Hewlett Packard
Enterprise

Helion OpenStack Carrier Grade 4.0

PLANNING

4.0

Helion OpenStack Carrier Grade 4.0

Planning

Contents

1

Introduction

Overview of Helion OpenStack Carrier Grade 4.0 Planning

Fully planning your Helion OpenStack Carrier Grade 4.0 (HCG 4.0) installation and configuration helps to expedite the process and ensure that you have everything required.

Planning also helps ensure that the requirements of your hosted applications can be met, and the requirements of your Cloud administration and operations team can be met. It also ensures proper integration of THCG 4.0 into the target Data Center or Telecom Office, and helps you plan up front for future cloud growth.

This planning guide is intended to help you select a HCG 4.0 system and deployment option and plan for your installation. It discusses planning for the main points of:

- Deployment Options
- Network Planning
- Storage Planning
- Node Installation Planning
- Node Resource Planning

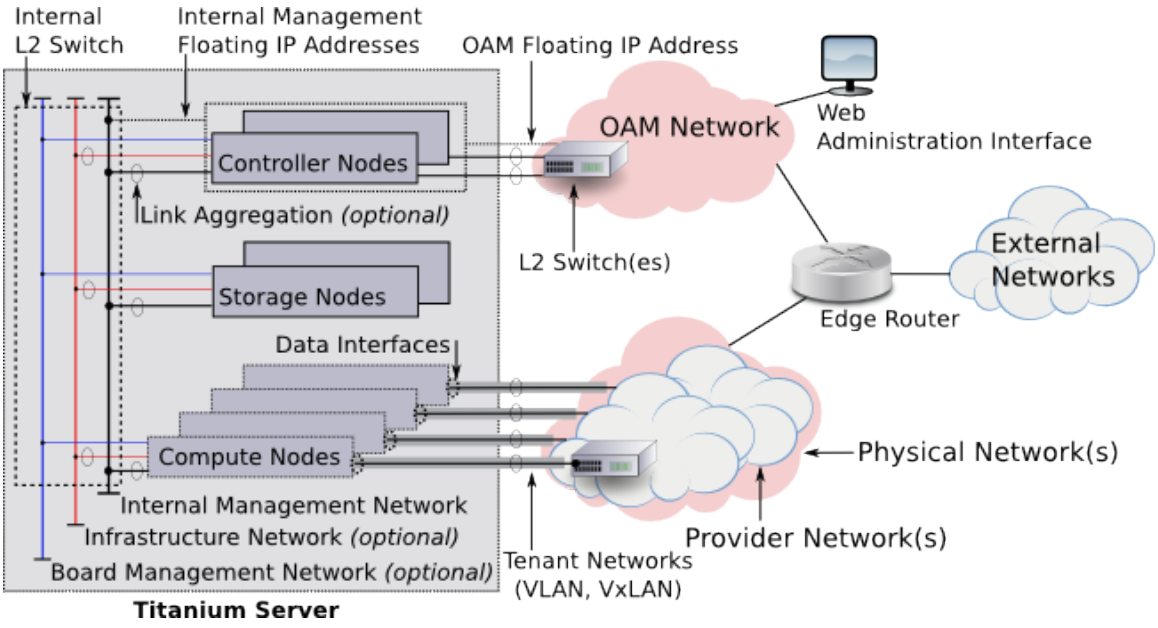
Architecture of a Helion OpenStack Carrier Grade 4.0

The HCG 4.0 architecture supports various types of host, networks, and networking hardware in different configurations. In this documentation, a reference configuration is used for discussion.

The network architecture of HCG 4.0 also supports an optional software-defined networking component, which can be enabled at installation. For more information, see the *Helion OpenStack Carrier Grade 4.0 Release Notes*.

A logical view of the reference hardware platform is illustrated in the following figure.

Figure 1: HCG 4.0 Reference Logical Architecture



NOTE: For clarity, connections to the internal, infrastructure, and board management networks are illustrated for just one host of each type.

NOTE: For a HCG 4.0 CPE configuration, the controller and compute functions are combined on a single host.

Controller Nodes

Run the HCG 4.0 services needed to manage the cloud infrastructure. The two controller nodes run the services in carrier grade mode, that is, as a high-availability cluster. For more information, see *HCG 4.0 Introduction: Controller Nodes and High Availability*.

Controller nodes manage other hosts over the attached internal management network. They also provide external administration interfaces to clients over the OAM network. Two controllers are necessary for HCG 4.0 to operate properly.

Storage Nodes

Run HCG 4.0 storage services. Specifically, storage nodes provide backend-storage for Cinder Block Storage, Glance Image Storage, Swift Object Storage, and remote Nova Ephemeral Storage.

Storage servers are optional, but when used:

- two of them are mandatory for reliability purposes
- they must connect to both the internal management and infrastructure networks

Compute Nodes

Run the HCG 4.0 compute services and hosts the virtual machines providing cpu, memory, optional local storage (local Nova Ephemeral Storage) and L2 Neutron Networking Services. The compute node also provides L3 Neutron Networking Services; L3 Routing, Floating IP, and NAT services.

A compute node connects to the controller nodes over the internal management network, to storage nodes over the infrastructure network, and to the provider networks using its data interfaces.

Data Interfaces

In a compute node, a data interface is a logical network adapter used to connect to one or more provider networks. It is created by mapping a physical network interface on the compute node to the target provider networks. The mapping can use multiple physical network interfaces to support a LAG connection. For more information about LAG modes see *HCG 4.0 Installation*.

Provider networks used by a single data interface must share the same Ethernet MTU value.

Internal L2 Switch

An L2 switching facility, often implemented on a single Top-of-Rack (ToR) switch, used to realize the internal management and infrastructure networks, and depending on the system configuration, the board management network and the pxeboot network. These networks are implemented as follows:

- The internal management network using a dedicated VLAN, typically port-based, as by default, this network is used for PXE booting of new hosts.
  - in some scenarios (IPv6, scope of the mgmt network, and so forth), the management network may not be usable for PXE booting; in this case:
  1. A pxeboot network, dedicated to just PXE booting, uses a dedicated VLAN (port-based), and
  2. the internal management network uses a separate dedicated VLAN (tagged) on the same port.
- The infrastructure network using a dedicated port-based or tagged VLAN.
- The board management network as a dedicated VLAN (port-based) to the BMC port of each node. This applies only if a board management network, for example, iLO (Integrated Lights Out) is in use, and is configured for internal access.
  - Note that the Controller Nodes additionally attach to this board management network, for issuing commands and querying sensors, via a tagged VLAN on the same port as being used by the Controller Node for connecting to the ‘internal management’ network.

As long as the integrity and isolation of the internal, infrastructure, and board management networks are ensured, the internal L2 switch can be realized over physical switching resources that provide connectivity to other networks.

Internal Management Network

An isolated L2 network implemented on the internal L2 switch, used to enable communications among HCG 4.0 hosts for software installation, and management of hosts and virtual machines. This network is only accessible within the HCG 4.0 cluster, and for the most part, it is transparent to management operations of the cloud.

Generally, the internal management network must be unique and dedicated to the HCG 4.0 Cluster. Sharing it with other HCG 4.0 Cluster, or other non-related equipment, is only supported in multi-region configurations.

Infrastructure Network

An optional network used to improve overall performance for a variety of operational functions. When available, it is used by HCG 4.0 during the following operations:

- control and data synchronization when migrating virtual machines between compute nodes
- isolation of storage traffic when accessing storage nodes providing storage services

Overall, an infrastructure network provides a target for the HCG 4.0 software to offload heavy traffic from the internal management network. This prevents sensitive traffic, such as internal heartbeat and monitoring messages, from being starved for bandwidth when background traffic, such as storage-related flows, peaks during normal operations.

When not configured, all infrastructure traffic is carried over the internal management network.

OAM Network

A physical network used to provide external access to the configuration and management facilities of HCG 4.0. It provides connectivity between the controller nodes and the edge router of the OAM network. The web administration interface, and the console interfaces (using SSH) to the controllers, are available on this network. Depending on the system configuration, the OAM network may also be used for controller access to the board management network.

- The OAM network provides access to the OpenStack and HCG 4.0-specific REST APIs, which can be used by users and third-party developers to develop high-level cloud orchestration services.
- The OAM network also provides the controller nodes with access to system-wide resources such as DNS and time servers. Access to the open Internet can also be provided if desired, at the discretion of each particular installation.
- The OAM Network is used for OpenFlow and OVSDB connectivity to an SDN Controller in HCG 4.0 SDN Configurations.
- The OAM Network is also used for syslog connectivity to a Remote Log Server if HCG 4.0 remote logging is configured.

Board Management Network

An optional network used by the controller nodes to perform out-of-band reset and power-on/power-off operations on hosts equipped with iLO (Integrated Lights Out) or Quanta board management modules.

This network can be configured for internal access or external access.

Internal access

In this configuration, the modules are accessible using a VLAN network implemented on an internal L2 switch.

External access

In this configuration, the modules are accessible using the OAM network.

Board management modules are optional. Associated maintenance operations are available only for hosts equipped with them.

For more information, refer to Board Management Network Planning on page 26.

PXE Boot Network

An optional network used in scenarios where the 'internal management' network cannot be used for PXE booting of hosts. For example, if the 'internal management' network needs to be IPv6 (not currently supported for PXE Booting), or in a multi-region environment where the 'internal management' network needs to be shared across regions, however, PXE Booting still needs to be isolated to individual regions.

In these scenarios, the pxeboot network uses a dedicated VLAN (port-based), and the internal management network uses a separate dedicated VLAN (tagged) on the same port.

Physical Network

A physical transport resource used to interconnect compute nodes among themselves and with external networks. Virtual networks, such as provider and tenant networks, are built on top of the physical network. Access to multiple physical networks can be defined by the HCG 4.0 administrator.

Physical networks are not configured by the HCG 4.0 administrator. They are physically provisioned by the data center where the HCG 4.0 cluster is deployed.

Provider Network

A Layer 2 virtual network used to provide the underlying network connectivity needed to instantiate the tenant networks. Multiple provider networks may be configured as required, and realized over the same or different physical networks. Access to external networks is typically granted to the compute nodes via the provider network. The extent of this connectivity, including access to the open Internet, is application dependent.

Provider networks are created by the HCG 4.0 administrator to make use of an underlying set of resources on a physical network. They can be created as being of one of the following types:

flat

A provider network mapped entirely over the physical network. The physical network is used as a single Layer 2 broadcast domain. Each physical network can realize at most one flat provider network.

A provider network of this type supports at most one tenant network, even if its corresponding shared flag is enabled.

VLAN

A provider network implemented over a range of IEEE 802.1Q VLAN identifiers supported by the physical network. This allows for multiple provider networks to be defined over the same physical network, all operating over non-overlapping sets of VLAN IDs.

A set of consecutive VLAN IDs over which the provider network is defined is referred to as a network's *segmentation range*. A provider network can have more than one segmentation range. Each VLAN ID in a segmentation range is used to support the implementation of a single tenant network.

A segmentation range can be shared by multiple tenants if its **shared** flag is set. Otherwise, the segmentation range supports tenant networks belonging to a single specified tenant.

VXLAN

A provider network implemented over a range of VXLAN Network Identifiers (VNIs.) This is similar to the VLAN option, in that it allows multiple provider networks to be defined over the same physical network using unique VNIs defined in segmentation ranges. In addition, VXLAN provides a Layer 2 overlay scheme on Layer 3 networks, enabling connectivity between Layer 2 segments separated by one or more Layer 3 routers.

Tenant Network

A virtual network associated with a tenant. A tenant network is instantiated on a compute node, and makes use of a provider network, either directly over a flat network, or using technologies such as VLAN and VXLAN.

Tenant networks use the high-performance virtual L2 switching capabilities built into the HCG 4.0 software stack of the compute node. They provide switching facilities to the virtual service instances, to communicate with external resources and with other virtual service instances running on the same or different compute nodes.

When instantiated over a provider network of the VLAN or VXLAN type, the VLAN ID or VNI for the tenant network is assigned automatically. The allocation algorithm selects the lowest available ID from any segmentation range owned by the tenant. If no such ID is available, it selects the lowest available ID from any shared segmentation range. The system reports an error when no available ID is found.

Tenant networks created by the administration can also be configured to use a pre-selected VLAN ID or VNI. This can be used to extend connectivity to external networks.

Controller Floating IP Address

A unique IP address shared by the cluster of controller nodes. Only the master controller node is active on this address at any time. The IP address is floating in the sense that it is automatically transferred from one controller node to the other, as dictated by the high-availability directives of the cluster.

Link Aggregation (LAG) or Aggregated Ethernet (AE)

A mechanism that allows multiple parallel Ethernet network connections between two hosts to be used as a single logical connection. HCG 4.0 supports LAG connections on all its Ethernet connections for the purpose of protection (fault-tolerance) only. Different modes of operation are supported. For details, see *HCG 4.0 Installation*.

OAM Network L2 Switch(es)

One or more switches used to provide an entry point into the OAM network.

Provider Network L2 Switches

Provide the entry point into the provider networks. It is through these L2 switches that the compute nodes get integrated as active end points into each of the provider networks.

Edge Router

Provides connectivity to external networks as needed by controller and compute nodes.

Reachability to the open Internet is not mandatory, but strictly application-dependent. Guest applications running on the compute nodes may need to access, or be reachable from, the Internet. Additionally, access to the OAM Network from external networks might be desirable.

Web Administration Interface

Provides the HCG 4.0 main management interface. It is available using any W3C standards-compliant web browser, from any point within the OAM Network where the OAM floating IP address is reachable.

Helion OpenStack Carrier Grade 4.0 Documentation

The HCG 4.0 documentation has been organized to help you locate information for specific types of activities, such as installation, administration, and VNF integration.

Table 1 **Helion OpenStack Carrier Grade 4.0 Documentation**

Document	Description
Introduction to HCG 4.0	This document gives an introduction to HCG 4.0 and provides an overview of system capabilities, information on planning, and recommended workflows. Planning also helps ensure that the requirements of your hosted applications can be met, and the requirements of your Cloud administration and operations team can be met. It also ensures proper integration of HCG 4.0 into the target Data Center or Telecom Office, and helps you plan up front for future cloud growth.
Helion OpenStack Carrier Grade 4.0 Planning	This helps you plan out your HCG 4.0 installation, ensuring that you are fully prepared once you start your installation and configuration.
Helion OpenStack Carrier Grade 4.0 Installation for CPE Systems	This document provides information and instructions for installing HCG 4.0 CPE.
Helion OpenStack Carrier Grade 4.0 Installation for Systems with Controller Storage	This document provides information and instructions for installing HCG 4.0 configurations that are initially deployed using LVM-backed block storage on controller nodes.
Helion OpenStack Carrier Grade 4.0 Installation for Systems with Dedicated Storage	This document provides information and instructions for installing HCG 4.0 configurations that that are initially deployed

Document	Description
	using Ceph-backed block storage on dedicated storage nodes.
Helion OpenStack Carrier Grade 4.0 System Administration	This provides information pertaining to ongoing administration of a HCG 4.0 system, including information for managing the physical nodes and physical networks.
Helion OpenStack Carrier Grade 4.0 Software Management	This document provides instructions for applying patches and software upgrades to HCG 4.0 hosts.
Helion OpenStack Carrier Grade 4.0 Cloud Administration	This guide provides information on topics that an OpenStack administrator would be responsible for, except for the management of the physical nodes and physical networks, which is covered in the System Administration guide.
Helion OpenStack Carrier Grade 4.0 Tenant User's Guide	This provides information about the operational actions that a tenant user can take.
Helion OpenStack Carrier Grade 4.0 VNF Integration	This guide provides information to help you integrate your VNFs into a HCG 4.0 system.
Helion OpenStack Carrier Grade 4.0 Server for Regions	This provides installation and configuration information for deploying HCG 4.0 in any of the supported Regions configurations.
Helion OpenStack Carrier Grade 4.0 Software Development Kit	The HCG 4.0 Software Development Kit (SDK) provides drivers, daemons, API libraries,and configuration files that you can include in a guest image to leverage the extended capabilities of HCG 4.0. These components can be used to enhance or extend the networking features of the applications and to access the virtual machine (VM) management capabilities of HCG 4.0.
Helion OpenStack Carrier Grade 4.0 Engineering Guidelines	This document provides engineering guidelines, rules, and system parameters to assist cloud architects, installers, and administrators in planning, deploying and scaling the HCG 4.0.
Helion OpenStack Carrier Grade 4.0 Software Defined Networking	This document provides information for using an SDN controller to manage Neutron services in HCG 4.0.
Helion OpenStack Carrier Grade 4.0 Release Notes	These include high level details of new features in the current release, as well as

Document	Description
	information about known anomalies or usage caveats.

2

Deployment Options

Deployment Options

Helion OpenStack Carrier Grade 4.0 presents several deployment options. These deployment options include:

- HCG 4.0 Standard Solutions
  - HCG 4.0 Standard Configuration with Dedicated Storage
  - HCG 4.0 Standard Configuration with Controller Storage
  - HCG 4.0 in Multi-Region Environment
- HCG 4.0 CPE

Deployment Models

The HCG 4.0 supports four basic deployment models. Each model supports the same functionality with different ranges of scalability and performance. The models are differentiated

by where the controller, storage, and compute functions reside, with the largest configuration (Standard Configuration with Dedicated Storage) distributing these functions on different hosts and the smallest configuration (HCG 4.0 CPE) combining all three functions onto a single redundant pair of hosts. The Standard Configuration with Controller Storage resides between these two and combines controller and storage functionality onto a redundant pair of hosts supporting one or more Compute Nodes. A HCG 4.0 Multi-Region deployment is also supported, which provides a mechanism for scaling a HCG 4.0 Cloud beyond the limits of a single HCG 4.0 Cloud or Region.

The following sections describe these configurations in more detail and provide guidelines on which situations each model should be used.

The following table provides a guideline for deployment model scaling.

Table 2 **Deployment Model Scaling**

	Controller Nodes	Storage Nodes	Compute Nodes	VMs
Standard Configuration with Dedicated Storage	2	2-8	100	1000 [10 per node, max tested / supported in this release, dependent on sufficient resources (vcpus, memory, storage)]
Standard Configuration with Controller Storage	2	0	10 (typical due to controller storage performance but not fixed)	100 [10 per node, dependent on sufficient resources (vcpus, memory, storage), and limited by controller disk capacity, throughput, and boot times]
HCG 4.0 CPE	2 (with compute function)	0	0	20 [10 per node, dependent on sufficient resources (vcpus, memory, storage)]

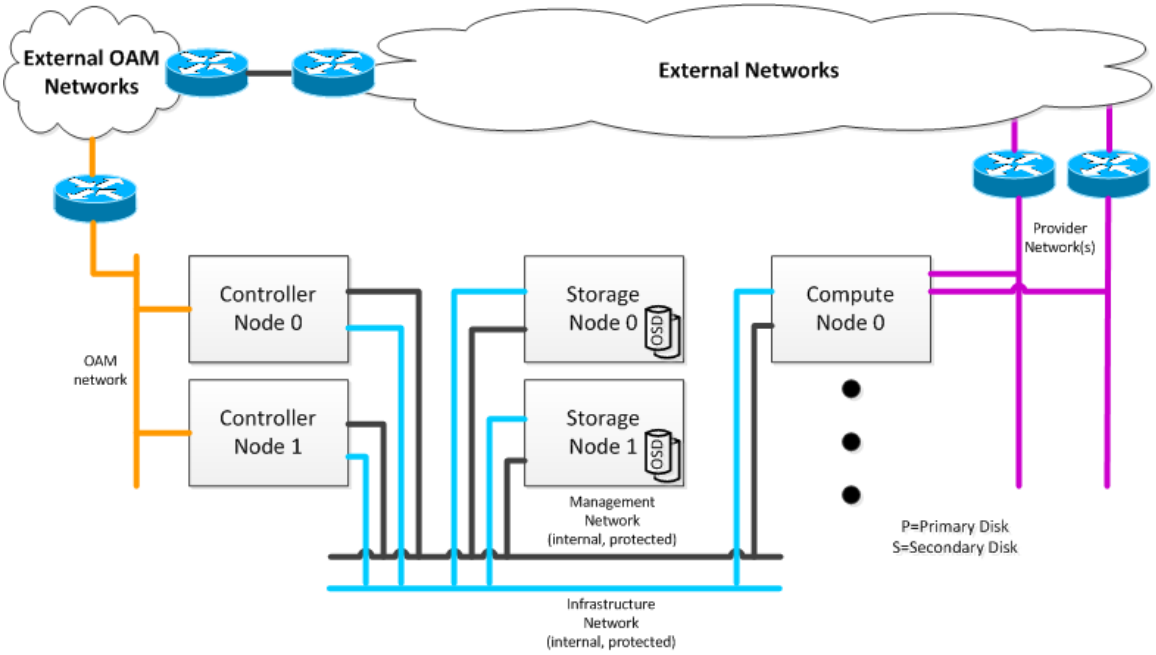
The Virtual Machine (VM) mix used as a baseline in creating this table is a heterogeneous mix of VMs and features – 1-4 cores, 1-4 GB memory, 600 MB and 3.6 GB images, 1/2/4/10/20/40 GB volumes, single and multi-NUMA nodes, local storage and remote storage, launch via nova and heat, dedicated and shared CPU policies, vm scaling, cpu scaling, and 1-3 network connections per VM.

Standard Configuration with Dedicated Storage

Deployment of HCG 4.0 with Dedicated Storage Nodes provides the highest capacity (single region) and performance with scalability up to 100 Compute Nodes and 1,000 VMs, assuming sufficient CPU, memory and storage resources are deployed and a VM distribution similar to that described in the previous section. The differentiating physical feature of this model is that the controller, storage, and compute functionalities are deployed on separate physical hosts allowing Controller Nodes, Storage Nodes, and Compute Nodes to scale independently from each other.

The following figure shows the nodes and logical networks supported in this configuration.

Figure 2: Dedicated Storage Configuration



The Controller Nodes provide the control function for the system and two Controller Nodes are required to provide active/standby redundancy. The Controller Nodes can be sized in terms of the server and peripherals, for example, CPU cores/speed, memory, storage, and network interfaces depending on requirements. Two to 8 Storage Nodes, deployed in pairs, provide the storage function for Glance images, Cinder volumes, remote Nova Ephemeral volumes and Swift Containers. Redundancy and scalability is provided through the number of Ceph Object Storage Device (OSD) pairs, with more OSDs providing more capacity and better storage performance. OSD size and speed, optional SSD Ceph journals, optional SSD Ceph cache tiering, CPU cores/speed, memory, disk controllers, and networking also impact the scalability and perofrmance of the storage function. This model supports up to 100 Compute Nodes, each of which can be sized independently in terms of CPU cores/speed, memory, local storage, and interfaces. The actual number of VMs supported may be limited by availability of Compute Node resources and has been tested to 1000 VMs in a 100 Compute Node configuration.

The following logical networks are supported in this configuration.

- External Operations and Maintenance (OAM) network (Controller Nodes only)
- Internal Management network (all nodes)
- Optional pxeboot Network (all nodes)
- Infrastructure Network
- Optional Board Management Network
- Provider Network(s) (all Compute Nodes)

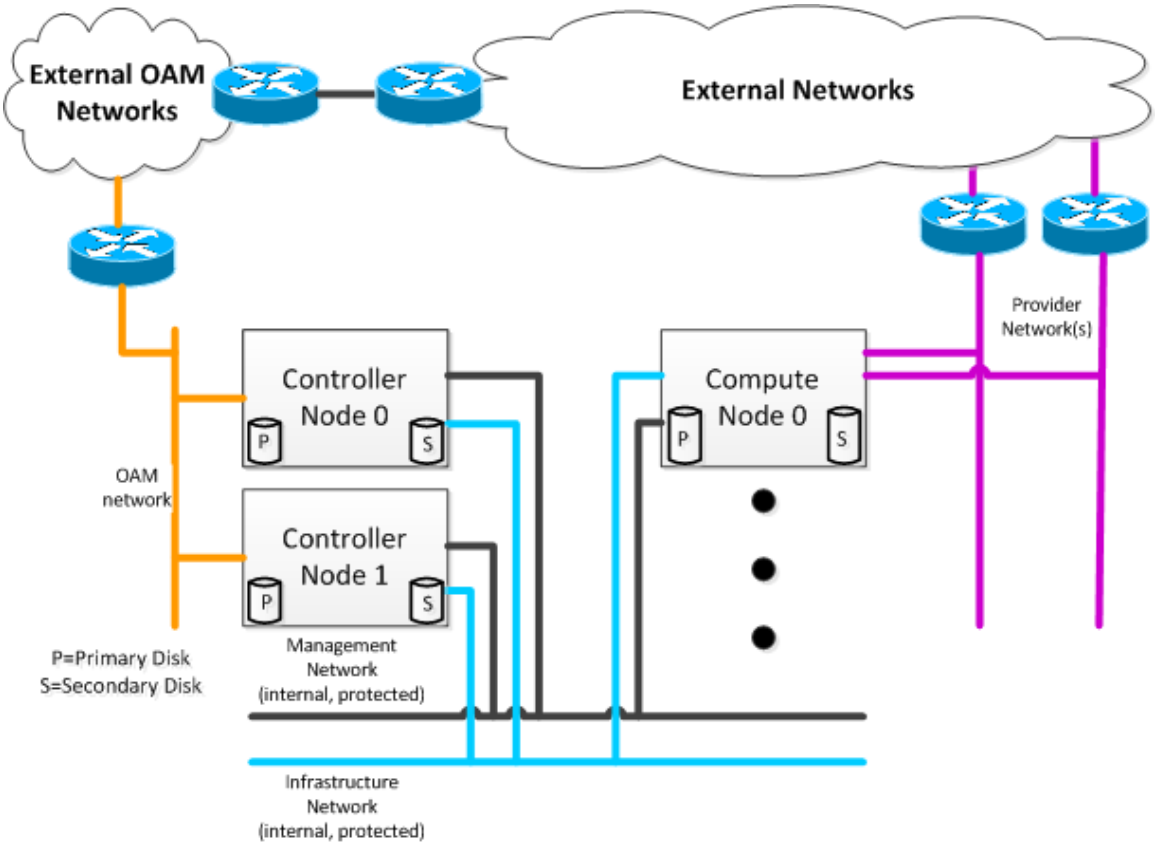
Logical interfaces can be dedicated to single or link aggregated interfaces for maximum performance and isolation or combined in various combinations with other logical interfaces onto a common physical interface or a link aggregated interface and separated into isolated L2 LAN segments in the top-of-rack switch.

Standard Configuration with Controller Storage

The HCG 4.0 supports a small-scale deployment option using Logical Volume Manager/Internet Small Computer System Interface (LVM/iSCSI) as a back end for cinder volumes on the Controller Nodes instead of using dedicated Storage Nodes. This configuration uses dedicated physical disks in the Controller Node synchronized with the other Controller Node. The Primary disk is used by the platform and for glance image storage. The secondary disk is used for cinder volumes. Because the primary and secondary disks are single devices, this configuration does not scale to the capacity of the Standard Configuration with Dedicated Storage nor does it provide the same performance and latencies for activities like VM creation/deletion. The number of Compute Nodes and VMs supported is limited by the cinder/glance storage backend capacity, storage throughput, and VM launch time requirements. The following figure shows a typical deployment.

The same OAM, Internal Management, optional PXEBoot, optional (here) Infrastructure, Board Management, and Provider Network networks are supported in this configuration as in the Standard Configuration with Dedicated Storage.

Figure 3: Standard Configuration with Controller Storage Diagram



Helion OpenStack Carrier Grade 4.0 in Multi-Region Environment

Helion OpenStack Carrier Grade 4.0 can be deployed in a Multi-Region configuration as the Primary Region, Secondary Region, or as both.

A Region is a discrete OpenStack environment with dedicated API endpoints that typically shares at least the Identity/Keystone service with other regions. Regions configurations provide for the scalability of HCG 4.0 clouds and inter-working with non-HCG 4.0 clouds.

HCG 4.0 provides the ability to be deployed in a Regions configuration. It is flexible, also, in regard to how it is deployed in a Regions configuration. Supported Regions configurations include:

- HCG 4.0 as Primary and Secondary Region
- Third-Party OpenStack Cloud as Primary Region and HCG 4.0 as Secondary Region

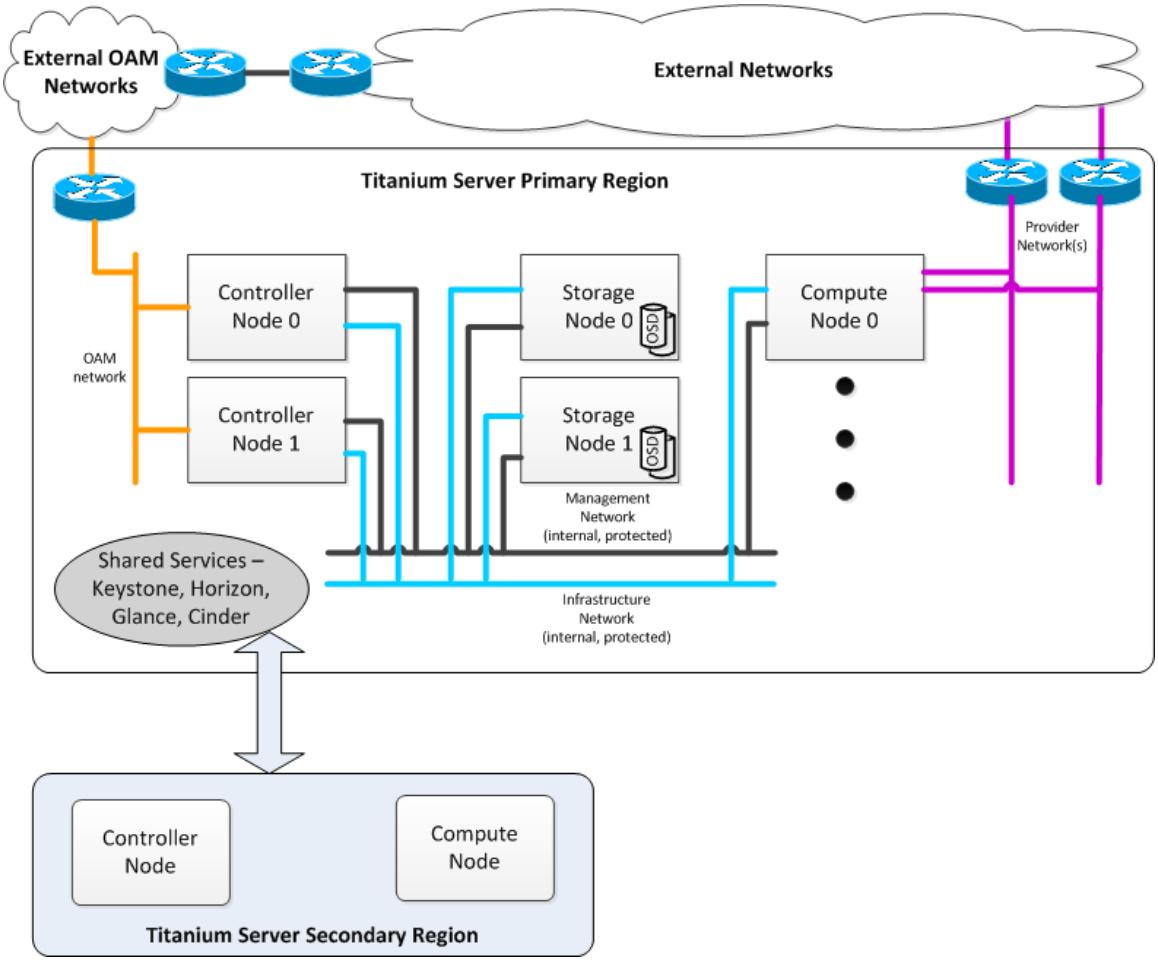
Refer to the *HCG 4.0 for Regions* guide for information about other Multi-Region Configurations supported, which include both additional shared services and Multi-Region configurations with non-HCG4.0 OpenStack Clouds.

Helion OpenStack Carrier Grade 4.0 as Primary and Secondary Region

This configuration provides for scalability. It enables you to scale the number of overall compute nodes to more than what is supported by a single HCG 4.0 Region, with some level of integration (through shared services) across all regions.

As shown in the figure below, selected Controller Services are shared between Regions. Shared Central Services provide multi-Region-wide management capabilities.

Figure 4: HCG 4.0 as Primary and Secondary Region



Helion OpenStack Carrier Grade 4.0 CPE

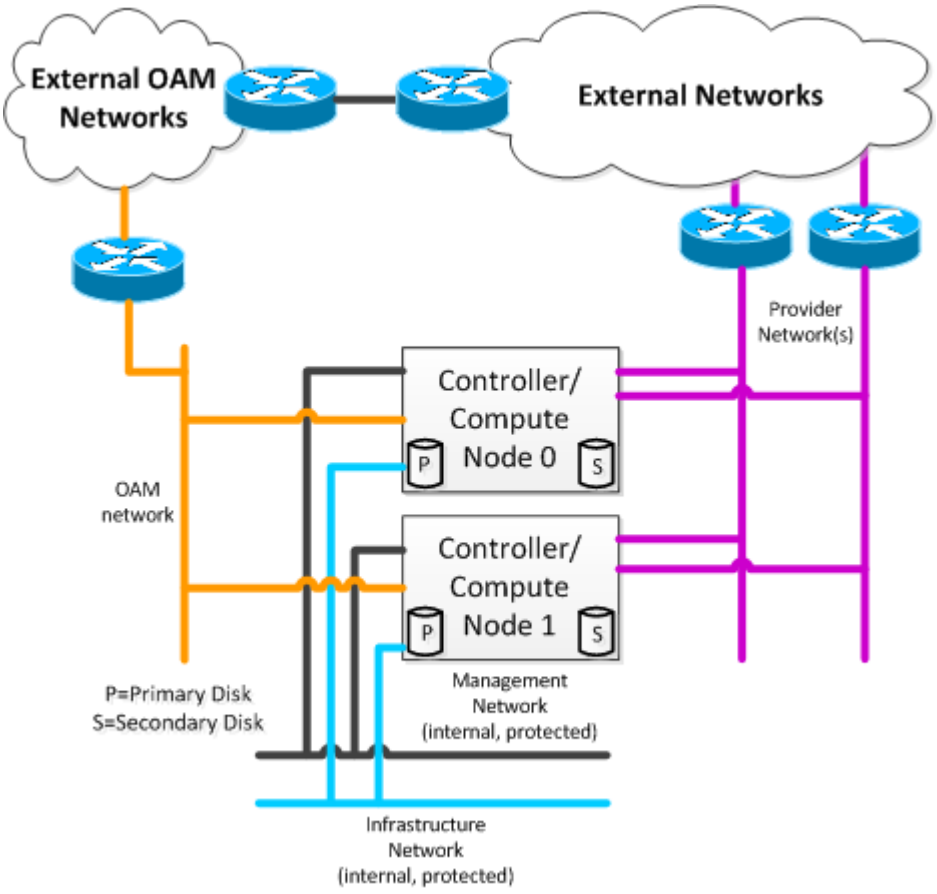
Helion OpenStack Carrier Grade 4.0 CPE (Customer Premises Equipment) provides a scaled down HCG 4.0 deployment option that combines controller, storage, and compute functionality on a redundant pair of hosts.

In this configuration, the controller functionality is active/standby across the nodes and the compute functionality is active/active, meaning VMs can be deployed on both hosts. LVM/iSCSI

is used as the cinder volume backend. Processor bandwidth and memory is reserved for the controller function.

The same OAM, Management, optional (here) Infrastructure, optional Board Management, and Provider Network interfaces are supported in this configuration.

Figure 5: HCG 4.0 CPE Configuration Diagram



3

Network Planning

Network Planning

When planning the deployment of HCG 4.0, it is important to understand the available networking options and how to take advantage of them for the benefit of the end users.

Network Requirements

Helion OpenStack Carrier Grade 4.0 uses several different types of networks, depending on the size of the system and the features in use.

Available networks include the PXE boot network, the internal management network, the OAM network, the infrastructure network, the board management network, and one or more provider networks.

The internal management network is required by all systems for internal communication; the OAM network is required for external control access; and at least one provider network is required for use by VMs. The need for other networks is determined by the system configuration (CPE, standard with controller storage, or standard with dedicated storage), the number of hosts supported, the expected system loads, and the availability of features such as Integrated Lights Out (iLO) board management.

You can optionally consolidate more than one network on a single physical interface. For more information, see Shared (VLAN) Ethernet Interfaces on page 34.

NOTE: The HCG 4.0 controllers use IP multicast messaging on the internal management, infrastructure, and OAM networks. To prevent loss of controller synchronization, ensure that the switches and other devices on these networks are configured with appropriate settings.

Networks for a Helion OpenStack Carrier Grade 4.0 CPE System

For a HCG 4.0 CPE system, HCG 4.0 recommends a minimal network configuration.

HCG 4.0 CPE uses a small hardware footprint consisting of two hosts, plus a network switch for connectivity. Network loading is typically low. The following network configuration typically meets the requirements of such a system:

- An internal management network
- An OAM network, optionally consolidated on the management interface
- One or more data networks, optionally consolidated on the management interface

NOTE: You can enable secure HTTPS connectivity on the OAM network during system installation. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Secure HTTPS External Connectivity*.

Networks for a Standard System with Controller Storage

For a system that uses controller storage, HCG 4.0 recommends an intermediate network configuration.

HCG 4.0 systems with controller storage use controller and compute hosts only. Network loading is low to moderate, depending on the number of compute hosts and VMs. The following network configuration typically meets the requirements of such a system:

- An internal management network
- An OAM network, optionally consolidated on the management interface
- Several data networks, which may optionally be consolidated on the management interface or the OAM interface or both, depending on the number of compute hosts and VMs to be supported
- An optional infrastructure interface, if required to offload the management network for a large number of compute hosts
- Optionally, a PXE Boot Server to support controller-0 initialization

NOTE: You can enable secure HTTPS connectivity on the OAM network during system installation. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Secure HTTPS External Connectivity*.

Networks for a Standard System with Dedicated Storage

For a system that uses dedicated storage, HCG 4.0 recommends a full network configuration.

HCG 4.0 with dedicated storage include storage hosts to provide Ceph-backed block storage. Network loading is moderate to high, depending on the number of compute hosts, VMs, and storage clusters. The following network configuration typically meets the requirements of such a system:

- An internal management network
- An infrastructure network (required for systems with storage hosts)

- An OAM network
- One or more data networks
- An optional PXE boot network
  - if the internal management network is required to be on a VLAN-tagged network
  - or, if the internal management network is shared with other equipment

On moderately loaded systems, the OAM and data networks can be consolidated on the management or infrastructure interfaces.

NOTE: You can enable secure HTTPS connectivity on the OAM network during system installation. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Secure HTTPS External Connectivity.*

Networks for a Helion OpenStack Carrier Grade 4.0 Multi-Region Environment

For a HCG 4.0 Region system, the network requirements depend in part on the region configuration.

HCG 4.0 Region systems act as components of larger network systems. The following network configuration typically meets the requirements of such a system:

- A PXE boot network (required)
- An internal management network that is “shared” with other Regions
- An infrastructure network, “shared” with other Regions (required for systems with storage hosts)
- An OAM network, optionally “shared” with other Regions
- One or more DATA Networks that *must not* be “shared” with other Regions

On moderately loaded systems, the OAM and data networks can be consolidated on the management or infrastructure interfaces.

NOTE: You can enable secure HTTPS connectivity on the OAM network during system installation. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Secure HTTPS External Connectivity.*

The PXE Boot Network

You can set up a PXE Boot network for booting all nodes, in order to configure the management network differently.

By default, the internal management network is used for PXE booting of new hosts, and therefore must be untagged. It is also limited to IPv4, because the HCG 4.0 installer does not support IPv6 UEFI booting. If, for deployment reasons, the internal management network needs to be on a VLAN-tagged network, or if it must support IPv6, you can configure the optional untagged PXE boot network for PXE booting of new hosts using IPv4.

The Internal Management Network

The internal management network must be implemented as a single, dedicated, Layer 2 broadcast domain for the exclusive use of each HCG 4.0 cluster. Sharing of this network by more than one HCG 4.0 cluster is only supported in a Multi-Region Configuration.

During the HCG 4.0 software installation process, several network services such as BOOTP, DHCP, and PXE, are expected to run over the internal management network. These services are used to bring up the different hosts to an operational state. Therefore, it is mandatory that this network be operational and available in advance, to ensure a successful installation.

On each host, the internal management network can be implemented using a 1 Gb or 10 Gb Ethernet port. In either case, requirements for this port are:

- it must be capable of PXE-booting
- it can be used by the motherboard as a primary boot device

NOTE: If required, the internal management network can be configured as a VLAN-tagged network. In this case, a separate IPv4 PXE boot network must be implemented as the untagged network on the same physical interface. This configuration must also be used if the management network must support IPv6.

Internal Management Network Planning

The internal management network is a private network, visible only to the hosts in the cluster.

You must consider the following guidelines:

- By default, the internal management network is used for PXE booting of new hosts, and therefore must be untagged. It is also limited to IPv4, because the HCG 4.0 installer does not support IPv6 UEFI booting. If, for deployment reasons, the internal management network needs to be on a VLAN-tagged network, or if it must support IPv6, you can configure the optional untagged PXE boot network for PXE booting of new hosts using IPv4.
- The internal management network and the infrastructure network must use the same IP version (IPv4 or IPv6).
- You can use any 1 G or 10 G interface on the hosts to connect to this network, provided that the interface supports network booting and can be configured from the BIOS as the primary boot device.
- The infrastructure network is automatically assigned the same type of IP address assignment as the internal management network (dynamic or static).
- If static IP address assignment is used, you must use the **system host-add** command to add new hosts, and to assign IP addresses manually. New hosts are *not* automatically added to the inventory when they are powered on, and they display the following message on the host console:

```
This system has been configured with static management
and infrastructure IP address allocation. This requires
that the node be manually provisioned in System
Inventory using the 'system host-add' CLI, GUI, or
sysinv-api equivalent.
```

- For the IPv4 address plan, use a private IPv4 subnet as specified in RFC 1918. This helps prevent unwanted cross-network traffic on this network.

It is suggested that you use the default subnet and addresses provided by the controller configuration script.

- You can assign a range of addresses on the management subnet for use by HCG 4.0. If you do not assign a range, HCG 4.0 takes ownership of all available addresses.
- The HCG 4.0 controllers use IP multicast messaging on the internal management, infrastructure, and OAM networks. To prevent loss of controller synchronization, ensure that the switches and other devices on these networks are configured with appropriate settings.
- For multi-region deployments, you can define a restricted multicast addressing range on the management network during system installation. Addresses for affected services are allocated automatically from the range. The same range also applies to the infrastructure network.

The Infrastructure Network

This network supplements the management network. It is required for Ceph-backed systems, and optional for other systems.

As with the internal management network, the infrastructure network must be implemented as a single, dedicated, Layer 2 broadcast domain for the exclusive use of each HCG 4.0 cluster. Sharing of this network by more than one HCG 4.0 cluster is only supported in a Multi-Region Configuration.

The infrastructure network can be implemented as a 10 Gb Ethernet network. In its absence, all infrastructure traffic is carried over the internal management network.

Infrastructure Network Planning

The infrastructure network is a private network visible only to the hosts in the cluster. It is optional unless storage nodes are part of the cluster.

For the most part, the infrastructure network shares the design considerations applicable to the internal management network. It can be implemented using a 10 Gb Ethernet interface. It can be VLAN-tagged, enabling it to share an interface with the management or OAM network. It can own the entire IP address range on the subnet, or a specified range.

You can add an infrastructure network after system installation. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Adding an Infrastructure Network*.

The infrastructure network supports dynamic or static IP address assignment. The setting is determined by the setting for the internal management network. If static assignment is used, you must manually assign IP addresses on the infrastructure network for the other hosts in the cluster, using the **system host-addr-add** command.

If a multicast messaging range is assigned on the management network, the infrastructure network uses the same range.

The infrastructure network must use the same IP version as the management network (IPv4 or IPv6).

The decision whether to implement an infrastructure network depends on the infrastructure traffic levels expected. The following table provides general guidelines on when to implement the infrastructure network, using the number of compute nodes in the HCG 4.0 cluster as an indication of expected traffic levels.

Number of Compute Nodes	Infrastructure Network Implementation
2 to 4	No infrastructure network required. All infrastructure traffic is carried over the internal management network.
5 or more	Use 10 Gb Ethernet

In addition to the number of compute nodes, other factors must be taken into account when deciding the type of infrastructure network to use. For example, live migration of large, memory-intensive, guest applications is likely to require additional network resources that might not be available in the current configuration. It is always safe to implement the fastest infrastructure network possible.

NOTE: The HCG 4.0 controllers use IP multicast messaging on the internal management, infrastructure, and OAM networks. To prevent loss of controller synchronization, ensure that the switches and other devices on these networks are configured with appropriate settings.

The OAM Network

This network provides for control access.

You should ensure that the following services are available on the OAM Network:

DNS Service

Needed to facilitate the name resolution of servers reachable on the OAM Network.

HCG 4.0 can operate without a configured DNS service. However, a DNS service should be in place to ensure that links to external references in the current and future versions of the Web administration interface work as expected.

NTP Service

The Network Time Protocol (NTP) can be optionally used by the HCG 4.0 controller nodes to synchronize their local clocks with a reliable external time reference. However, HCG 4.0 strongly recommends that this service be available, among other things, to ensure that system-wide log reports present a unified view of the day-to-day operations.

The HCG 4.0 compute nodes and storage nodes always use the controller nodes as the de-facto time server for the entire HCG 4.0 cluster.

OAM Network Planning

The OAM (operations, administration, and management) network enables the Web administration interface, the command-line management clients, SNMP interface, and the REST APIs to manage the HCG 4.0 cluster.

The OAM Network is also used for OpenFlow and OVSDB connectivity to an SDN Controller in HCG 4.0 SDN Configurations, and for syslog connectivity to a Remote Log Server if HCG 4.0 remote logging is configured.

On systems where an optional board management network is configured for external access, the OAM network also provides access to the board management modules.

The OAM network supports IPv4 or IPv6 addressing. Consider the following guidelines:

- Deploy proper firewall mechanisms to access this network. Ensuring that access to the HCG 4.0 management interfaces is not compromised should be of primary concern.

  HCG 4.0 includes a default firewall for the OAM network, using the **Netfilter** framework. You can optionally configure the system to support additional rules. For more information, see Firewall Options on page 25.
- Consider whether the OAM network needs access to the open Internet. Limiting access to an internal network might be advisable, keeping in mind that access to configured DNS and NTP servers may still be needed.
- VLAN tagging is supported, enabling this network to share an interface with the internal management or infrastructure networks.
- The IP addresses of the DNS and NTP servers must match the IP address plan (IPv4 or IPv6) of the OAM network.
- For an IPv4 address plan:
  - The OAM floating IP address is the only address that needs to be visible externally. Therefore you must plan for valid definitions of its IPv4 subnet and default gateway.
  - The physical IPv4 addresses for the controllers don't need to be visible externally, unless you plan to use them during SSH sessions to prevent potential service breaks during the connection. You still need to plan for their IPv4 subnet, but you can limit access to them as required.
  - Outgoing packets from the active or secondary controller use the controller's IPv4 physical address, not the OAM floating IP address, as the source address.
- For an IPv6 address plan:
  - Outgoing packets from the active controller use the OAM floating IP address as the source address. Outgoing packets from the secondary controller use the secondary controller's IPv6 physical IP address.
- The HCG 4.0 controllers use IP multicast messaging on the internal management, infrastructure, and OAM networks. To prevent loss of controller synchronization, ensure that the switches and other devices on these networks are configured with appropriate settings.
- For multi-region deployments, you can define a restricted multicast addressing range on the OAM network during system installation.

Firewall Options

HCG 4.0 incorporates a firewall for the OAM network. During initial configuration, you can specify an additional file in order to augment or override the default rules.

For more information about specifying a firewall rules file during controller configuration, see *HCG 4.0 Installation: The Controller Configuration Script*.

The HCG 4.0 firewall uses the **Netfilter** framework to implement a firewall on the OAM network. If the system is configured to support an optional firewall rules file, you can introduce custom rules by adding entries in this file.

Two input chains are supported for custom rules: **INPUT-custom-pre**, for rules to be processed before the default rules, and **INPUT-custom-post**, for rules to be processed after the default rules.

A minimal set of rules is always applied before any custom rules, as follows:

- Non-OAM traffic is always accepted.
- Egress traffic is always accepted.
- Service manager (SM) traffic is always accepted.
- SSH traffic is always accepted.

For the default rules used by HCG 4.0, see *Helion OpenStack Carrier Grade 4.0 System Administration: Default Firewall Rules*. For more about custom rules, see *Helion OpenStack Carrier Grade 4.0 System Administration: Firewall Rules File Format*.

You can validate the file using the following command:

```
$ iptables-restore --noflush --test < filename
```

where *filename* is the path and name of the file.

⚠ **CAUTION:** You should validate the file before you run the configuration controller script. If the file is invalid, you must correct any errors and then start the script again from the beginning.

The Board Management Network

This network is optionally available to manage hosts equipped with board management modules.

The board management network implementation depends on the system configuration.

Internal access

On a system configured for internal access to the board management network, the board management modules are connected to VLAN-tagged ports on the internal L2 switch, and the controller is attached to the VLAN using the internal management network interface. For more about the internal L2 switch, refer to Architecture of a HCG 4.0 on page 2.

External access

On a system configured for external access to the board management network, the board management modules are assigned IP addresses accessible from the OAM network, and the controller uses the OAM network to connect to them.

The system configuration is determined at installation. If a board management network is specified during the controller configuration script, then the network is configured for internal access. Otherwise, the board management network uses external access.

Board Management Network Planning

The board management network is an optional network used for hardware management facilities.

This network provides access to iLO3 or iLO4 board management modules optionally installed in the hosts. These modules provide support for remote reset and power control of the hosts, and in some cases for hardware sensor monitoring and reporting.

The board management network can be implemented so that it is accessible externally using the OAM network, or only accessible internally to the hosts in the cluster. This choice is made at

system installation, during the controller configuration script. For details, see *HCG 4.0 Installation: The Controller Configuration Script*.

Configuration for External Access

For external access, do not configure a board management network during the controller configuration script (press **n** when prompted).

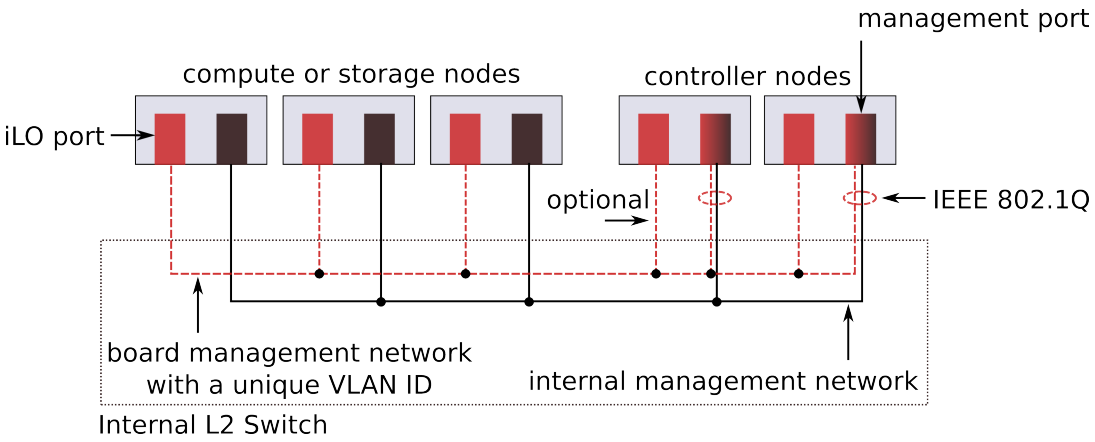
After completing the script, you must also do the following:

- Configure each iLO module to use static IP addressing.
- Peform additional host and module provisioning, as described in Host and iLO Module Provisioning for a Board Management Network on page 28.
- Ensure that the OAM Default Gateway specified during controller configuration has access to the board management network.

Configuration for Internal Access

For internal access, configure a board management network during the controller configuration script (press **y** when prompted, and then specify a VLAN ID and a subnet). This configures a VLAN ID on the controller management interface for use with a board management network. To complete the VLAN, you must connect the modules to VLAN-tagged ports on an internal L2 switch. The modules are assigned IP addresses from the specified subnet.

Figure 6: Internal-access board management network



The figure above illustrates how the board management network operates when configured for internal access. Compute and storage nodes connect using iLO modules, while controller nodes use the management port in tagged mode. As indicated in the figure, the controller nodes can optionally attach using their own iLO modules, if available.

- On the HCG 4.0 networks:
  - Designate a VLAN ID for use by the board management network.
  - Configure this VLAN ID as part of the software installation process.
  - Ensure that the internal management network is operational and that all participant hosts are attached to it.
- On the HCG 4.0 internal L2 switch:

  - Configure a VLAN using the designated VLAN ID for the board management network.
  - Add each of the ports used to connect an iLO module to the VLAN.
  - Add the ports used to connect the controllers to the internal management network to the VLAN. You must ensure that these ports are tagged ports for this VLAN, that is, that outgoing board management traffic (toward the controllers) uses IEEE 802.1Q tagging. Other management traffic on this port is untagged.
- Configure each iLO module to use DHCP.
- Peform additional host and module provisioning, as described in Host and iLO Module Provisioning for a Board Management Network on page 28.

For details on configuring or provisioning switch ports or iLO modules, consult the user documentation supplied with the equipment.

Host and iLO Module Provisioning for a Board Management Network

For internal or external access, you must perform the following additional provisioning:

- Provision each iLO module with a username and password for secure access.
- Provision each host with the MAC address of the attached iLO module. This is required so that the system can associate the module's assigned IP address with the correct host.
- For external-access configurations only, provision each host with the IP address of the attached iLO module.
- Provision each host with the username and password of the attached iLO module. This is required for controller access to the module.

For module provisioning, consult the user documentation supplied with the equipment.

For host provisioning, you can use the Board Management tab on the Edit Host dialog box, or the CLI. For more information, see *HCG 4.0 Installation*.

Data Networks

The physical Ethernet interfaces on HCG 4.0 nodes can be configured to use one or more data networks.

The management or infrastructure interfaces, or both, can be configured with an additional data network. Data networks can use VLAN tagging, allowing them to share an Ethernet or aggregated Ethernet interface with other networks.

Data Network Planning

The data network is the backing network for the overlay tenant networks and therefore has a direct impact on the networking performance of the guest.

There are several factors that contribute to the overall network performance.

HCG 4.0 supports the use of consolidated interfaces for the management, infrastructure, OAM, and data networks. For best performance, HCG 4.0 recommends that you use dedicated interfaces.

For detailed information about network performance (including vSwitch, SRIOV, and PCI-PT dimensioning), see *HCG 4.0 Engineering Guidelines: Networking dimensioning*.

For other factors that may impact the overall guest network performance, see *HCG 4.0 System Engineering Guidelines: Compute Node Dimensioning*.

L2 Access Switches

L2 access switches connect the HCG 4.0 hosts to the different networks. Proper configuration of the access ports is necessary to ensure proper traffic flow.

One or more L2 switches can be used to connect the HCG 4.0 hosts to the different networks. When sharing a single L2 switch you must ensure proper isolation of the network traffic. Here is an example of how to configure a shared L2 switch:

- one port-based VLAN for the internal management network
- one port-based VLAN for the OAM network
- one port-based VLAN for an optional board management network, if the network is configured for internal access
- one or more sets of VLANs for provider networks. For example:
  - one set of VLANs with good QoS for bronze tenants
  - one set of VLANs with better QoS for silver tenants
  - one set of VLANs with the best QoS for gold tenants

When using multiple L2 switches, there are several deployment possibilities. Here are some examples:

- A single L2 switch for the internal management and OAM networks. Port- or MAC-based network isolation is mandatory.
- One or more L2 switches, not necessarily inter-connected, with one L2 switch per provider network.
- Redundant L2 switches to support link aggregation, using either a failover model, or Virtual Port Channel (VPC) for more robust redundancy. For more information, see Deploying Redundant Top-of-Rack Switches on page 30.

Switch ports that send tagged traffic are referred to as *trunk* ports. They usually participate in the Spanning Tree Protocol (STP) from the moment the link goes up, which usually translates into several seconds of delay before the trunk port moves to the forwarding state. This delay is likely to impact services such as DHCP and PXE which are used during regular operations of HCG 4.0.

Therefore, you must consider configuring the switch ports to which the management interfaces are attached to transition to the forwarding state immediately after the link goes up. This option is usually referred to as a *PortFast*.

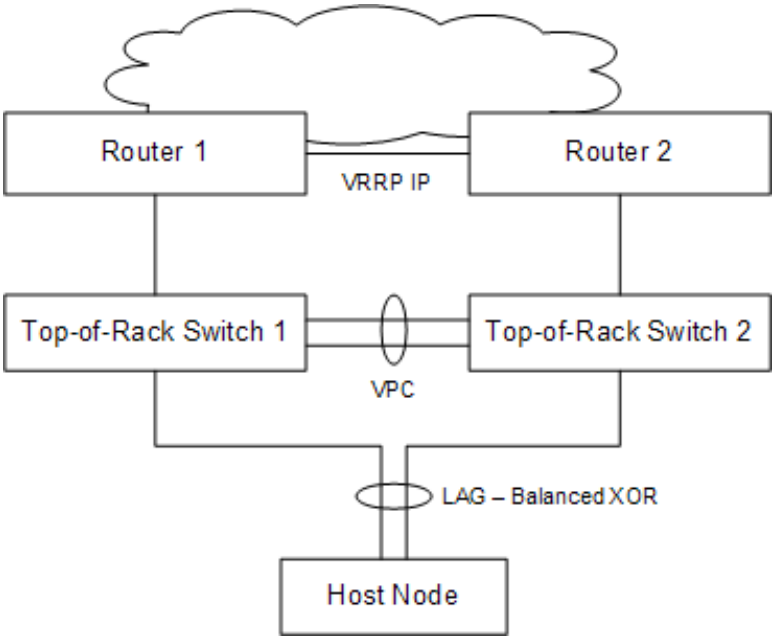
You should also consider configuring these ports to prevent them from participating on any STP exchanges. This is usually done by configuring them to avoid processing inbound and outbound BDPU STP packets completely. Consult your switch's manual for details.

Deploying Redundant Top-of-Rack Switches

For a system that uses link aggregation on some or all networks, you can configure redundant top-of-rack switches for additional reliability.

In a redundant ToR switch configuration, each link in a link aggregate is connected to a different switch, as shown in the accompanying figure. If one switch fails, another is available to service the link aggregate.

Figure 7: Redundant Top-of-Rack Switches



HCG 4.0 recommends that you use switches that support Virtual Port Channel (VPC). When VPC is used, the aggregated links on the switches act as a single LAG interface. Both switches are normally active, providing full bandwidth to the LAG. If there are multiple failed links on both switches, at least one connection in each aggregate pair is still functional. If one switch fails, the other continues to provide connections for all LAG links that are operational on that switch. For more about configuring VPC, refer to your switch documentation.

You can also use an active/standby failover model for the switches, but at some cost to overall reliability. If there are multiple failed links on both switches, then optimally the switch with the greatest number of functioning links is activated, but some links on that switch could be in a failed state. In addition, when only one link in an aggregate is connected to an active switch, the LAG bandwidth is limited to the single link.

NOTE: You can further enhance system reliability by using redundant routers. For more information, refer to your router documentation.

DNS and NTP Servers

HCG 4.0 supports up to three DNS servers and three NTP servers.

These can be specified when running the controller configuration script or at any time afterwards.

Ethernet Interfaces

Ethernet interfaces, both physical and virtual, play a key role in the overall performance of the virtualized network. Therefore, it is important to understand the available interface types, their configuration options, and their impact on network design.

About LAG/AE Interfaces

You can use link aggregation (LAG) for Ethernet interfaces. HCG 4.0 supports up to four ports in a LAG group.

Ethernet interfaces in a LAG group can be attached either to the same L2 switch, or to multiple switches in a redundant configuration. For more information about L2 switch configurations, see L2 Access Switches on page 29. For information about the different LAG modes, see *Helion OpenStack Carrier Grade 4.0 System Administration:Link Aggregation Settings*.

Ethernet Interface Configuration

You can review and modify the configuration for physical or virtual Ethernet interfaces using the Web administration interface or the CLI.

Physical Ethernet Interfaces

The physical Ethernet interfaces on HCG 4.0 nodes are configured to use the following networks:

- the internal management network
- the external OAM network
- the infrastructure network, if present
- one or more data networks

A single interface can optionally be configured to support more than one network using VLAN tagging (see Shared (VLAN) Ethernet Interfaces on page 34). In addition, the management or infrastructure interfaces, or both, can be configured with an additional data network (see *Helion OpenStack Carrier Grade 4.0 System Administration: Editing Interface Settings*).

On the controller nodes, all Ethernet interfaces are configured automatically when the nodes are initialized, based on the information provided in the controller configuration script (see *HCG 4.0 Installation: The Controller Configuration Script*). On compute and storage nodes, the Ethernet interfaces for the internal management network are configured automatically. The remaining interfaces require manual configuration.

NOTE: If a network attachment uses LAG, the corresponding interfaces on the storage and compute nodes must also be configured manually to specify the interface type.

You can review and modify physical interface configurations from the Web administration interface or the CLI. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Editing Interface Settings*.

You can also save the interface configurations for a particular node to use as a *profile* or template when setting up other nodes. For more information, see *Helion OpenStack Carrier Grade 4.0 Installation: Hardware Profiles*.

Virtual Ethernet Interfaces

The virtual Ethernet interfaces for guest VMs running on HCG 4.0 are defined when an instance is launched. They connect the VM to *tenant networks*, which are virtual networks defined over *provider networks*, which in turn are abstractions associated with physical interfaces assigned to data networks on the compute nodes.

The following virtual network interfaces are available:

- **avp** (Accelerated Virtual Port)
- **e1000** (Intel e1000 Emulation)
- **ne2k_pci** (NE2000 Emulation)
- **pcnet** (AMD PCnet/PCI Emulation)
- **rtl8139** (Realtek 8139 Emulation)
- **virtio** (VirtIO Network)
- **pci-passthrough** (PCI Passthrough Device)
- **pci-sriov** (SR-IOV device)
- Unmodified guests can use Linux networking and virtio drivers. This provides a mechanism to bring existing applications into the production environment immediately.

  For virtio interfaces, HCG 4.0 supports **vhost-user** transparently by default. This allows QEMU and AVS to share virtio queues through shared memory, resulting in improved performance over standard virtio. The transparent implementation provides a simplified alternative to the open-source AVP kernel and AVP-PMD drivers included with HCG 4.0. The availability of **vhost-user** also offers additional performance enhancements through optional multi-queue support for virtio interfaces.
- For backwards compatibility, Guest OS can still be configured to leverage the open-source Accelerated Virtual Port (AVP-KMOD) drivers available at https://github.com/HCG 4.0. Prior to the availability of vhost-user, AVP-KMOD ports provided increased throughput over the original plain virtio drivers when connected to the AVS.
- For the highest performance, guest applications can be modified to make use of Intel DPDK libraries, and the open-source AVP-PMD poll-mode drivers available at https://github.com/HCG 4.0, to connect with the AVS Switch.

In addition to AVS, HCG 4.0 incorporates DPDK-Accelerated Neutron Virtual Router L3 Forwarding (AVR). Accelerated forwarding is used for directly attached tenant networks and subnets, as well as for gateway, SNAT, DNAT, and floating IP functionality.

HCG 4.0 also supports direct guest access to NICs using PCI passthrough or SR-IOV, with enhanced NUMA scheduling options compared to standard OpenStack. This offers very high performance, but because access is not managed by HCG 4.0 or the vSwitch process, there is no support for live migration, HCG 4.0-provided LAG, host interface monitoring, QoS, or ACL. If VLANs are used, they must be managed by the guests.

For further performance improvements, HCG 4.0 supports direct access to PCI-based hardware accelerators, such as the Coleto Creek encryption accelerator from Intel. HCG 4.0 manages the allocation of SR-IOV VFs to VMs, and provides intelligent scheduling to optimize NUMA node affinity.

The Ethernet MTU

The Maximum Transmission Unit (MTU) of an Ethernet frame is a configurable attribute in HCG 4.0. Changing its default size must be done in coordination with other network elements on the Ethernet link.

In the context of HCG 4.0, the Maximum Transmission Unit (MTU) refers to the largest possible payload on the Ethernet frame on a particular network link. The payload is enclosed by the Ethernet header (14 bytes) and the CRC (4 bytes), resulting in an Ethernet frame that is 18 bytes longer than the MTU size.

The original IEEE 802.3 specification defines a valid standard Ethernet frame size to be from 64 to 1518 bytes, accommodating payloads ranging in size from 46 to 1500 bytes. Ethernet frames with a payload larger than 1500 bytes are considered to be jumbo frames.

For a VLAN network, the frame also includes a 4-byte VLAN ID header, resulting in a frame size 22 bytes longer than the MTU size.

For a VXLAN network, the frame is either 54 or 74 bytes longer, depending on whether IPv4 or IPv6 protocol is used. This is because, in addition to the Ethernet header and CRC, the payload is enclosed by an IP header (20 bytes for Ipv4 or 40 bytes for IPv6), a UDP header (8 bytes), and a VXLAN header (8 bytes).

In HCG 4.0, you can configure the MTU size for the following interfaces and networks:

- The management and OAM network interfaces on the controller. The MTU size for these interfaces is set during initial installation; for more information, see *Helion OpenStack Carrier Grade 4.0 Installation: The Controller Configuration Script*. To make changes after installation, see *Helion OpenStack Carrier Grade 4.0 System Administration: Changing the MTU of the OAM Interface*.
- Data interfaces on compute nodes. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Changing the MTU for a Data Interface*.
- Provider networks. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Provider Networks on page 36*.

In all cases, the default MTU size is 1500. The minimum value is 576, and the maximum is 9216.

NOTE: You cannot change the MTU for an infrastructure interface. The value from the network resource is always used.

Because data interfaces are defined over physical interfaces connecting to provider networks, it is important that you consider the implications of modifying the default MTU size:

- The MTU sizes for a data interface and the corresponding Ethernet interface on the edge router or switch must be compatible. You must ensure that each side of the link is configured to accept the maximum frame size that can be delivered from the other side. For example, if the data interface is configured with a MTU size of 9216 bytes, the corresponding switch interface must be configured to accept a maximum frame size of 9238 bytes, assuming a VLAN tag is present.

  The way switch interfaces are configured varies from one switch manufacturer to another. In some cases you configure the MTU size directly, while in some others you configure the maximum Ethernet frame size instead. In the latter case, it is often unclear whether the frame

size includes VLAN headers or not. In any case, you must ensure that both sides are configured to accept the expected maximum frame sizes.

- For a VXLAN network, the additional IP, UDP, and VXLAN headers are invisible to the data interface, which expects a frame only 18 bytes larger than the MTU. To accommodate the larger frames on a VXLAN network, you must specify a larger nominal MTU on the data interface. For simplicity, and to avoid issues with stacked VLAN tagging, some third party vendors recommend rounding up by an additional 100 bytes for calculation purposes. For example, to attach to a VXLAN provider network with an MTU of 1500, a data interface with an MTU of 1600 is recommended.
- A provider network can only be associated with a compute node data interface with an MTU of equal or greater value.
- The MTU size of a compute node data interface cannot be modified to be less than the MTU size of any of its associated provider networks.
- The MTU size of a provider network is automatically propagated to new tenant networks. Changes to the provider network MTU are *not* propagated to existing tenant networks.
- The Neutron L3 and DHCP agents automatically propagate the MTU size of their networks to their Linux network interfaces.
- The Neutron DHCP agent makes the option **interface-mtu** available to any DHCP client request from a virtual machine. The request response from the server is the current interface's MTU size, which can then be used by the client to adjust its own interface MTU size.
- The AVS prevents any AVP-Kernel or AVP-DPDK instances from setting a link MTU size that exceeds the maximum allowed on the corresponding tenant network. No such verification is available for virtio VM instances.

Shared (VLAN) Ethernet Interfaces

The management, OAM, infrastructure, and data networks can share Ethernet or aggregated Ethernet interfaces using VLAN tagging.

The OAM, infrastructure, and data networks can use VLAN tagging, allowing them to share an Ethernet or aggregated Ethernet interface with other networks. The internal management network can also be implemented as a VLAN-tagged network on the interface used for PXE booting.

NOTE: You cannot configure a data VLAN or add a data network on an aggregated Ethernet interface.

For a system using all four networks, the following arrangements are possible:

- One interface for the internal management network, another interface for the OAM network, a third for the infrastructure network, and one or more additional interfaces for data networks.
- One interface for the internal management network, and a second interface for either the OAM or infrastructure network, with the remaining networks implemented using VLAN tagging on either interface.

- One interface for the internal management network, and a second carrying the OAM and infrastructure networks, both implemented using VLAN tagging, with data networks implemented on either or both interfaces using VLAN tagging.
- One interface for the internal management network, with the OAM, infrastructure, and data networks also implemented on it using VLAN tagging.

NOTE: Data networks implemented using VLAN tagging are not compatible with VLAN-based provider networks. (Stacked VLANs are not supported.). To support VLAN provider networks, you can configure a data network on a management or infrastructure interface by editing the interface and selecting both types of network, and then selecting the VLAN provider network. For more information, see the network type discussion in *Helion OpenStack Carrier Grade 4.0 System Administration: Interface Settings*.

Options to share an interface using VLAN tagging are presented during the configuration controller script. To attach an interface to other networks after configuration, you can edit the interface.

For more information about configuring VLAN interfaces, see *Helion OpenStack Carrier Grade 4.0 System Administration: Configuring VLAN Interfaces*

Interface Configuration Scenarios

HCG 4.0 supports the use of consolidated interfaces for the management, infrastructure, OAM, and data networks.

Interface Configuration Scenarios

Some typical configurations are shown in the following table. For best performance, HCG 4.0 recommends dedicated interfaces.

LAG is optional in all instances.

Scenario	Controller	Storage	Compute
Extra Small	2x 1GE LAG: • Mgmt (untagged) • OAM (tagged)	N/A	2x 10GE LAG: • Mgmt (untagged) • OAM (tagged)
Small	2x 10GE LAG: • Mgmt (untagged) 2x 1GE LAG: • OAM (tagged)	N/A	2x 10GE LAG: • Mgmt (untagged) 2x 10GE LAG: • Data (tagged) [… more data interfaces …]
Medium	2x 10GE LAG:	2x 10GE LAG: • Mgmt (untagged) • Infra (tagged)	2x 10GE LAG:

Scenario	Controller	Storage	Compute
	• Mgmt (untagged) • Infra (tagged) 2x 1GE LAG: • OAM (untagged)		• Mgmt (untagged) • Infra (tagged) 2x 10GE LAG • Data (tagged) [… more data interfaces …]
Large	2x 1GE LAG: • Mgmt (untagged) 2x 10GE LAG: • Infra (tagged) 2x 1GE LAG: • OAM (untagged)	2x 1GE LAG • Mgmt (untagged) 2x 10GE LAG: • Infra (tagged)	2x 1GE LAG • Mgmt (untagged) 2x 10GE LAG: • Infra (tagged) 2x 10GE LAG: • Data (tagged) [… more data interfaces …]

Virtual or Cloud Networks

In addition to the physical networks used to connect the HCG 4.0 hosts, HCG 4.0 uses virtual networks to support VMs.

Virtual networks, which include provider networks and tenant networks, are defined and implemented internally. They are connected to system hosts and to the outside world using physical networks attached to data interfaces on compute nodes.

Each physical network supports one or more provider networks, which may be implemented as a flat, VLAN, or VXLAN network. The provider networks in turn support tenant networks, which are allocated for use by different tenants and their VMs, and which can be isolated from one another.

Provider Networks

Provider networks are used to attach data interfaces.

There are no specific requirements for network services to be available on the provider network. However, you must ensure that all network services required by the guests running in the compute nodes are available. For configuration purposes, the compute nodes themselves are entirely served by the services provided by the controller nodes over the internal management network.

Provider Network Planning

Provider networks are the payload-carrying networks used implicitly by end users when they move traffic over their tenant networks.

You can review details for existing provider networks using the web administration interface or the CLI. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Displaying Provider Network Information*.

When planning provider networks, you must consider the following guidelines:

- From the point of view of the tenants, all networking happens over the tenant networks created by them, or by the **admin** user on their behalf. Tenants are not necessarily aware of the available provider networks. In fact, they cannot create tenant networks over provider networks not already accessible to them. For this reason, the system administrator must ensure that proper communication mechanisms are in place for tenants to request access to specific provider networks when required.
  For example, a tenant may be interested in creating a new tenant network with access to a specific network access device in the data center, such as an access point for a wireless transport. In this case, the system administrator must create a new tenant network on behalf of the tenant, using a VLAN ID in the provider network's segmentation range that provides connectivity to the said network access point.
- Consider how different offerings of bandwidth, throughput commitments, and class-of-service, can be used by your users. Having different provider network offerings available to your tenants enables end users to diversify their own portfolio of services. This in turn gives the HCG 4.0 administration an opportunity to put different revenue models in place.
- For the IPv4 address plan, consider the following:
  - Tenant networks attached to a public network, such as the Internet, have to have external addresses assigned to them. Therefore you must plan for valid definitions of their IPv4 subnets and default gateways.
  - As with the OAM network, you must ensure that suitable firewall services are in place on any tenant network with a public address.
- Segmentation ranges defined on a provider network may be owned by the administrator, a specific tenant, or may be shared by all tenants. With this ownership model:
  - A base deployment scenario has each compute node using a single data interface defined over a single provider network. In this scenario, all required tenant networks can be instantiated making use of the available VLANs or VNIs in each corresponding segmentation range. You may need more than one provider network when the underlying physical networks demand different MTU sizes, or when boundaries between provider networks are dictated by policy or other non-technical considerations.
  - Segmentation ranges can be reserved and assigned on-demand without having to lock and unlock the compute nodes. This facilitates day-to-day operations which can be performed without any disruption to the running environment.
- In some circumstances, provider networks can be configured to support VLAN Transparent mode on tenant networks. In this mode VLAN tagged packets are encapsulated within a provider network segment without removing or modifying the guest VLAN tag. For more information, see *Helion OpenStack Carrier Grade 4.0 Tenant User's Guide: VLAN Transparent*.

Tenant Networks

Tenant networks are logical networking entities visible to tenant users, and around which working network topologies are built.

Tenant networks need support from the physical layers to work as intended. This means that the access L2 switches, providers' networks, and data interface definitions on the compute nodes, must all be properly configured. In particular, when using provider networks of the VLAN or VXLAN type, getting the proper configuration in place requires additional planning.

For provider networks of the VLAN type, consider the following guidelines:

- All ports on the access L2 switches must be statically configured to support all the VLANs defined on the provider networks they provide access to. The dynamic nature of the cloud might force the set of VLANs in use by a particular L2 switch to change at any moment.
- The set of VLANs used by each compute node is not fixed; it changes over time. The current VLAN set in use is determined by the configuration details of the tenant networks, and the scheduling on the compute nodes of the virtual machines that use them. This information is provided to the Neutron's AVS plugin, which then uses it to configure the AVS as required.

  When a tenant creates a new network, the Neutron segmentation allocation strategy is to look first for an available segmentation identifier owned by the tenant. If none is available, the search continues over the available shared segmentation identifiers. The allocation process returns an error if no segmentation identifiers are available.

  The VLAN ID assigned to a tenant network is fixed for as long as the tenant network is defined in the system. If for some reason the VLAN ID has to change, the tenant network must be deleted and recreated again.
- Configuring a tenant network to have access to external networks (not just providing local networking) requires the following elements:
  - A physical router, and the provider network's access L2 switch, must be part of the same Layer-2 network. Because this Layer 2 network uses a unique VLAN ID, this means also that the router's port used in the connection must be statically configured to support the corresponding VLAN ID.
  - The router must be configured to be part of the same IP subnet that the tenant network is intending to use.
  - When configuring the IP subnet, the tenant must use the router's port IP address as its external gateway.
  - The tenant network must have the **external** flag set. Only the **admin** user can set this flag when the tenant network is created.

For provider networks of the VXLAN type, consider the following guidelines:

- Layer 3 routers used to interconnect compute nodes must be multicast-enabled, as required by the VXLAN protocol.
- To minimize flooding of multicast packets, IGMP and MLD snooping is recommended on all Layer 2 switches. The AVS switch supports IGMP V1, V2 and V3, and MLD V1 and V2.
- To support IGMP and MDL snooping, Layer 3 routers must be configured for IGMP and MDL querying.

- To accommodate VXLAN encapsulation, the MTU values for Layer 2 switches and compute node data interfaces must allow for additional headers. For more information, see *HCG 4.0 Planning: The Ethernet MTU on page 33*.
- To participate in a VXLAN network, the data interfaces on the compute nodes must be configured with IP addresses, and with route table entries for the destination subnets or the local gateway. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Managing Data Interface Static IP Addresses Using the CLI*, and *Helion OpenStack Carrier Grade 4.0 System Administration: Adding and Maintaining Routes for a VXLAN Network*.

In some circumstances, tenant networks can be configured to use VLAN Transparent mode, in which VLAN tagged packets from the guest are encapsulated within a provider network segment (VLAN) without removing or modifying the guest VLAN tag. For more information, see *Helion OpenStack Carrier Grade 4.0 Server Tenant User's Guide: VLAN Transparent*. Alternately, guest VLAN-tagged traffic can be supported by HCG 4.0 tenants explicitly defining one or more VLAN-tagged IP subnets on a single tenant network. With this approach, the guest VLAN-tagged IP subnets have access to all of the services of the virtualized network infrastructure, such as DHCP, virtual routing, meta-data server, etc. For more information, see *Helion OpenStack Carrier Grade 4.0 Introduction: Helion OpenStack Carrier Grade 4.0 Overview*.

Tenant Network Planning

In addition to the standard considerations for OpenStack tenant networks, such as the ability to designate managed and unmanaged subnets, tenant network planning on a HCG 4.0 system can take advantage of the architecture and extended capabilities of HCG 4.0.

These capabilities include AVS or vSwitch accelerated virtual switching, support for VXLANs, support for SR-IOV and PIC passthrough interfaces, the ability to implement Guest VLANs, options to use virtio or AVP interface drivers on VMs, and so forth.

For more details on tenant networks, see *HCG 4.0 Engineering Guidelines: Networking dimensioning*.

Managed and Unmanaged Subnets

Use the **System Managed Subnet** and **Enable DHCP** subnet attributes to determine how IP addresses are allocated and offered on an IP subnet.

With the proper configuration in place, DHCP services can be provided by the built-in Neutron DHCP server, by a standalone server available from an external network infrastructure, or by both.

When creating a new IP subnet for a tenant network you can specify the following attributes:

System Managed Subnet

When this attribute is enabled, the subnet is *system managed*. The Neutron service automatically allocates an IP address from the address allocation pools defined for the subnet to any new virtual machine (VM) instance with a virtual Ethernet interface attached to the tenant network. Once allocated, the pair (MAC address, IP address) is registered in the Neutron database as part of the overall registration process for the new virtual machine.

When the system managed subnet attribute is disabled, the subnet is *unmanaged*. No automatic allocation of IP addresses takes place, and the Neutron DHCP service for the subnet is disabled. Allocation of IP addresses for new virtual machines must be done at boot time using the CLI or the API interfaces.

Enable DHCP

When this attribute is enabled, a virtual DHCP server becomes available when the subnet is created. It uses the (MAC address, IP address) pairs registered in the Neutron database to offer IP addresses in response to DHCP discovery requests broadcast on the subnet. DHCP discovery requests from unknown MAC addresses are ignored.

The Neutron DHCP server can only be enabled on system managed subnets. DHCP services for unmanaged subnets, if required, must be provisioned by external, non-Neutron, DHCP servers.

When the DHCP attribute is disabled, all DHCP and DNS services, and all static routes, if any, must be provisioned externally.

Allocation Pools

This a list attribute where each element in the list specifies an IP address range, or address pool, in the subnet address space that can be used for dynamic offering of IP addresses. By default there is a single allocation pool comprised of the entire subnet's IP address space, with the exception of the default gateway's IP address.

An external, non-Neutron, DHCP server can be attached to a system managed subnet to support specific deployment needs as required. For example, it can be configured to offer IP addresses on ranges outside the Neutron allocation pools to service physical devices attached to the tenant network, such as testing equipment and servers.

Allocation pools can only be specified on system managed subnets.

The method used to access the server depends on your deployment scenario.

- If the VM is attached to a tenant network with a virtual router then the metadata server is reachable using the virtual router as the route gateway.
- If the VM instance is attached to multiple tenant networks, each with access to a virtual router, then any of the virtual routers provides access to the metadata server. However, installing multiple default routes on the VM might impact the VM's ability to route packets back to external networks.
- If the VM is attached to a tenant network that does not have a virtual router then the a route to the metadata server can be retrieved from the Neutron DHCP service, provided that this service is enabled. If DHCP is enabled on a tenant network, then the VM can use DHCP option 121, Classless Static Route, to retrieve the metadata server static route information, which uses the DHCP server's address as the gateway.

  Note that when using DHCP option 121, the answer from the DHCP server will also include other applicable static routes. They include:

  - any static routes configured when the IP subnet was created
  - default route to the subnet gateway IP address, if one is configured on the subnet
  - on-link routes for all other subnets on the same network and same guest VLAN

The DHCP service only responds to option 121 is there are no virtual routers on the network.

The following requirements must be satisfied in order for a guest application to access the metadata service:

- There is a route table entry to route traffic destined to the 169.254.169.254 address via a Neutron router, or via a suitable static route to the 169.254.169.254 address.
- The metadata server knows about the virtual machine instance associated with the MAC and IP addresses of the virtual Ethernet interface issuing the metadata service request. This is

necessary for the metadata server to be able to validate the request, and to identify the virtual machine's specific data to be returned.

On system managed subnets, the Neutron service has all address information associated with the virtual machines in its database.

On unmanaged subnets, you must tell the Neutron service the IP address of the network interface issuing the metadata service requests.

Virtual Routers

Virtual routers provide internal and external network connectivity for tenants.

The user associated with a tenant can add a designated number of virtual routers (Neutron routers) to the tenant. The maximum number is specified in the quotas for the tenant.

The virtual router automatically provides a connection to system services required by the tenant, such as the metadata service that supplies instances with user data. You can configure the virtual routers with interfaces to tenant networks to provide internal and external connectivity.

A virtual router can be implemented as a centralized router, or a distributed virtual router (DVR).

Only the **admin** user can specify a distributed router. For other tenants, this choice is not available, and a centralized router is implemented by default. The **admin** user can change a centralized router to a distributed router on behalf of other tenants (see *Helion OpenStack Carrier Grade 4.0 Tenant User's Guide: Virtual Router Administration*).

Centralized

A centralized virtual router is instantiated on a single compute node. All traffic using the router must pass through the compute node.

Distributed

A distributed virtual router is instantiated in a distributed manner across multiple hosts. Distributed virtual routers provide more efficient routing than standard virtual routers for east-west (tenant-to-tenant) or floating IP address traffic. Local traffic is routed within the local host, while external L3 traffic is routed directly between the local host and the gateway router.

To implement the distributed model, a centralized-portion of the router is still deployed on one host. The centralized portion manages north-south (external network) traffic and source network address translation (SNAT) traffic, as well as agents deployed on other hosts. The agents offload the centralized router for east-west (tenant-to-tenant) routing and floating IP network address translation.

In some cases, the use of virtual routers on the tenant networks can result in multiple default routes for a virtual machine (VM). If this happens, you can establish alternative VM access to the metadata server; see #unique_61/unique_61_Connect_42_accessing_metadata_server on page 40.

You can enable SNAT on a virtual router. For more information, see *Helion OpenStack Carrier Grade 4.0 Tenant User's Guide: Configuring SNAT on a Virtual Router*.

You can also enable DNAT on a virtual router. For more information, see *Helion OpenStack Carrier Grade 4.0 Tenant User's Guide: Configuring Port-based DNAT on a Virtual Router*.

To add a virtual router, see *Helion OpenStack Carrier Grade 4.0 Tenant User's Guide: Adding Virtual Routers*. To create interfaces, see *HCG 4.0 Cloud Administration: Adding Virtual Router Interfaces*.

DiffServ-Based Priority Queuing and Scheduling

Differentiated Services, or DiffServ, is a packet-based traffic management mechanism that allows end devices to specify an expected per-hop behavior (PHB) from upstream switches and routers when handling their traffic during overload conditions.

DiffServ marking and architecture are IEEE specifications, IEEE RFC 2474 and RFC 2475.

Support for DiffServ is implemented on the AVS for all input and output ports, on all attached tenant networks. On each port, it uses eight queues associated with the CS0 to CS7 DiffServ class selectors, which are processed by a round-robin scheduler with weights of 1, 1, 2, 2, 4, 8, 16, and 32. Overflow traffic is tail-drop discarded on each queue. Guest applications in the HCG 4.0 cluster can set a Differentiated Services Code Point (DSCP) value in the Differentiated Service (DS) field of the IP header to mark the desired upstream PHB.

On ingress, a packet being processed to be sent to the VM is directed to the appropriate DiffServ output queue. On overflow, that is, if the virtual machine cannot keep up with the incoming traffic rate, lower priority packets are discarded first.

On egress, a packet sent from the virtual machine (VM) to the AVS is first enqueued into the appropriate DiffServ input queue according to the specified DSCP value, the CS0 queue being the default. On overload, that is, if the AVS cannot keep up with the incoming traffic, lower priority packets are discarded first; in this case, you should consider re-engineering the AVS, possibly assigning it more processing cores. Once serviced by the DiffServ class scheduler, the packet is processed according to the QoS policy in place for the tenant network, as described in *Helion OpenStack Carrier Grade 4.0 Tenant User's Guide: Quality of Service Policies*.

Quality of Service Policies Support

Quality of Service (QoS) policies specify relative packet processing priorities applied by the AVS switch on each compute node to incoming tenant network's traffic during overload conditions.

The QoS polices play no role under normal traffic loads, when no input traffic queues in the AVS are close to their overflow limits.

QoS policies are created by the cluster administrator, and selected by the tenant users to apply on a per-tenant network basis.

VXLANs

You can use Virtual eXtensible Local Area Networks (VXLANs) to connect VM instances across non-contiguous Layer 2 segments (that is, Layer 2 segments connected by one or more Layer 3 routers).

A VXLAN is a Layer 2 overlay network scheme on a Layer 3 network infrastructure. Packets originating from VMs and destined for other VMs are encapsulated with IP, UDP, and VXLAN headers and sent as Layer 3 packets. The IP addresses of the source and destination compute nodes are included in the headers.

Guest VLANs

Use guest VLANs to segregate IP traffic from a single virtual Ethernet interface on a virtual machine into dedicated VLANs. Together with the capability to define multiple IP subnets on a single tenant network, guest VLANs facilitate the transitioning of existing guest applications to run on the HCG 4.0 virtualized network environment.

Guest VLANs are useful when guest applications rely on the capability to configure a single virtual Ethernet interface with multiple, probably overlapping, IP addresses. From the point of

view of the guest, this is done by defining VLAN Ethernet interfaces, and associating one or more IP addresses to them. If implementing overlapping IP addresses, typically in support of VPN applications, the guest must use different VLAN IDs to separate traffic from different VPNs.

For example, on a Linux guest, the virtual interfaces eth0.10:1, eth0.10:2, and eth0.20 refer to the same eth0 physical interface with two virtual interfaces on VLAN ID 10, and a single virtual interface on VLAN ID 20. A common scenario in a VLAN application is to allocate distinct IP addresses from the same IP subnet to virtual interfaces on the same VLAN. In this example, eth0.10:1 and eth0.10:2 could be assigned distinct IP addresses from the subnet 192.168.1.0/24, and eth0.20 an address from the subnet 192.168.2.0/24. In the case of a VPN application, overlapping IP addresses are allowed to exist on eth0.20 and either eth0.10:1 or eth0.10:2.

HCG 4.0 supports these deployment scenarios with the help of guest VLANs which enable the transport of IP subnets traffic over VLAN-tagged Ethernet frames. To support the example above, a tenant user would define the following two IP subnets, both on the same tenant network, using guest VLAN IDs as follows:

- Subnet 192.168.1.0/24 with guest VLAN ID set to 10
- Subnet 192.168.2.0/24 with guest VLAN ID set to 20

The subnet-to-VLAN_ID mapping can be one-to-one, as in this example, or many-to-one. This means that tenant users can use a single VLAN ID of their choice to encapsulate traffic from one or more IP subnets.

Alternately, tenant networks can be configured to use VLAN Transparent mode, in which VLAN tagged guest packets are encapsulated within a provider network segment without removing or modifying the guest VLAN tag. For more information, see *Helion OpenStack Carrier Grade 4.0 Tenant User's Guide: VLAN Transparent*.

Guest VLAN Implementation

Guest VLANs are implemented using available segmentation ranges from suitable provider networks, just as it is done when new tenant networks are created. Therefore all network design considerations regarding the configuration of L2 switches and the Neutron allocation strategy, described in Tenant Networks on page 38, must be taken into consideration.

Additionally, note that the AVS will silently discard incoming VLAN-tagged VM traffic with unknown VLAN IDs, that is, with VLAN IDs not defined as guest VLANs on the particular tenant network.

VM Network Interface Options

HCG 4.0 supports a variety of standard and performance-optimized network interface drivers in addition to the standard OpenStack choices.

- Unmodified guests can use Linux networking and virtio drivers. This provides a mechanism to bring existing applications into the production environment immediately.

  For virtio interfaces, HCG 4.0 supports **vhost-user** transparently by default. This allows QEMU and AVS to share virtio queues through shared memory, resulting in improved performance over standard virtio. The transparent implementation provides a simplified alternative to the open-source AVP kernel and AVP-PMD drivers included with HCG 4.0. The availability of **vhost-user** also offers additional performance enhancements through optional multi-queue support for virtio interfaces.

NOTE: The virtio devices on a VM cannot use vhost-user for enhanced performance if either of the following is true:

- The VM is not backed by huge pages.
- The VM is live-migrated from an older platform that does not support vhost-user.

- For backwards compatibility, Guest OS can still be configured to leverage the open-source Accelerated Virtual Port (AVP-KMOD) drivers available at https://github.com/Wind-River/. Prior to the availability of vhost-user, AVP-KMOD ports provided increased throughput over the original plain virtio drivers when connected to the AVS.
- For the highest performance, guest applications can be modified to make use of Intel DPDK libraries, and the open-source AVP-PMD poll-mode drivers available at https://github.com/Wind-River/titanium-server, to connect with the AVS Switch.

Address Filtering on Virtual Interfaces

The AVS on compute nodes can be configured to filter out packets based on the source MAC address.

MAC addresses for virtual network interfaces on virtual machines are dynamically allocated by the system. For most scenarios, the assigned MAC addresses are expected to be used on all outgoing packets from the virtual machine instances. However, there are scenarios where the source MAC address is not expected to match the original assignment, such as when a L2 switch is implemented internally on the virtual machine.

By default, the AVS on compute nodes accepts any source MAC address on the attached virtual network interfaces. However, it can be configured to filter out all incoming packets with non-system-generated source MAC address, if required. When evaluating the use of the filtering capability, you must consider the following:

- Source MAC address filtering can be enabled and disabled by the administrator user only, not by tenants.
- Filtering is enabled on a per-tenant basis only. Higher granularity, such as per-instance filtering, is not supported.
- When enabled, source MAC address filtering applies to all new virtual interfaces created by the Neutron service. Address filtering is not active on virtual interfaces created before filtering is enabled.

You can use the following CLI command to enable source MAC address filtering:

```
~(keystone_admin)$ neutron setting-update --tenant-id=<TENANTID> \
--mac-filtering={True|False}
```

PCI Passthrough Ethernet Interfaces

A passthrough Ethernet interface is a physical PCI Ethernet NIC on a compute node to which a virtual machine is granted direct access.

This minimizes packet processing delays but at the same time demands special operational considerations.

For all purposes, a PCI passthrough interface behaves as if it were physically attached to the virtual machine. Therefore, any potential throughput limitations coming from the virtualized environment, such as the ones introduced by internal copying of data buffers, are eliminated. However, by bypassing the virtualized environment, the use of PCI passthrough Ethernet devices introduces several restrictions that you must take into consideration. They include:

- no support for LAG, QoS, ACL, or host interface monitoring
- no support for live migration
- no access to the compute node's AVS switch

SR-IOV Ethernet Interfaces

A SR-IOV Ethernet interface is a physical PCI Ethernet NIC that implements hardware-based virtualization mechanisms to expose multiple virtual network interfaces that can be used by one or more virtual machines simultaneously.

The PCI-SIG Single Root I/O Virtualization and Sharing (SR-IOV) specification defines a standardized mechanism to create individual virtual Ethernet devices from a single physical Ethernet interface. For each exposed virtual Ethernet device, formally referred to as a *Virtual Function* (VF), the SR-IOV interface provides separate management memory space, work queues, interrupts resources, and DMA streams, while utilizing common resources behind the host interface. Each VF therefore has direct access to the hardware and can be considered to be an independent Ethernet interface.

4

Storage Planning

Storage Planning

HCG 4.0 uses storage resources on the controller and compute hosts, as well as on storage hosts if they are present.

The storage resources are related to system requirements and the type of storage being configured.

For detailed storage requirement calculations, refer to the *HCG 4.0 System Engineering Guidelines*.

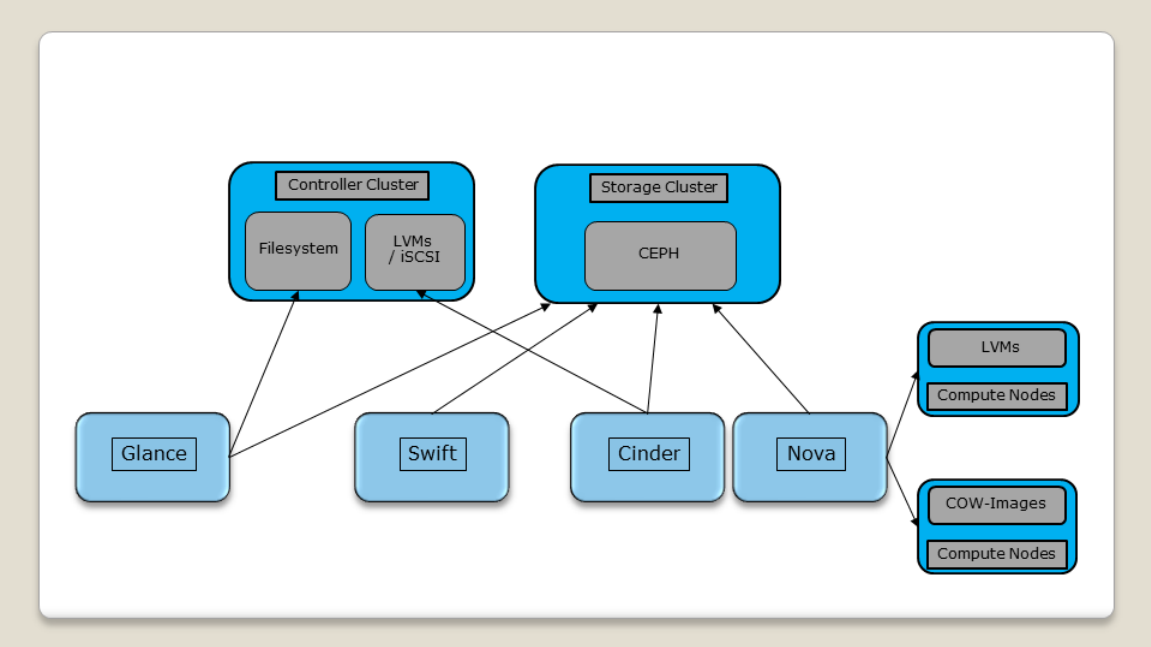
Storage Flexibility on HCG 4.0

HCG 4.0 provides a range of storage options for flexibility and for scaling. Multiple storage services are supported simultaneously, including:

- Cinder
- Nova
- Swift
- Glance

The figure below shows the storage options and backends.

Figure 8: HCG 4.0 Storage Options and Backends



As shown in the figure above, the backends for each of these storage services include:

- Cinder - persistent HA-protected block storage for virtual disks
  - Backends:
    - LVM on Controller Nodes
    - CEPH on Storage Nodes
    - Remote EMC SAN Cluster
- Nova - ephemeral block storage for virtual disks
  - Backends:
    - LVM on Compute Nodes
    - COW-Image on Compute Nodes
    - CEPH on Storage Nodes
- Glance
  - Backends:

  - LVM
  - CEPH
- Swift
  - Backends:
    - CEPH

Related Links

Storage Tab

Storage on Controller Hosts

Controller hosts provide storage for the system database, and for system backup operations. On systems with controller storage, they also provide persistent storage for virtual machine images, using a secondary disk.

Controller storage is configured initially at installation, using the controller configuration script. To utilize the maximum available space on the storage media, HCG 4.0 recommends that you use the default settings. You can change the allocations at any time after installation. Controller hosts provide the following types of storage:

Database storage

the storage allotment for the OpenStack database

Image storage

for a system that provides LVM-backed controller storage for VMs, the size of the partition to use for image storage

Backup storage

the storage allotment for backup operations

Volume storage

for a system that provides LVM-backed controller storage for VMs, the storage allotment for all Cinder volumes used by guest instances; also called *Cinder storage*

Image Conversion Space

the storage allotment for image caching and temporary image conversion

Ceph Mon Storage

for a system using Ceph storage, the storage allotment on the controller for Ceph monitoring

NOTE: For clusters using a Ceph backend, volume storage and image storage are allotted on storage nodes, not on the controller node. To change the Cinder volume storage for a Ceph backend, see *Helion OpenStack Carrier Grade 4.0 System Administration: Replacing Storage Node Hardware*.

The storage allotments are configured initially during software installation. You can change them using the Web administration interface or the CLI. For more information, see *THelion OpenStack Carrier Grade 4.0 System Administration: Increasing Storage Space Allotments on the Controller*).

To accommodate changes, there must be enough disk space on the controller, including headroom needed to complete the operation. The headroom required is 45 GiB on the primary disk for a cluster using controller storage with an LVM backend, or 65 GiB for a cluster using dedicated storage with a Ceph backend. This is in addition to the space required for any new allotments. The requested changes are checked against available space on the affected disks; if there is not enough, the changes are disallowed.

To provide more space, you can replace the affected disk or disks. Database, image, and backup storage use space on the primary disk. Cinder volume storage (on a cluster with an LVM backend) uses space on a disk selected by device node number during controller configuration. The replacement disk must occupy the same device node number. Changes to the Cinder volume storage can also affect the primary disk because of the headroom requirement.

Ceph monitor storage uses the primary disk by default. If a Ceph backend is added to an existing system, and there is insufficient space on the primary disk for the requested Ceph monitor storage, you can specify a secondary disk. Note that this requires a controller re-installation.

To pass the disk-space checks, any replacement disks must be installed before the allotments are changed.

Storage for System Use

Internal database storage is provided using DRBD-synchronized partitions on the controller primary disks. The size of the database grows with the number of system resources created by the system administrator and the tenants. This includes objects of all kinds such as compute nodes, provider networks, images, flavors, tenant networks, subnets, virtual machine instances, and NICs. As a reference point, consider the following deployment scenario:

- two controllers
- four compute nodes with dual Intel(R) Xeon(R) CPU E5-2670 v2 @ 2.50GHz each.
- 40 virtual machine instances
- 120 tenant networks
- steady collection of power management statistics

The size of the database in this case is approximately 9 GB. With a suggested default of 20 GB, there is still plenty of room to grow. However, you should periodically monitor the size of the database to ensure that it does not become a bottleneck when delivering new services.

For more information, see *HCG 4.0 Engineering Guidelines*.

Storage on Compute Hosts

Compute hosts provide local ephemeral storage for virtual machine (VM) disks.

This is the default type of storage for VM swap and ephemeral disks, and for boot-from-image root disks; you must configure this storage at installation before you can unlock a compute host. You can change the configuration after installation; for more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Adjusting Resources on a Compute Node*. Note that live migration is not always supported for VM disks using this type of storage. For more information, seeVM Storage Settings for Migration, Resize, or Evacuation on page 54 .

On compute nodes, dedicated local storage space is required by the **nova** service. For flexibility and scalability, this space is implemented as a local volume group, called **nova-local**. The group can include one or more non-root disks as physical resources. Depending on whether LVM-backed or CoW-image-backed local storage is configured on the compute host, **nova-local** contains one or more volumes.

NOTE: As an alternative to LVM-backed local storage, compute hosts can be configured to offer image-backed local storage. For more information, see *Helion OpenStack Carrier Grade 4.0* System *Administration: Managing Local Volume Groups*.

The first volume in **nova-local** is called the *instances logical volume*, or **Instances LV**. It contains the **/etc/nova/instances** file system, and is used for the following:

- the nova image cache, containing images downloaded from Glance
- various small nova control and log files, such as the **libvirt.xml** file, which is used to pass parameters to **libvirt** at launch, and the **console.log** file
- on a host configured for CoW-image-backed local storage, the CoW image files that constitute the local disks for VMs

For CoW-image-backed local storage, the **Instances LV** is the only volume in **nova-local**. For LVM-backed local storage, additional volumes are required to realize local disks for VMs. To reserve space for these volumes, the size of the **Instances LV** must be appropriately configured.

By default, no size is specified for the **Instances LV**. The minimum required space is 2 GB for a **nova-local** volume group with a total size less that 80 GB, and 5 GB for a **nova-local** volume group larger or equal than 80 GB; you must specify at least this amount. You can allocate more **Instances LV** space to support the anticipated number of boot-from-image VMs, up to 50% of the maximum available storage of the local volume group. At least 50% free space in the volume group is required to provide space for allocating logical volume disks for launched instances. The value provided for the **Instance LV Size** is limited by this maximum.

Instructions for allocating the **Instances LV Size** using the Web administration interface or the CLI are included in *HCG 4.0 Installation* as part of configuring the compute nodes. For the command syntax, see *Helion OpenStack Carrier Grade 4.0 System Administration: Managing Local Volume Groups*.

CAUTION: If less than the minimum required space is available, the compute host cannot be unlocked.

Storage on Storage Hosts

Storage hosts provide persistent and highly available storage for virtual machine (VM) images and disk volumes.

They can also be used to provide remote ephemeral storage for virtual machine disks, making live migration possible for VM ephemeral and swap disks, as well as boot-from-image root disks.

To use storage hosts, a HCG 4.0 system with Ceph-backed storage is required. You can configure this at installation using the controller configuration script. On systems with controller-based storage, you can also add support for Ceph-backed storage later from the CLI; for more

information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Dedicated Storage for Systems Using Controller Storage*.

Storage hosts are paired for redundancy. On a system using Ceph-backed storage, at least one pair is required , and up to four pairs are supported. You can add up to eight object storage devices (OSDs) per storage host for data storage. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Provisioning Storage on a Storage Host*.

Space on the storage hosts must be configured at installation before you can unlock the hosts. You can change the configuration after installation by adding resources to existing storage hosts (see *Helion OpenStack Carrier Grade 4.0 System Administration: Replacing Storage Node Hardware*) or adding more storage hosts (see the installation procedure in the *HCG 4.0 Installation* document that pertains to your HCG 4.0 configuration).

HCG 4.0 creates default Ceph storage pools for images, volumes, ephemeral data, and object data. You can modify the storage pools after installation. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Ceph Storage Pools*.

Storage hosts can achieve faster data access using SSD-backed transaction journals (*journal functions*) or additional hosts in a caching tier configuration, or both. Caching tier hosts overlay fast SSD-backed storage pools over the slower HDD-backed storage pools used in the standard backing tier.

NOTE: SSD-backed journals cannot be used on a storage host assigned to the caching tier.

Journal Functions

Each OSD on a storage host has an associated Ceph transaction journal, which tracks changes to be committed to disk for data storage and replication, and if required, for data recovery. This is a full Ceph journal, containing both metadata and data. By default, it is collocated on the OSD, which typically uses slower but less expensive HDD-backed storage. For faster commits and improved reliability, you can use a dedicated solid-state drive (SSD) installed on the host and assigned as a *journal function*. You can dedicate more than one SSD as a journal function.

NOTE: You can also assign an SSD for use as an OSD, but you cannot assign the same SSD as a journal function.

If a journal function is available, you can configure individual OSDs to use journals located on the journal function. Each journal is implemented as a partition. You can adjust the size and location of the journals. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Storage Functions: OSDs and SSD-backed Journals*.

For OSDs implemented on rotational disks, HCG 4.0 strongly recommends that you use a journal function. For OSDs implemented on SSDs, collocated journals can be used with no performance cost.

Cache Tier

For systems where the same few data objects are accessed frequently, you can improve read-write times by implementing a cache tier. This uses a dedicated set of Ceph-caching storage hosts equipped with SSDs, in addition to a set of Ceph-backing storage hosts. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Cache Tiering*.

NOTE: Since all disks on a caching host are SSDs, journal functions are neither required nor supported on caching-tier hosts.

HCG 4.0's cache tiering support is based on the Ceph cache tiering functionality. To ensure cache tiering is appropriate for your requirements, review the Ceph public documentation for caveats surrounding the use of this feature (http://docs.ceph.com/docs/master/rados/operations/cache-tiering/?highlight=tier#a-word-of-caution).

For valid HCG 4.0 storage cluster configurations when using cache tiering, see *Helion OpenStack Carrier Grade 4.0 System Administration: Valid Storage Cluster Configurations for Cache Tiering*

Block Storage for Virtual Machines

Virtual machines use HCG 4.0 storage resources for root and ephemeral disks. You can

allocate root disk storage for virtual machines using the following:

- a Cinder volume
- ephemeral storage; one of
  - ephemeral local storage on compute nodes, backed by LVM
  - ephemeral local storage on compute nodes, backed by image file
  - ephemeral remote storage on storage nodes, backed by Ceph

The use of a Cinder volume or ephemeral storage is determined by the **Instance Boot Source** setting when an instance is launched. **Boot from volume** results in the use of a Cinder volume, while **Boot from image** results in the use of ephemeral storage

Cinder-backed persistent storage for virtual machines is provided using either Ceph-backed OSD disks on high-availability storage hosts, or LVM-backed, DRBD-synchronized controller secondary disks on systems that do not use storage hosts.

For a controller-based LVM Cinder backend, you can configure HCG 4.0 at installation to use thin or thick provisioning. Thin provisioning allocates space for the volume dynamically on the underlying physical disk. Thick provisioning creates a fixed-size volume. Thin provisioning offers support for fast secure deletion, but requires longer volume creation times. You can improve volume creation times by using SSDs for the underlying disks.

⚠ **CAUTION:** The choice of thick or thin provisioning cannot be changed after installation.

Ephemeral storage for virtual machines, including swap disk storage, ephemeral disk storage, and root disk storage if the **Instance Boot Source** is set to **Boot from Image**, is by default provided locally on the compute nodes where the VMs are instantiated (local ephemeral storage). On Ceph systems, you can change this configuration to use storage node resources instead (remote ephemeral storage).

On each individual compute host, you can configure the ephemeral storage to use:

- a local LVM-based backend, to optimize run-time I/O performance
- a CoW (Copy on Write) sparse-image-format backend, to optimize launch and delete performance
- a Ceph backend (on a system with storage nodes), to optimize migration capabilities

The ephemeral storage type is defined during installation, and can be modified using the Web administration interface or the CLI. For more information, see *Helion OpenStack Carrier Grade 4.0* System *Administration Guide: Managing Local Volume Groups* .

⚠ **CAUTION:** Unlike Cinder-based storage, ephemeral storage does not persist if the instance is terminated or the compute node fails.

In addition, for local ephemeral storage, migration and resizing support depends on the storage backing type specified for the instance, as well as the boot source selected at launch.

The choice of storage type affects migration behavior. For more information, see VM Storage Settings for Migration, Resize, or Evacuation on page 54.

Swift Object Storage

Systems with dedicated storage hosts can provide object storage using OpenStack Swift. VMs and system users can use this to store and exchange Ceph-backed files.

Swift object storage uses *Swift containers*. These are similar to directories in a file system, except that they cannot be nested. However, each Swift container can contain areas called *folders*, which you can use to organize content. Hierarchies of folders are supported.

VMs and OpenStack users (including OpenStack services) can create Swift containers for public or private use, and then access them for file uploads and downloads.

In HCG 4.0, a storage pool implemented on Ceph-backed storage hosts is used to hold Swift containers and objects. The pool is created when the Swift service is started. For VMs, this offers a place to store files that persist independently and can be exchanged with other VMs or with the HCG 4.0 platform.

You can add Swift support from the command line at any time after installation. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration:Configuring Swift Object Storage*.

System administrators and tenant users can manage Swift containers and their contents from the CLI or the Web administration interface. VMs can use a Swift client or the Swift REST API to manage and access Swift containers and objects. For more information, consult the public OpenStack documentation.

VM Storage Settings for Migration, Resize, or Evacuation

The migration, resize, or evacuation behavior for an instance depends on the instance boot configuration and the type of ephemeral storage used.

The following table summarizes the boot and local storage configurations needed to support various behaviors.

Instance Boot Type and Ephemeral and Swap Disks from flavor	Local Storage Backing	Live Migration with Block Migration	Live Migration w/o Block Migration	Cold Migration	Local Disk Resize	Evacuation
From Cinder Volume (no local disks)	N/A	N	Y	Y	N/A	Y
From Cinder Volume (w/ remote Ephemeral and/or Swap)	N/A	N	Y	Y	N/A	Y
From Cinder Volume (w/ local Ephemeral and/or Swap)	LVM	N	N	Y Ephemeral/ Swap data loss	Y Data loss if to new node	Y Ephemeral/ Swap data loss
	CoW	N	N	Y	Y	Y Ephemeral/ Swap data loss
From Glance Image (all flavor disks are local)	LVM	N	N	Y Local disk data loss	Y Data loss if to new node	Y Local disk data loss
	CoW	Y	N	Y	Y	Y Local disk data loss
From Glance Image (all flavor disks are local + attached Cinder Volumes)	LVM	N	N	Y Local disk data loss	Y Data loss if to new node	Y Local disk data loss
	CoW	N	N	Y	Y	Y Local disk data loss

NOTE: The **Local Storage Backing** is a consideration only for instances that use local ephemeral or swap disks.

The boot configuration for an instance is determined by the **Instance Boot Source** selected at launch. For more information, see *Helion OpenStack Carrier Grade 4.0 Tenant User's Guide: Launching Virtual Machine Instances*.

The type of ephemeral-disk storage backing used by an instance is determined by a flavor extra specification. For more information, see *Helion OpenStack Carrier Grade 4.0 System Administration: Specifying the Storage Type for VM Ephemeral Disks*.

5

Installation and Resource Planning

Licensing Requirements

To install and use HCG 4.0, you require a license from HCG 4.0.

HCG 4.0 Evaluation License

You can use an evaluation license to try out HCG 4.0. This is a fully functional, time-limited license, provided for evaluation purposes only.

HCG 4.0 Product License

You can use a product license to operate HCG 4.0 for your business requirements. HCG 4.0

CPE Evaluation License

You can use an evaluation license to try out HCG 4.0 CPE. This is a fully functional, time-limited license, provided for evaluation purposes only.

HCG 4.0 CPE Product License

You can use a HCG 4.0 CPE product license to install HCG 4.0 CPE.

For complete information about HCG 4.0 licenses and licensing terms, contact your HCG 4.0 sales representative.

Obtaining a License

To obtain a license, contact your HCG 4.0 sales representative. The license is sent to the contact email address you provide when you make a request or place an order.

Installing a License

To install a license initially, follow the procedure for installing and configuring the HCG 4.0. During this procedure, you must copy the license file to a designated license directory on the controller host. The default designated directory is **/home/wrsroot**; you can specify a different one during installation.

Updating a License

After you have installed a license, you can update it by copying the new license file to the designated license directory on the active controller host, and then running the **license-install** utility as shown in the following example.

```
$ sudo /usr/sbin/license-install license_file
```

WARNING: HCG 4.0 recommends that you update licenses before they expire. Manual recovery of VMs may be required if the license is upgraded after expiry.

Operational Behavior with Expired or Invalid Licenses

HCG 4.0 and HCG 4.0 CPE assume the operational behavior described below when an expired or invalid license is detected.

Mismatched licenses are considered invalid. For example, a HCG 4.0 license is considered an invalid license if the server is installed and provisioned as a HCG 4.0 CPE product.

Expired and invalid licenses cause the following behavior:

1. A service log (401.003) is logged every 1 hour.
2. A service alarm (400.003) is triggered 8 hours after the expired or invalid license is detected, and every 1 hour thereafter.
3. 72 hours after the expired or invalid license is detected for the first time, the controller stops service.

HTTPS and Certificates

For secure HTTPS access, you require a CA-signed certificate during the installation process.

Enabling HTTPS Access (optional)

To enable secure HTTPS access for REST API applications and the web server, a digital certificate is required during software installation. When secure HTTPS connectivity is chosen, HTTP is disabled.

The use of a CA-signed certificate is strongly recommended. However, for evaluation purposes, you have the option to select a self-signed certificate included with HCG 4.0. You must obtain the certificate and copy it to the controller host before starting the controller configuration script. You can update the certificate at any time after installation.

Installing a digital certificate

To install a CA-signed digital certificate, follow the procedure for installing and configuring the HCG 4.0. During this procedure, you must copy the certificate PEM file to a designated license directory on the controller host. The default designated directory is **/home/wrsroot**; you can specify a different one during installation.

Updating a digital certificate

After you have installed a digital certificate, whether CA-signed or self-signed, you can update it by copying the new certificate to the designated directory on the active controller host, and then running the **https-certificate-install** utility as shown in the following example.

```
$ sudo /usr/sbin/https-certificate-install pem_file
```

HCG 4.0 Hardware Requirements

HCG 4.0 has been tested to work with specific hardware configurations.

If the minimum hardware requirements are not met, system performance cannot be guaranteed.

For system engineering considerations related to hardware, refer to the *HCG 4.0 Engineering Guidelines*.

Controller, Compute, and Storage Hosts

Table 3 **Hardware Requirements — Systems with Dedicated Controllers**

Minimum Requirement	Controller	Storage	Compute
Minimum Qty of Servers	2 (required)	2-8 (if Ceph storage used)	2 – 100
Minimum Processor Class	Dual-CPU Intel® Xeon® E5 26xx Family (SandyBridge) 8 cores/socket		
Minimum Processor Speed	2.5 GHz		1.8 GHz
Minimum Memory	64 GB	64 GB	32 GB
Minimum Primary Disk (two-disk hardware RAID suggested)	500 GB - SSD	120 GB (min. 10K RPM)	
	NOTE: Software RAID is not supported.		

Minimum Requirement	Controller	Storage	Compute
Additional Disks	1x 500 GB (min. 10K RPM)	500 GB (min. 10K RPM) for OSD storage one or more SSDs (recommended for Ceph journals); min. 1024 MiB per journal	500 GB (min. 10K RPM) — 1 or more (for local storage)
Network Ports	(Typical deployment.)		
	• Mgmt: 2 x 10GE LAG • Infra: 2 x 10GE LAG • OAM: 2 x 1GE LAG	• Mgmt: 2 x 10GE LAG • Infra: 2 x 10GE LAG	• Mgmt: 2 x 10GE LAG • Infra: 2 x 10GE LAG • Data: 2 x 1GE LAG
USB Interface	1	not required	
Power Profile	Max Performance Min Proc Idle Power:No C States		
Boot Order	HD, PXE, USB	HD, PXE	
BIOS Mode	BIOS or UEFI **NOTE:** UEFI Secure Boot and UEFI PXE boot over IPv6 are not supported. On systems with an IPv6 management network, you can use a separate IPv4 network for PXE boot. For more information, see The PXE Boot Network on page 21.		
Intel Hyperthreading	Disabled or Enabled		
Intel Virtualization (VTD, VTX)	Disabled		Enabled

Verified Commercial Hardware

Verified and approved hardware components for use with HCG 4.0 are listed in the following table.

Table 4 **Verified Components**

Component	Approved Hardware
Hardware Platforms	• Hewlett-Packard - HP360 Proliant DL360P Gen8 Server - HP360 Proliant DL360P Gen9 Server

Component	Approved Hardware
	- HP380 Proliant DL380P Gen8 Server - HP380 Proliant DL380P Gen9 Server - c7000 Enclosure with HP460 Proliant BL460 Gen9 Server **CAUTION:** LAG support is dependent on the switch cards deployed with the c7000 enclosure. To determine whether LAG can be configured, consult the switch card documentation. • Dell - Dell PowerEdge R720
Supported Reference Platforms	• Intel Iron Pass • Intel Canoe Pass • Intel Grizzly Pass • Intel Wildcat Pass
Disk Controllers	• HP SAS Controllers - P440ar - P420i • LSI 2308 • LSI 3008
NICs Verified for PXE Boot, Management, and OAM Networks	• Intel I210 (Springville) 1G • Intel I350 (Powerville) 1G • Intel 82599 (Niantic) 10G • Intel X540 10G • Intel X710/XL710 (Fortville) 10G • Emulex XE102 10G • Broadcom BCM5719 1G • Broadcom BCM57810 10G
NICs Verified for Infrastructure Network	• Intel 82599 (Niantic) 10G • Intel X540 10G • Intel X710/XL710 (Fortville) 10G

Component	Approved Hardware
	• Emulex XE102 10G • Broadcom BCM57810 10G
NICs Verified for Data Interfaces (Compute Nodes)	The following NICs are supported by DPDK: • Intel I350 (Powerville) 1G • Intel 82599 (Niantic) 10G • Intel X710/XL710 (Fortville) 10 G • Mellanox Technologies - MT27500 Family - [ConnectX-3] 10G/40G The following NICs have been verified in non-accelerated mode for use with data interfaces (Low Speed Integration): • Emulex XE-102 10G • Broadcom (BCM57810) 10G • Broadcom (BCM5719) 1G
PCI passthrough or PCI SR-IOV NICs	• Intel 82599 (Niantic) 10 G • Intel X710/XL710 (Fortville) 10G **NOTE:** The maximum number of VFs per NIC is 32.
PCI SR-IOV Hardware Accelerators	• Intel Coleto Creek 8925/8950 chipset w/ QuickAssist
Board Management Controllers	• HP iLO3 • HP iLO4

Interface Configuration Scenarios

HCG 4.0 supports the use of consolidated interfaces for the management, infrastructure, OAM, and data networks. Some typical configurations are shown in the following table. For best performance, HCG 4.0 recommends dedicated interfaces.

LAG is optional in all instances.

Scenario	Controller	Storage	Compute

Scenario	Controller	Storage	Compute
• Physical interfaces on servers limited to two pairs • Estimated aggregate average VM storage traffic less than 5G	2x 10GE LAG: • Mgmt (untagged) • Infra (tagged) 2x 1GE LAG: • OAM (untagged)	2x 10GE LAG: • Mgmt (untagged) • Infra (tagged)	2x 10GE LAG: • Mgmt (untagged) • Infra (tagged) 2x 10GE LAG • Data (tagged) [… more data interfaces …]
• No specific limit on number of physical interfaces • Estimated aggregate average VM storage traffic greater than 5G	2x 1GE LAG: • Mgmt (untagged) 2x 10GE LAG: • Infra (tagged) 2x 1GE LAG: • OAM (untagged)	2x 1GE LAG • Mgmt (untagged) 2x 10GE LAG: • Infra (tagged)	2x 1GE LAG • Mgmt (untagged) 2x 10GE LAG: • Infra (tagged) 2x 10GE LAG: • Data (tagged) [… more data interfaces …]

Boot Sequence Considerations

During HCG 4.0 software installation, each host must boot from different devices at different times. In some cases, you may need to adjust the boot order.

The first controller node must be booted initially from a removable storage device to install an operating system. The host then reboots from the hard drive.

Each remaining host must be booted initially from the network using PXE to install an operating system. The host then reboots from the hard drive.

To facilitate this process, ensure that the hard drive does not already contain a bootable operating system, and set the following boot order in the BIOS.

1. removable storage device (USB flash drive or DVD drive)
2. hard drive
3. network (PXE), over an interface connected to the internal management network

For BIOS configuration details, refer to the OEM documentation supplied with the computing node.

NOTE: If a host contains a bootable hard drive, either erase the drive beforehand, or ensure that the host is set to boot from the correct source for initial configuration. If necessary, you can change the boot device at boot time by pressing a dedicated key. For more information, refer to the OEM documentation for the compute node.