Ultra M Solutions Guide, Release 6.13

First Published: 2020-04-30

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS

About This Guide

This preface describes the *Ultra M Solution Guide*, how it is organized, and its document conventions.

Ultra M is a pre-packaged and validated virtualized mobile packet core solution designed to simplify the deployment of virtual network functions (VNFs).

Conventions Used

The following tables describe the conventions used throughout this documentation.

Notice Type	Description
Information Note	Provides information about important features or instructions.
Caution	Alerts you of potential damage to a program, device, or system.
Warning	Alerts you of potential personal injury or fatality. May also alert you of potential electrical hazards.

Typeface Conventions	Description
Text represented as a `screen display`	This typeface represents displays that appear on your terminal screen, for example: `Login:`
Text represented as **commands**	This typeface represents commands that you enter, for example: **show ip access-list** This document always gives the full form of a command in lowercase letters. Commands are not case sensitive.

Typeface Conventions	Description
Text represented as a **command** *variable*	This typeface represents a variable that is part of a command, for example: **show card** *slot_number* *slot_number* is a variable representing the desired chassis slot number.
Text represented as menu or sub-menu names	This typeface represents menus and sub-menus that you access within a software application, for example: Click the **File** menu, then click **New**

Supported Documents and Resources

Related Documentation

The most up-to-date information for the UWS is available in the product *Release Notes* provided with each product release.

The following common documents are available:

- *Ultra Gateway Platform System Administration Guide*
- *Ultra-M Deployment Guide*
- *Ultra Services Platform Deployment Automation Guide*
- *Ultra Services Platform NETCONF API Guide*
- *VPC-DI System Administration Guide*
- *StarOS Product-specific and Feature-specific Administration Guides*

Obtaining Documentation

Nephelo Documentation

The most current Nephelo documentation is available on the following website: http://nephelo.cisco.com/page_vPC.html

StarOS Documentation

The most current Cisco documentation is available on the following website: http://www.cisco.com/cisco/web/psa/default.html

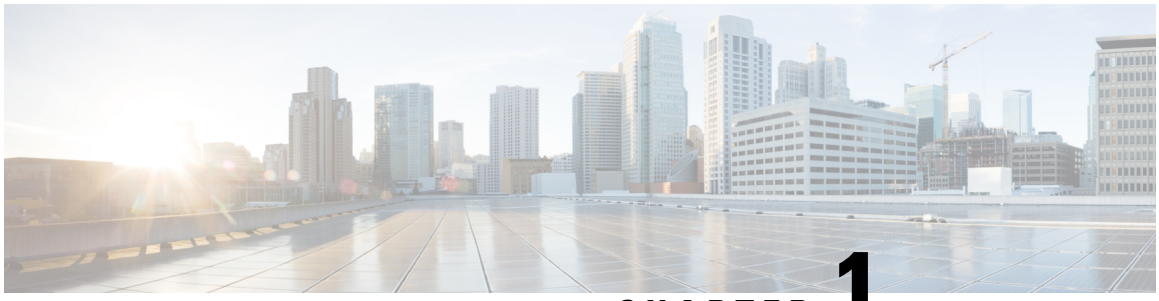
Use the following path selections to access the StarOS documentation:

Products > Wireless > Mobile Internet > Platforms > Cisco ASR 5000 Series > Configure > Configuration Guides

Contacting Customer Support

Use the information in this section to contact customer support.

Refer to the support area of http://www.cisco.com for up-to-date product documentation or to submit a service request. A valid username and password are required to access this site. Please contact your Cisco sales or service representative for additional information.



CHAPTER 1

Ultra M Overview

Ultra M is a pre-packaged and validated virtualized mobile packet core solution designed to simplify the deployment of virtual network functions (VNFs).

The solution combines the Cisco Ultra Service Platform (USP) architecture, Cisco Validated OpenStack infrastructure, and Cisco networking and computing hardware platforms into a fully integrated and scalable stack. As such, Ultra M provides the tools to instantiate and provide basic lifecycle management for VNF components on a complete OpenStack virtual infrastructure manager.

VNF Support

In this release, Ultra M supports the Ultra Gateway Platform (UGP) VNF.

The UGP currently provides virtualized instances of the various 3G and 4G mobile packet core (MPC) gateways that enable mobile operators to offer enhanced mobile data services to their subscribers. The UGP addresses the scaling and redundancy limitations of VPC-SI (Single Instance) by extending the StarOS boundaries beyond a single VM. UGP allows multiple VMs to act as a single StarOS instance with shared interfaces, shared service addresses, load balancing, redundancy, and a single point of management.

Ultra M Model(s)

The Ultra M Extra Small (XS) model is currently available. It is based on OpenStack 10 and implements a Hyper-Converged architecture that combines the Ceph Storage and Compute node. The converged node is referred to as an OSD compute node.

This model includes 6 Active Service Functions (SFs) per VNF and is supported in deployments from 1 to 4 VNFs.

Functional Components

As described in Hardware Specifications, on page 5, the Ultra M solution consists of multiple hardware components including multiple servers that function as controller, compute, and storage nodes. The various functional components that comprise the Ultra M are deployed on this hardware:

- **OpenStack Controller**: Serves as the Virtual Infrastructure Manager (VIM).

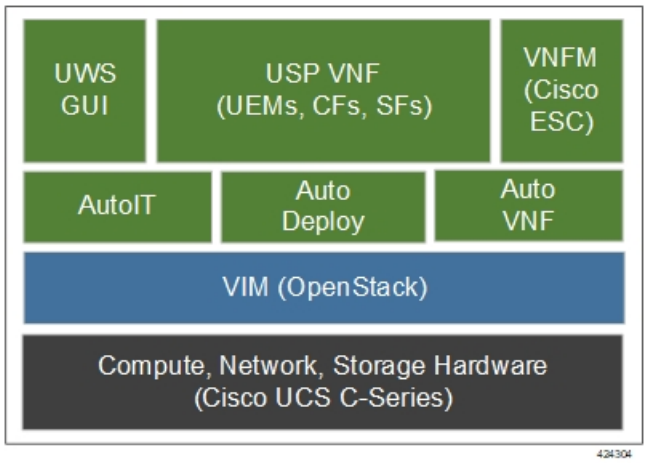
  Important In this release, all VNFs in a multi-VNF Ultra M are deployed as a single "site" leveraging a single VIM.

- **Ultra Automation Services (UAS)**: A suite of tools provided to simplify the deployment process:
  - **AutoIT**: Automates the VIM Orchestrator and VIM installation processes and provides storage and management for system ISOs.
  - **AutoDeploy**: Initiates the deployment of the VNFM and VNF components through a single deployment script.
  - **AutoVNF**: Initiated by AutoDeploy, AutoVNF is directly responsible for deploying the VNFM and VNF components based on inputs received from AutoDeploy.
  - **Ultra Web Service (UWS)**: The Ultra Web Service (UWS) provides a web-based graphical user interface (GUI) and a set of functional modules that enable users to manage and interact with the USP VNF.
- **Cisco Elastic Services Controller (ESC)**: Serves as the Virtual Network Function Manager (VNFM).

  Important ESC is the only VNFM supported in this release.

- **VNF Components**: USP-based VNFs are comprised of multiple components providing different functions:
  - **Ultra Element Manager (UEM)**: Serves as the Element Management System (EMS, also known as the VNF-EM); it manages all of the major components of the USP-based VNF architecture.
  - **Control Function (CF)**: A central sub-system of the UGP VNF, the CF works with the UEM to perform lifecycle events and monitoring for the UGP VNF.
  - **Service Function (SF)**: Provides service context (user I/O ports), handles protocol signaling, session processing tasks, and flow control (demux).

Figure 1: Ultra M Components



Virtual Machine Allocations

Each of the Ultra M functional components are deployed on one or more virtual machines (VMs) based on their redundancy requirements as identified in Table 1: Function VM Requirements per Ultra M Model, on page 3. Some of these component VMs are deployed on a single compute node as described in VM Deployment per Node Type, on page 9. All deployment models use three OpenStack controllers to provide VIM layer redundancy and upgradability.

Table 1: Function VM Requirements per Ultra M Model

	Hyper-Converged	
Function(s)	**XS Single VNF**	**XS Multi VNF**
OSP-D*	1	1
AutoIT**	2	2
AutoDeploy**	2	2
AutoVNF	2	2
ESC (VNFM)	2	2 per NSD
UEM***	2 or 3	3 per VNF or 2 per NSD
CF	2	2 per VNF
SF	8	8 per VNF
* OSP-D is deployed as a VM for Hyper-Converged Ultra M models. ** AutoIT and AutoDeploy each require 2 VMs when deployed in HA mode (recommeded). *** UEM VM requirement is dependent on VNFD configuration. It can be either 2 or 3 instances.		

VM Resource Requirements

The CF, SF, UEM, and ESC VMs require the resource allocations identified in Table 2: VM Resource Allocation, on page 4. The host resources are included in these numbers.

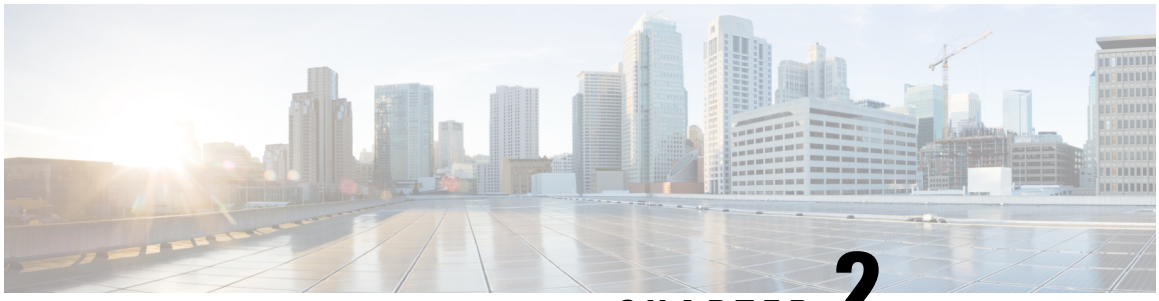
Table 2: VM Resource Allocation

Virtual Machine	vCPU	RAM (GB)	Root Disk (GB)
OSP-D*	16	32	200
AutoIT**	2	8	80
AutoDeploy**	2	8	80
AutoVNF	2	4	40
ESC	2	4	40
UEM	2	4	40
CF	8	16	6
SF	24	96	4

Note 4 vCPUs, 2 GB RAM, and 54 GB root disks are reserved for host reservation.

* OSP-D is deployed as a VM for Hyper-Converged Ultra M models. Though the recommended root disk size is 200GB, additional space can be allocated if available.

** AutoIT is used to deploy the VIM Orchestrator (Undercloud) and VIM (Overcloud) for Hyper-Converged Ultra M models. AutoIT, AutoDeploy, and OSP-D are installed as VMs on the same physical server in this scenario.



CHAPTER 2

Hardware Specifications

Ultra M deployments use the following hardware:

Note The specific component software and firmware versions identified in the sections that follow have been validated in this Ultra M solution release.

Cisco Catalyst Switches

Cisco Catalyst Switches provide as physical layer 1 switching for Ultra M components to the management and provisioning networks. One of two switch models is used based on the Ultra M model being deployed:

Catalyst C2960XR-48TD-I Switch

The Catalyst C2960XR-48TD-I has 48 10/100/1000 ports.

Table 3: Catalyst 2960-XR Switch Information

Ultra M Model(s)	Quantity	Software Version	Firmware Version
Ultra M XS Single VNF	2	IOS 15.2.(2) E5	Boot Loader: 15.2(3r)E1
Ultra M XS Multi-VNF	1 per rack	IOS 15.2.(2) E5	Boot Loader: 15.2(3r)E1

Catalyst 3850-48T-S Switch

The Catalyst 3850 48T-S has 48 10/100/1000 ports.

Table 4: Catalyst 3850-48T-S Switch Information

Ultra M Models	Quantity	Software Version	Firmware Version
Ultra M XS Single VNF	2	IOS: 03.06.06E	Boot Loader: 3.58
Ultra M XS Multi-VNF	1 per Rack	IOS: 03.06.06E	Boot Loader: 3.58

Cisco Nexus Switches

Cisco Nexus Switches serve as top-of-rack (TOR) leaf and end-of-rack (EOR) spine switches provide out-of-band (OOB) network connectivity between Ultra M components. Two switch models are used for the various Ultra M models:

- Nexus 93180-YC-EX, on page 6
- Nexus 9236C , on page 6
- Nexus 9364C, on page 7

Nexus 93180-YC-EX

Nexus 93180 switches serve as network leafs within the Ultra M solution. Each switch has 48 10/25-Gbps Small Form Pluggable Plus (SFP+) ports and 6 40/100-Gbps Quad SFP+ (QSFP+) uplink ports.

Table 5: Nexus 93180-YC-EX

Ultra M Model(s)	Quantity	Software Version	Firmware Version
Ultra M XS Single VNF	2	NX-OS: 7.0(3)I7(3)	BIOS: 7.59
Ultra M XS Multi-VNF	2 per Rack	NX-OS: 7.0(3)I7(3)	BIOS: 7.59

Nexus 9236C

Nexus 9236 switches serve as network spines within the Ultra M solution. Each switch provides 36 10/25/40/50/100 Gbps ports.

The following table provides the Nexus switch recommendation for both Ultra M B1.0 and B1.1 models.

Table 6: Nexus 9236C

Ultra M Model(s)	Quantity	Software Version	Firmware Version
Ultra M XS Single VNF	2	For Ultra M B1.0 model — NX-OS: 7.0(3)I7(3) For Ultra M B1.1 model — NX-OS: 7.0(3)I7(4)	For Ultra M B1.0 model — BIOS: 7.59 For Ultra M B1.1 model — BIOS: 7.61
Ultra M XS Multi-VNF	2	For Ultra M B1.0 model — NX-OS: 7.0(3)I7(3) For Ultra M B1.1 model — NX-OS: 7.0(3)I7(4)	For Ultra M B1.0 model — BIOS: 7.59 For Ultra M B1.1 model — BIOS: 7.61

Nexus 9364C

Nexus 9364 switches serve as network leaf within the Ultra M solution. Each switch provides 64 40-/100-Gigabit QSFP28 interface ports.

The following table provides the Nexus switch recommendation for the Ultra M B1.1 model.

Table 7: Nexus 9364C

Ultra M Model(s)	Quantity	Software Version	Firmware Version
Ultra M XS Single VNF	2	NX-OS: 7.0(3)I7(4)	BIOS: 5.28
Ultra M XS Multi-VNF	2	NX-OS: 7.0(3)I7(4)	BIOS: 5.28

UCS C-Series Servers

Cisco UCS C240 M4S SFF servers and UCS C220 M5S servers host the functions and virtual machines (VMs) required by Ultra M.

Important Note that both the M4 and M5 UCS servers cannot be used in the same PoD, and cannot be replaced with each other.

Server Functions and Quantities

Server functions and quantity differ depending on the Ultra M model you are deploying:

- **Ultra M Manager Node**: Required only for Ultra M models based on the Hyper-Converged architecture, this server hosts the following:
  - AutoIT HA VMs
  - AutoDeploy HA VMs

- OSP-D VM

- **OpenStack Controller Nodes**: These servers host the high availability (HA) cluster that serves as the VIM within the Ultra M solution. In addition, they facilitate the Ceph storage monitor function required by the Ceph Storage Nodes and/or OSD Compute Nodes.
- **OSD Compute Nodes**: Required only for Hyper-converged Ultra M models, these servers provide Ceph storage functionality in addition to hosting VMs for the following:
  - AutoVNF HA VMs
  - Elastic Services Controller (ESC) Virtual Network Function Manager (VNFM) active and standby VMs
  - Ultra Element Manager (UEM) VM HA cluster
  - Ultra Service Platform (USP) Control Function (CF) active and standby VMs
- **Compute Nodes**: For all Ultra M models, these servers host the active, standby, and demux USP Service Function (SF) VMs.

Table 8: Ultra M Server Quantities by Model and Function

Ultra M Model(s)	Server Quantity (max)	Ultra M Manager Node	Controller Nodes	OSD Compute Nodes	Compute Nodes (max)	Additional Specifications
Ultra M XS Single VNF	14	1	3	3	7	Based on node type as described in Table 9: Hyper-Converged Ultra M Single and Multi-VNF UCS C240 M4 Server Specifications by Node Type, on page 10.
Ultra M XS Multi-VNF	45	1	3	3	38*	Based on node type as described in Table 9: Hyper-Converged Ultra M Single and Multi-VNF UCS C240 M4 Server Specifications by Node Type, on page 10.
* Supports a maximum of 4 VNFs – 8 for the first VNF, 10 for each subsequent VNF.						

VM Deployment per Node Type

Figure 2: VM Distribution on Server Nodes for Hyper-converged Ultra M Single VNF Models

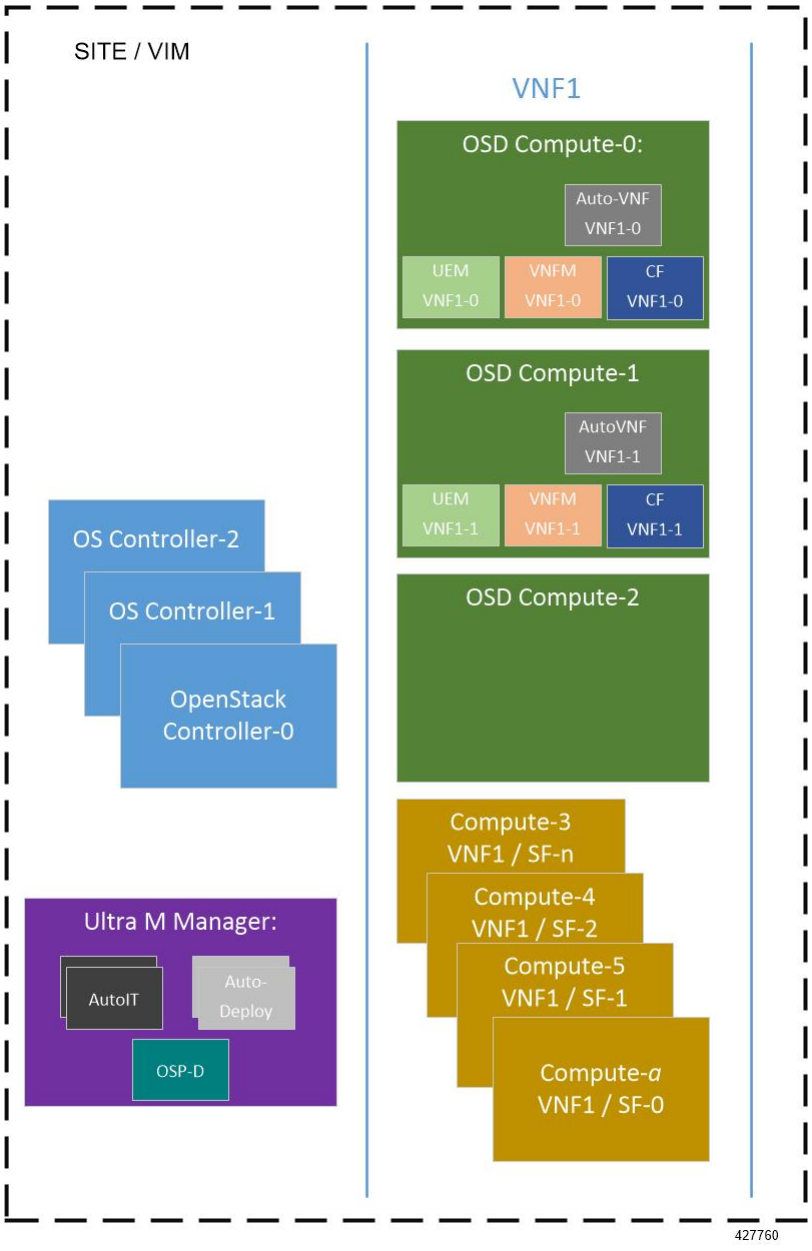
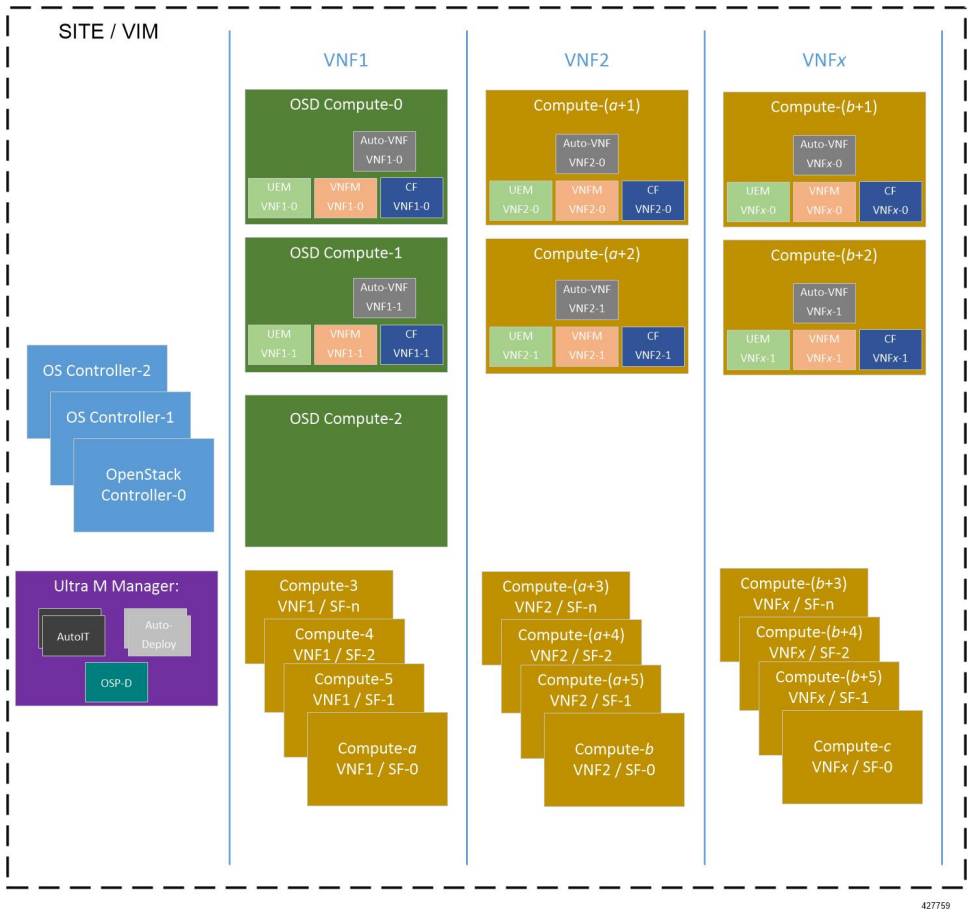


Figure 3: VM Distribution on Server Nodes for Hyper-converged Ultra M Multi-VNF Models



Server Configurations

Table 9: Hyper-Converged Ultra M Single and Multi-VNF UCS C240 M4 Server Specifications by Node Type

Node Type	CPU	RAM	Storage	Software Version	Firmware Version
Ultra M Manager Node*	2x 2.60 GHz	4x 32GB DDR4-2400-MHz RDIMM/PC4	2x 1.2 TB 12G SAS HDD	MLOM: 4.1(3f)	CIMC: 3.0(4d) System BIOS: C240M4.3.0.4a.0.0226182314
Controller	2x 2.60 GHz	4x 32GB DDR4-2400-MHz RDIMM/PC4	2x 1.2 TB 12G SAS HDD	MLOM: 4.1(3f)	CIMC: 3.0(4d) System BIOS: C240M4.3.0.4a.0.0226182314
Compute	2x 2.60 GHz	8x 32GB DDR4-2400-MHz RDIMM/PC4	2x 1.2 TB 12G SAS HDD	MLOM: 4.1(3f)	CIMC: 3.0(4d) System BIOS: C240M4.3.0.4a.0.0226182314

Node Type	CPU	RAM	Storage	Software Version	Firmware Version
OSD Compute	2x 2.60 GHz	8x 32GB DDR4-2400-MHz RDIMM/PC4	4x 1.2 TB 12G SAS HDD 2x 300G 12G SAS HDD HDD 1x 480G 6G SAS SATA SSD	MLOM: 4.1(3f)	CIMC: 3.0(4d) System BIOS: C240M4.3.0.4a.0.0226182314
* OSP-D is deployed as a VM on the Ultra M Manager Node for Hyper-Converged Ultra M model(s).					

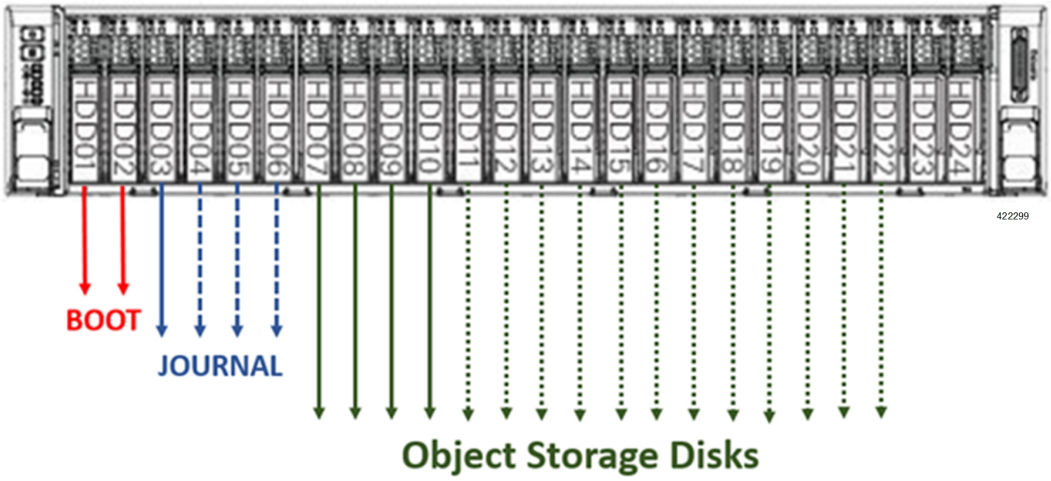
Table 10: UCS C220 M5 Server Specifications by Node Type

Node Type	CPU	RAM	Storage	Software Version	Firmware Version
OSPD Controller Compute	2x 6148 (2.4 GHz 20C)	384 GB RAM	2x 800GB SSD as RAID 1 VD	MLOM: UCS VIC 1387: 4.3(2a)	CIMC: 4.0(2d) System BIOS: C220M5.4.0.2a.0.1102180244
HCI Compute	2x 6148 (2.4 GHz 20C)	384 GB RAM	2x 800GB SSD as RAID 1 VD 4x 800GB SSD as RAID 0 VD	MLOM: UCS VIC 1387: 4.3(2a)	CIMC: 4.0(2d) System BIOS: C220M5.4.0.2a.0.1102180244

Storage

Figure 4: UCS C240 Front-Plane, on page 12 displays the storage disk layout for the UCS C240 series servers used in the Ultra M solution.

Figure 4: UCS C240 Front-Plane



NOTES:

- The Boot disks contain the operating system (OS) image with which to boot the server.
- The Journal disks contain the Ceph journal file(s) used to repair any inconsistencies that may occur in the Object Storage Disks.
- The Object Storage Disks store object data for USP-based VNFs.
- Ensure that the HDD and SSD used for the Boot Disk, Journal Disk, and object storage devices (OSDs) are available as per the Ultra M BoM and installed in the appropriate slots as identified in Table 11: UCS C240 M4S SFF Storage Specifications by Node Type, on page 12 and Table 12: UCS C220 M5 Storage Specifications by Node Type, on page 13.

Table 11: UCS C240 M4S SFF Storage Specifications by Node Type

Ultra M Manager Node and Staging Server	2 x 1.2 TB HDD – For Boot OS configured as Virtual Drive in RAID1 – placed on Slots 1 & 2
Controllers, Computes	2 x 1.2 TB HDD – For Boot OS configured as Virtual Drive in RAID1 – placed on Slots 1 & 2
OSD Computes	2 x 300 GB HDD – For Boot OS configured as Virtual Drive in RAID1 – placed on Slots 1 & 2 1 x 480 GB SSD – For Journal Disk as Virtual Drive in RAID0 – Slot 3 (Reserve for SSD Slot 3, 4, 5, 6 future scaling needs) 4 x 1.2 TB HDD – For OSD's configured as Virtual Drive in RAID0 each – Slot 7, 8, 9, 10 (Reserve for OSD 7, 8, 9, 10…., 24)

Figure 5: UCS C220 Front-Plane, on page 13 displays the storage disk layout for the UCS C220 series servers used in the Ultra M solution.

Figure 5: UCS C220 Front-Plane

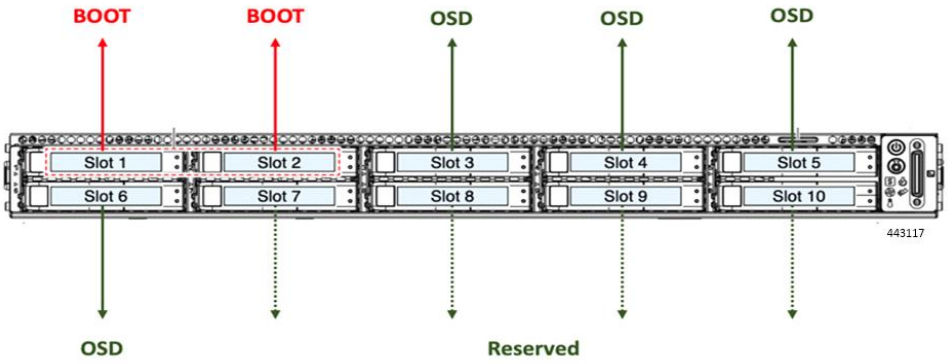


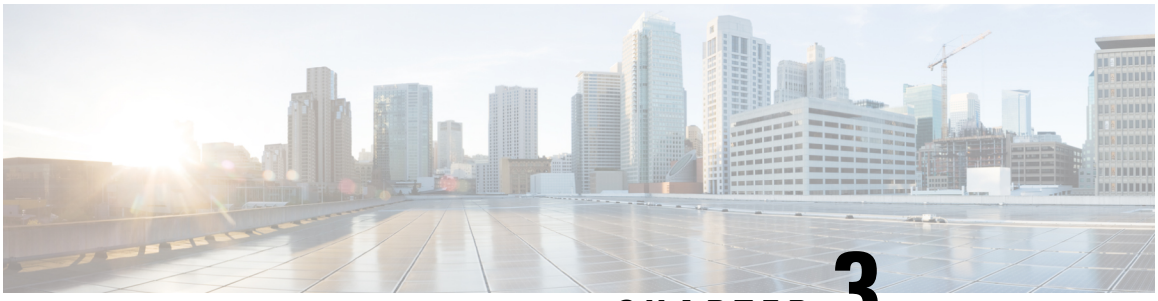
Table 12: UCS C220 M5 Storage Specifications by Node Type

Ultra M Manager Node and Staging Server	2 x 1.2 TB HDD – For Boot OS configured as Virtual Drive in RAID1 – placed on Slots 1 & 2
Controllers, Computes	2 x 1.2 TB HDD – For Boot OS configured as Virtual Drive in RAID1 – placed on Slots 1 & 2
OSD Computes	2 x 1.2 TB HDD – For Boot OS configured as Virtual Drive in RAID1 – placed on Slots 1 & 2 4 x 1.2 TB HDD – For OSD's configured as Virtual Drive in RAID0 each – Slot 3, 4, 5, 6

- Ensure that the RAIDs are sized such that:

  Boot Disks < Journal Disk(s) < OSDs

- Ensure that FlexFlash is disabled on each UCS-C240 M4 (factory default) or UCS C220 M5.
- Ensure that all nodes are in *Unconfigured Good* state under **Cisco SAS RAID Controllers** (factory default).



CHAPTER 3

Software Specifications

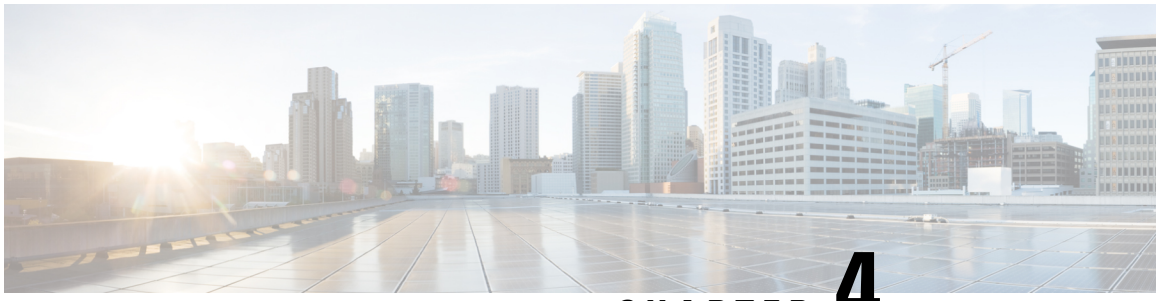
Required Software

Table 13: Required Software

Software	Value/Description
Operating System	In releases prior to 6.0: Red Hat Enterprise Linux 7.3 In 6.0 through 6.2.x releases: Red Hat Enterprise Linux 7.4 In 6.2.bx release, 6.3 release through 6.8 release: Red Hat Enterprise Linux 7.5 In 6.9 and later: Red Hat Enterprise Linux 7.5 or Red Hat Enterprise Linux 7.6
Hypervisor	Qemu (KVM)
VIM	**Hyper-converged Ultra M Single and Multi-VNF Models:** Red Hat OpenStack Platform 10 (OSP 10 - Newton) or Red Hat OpenStack Platform 13 (OSP 13 - Queens)

Software	Value/Description
VNF	In releases prior to 6.0: StarOS release 21.4 In releases 6.0 and 6.1: StarOS release 21.6 In release 6.2.x: StarOS release 21.8 In release 6.2.bx: StarOS release 21.6.x In release 6.3: StarOS release 21.9 In release 6.4: StarOS release 21.10 In release 6.5: StarOS release 21.11 In release 6.6: StarOS release 21.12 In release 6.7: StarOS release 21.13 In release 6.8: StarOS release 21.14 In release 6.9: StarOS release 21.15 In release 6.10: StarOS release 21.16 In release 6.11: StarOS release 21.17 In release 6.12: StarOS release 21.18 In release 6.13: StarOS release 21.19
VNFM	In releases prior to 6.0: ESC 3.1.0.116 In releases 6.0 and 6.1: ESC 3.1.0.145 In release 6.2: ESC 4.0.0.104 In release 6.3: ESC 4.2 In release 6.4 and 6.5: ESC 4.3.0.121 In release 6.6 and 6.7: ESC 4.4.0.88 In release 6.8 and later: ESC 4.5.0.112

Software	Value/Description
UEM	In releases prior to 6.0: UEM 5.7 In releases 6.0 and 6.1: UEM 6.0 In release 6.2: UEM 6.2 In release 6.3: UEM 6.3 In release 6.4: UEM 6.4 In release 6.5: UEM 6.5 In release 6.6: UEM 6.6 In release 6.7: UEM 6.7 In release 6.8: UEM 6.8 In release 6.9: UEM 6.9 In release 6.10: UEM 6.10 In release 6.11: UEM 6.11 In release 6.12: UEM 6.12 In release 6.13: UEM 6.13
USP	In releases prior to 6.0: USP 5.7 In releases 6.0 and 6.1: USP 6.0 In release 6.2: USP 6.2.x/6.2.bx In release 6.3: USP 6.3 In release 6.4: USP 6.4 In release 6.5: USP 6.5 In release 6.6: USP 6.6 In release 6.7: USP 6.7 In release 6.8: USP 6.8 In release 6.9: USP 6.9 In release 6.10: USP 6.10 In release 6.11: USP 6.11 In release 6.12: USP 6.12 In release 6.13: USP 6.13



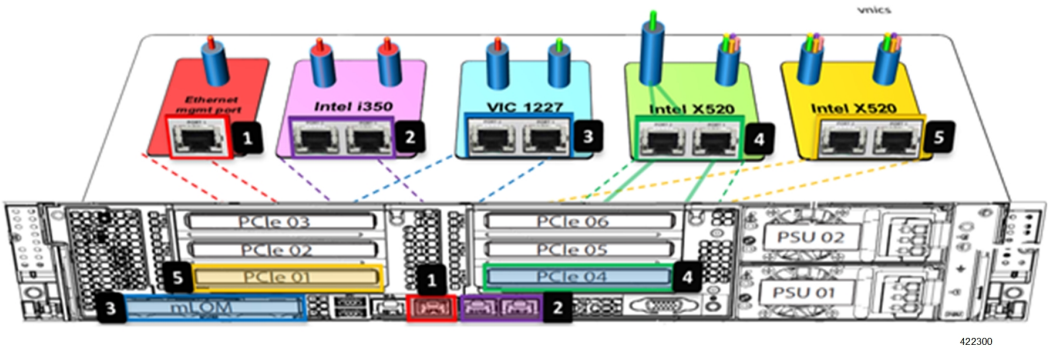
CHAPTER 4

Networking Overview

This section provides information on Ultra M networking requirements and considerations.

UCS-C240 Network Interfaces

Figure 6: UCS-C240 Back-Plane



Number	Designation	Description	Applicable Node Types
1	CIMC/IPMI/M	The server's *Management* network interface used for accessing the UCS Cisco Integrated Management Controller (CIMC) application, performing Intelligent Platform Management Interface (IPMI) operations.	All

Number	Designation	Description	Applicable Node Types
2	Intel Onboard	Port 1: VIM Orchestration (Undercloud) *Provisioning* network interface.	All
		Port 2: *External* network interface for Internet access. It must also be routable to External floating IP addresses on other nodes.	Ultra M Manager Node Staging Server
3	Modular LAN on Motherboard (mLOM)	VIM networking interfaces used for:	
		• External floating IP network.	Controller
		• Internal API network	Controller
		• Storage network	Controller Compute OSD Compute Ceph
		• Storage Management network	Controller Compute OSD Compute Ceph
		• Tenant network (virtio only – VIM provisioning, VNF Management, and VNF Orchestration)	Controller Compute OSD Compute
4	PCIe 4	Port 1: With NIC bonding enabled, this port provides the active Service network interfaces for VNF ingress and egress connections.	Compute
		Port 2: With NIC bonding enabled, this port provides the standby *Di-internal* network interface for inter-VNF component communication.	Compute OSD Compute
5	PCIe 1	Port 1: With NIC bonding enabled, this port provides the active *Di-internal* network interface for inter-VNF component communication.	Compute OSD Compute
		Port 2: With NIC bonding enabled, this port provides the standby Service network interfaces for VNF ingress and egress connections.	Compute

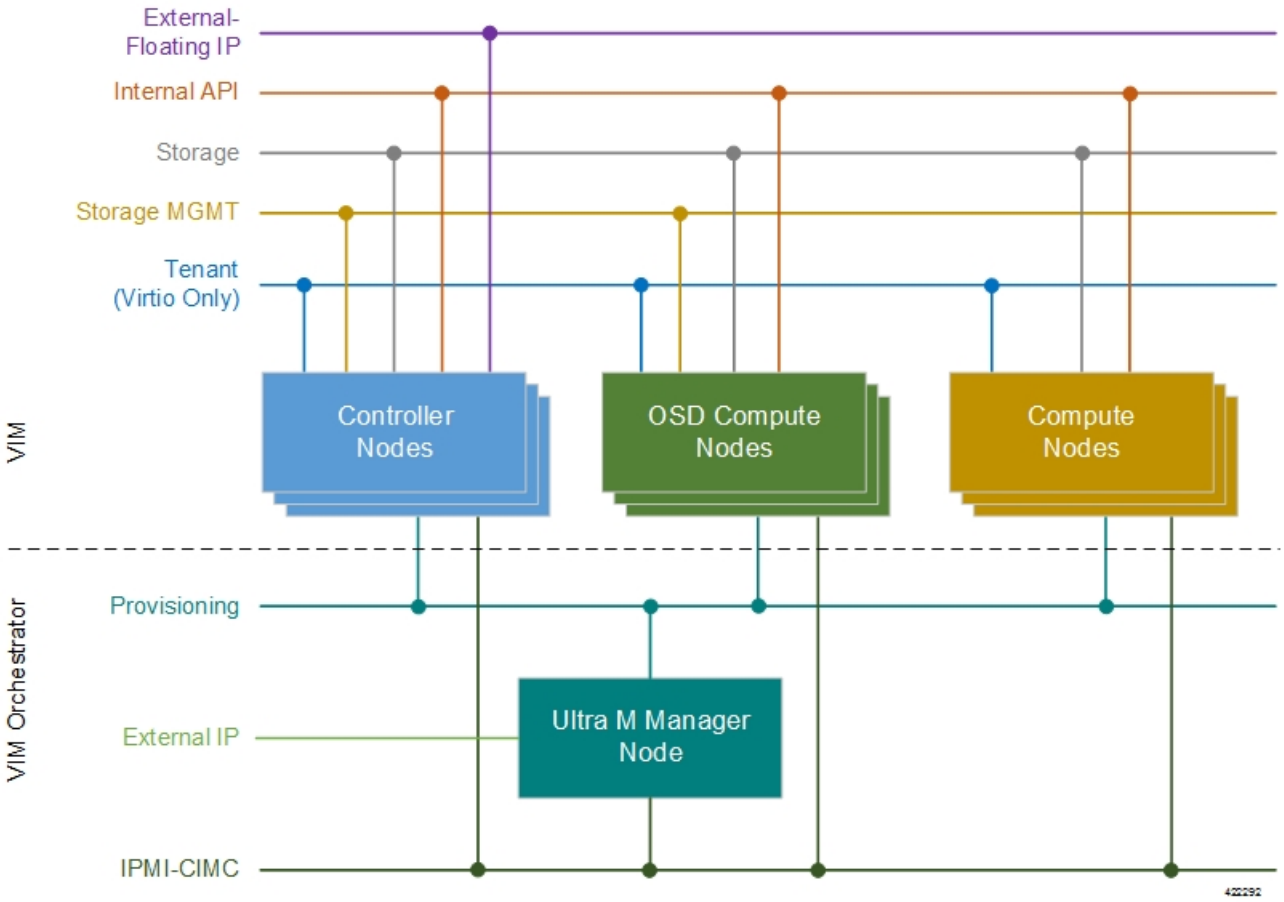
VIM Network Topology

Ultra M's VIM is based on the OpenStack project TripleO ("OpenStack-On-OpenStack") which is the core of the OpenStack Platform Director (OSP-D). TripleO allows OpenStack components to install a fully operational OpenStack environment.

Two cloud concepts are introduced through TripleO:

- **VIM Orchestrator (Undercloud)**: The VIM Orchestrator is used to bring up and manage the VIM. Though OSP-D and Undercloud are sometimes referred to synonymously, the OSP-D bootstraps the Undercloud deployment and provides the underlying components (e.g. Ironic, Nova, Glance, Neutron, etc.) leveraged by the Undercloud to deploy the VIM. Within the Ultra M Solution, OSP-D and the Undercloud are hosted on the same server.
- **VIM (Overcloud)**: The VIM consists of the compute, controller, and storage nodes on which the VNFs are deployed.

Figure 7: Hyper-converged Ultra M Single and Multi-VNF Model OpenStack VIM Network Topology



Some considerations for VIM Orchestrator and VIM deployment are as follows:

- External network access (e.g. Internet access) can be configured in one of the following ways:

  - Across all node types: A single subnet is configured on the Controller HA, VIP address, floating IP addresses and OSP-D/Staging server's external interface provided that this network is data-center routable as well as it is able to reach the internet.
  - Limited to OSP-D: The *External IP* network is used by Controllers for HA and Horizon dashboard as well as later on for Tenant Floating IP address requirements. This network must be data-center routable. In addition, the *External IP* network is used only by OSP-D/Staging Server node's external interface that has a single IP address. The *External IP* network must be lab/data-center routable must also have internet access to Red Hat cloud. It is used by OSP-D/Staging Server for subscription purposes and also acts as an external gateway for all controllers, computes and Ceph-storage nodes.

- IPMI must be enabled on all nodes.
- Two networks are needed to deploy the VIM Orchestrator:
  - IPMI/CIMC Network
  - Provisioning Network
- The OSP-D/Staging Server must have reachability to both IPMI/CIMC and Provisioning Networks. (VIM Orchestrator networks need to be routable between each other or have to be in one subnet.)
- DHCP-based IP address assignment for Introspection PXE from Provisioning Network (Range A)
- DHCP based IP address assignment for VIM PXE from Provisioning Network (Range B) must be separate from Introspection.
- The Ultra M Manager Node/Staging Server acts as a gateway for Controller, Ceph and Computes. Therefore, the external interface of this node/server needs to be able to access the Internet. In addition, this interface needs to be routable with the Data-center network. This allows the External interface IP-address of the Ultra M Manager Node/Staging Server to reach Data-center routable Floating IP addresses as well as the VIP addresses of Controllers in HA Mode.
- Prior to assigning floating and virtual IP addresses, make sure that they are not already allocated through OpenStack. If the addresses are already allocated, then they must be freed up for use or you must assign a new IP address that is available in the VIM.
- Multiple VLANs are required in order to deploy OpenStack VIM:
  - 1 for the Management and Provisioning networks interconnecting all the nodes regardless of type
  - 1 for the Staging Server/OSP-D Node external network
  - 1 for Compute, Controller, and Ceph Storage or OSD Compute Nodes
  - 1 for Management network interconnecting the Leafs and Spines
- Login to individual Compute nodes will be from OSP-D/Staging Server using heat user login credentials.

  The OSP-D/Staging Server acts as a "jump server" where the br-ctlplane interface address is used to login to the Controller, Ceph or OSD Computes, and Computes post VIM deployment using heat-admin credentials.

Layer 1 networking guidelines for the VIM network are provided in Layer 1 Leaf and Spine Topology, on page 26. In addition, a template is provided in Network Definitions (Layer 2 and 3), on page 85 to assist you with your Layer 2 and Layer 3 network planning.

Openstack Tenant Networking

The interfaces used by the VNF are based on the PCIe architecture. Single root input/output virtualization (SR-IOV) is used on these interfaces to allow multiple VMs on a single server node to use the same network interface as shown in Figure 8: Physical NIC to Bridge Mappings, on page 23. SR-IOV Networking is network type *Flat* under OpenStack configuration. NIC Bonding is used to ensure port level redundancy for PCIe Cards involved in SR-IOV Tenant Networks as shown in Figure 9: NIC Bonding, on page 24.

Figure 8: Physical NIC to Bridge Mappings

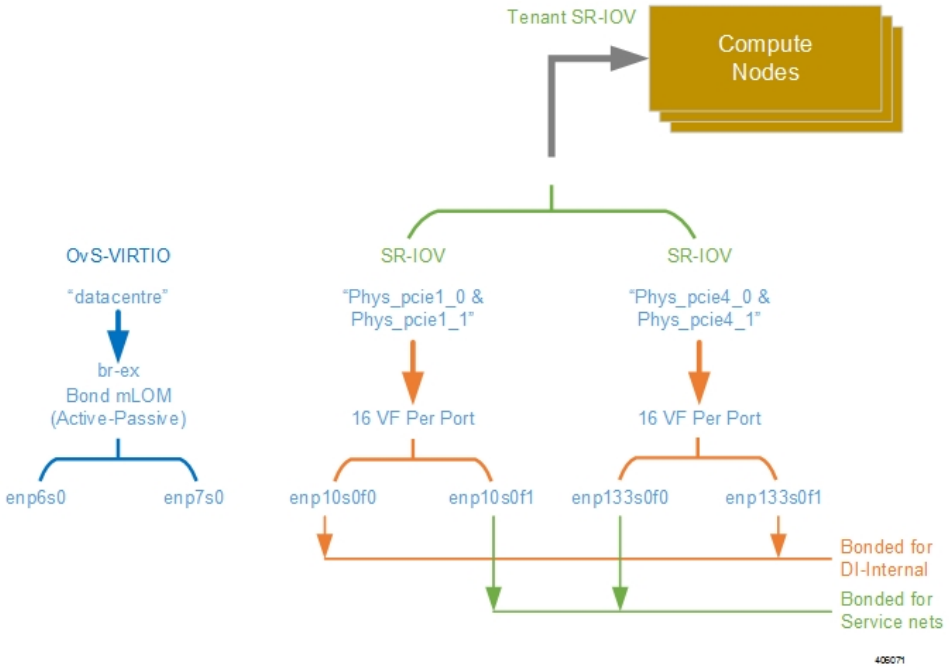
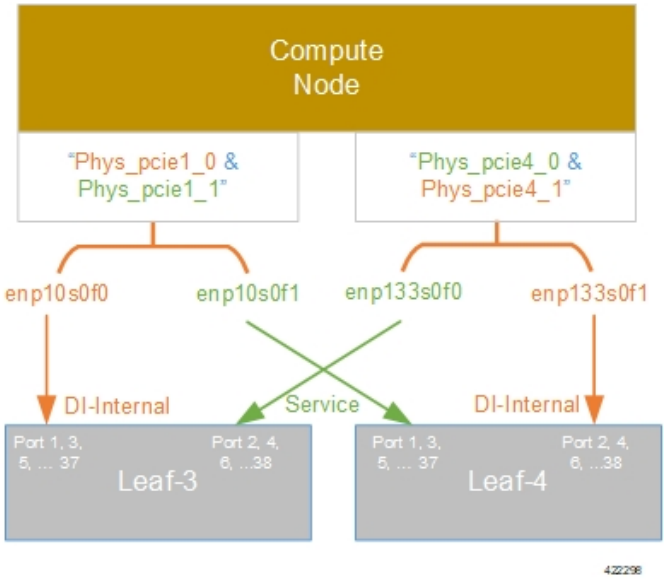


Figure 9: NIC Bonding

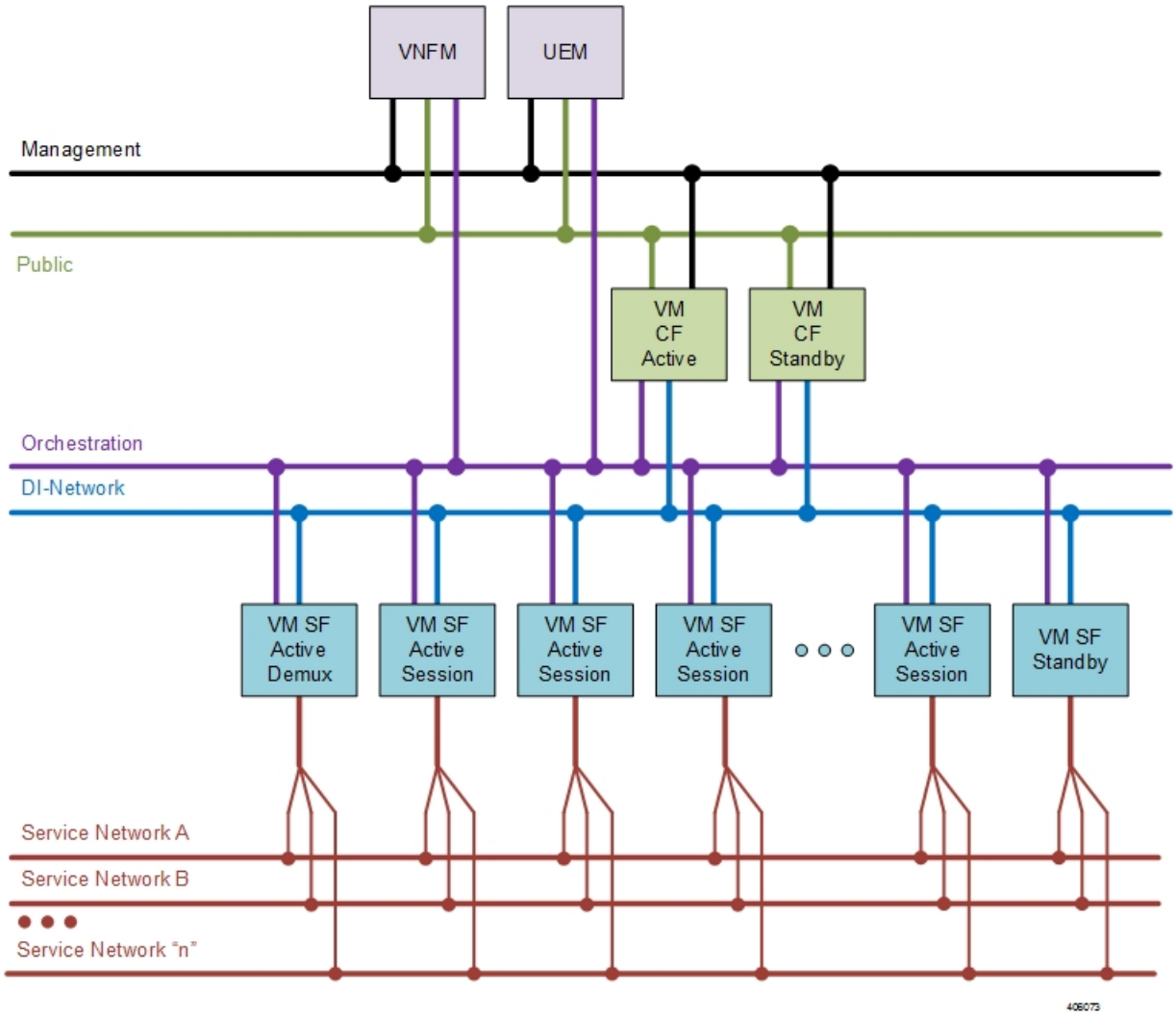


VNF Tenant Networks

While specific VNF network requirements are described in the documentation corresponding to the VNF, Figure 10: Typical USP-based VNF Networks, on page 25 displays the types of networks typically required by USP-based VNFs.

In this release, a cluster of UEM supports multiple VNF instances deployed in different tenants in a single "site" leveraging a single VIM.

Figure 10: Typical USP-based VNF Networks



The USP-based VNF networking requirements and the specific roles are described here:

- **Public**: *External public network*. The router has an external gateway to the public network. All other networks (except DI-Internal and ServiceA-*n*) have an internal gateway pointing to the router. And the router performs secure network address translation (SNAT).
- **DI-Internal**: This is the DI-internal network which serves as a 'backplane' for CF-SF and CF-CF communications. Since this network is internal to the UGP, it does not have a gateway interface to the router in the OpenStack network topology. A unique DI internal network must be created for each instance of the UGP. The interfaces attached to these networks use performance optimizations.
- **Management**: This is the local management network between the CFs and other management elements like the UEM and VNFM. This network is also used by OSP-D to deploy the VNFM and AutoVNF. To allow external access, an OpenStack floating IP address from the Public network must be associated with the UGP VIP (CF) address.

You can ensure that the same floating IP address can assigned to the CF, UEM, and VNFM after a VM restart by configuring parameters in the AutoDeploy configuration file or the UWS service delivery configuration file.

Note Prior to assigning floating and virtual IP addresses, make sure that they are not already allocated through OpenStack. If the addresses are already allocated, then they must be freed up for use or you must assign a new IP address that is available in the VIM.

- **Orchestration**: This is the network used for VNF deployment and monitoring. It is used by the VNFM to onboard the USP-based VNF.
- **ServiceA-*n***: These are the service interfaces to the SF. Up to 12 service interfaces can be provisioned for the SF with this release. The interfaces attached to these networks use performance optimizations.

  Layer 1 networking guidelines for the VNF network are provided in Layer 1 Leaf and Spine Topology, on page 26. In addition, a template is provided in Network Definitions (Layer 2 and 3), on page 85 to assist you with your Layer 2 and Layer 3 network planning.

Supporting Trunking on VNF Service ports

Service ports within USP-based VNFs are configured as trunk ports and traffic is tagged using the VLAN command. This configuration is supported by trunking to the uplink switch via the *sriovnicswitch mechanism* driver.

This driver supports Flat network types in OpenStack, enabling the guest OS to tag the packets.

Flat networks are untagged networks in OpenStack. Typically, these networks are previously existing infrastructure, where OpenStack guests can be directly applied.

Layer 1 Leaf and Spine Topology

Ultra M implements a Leaf and Spine network topology. Topology details differ between Ultra M models based on the scale and number of nodes.

Note When connecting component network ports, ensure that the destination ports are rated at the same speed as the source port (e.g. connect a 10G port to a 10G port). Additionally, the source and destination ports must support the same physical medium (e.g. Ethernet) for interconnectivity.

Hyper-converged Ultra M Single and Multi-VNF Model Network Topology

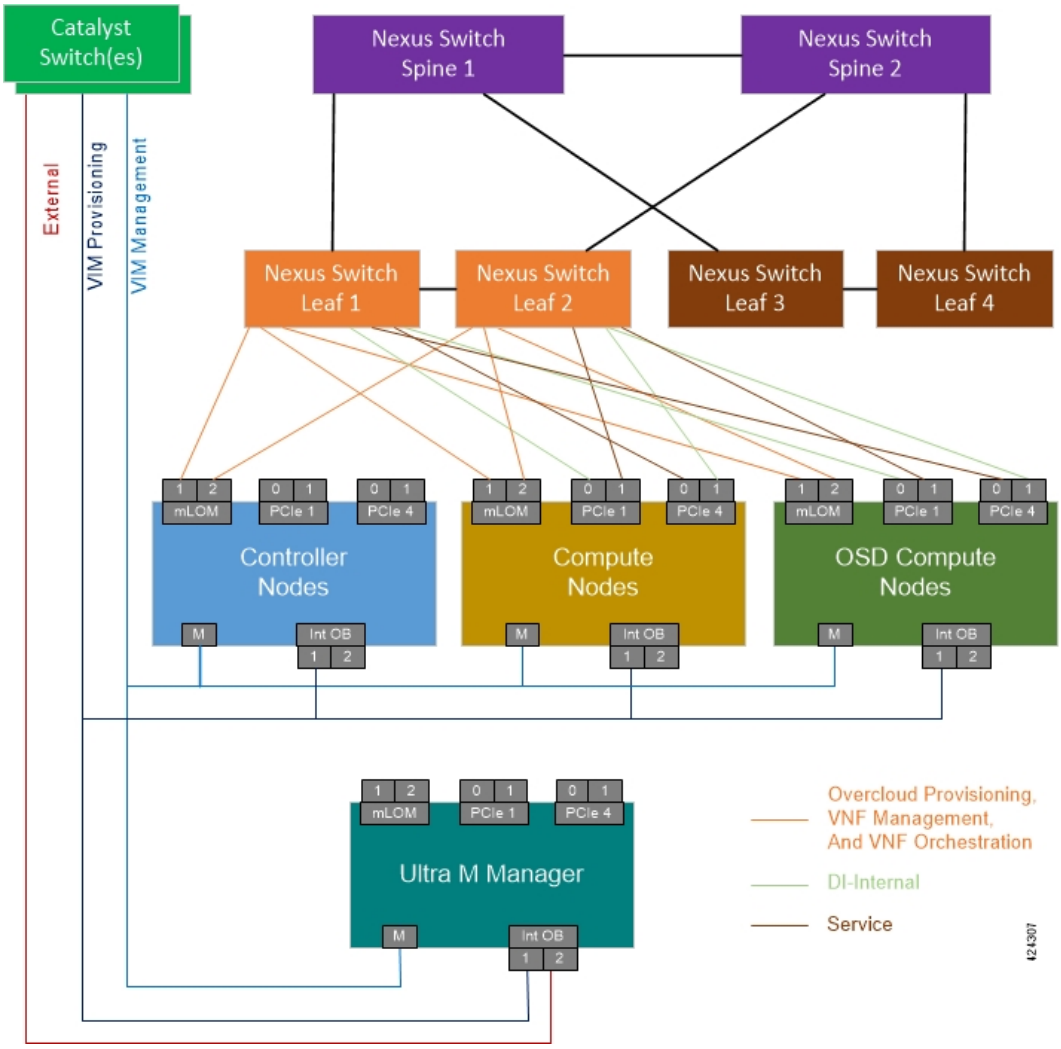
Figure 11: Hyper-converged Ultra M Single and Multi-VNF Leaf and Spine Topology, on page 27 illustrates the logical leaf and spine topology for the various networks required for the Hyper-converged Ultra M models.

In this figure, two VNFs are supported. (Leafs 1 and 2 pertain to VNF1, Leafs 3 and 4 pertain to VNF 2). If additional VNFs are supported, additional Leafs are required (e.g. Leafs 5 and 6 are needed for VNF 3, Leafs

7 and 8 for VNF4). Each set of additional Leafs would have the same meshed network interconnects with the Spines and with the Controller, OSD Compute, and Compute Nodes.

For single VNF models, Leaf 1 and Leaf 2 facilitate all of the network interconnects from the server nodes and from the Spines.

Figure 11: Hyper-converged Ultra M Single and Multi-VNF Leaf and Spine Topology



As identified in Cisco Nexus Switches, on page 6, the number of leaf and spine switches differ between the Ultra M models. Similarly, the specific leaf and spine ports used also depend on the Ultra M solution model being deployed. That said, general guidelines for interconnecting the leaf and spine switches in an Ultra M XS multi-VNF deployment are provided in Table 14: Catalyst Management Switch 1 (Rack 1) Port Interconnects, on page 28 through Table 23: Spine 2 Port Interconnect Guidelines, on page 41. Using the information in these tables, you can make appropriate adjustments to your network topology based on your deployment scenario (e.g. number of VNFs and number of Compute Nodes).

Table 14: Catalyst Management Switch 1 (Rack 1) Port Interconnects

From Switch Port(s)	To			Notes
	Device	Network	Port(s)	
1	Ultra M Manager Node	Management	CIMC	Management Switch 1 only
3, 12	OSD Compute Nodes	Management	CIMC	3 non-sequential ports - 1 per OSD Compute Node OSD Compute Node 0 connects to port 2 OSD Compute Node 1 connects to port 3 OSD Compute Node 2 connects to port 12
4-11 (inclusive)	Compute Nodes	Management	CIMC	8 sequential ports - 1 per Compute Node; Compute Nodes 0-7 connect to ports 4 through 11 respectively
13	Controller 0	Management	CIMC	
15	Cat Management Switch 2	Management	39	
25	Ultra M Manager Node (OSPD VM)	Provisioning	Mgmt	**NOTE**: This is for deployments of Ultra M Manager Node on bare metal.
26, 27. 36	OSD Compute Nodes	Provisioning	Mgmt	3 non-sequential ports - 1 per OSD Compute Node OSD Compute Node 0 connects to port 26 OSD Compute Node 1 connects to port 27 OSD Compute Node 2 connects to port 36
28-35 (inclusive)	Compute Nodes	Provisioning	Mgmt	8 sequential ports - 1 per Compute Node; Compute Nodes 0-7 connect to ports 28-35 respectively
37	Controller 0	Management	Mgmt	
43	Leaf 1	Management	48	Switch port 43 connects with Leaf 1 port 48
44	Leaf 2	Management	48	Switch port 44 connects with Leaf 2 port 48
45	Leaf 1	Management	Mgmt	
46	Leaf 2	Management	Mgmt	
47	Spine 1	Management	Mgmt	

Table 15: Catalyst Management Switch 2 (Rack 2) Port Interconnects

From Switch Port(s)	To			Notes
	Device	Network	Port(s)	
1-10	Compute Nodes	Management	CIMC	10 sequential ports - 1 per Compute Node; Compute Nodes 8-17 connect to ports 1-10 respectively
11	Controller 2	Management	CIMC	
12	Controller 1	Management	CIMC	
13-22	Compute Nodes	Provisioning	Mgmt	10 sequential ports - 1 per Compute Node Compute Nodes 8 17 connect to ports 13-22 respectively
23	Controller 2	Provisioning	Mgmt	
24	Controller 1	Provisioning	Mgmt	
39	Cat Management Switch 1	Management	15	
43	Leaf 3	Management	48	Switch port 43 connects with Leaf 3 port 48
44	Leaf 4	Management	48	Switch port 44 connects with Leaf 4 port 48
45	Leaf 3	Management	Mgmt	
46	Leaf 4	Management	Mgmt	
47	Spine 2	Management	Mgmt	

Table 16: Catalyst Management Switch 3 (Rack 3) Port Interconnects

From Switch Port(s)	To			Notes
	Device	Network	Port(s)	
1-10	Compute Nodes	Management	CIMC	10 sequential ports - 1 per Compute Node; Compute Nodes 18-27 connect to ports 1-10 respectively
13-22	Compute Nodes	Provisioning	Mgmt	10 sequential ports - 1 per Compute Node; Compute Nodes 18-27 connect to ports 13-22 respectively

From Switch Port(s)	To			Notes
	Device	Network	Port(s)	
15	Cat Management Switch 4	Management	39	
43	Leaf 5	Management	48	Switch port 43 connects with Leaf 5 port 48
44	Leaf 6	Management	48	Switch port 44 connects with Leaf 6 port 48
45	Leaf 5	Management	Mgmt	
46	Leaf 6	Management	Mgmt	
47	Spine 1	Management	Mgmt	

Table 17: Catalyst Management Switch 4 (Rack 4) Port Interconnects

From Switch Port(s)	To			Notes
	Device	Network	Port(s)	
1-10	Compute Nodes	Management	CIMC	10 sequential ports - 1 per Compute Node; Compute Nodes 28-37 connect to ports 1-10 respectively
13-22	Compute Nodes	Provisioning	Mgmt	10 sequential ports - 1 per Compute Node; Compute Nodes 28-37 connect to ports 13-22 respectively
39	Cat Management Switch 3	Management	15	
43	Leaf 7	Management	48	Switch port 43 connects with Leaf 7 port 48
44	Leaf 8	Management	48	Switch port 44 connects with Leaf 8 port 48
45	Leaf 7	Management	Mgmt	
46	Leaf 8	Management	Mgmt	
47	Spine 2	Management	Mgmt	

*Table 18: Leaf 1 and 2 (Rack 1) Port Interconnects**

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
Leaf 1				

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
Mgmt	Cat Management Switch 1	Management	45	
1	Controller 0 Node	Management & Orchestration (active)	MLOM P1	
3, 12, 13	OSD Compute Nodes	Management & Orchestration (active)	MLOM P1	3 non-sequential ports - 1 per OSD Compute Node: OSD Compute Node 2 connects to port 3 OSD Compute Node 1 connects to port 12 OSD Compute Node 0 connects to port 13
4-11 (inclusive)	Compute Nodes	Management & Orchestration (active)	MLOM P1	Sequential ports based on the number of Compute Nodes - 1 per Compute Node; Compute Nodes 0-7 connect to ports 11-4 respectively
18, 27, 28	OSD Compute Nodes	Di-internal (active)	PCIe01 P1	3 non-sequential ports - 1 per OSD Compute Node OSD Compute Node 2 connects to port 18 OSD Compute Node 1 connects to port 27 OSD Compute Node 0 connects to port 28
19-26 (inclusive)	Compute Nodes	Di-internal (active)	PCIe01 P1	Sequential ports based on the number of Compute Nodes - 1 per Compute Node; Compute Nodes 0-7 connect to ports 26-19 respectively

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
33-42 (inclusive)	Compute Nodes / OSD Compute Nodes	Service (active)	PCIe04 P1	Sequential ports based on the number of Compute Nodes and/or OSD Compute Nodes - 1 per OSD Compute Node and/or Compute Node OSD Compute Node 2 connects to port 33 Compute Nodes 0-7 connect to ports 41-34 respectively OSD Compute Node 1 connects to port 42 OSD Compute Node 0 connects to port 43 **Note** Though the OSD Compute Nodes do not use the Service Networks, they are provided to ensure compatibility within the OpenStack Overcloud (VIM) deployment.
48	Catalyst Management Switches	Management	43	Leaf 1 connects to Switch 1
49-50	Spine 1	Downlink	1-2	Leaf 1 port 49 connects to Spine 1 port 1 Leaf 1 port 50 connects to Spine 1 port 2
53-54	Leaf 2	Downlink	53-54	Leaf 1 port 53 connects to Leaf 2 port 53
Leaf 2				
Mgmt	Cat Management Switch 1	Management	46	
1	Controller 0 Node	Management & Orchestration (redundant)	MLOM P2	
3, 12, 13	OSD Compute Nodes	Management & Orchestration (redundant)	MLOM P2	3 non-sequential ports - 1 per OSD Compute Node: OSD Compute Node 2 connects to port 3 OSD Compute Node 1 connects to port 12 OSD Compute Node 0 connects to port 13
4-11 (inclusive)	Compute Nodes	Management & Orchestration (redundant)	MLOM P2	Sequential ports based on the number of Compute Nodes - 1 per Compute Node; Compute Nodes 0-7 connect to ports 11-4 respectively

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
18-28 (inclusive)	Compute Nodes / OSD Compute Nodes	Service (redundant)	PCIe01 P2	Sequential ports based on the number of Compute Nodes and/or OSD Compute Nodes - 1 per OSD Compute Node and/or Compute Node: OSD Compute Node 2 connects to port 18 Compute Nodes 0-7 connect to ports 26-19 respectively OSD Compute Node 1 connects to port 27 OSD Compute Node 0 connects to port 28 **Important** NOTE: Though the OSD Compute Nodes do not use the Service Networks, they are provided to ensure compatibility within the OpenStack Overcloud (VIM) deployment.
33, 42, 43	OSD Compute Nodes	Di-internal (redundant)	PCIe04 P2	3 non-sequential ports - 1 per OSD Compute Node OSD Compute Node 2 connects to port 33 OSD Compute Node 1 connects to port 42 OSD Compute Node 0 connects to port 43
34-41 (inclusive)	Compute Nodes	Di-internal (redundant)	PCIe04 P2	Sequential ports based on the number of Compute Nodes - 1 per Compute Node; Compute Nodes 0-7 connect to ports 34-41 respectively
48	Catalyst Management Switches	Management	44	Leaf 2 connects to Switch 1
49-50	Spine 2	Downlink	1-2	Leaf 2 port 49 connects to Spine 2 port 1 Leaf 2 port 50 connects to Spine 2 port 2
53-54	Leaf 1	Downlink	53-54	Leaf 2 port 53 connects to Leaf 1 port 53 Leaf 2 port 54 connects to Leaf 1 port 54

Table 19: Leaf 3 and 4 (Rack 2) Port Interconnects

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
Leaf 3				

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
Mgmt	Cat Management Switch 2	Management	45	
1 - 10 (inclusive)	Compute Nodes	Management & Orchestration (active)	MLOM P1	Sequential ports based on the number of Compute Nodes - 1 per Compute Node **Important** Leaf Ports 1 and 2 are used for the first two Compute Nodes on VNFs other than VNF1 (Rack 1). These are used to host management-related VMs as shown in Figure 3: VM Distribution on Server Nodes for Hyper-converged Ultra M Multi-VNF Models, on page 10.
13-14 (inclusive)	Controller Nodes	Management & Orchestration (active)	MLOM P1	Leaf 3 port 13 connects to Controller 1 MLOM P1 port Leaf 3 port 14 connects to Controller 1 MLOM P1 port
17-26 (inclusive)	Compute Nodes	Di-internal (active)	PCIe01 P1	Sequential ports based on the number of Compute Nodes - 1 per Compute Node **Important** Leaf Ports 17 and 18 are used for the first two Compute Nodes on VNFs other than VNF1. These are used to host management-related VMs as shown in Figure 3: VM Distribution on Server Nodes for Hyper-converged Ultra M Multi-VNF Models, on page 10.
33-42 (inclusive)	Compute Nodes	Service (active)	PCIe04 P1	Sequential ports based on the number of Compute Nodes - 1 per Compute Node
48	Catalyst Management Switches	Management	43	Leaf 3 connects to Switch 2
49-50	Spine 1	Downlink	5-6	Leaf 3 port 49 connects to Spine 1 port 5 Leaf 3 port 50 connects to Spine 1 port 6

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
53-54	Leaf 4	Downlink	53-54	Leaf 3 port 53 connects to Leaf 4 port 53 Leaf 3 port 54 connects to Leaf 4 port 54
Leaf 4				
Mgmt	Cat Management Switch 2	Management	46	
1 - 10 (inclusive)	Compute Nodes	Management & Orchestration (redundant)	MLOM P2	Sequential ports based on the number of Compute Nodes - 1 per Compute Node **Important** Leaf Ports 1 and 2 are used for the first two Compute Nodes on VNFs other than VNF1. These are used to host management-related VMs as shown in Figure 3: VM Distribution on Server Nodes for Hyper-converged Ultra M Multi-VNF Models, on page 10.
13-14 (inclusive)	Controller Nodes	Management & Orchestration (redundant)	MLOM P2	Leaf 4 port 13 connects to Controller 1 MLOM P2 port Leaf 4 port 14 connects to Controller 1 MLOM P2 port
17-26 (inclusive)	Compute Nodes	Di-internal (redundant)	PCIe04 P2	Sequential ports based on the number of Compute Nodes - 1 per Compute Node **Important** Leaf Ports 17 and 18 are used for the first two Compute Nodes on VNFs other than VNF1. These are used to host management-related VMs as shown in Figure 3: VM Distribution on Server Nodes for Hyper-converged Ultra M Multi-VNF Models, on page 10.
33-42 (inclusive)	Compute Nodes	Service (redundant)	PCIe01 P2	Sequential ports based on the number of Compute Nodes - 1 per Compute Node
48	Catalyst Management Switches	Management	44	Leaf 4 connects to Switch 2

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
49-50	Spine 2	Downlink	5-6	Leaf 4 port 49 connects to Spine 2 port 5 Leaf 4 port 50 connects to Spine 2 port 6
53-54	Leaf 3	Downlink	53-54	Leaf 4 port 53 connects to Leaf 3 port 53 Leaf 4 port 54 connects to Leaf 3 port 54

Table 20: Leaf 5 and 6 (Rack 3) Port Interconnects

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
Leaf 5				
Mgmt	Cat Management Switch 3	Management	45	
1 - 10 (inclusive)	Compute Nodes	Management & Orchestration (active)	MLOM P1	Sequential ports based on the number of Compute Nodes - 1 per Compute Node **Important** Leaf Ports 1 and 2 are used for the first two Compute Nodes on VNFs other than VNF1. These are used to host management-related VMs as shown in Figure 3: VM Distribution on Server Nodes for Hyper-converged Ultra M Multi-VNF Models, on page 10.
17-26 (inclusive)	Compute Nodes	Di-internal (active)	PCIe01 P1	Sequential ports based on the number of Compute Nodes - 1 per Compute Node **Important** Leaf Ports 17 and 18 are used for the first two Compute Nodes on VNFs other than VNF1. These are used to host management-related VMs as shown in Figure 3: VM Distribution on Server Nodes for Hyper-converged Ultra M Multi-VNF Models, on page 10.
33-42 (inclusive)	Compute Nodes	Service (active)	PCIe04 P1	Sequential ports based on the number of Compute Nodes - 1 per Compute Node

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
48	Catalyst Management Switches	Management	47	Leaf 5 connects to Switch 3
49-50	Spine 1	Downlink	9-10	Leaf 5 port 49 connects to Spine 1 port 9 Leaf 5 port 50 connects to Spine 1 port 10
53-54	Leaf 6	Downlink	53-54	Leaf 5 port 53 connects to Leaf 6 port 53 Leaf 5 port 54 connects to Leaf 6 port 54
Leaf 6				
Mgmt	Cat Management Switch 3	Management	46	
1 - 10 (inclusive)	Compute Nodes	Management & Orchestration (redundant)	MLOM P2	Sequential ports based on the number of Compute Nodes - 1 per Compute Node **Important** Leaf Ports 1 and 2 are used for the first two Compute Nodes on VNFs other than VNF1. These are used to host management-related VMs as shown in Figure 3: VM Distribution on Server Nodes for Hyper-converged Ultra M Multi-VNF Models, on page 10.
17-26 (inclusive)	Compute Nodes	Di-internal (redundant)	PCIe04 P2	Sequential ports based on the number of Compute Nodes - 1 per Compute Node **Important** Leaf Ports 17 and 18 are used for the first two Compute Nodes on VNFs other than VNF1. These are used to host management-related VMs as shown in Figure 3: VM Distribution on Server Nodes for Hyper-converged Ultra M Multi-VNF Models, on page 10.
33-42 (inclusive)	Compute Nodes	Service (redundant)	PCIe01 P2	Sequential ports based on the number of Compute Nodes - 1 per Compute Node

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
48	Catalyst Management Switches	Management	48	Leaf 6 connects to Switch 3
49-50	Spine 2	Downlink	9-10	Leaf 6 port 49 connects to Spine 2 port 9 Leaf 6 port 50 connects to Spine 2 port 10
53-54	Leaf 5	Downlink	53-54	Leaf 6 port 53 connects to Leaf 5 port 53 Leaf 6 port 54 connects to Leaf 5 port 54

Table 21: Leaf 7 and 8 (Rack 4) Port Interconnects

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
Leaf 7				
Mgmt	Cat Management Switch 4	Management	45	
1 - 10 (inclusive)	Compute Nodes	Management & Orchestration (active)	MLOM P1	Sequential ports based on the number of Compute Nodes - 1 per Compute Node **Important** Leaf Ports 1 and 2 are used for the first two Compute Nodes on VNFs other than VNF1. These are used to host management-related VMs as shown in Figure 3: VM Distribution on Server Nodes for Hyper-converged Ultra M Multi-VNF Models, on page 10.

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
17-26 (inclusive)	Compute Nodes	Di-internal (active)	PCIe01 P1	Sequential ports based on the number of Compute Nodes - 1 per Compute Node **Important** Leaf Ports 17 and 18 are used for the first two Compute Nodes on VNFs other than VNF1. These are used to host management-related VMs as shown in Figure 3: VM Distribution on Server Nodes for Hyper-converged Ultra M Multi-VNF Models, on page 10.
33-42 (inclusive)	Compute Nodes	Service (active)	PCIe04 P1	Sequential ports based on the number of Compute Nodes - 1 per Compute Node
48	Catalyst Management Switches	Management	47	Leaf 7 connects to Switch 4
49-50	Spine 1	Downlink	13-14	Leaf 7 port 49 connects to Spine 1 port 13 Leaf 7 port 50 connects to Spine 1 port 14
53-54	Leaf 8	Downlink	53-54	Leaf 7 port 53 connects to Leaf 8 port 53 Leaf 7 port 54 connects to Leaf 8 port 54
Leaf 8				
Mgmt	Cat Management Switch 3	Management	46	
1 - 10 (inclusive)	Compute Nodes	Management & Orchestration (redundant)	MLOM P2	Sequential ports based on the number of Compute Nodes - 1 per Compute Node **Important** Leaf Ports 1 and 2 are used for the first two Compute Nodes on VNFs other than VNF1. These are used to host management-related VMs as shown in Figure 3: VM Distribution on Server Nodes for Hyper-converged Ultra M Multi-VNF Models, on page 10.

From Leaf Port(s)	To			Notes
	Device	Network	Port(s)	
17-26 (inclusive)	Compute Nodes	Di-internal (redundant)	PCIe04 P2	Sequential ports based on the number of Compute Nodes - 1 per Compute Node **Important** Leaf Ports 17 and 18 are used for the first two Compute Nodes on VNFs other than VNF1. These are used to host management-related VMs as shown in Figure 3: VM Distribution on Server Nodes for Hyper-converged Ultra M Multi-VNF Models, on page 10#unique_14 unique_14_Connect_42_fig_sy3_vth_qcb.
33-42 (inclusive)	Compute Nodes	Service (redundant)	PCIe01 P2	Sequential ports based on the number of Compute Nodes - 1 per Compute Node
48	Catalyst Management Switches	Management	48	Leaf 8 connects to Switch 4
49-50	Spine 2	Downlink	13-14	Leaf 8 port 49 connects to Spine 2 port 13 Leaf 8 port 50 connects to Spine 2 port 14
53-54	Leaf 7	Downlink	53-54	Leaf 8 port 53 connects to Leaf 7 port 53 Leaf 8 port 54 connects to Leaf 7 port 54

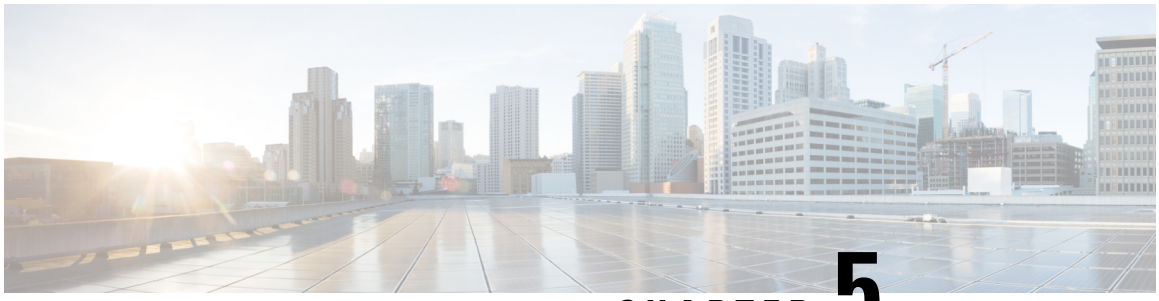
Table 22: Spine 1 Port Interconnect Guidelines

From Spine Port(s)	To			Notes
	Device	Network	Port(s)	
1-2, 5-6, 9-10, 13-14	Leaf 1, 3, 5, 7	Downlink	49-50	Spine 1 ports 1 and 2 connect to Leaf 1 ports 49 and 50 respectively Spine 1 ports 5 and 6 connect to Leaf 3 ports 49 and 50 respectively Spine 1 ports 9 and 10 connect to Leaf 5 ports 49 and 50 respectively Spine 1 ports 13 and 14 connect to Leaf 7 ports 49 and 50 respectively

From Spine Port(s)	To			Notes
	Device	Network	Port(s)	
29-30, 31, 32, 33-34	Spine 2	Interlink	29-30, 31, 32, 33-34	Spine 1 ports 29-30 connect to Spine 2 ports 29-30 respectively Spine 1 port 31 connects to Spine 2 port 31 respectively Spine 1 port 32 connects to Spine 2 port 32 respectively Spine 1 ports 33-34 connect to Spine 2 ports 33-34 respectively
21-22, 23-24, 25-26	Router	Uplink	-	

Table 23: Spine 2 Port Interconnect Guidelines

From Spine Port(s)	To			Notes
	Device	Network	Port(s)	
3-4, 7-8, 11-12, 15-16	Leaf 2, 4, 6, 8	Downlink	51-52	Spine 2 ports 3 and 4 connect to Leaf 2 ports 51 and 52 respectively Spine 2 ports 7 and 8 connect to Leaf 4 ports 51 and 52 respectively Spine 2 ports 11 and 12 connect to Leaf 6 ports 51 and 52 respectively Spine 2 ports 15 and 16 connect to Leaf 8 ports 51 and 52 respectively
29-30, 31, 32, 33-34	Spine 1	Interconnect	29-30, 31, 32, 33-34	Spine 2 ports 29-30 connect to Spine 1 ports 29-30 respectively Spine 2 port 31 connects to Spine 1 port 31 Spine 2 port 32 connects to Spine 1 port 32 Spine 2 ports 33-34 connect to Spine 1 ports 33-34
21-22, 23-24, 25-26	Router	Uplink	-	



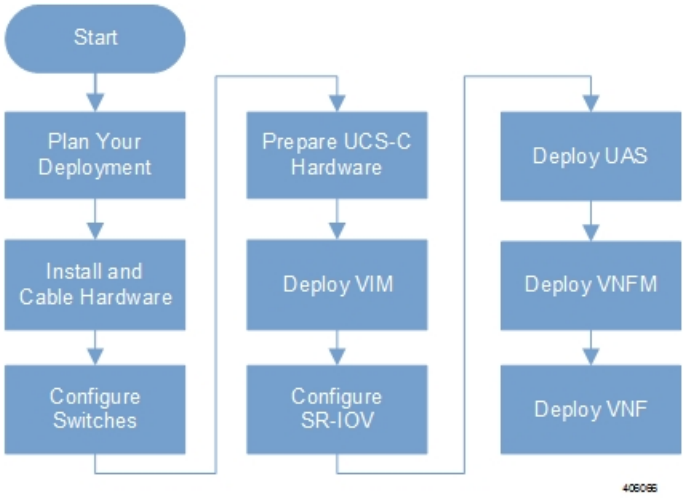
CHAPTER 5

Deploying the Ultra M Solution

Ultra M is a multi-product solution. Detailed instructions for installing each of these products is beyond the scope of this document. Instead, the sections that follow identify the specific, non-default parameters that must be configured through the installation and deployment of those products in order to deploy the entire solution.

Deployment Workflow

Figure 12: Ultra M Deployment Workflow



Plan Your Deployment

Before deploying the Ultra M solution, it is very important to develop and plan your deployment.

Network Planning

Networking Overview, on page 19 provides a general overview and identifies basic requirements for networking the Ultra M solution.

With this background, use the tables in Network Definitions (Layer 2 and 3), on page 85 to help plan the details of your network configuration.

Install and Cable the Hardware

This section describes the procedure to install all the components included in the Ultra M Solution.

Related Documentation

To ensure hardware components of the Ultra M solution are installed properly, refer to the installation guides for the respective hardware components.

- **Catalyst 2960-XR Switch** — http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst2960xr/hardware/installation/guide/b_c2960xr_hig.html
- **Catalyst 3850 48T-S Switch** — http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst3850/hardware/installation/guide/b_c3850_hig.html
- **Nexus 93180-YC 48 Port** — http://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus9000/hw/n93180ycex_hig/guide/b_n93180ycex_nxos_mode_hardware_install_guide.html
- **Nexus 9236C 36 Port** — http://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus9000/hw/n9236c_hig/guide/b_c9236c_nxos_mode_hardware_install_guide.html
- **UCS C240 M4SX Server** — http://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/c/hw/C240M4/install/C240M4.html
- **UCS C220 M5SX Server** — https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/c/hw/C220M5/install/C220M5.html

Rack Layout

Hyper-converged Ultra M XS Single VNF Deployment

Table 24: Hyper-converged Ultra M XS Single VNF Deployment Rack Layout, on page 45 provides details for the recommended rack layout for the Hyper-converged Ultra M XS Single VNF deployment model.

Table 24: Hyper-converged Ultra M XS Single VNF Deployment Rack Layout

	Rack #1	Rack #2
RU-1	Empty	Empty
RU-2	Spine EOR Switch A: Nexus 9236C	Spine EOR Switch B: Nexus 9236C
RU-3	Empty	Empty
RU-4	VNF Mgmt Switch: Catalyst C3850-48T-S OR C2960XR-48TD	Empty
RU-5	VNF Leaf TOR Switch A: Nexus 93180YC-EX	Empty
RU-6	VNF Leaf TOR Switch B: Nexus 93180YC-EX	Empty
RU-7/8	Ultra UEM 1A: UCS C240 M4 SFF	Empty
RU-9/10	Ultra UEM 1B: UCS C240 M4 SFF	Empty
RU-11/12	Empty	Empty
RU-13/14	Demux SF: UCS C240 M4 SFF	Empty
RU-15/16	Standby SF: UCS C240 M4 SFF	Empty
RU-17/18	Active SF 1: UCS C240 M4 SFF	Empty
RU-19/20	Active SF 2: UCS C240 M4 SFF	Empty
RU-21/22	Active SF 3: UCS C240 M4 SFF	Empty
RU-23/24	Active SF 4: UCS C240 M4 SFF	Empty
RU-25/26	Active SF 5: UCS C240 M4 SFF	Empty
RU-27/28	Active SF 6: UCS C240 M4 SFF	Empty
RU-29/30	Empty	Empty
RU-31/32	Empty	Empty
RU-33/34	Empty	Empty
RU-35/36	Ultra UEM 1C	OpenStack Control C: UCS C240 M4 SFF
RU-37/38	Ultra M Manager: UCS C240 M4 SFF	Empty
RU-39/40	OpenStack Control A: UCS C240 M4 SFF	OpenStack Control B: UCS C240 M4 SFF
RU-41/42	Empty	Empty

	Rack #1	Rack #2
Cables	Controller Rack Cables	Controller Rack Cables
Cables	Spine Uplink/Interconnect Cables	Spine Uplink/Interconnect Cables
Cables	Leaf TOR To Spine Uplink Cables	Empty
Cables	VNF Rack Cables	Empty

Hyper-converged Ultra M XS Multi-VNF Deployment

Table 25: Hyper-converged Ultra M XS Multi-VNF Deployment Rack Layout, on page 46 provides details for the recommended rack layout for the Hyper-converged Ultra M XS Multi-VNF deployment model.

Table 25: Hyper-converged Ultra M XS Multi-VNF Deployment Rack Layout

	Rack #1	Rack #2	Rack #3	Rack #4
RU-1	Empty	Empty	Empty	Empty
RU-2	Spine EOR Switch A: Nexus 9236C	Spine EOR Switch B: Nexus 9236C	Empty	Empty
RU-3	Empty	Empty	Empty	Empty
RU-4	VNF Mgmt Switch: Catalyst C3850-48T-S OR C2960XR-48TD	VNF Mgmt Switch: Catalyst C3850-48T-S OR C2960XR-48TD	VNF Mgmt Switch: Catalyst C3850-48T-S OR C2960XR-48TD	VNF Mgmt Switch: Catalyst C3850-48T-S OR C2960XR-48TD
RU-5	VNF Leaf TOR Switch A: Nexus 93180YC-EX	VNF Leaf TOR Switch A: Nexus 93180YC-EX	VNF Leaf TOR Switch A: Nexus 93180YC-EX	VNF Leaf TOR Switch A: Nexus 93180YC-EX
RU-6	VNF Leaf TOR Switch B: Nexus 93180YC-EX	VNF Leaf TOR Switch B: Nexus 93180YC-EX	VNF Leaf TOR Switch B: Nexus 93180YC-EX	VNF Leaf TOR Switch B: Nexus 93180YC-EX
RU-7/8	Ultra UEM 1A: UCS C240 M4 SFF	Ultra UEM 2A: UCS C240 M4 SFF	Ultra UEM 3A: UCS C240 M4 SFF	Ultra UEM 4A: UCS C240 M4 SFF
RU-9/10	Ultra UEM 1B: UCS C240 M4 SFF	Ultra UEM 2B: UCS C240 M4 SFF	Ultra UEM 3B: UCS C240 M4 SFF	Ultra UEM 4B: UCS C240 M4 SFF
RU-11/12	Empty	Empty	Empty	Empty
RU-13/14	Demux SF: UCS C240 M4 SFF	Demux SF: UCS C240 M4 SFF	Demux SF: UCS C240 M4 SFF	Demux SF: UCS C240 M4 SFF
RU-15/16	Standby SF: UCS C240 M4 SFF	Standby SF: UCS C240 M4 SFF	Standby SF: UCS C240 M4 SFF	Standby SF: UCS C240 M4 SFF

	Rack #1	Rack #2	Rack #3	Rack #4
RU-17/18	Active SF 1: UCS C240 M4 SFF	Active SF 1: UCS C240 M4 SFF	Active SF 1: UCS C240 M4 SFF	Active SF 1: UCS C240 M4 SFF
RU-19/20	Active SF 2: UCS C240 M4 SFF	Active SF 2: UCS C240 M4 SFF	Active SF 2: UCS C240 M4 SFF	Active SF 2: UCS C240 M4 SFF
RU-21/22	Active SF 3: UCS C240 M4 SFF	Active SF 3: UCS C240 M4 SFF	Active SF 3: UCS C240 M4 SFF	Active SF 3: UCS C240 M4 SFF
RU-23/24	Active SF 4: UCS C240 M4 SFF	Active SF 4: UCS C240 M4 SFF	Active SF 4: UCS C240 M4 SFF	Active SF 4: UCS C240 M4 SFF
RU-25/26	Active SF 5: UCS C240 M4 SFF	Active SF 5: UCS C240 M4 SFF	Active SF 5: UCS C240 M4 SFF	Active SF 5: UCS C240 M4 SFF
RU-27/28	Active SF 6: UCS C240 M4 SFF	Active SF 6: UCS C240 M4 SFF	Active SF 6: UCS C240 M4 SFF	Active SF 6: UCS C240 M4 SFF
RU-29/30	Empty	Empty	Empty	Empty
RU-31/32	Empty	Empty	Empty	Empty
RU-33/34	Empty	Empty	Empty	Empty
RU-35/36	Ultra UEM 1C,2C,3C,4C	OpenStack Control C: UCS C240 M4 SFF	Empty	Empty
RU-37/38	Ultra M Manager: UCS C240 M4 SFF	Empty	Empty	Empty
RU-39/40	OpenStack Control A: UCS C240 M4 SFF	OpenStack Control B: UCS C240 M4 SFF	Empty	Empty
RU-41/42	Empty	Empty	Empty	Empty
Cables	Controller Rack Cables	Controller Rack Cables	Controller Rack Cables	Empty
Cables	Spine Uplink/Interconnect Cables	Spine Uplink/Interconnect Cables	Empty	Empty
Cables	Leaf TOR To Spine Uplink Cables	Leaf TOR To Spine Uplink Cables	Leaf TOR To Spine Uplink Cables	Leaf TOR To Spine Uplink Cables
Cables	VNF Rack Cables	VNF Rack Cables	VNF Rack Cables	VNF Rack Cables

Cable the Hardware

After the hardware has been installed, install all power and network cabling for the hardware using the information and instructions in the documentation for the specific hardware product. Refer to Related Documentation, on page 44 for links to the hardware product documentation. Ensure that you install your network cables according to your network plan.

Configure the Switches

All of the switches must be configured according to your planned network specifications.

Note Refer to Network Planning, on page 44 for information and consideration for planning your network.

Refer to the user documentation for each of the switches for configuration information and instructions:

- **Catalyst C2960XR-48TD-I:** http://www.cisco.com/c/en/us/support/switches/catalyst-2960xr-48td-i-switch/model.html
- **Catalyst 3850 48T-S:** http://www.cisco.com/c/en/us/support/switches/catalyst-3850-48t-s-switch/model.html
- **Nexus 93180-YC-EX:** http://www.cisco.com/c/en/us/support/switches/nexus-93180yc-fx-switch/model.html
- **Nexus 9236C:** http://www.cisco.com/c/en/us/support/switches/nexus-9236c-switch/model.html

Prepare the UCS C-Series Hardware

UCS-C hardware preparation is performed through the Cisco Integrated Management Controller (CIMC). The tables in the following sections list the non-default parameters that must be configured per server type:

- Prepare the Staging Server/Ultra M Manager Node, on page 49
- Prepare the Controller Nodes, on page 50
- Prepare the Compute Nodes, on page 51
- Prepare the OSD Compute Nodes, on page 52

Refer to the UCS C-series product documentation for more information:

- **UCS C-Series Hardware** — https://www.cisco.com/c/en/us/support/servers-unified-computing/ucs-c240-m4-rack-server/model.html or https://www.cisco.com/c/en/us/support/servers-unified-computing/ucs-c240-m5-rack-server/model.html
- **CIMC Software** — https://www.cisco.com/c/en/us/support/servers-unified-computing/ucs-c-series-integrated-management-controller/tsd-products-support-series-home.html

Important Part of the UCS server preparation is the configuration of virtual drives. If there are virtual drives present which need to be deleted, select the **Virtual Drive Info** tab, select the virtual drive you wish to delete, then click **Delete Virtual Drive**. Refer to the CIMC documentation for more information.

Important The information in this section assumes that the server hardware was properly installed per the information and instructions in Install and Cable the Hardware, on page 44.

Prepare the Staging Server/Ultra M Manager Node

Table 26: Staging Server/Ultra M Manager Node Parameters

Parameters and Settings	Description
CIMC Utility Setup	
Enable IPV4 Dedicated No redundancy IP address Subnet mask Gateway address DNS address	Configures parameters for the dedicated management port.
Admin > User Management	
Username Password	Configures administrative user credentials for accessing the CIMC utility.
Admin > Communication Services	
IPMI over LAN Properties = Enabled	Enables the use of Intelligent Platform Management Interface capabilities over the management port.
Server > BIOS > Configure BIOS > Advanced	
Intel(R) Hyper-Threading Technology = Disabled	Disable hyper-threading on server CPUs to optimize Ultra M system performance.
Storage > Cisco 12G SAS Modular RAID Controller > Physical Drive Info	
Status = *Unconfigured Good*	Ensures that the hardware is ready for use.

Prepare the Controller Nodes

Table 27: Controller Node Parameters

Parameters and Settings	Description
CIMC Utility Setup	
Enable IPV4 Dedicated No redundancy IP address Subnet mask Gateway address DNS address	Configures parameters for the dedicated management port.
Admin > User Management	
Username Password	Configures administrative user credentials for accessing the CIMC utility.
Admin > Communication Services	
IPMI over LAN Properties = Enabled	Enables the use of Intelligent Platform Management Interface capabilities over the management port.
Admin > Communication Services	
IPMI over LAN Properties = Enabled	Enables the use of Intelligent Platform Management Interface capabilities over the management port.
Server > BIOS > Configure BIOS > Advanced	
Intel(R) Hyper-Threading Technology = Disabled	Intel(R) Hyper-Threading Technology = Disabled
Storage > Cisco 12G SAS Modular RAID Controller > Physical Drive Info	
Status = *Unconfigured Good*	Ensures that the hardware is ready for use.
Storage > Cisco 12G SAS Modular RAID Controller > Controller Info	

Parameters and Settings	Description
Virtual Drive Name = OS Read Policy = No Read Ahead RAID Level = RAID 1 Cache Policy: Direct IO Strip Size: 64KB Disk Cache Policy: Unchanged Access Policy: Read Write Size: 1143455 MB Write Policy: Write Through	Creates the virtual drives required for use by the operating system (OS).
Storage > Cisco 12G SAS Modular RAID Controller > Virtual Drive Info	
Initialize Type = Fast Initialize	Initializes this virtual drive. A fast initialization quickly writes zeroes to the first and last 10-MB regions of the new virtual drive and completes the initialization in the background.

Prepare the Compute Nodes

Table 28: Compute Node Parameters

Parameters and Settings	Description
CIMC Utility Setup	
Enable IPV4 Dedicated No redundancy IP address Subnet mask Gateway address DNS address	Configures parameters for the dedicated management port.
Admin > User Management	
Username Password	Configures administrative user credentials for accessing the CIMC utility.
Admin > Communication Services	
IPMI over LAN Properties = Enabled	Enables the use of Intelligent Platform Management Interface capabilities over the management port.

Parameters and Settings	Description
Server > BIOS > Configure BIOS > Advanced	
Intel(R) Hyper-Threading Technology = Disabled	Intel(R) Hyper-Threading Technology = Disabled
Storage > Cisco 12G SAS Modular RAID Controller > Physical Drive Info	
Status = *Unconfigured Good*	Ensures that the hardware is ready for use.
Storage > Cisco 12G SAS Modular RAID Controller > Controller Info	
Virtual Drive Name = BOOTOS Read Policy = No Read Ahead RAID Level = RAID 1 Cache Policy: Direct IO Strip Size: 64KB Disk Cache Policy: Unchanged Access Policy: Read Write Size: 1143455 MB Write Policy: Write Through	Creates the virtual drives required for use by the operating system (OS).
Storage > Cisco 12G SAS Modular RAID Controller > Virtual Drive Info, BOOTOS	
Initialize Type = Fast Initialize	Initializes this virtual drive. A fast initialization quickly writes zeroes to the first and last 10-MB regions of the new virtual drive and completes the initialization in the background.
Set as Boot Drive	Sets the BOOTOS virtual drive as the system boot drive.

Prepare the OSD Compute Nodes

Note OSD Compute Nodes are only used in Hyper-converged Ultra M models as described in UCS C-Series Servers, on page 7.

Table 29: OSD Compute Node Parameters

Parameters and Settings	Description
CIMC Utility Setup	

Parameters and Settings	Description
Enable IPV4 Dedicated No redundancy IP address Subnet mask Gateway address DNS address	Configures parameters for the dedicated management port.
Admin > User Management	
Username Password	Configures administrative user credentials for accessing the CIMC utility.
Admin > Communication Services	
IPMI over LAN Properties = Enabled	Enables the use of Intelligent Platform Management Interface capabilities over the management port.
Server > BIOS > Configure BIOS > Advanced	
Intel(R) Hyper-Threading Technology = Disabled	Intel(R) Hyper-Threading Technology = Disabled
Storage > Cisco 12G SAS Modular RAID Controller > Physical Drive Info	
Status = *Unconfigured Good*	Ensures that the hardware is ready for use.
SLOT-HBA Physical Drive Numbers = 1 2 3 7 8 9 10	Ensure the UCS slot host-bus adapter for the drives are configured accordingly.
Storage > Cisco 12G SAS Modular RAID Controller > Physical Drive Number = 1	

Parameters and Settings	Description
Virtual Drive Name = BOOTOS Read Policy = No Read Ahead RAID Level = RAID 1 Cache Policy: Direct IO Strip Size: 64KB Disk Cache Policy: Unchanged Access Policy: Read Write Size: 285148 MB Write Policy: Write Through	Creates a virtual drive leveraging the storage space available to physical drive number 1. **Note** Ensure that the size of this virtual drive is less than the size of the designated journal and storage drives.
Storage > Cisco 12G SAS Modular RAID Controller > Virtual Drive Info, BOOTOS, Physical Drive Number = 1	
Initialize Type = Fast Initialize	Initializes this virtual drive. A fast initialization quickly writes zeroes to the first and last 10-MB regions of the new virtual drive and completes the initialization in the background.
Set as Boot Drive	Sets the BOOTOS virtual drive as the system boot drive.
Storage > Cisco 12G SAS Modular RAID Controller > Physical Drive Number = 2	
Virtual Drive Name = BOOTOS Read Policy = No Read Ahead RAID Level = RAID 1 Cache Policy: Direct IO Strip Size: 64KB Disk Cache Policy: Unchanged Access Policy: Read Write Size: 285148 MB Write Policy: Write Through	Creates a virtual drive leveraging the storage space available to physical drive number 2. **Note** Ensure that the size of this virtual drive is less than the size of the designated journal and storage drives.
Storage > Cisco 12G SAS Modular RAID Controller > Virtual Drive Info, BOOTOS, Physical Drive Number = 2	
Initialize Type = Fast Initialize	Initializes this virtual drive. A fast initialization quickly writes zeroes to the first and last 10-MB regions of the new virtual drive and completes the initialization in the background.
Set as Boot Drive	Sets the BOOTOS virtual drive as the system boot drive.
Storage > Cisco 12G SAS Modular RAID Controller > Physical Drive Number = 3	

Parameters and Settings	Description
Virtual Drive Name = JOURNAL Read Policy = No Read Ahead RAID Level = RAID 0 Cache Policy: Direct IO Strip Size: 64KB Disk Cache Policy: Unchanged Access Policy: Read Write Size: 456809 MB Write Policy: Write Through	Creates a virtual drive leveraging the storage space available to physical drive number 3.
Storage > Cisco 12G SAS Modular RAID Controller > Virtual Drive Info, JOURNAL, Physical Drive Number = 3	
Initialize Type = Fast Initialize	Initializes this virtual drive. A fast initialization quickly writes zeroes to the first and last 10-MB regions of the new virtual drive and completes the initialization in the background.
Storage > Cisco 12G SAS Modular RAID Controller > Physical Drive Number = 7	
Virtual Drive Name = OSD1 Read Policy = No Read Ahead RAID Level = RAID 0 Cache Policy: Direct IO Strip Size: 64KB Disk Cache Policy: Unchanged Access Policy: Read Write Size: 1143455 MB Write Policy: Write Through	Creates a virtual drive leveraging the storage space available to physical drive number 7.
Storage > Cisco 12G SAS Modular RAID Controller > Virtual Drive Info, OSD1, Physical Drive Number = 7	
Initialize Type = Fast Initialize	Initializes this virtual drive. A fast initialization quickly writes zeroes to the first and last 10-MB regions of the new virtual drive and completes the initialization in the background.
Storage > Cisco 12G SAS Modular RAID Controller > Physical Drive Number = 8	

Parameters and Settings	Description
Virtual Drive Name = OSD2 Read Policy = No Read Ahead RAID Level = RAID 0 Cache Policy: Direct IO Strip Size: 64KB Disk Cache Policy: Unchanged Access Policy: Read Write Size: 1143455 MB Write Policy: Write Through	Creates a virtual drive leveraging the storage space available to physical drive number 8.
Storage > Cisco 12G SAS Modular RAID Controller > Virtual Drive Info, OSD2, Physical Drive Number = 8	
Initialize Type = Fast Initialize	Initializes this virtual drive. A fast initialization quickly writes zeroes to the first and last 10-MB regions of the new virtual drive and completes the initialization in the background.
Storage > Cisco 12G SAS Modular RAID Controller > Physical Drive Number = 9	
Virtual Drive Name = OSD3 Read Policy = No Read Ahead RAID Level = RAID 0 Cache Policy: Direct IO Strip Size: 64KB Disk Cache Policy: Unchanged Access Policy: Read Write Size: 1143455 MB Write Policy: Write Through	Creates a virtual drive leveraging the storage space available to physical drive number 9.
Storage > Cisco 12G SAS Modular RAID Controller > Virtual Drive Info, OSD3, Physical Drive Number = 9	
Initialize Type = Fast Initialize	Initializes this virtual drive. A fast initialization quickly writes zeroes to the first and last 10-MB regions of the new virtual drive and completes the initialization in the background.
Storage > Cisco 12G SAS Modular RAID Controller > Physical Drive Number = 10	

Parameters and Settings	Description
Virtual Drive Name = OSD4 Read Policy = No Read Ahead RAID Level = RAID 0 Cache Policy: Direct IO Strip Size: 64KB Disk Cache Policy: Unchanged Access Policy: Read Write Size: 1143455 MB Write Policy: Write Through	Creates a virtual drive leveraging the storage space available to physical drive number 10.
Storage > Cisco 12G SAS Modular RAID Controller > Virtual Drive Info, OSD4, Physical Drive Number = 10	
Initialize Type = Fast Initialize	Initializes this virtual drive. A fast initialization quickly writes zeroes to the first and last 10-MB regions of the new virtual drive and completes the initialization in the background.

Deploy the Virtual Infrastructure Manager

Within the Ultra M solution, OpenStack Platform Director (OSP-D) functions as the virtual infrastructure manager (VIM).

The method by which the VIM is deployed depends on the architecture of your Ultra M model. Refer to the following section for information related to your deployment scenario.

Deploy the VIM for Hyper-Converged Ultra M Models

Deploying the VIM for Hyper-Converged Ultra M Models is performed using an automated workflow enabled through software modules within Ultra Automation Services (UAS). These services leverage user-provided configuration information to automatically deploy the VIM Orchestrator (Undercloud) and the VIM (Overcloud).

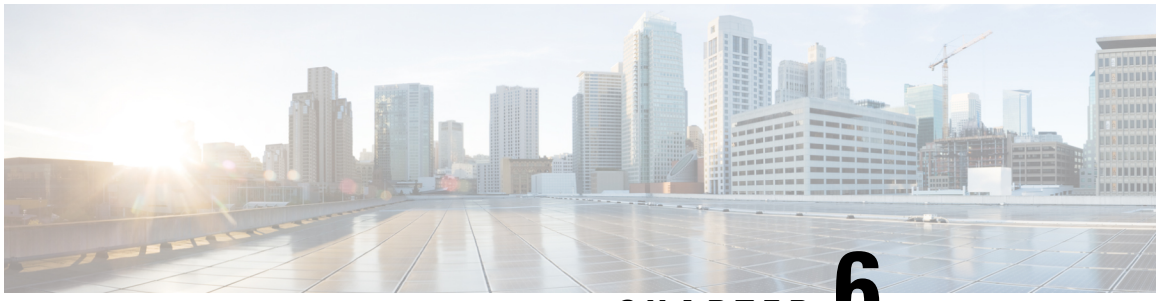
For information on using this automated process, in the *USP Deployment Automation Guide*, refer to the *Virtual Infrastructure Manager Installation Automation* section.

Deploy the USP-Based VNF

After the OpenStack Undercloud (VIM Orchestrator) and Overcloud (VIM) have been successfully deployed on the Ultra M hardware, you must deploy the USP-based VNF.

This process is performed through the Ultra Automation Services (UAS). UAS is an automation framework consisting of a set of software modules used to automate the USP-based VNF deployment and related components such as the VNFM.

For detailed information on the automation workflow, refer to the *Ultra Service Platform Deployment Automation Guide*.

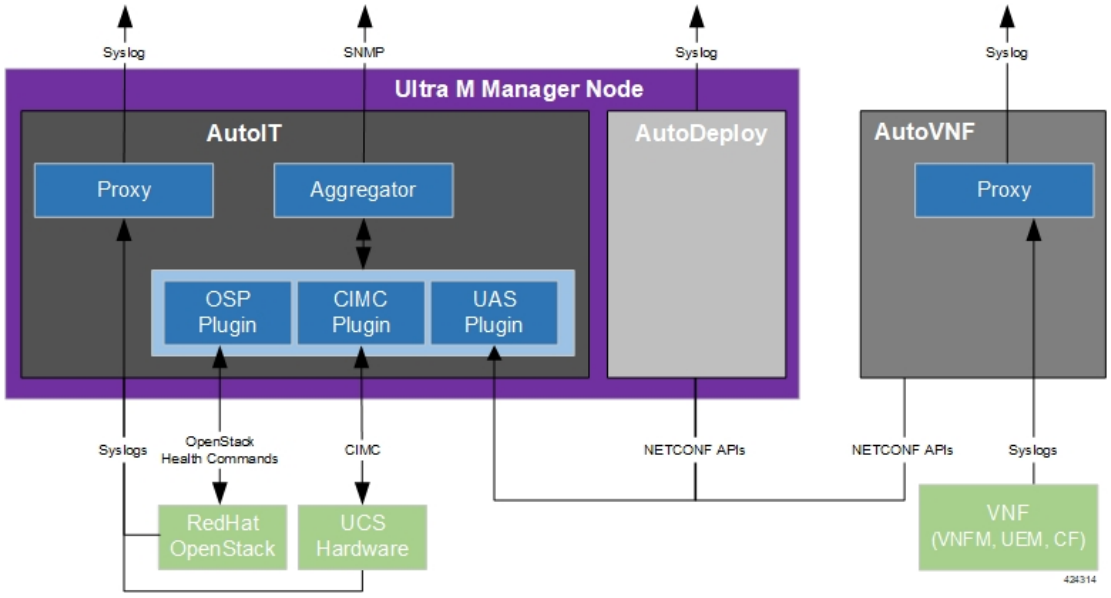


CHAPTER 6

Health Monitoring in the Ultra M Solution

Hyper-Converged Ultra M solution models support a centralized health monitor and management function. This function provides a central aggregation point for events (faults and alarms) and a proxy point for syslogs generated by the different components within the solution as identified in Table 30: Component Event Source Domains, on page 69.

Figure 13: Ultra M Health Monitoring Functions



This functionality is installed with the UAS software modules.

Important The UAS-based health functionality is currently supported only with Ultra M UGP VNF deployments based on OSP 10 or OSP 13 and that leverage the Hyper-Converged architecture. The Ultra M Manager RPM is still distributed separately and is intended only for use in specific deployment scenarios. Contact your local sales or support representative for more information.

Once installed, additional configuration is required based on the desired functionality as described in the following sections:

- Syslog Proxy, on page 60

• Event Aggregation , on page 69
• Configuring Fault Suppression, on page 79

Syslog Proxy

Syslog proxy functionality is supported at the following levels:

- UCS server hardware
- OpenStack services
- UAS software modules
- VNFM, UEM, and CF VNF components

NOTES:

- This functionality is currently supported only with Ultra M UGP VNF deployments based on OSP 10 or OSP 13 and that leverage the Hyper-Converged architecture.
- You must configure a remote collection server to receive and filter log files sent by the Ultra M Manager Node.
  - Take note of the TCP and UDP ports configured on the server for syslogging as the syslog proxy functionality on Ultra M must be configured with the same ports.
  - Ensure that the collection server's IP table rules are configured to accept TCP/UDP connection on the configured port.
- Though you can configure syslogging at any severity level your deployment scenario requires, it is recommended that you only configure syslog levels with severity levels 0 (emergency) through 4 (warning). If the severity level is not set, then by default, the severity level 6 is used.

Important If you wish to enable syslogging for the components that comprise the Ultra M solution but do not wish to use the syslog proxy functionality (e.g. send syslogs directly to an external collection server), refer to Configuring Syslogging to an External Collection Server, on page 65.

Configuring Syslog Proxy for UCS Server Hardware

AutoIT can be configured to serve as a proxy for UCS server hardware syslogs.

Important AutoIT must be configured with information for the syslog collection server at the time it is deployed. Refer to the *Ultra Services Platform Deployment Automation Guide* for more information.

The UCS server list is based on the configuration specified in the VIM Orchestrator and VIM NSD configuration file. As such, syslog proxy functionality for the hardware must be performed after the VIM has been deployed.

Syslog proxy functionality is configured through a fault management descriptor (FMD) configuration file that is comprised of the required NETCONF parameters. Refer to Sample FMD Configuration File, on page 141 for a sample configuration file.

Important Though the FMD configuration can be included in the network service descriptor (NSD) for your VNF, it is recommended that the configuration for this functionality be maintained in a separate, FMD-specific NSD configuration file.

To configure syslog proxy functionality for UCS server hardware:

1. Log on to the primary AutoIT VM as the root user.
2. Prepare the FMD configuration file for your deployment scenario. Your file should include the following parameters:

   ```
   domain hardware
     syslog uas-proxy
     syslog severity <severity_level>
   ```

   Note that the **severity** parameter is optional. The default severity level is 6.

   For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

   Refer to Sample FMD Configuration File, on page 141 for a sample configuration file.
3. Login to the ConfD CLI as the *admin* user.

   ```
   confd_cli -u admin -C
   ```
4. Enter the *admin* user password when prompted.
5. Enter the ConfD configuration mode.

   ```
   config
   ```
6. Load the FMD configuration file.

   ```
   load merge <your_fmd_file_name>.cfg
   commit
   end
   ```
7. Activate the FMD configuration.

   ```
   activate nsd-id <nsd_name> fmd <fmd_name>
   ```

Important The output of this command is a transaction-id which can be used to monitor the deployment progress. If need be, the FMD configuration can be deactivated using the **deactivate** variant of this command.

8. Monitor the progress of the FMD creation by viewing transaction logs:

   ```
   show log <transaction_id> | display xml
   ```

   transaction_id is the ID displayed as a result of the **activate** command executed in step 7, on page 61.

Configuring Syslog Proxy for OpenStack Services

AutoIT can be configured to serve as a proxy for OpenStack service syslogs.

Important AutoIT must be configured with information for the syslog collection server at the time it is deployed. Refer to the *Ultra Services Platform Deployment Automation Guide* for more information.

The list of servers on which OpenStack is running is based on the configuration specified in the VIM Orchestrator and VIM NSD configuration file. As such, syslog proxy functionality for the hardware must be performed after the VIM has been deployed.

If syslogging is enabled, syslogs for the following OpenStack services are proxied:

- Nova
- Cinder
- Keystone
- Glance
- Ceph monitor (Controller nodes only)
- Ceph OSD (OSD Compute nodes only)

Syslog proxy functionality is configured through a fault management descriptor (FMD) configuration file that is comprised of the required NETCONF parameters. Refer to Sample FMD Configuration File, on page 141 for a sample configuration file.

Important Though the FMD configuration can be included in the network service descriptor (NSD) for your VNF, it is recommended that the configuration for this functionality be maintained in a separate, FMD-specific NSD configuration file.

To configure syslog proxy functionality for UCS server hardware:

1. Log on to the primary AutoIT VM as the root user.
2. Prepare the FMD configuration file for your deployment scenario. Your file should include the following parameters:

   ```
   domain vim
     syslog uas-proxy
     syslog severity <severity_level>
   ```

   Note that the **severity** parameter is optional. The default severity level is 6.

   For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

   Refer to Sample FMD Configuration File, on page 141 for a sample configuration file.
3. Login to the ConfD CLI as the *admin* user.

   ```
   confd_cli -u admin -C
   ```

4. Enter the *admin* user password when prompted.
5. Enter the ConfD configuration mode.

   config
6. Load the FMD configuration file.

   load merge *<your_fmd_file_name>*.**cfg**

   commit

   end
7. Activate the FMD configuration.

   activate nsd-id *<nsd_name>* **fmd** *<fmd_name>*

Important The output of this command is a transaction-id which can be used to monitor the deployment progress. If need be, the FMD configuration can be deactivated using the **deactivate** variant of this command.

8. Monitor the progress of the FMD creation by viewing transaction logs:

   show log *<transaction_id>* **| display xml**

   transaction_id is the ID displayed as a result of the **activate** command executed in step 7, on page 63.

Configuring Syslogging for UAS Software Modules

Each UAS software module can be configured to send logs and syslogs to one or more external collection servers.

AutoDeploy and AutoIT

Logs and syslogs are sent directly to one or more external syslog collection servers configured when these modules are first installed. The configured collection servers are also the receivers for UCS server hardware and OpenStack services for which AutoIT is a proxy.

The following logs are sent:

- **AutoDeploy:**
  - /var/log/upstart/autodeploy.log
  - /var/log/syslog
- **AutoIT:**
  - /var/log/upstart/autoit.log
  - /var/log/syslog

In order to support syslogging functionality, additional operators were added to the *boot_uas.py* script used to install these modules:

- **--syslog-ip**<*ext_syslog_server_address*>

- **--port**<*syslog_port_number*>
- **--severity**<*syslog_severity_to_send*>

Refer to the *Ultra Services Platform Deployment Automation Guide* for more information on deploying AutoIT and AutoDeploy.

AutoVNF

AutoVNF serves as the syslog proxy for the VNFM, UEM, and CF VNF components (VNFCs). It also sends its own logs to the same external syslog collection server:

- /var/log/upstart/autovnf.log
- /var/log/syslog

Syslogging for the AutoVNF module is configured through the AutoVNF VNFC configuration within the VNF Rack and VNF NSD configuration file. Refer to the *Ultra Services Platform Deployment Automation Guide* for more information.

AutoVNF should always configure the external syslog server. For AutoVNF, the information and instructions provided in those sources also remain identical but with the exception of the parameters used in the corresponding VNFC section of the VNF Rack and VNF NSD configuration file.

```
syslog server <ip_address>
syslog port <tcp_udp_port>
syslog severity <severity_level>
```

Note that the **port** and **severity** parameters are optional. The default values of **port** and **severity** parameters are 514 and 6 respectively.

For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

Configuring Syslog Proxy for the VNFM, UEM, and CF VNFCs

AutoVNF can be configured as the syslog proxy for the following VNFM, UEM, and CF VNF component (VNFC) logs:

- **VNFM (ESC):** /var/log/messages

Important *escmanager* and *mona* logs are not configured as part of syslog automation. ESC can be manually configured to send these logs to the syslog proxy or to an external syslog collection server. Refer to Manual ESC escmanager and mona Log Configuration, on page 67 for more information.

- **UEM:**
  - /var/log/em/vnfm-proxy/vnfm-proxy
  - /var/log/em/ncs/ncs-java-vm
  - /var/log/em/zookeeper/zookeeper

- /var/log/syslog

- **CF:** All syslogs configured within the StarOS-based VNF.

Syslogging for the VNFM, UEM, and CF is configured through their respective VNFC configurations within the VNF Rack and VNF NSD configuration file. Refer to the *Ultra Services Platform Deployment Automation Guide* for more information.

The following parameters should be configured for each VNFC:

```
syslog uas-proxy
syslog severity <severity_level>
```

Note that the **severity** parameter is optional. The default severity level is 6.

For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

Configuring Syslogging to an External Collection Server

Syslogging for the Ultra M solution components can be enabled without leveraging the syslog proxy functionality. In this scenario, syslogs are sent directly from each component to an external collection server.

Important Regardless of the domain level at which you're configuring syslogging functionality for, you must ensure that the external collection server to which your sending syslogs is reachable over the network by the component sending the syslog.

UCS Server Hardware

The instructions for configuring UCS servers to send syslogs to an external collection server are identical to those described in Configuring Syslog Proxy for UCS Server Hardware, on page 60 with the exception of the parameters used in the FMD configuration file.

To configure external collection servers for UCS server hardware, use the following parameters:

```
domain hardware
  syslog server <ip_address>
  syslog port <tcp_udp_port>
  syslog severity <severity_level>
```

Note that the **port** and **severity** parameters are optional. The default values of **port** and **severity** parameters are 514 and 6 respectively.

For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

Important Though multiple external collection servers can be configured, the UCS server hardware support a maximum of two servers. If more than two servers are configured in the FMD, only the first two are configured on the UCS servers. Additionally, only one severity level can be configured on the UCS servers. It is used for both configured collection servers.

OpenStack Services

The instructions for configuring OpenStack services to send syslogs to an external collection server are identical to those described in Configuring Syslog Proxy for OpenStack Services, on page 62 with the exception of the parameters used in the FMD configuration file.

To configure external collection servers for OpenStack services, use the following parameters:

```
domain vim
  syslog server <ip_address>
  syslog port <tcp_udp_port>
  syslog severity <severity_level>
```

Note that the **port** and **severity** parameters are optional. The default values of **port** and **severity** parameters are 514 and 6 respectively.

For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

UAS Software Modules

The information and instructions provided in Configuring Syslogging for UAS Software Modules, on page 63 and in the *Ultra Services Platform Deployment Automation Guide* that pertain to AutoDeploy and AutoIT configure them to communicate with external collection servers.

To configure external collection servers for the AutoVNF, use the following parameters:

```
  syslog server <ip_address>
  syslog port <tcp_udp_port>
  syslog severity <severity_level>
```

Note that the **port** and **severity** parameters are optional. The default values of **port** and **severity** parameters are 514 and 6 respectively.

For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

VNFM, UEM, and CF VNF Components

The instructions for configuring the VNFM, UEM, and CFs to send syslogs to an external collection server are identical to those described in Configuring Syslog Proxy for the VNFM, UEM, and CF VNFCs, on page 64 and in the *Ultra Services Platform Deployment Automation Guide* with the exception of the parameters used in the corresponding VNFC section of the VNF Rack and VNF NSD configuration file.

To configure external collection servers for the VNFCs, use the following parameters for each VNFC:

```
  syslog server <ip_address>
  syslog port <tcp_udp_port>
  syslog severity <severity_level>
```

Important

- Though multiple external collection servers can be configured, the ESC-based VNFM supports the configuration of only a single server. If multiple servers are configured in the VNFC, only the first is configured on the ESC-based VNFM. Additionally, all severity levels are enabled for the ESC-based VNFM regardless of the severity specified in the configuration.
- The ESC *escmanager* and *mona* logs are not configured as part of syslog automation. ESC can be manually configured to send these logs to the syslog proxy or to an external syslog collection server. Refer to Manual ESC escmanager and mona Log Configuration, on page 67 for more information.
- For the CF component within the VNF, neither the syslog port or the syslog severity need to be configured. The default syslog port of 514 and the default severity of 7, debug, is used.

  For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

Manual ESC *escmanager* and *mona* Log Configuration

ESC's *escmanager* and *mona* logs are not configured as part of syslog automation. However, ESC can be manually configured to send these logs to either the syslog proxy server (i.e. AutoVNF) or to an external collection server.

To manually configure ESC to send these logs:

1. Log on to the active ESC VNFM VM as the user *admin*.
2. Navigate to the */etc/rsyslog.d* directory.

   `cd /etc/rsyslog.d`
3. Create a configuration file for the *escmanager* log called *00-escmanager.conf*. The file should have the following configuration information which includes the IP address of the syslog server (either the syslog proxy server or the external collection server):

   ```
   $ModLoad imfile
   $InputFileName /var/log/esc/escmanager.log
   $InputFileTag escmanager:
   $InputFileStateFile stat-escmanager
   $InputRunFileMonitor

   $template escmanager_log, "%syslogtag:::% %msg%"

   if $programname == 'escmanager' then @@<syslog-server-ip>:<port-number>;escmanager_log

   if $programname == 'escmanager' then stop
   ```

   <*syslog-server-ip*> is the one of the following:

   - AutoVNF HA VIP address if you want the logs sent to the syslog proxy server, OR
   - IP address of the external syslog collection server.

   <*port-number*> is the TCP/UDP port used for syslog. For the syslog proxy functionality, the default port of 514 is used.

Important The server IP address and port number must be identical to those configured at the VNFC-level for the VNFM.

4. Create a configuration file for the *mona* log called *02-mona.conf*. The file should have the following configuration information which includes the IP address of the syslog server (either the syslog proxy server or the external collection server):

```
$ModLoad imfile
$InputFileName /var/log/esc/mona/mona.log
$InputFileTag mona:
$InputFileStateFile stat-mona
$InputRunFileMonitor

$template mona_log, "%syslogtag:::% %msg%"

if $programname == 'mona' then @@<syslog-server-ip>:<port-number>;mona_log

if $programname == 'mona' then stop
```

<*syslog-server-ip*> is the one of the following:

- AutoVNF HA VIP address if you want the logs sent to the syslog proxy server, OR
- IP address of the external syslog collection server.

<*port-number*> is the TCP/UDP port used for syslog. For the syslog proxy functionality, the default port of 514 is used.

Important The server IP address and port number must be identical to those configured at the VNFC-level for the VNFM.

5. Change the file permissions for the *escmanager.log* file.

```
ls -al /var/log/esc/escmanager.log
-rw-r--r--. 1 esc-user esc-user 12671993 Sep 12 23:32 /var/log/esc/escmanager.log
sudo chmod 666 /var/log/esc/escmanager.log
ls -al /var/log/esc/escmanager.log
-rw-rw-rw-. 1 esc-user esc-user 12671993 Sep 12 23:32 /var/log/esc/escmanager.log
```

6. Change the file permissions for the *mona.log* file.

```
ls -al /var/log/esc/mona/mona.log
-rw-r--r--. 1 esc-user esc-user 3937424 Sep 13 01:10 /var/log/esc/mona/mona.log
sudo chmod 666 /var/log/esc/mona/mona.log
ls -al /var/log/esc/mona/mona.log
-rw-rw-rw-. 1 esc-user esc-user 3940388 Sep 13 01:11 /var/log/esc/mona/mona.log
```

7. Restart the syslog service.

```
sudo service rsyslog restart
```

8. Repeat steps 1, on page 67 through 7, on page 68 on the standby ESC VNFM VM.

Event Aggregation

The AutoIT module within the Ultra M Manager Node can be configured to aggregate events received from different Ultra M components as identified in Table 30: Component Event Source Domains, on page 69.

Important This functionality is currently supported only with Ultra M UGP VNF deployments based on OSP 10 or OSP 13 and that leverage the Hyper-Converged architecture. In pre-6.2 releases, this functionality was made available through the Ultra M Manager utility. The Ultra M Manager RPM is still distributed separately and is intended only for use in specific deployment scenarios. Contact your local sales or support representative for more information.

Table 30: Component Event Source Domains

Solution Component Domain	Event Source Type	Details
hardware (UCS server hardware)	CIMC	Reports on events collected from UCS C-series hardware via CIMC-based subscription. These events are monitored in real-time.
vim (VIM (Overcloud))	OpenStack service health	Reports on OpenStack service fault events pertaining to: • Failures (stopped, restarted) • High availability • Ceph / storage • Neutron / compute host and network agent • Nova scheduler (VIM instances) Refer to Table 31: Monitored OpenStack Services, on page 71 for a complete list of services.
uas (UAS AutoVNF)	UAS cluster	Reports on UAS service fault events pertaining to: • Service failure (stopped, restarted) • High availability
vnfm (ESC-based VNFM)	ESC (VNFM) event notifications	Reports on ESC-based VNFM service fault events pertaining to: • Service failure (stopped, restarted) **Important** Events on a per-VNFM VM level.

Solution Component Domain	Event Source Type	Details
vnf-em (UEM)	USP management component events	Reports on UEM service fault events pertaining to: • Service failure (stopped, restarted) • High availability • Internal application errors (e.g. SCM, LCM, etc.)
vnf (VNF VM Status)	ESC (VNFM) event notifications	Reports on VNF VM deployment state events generated by ESC (the VNFM). The following events are supported: • VM_DEPLOYED • VM_ALIVE • VM_UNDEPLOYED • VM_REBOOTED • VM_RECOVERY_REBOOT • VM_RECOVERY_UNDEPLOYED • VM_RECOVERY_DEPLOYED • VM_RECOVERY_COMPLETE • VM_STOPPED **Important** This feature was introduced in 6.0. It was not fully qualified and made available only for testing purposes. In 6.0, AutoVNF monitors for event notifications from ESC in real time. Though AutoVNF updates the VNFR for the VNF and VNFC the event pertains to upon receipt of an event, it does not generate a corresponding SNMP trap. It is fully qualified and fully functional as of the 6.2 release.

Table 31: Monitored OpenStack Services

Node Type	OpenStack Module	OpenStack Services
Controller	cinder	• openstack-cinder-api.service, • openstack-cinder-scheduler.service
	glance	• openstack-glance-api.service, • openstack-glance-registry.service
	heat-engine	openstack-heat-engine.service
	heat-api	• openstack-heat-api-cfn.service, • openstack-heat-api-cloudwatch.service, • openstack-heat-api.service
	heat	• openstack-heat-api-cfn.service, • openstack-heat-api-cloudwatch.service, • openstack-heat-api.service
	nova	• openstack-nova-api.service, • openstack-nova-conductor.service, • openstack-nova-consoleauth.service, • openstack-nova-novncproxy.service, • openstack-nova-scheduler.service
	swift-object	• openstack-swift-object-auditor.service, • openstack-swift-object-replicator.service, • openstack-swift-object-updater.service, • openstack-swift-object.service
	swift-account	• openstack-swift-account-auditor.service, • openstack-swift-account-reaper.service, • openstack-swift-account-replicator.service, • openstack-swift-account.service
	swift-container	

Node Type	OpenStack Module	OpenStack Services
		• openstack-swift-container-auditorservice, • openstack-swift-container-replicatorservice, • openstack-swift-container-updaterservice, • openstack-swift-container.service
	swift-proxy	openstack-swift-proxy.service
	swift	All above swift services
	ntpd	ntpd.service
	mongod	mongod.service
	memcached	memcached
	neutron-dhcp-agent	neutron-dhcp-agent.service
	neutron-l3-agent	neutron-l3-agent.service
	neutron-metadata-agent	neutron-metadata-agent.service
	neutron-openvswitch-agent	neutron-openvswitch-agent.service
	neutron-server	neutron-server.service
	httpd	httpd.service
OSD Compute	ceph-mon.target	ceph-mon.target
	ceph-radosgw.target	ceph-radosgw.target
	ceph.target	ceph.target
	openvswitch.service	openvswitch.service
	neutron-sriov-nic-agent	neutron-sriov-nic-agent.service
	neutron-openvswitch-agent	neutron-openvswitch-agent.service
	ntpd	ntpd.service
	nova-compute	openstack-nova-compute.service
	libvirtd	libvirtd.service

Node Type	OpenStack Module	OpenStack Services
Compute	ceph-mon.target	ceph-mon.target
	ceph-radosgw.target	ceph-radosgw.target
	ceph.target	ceph.target
	openvswitch.service	openvswitch.service
	neutron-sriov-nic-agent	neutron-sriov-nic-agent.service
	neutron-openvswitch-agent	neutron-openvswitch-agent.service
	ntpd	ntpd.service
	nova-compute	openstack-nova-compute.service
	libvirtd	libvirtd.service

Faults can be enabled or disabled at various levels as described in Configuring Fault Suppression, on page 79.

Events received from the solution components, regardless of the source type, are mapped against the Ultra M SNMP MIB (CISCO-ULTRAM-MIB.my, refer to Ultra M MIB, on page 91). The event data is parsed and categorized against the following conventions:

- **Fault code:** Identifies the area in which the fault occurred for the given component. Refer to the "CFaultCode" convention within the Ultra M MIB for more information.
- **Severity:** The severity level associated with the fault. Refer to the "CFaultSeverity" convention within the Ultra M MIB for more information. Since the Ultra M Manager Node aggregates events from different components within the solution, the severities supported within the Ultra M Manager Node MIB map to those for the specific components. Refer to Ultra M Component Event Severity and Fault Code Mappings, on page 99 for details.
- **Domain:** The component in which the fault occurred (e.g. UCS hardware, VIM, UEM, etc.). Refer to the "CFaultDomain" convention within the Ultra M MIB for more information.

UAS and OpenStack events are monitored at the configured polling interval as described in Table 32: SNMP Fault Entry Table Element Descriptions, on page 75. At the polling interval, the Ultra M Manager Node:

1. Collects data from UAS and OpenStack.
2. Generates/updates .log and .report files and an SNMP-based fault table with this information. It also includes related data about the fault such as the specific source, creation time, and description.
3. Processes any events that occurred:
   a. If an error or fault event is identified, then a .error file is created and an SNMP trap is sent.
   b. If the event received is a clear condition, then an informational SNMP trap is sent to "clear" an active fault.
   c. If no event occurred, then no further action is taken beyond Step 2.

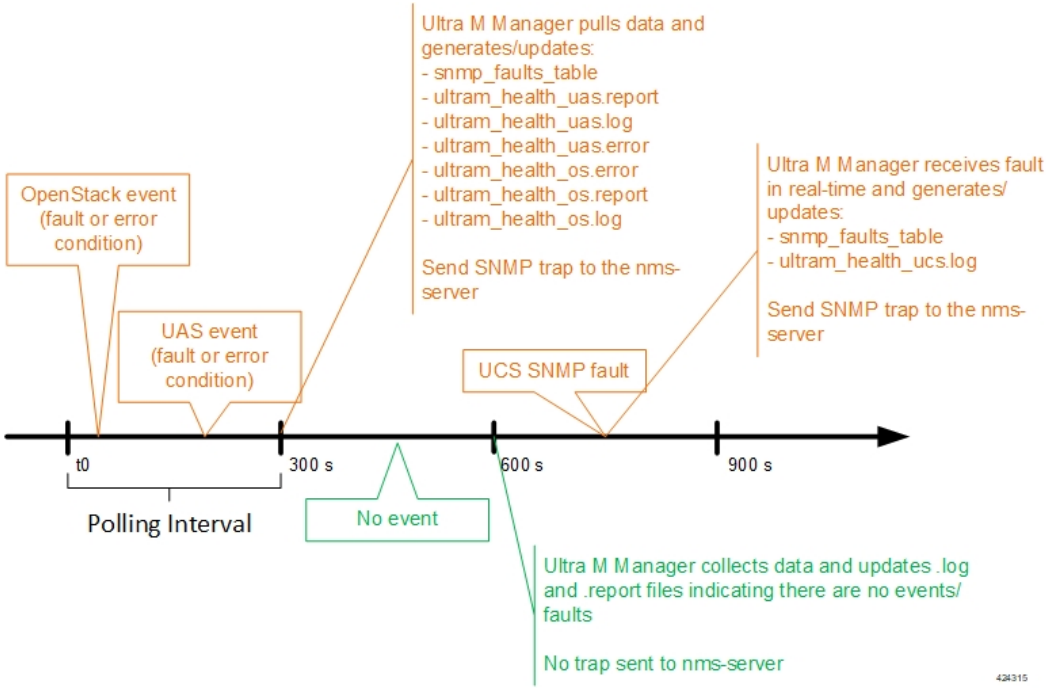
UCS and ESC VM events are monitored and acted upon in real-time. When events occur, the Ultra M Manager generates a .log file and the SNMP fault table. In the case of VM events reported by ESC, upon receipt of an event, AutoVNF updates the VNFR for the VNF and VNFC the event pertains to. In parallel, it passes the event information to the Ultra M Manager functionality within AutoIT. The Ultra M Manager then generates corresponding SNMP traps for each event.

Active faults are reported “only” once and not on every polling interval. As a result, there is only one trap as long as this fault is active. Once the fault is “cleared”, an informational trap is sent.

Important UCS events are considered to be the “same” if a previously received fault has the same distinguished name (DN), severity, and lastTransition time. UCS events are considered as “new” only if any of these elements change.

These processes are illustrated in Figure 14: Ultra M Manager Node Event Aggregation Operation, on page 74. Refer to About Ultra M Manager Log Files, on page 113 for more information.

Figure 14: Ultra M Manager Node Event Aggregation Operation

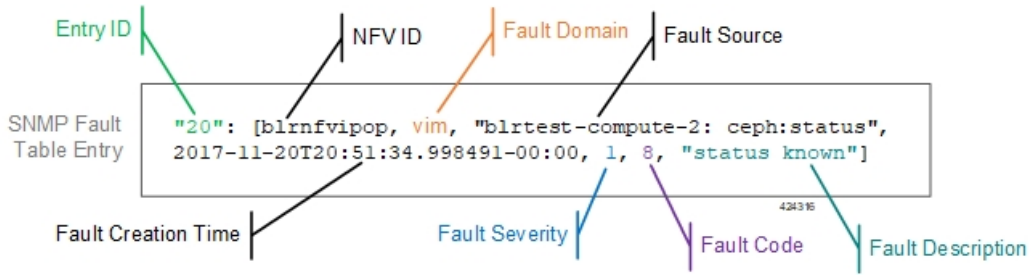


An example of the snmp_faults_table file is shown below and the entry syntax is described in Figure 15: SNMP Fault Table Entry Description, on page 75:

```
"0": [3 "neutonoc-osd-compute-0: neutron-sriov-nic-agent.service" 1 8 "status known"] "1":
 [3 "neutonoc-osd-compute-0: ntpd" 1 8 "Service is not active state: inactive"] "2": [3
"neutonoc-osd-compute-1: neutron-sriov-nic-agent.service" 1 8 "status known"] "3": [3
"neutonoc-osd-compute-1: ntpd" 1 8 "Service is not active state: inactive"] "4": [3
"neutonoc-osd-compute-2: neutron-sriov-nic-agent.service" 1 8 "status known"] "5": [3
"neutonoc-osd-compute-2: ntpd" 1 8 "Service is not active state: inactive"]
```

Refer to About Ultra M Manager Log Files, on page 113 for more information.

Figure 15: SNMP Fault Table Entry Description



Each element in the SNMP Fault Table Entry corresponds to an object defined in the Ultra M SNMP MIB as described in Table 32: SNMP Fault Entry Table Element Descriptions, on page 75. (Refer also to Ultra M MIB, on page 91.)

Table 32: SNMP Fault Entry Table Element Descriptions

SNMP Fault Table Entry Element	MIB Object	Additional Details
Site ID	cultramSiteId	Identify fault at site level
Entry ID	cultramFaultIndex	A unique identifier for the entry
NFV ID	cultramNFVIdenity	Ultra M PoD on which this fault is occurring
Fault Domain	cultramFaultDomain	The component area in which the fault occurred. Refer to Table 30: Component Event Source Domains, on page 69 for information on domains supported in this release.
Fault Source	cultramFaultSource	Information identifying the specific component within the Fault Domain that generated the event. The format of the information is different based on the Fault Domain. Refer to Table 33: cultramFaultSource Format Values, on page 77 for details.
Fault Creation Time	cultramFaultCreationTime	The date and time when the fault was occurred.

SNMP Fault Table Entry Element	MIB Object	Additional Details
Fault Severity	cultramFaultSeverity	The severity associated with the fault as one of the following: • **emergency(1) :** System level FAULT impacting multiple VNFs/Services • **critical(2) :** Critical Fault specific to VNF/Service • **major(3) :** component level failure within VNF/service. • **alert(4) :** warning condition for a service/VNF, may eventually impact service. • **informational(5) :** informational only, does not impact service Refer to Ultra M Component Event Severity and Fault Code Mappings, on page 99 for details on how these severities map to events generated by the various Ultra M components.
Fault Code	cultramFaultCode	A unique ID representing the type of fault as. The following codes are supported: • **other(1) :** Other events • **networkConnectivity(2) :** Network Connectivity Failure Events • **resourceUsage(3) :** Resource Usage Exhausted Event • **resourceThreshold(4) :** Resource Threshold crossing alarms • **hardwareFailure(5) :** Hardware Failure Events • **securityViolation(6) :** Security Alerts • **configuration(7) :** Config Error Events • **serviceFailure(8) :** Process/Service failures Refer to Ultra M Component Event Severity and Fault Code Mappings, on page 99 for details on how these fault codes map to events generated by the various Ultra M components.
Fault Description	cultramFaultDescription	A message containing details about the fault.

Table 33: cultramFaultSource Format Values

FaultDomain	Format Value of cultramFaultSource
Hardware (UCS Servers)	**Node:** *<UCS-SERVER-IP-ADDRESS>*, **affectedDN:** *<FAULT-OBJECT-DISTINGUSIHED-NAME>* Where: <*UCS-SERVER-IP-ADDRESS*> *:* The management IP address of the UCS server that generated the fault. <*FAULT-OBJECT-DISTINGUSIHED-NAME*> *:* The distinguished name of the affected UCS object.
UAS	**Node:** *<UAS-MANAGEMENT-IP>* Where: <*UAS-MANAGEMENT-IP*> *:* The management IP address for the UAS instance.
VIM (OpenStack)	*<OS-HOSTNAME>*: *<SERVICE-NAME>* Where: <*OS-HOSTNAME*> : The OpenStack node hostname that generated the fault. <*SERVICE-NAME*> : Then name of the OpenStack service that generated the fault.

SNMP Version Support

The following commands are supported for both SNMP Version 2 and Version 3:

- GET
- Walk
- GETNEXT
- GETBULK

The following security algorithms are supported for SNMP Version 3:

Table 34: Supported SNMP Version 3 Security Algorithms

Protocol	Algorithms
Authentication	• usmNoAuthProtocol • usmHMACMD5AuthProtocol • usmHMACSHAAuthProtocol

Protocol	Algorithms
Privacy	• usmNoPrivProtocol • usmDESPrivProtocol • usm3DESEDEPrivProtocol • usmAesCfb128Protocol • usmAesCfb192Protocol • usmAesCfb256Protocol

For SNMP Version 3, the SNMP Engine ID is generated in accordance with RFC 3411:

```
(80000000 OR HEX value of enterprise ID) + 04 + (HEX value of Administratively Assigned
String)
```

Important The name of the network service descriptor (NSD) in which fault management functionality is configured is used as the 'Administratively Assigned String'. For deployment scenarios that require the Ultra M Manager RPM for fault maangement functionality, the name of the UCS cluster is used.

SNMP configuration is based on parameters configured in the fault management descriptor (FMD) along with other parameters pertaining to Ultra M health monitoring. Refer to Configuring Event Aggregation, on page 78 for more information on configuring and activating the FMD. Refer to the *Cisco Ultra Services Platform NETCONF API Guide* for more information on the specific parameters that comprise the FMD.

Configuring Event Aggregation

Event aggregation functionality is configured through NETCONF API-based remote procedure calls invoked via AutoIT. In either scenario, the parameters related to this functionality are defined by/within the fault management descriptor (FMD). When the VNF is deployed, the FMD configuration is merged into the existing NSD configuration. (Refer to the *Cisco Ultra Services Platform NETCONF API Guide* for details on the parameters supported within the FMD.)

Though the FMD configuration can be included in the NSD configuration file, it is recommended that the configuration for this functionality be maintained in a separate, FMD-specific NSD configuration file.

Important The instructions in this section assume that the Ultra M solution has been completely deployed prior to proceeding. This includes the VIM Orchestrator, the VIM, the UAS components, and the VNF.

To enable this functionality on the Ultra M solution:

1. Log on to the primary AutoIT VM as the root user.
2. Prepare the FMD configuration file for your deployment scenario. Your file should include the following parameters:
   - SNMP user configuration
   - Fault management descriptor (FMD) configuration

- Domain configuration (e.g. hardware, vim, uas, etc.)
- SNMP version and receiver configuration

Refer to Sample FMD Configuration File, on page 141 for a sample configuration file. Refer to the *Cisco Ultra Services Platform NETCONF API Guide* for a complete list of supported parameters.

3. Login to the ConfD CLI as the *admin* user.

   confd_cli -u admin -C

4. Enter the *admin* user password when prompted.
5. Enter the ConfD configuration mode.

   config

6. Load the FMD configuration file.

   load merge *<your_fmd_file_name>*.**cfg**

   commit

   end

7. Activate the FMD configuration.

   activate nsd-id *<nsd_name>* **fmd** *<fmd_name>*

Important The output of this command is a transaction-id which can be used to monitor the deployment progress. If need be, the FMD configuration can be deactivated using the **deactivate** variant of this command.

8. Monitor the progress of the FMD creation by viewing transaction logs:

   show log *<transaction_id>* **| display xml**

   transaction_id is the ID displayed as a result of the **activate** command executed in step 7, on page 79.

Configuring Fault Suppression

AutoIT can be configured to monitor the fault events for a specified domain. The fault suppression functionality for VNFC(s) must be performed after the VIM has been deployed.

Important AutoIT must be configured with information for the event (fault and alarm) monitoring at the time it is deployed. Refer to the *Ultra Services Platform Deployment Automation Guide* for more information.

Fault suppression functionality is configured through a fault management descriptor (FMD) configuration file that is comprised of the required NETCONF parameters.

Depending on the configuration of this functionality, the faults can be automatically suppressed at the following levels:

- UCS server:

- **UCS cluster**: All events for all UCS nodes are suppressed.
- **UCS fault object distinguished names (DNs)**: All events for one or more specified UCS object DNs within are suppressed.
- **UCS faults**: One or more specified UCS faults are suppressed.

Important Fault suppression can be simultaneously configured at both the UCS object DN and fault levels.

- UAS and VNF components:
  - **UAS component cluster**: All events for all UAS components are suppressed.
  - **UAS component events**: One or more specified UAS component events are suppressed.

When faults are suppressed, event monitoring occurs as usual and the log report file shows the faults. However, suppressed faults are not reported over SNMP. Within the log file, suppress faults are preceded by the word "Skipping".

Suppressing UCS Faults

AutoIT can be configured to suppress UCS hardware faults based on fault ID or affected fault object distinguished names (DNs).

UCS incorporates the concept of DN where each entity is been assigned unique ID or namespace. Suppressing events for a given UCS fault object distinguished name (DN) stops the reporting of all events related to the DN. Suppression can be enabled for one or more DNs.

Important Fault suppression can be simultaneously configured at both the UCS object DN and fault levels.

Fault suppression functionality is configured through a fault management descriptor (FMD) configuration file that is comprised of the required NETCONF parameters. Refer to Sample FMD Configuration File, on page 141 for a sample configuration file.

Important Though the FMD configuration can be included in the network service descriptor (NSD) for your VNF, it is recommended that the configuration for this functionality be maintained in a separate, FMD-specific NSD configuration file.

Refer to the latest Cisco UCS Faults and Error Messages Reference Guide for more information https://www.cisco.com/c/en/us/support/servers-unified-computing/unified-computing-system/products-system-message-guides-list.html .

To configure fault suppression functionality for UCS server hardware:

1. Log on to the primary AutoIT VM as the root user.

2. Prepare the FMD configuration file for your deployment scenario. Your file should include the following parameters:

   ```
   domain hardware
     monitoring suppress-hw-affected-dn [ <dn_1>…<dn_n> ]
     monitoring suppress-hw-fault-id [ <fault_id_1>…<fault_id_n> ]
   ```

   The operators **suppress-hw-affected-dn** and **suppress-hw-fault-id** are optional. If these are not configured, the faults can be raised.

   Refer to Sample FMD Configuration File, on page 141 for a sample configuration file.

   For information related to UCS faults, see https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/ts/faults/reference/3-0/UCSFaultsErrorsRef_3-0/UCS_SEMs_3-0.html.

3. Login to the ConfD CLI as the *admin* user.

   ```
   confd_cli -u admin -C
   ```

4. Enter the *admin* user password when prompted.

5. Enter the ConfD configuration mode.

   ```
   config
   ```

6. Load the FMD configuration file.

   ```
   load merge <your_fmd_file_name>.cfg
   commit
   end
   ```

7. Activate the FMD configuration.

   ```
   activate nsd-id <nsd_name> fmd <fmd_name>
   ```

Important The output of this command is a transaction-id which can be used to monitor the deployment progress. If need be, the FMD configuration can be deactivated using the **deactivate** variant of this command.

8. Monitor the progress of the FMD creation by viewing transaction logs:

   ```
   show log <transaction_id> | display xml
   ```

   transaction_id is the ID displayed as a result of the **activate** command executed in step 7, on page 63.

Suppressing UAS Faults

AutoIT can be configured to suppress UAS faults based on UAS components (AutoVNF, UEM, and the VNFM (ESC)) or type of failure within the component. Fault suppression functionality is configured through a fault management descriptor (FMD) configuration file that is comprised of the required NETCONF parameters. Refer to Sample FMD Configuration File, on page 141 for a sample configuration file.

Important Though the FMD configuration can be included in the network service descriptor (NSD) for your VNF, it is recommended that the configuration for this functionality be maintained in a separate, FMD-specific NSD configuration file.

The health check must be enabled for UAS, VNFM, VNF and UEM before configuring fault suppression functionality.

To configure fault suppression functionality for UAS components:

1. Log on to the primary AutoIT VM as the root user.
2. Prepare the FMD configuration file for your deployment scenario. Your file should include the following parameters:

```
domain uas
  monitoring suppress-uas-fault [ overall ]
domain vnfm
  monitoring suppress-uas-fault [ overall ]
domain vnf-EM
  monitoring suppress-uas-fault [ api-endpoint ha-event cluster-ha]
domain vnf
  monitoring suppress-uas-fault [ overall ]
```

   Refer to Sample FMD Configuration File, on page 141 for a sample configuration file.

   NOTES:

   - **suppress-uas-fault**: This operator is optional and it accepts enum value in the YANG model.
   - **overall**: Suppresses faults for all the configured domains.
   - **api-endpoint**: This is applicable only to UEM domain. The fault is raised when EM applications/internal components are not in healthy state i.e. SCM/SLA/VNFM-PROXY is down.
   - **ha-event**: This is applicable to UAS and UEM domain. The fault is raised when EM HA endpoint is changed i.e. during HA switch over (one of the UAS VMs in a cluster rebooted and a switchover/failover has been performed resulting in the election of a new master).
   - **cluster-ha**: This is applicable to UAS and UEM domain. The fault is raised when EM VMs are failed to form HA cluster.
   - The **api-endpoint** operator is not applicable for the UAS and ESC domains as the UAS/ESC health check procedure takes care of these errors and corresponding recovery procedures.

3. Login to the ConfD CLI as the *admin* user.

```
confd_cli -u admin -C
```

4. Enter the *admin* user password when prompted.
5. Enter the ConfD configuration mode.

config

6. Load the FMD configuration file.

   load merge *<your_fmd_file_name>*.**cfg**

   commit

   end

7. Activate the FMD configuration.

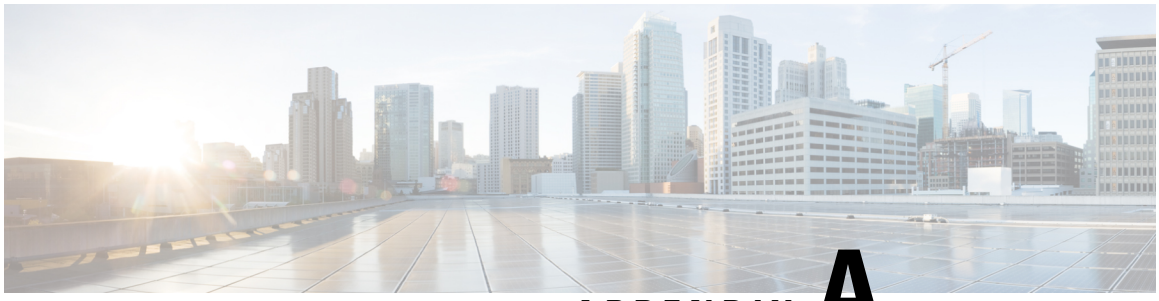
   activate nsd-id *<nsd_name>* **fmd** *<fmd_name>*

Important The output of this command is a transaction-id which can be used to monitor the deployment progress. If need be, the FMD configuration can be deactivated using the **deactivate** variant of this command.

8. Monitor the progress of the FMD creation by viewing transaction logs:

   show log *<transaction_id>* **| display xml**

   transaction_id is the ID displayed as a result of the **activate** command executed in step 7, on page 63.



APPENDIX A

Network Definitions (Layer 2 and 3)

Table 35: Layer 2 and 3 Network Definition, on page 85 is intended to be used as a template for recording your Ultra M network Layer 2 and Layer 3 deployments.

Some of the Layer 2 and 3 networking parameters identified in Table 35: Layer 2 and 3 Network Definition, on page 85 are configured directly on the UCS hardware via CIMC. Other parameters are configured as part of the VIM Orchestrator or VIM configuration. This configuration is done through various configuration files depending on the parameter:

- undercloud.conf
- network.yaml
- layout.yaml
- AutoDeploy Configuration file for the pod

Table 35: Layer 2 and 3 Network Definition

VLAN ID / Range	Network	Gateway	IP Range Start	IP Range End	Description	Where Configured	Routable?
External-Internet Meant for OSP-D Only							
100	192.168.1.0 /24	192.168.1.1			Internet access required: - 1 IP Address for OSP-D - 1 IP for default gateway	On Ultra M Manger Node hardware	Yes
External – Floating IP Addresses (Virtio)*							

VLAN ID / Range	Network	Gateway	IP Range Start	IP Range End	Description	Where Configured	Routable?
101	192.168.10.0 /24	192.168.10.1			Routable addresses required: - 3 IP addresses for Controllers - 1 VIP for master Controller Node (Horizon IP address) 4 Floating IP Addresses per VNF for management VMs (CF, VNFM, UEM, and UAS software modules) - 1 IP for default gateway	*network.yaml* and/or *layout.yaml***	Yes
Provisioning							
105	**192.0.0.0/ 8**		**192.200.0. 100**	**192.200.0. 254**	Required to provision all configuration via PXE boot from OSP-D for Ceph, Controller and Compute. Intel-On-Board Port 1 (1G).	*undercloud. conf*	No
IPMI-CIMC							
105	**192.0.0.0/ 8**		**192.100.0. 100**	**192.100.0. 254**		On UCS servers through CIMC	No
Tenant (Virtio)							
17	**11.117.0.0/ 24**				All Virtio based tenant networks. (MLOM)	*network.yaml* and/or *layout.yaml***	No

VLAN ID / Range	Network	Gateway	IP Range Start	IP Range End	Description	Where Configured	Routable?
Storage (Virtio)							
18	**11.118.0.0/ 24**				Required for Controllers, Computes and Ceph for read/write from and to Ceph. (MLOM)	*network.yaml* and/or *layout.yaml***	No
Storage-MGMT (Virtio)							
19	**11.119.0.0/ 24**				Required for Controllers and Ceph only as Storage Cluster internal network. (MLOM)	*network.yaml* and/or *layout.yaml***	No
Internal-API (Virtio)							
20	**11.120.0.0/ 24**				Required for Controllers and Computes for openstack manageability. (MLOM)	*network.yaml* and/or *layout.yaml***	No
Mgmt (Virtio)							
21	**172.16.181.0/ 24**		**172.16.181. 100**	**172.16.181. 254**	Tenant based virtio network on openstack.	*network.yaml* and/or *layout.yaml***	No
Other-Virtio							
1001: 1500					Tenant based virtio networks on openstack.	*network.yaml* and/or *layout.yaml***	No
SR-IOV (Phys-PCIe1)							

VLAN ID / Range	Network	Gateway	IP Range Start	IP Range End	Description	Where Configured	Routable?
2101: **2500**					Tenant SRIOV network on openstack. (Intel NIC on PCIe1) NOTE: A unique VLAN from this range is used by each VNF for the DI-internal network.	*network.yaml* and/or *layout.yaml***	Yes
SR-IOV (Phys-PCIe4)							
2501: **2900**					Tenant SRIOV network on openstack. (Intel NIC on PCIe4) NOTE: Ensure that the same DI-internal nework VLAN ID is configured for both PCIe1 and PCIe4 for the same VNF. (For example, if VLAN ID 2111 is configured for VNF1 on PCIe1, VLAN ID 2111 must be configured on PCIe4 for VNF1)	*network.yaml* and/or *layout.yaml***	Yes

VLAN ID / Range	Network	Gateway	IP Range Start	IP Range End	Description	Where Configured	Routable?

NOTE: **Bold underlined** text is provided as example configuration information. Your deployment requirements will vary. The IP addresses in **bold** text are the recommended address used for internal routing between VNF components. All other IP addresses and VLAN IDs may be changed/assigned.

* You can ensure that the same floating IP address can assigned to the AutoVNF, CF, UEM, and VNFM after a VM restart by configuring parameters in the AutoDeploy configuration file or the UWS service delivery configuration file. Refer to Table 36: Floating IP address Reuse Parameters, on page 89 for details.

** For Hyper-converged Ultra M models based on OpenStack 10, these parameters must configured in the both the *networks.yaml* and the *layout.yaml* files unless the VIM installation automation feature is used. Refer to the *Ultra Services Platform Deployment Automation Guide* for details.

Caution IP address ranges used for the Tenant (Virtio), Storage (Virtio), and Internal-API (Virtio) in *network.yaml* cannot conflict with the IP addresses specified in *layout.yaml* for the corresponding networks. Address conflicts will prevent the VNF from functioning properly.

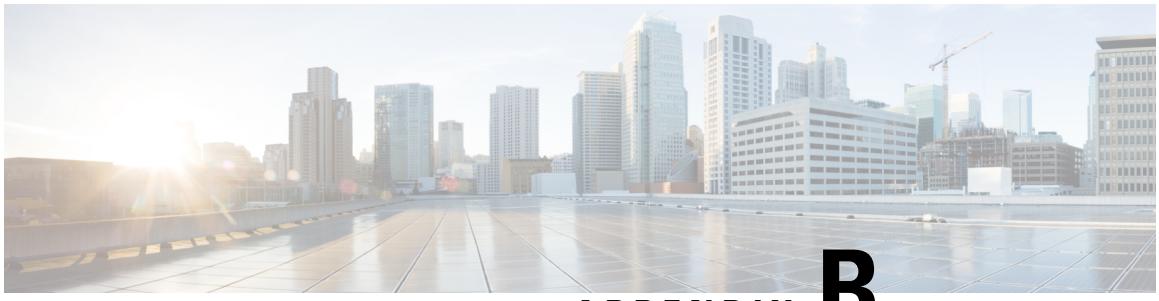
Table 36: Floating IP address Reuse Parameters

Component	Construct	AutoDeploy Configuration File Parameters	UWS Service Deployment Configuration File
AutoVNF	autovnfd	**networks management floating-ip true** **networks management ha-vip** <*vip_address*> **networks management floating-ip-address** <*floating_address*>	<management> <---SNIP---> **<floating-ip>true </floating-ip>** **<ha-vip>** *vip_address***</ha-vip>** **<floating-ip-address>** *floating_address* **</floating-ip-address>** </management>
VNFM	vnfmd	**floating-ip true** **ha-vip** <*vip_address*> **floating-ip-address** <*floating_address*>	<management> <---SNIP---> **<floating-ip>true </floating-ip>** **<ha-vip>** *vip_address***</ha-vip>** **<floating-ip-address>***floating_address* **</floating-ip-address>** </management>

Component	Construct	AutoDeploy Configuration File Parameters	UWS Service Deployment Configuration File
UEM	vnfd	**vnf-em ha-vip** <*vip_address*> **vnf-em floating-ip true** **vnf-em floating-ip-address** <*floating_address*>	<vnf-em> <---SNIP---> **<ha-vip>** *vip_address***</ha-vip>** <---SNIP---> **<floating-ip>true </floating-ip>** **<floating-ip-address>** *floating_address* **</floating-ip-address>** <---SNIP---> </vnf-em>
CF	vnfd	interfaces mgmt <---SNIP---> **enable-ha-vip** <*vip_address*> **floating-ip true** **floating-ip-address** <*floating_address*> <---SNIP--->	<interfaces> <---SNIP---> **<enable-ha-vip>** *vip_address***</enable-ha-vip>** **<floating-ip>true </floating-ip>** **<floating-ip-address>** *floating_address* **</floating-ip-address>** <---SNIP---> </interfaces>
Controllers	networking network-types external	<---SNIP---> **ip-prefix**<*floating_address_ network* >/<*mask_bits*> **vlan-id** <*vlan_id*> **allocation-pool start**<*1st_floating_address_for_controllers*> **allocation-pool end** <*4th_floating_address_for_ controllers* > **default-route**<*actual_gw_Ip_ddress of_floating_ip_network*> <---SNIP--->	

Note This functionality is disabled by default. Set the **floating-ip** and/or **<floating-ip>** parameters to *true* to enable this functionality.

Note Prior to assigning floating and virtual IP addresses, make sure that they are not already allocated through OpenStack. If the addresses are already allocated, then they must be freed up for use or you must assign a new IP address that is available in the VIM.



APPENDIX B

Ultra M MIB

Important Not all aspects of this MIB are supported in this release. Refer to Health Monitoring in the Ultra M Solution, on page 59 for information on the capabilities supported in this release.

```
-- ****************************************************************
-- CISCO-ULTRAM-MIB.my
-- Copyright (c) 2017 by Cisco Systems, Inc.
-- All rights reserved.
--
-- ****************************************************************

CISCO-ULTRAM-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY,
    OBJECT-TYPE,
    NOTIFICATION-TYPE,
    Unsigned32
        FROM SNMPv2-SMI
    MODULE-COMPLIANCE,
    NOTIFICATION-GROUP,
    OBJECT-GROUP
        FROM SNMPv2-CONF
    TEXTUAL-CONVENTION,
    DateAndTime
        FROM SNMPv2-TC
    ciscoMgmt
        FROM CISCO-SMI;


ciscoUltramMIB MODULE-IDENTITY
    LAST-UPDATED    "201711200000Z"
    ORGANIZATION    "Cisco Systems, Inc."
    CONTACT-INFO
            "Cisco Systems
            Customer Service

            Postal: 170 W Tasman Drive
            San Jose, CA  95134
            USA

            Tel: +1 800 553-NETS"
    DESCRIPTION
        "The MIB module to management of Cisco Ultra Services Platform
```

```
        (USP) also called Ultra-M Network Function Virtualization (NFV)
        platform. The Ultra-M platform is Cisco validated turnkey
        solution based on ETSI(European Telecommunications Standards
        Institute) NFV architetcure.

        It comprises of following architectural domains:

         1. Management and Orchestration (MANO), these componets
        enables infrastructure virtualization and life cycle management
        of Cisco Ultra Virtual Network Functions (VNFs).

         2. NFV Infrastructure (NFVI), set of physical resources to
        provide NFV infrastructre, for example servers, switch, chassis,
        and so on.

         3. Virtualized Infrastructure Manager (VIM)

         4. One or more Ultra VNFs.

        Ultra-M platform provides a single point of management
        (including SNMP, APIs, Web Console and CLI/Telnet Console) for
        the resources across these domains within NFV PoD (Point of
        Delivery).
        This is also called Ultra-M manager throughout the context of
        this MIB."
    REVISION        "201711200000Z"
    DESCRIPTION
        "- Added cultramSiteId as Index to cultramFaultEntry Table."
    REVISION        "201707050000Z"
    DESCRIPTION
        "- cultramFaultDomain changed to read-only in compliance.
        - Added a new fault code serviceFailure under
        'CultramFaultCode'.
        - Added a new notification cultramFaultClearNotif.
        - Added new notification group ciscoUltramMIBNotifyGroupExt.
        - Added new compliance group ciscoUltramMIBModuleComplianceRev01
        which deprecates ciscoUltramMIBModuleCompliance."
    REVISION        "201706260000Z"
    DESCRIPTION
        "Initial version of this MIB module."
    ::= { ciscoMgmt 849 }



CFaultCode ::= TEXTUAL-CONVENTION
    STATUS          current
    DESCRIPTION
        "A code identifying a class of fault."
    SYNTAX          INTEGER  {
                        other(1), -- Other events
                        networkConnectivity(2), -- Network Connectivity
                                                -- Failure Events.
                        resourceUsage(3), -- Resource Usage Exhausted
                                          -- Event.
                        resourceThreshold(4), -- Resource Threshold
                                              -- crossing alarms
                        hardwareFailure(5), -- Hardware Failure Events
                        securityViolation(6), -- Security Alerts
                        configuration(7), -- Config Error Events
                        serviceFailure(8) -- Process/Service failures

                    }

CFaultSeverity ::= TEXTUAL-CONVENTION
```

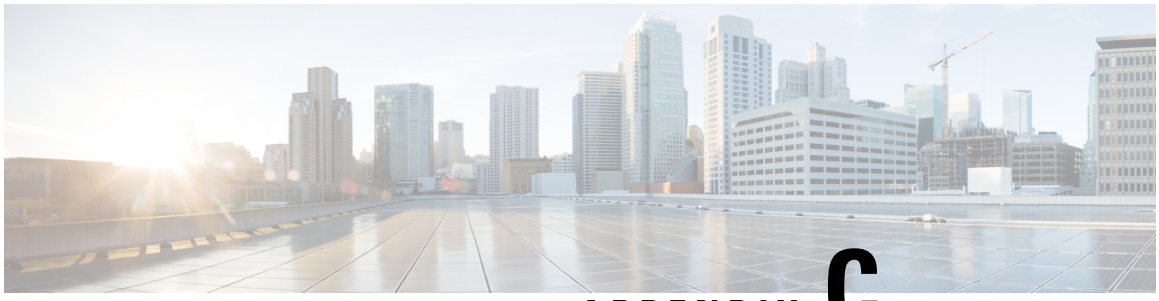
```
    STATUS          current
    DESCRIPTION
        "A code used to identify the severity of a fault."
    SYNTAX          INTEGER  {
                        emergency(1), -- System level FAULT impacting
                                      -- multiple VNFs/Services
                        critical(2), -- Critical Fault specific to
                                     -- VNF/Service
                        major(3), -- component level failure within
                                  -- VNF/service.
                        alert(4), -- warning condition for a service/VNF,
                                  -- may eventually impact service.
                        informational(5) -- informational only, does not
                                         -- impact service
                    }

CFaultDomain ::= TEXTUAL-CONVENTION
    STATUS          current
    DESCRIPTION
        "A code used to categorize Ultra-M fault domain."
    SYNTAX          INTEGER  {
                        hardware(1), -- Harware including Servers, L2/L3
                                     -- Elements
                        vimOrchestrator(2), -- VIM under-cloud
                        vim(3), -- VIM manager such as OpenStack
                        uas(4), -- Ultra Automation Services Modules
                        vnfm(5), -- VNF manager
                        vnfEM(6), -- Ultra VNF Element Manager
                        vnf(7) -- Ultra VNF
                    }
-- Textual Conventions definition will be defined before this line

ciscoUltramMIBNotifs  OBJECT IDENTIFIER
    ::= { ciscoUltramMIB 0 }

ciscoUltramMIBObjects  OBJECT IDENTIFIER
    ::= { ciscoUltramMIB 1 }

ciscoUltramMIBConform  OBJECT IDENTIFIER
    ::= { ciscoUltramMIB 2 }

-- Conformance Information Definition

ciscoUltramMIBCompliances  OBJECT IDENTIFIER
    ::= { ciscoUltramMIBConform 1 }

ciscoUltramMIBGroups  OBJECT IDENTIFIER
    ::= { ciscoUltramMIBConform 2 }


ciscoUltramMIBModuleCompliance MODULE-COMPLIANCE
    STATUS          deprecated
    DESCRIPTION
        "The compliance statement for entities that support
        the Cisco Ultra-M Fault Managed Objects"
    MODULE          -- this module
    MANDATORY-GROUPS {
                        ciscoUltramMIBMainObjectGroup,
                        ciscoUltramMIBNotifyGroup
                    }
    ::= { ciscoUltramMIBCompliances 1 }

ciscoUltramMIBModuleComplianceRev01 MODULE-COMPLIANCE
    STATUS          current
```

```
    DESCRIPTION
        "The compliance statement for entities that support
        the Cisco Ultra-M Fault Managed Objects."
    MODULE          -- this module
    MANDATORY-GROUPS {
                        ciscoUltramMIBMainObjectGroup,
                        ciscoUltramMIBNotifyGroup,
                        ciscoUltramMIBNotifyGroupExt
                    }

    OBJECT          cultramFaultDomain
    MIN-ACCESS      read-only
    DESCRIPTION
        "cultramFaultDomain is read-only."
    ::= { ciscoUltramMIBCompliances 2 }

ciscoUltramMIBMainObjectGroup OBJECT-GROUP
    OBJECTS         {
                        cultramNFVIdenity,
                        cultramFaultDomain,
                        cultramFaultSource,
                        cultramFaultCreationTime,
                        cultramFaultSeverity,
                        cultramFaultCode,
                        cultramFaultDescription
                    }
    STATUS          current
    DESCRIPTION
        "A collection of objects providing Ultra-M fault information."
    ::= { ciscoUltramMIBGroups 1 }

ciscoUltramMIBNotifyGroup NOTIFICATION-GROUP
   NOTIFICATIONS    { cultramFaultActiveNotif }
    STATUS          current
    DESCRIPTION
        "The set of Ultra-M notifications defined by this MIB"
    ::= { ciscoUltramMIBGroups 2 }

ciscoUltramMIBNotifyGroupExt NOTIFICATION-GROUP
   NOTIFICATIONS    { cultramFaultClearNotif }
    STATUS          current
    DESCRIPTION
        "The set of Ultra-M notifications defined by this MIB"
    ::= { ciscoUltramMIBGroups 3 }

cultramFaultTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF CUltramFaultEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "A table of Ultra-M faults. This table contains active
        faults."
    ::= { ciscoUltramMIBObjects 1 }

cultramFaultEntry OBJECT-TYPE
    SYNTAX          CUltramFaultEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "An entry in the Ultra-M fault table."
    INDEX           { cultramSiteId, cultramFaultIndex }
    ::= { cultramFaultTable 1 }

CUltramFaultEntry ::= SEQUENCE {
```

```
        cultramSiteId            OCTET STRING,
        cultramFaultIndex        Unsigned32,
        cultramNFVIdenity        OCTET STRING,
        cultramFaultDomain       CFaultDomain,
        cultramFaultSource       OCTET STRING,
        cultramFaultCreationTime DateAndTime,
        cultramFaultSeverity     CFaultSeverity,
        cultramFaultCode         CFaultCode,
        cultramFaultDescription  OCTET STRING
}

cultramSiteId OBJECT-TYPE
    SYNTAX          OCTET STRING (SIZE  (1..32))
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This object uniquely identifies a specific instance of a
        Ultra-M fault at site level."
    ::= { cultramFaultEntry 1 }

cultramFaultIndex OBJECT-TYPE
    SYNTAX          Unsigned32
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This object uniquely identifies a specific instance of a
        Ultra-M fault.

        For example, if two separate computes have a service level
        Failure, then each compute will have a fault instance with a
        unique index."
    ::= { cultramFaultEntry 2 }

cultramNFVIdenity OBJECT-TYPE
    SYNTAX          OCTET STRING (SIZE  (1..512))
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION
        "This object uniquely identifies the Ultra-M PoD on which this
        fault is occurring.
        For example, this identity can include host-name as well
        management IP where manager node is running,
        'Ultra-M-San-Francisco/172.10.185.100'."
    ::= { cultramFaultEntry 3 }

cultramFaultDomain OBJECT-TYPE
    SYNTAX          CFaultDomain
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION
        "A unique Fault Domain that has fault."
    ::= { cultramFaultEntry 4 }

cultramFaultSource OBJECT-TYPE
    SYNTAX          OCTET STRING (SIZE  (1..512))
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "This object uniquely identifies the resource with the fault
        domain where this fault is occurring. For example, this can
        include host-name as well management IP of the resource,
        'UCS-C240-Server-1/192.100.0.1'."
    ::= { cultramFaultEntry 5 }
```

```
cultramFaultCreationTime OBJECT-TYPE
    SYNTAX          DateAndTime
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "The date and time when the fault was occured."
    ::= { cultramFaultEntry 6 }

cultramFaultSeverity OBJECT-TYPE
    SYNTAX          CFaultSeverity
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "A code identifying the perceived severity of the fault."
    ::= { cultramFaultEntry 7 }

cultramFaultCode OBJECT-TYPE
    SYNTAX          CFaultCode
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "A code uniquely identifying the fault class."
    ::= { cultramFaultEntry 8 }

cultramFaultDescription OBJECT-TYPE
    SYNTAX          OCTET STRING (SIZE  (1..2048))
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "A human-readable message providing details about the fault."
    ::= { cultramFaultEntry 9 }



cultramFaultActiveNotif NOTIFICATION-TYPE
    OBJECTS         {
                        cultramNFVIdenity,
                        cultramFaultDomain,
                        cultramFaultSource,
                        cultramFaultCreationTime,
                        cultramFaultSeverity,
                        cultramFaultCode,
                        cultramFaultDescription
                    }
    STATUS          current
    DESCRIPTION
        "This notification is generated by a Ultra-M manager whenever a
        fault is active."
   ::= { ciscoUltramMIBNotifs 1 }

cultramFaultClearNotif NOTIFICATION-TYPE
    OBJECTS         {
                        cultramNFVIdenity,
                        cultramFaultDomain,
                        cultramFaultSource,
                        cultramFaultCreationTime,
                        cultramFaultSeverity,
                        cultramFaultCode,
                        cultramFaultDescription
                    }
    STATUS          current
    DESCRIPTION
        "This notification is generated by a Ultra-M manager whenever a
        fault is cleared."
```

```
    ::= { ciscoUltramMIBNotifs 2 }

END
```



APPENDIX C

Ultra M Component Event Severity and Fault Code Mappings

Events are assigned to one of the following severities (refer to CFaultSeverity in Ultra M MIB, on page 91):

- emergency(1), -- System level FAULT impacting multiple VNFs/Services
- critical(2), -- Critical Fault specific to VNF/Service
- major(3), -- component level failure within VNF/service.
- alert(4), -- warning condition for a service/VNF, may eventually impact service.
- informational(5) -- informational only, does not impact service

Events are also mapped to one of the following fault codes (refer to cFaultCode in the Ultra M MIB, on page 91):

- other(1), -- Other events
- networkConnectivity(2), -- Network Connectivity -- Failure Events.
- resourceUsage(3), -- Resource Usage Exhausted -- Event.
- resourceThreshold(4), -- Resource Threshold -- crossing alarms
- hardwareFailure(5), -- Hardware Failure Events
- securityViolation(6), -- Security Alerts
- configuration(7), -- Config Error Events serviceFailure(8) -- Process/Service failures

The Ultra M Manager Node serves as an aggregator for events received from the different Ultra M components. These severities and fault codes are mapped to those defined for the specific components. The information in this section provides severity mapping information for the following:

- OpenStack Events, on page 100
- UCS Server Events, on page 104
- UAS Events, on page 104
- ESC VM Events, on page 105

OpenStack Events

Component: Ceph

Table 37: Component: Ceph

Failure Type	Ultra M Severity	Fault Code
CEPH Status is not healthy	Emergency	serviceFailure
One or more CEPH monitors are down	Emergency	serviceFailure
Disk usage exceeds threshold	Critical	resourceThreshold
One or more OSD nodes are down	Critical	serviceFailure
One or more OSD disks are failed	Critical	resourceThreshold
One of the CEPH monitor is not healthy.	Major	serviceFailure
One or more CEPH monitor restarted.	Major	serviceFailure
OSD disk weights not even across the board.		resourceThreshold

Component: Cinder

Table 38: Component: Cinder

Failure Type	Ultra M Severity	Fault Code
Cinder Service is down	Emergency	serviceFailure

Component: Neutron

Table 39: Component: Neutron

Failure Type	Ultra M Severity	Fault Code
One of Neutron Agent Down	Critical	serviceFailure

Component: Nova

Table 40: Component: Nova

Failure Type	Ultra M Severity	Fault Code
Compute service down	Critical	serviceFailure

Component: NTP

Table 41: Component: NTP

Failure Type	Ultra M Severity	Fault Code
NTP skew limit exceeds configured threshold.	Critical	serviceFailure

Component: PCS

Table 42: Component: PCS

Failure Type	Ultra M Severity	Fault Code
One or more controller nodes are down	Critical	serviceFailure
Ha-proxy is down on one of the node	Major	serviceFailure
Galera service is down on one of the node.	Critical	serviceFailure
Rabbitmq is down.	Critical	serviceFailure
Radis Master is down.	Emergency	serviceFailure
One or more Radis Slaves are down.	Critical	serviceFailure
corosync/pacemaker/pcsd - not all daemons active	Critical	serviceFailure
Cluster status changed.	Major	serviceFailure
Current DC not found.	Emergency	serviceFailure
Not all PCDs are online.	Critical	serviceFailure
Stonith service is down on one or more nodes	Critical	serviceFailure

Component: Rabbitmqctl

Table 43: Component: Rabbitmqctl

Failure Type	Ultra M Severity	Fault Code
Cluster Status is not healthy	Emergency	serviceFailure

Component: Services

Table 44: Component: Services

Failure Type	Ultra M Severity	Fault Code
Service is disabled.	Critical	serviceFailure
Service is down.	Emergency	serviceFailure
Service Restarted.	Major	serviceFailure

The following OpenStack services are monitored:

- Controller Nodes:
  - httpd.service
  - memcached
  - mongod.service
  - neutron-dhcp-agent.service
  - neutron-l3-agent.service
  - neutron-metadata-agent.service
  - neutron-openvswitch-agent.service
  - neutron-server.service
  - ntpd.service
  - openstack-cinder-api.service
  - openstack-cinder-scheduler.service
  - openstack-glance-api.service
  - openstack-glance-registry.service
  - openstack-heat-api-cfn.service
  - openstack-heat-api-cloudwatch.service
  - openstack-heat-api.service
  - openstack-heat-engine.service

- openstack-nova-api.service
- openstack-nova-conductor.service
- openstack-nova-consoleauth.service
- openstack-nova-novncproxy.service
- openstack-nova-scheduler.service
- openstack-swift-account-auditor.service
- openstack-swift-account-reaper.service
- openstack-swift-account-replicator.service
- openstack-swift-account.service
- openstack-swift-container-auditor.service
- openstack-swift-container-replicator.service
- openstack-swift-container-updater.service
- openstack-swift-container.service
- openstack-swift-object-auditor.service
- openstack-swift-object-replicator.service
- openstack-swift-object-updater.service
- openstack-swift-object.service
- openstack-swift-proxy.service

- Compute Nodes:
  - ceph-mon.target
  - ceph-radosgw.target
  - ceph.target
  - libvirtd.service
  - neutron-sriov-nic-agent.service
  - neutron-openvswitch-agent.service
  - ntpd.service
  - openstack-nova-compute.service
  - openvswitch.service

- OSD Compute Nodes:
  - ceph-mon.target
  - ceph-radosgw.target

- ceph.target
- libvirtd.service
- neutron-sriov-nic-agent.service
- neutron-openvswitch-agent.service
- ntpd.service
- openstack-nova-compute.service
- openvswitch.service

UCS Server Events

UCS Server events are described here:https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/ts/faults/reference/ErrMess/FaultsIntroduction.html

The following table maps the UCS severities to those within the Ultra M MIB.

Table 45: UCS Server Severities

UCS Server Severity	Ultra M Severity	Fault Code
Critical	Critical	hardwareFailure
Info	Informational	hardwareFailure
Major	Major	hardwareFailure
Warning	Alert	hardwareFailure
Alert	Alert	hardwareFailure
Cleared	Informational	Not applicable

UAS Events

Table 46: UAS Events

Failure Type	Ultra M Severity	Fault Code
UAS Service Failure	Critical	serviceFailure*
UAS Service Recovered	Informational	serviceFailure*
* *serviceFailure* is used except where the Ultra M Health Monitor is unable to connect to any of the modules. In this case, the fault code is set to *networkConnectivity*.		

ESC VM Events

By default, the Ultra M Manager continuously monitors and processes VNF VM event notifications from ESC as reported through NETCONF.

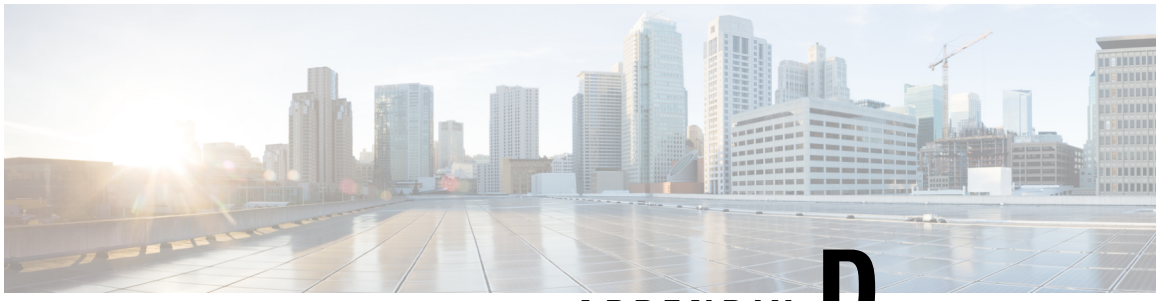
Important In release 6.0, the feature that enables monitoring of UGP VNFCs using ESC was not fully qualified and made available only for testing purposes. In 6.2 and later releases, this feature has been fully qualified for use in the appropriate deployment scenarios. For more information, contact your Cisco Accounts representative.

Table 47: ESC VM Event Severities

ESC Event	Ultra M Severity	Fault Code	VNFR State
VM_DEPLOYED	Major	Service Failure	deployed
VM_ALIVE	Info	Other	alive
VM_UNDEPLOYED	Critical	Service Failure	offline
VM_REBOOTED	Major	Service Failure	rebooting
VM_RECOVERY_REBOOT	Major	Service Failure	If the event completion status is successful, then it is rebooting. If the event completion status is failure, then it is an error.
VM_RECOVERY_UNDEPLOYED	Critical	Service Failure	If the event completion status is successful, then it is offline. If the event completion status is failure, then it is an error.
VM_RECOVERY_DEPLOYED	Major	Service Failure	If the event completion status is successful, then it is rebooting. If the event completion status is failure, then it is an error.

Ultra M Component Event Severity and Fault Code Mappings

ESC Event	Ultra M Severity	Fault Code	VNFR State
VM_RECOVERY_COMPLETE	Info	Other	If the event completion status is successful, then it is alive. If the event completion status is failure, then it is an error.
VM_STOPPED	Alert	Service Failure	stop_requested



APPENDIX D

Ultra M Troubleshooting

Ultra M Component Reference Documentation

The following sections provide links to troubleshooting information for the various components that comprise the Ultra M solution.

UCS C-Series Server

- Obtaining Showtech Support to TAC
- Display of system Event log events
- Display of CIMC Log
- Run Debug Firmware Utility
- Run Diagnostics CLI
- Common Troubleshooting Scenarios
- Troubleshooting Disk and Raid issues
- DIMM Memory Issues
- Troubleshooting Server and Memory Issues
- Troubleshooting Communication Issues

Nexus 9000 Series Switch

- Troubleshooting Installations, Upgrades, and Reboots
- Troubleshooting Licensing Issues
- Troubleshooting Ports

- Troubleshooting vPCs
- Troubleshooting VLANs
- Troubleshooting STP
- Troubleshooting Routing
- Troubleshooting Memory
- Troubleshooting Packet Flow Issues
- Troubleshooting PowerOn Auto Provisioning
- Troubleshooting the Python API
- Troubleshooting NX-API
- Troubleshooting Service Failures
- Before Contacting Technical Support
- Troubleshooting Tools and Methodology

Catalyst 2960 Switch

- Diagnosing Problems
- Switch POST Results
- Switch LEDs
- Switch Connections
- Bad or Damaged Cable
- Ethernet and Fiber-Optic Cables
- Link Status
- 10/100/1000 Port Connections
- 10/100/1000 PoE+ Port Connections
- SFP and SFP+ Module
- Interface Settings
- Ping End Device
- Spanning Tree Loops
- Switch Performance
- Speed, Duplex, and Autonegotiation
- Autonegotiation and Network Interface Cards
- Cabling Distance
- Clearing the Switch IP Address and Configuration

- Finding the Serial Number
- Replacing a Failed Stack Member

Red Hat

- Troubleshooting Director issue
- Backup and Restore Director Undercloud

OpenStack

- Red Hat Openstack Troubleshooting commands and sceanrios

UAS

Refer to the *USP Deployment Automation Guide*.

UGP

Refer to the *Ultra Gateway Platform System Administration Guide*.

Collecting Support Information

From UCS:

- Collect support information:

```
chassis show tech support
show tech support (if applicable)
```

- Check which UCS MIBS are being polled (if applicable). Refer to https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/sw/mib/c-series/b_UCS_Standalone_C-Series_MIBRef/b_UCS_Standalone_C-Series_MIBRef_chapter_0100.html

From Host/Server/Compute/Controler/Linux:

- Identify if Passthrought/SR-IOV is enabled.
- Run sosreport:

Note This functionality is enabled by default on Red Hat, but not on Ubuntu. It is recommended that you enable *sysstat* and *sosreport* on Ubuntu (run **apt-get install sysstat** and **apt-get install sosreport**). It is also recommended that you install sysstat on Red Hat (run **yum install sysstat**).

- Get and run the **os_ssd_pac** script from Cisco:
  - Compute (all):

```
./os_ssd_pac.sh -a
./os_ssd_pac.sh -k -s
```

Note For initial collection, it is always recommended to include the **-s** option (sosreport). Run **./os_ssd_pac.sh -h** for more information.

  - Controller (all):

```
./os_ssd_pac.sh -f
./os_ssd_pac.sh -c -s
```

Note For initial collection it is always recommended to include the -s option (sosreport). Run **./os_ssd_pac.sh -h** for more information.

- For monitoring purposes, from *crontab* use option: **-m** (for example run every 5 or 10 minutes)

From Switches

From all switches connected to the Host/Servers. (This also includes other switches which have same vlans terminated on the Host/Servers.)

```
show tech-support
syslogs
snmp traps
```

Note It is recommended that mac-move notifications are enabled on all switches in network by running mac address-table notification mac-move.

From ESC (Active and Standby)

Note It is recommended that you take a backup of the software and data before performing any of the following operations. Backups can be taken by executing opt/cisco/esc/esc-scripts/esc_dbtool.py backup. (Refer to https://www.cisco.com/c/en/us/td/docs/net_mgmt/elastic_services_controller/2-3/user/guide/Cisco-Elastic-Services-Controller-User-Guide-2-3/Cisco-Elastic-Services-Controller-User-Guide-2-2_chapter_010010.html#id_18936 for more information.)

```
/opt/cisco/esc/esc-scripts/health.sh
/usr/bin/collect_esc_log.sh
./os_ssd_pac -a
```

From UAS

- Monitor ConfD:

```
confd -status
confd --debug-dump /tmp/confd_debug-dump
confd --printlog /tmp/confd_debug-dump
```

Note Once the file **/tmp/confd_debug-dump**> is collected, it can be removed (**rm /tmp/confd_debug-dump**).

- Monitor UAS Components:

```
source /opt/cisco/usp/uas/confd-6.1/confdrc
confd_cli -u admin -C
show uas
show uas ha-vip
show uas state
show confd-state
show running-config
show transactions date-and-time
show log | display xml
show errors displaylevel 64
show notification stream uas_notify last 1000
show autovnf-oper:vnfm
show autovnf-oper:vnf-em
show autovnf-oper:vdu-catalog
show autovnf-oper:transactions
show autovnf-oper:network-catalog
show autovnf-oper:errors
show usp
show confd-state internal callpoints
```

```
show confd-state webui listen
show netconf-state
```

Important Executing the **confd_cli -u admin -C** command prompts you to enter *admin user* password.

- Monitor Zookeeper:

```
/opt/cisco/usp/packages/zookeeper/current/bin/zkCli.sh -server
x.x.x.x:2181 ls /config/control-function
/opt/cisco/usp/packages/zookeeper/current/bin/zkCli.sh -server
x.x.x.x:2181 ls /config/element-manager
/opt/cisco/usp/packages/zookeeper/current/bin/zkCli.sh -server
x.x.x.x:2181 ls /config/session-function
/opt/cisco/usp/packages/zookeeper/current/bin/zkCli.sh -server
x.x.x.x:2181 ls /
/opt/cisco/usp/packages/zookeeper/current/bin/zkCli.sh -server
x.x.x.x:2181 ls /stat
/opt/cisco/usp/packages/zookeeper/current/bin/zkCli.sh -server
x.x.x.x:2181 ls /log
```

Note Port number (2181) is not mandatory when executing the **zkCli.sh** command.

- Collect Zookeeper data:

```
cd /tmp
tar zcfv zookeeper_data.tgz /var/lib/zookeeper/data/version-2/
ls -las /tmp/zookeeper_data.tgz
```

- Get support details

```
./os_ssd_pac -a
```

From UEM (Active and Standby)

- Collect logs

```
/opt/cisco/em-scripts/collect-em-logs.sh
```

- Monitor NCS:

```
ncs -status
ncs --debug-dump /tmp/ncs_debug-dump
ncs --printlog /tmp/ncs_debug-dump
```

Note Once the file **/tmp/ncs_debug-dump** is collected, it can be removed (**rm /tmp/ncs_debug-dump**).

- Collect support details:

```
./os_ssd_pac -a
```

From UGP (Through StarOS)

- Collect the multiple outputs of the **show support details**.

Note It is recommended to collect at least two samples, 60 minutes apart if possible.

- Collect raw bulkstats before and after events.
- Collect syslogs and snmp traps before and after events.
- Collect PCAP or sniffer traces of all relevant interfaces if possible.

Note Familiarize yourself with how running SPAN/RSPAN on Nexus and Catalyst switches. This is important for resolving Passthrough/SR-IOV issues.

- Collect console outputs from all nodes.
- Export CDRs and EDRs.
- Collect the outputs of **monitor subscriber next-call** or **monitor protocol** depending on the activity
- Refer to https://supportforums.cisco.com/sites/default/files/cisco_asr5000_asr5500_troubleshooting_guide.pdf for more information.

About Ultra M Manager Log Files

All Ultra M Manager log files are created under "/var/log/cisco/ultram-manager".

```
cd /var/log/cisco/ultram-manager
ls -alrt
```

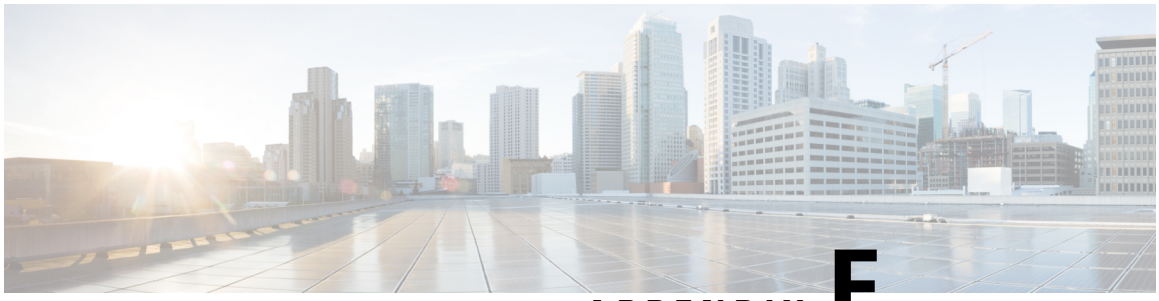
Example output:

```
total 116
drwxr-xr-x. 3 root root  4096 Sep 10 17:41 ..
-rw-r--r--. 1 root root     0 Sep 12 15:15 ultram_health_snmp.log
-rw-r--r--. 1 root root   448 Sep 12 15:16 ultram_health_uas.report
```

```
-rw-r--r--. 1 root root   188 Sep 12 15:16 ultram_health_uas.error
-rw-r--r--. 1 root root   580 Sep 12 15:16 ultram_health_uas.log
-rw-r--r--. 1 root root 24093 Sep 12 15:16 ultram_health_ucs.log
-rw-r--r--. 1 root root  8302 Sep 12 15:16 ultram_health_os.error
drwxr-xr-x. 2 root root  4096 Sep 12 15:16 .
-rw-r--r--. 1 root root 51077 Sep 12 15:16 ultram_health_os.report
-rw-r--r--. 1 root root  6677 Sep 12 15:16 ultram_health_os.log
```

NOTES:

- The files are named according to the following conventions:
  - ultram_health_os: Contain information related to OpenStack
  - ultram_health_ucs: Contain information related to UCS
  - ultram_health_uas: Contain information related to UAS
- Files with the "*.log" extension contain debug/error outputs from different components. These files get added to over time and contain useful data for debugging in case of issues.
- Files with the ".report" extension contain the current report. These files get created on every tun.
- Files with the ".error" extension contain actual data received from the nodes as part of health monitoring. These are the events that causes the Ultra M health monitor to send traps out. These files are updated every time a component generates an event.



APPENDIX E

Using the UCS Utilities Within the Ultra M Manager

This appendix describes the UCS facilities within the Ultra M Manager.

Overview

Cisco UCS server BIOS, MLOM, and CIMC software updates may be made available from time to time.

Utilities have been added to the AutoIT software module to simplify the process of upgrading the UCS server software (firmware) within the Ultra M solution.

These utilities are available through a script called *ultram_ucs_utils.py* located in the */opt/cisco/usp/ultram-manager* directory. Refer to ultram_ucs_utils.py Help, on page 139 for more information on this script.

NOTES:

- This functionality is currently supported only with Ultra M deployments based on OSP 10 or OSP 13 and that leverage the Hyper-Converged architecture.
- UCS server utilities are provided through the AutoIT software module which is deployed as a VM on the Ultra M Manager Node. As such, AutoIT must be deployed prior to using the utilities.
- You should only upgrade your UCS server software to versions that have been validated for use within the Ultra M solution.
- All UCS servers within the Ultra M solution stack should be upgraded to the same firmware versions.

- Though it is highly recommended that all server upgrades be performed during a single maintenance window, it is possible to perform the upgrade across multiple maintenance windows based on Node type (e.g. Compute, OSD Compute, and Controller).

There are two upgrade scenarios:

- **Upgrading servers in an existing deployment.** In the scenario, the servers are already in use hosting the Ultra M solution stack. This upgrade procedure is designed to maintain the integrity of the stack.
  - Compute Nodes are upgraded in parallel.
  - OSD Compute Nodes are upgraded sequentially.
  - Controller Nodes are upgraded sequentially.
- **Upgrading bare metal servers.** In this scenario, the bare metal servers have not yet been deployed within the Ultra M solution stack. This upgrade procedure leverages the parallel upgrade capability within the UCS utilities to upgrade the servers in parallel.

To use the UCS utilities to upgrade software for UCS servers in an existing deployment:

1. Perform Pre-Upgrade Preparation.
2. Shutdown the ESC VMs, on page 119.
3. Upgrade the Compute Node Server Software.
4. Upgrade the OSD Compute Node Server Software, on page 123.
5. Restart the UAS and ESC (VNFM) VMs, on page 126.
6. Upgrade the Controller Node Server Software, on page 126.
7. Upgrade Firmware on the OSP-D Server/Ultra M Manager Node, on page 134.

To use UItra M Manager UCS utilities to upgrade software for bare metal UCS servers:

1. Perform Pre-Upgrade Preparation.
2. Upgrade Firmware on UCS Bare Metal, on page 129.
3. Upgrade Firmware on the OSP-D Server/Ultra M Manager Node, on page 134.

Perform Pre-Upgrade Preparation

Prior to performing the actual UCS server software upgrade, you must perform the steps in this section to prepare your environment for the upgrade.

NOTES:

- These instructions assume that all hardware is fully installed, cabled, and operational.
- These instructions assume that the VIM Orchestrator and VIM have been successfully deployed.
- These instructions assume that AutoIT has been successfully deployed.
- UCS server software is distributed separately from the USP software ISO.

To prepare your environment prior to upgrading the UCS server software:

1. Log on to the AutoIT VM running on the Ultra M Manager Node.
2. Create a directory called */var/www/html/firmwares* to contain the upgrade files.

   mkdir -p /var/www/html/firmwares
3. Download the UCS software ISO to the directory you just created.

   UCS software is available for download from https://software.cisco.com/download/type.html?mdfid=286281356&flowid=71443
4. Extract the *bios.cap* file.

   mkdir /tmp/UCSISO

   sudo mount -t iso9660 -o loop ucs-c240m4-huu-*<version>*.iso UCSISO/

   ```
   mount: /dev/loop2 is write-protected, mounting read-only
   ```

   cd UCSISO/

   ls

   ```
   EFI                 GETFW           isolinux  Release-Notes-DN2.txt  squashfs_img.md5
   tools.squashfs.enc
   firmware.squashfs.enc  huu-release.xml  LiveOS    squashfs_img.enc.md5
   TOC_DELNORTE2.xml  VIC_FIRMWARE
   ```

   cd GETFW/

   ls

   ```
   getfw  readme.txt
   ```

   mkdir -p /tmp/HUU
   sudo ./getfw -s /tmp/ucs-c240m4-huu-*<version>*.iso -d /tmp/HUU

   ```
   Nothing was selected hence getting only CIMC and BIOS
   FW/s available at '/tmp/HUU/ucs-c240m4-huu-<version>'
   ```

   cd /tmp/HUU/ucs-c240m4-huu-*<version>*/bios/

   ls

   ```
   bios.cap
   ```
5. Copy the *bios.cap* and *huu.iso* to the */var/www/html/firmwares/* directory.

   sudo cp bios.cap /var/www/html/firmwares/

   ls -lrt /var/www/html/firmwares/

   ```
   total 692228
   -rw-r--r--.  1 root root 692060160 Sep 28 22:43 ucs-c240m4-huu-<version>.iso
   -rwxr-xr-x.  1 root root  16779416 Sep 28 23:55 bios.cap
   ```
6. *Optional.* If you are upgrading software for UCS servers in an existing Ultra M solution stack, then create UCS server node list configuration files for each node type as shown in the following table.

Configuration File Name	File Contents
compute.cfg	A list of the CIMC IP addresses for all of the Compute Nodes.
osd_compute_0.cfg	The CIMC IP address of the primary OSD Compute Node (osd-compute-0).
osd_compute_1.cfg	The CIMC IP address of the second OSD Compute Node (osd-compute-1).
osd_compute_2.cfg	The CIMC IP address of the third OSD Compute Node (osd-compute-2).
controller_0.cfg	The CIMC IP address of the primary Controller Node (controller-0).
controller_1.cfg	The CIMC IP address of the second Controller Node (controller-1).
controller_2.cfg	The CIMC IP address of the third Controller Node (controller-2).

Note Each address must be preceded by a dash and a space ("-"). The following is an example of the required format:

```
- 192.100.0.9
- 192.100.0.10
- 192.100.0.11
- 192.100.0.12
```

Separate configuration files are required for each OSD Compute and Controller Node in order to maintain the integrity of the Ultra M solution stack throughout the upgrade process.

7. Create a configuration file called *hosts.cfg* containing a list of the CIMC IP addresses for all of the servers to be used within the Ultra M solution stack except the OSP-D server/Ultra M Manager Node.

Note Each address must be preceded by a dash and a space (-). The following is an example of the required format:

```
- 192.100.0.9
- 192.100.0.10
- 192.100.0.11
- 192.100.0.12
```

8. Create a configuration file called *ospd.cfg* containing the CIMC IP address of the OSP-D Server/Ultra M Manager Node.

Note The address must be preceded by a dash and a space ("-"). The following is an example of the required format:

```
- 192.300.0.9
```

9. Validate your configuration files by performing a sample test of the script to pull existing firmware versions from all Controller, OSD Compute, and Compute Nodes in your Ultra M solution deployment.

   ./ultram_ucs_utils.py --cfg *"<config_file_name>"* **--login** *<cimc_username> <cimc_user_password>* **--status 'firmwares'**

   The following is an example output for a *hosts.cfg* file with a single Compute Node (192.100.0.7):

```
2017-10-01 10:36:28,189 - Successfully logged out from the server: 192.100.0.7
2017-10-01 10:36:28,190 -
--------------------------------------------------------------------------------------
Server IP      | Component                                    | Version
--------------------------------------------------------------------------------------
192.100.0.7    | bios/fw-boot-loader                        | C240M4.3.0.3c.0.0831170228

               | mgmt/fw-boot-loader                          | 3.0(3e).36
               | mgmt/fw-system                               | 3.0(3e)
               | adaptor-MLOM/mgmt/fw-boot-loader             | 4.1(2d)
               | adaptor-MLOM/mgmt/fw-system                  | 4.1(3a)
               | board/storage-SAS-SLOT-HBA/fw-boot-loader    |
6.30.03.0_4.17.08.00_0xC6130202
               | board/storage-SAS-SLOT-HBA/fw-system         | 4.620.00-7259
               | sas-expander-1/mgmt/fw-system                | 65104100
               | Intel(R) I350 1 Gbps Network Controller      | 0x80000E75-1.810.8
               | Intel X520-DA2 10 Gbps 2 port NIC            | 0x800008A4-1.810.8
               | Intel X520-DA2 10 Gbps 2 port NIC            | 0x800008A4-1.810.8
               | UCS VIC 1227 10Gbps 2 port CNA SFP+          | 4.1(3a)
               | Cisco 12G SAS Modular Raid Controller        | 24.12.1-0203
--------------------------------------------------------------------------------------
```

   If you receive errors when executing the script, ensure that the CIMC username and password are correct. Additionally, verify that all of the IP addresses have been entered properly in the configuration files.

Note It is highly recommended that you save the data reported in the output for later reference and validation after performing the upgrades.

10. Take backups of the various configuration files, logs, and other relevant information using the information and instructions in the *Backing Up Deployment Information* appendix in the *Ultra Services Platform Deployment Automation Guide*.

11. Continue the upgrade process based on your deployment status.

    - Proceed to Shutdown the ESC VMs, on page 119 if you are upgrading software for servers that were previously deployed as part of the Ultra M solution stack.
    - Proceed to Upgrade Firmware on UCS Bare Metal, on page 129 if you are upgrading software for servers that have not yet been deployed as part of the Ultra M solution stack.

Shutdown the ESC VMs

The Cisco Elastic Services Controller (ESC) serves as the VNFM in Ultra M solution deployments. ESC is deployed on a redundant pair of VMs. These VMs must be shut down prior to performing software upgrades on the UCS servers in the solution deployment.

To shut down the ESC VMs:

1. Login to OSP-D and make sure to "su - stack" and "source stackrc".
2. Run Nova list to get the UUIDs of the ESC VMs.

   nova list --fields name,host,status | grep *<vnf_deployment_name>*

   Example output:

```
<--- SNIP --->
| b470cfeb-20c6-4168-99f2-1592502c2057 | vnf1-ESC-ESC-
0                                              | tb5-ultram-osd-compute-2.localdomain |
 ACTIVE |
| 157d7bfb-1152-4138-b85f-79afa96ad97d | vnf1-ESC-ESC-
1                                              | tb5-ultram-osd-compute-1.localdomain |
 ACTIVE |
<--- SNIP --->
```

3. Stop the standby ESC VM.

   nova stop *<standby_vm_uuid>*

4. Stop the active ESC VM.

   nova stop *<active_vm_uuid>*

5. Verify that the VMs have been shutoff.

   nova list --fields name,host,status | grep *<vnf_deployment_name>*

   Look for the entries pertaining to the ESC UUIDs.

   Example output:

```
<--- SNIP --->
| b470cfeb-20c6-4168-99f2-1592502c2057 | vnf1-ESC-ESC-
0                                              | tb5-ultram-osd-compute-2.localdomain |
 SHUTOFF |
| 157d7bfb-1152-4138-b85f-79afa96ad97d | vnf1-ESC-ESC-
1                                              | tb5-ultram-osd-compute-1.localdomain |
 SHUTOFF |
<--- SNIP --->
```

6. Proceed to Upgrade the Compute Node Server Software, on page 120.

Upgrade the Compute Node Server Software

NOTES:

- Ensure that the ESC VMs have been shutdown according to the procedure in Shutdown the ESC VMs, on page 119.
- This procedure assumes that you are already logged in to the AutoIT VM running on the Ultra M Manager Node.
- This procedure requires the *compute.cfg* file created as part of the procedure detailed in Perform Pre-Upgrade Preparation, on page 116.

- It is highly recommended that all Compute Nodes be upgraded using this process during a single maintenance window.

To upgrade the UCS server software on the Compute Nodes:

1. Upgrade the BIOS on the UCS server-based Compute Nodes.

   ./ultram_ucs_utils.py --cfg "compute.cfg" --login *<cimc_username>* *<cimc_user_password>* **--upgrade bios --server** *<rhel_introspection_ip_address>* **--timeout 30 --file /firmwares/bios.cap**

   Example output:

```
2017-09-29 09:15:48,753 - Updating BIOS firmware on all the servers
2017-09-29 09:15:48,753 - Logging on UCS Server: 192.100.0.7
2017-09-29 09:15:48,758 - No session found, creating one on server: 192.100.0.7
2017-09-29 09:15:50,194 - Login successful to server: 192.100.0.7
2017-09-29 09:16:13,269 - 192.100.0.7 => updating | Image Download (5 %), OK
2017-09-29 09:17:26,669 - 192.100.0.7 => updating | Write Host Flash (75 %), OK
2017-09-29 09:18:34,524 - 192.100.0.7 => updating | Write Host Flash (75 %), OK
2017-09-29 09:19:40,892 - 192.100.0.7 => Activating BIOS
2017-09-29 09:19:55,011 -
--------------------------------------------------------------------------
Server IP       | Overall | Updated-on          | Status
--------------------------------------------------------------------------
192.100.0.7     | SUCCESS | NA                  | Status: success, Progress: Done, OK
```

Note The Compute Nodes are automatically powered down after this process leaving only the CIMC interface available.

2. Upgrade the UCS server using the Host Upgrade Utility (HUU).

   ./ultram_ucs_utils.py --cfg "compute.cfg" --login *<cimc_username>* *<cimc_user_password>* **--upgrade huu --server** *<rhel_introspection_ip_address>* **--file /firmwares/***<ucs_huu_iso_filename>*

Note This software is available via the HTTP Apache server.

   If the HUU script times out before completing the upgrade, the process might still be running on the remote hosts. You can periodically check the upgrade process by entering:

   ./ultram_ucs_utils.py --cfg "compute.cfg" --login *<cimc_username>* *<cimc_user_password>* **--status huu-upgrade**

   Example output:

```
--------------------------------------------------------------------------
Server IP       | Overall | Updated-on          | Status
--------------------------------------------------------------------------
192.100.0.7    | SUCCESS | 2017-10-20 07:10:11 | Update Complete  CIMC Completed, SasExpDN
 Completed, I350 Completed, X520 Completed, X520 Completed, 3108AB-8i Completed, UCS VIC
 1227 Completed, BIOS Completed,
--------------------------------------------------------------------------
```

3. Verify that the BIOS firmware and HUU upgrade was successful by checking the post-upgrade versions.

```
./ultram_ucs_utils.py --cfg "compute.cfg" --login <cimc_username>
<cimc_user_password> --status  firmwares
```

4. Set the package-c-state-limit CIMC setting.

```
./ultram_ucs_utils.py --mgmt set-bios --bios-param
biosVfPackageCStateLimit --bios-values vpPackageCStateLimit=C0/C1 --cfg
 compute.cfg --login<cimc_username> <cimc_user_password>
```

5. Verify that the package-c-state-limit CIMC setting has been made.

```
./ultram_ucs_utils.py  --status bios-settings --cfg compute.cfg --login
 <cimc_username> <cimc_user_password>
```

   Look for **PackageCStateLimit** to be set to *C0/C1*.

6. Modify the Grub configuration on each Compute Node.

   a. Log into your first compute (compute-0) and update the grub setting with "processor.max_cstate=0 intel_idle.max_cstate=0".

```
sudo grubby --info=/boot/vmlinuz-`uname -r`
sudo grubby --update-kernel=/boot/vmlinuz-`uname -r`
--args="processor.max_cstate=0
intel_idle.max_cstate=0"
```

   b. Verify that the update was successful.

```
sudo grubby --info=/boot/vmlinuz-`uname -r`
```

      Look for the "processor.max_cstate=0 intel_idle.max_cstate=0" arguments in the output.

   c. Reboot the Compute Nodes.

```
sudo reboot
```

   d. Repeat steps 6.a, on page 122 through 6.c, on page 122 for all other Compute Nodes.

7. Recheck all CIMC and kernel settings.

   a. Log in to the AutoIT VM running on the Ultra M Manager Node.

   b. Verify CIMC settings

```
./ultram_ucs_utils.py  --status bios-settings --cfg compute.cfg
--login<cimc_username> <cimc_user_password>
```

   c. Verify the processor c-state.

```
for ip in `nova list | grep -i compute | awk '{print $12}' | sed
's/ctlplane=//g'`; do ssh heat-admin@$ip 'sudo cat
/sys/module/intel_idle/parameters/max_cstate'; done
for ip in `nova list | grep -i compute | awk '{print $12}' | sed
's/ctlplane=//g'`; do ssh heat-admin@$ip 'sudo cpupower idle-info';
 done
```

8. Proceed to Upgrade the OSD Compute Node Server Software.

Note Other Node types can be upgraded at a later time. If you'll be upgrading them during a later maintenance window, proceed to Restart the UAS and ESC (VNFM) VMs, on page 126.

Upgrade the OSD Compute Node Server Software

NOTES:

- This procedure requires the *osd_compute_0.cfg*, *osd_compute_1.cfg*, and *osd_compute_2.cfg* files created as part of the procedure detailed in Perform Pre-Upgrade Preparation, on page 116.
- It is highly recommended that all OSD Compute Nodes be upgraded using this process during a single maintenance window.

To upgrade the UCS server software on the OSD Compute Nodes:

1. Move the Ceph storage to maintenance mode.
   a. Log on to the lead Controller Node (controller-0).
   b. Move the Ceph storage to maintenance mode.

```
sudo ceph status
sudo ceph osd set noout
sudo ceph osd set norebalance
sudo ceph status
```

2. *Optional.* If they've not already been shut down, shut down both ESC VMs using the instructions in Shutdown the ESC VMs, on page 119.
3. Log on to the AutoIT VM running on the Ultra M Manager Node.
4. Upgrade the BIOS on the initial UCS server-based OSD Compute Node (osd-compute-1).

```
./ultram_ucs_utils.py --cfg "osd_compute_0.cfg" --login <cimc_username>
 <cimc_user_password> --upgrade bios --server <rhel_introspection_ip_address>
--timeout 30 --file /firmwares/bios.cap
```

Example output:

```
2017-09-29 09:15:48,753 - Updating BIOS firmware on all the servers
2017-09-29 09:15:48,753 - Logging on UCS Server: 192.100.0.17
2017-09-29 09:15:48,758 - No session found, creating one on server: 192.100.0.17
2017-09-29 09:15:50,194 - Login successful to server: 192.100.0.17
2017-09-29 09:16:13,269 - 192.100.0.17 => updating | Image Download (5 %), OK
2017-09-29 09:17:26,669 - 192.100.0.17 => updating | Write Host Flash (75 %), OK
2017-09-29 09:18:34,524 - 192.100.0.17 => updating | Write Host Flash (75 %), OK
2017-09-29 09:19:40,892 - 192.100.0.17 => Activating BIOS
2017-09-29 09:19:55,011 -
--------------------------------------------------------------------------
Server IP       | Overall | Updated-on          | Status
--------------------------------------------------------------------------
192.100.0.17     | SUCCESS | NA                  | Status: success, Progress: Done,
OK
```

Note The Compute Nodes are automatically powered down after this process leaving only the CIMC interface available.

5. Upgrade the UCS server using the Host Upgrade Utility (HUU).

   ./ultram_ucs_utils.py --cfg "osd_compute.cfg" --login *<cimc_username> <cimc_user_password>* **--upgrade huu --server** *<rhel_introspection_ip_address>* **--file /firmwares/***<ucs_huu_iso_filename>*

Note This software is available via the HTTP Apache server.

   If the HUU script times out before completing the upgrade, the process might still be running on the remote hosts. You can periodically check the upgrade process by entering:

   ./ultram_ucs_utils.py --cfg "osd_compute.cfg" --login *<cimc_username> <cimc_user_password>* **--status huu-upgrade**

   Example output:

```
--------------------------------------------------------------------------
Server IP       | Overall | Updated-on          | Status
--------------------------------------------------------------------------
192.100.0.17    | SUCCESS | 2017-10-20 07:10:11 | Update Complete  CIMC Completed,
SasExpDN Completed, I350 Completed, X520 Completed, X520 Completed, 3108AB-8i Completed,
 UCS VIC 1227 Completed, BIOS Completed,
--------------------------------------------------------------------------
```

6. Verify that the BIOS firmware and HUU upgrade was successful by checking the post-upgrade versions.

   ./ultram_ucs_utils.py --cfg "osd_compute_0.cfg" --login *<cimc_username> <cimc_user_password>* **--status firmwares**

7. Set the package-c-state-limit CIMC setting.

   ./ultram_ucs_utils.py --mgmt set-bios --bios-param biosVfPackageCStateLimit --bios-values vpPackageCStateLimit=C0/C1 --cfg osd_compute_0.cfg --login *<cimc_username> <cimc_user_password>*

8. Verify that the package-c-state-limit CIMC setting has been made.

   ./ultram_ucs_utils.py --status bios-settings --cfg osd_compute_0.cfg --login *<cimc_username> <cimc_user_password>*

   Look for **PackageCStateLimit** to be set to *C0/C1*.

9. Modify the Grub configuration on the primary OSD Compute Node.

   a. Log on to the OSD Compute (osd-compute-0) and update the grub setting with "processor.max_cstate=0 intel_idle.max_cstate=0".

      sudo grubby --info=/boot/vmlinuz-`uname -r`
      sudo grubby --update-kernel=/boot/vmlinuz-`uname -r`
      --args="processor.max_cstate=0
      intel_idle.max_cstate=0"

b. Verify that the update was successful.

```
sudo grubby --info=/boot/vmlinuz-`uname -r`
```

Look for the "processor.max_cstate=0 intel_idle.max_cstate=0" arguments in the output.

c. Reboot the OSD Compute Nodes.

```
sudo reboot
```

10. Recheck all CIMC and kernel settings.

a. Verify the processor c-state.

```
cat /sys/module/intel_idle/parameters/max_cstate
cpupower idle-info
```

b. Login to AutoIT VM running on the Ultra M Manager Node.

c. Verify CIMC settings.

```
./ultram_ucs_utils.py  --status bios-settings --cfg
osd_compute_0.cfg --login <cimc_username> <cimc_user_password>
```

11. Repeat steps 4, on page 123 through 10, on page 125 on the second OSD Compute Node (osd-compute-1).

Note Be sure to use the *osd_compute_1.cfg* file where needed.

12. Repeat steps 4, on page 123 through 10, on page 125 on the third OSD Compute Node (osd-compute-2).

Note Be sure to use the *osd_compute_2.cfg* file where needed.

13. Check the ironic node-list and restore any hosts that went into maintenance mode true state.

a. Login to OSP-D and make sure to "su - stack" and "source stackrc".

b. Perform the check and any required restorations.

```
ironic node-list
ironic node-set-maintenance $NODE_<node_uuid> off
```

14. Move the Ceph storage out of maintenance mode.

a. Log on to the lead Controller Node (controller-0).

b. Move the Ceph storage to maintenance mode.

```
sudo ceph status
sudo ceph osd unset noout
sudo ceph osd unset norebalance
sudo ceph status
sudo pcs status
```

15. Proceed to Restart the UAS and ESC (VNFM) VMs, on page 126.

Restart the UAS and ESC (VNFM) VMs

Upon performing the UCS server software upgrades, VMs that were previously shutdown must be restarted.

To restart the VMs:

1. Login to OSP-D and make sure to "su - stack" and "source stackrc".
2. Run Nova list to get the UUIDs of the ESC VMs.
3. Start the AutoIT VM.

   nova start *<autoit_vm_uuid>*
4. Start the AutoDeploy VM.

   nova start *<autodeploy_vm_uuid>*
5. Start the standby ESC VM.

   nova start *<standby_vm_uuid>*
6. Start the active ESC VM.

   nova start *<active_vm_uuid>*
7. Verify that the VMs have been restarted and are ACTIVE.

   nova list --fields name,host,status | grep *<vnf_deployment_name>*

   Once ESC is up and running, it triggers the recovery of rest of the VMs (AutoVNF, UEMs, CFs and SFs).
8. Login to each of the VMs and verify that they are operational.

Upgrade the Controller Node Server Software

NOTES:

- This procedure requires the *controller_0.cfg*, *controller_1.cfg*, and *controller_2.cfg* files created as part of the procedure detailed in Perform Pre-Upgrade Preparation, on page 116.
- It is highly recommended that all Controller Nodes be upgraded using this process during a single maintenance window.

To upgrade the UCS server software on the Controller Nodes:

1. Check the Controller Node status and move the Pacemaker Cluster Stack (PCS) to maintenance mode.
   a. Login to the primary Controller Node (controller-0) from the OSP-D Server.
   b. Check the state of the Controller Node Pacemaker Cluster Stack (PCS).

      sudo pcs status

Note Resolve any issues prior to proceeding to the next step.

c. Place the PCS cluster on the Controller Node into standby mode.

sudo pcs cluster standby *<controller_name>*

d. Recheck the Controller Node status again and make sure that the Controller Node is in standby mode for the PCS cluster.

sudo pcs status

2. Log on to the AutoIT VM running on the Ultra M Manager Node.

3. Upgrade the BIOS on the primary UCS server-based Controller Node (controller-0).

./ultram_ucs_utils.py --cfg "controller_0.cfg" --login *<cimc_username>* *<cimc_user_password>* **--upgrade bios --server** *<rhel_introspection_ip_address>* **--timeout 30 --file /firmwares/bios.cap**

Example output:

```
2017-09-29 09:15:48,753 - Updating BIOS firmware on all the servers
2017-09-29 09:15:48,753 - Logging on UCS Server: 192.100.2.7
2017-09-29 09:15:48,758 - No session found, creating one on server: 192.100.2.7
2017-09-29 09:15:50,194 - Login successful to server: 192.100.2.7
2017-09-29 09:16:13,269 - 192.100.2.7 => updating | Image Download (5 %), OK
2017-09-29 09:17:26,669 - 192.100.2.7 => updating | Write Host Flash (75 %), OK
2017-09-29 09:18:34,524 - 192.100.2.7 => updating | Write Host Flash (75 %), OK
2017-09-29 09:19:40,892 - 192.100.2.7 => Activating BIOS
2017-09-29 09:19:55,011 -
--------------------------------------------------------------------------
Server IP       | Overall | Updated-on          | Status
--------------------------------------------------------------------------
192.100.2.7     | SUCCESS | NA                  | Status: success, Progress: Done, OK
```

Note The Compute Nodes are automatically powered down after this process leaving only the CIMC interface available.

4. Upgrade the UCS server using the Host Upgrade Utility (HUU).

./ultram_ucs_utils.py --cfg "controller_0.cfg" --login *<cimc_username>* *<cimc_user_password>* **--upgrade huu --server** *<rhel_introspection_ip_address>* **--file /firmwares/***<ucs_huu_iso_filename>*

Note This software is available via the HTTP Apache server.

If the HUU script times out before completing the upgrade, the process might still be running on the remote hosts. You can periodically check the upgrade process by entering:

./ultram_ucs_utils.py --cfg "controller_0.cfg" --login *<cimc_username>* *<cimc_user_password>* **--status huu-upgrade**

Example output:

```
--------------------------------------------------------------------------
Server IP       | Overall | Updated-on          | Status
--------------------------------------------------------------------------
```

```
192.100.2.7    | SUCCESS | 2017-10-20 07:10:11 | Update Complete  CIMC Completed,
SasExpDN Completed, I350 Completed, X520 Completed, X520 Completed, 3108AB-8i Completed,
 UCS VIC 1227 Completed, BIOS Completed,
--------------------------------------------------------------------------------
```

5. Verify that the BIOS firmware and HUU upgrade was successful by checking the post-upgrade versions.

   ./ultram_ucs_utils.py --cfg "controller_0.cfg" --login *<cimc_username> <cimc_user_password>* **--status firmwares**

6. Set the package-c-state-limit CIMC setting.

   ./ultram_ucs_utils.py --mgmt set-bios --bios-param biosVfPackageCStateLimit --bios-values vpPackageCStateLimit=C0/C1 --cfg controller_0.cfg --login *<cimc_username> <cimc_user_password>*

7. Verify that the package-c-state-limit CIMC setting has been made.

   ./ultram_ucs_utils.py --status bios-settings --cfg controller_0.cfg --login *<cimc_username> <cimc_user_password>*

   Look for **PackageCStateLimit** to be set to *C0/C1*.

8. Modify the Grub configuration on the primary OSD Compute Node.

   a. Log on to the OSD Compute (osd-compute-0) and update the grub setting with "processor.max_cstate=0 intel_idle.max_cstate=0".

      ```
      sudo grubby --info=/boot/vmlinuz-`uname -r`
      sudo grubby --update-kernel=/boot/vmlinuz-`uname -r`
      --args="processor.max_cstate=0
      intel_idle.max_cstate=0"
      ```

   b. Verify that the update was successful.

      ```
      sudo grubby --info=/boot/vmlinuz-`uname -r`
      ```

      Look for the "processor.max_cstate=0 intel_idle.max_cstate=0" arguments in the output.

   c. Reboot the OSD Compute Nodes.

      ```
      sudo reboot
      ```

9. Recheck all CIMC and kernel settings.

   a. Verify the processor c-state.

      ```
      cat /sys/module/intel_idle/parameters/max_cstate
      cpupower idle-info
      ```

   b. Login to AutoIT VM running on the Ultra M Manager Node.

   c. Verify CIMC settings.

      ./ultram_ucs_utils.py --status bios-settings --cfg controller_0.cfg --login *<cimc_username> <cimc_user_password>*

10. Check the ironic node-list and restore the Controller Node if it went into maintenance mode true state.

    a. Login to OSP-D and make sure to "su - stack" and "source stackrc".

    b. Perform the check and any required restorations.

```
ironic node-list
ironic node-set-maintenance $NODE_<node_uuid> off
```

11. Take the Controller Node out of the PCS standby state.

    sudo pcs cluster unstandby *<controller-0-id>*

12. Wait 5 to 10 minutes and check the state of the PCS cluster to verify that the Controller Node is ONLINE and all services are in good state.

    sudo pcs status

13. Repeat steps 3, on page 127 through 11, on page 129 on the second Controller Node (controller-1).

Note Be sure to use the *controller_1.cfg* file where needed.

14. Repeat steps 3, on page 127 through 11, on page 129 on the third Controller Node (controller-2).

Note Be sure to use the *controller_2.cfg* file where needed.

15. Proceed to Upgrade Firmware on the OSP-D Server/Ultra M Manager Node, on page 134.

Upgrade Firmware on UCS Bare Metal

NOTES:

- This procedure assumes that the UCS servers receiving the software (firmware) upgrade have not previously been deployed as part of an Ultra M solution stack.
- The instructions in this section pertain to all servers to be used as part of an Ultra M solution stack except the OSP-D Server/Ultra M Manager Node.
- This procedure requires the *hosts.cfg* file created as part of the procedure detailed in Perform Pre-Upgrade Preparation, on page 116.

To upgrade the software on the UCS servers:

1. Log on to the AutoIT VM running on the Ultra M Manager Node.
2. Upgrade the BIOS on the UCS servers.

   ./ultram_ucs_utils.py --cfg "hosts.cfg" --login *<cimc_username>* *<cimc_user_password>* **--upgrade bios --server** *<rhel_introspection_ip_address>* **--timeout 30 --file /firmwares/bios.cap**

   Example output:

```
2018-03-26 09:15:48,753 - Updating BIOS firmware on all the servers
2018-03-26 09:15:48,753 - Logging on UCS Server: 192.100.1.2
2018-03-26 09:15:48,758 - No session found, creating one on server: 192.100.1.2
2018-03-26 09:15:50,194 - Login successful to server: 192.100.1.2
2018-03-26 09:16:13,269 - 192.100.1.2 => updating | Image Download (5 %), OK
```

Using the UCS Utilities Within the Ultra M Manager

```
2018-03-26 09:17:26,669 - 192.100.1.2 => updating | Write Host Flash (75 %), OK
2018-03-26 09:18:34,524 - 192.100.1.2 => updating | Write Host Flash (75 %), OK
2018-03-26 09:19:40,892 - 192.100.1.2 => Activating BIOS
2018-03-26 09:19:55,011 -
--------------------------------------------------------------------
Server IP       | Overall | Updated-on          | Status
--------------------------------------------------------------------
192.100.1.2     | SUCCESS | NA                  | Status: success, Progress: Done, OK
```

Note The Compute Nodes are automatically powered down after this process leaving only the CIMC interface available.

3. Upgrade the UCS server using the Host Upgrade Utility (HUU).

   ./ultram_ucs_utils.py --cfg "hosts.cfg" --login *<cimc_username>* *<cimc_user_password>* **--upgrade huu --server** *<rhel_introspection_ip_address>* **--file /firmwares/***<ucs_huu_iso_filename>*

Note This software is available via the HTTP Apache server.

   If the HUU script times out before completing the upgrade, the process might still be running on the remote hosts. You can periodically check the upgrade process by entering:

   ./ultram_ucs_utils.py --cfg "hosts.cfg" --login *<cimc_username>* *<cimc_user_password>* **--status huu-upgrade**

   Example output:

```
--------------------------------------------------------------------
Server IP       | Overall | Updated-on          | Status
--------------------------------------------------------------------
192.100.1.2     | SUCCESS | 2018-03-19 08:54:06 | Update Complete CIMC Completed, I350
Completed, I350-PCI Completed, I350-PCI Completed, 9271-8i Completed, BIOS Completed,
--------------------------------------------------------------------
```

4. Verify that the BIOS firmware and HUU upgrade was successful by checking the post-upgrade versions.

   ./ultram_ucs_utils.py --cfg "hosts.cfg" --login *<cimc_username>* *<cimc_user_password>* **--status firmwares**

5. Recheck all CIMC and BIOS settings.

   a. Log in to the AutoIT VM running on the Ultra M Manager Node.

   b. Verify CIMC settings.

      ./ultram_ucs_utils.py --status bios-settings --cfg hosts.cfg --login *<cimc_username>* *<cimc_user_password>*

6. Verify that the BIOS and HUU upgrades were successful by checking the posting upgrade versions.

   ./ultram_ucs_utils.py --cfg *"<config_file_name>"* **--login** *<cimc_username>* *<cimc_user_password>* **--status 'firmwares'**

   Compare the new output of the command for the hosts.cfg file against one previously collected in Perform Pre-Upgrade Preparation, on page 116.

7. Verify the UCS server status.

   ./ultram_ucs_utils.py --cfg *"<config_file_name>"* **--login** *<cimc_username> <cimc_user_password>* **--status server**

   The following is an example output for a *hosts.cfg* file with a single Compute Node (192.100.0.7):

```
2018-03-29 06:42:29,516 -
------------------------------------------------------------
Server IP         | Status
------------------------------------------------------------
192.100.0.7     | dn: sys/rack-unit-1
                      | adminPower: policy
                      | availableMemory: 262144
                      | model: UCSC-C240-M4SX
                      | memorySpeed: 2400
                      | name: UCS C240 M4SX
                      | numOfAdaptors: 1
                      | numOfCores: 28
                       | numOfCoresEnabled: 28
                          | numOfCpus: 2
                      | numOfEthHostIfs: 2
                      | numOfFcHostIfs: 2
                      | numOfThreads: 56
                      | operPower: on
                      | originalUuid: 03AFB6F7-4C50-4272-8B37-AD582A7ADA02
                      | presence: equipped
                      | serverId: 1
                      | serial: FCH2103V1LA
                      | totalMemory: 262144
                      | usrLbl:
                      | uuid: 03AFB6F7-4C50-4272-8B37-AD582A7ADA02
                      | vendor: Cisco Systems Inc
                      | cimcResetReason: ac-cycle
                      | adaptorSecureUpdate: Enabled
------------------------------------------------------------
```

8. Verify the BIOS settings.

   ./ultram_ucs_utils.py --cfg *"<config_file_name>"* **--login** *<cimc_username> <cimc_user_password>* **--status bios-settings**

   The following is an example output for a *hosts.cfg* file with a single Compute Node (192.100.0.7):

```
2018-03-29 06:47:56,175 -
------------------------------------------------------------------------------
Server IP       | BIOS Settings
------------------------------------------------------------------------------
192.100.0.7     | biosVfHWPMEnable
                |  vpHWPMEnable: Disabled
                | biosVfLegacyUSBSupport
                |  vpLegacyUSBSupport: enabled
                | biosVfPciRomClp
                |  vpPciRomClp: Disabled
                | biosVfSelectMemoryRASConfiguration
                |  vpSelectMemoryRASConfiguration: maximum-performance
                | biosVfExtendedAPIC
                |  vpExtendedAPIC: XAPIC
                | biosVfOSBootWatchdogTimerPolicy
                |  vpOSBootWatchdogTimerPolicy: power-off
                | biosVfCoreMultiProcessing
                |  vpCoreMultiProcessing: all
                | biosVfQPIConfig
                |  vpQPILinkFrequency: auto
                | biosVfOutOfBandMgmtPort
```

Using the UCS Utilities Within the Ultra M Manager

```
|  vpOutOfBandMgmtPort: Disabled
| biosVfVgaPriority
|  vpVgaPriority: Onboard
| biosVfMemoryMappedIOAbove4GB
|  vpMemoryMappedIOAbove4GB: enabled
| biosVfEnhancedIntelSpeedStepTech
|  vpEnhancedIntelSpeedStepTech: enabled
| biosVfCmciEnable
|  vpCmciEnable: Enabled
| biosVfAutonumousCstateEnable
|  vpAutonumousCstateEnable: Disabled
| biosVfOSBootWatchdogTimer
|  vpOSBootWatchdogTimer: disabled
| biosVfAdjacentCacheLinePrefetch
|  vpAdjacentCacheLinePrefetch: enabled
| biosVfPCISlotOptionROMEnable
|  vpSlot1State: Disabled
|  vpSlot2State: Disabled
|  vpSlot3State: Disabled
|  vpSlot4State: Disabled
|  vpSlot5State: Disabled
|  vpSlot6State: Disabled
|  vpSlotMLOMState: Enabled
|  vpSlotHBAState: Enabled
|  vpSlotHBALinkSpeed: GEN3
|  vpSlotN1State: Disabled
|  vpSlotN2State: Disabled
|  vpSlotFLOMLinkSpeed: GEN3
|  vpSlotRiser1Slot1LinkSpeed: GEN3
|  vpSlotRiser1Slot2LinkSpeed: GEN3
|  vpSlotRiser1Slot3LinkSpeed: GEN3
|  vpSlotSSDSlot1LinkSpeed: GEN3
|  vpSlotSSDSlot2LinkSpeed: GEN3
|  vpSlotRiser2Slot4LinkSpeed: GEN3
|  vpSlotRiser2Slot5LinkSpeed: GEN3
|  vpSlotRiser2Slot6LinkSpeed: GEN3
| biosVfProcessorC3Report
|  vpProcessorC3Report: disabled
| biosVfPCIeSSDHotPlugSupport
|  vpPCIeSSDHotPlugSupport: Disabled
| biosVfExecuteDisableBit
|  vpExecuteDisableBit: enabled
| biosVfCPUEnergyPerformance
|  vpCPUEnergyPerformance: balanced-performance
| biosVfAltitude
|  vpAltitude: 300-m
| biosVfSrIov
|  vpSrIov: enabled
| biosVfIntelVTForDirectedIO
|  vpIntelVTDATSSupport: enabled
|  vpIntelVTDCoherencySupport: disabled
|  vpIntelVTDInterruptRemapping: enabled
|  vpIntelVTDPassThroughDMASupport: disabled
|  vpIntelVTForDirectedIO: enabled
| biosVfCPUPerformance
|  vpCPUPerformance: enterprise
| biosVfPchUsb30Mode
|  vpPchUsb30Mode: Disabled
| biosVfTPMSupport
|  vpTPMSupport: enabled
| biosVfIntelHyperThreadingTech
|  vpIntelHyperThreadingTech: disabled
| biosVfIntelTurboBoostTech
|  vpIntelTurboBoostTech: enabled
```

```
| biosVfUSBEmulation
|  vpUSBEmul6064: enabled
| biosVfMemoryInterleave
|  vpChannelInterLeave: auto
|  vpRankInterLeave: auto
| biosVfConsoleRedirection
|  vpBaudRate: 115200
|  vpConsoleRedirection: disabled
|  vpFlowControl: none
|  vpTerminalType: vt100
|  vpPuttyKeyPad: ESCN
|  vpRedirectionAfterPOST: Always Enable
| biosVfQpiSnoopMode
|  vpQpiSnoopMode: auto
| biosVfPStateCoordType
|  vpPStateCoordType: HW ALL
| biosVfProcessorC6Report
|  vpProcessorC6Report: enabled
| biosVfPCIOptionROMs
|  vpPCIOptionROMs: Enabled
| biosVfDCUPrefetch
|  vpStreamerPrefetch: enabled
|  vpIPPrefetch: enabled
| biosVfFRB2Enable
|  vpFRB2Enable: enabled
| biosVfLOMPortOptionROM
|  vpLOMPortsAllState: Enabled
|  vpLOMPort0State: Enabled
|  vpLOMPort1State: Enabled
| biosVfPatrolScrub
|  vpPatrolScrub: enabled
| biosVfNUMAOptimized
|  vpNUMAOptimized: enabled
| biosVfCPUPowerManagement
|  vpCPUPowerManagement: performance
| biosVfDemandScrub
|  vpDemandScrub: enabled
| biosVfDirectCacheAccess
|  vpDirectCacheAccess: auto
| biosVfPackageCStateLimit
|  vpPackageCStateLimit: C6 Retention
| biosVfProcessorC1E
|  vpProcessorC1E: enabled
| biosVfUSBPortsConfig
|  vpAllUsbDevices: enabled
|  vpUsbPortRear: enabled
|  vpUsbPortFront: enabled
|  vpUsbPortInternal: enabled
|  vpUsbPortKVM: enabled
|  vpUsbPortVMedia: enabled
| biosVfSataModeSelect
|  vpSataModeSelect: AHCI
| biosVfOSBootWatchdogTimerTimeout
|  vpOSBootWatchdogTimerTimeout: 10-minutes
| biosVfWorkLoadConfig
|  vpWorkLoadConfig: Balanced
| biosVfCDNEnable
|  vpCDNEnable: Disabled
| biosVfIntelVirtualizationTechnology
|  vpIntelVirtualizationTechnology: enabled
| biosVfHardwarePrefetch
|  vpHardwarePrefetch: enabled
| biosVfPwrPerfTuning
```

```
                    |  vpPwrPerfTuning: os
-----------------------------------------------------------------------------
```

Upgrade Firmware on the OSP-D Server/Ultra M Manager Node

1. Open your web browser.
2. Enter the CIMC address of the OSP-D Server/Ultra M Manager Node in the URL field.
3. Login to the CIMC using the configured user credentials.
4. Click **Launch KVM Console**.
5. Click **Virtual Media**.
6. Click **Add Image** and select the HUU ISO file pertaining to the version you wish to upgrade to.
7. Select the ISO that you have added in the **Mapped** column of the **Client View**. Wait for the selected ISO to appear as a mapped device.
8. Boot the server and press F6 when prompted to open the **Boot Menu**.
9. Select the desired ISO.
10. Select **Cisco vKVM-Mapped vDVD1.22**, and press **Enter**. The server boots from the selected device.
11. Follow the onscreen instructions to update the desired software and reboot the server. Proceed to the next step once the server has rebooted.
12. Log on to the AutoIT VM running on the Ultra M Manager Node.
13. Set the package-c-state-limit CIMC setting.

    ```
    ./ultram_ucs_utils.py --mgmt set-bios --bios-param
    biosVfPackageCStateLimit --bios-values
    vpPackageCStateLimit=C0/C1 --cfg ospd.cfg --login <cimc_username>
    <cimc_user_password>
    ```

14. Verify that the package-c-state-limit CIMC setting has been made.

    ```
    ./ultram_ucs_utils.py --status bios-settings --cfg controller.cfg
    --login <cimc_username> <cimc_user_password>
    ```

    Look for **PackageCStateLimit** to be set to C0/C1.

15. Update the grub setting with "processor.max_cstate=0 intel_idle.max_cstate=0".

    ```
    sudo grubby --info=/boot/vmlinuz-`uname -r`
    sudo grubby --update-kernel=/boot/vmlinuz-`uname -r`
    --args="processor.max_cstate=0 intel_idle.max_cstate=0"
    ```

16. Verify that the update was successful.

    ```
    sudo grubby --info=/boot/vmlinuz-`uname -r`
    ```

    Look for the "processor.max_cstate=0 intel_idle.max_cstate=0" arguments in the output.

17. Reboot the server.

```
sudo reboot
```

18. Recheck all CIMC and kernel settings upon reboot.
    a. Verify CIMC settings

    ```
    ./ultram_ucs_utils.py --status bios-settings --cfg ospd.cfg --login
     <cimc_username> <cimc_user_password>
    ```

    b. Verify the processor c-state.

    ```
    cat /sys/module/intel_idle/parameters/max_cstate
    cpupower idle-info
    ```

Controlling UCS BIOS Parameters Using *ultram_ucs_utils.py* Script

The ultram_ucs_utils.py script can be used to modify and verify parameters within the UCS server BIOS. This script is in the */opt/cisco/usp/ultram-manager* directory.

Important Refer to the UCS server documentation BIOS documentation for information on parameters and their respective values.

To configure UCS server BIOS parameters:

1. Log on to the AutoIT VM running on the Ultra M Manager Node.
2. Modify the desired BIOS parameters.

   ```
   ./ultram_ucs_utils.py --cfg "config_file_name" --login cimc_username
   cimc_user_password --mgmt 'set-bios' --bios-param bios_paramname --bios-values
    bios_value1 bios_value2
   ```

   Example:

   ```
   ./ultram_ucs_utils.py --cfg cmp_17 --login admin abcabc --mgmt 'set-bios --bios-param
   biosVfUSBPortsConfig --bios-values vpAllUsbDevices=Disabled vpUsbPortRear=Disabled
   ```

   Example output:

   ```
   2017-10-06 19:48:39,241 - Set BIOS Parameters
   2017-10-06 19:48:39,241 - Logging on UCS Server: 192.100.0.25
   2017-10-06 19:48:39,243 - No session found, creating one on server: 192.100.0.25
   2017-10-06 19:48:40,711 - Login successful to server: 192.100.0.25
   2017-10-06 19:48:52,709 - Logging out from the server: 192.100.0.25
   2017-10-06 19:48:53,893 - Successfully logged out from the server: 192.100.0.25
   ```

3. Verify that your settings have been incorporated.

   ```
   ./ultram_ucs_utils.py --cfg "config_file_name" --login cimc_username
   cimc_user_password -- status bios-settings
   ```

   Example output:

   ```
   ./ultram_ucs_utils.py --cfg cmp_17 --login admin abcabc --status
   bios-settings
   ```

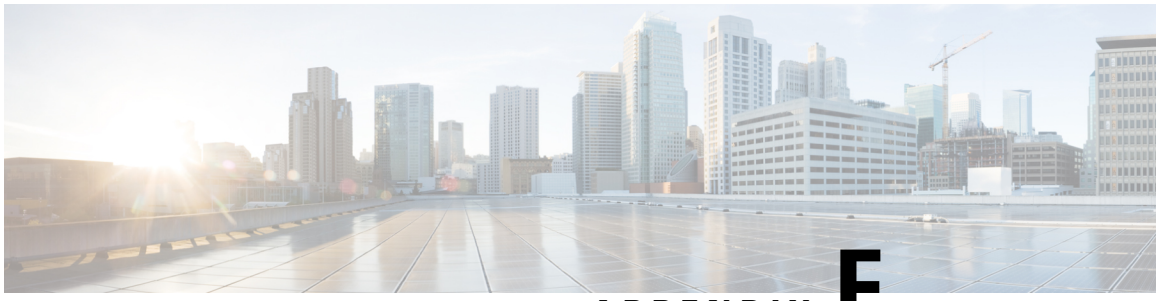
Using the UCS Utilities Within the Ultra M Manager

```
2017-10-06 19:49:12,366 - Getting status information from all the servers
2017-10-06 19:49:12,366 - Logging on UCS Server: 192.100.0.25
2017-10-06 19:49:12,370 - No session found, creating one on server: 192.100.0.25
2017-10-06 19:49:13,752 - Login successful to server: 192.100.0.25
2017-10-06 19:49:19,739 - Logging out from the server: 192.100.0.25
2017-10-06 19:49:20,922 - Successfully logged out from the server: 192.100.0.25
2017-10-06 19:49:20,922 -
--------------------------------------------------------------------------
Server IP      | BIOS Settings
--------------------------------------------------------------------------
192.100.0.25   | biosVfHWPMEnable
               |  vpHWPMEnable: Disabled
               | biosVfLegacyUSBSupport
               |  vpLegacyUSBSupport: enabled
               | biosVfPciRomClp
               |  vpPciRomClp: Disabled
               | biosVfSelectMemoryRASConfiguration
               |  vpSelectMemoryRASConfiguration: maximum-performance
               | biosVfExtendedAPIC
               |  vpExtendedAPIC: XAPIC
               | biosVfOSBootWatchdogTimerPolicy
               |  vpOSBootWatchdogTimerPolicy: power-off
               | biosVfCoreMultiProcessing
               |  vpCoreMultiProcessing: all
               | biosVfQPIConfig
               |  vpQPILinkFrequency: auto
               | biosVfOutOfBandMgmtPort
               |  vpOutOfBandMgmtPort: Disabled
               | biosVfVgaPriority
               |  vpVgaPriority: Onboard
               | biosVfMemoryMappedIOAbove4GB
               |  vpMemoryMappedIOAbove4GB: enabled
               | biosVfEnhancedIntelSpeedStepTech
               |  vpEnhancedIntelSpeedStepTech: enabled
               | biosVfCmciEnable
               |  vpCmciEnable: Enabled
               | biosVfAutonumousCstateEnable
               |  vpAutonumousCstateEnable: Disabled
               | biosVfOSBootWatchdogTimer
               |  vpOSBootWatchdogTimer: disabled
               | biosVfAdjacentCacheLinePrefetch
               |  vpAdjacentCacheLinePrefetch: enabled
               | biosVfPCISlotOptionROMEnable
               |  vpSlot1State: Disabled
               |  vpSlot2State: Disabled
               |  vpSlot3State: Disabled
               |  vpSlot4State: Disabled
               |  vpSlot5State: Disabled
               |  vpSlot6State: Disabled
               |  vpSlotMLOMState: Enabled
               |  vpSlotHBAState: Enabled
               |  vpSlotHBALinkSpeed: GEN3
               |  vpSlotN1State: Disabled
               |  vpSlotN2State: Disabled
               |  vpSlotFLOMLinkSpeed: GEN3
               |  vpSlotRiser1Slot1LinkSpeed: GEN3
               |  vpSlotRiser1Slot2LinkSpeed: GEN3
               |  vpSlotRiser1Slot3LinkSpeed: GEN3
               |  vpSlotSSDSlot1LinkSpeed: GEN3
               |  vpSlotSSDSlot2LinkSpeed: GEN3
               |  vpSlotRiser2Slot4LinkSpeed: GEN3
               |  vpSlotRiser2Slot5LinkSpeed: GEN3
               |  vpSlotRiser2Slot6LinkSpeed: GEN3
               | biosVfProcessorC3Report
```

```
|  vpProcessorC3Report: disabled
| biosVfPCIeSSDHotPlugSupport
|  vpPCIeSSDHotPlugSupport: Disabled
| biosVfExecuteDisableBit
|  vpExecuteDisableBit: enabled
| biosVfCPUEnergyPerformance
|  vpCPUEnergyPerformance: balanced-performance
| biosVfAltitude
|  vpAltitude: 300-m
| biosVfSrIov
|  vpSrIov: enabled
| biosVfIntelVTForDirectedIO
|  vpIntelVTDATSSupport: enabled
|  vpIntelVTDCoherencySupport: disabled
|  vpIntelVTDInterruptRemapping: enabled
|  vpIntelVTDPassThroughDMASupport: disabled
|  vpIntelVTForDirectedIO: enabled
| biosVfCPUPerformance
|  vpCPUPerformance: enterprise
| biosVfPchUsb30Mode
|  vpPchUsb30Mode: Disabled
| biosVfTPMSupport
|  vpTPMSupport: enabled
| biosVfIntelHyperThreadingTech
|  vpIntelHyperThreadingTech: disabled
| biosVfIntelTurboBoostTech
|  vpIntelTurboBoostTech: enabled
| biosVfUSBEmulation
|  vpUSBEmul6064: enabled
| biosVfMemoryInterleave
|  vpChannelInterLeave: auto
|  vpRankInterLeave: auto
| biosVfConsoleRedirection
|  vpBaudRate: 115200
|  vpConsoleRedirection: disabled
|  vpFlowControl: none
|  vpTerminalType: vt100
|  vpPuttyKeyPad: ESCN
|  vpRedirectionAfterPOST: Always Enable
| biosVfQpiSnoopMode
|  vpQpiSnoopMode: auto
| biosVfPStateCoordType
|  vpPStateCoordType: HW ALL
| biosVfProcessorC6Report
|  vpProcessorC6Report: enabled
| biosVfPCIOptionROMs
|  vpPCIOptionROMs: Enabled
| biosVfDCUPrefetch
|  vpStreamerPrefetch: enabled
|  vpIPPrefetch: enabled
| biosVfFRB2Enable
|  vpFRB2Enable: enabled
| biosVfLOMPortOptionROM
|  vpLOMPortsAllState: Enabled
|  vpLOMPort0State: Enabled
|  vpLOMPort1State: Enabled
| biosVfPatrolScrub
|  vpPatrolScrub: enabled
| biosVfNUMAOptimized
|  vpNUMAOptimized: enabled
| biosVfCPUPowerManagement
|  vpCPUPowerManagement: performance
| biosVfDemandScrub
|  vpDemandScrub: enabled
```

Using the UCS Utilities Within the Ultra M Manager

```
                    | biosVfDirectCacheAccess
                    |  vpDirectCacheAccess: auto
                    | biosVfPackageCStateLimit
                    |  vpPackageCStateLimit: C6 Retention
                    | biosVfProcessorC1E
                    |  vpProcessorC1E: enabled
                    | biosVfUSBPortsConfig
                    |  vpAllUsbDevices: disabled
                    |  vpUsbPortRear: disabled
                    |  vpUsbPortFront: enabled
                    |  vpUsbPortInternal: enabled
                    |  vpUsbPortKVM: enabled
                    |  vpUsbPortVMedia: enabled
                    | biosVfSataModeSelect
                    |  vpSataModeSelect: AHCI
                    | biosVfOSBootWatchdogTimerTimeout
                    |  vpOSBootWatchdogTimerTimeout: 10-minutes
                    | biosVfWorkLoadConfig
                    |  vpWorkLoadConfig: Balanced
                    | biosVfCDNEnable
                    |  vpCDNEnable: Disabled
                    | biosVfIntelVirtualizationTechnology
                    |  vpIntelVirtualizationTechnology: enabled
                    | biosVfHardwarePrefetch
                    |  vpHardwarePrefetch: enabled
                    | biosVfPwrPerfTuning
                    |  vpPwrPerfTuning: os
--------------------------------------------------------------------------------
```



APPENDIX F

ultram_ucs_utils.py Help

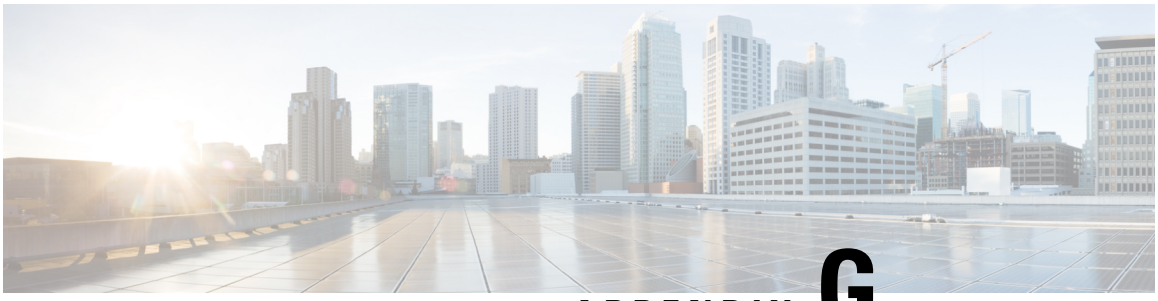
Enter the following command to display help for the UCS utilities available through the Ultra M Manager:

./ultram_ucs_utils.py h

```
usage: ultram_ucs_utils.py [-h] --cfg CFG --login UC_LOGIN UC_LOGIN
                           (--upgrade  | --mgmt  | --status  | --undercloud UC_RC)
                           [--mode] [--serial-delay SERIAL_DELAY]
                           [--server SERVER] [--file FILE]
                           [--protocol {http,https,tftp,sftp,ftp,scp}]
                           [--access ACCESS ACCESS] [--secure-boot]
                           [--update-type {immediate,delayed}] [--reboot]
                           [--timeout TIMEOUT] [--verify] [--stop-on-error]
                           [--bios-param BIOS_PARAM]
                           [--bios-values BIOS_VALUES [BIOS_VALUES ...]]

optional arguments:
  -h, --help            show this help message and exit
  --cfg CFG             Configuration file to read servers
  --login UC_LOGIN UC_LOGIN
                        Common Login UserName / Password to authenticate UCS servers
  --vim-type            Type of Virtual Infrastructure Manager(OSP10 or CVIM or VCD)
  --upgrade             Firmware upgrade, choose one from:
                          'bios': Upgrade BIOS firmware version
                          'cimc': Upgrade CIMC firmware version
                          'huu' : Upgrade All Firmwares via HUU based on ISO
  --mgmt                Server Management Tasks, choose one from:
                          'power-up'      :  Power on the server immediately
                          'power-down'    :  Power down the server (non-graceful)
                          'soft-shut-down':  Shutdown the server gracefully
                          'power-cycle'   :  Power Cycle the server immediately
                          'hard-reset'    :  Hard Reset the server
                          'cimc-reset'    :  Reboot CIMC
                          'cmos-reset'    :  Reset CMOS
                          'set-bios'      :  Set BIOS Parameter
  --status              Firmware Update Status:
                          'bios-upgrade'  :  Last BIOS upgrade status
                          'cimc-upgrade'  :  Last CIMC upgrade status
                          'huu-upgrade'   :  Last ISO upgrade via Host Upgrade Utilties
                          'firmwares'     :  List Current set of running firmware versions
                          'server'        :  List Server status
                          'bios-settings' :  List BIOS Settings
  --undercloud UC_RC    Get the list of servers from undercloud
  --mode                Execute action in serial/parallel
  --serial-delay SERIAL_DELAY
                        Delay (seconds) in executing firmware upgrades on node in case of
serial mode
```

ultram_ucs_utils.py Help

```
Firmware Upgrade Options::
  --server SERVER       Server IP hosting the file via selected protocol
  --file FILE           Firmware file path for UCS server to access from file server
  --protocol {http,https,tftp,sftp,ftp,scp}
                        Protocol to get the firmware file on UCS server
  --access ACCESS ACCESS
                        UserName / Password to access the file from remote server using
https,sftp,ftp,scp
  --secure-boot         Use CIMC Secure-Boot.
  --update-type {immediate,delayed}
                        Update type whether to send delayed update to server or immediate
  --reboot              Reboot CIMC before performing update
  --timeout TIMEOUT     Update timeout in mins should be more than 30 min and less than 200
 min
  --verify              Use this option to verify update after reboot
  --stop-on-error       Stop the firmware update once an error is encountered

BIOS Parameters configuratioon:
  --bios-param BIOS_PARAM
                        BIOS Paramater Name to be set
  --bios-values BIOS_VALUES [BIOS_VALUES ...]
                       BIOS Paramater values in terms of key=value pair separated by space
```



APPENDIX G

Sample FMD Configuration File

The sample fault management descriptor (FMD) configuration file provides AutoIT with the necessary information for configuring health monitoring through a network service descriptor (NSD) configuration file activated through AutoDeploy.

Caution This is only a sample configuration file provided solely for your reference. You must create and modify your own configuration file according to the specific needs of your deployment.

```
nsd blr-bgl15
 fmd               fmd-blr
 vim-orch-identity vim-orch-id-blr
!
vim vim-orch-id-blr
 api-version v2
 auth-url    http://192.168.10.1:5000/v2.0
 user        openstack
 tenant      admin
!
secure-token openstack
 user     $8$UJs47fsQ1i1T71osYwO+wFRj79O2bUIQtGI+Jv+Xc6Q=
 password $8$1tfw7vpnJKztjx6PneESzQJ1e3Qz5KhMnVt2y9ArqRE=
!
secure-token os-login
 user     $8$r8aZRlWoN3JN8czUofGs2eXMD3ZDOUeUGVCsZDu87ew=
 password $8$6Zj5yVjOxIeKXFqMPl3EsCAU80wOS1FvI6QfNVp1pjs=
!
secure-token autoit-nfvi-admin
 user     $8$b0omJaBlEWe5uP1wq3+thoSSFTNq65LmhMui0hMqfhM=
 password $8$j4HT9/V5Ww8xII36iOY+DS3GD7UBYUrGVfEB4i0RbiY=
!
secure-token autoit-nfvi2-admin
 user     $8$U+Bym8O56+oowjEXHXgC/wkX8bgkoLCFreSMeo9uAz4=
 password $8$15V9FW0PDe2fgBIxtLE98/2vy1Z2Ns5vIRYkyDyeSG4=
!
secure-token nfvi-login
 user     $8$Rh+7C1BvgmCE2O6tZVqXPz8RtjFYmGb8xd5XY4ZV/nw=
 password $8$lM+6HCkhjSM9AX8D++35F/CQfI5/0OOWLFYRCB2y5D8=
!
secure-token login
 user     $8$CdqKq6es3f2yVtFaOy6ZB1VxFR7BJTsXh0ojZqg/xW4=
 password $8$7/ag39+Qw8XapzY6ijzntf7Hvebbq0YEalYeHnQ6Pqc=
!
secure-token scm-admin
 user     $8$weWIhrbLqoV8dq9VuMrT1zO0cbngB12W3A1BSPFeTX8=
```

Sample FMD Configuration File

```
 password $8$mhbx7OpTDhD4sIt2/wM9KVAGjNAbBiWPdJU5QRcw3k4=
!
secure-token scm-oper
 user     $8$PXqvExP/rzVepLSnuB5uKt/FXz73/YL4yP9Q0Tqbt4c=
 password $8$DSyzHE6PZVLptyoI48Y2GRS0IY0D5wl0KFP88GVx4aU=
!
secure-token scm-security
 user     $8$9dhhL8UxFaFK0HAV15BwOnZ/wMJ+Y/nC7CTeigMEegg=
 password $8$k4gfP27grFNqK4wdjjUL6iloJGXfAQswWhyIph+XbxM=
!
secure-token em_login
 user     $8$UdD6nOI1dvBH+yliqLOL46cQRmGBMVRB5GFEX3M/LS8=
 password $8$9SXDc2OjaulrijV5ripRPHCp3p9gKH154m1ZX48RgIw=
!
secure-token staros
 user     $8$NAd0RRbfvrPrN7KBdxJKbW0/PVMYYgvejzVemMSvPG0=
 password $8$JKWanV1gEW2ivmH0afZ3TASo+KmjLF1UvRHExtkZT+M=
!
secure-token esc_netconf
 user     $8$u+ARmNdO3YrjHz/UeBlmMeGbHH1R12Vg8zLyRFSJV/o=
 password $8$uFfZUDm3QqedFvTnZZx6jotSnjJ/uUaAJXIQMBicpLw=
!
secure-token esc_login
 user     $8$hbbCEIg+A77OkjyK1Js4bGh5tXoM9byhfjP7vNdYiY4=
 password $8$I67RNVBIeUiBiOlm0sHIetFhqLZK+IELQ/kG3yATgt4=
!
scm scm
 admin    scm-admin
 oper     scm-oper
 security scm-security
!
snmp-user test
 auth md5 $8$AIKR4Cx10phDn5M9QdXdIgnvKy6QcS/WwE7AruZAytc=
 priv cbc-des $8$IqaVBLqHhppPeAsn5w+gcxCIRzEPG5LDXi5lrkUYHng=
!
fmd fmd-blr
 domain hardware
  monitoring suppress-hw-affected-dn [ sys/rack-unit-1 sys/rack-unit-2 ]
  monitoring suppress-hw-fault-id [ F0438 F0743 F0748 ]
  syslog uas-proxy
 !
 domain vim
  syslog uas-proxy
 !
 domain uas
  monitoring suppress-uas-fault [ ha-event cluster-ha ]
 !
 domain vnfm
  monitoring suppress-uas-fault [ overall ]
 !
 domain vnf-EM
  monitoring suppress-uas-fault [ api-endpoint ha-event cluster-ha ]
 !
 domain vnf
  monitoring suppress-uas-fault [ overall ]
 !
 snmp enterprise-id 9
 snmp v2c disabled
 snmp v3 enabled
 snmp v3 user test
 snmp v3 target 10.105.159.172
  port 162
  user test
```

!
!

Sample FMD Configuration File