solutions Architect in Cisco's Advanced Services group, where he worked closely with Cisco's large enterprise and service provider customers delivering data center and cloud solutions. Afzal holds an MBA in Finance and a BS in Computer Engineering.

Acknowledgements

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

- Karthik Kulkarni, Architect, Computing Systems Product Group, Cisco Systems, Inc.
- Silesh Bijjahalli, Product Management, Computing Systems Product Group, Cisco Systems, Inc.
- Ali Bajwa, Hortonworks
- Dhananjay Mehta, Hortonworks
- Harsh Shah, Hortonworks
- Wangda Tan, Hortonworks
- Sicong Ji, NVIDIA
- Chetan Tekur, NVIDIA