

# CEE Technical Description

Cloud Execution Environment

TECHNICAL PRODUCT DESCRIPTION

**Copyright**

© Ericsson AB 2016–2018. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

**Disclaimer**

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.

**Trademark List**

All trademarks mentioned herein are the property of their respective owners. These are shown in the document Trademark Information.



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Product Overview</b>	<b>2</b>
2.1	CEE	2
2.2	Software Versions in CEE	6
<b>3</b>	<b>Architectural Goals and Constraints</b>	<b>8</b>
3.1	Carrier Grade Support	8
3.2	Infrastructure as a Service	8
<b>4</b>	<b>Functional Description</b>	<b>9</b>
4.1	Virtual Resource Control	9
4.2	Cloud Infrastructure Controller	11
4.3	Compute Server	13
4.4	vFuel	13
4.5	Multi-Server Network Configuration	13
4.6	Single Server Network Configuration	17
4.7	Storage	18
4.8	Data Center Gateway	20
4.9	Data Center Firewall	20
4.10	Traffic Networking	22
4.11	Cloud SDN Switch	32
4.12	Software-Defined Networking in CEE	32
4.13	Switching Fabric	33
4.14	Cloud Management System	34
4.15	Software Management	35
4.16	Backup and Restore	36
4.17	Audit and Health Check	36
4.18	Licensing	36
4.19	Performance Management	37
4.20	High Availability	37
4.21	Security	38
4.22	End-User Access	39
	<b>Reference List</b>	<b>41</b>





# 1 Introduction

This document describes the Cloud Execution Environment (CEE) 6 release, which is part of the larger Ericsson Cloud System solution. CEE is based on OpenStack® software.

This document describes the following:

- Generic OpenStack cloud concepts
- Added concepts of CEE
- Overall architecture of CEE
- Actors in CEE
- Main characteristics of CEE



## 2 Product Overview

CEE is an Infrastructure-as-a-Service (IaaS) solution. CEE is based on OpenStack, with additional features that expand its flexibility of use and meet the needs of telecommunication service providers.

### 2.1 CEE

CEE is a software product, which can execute on several hardware configurations. CEE consists of the following parts:

- Compute
- Networking
- Storage

These resources are managed via the Atlas Dashboard or via CEE OpenStack Rest API and combined they are referred to as a CEE region. An overview of the CEE region is shown in Figure 1.

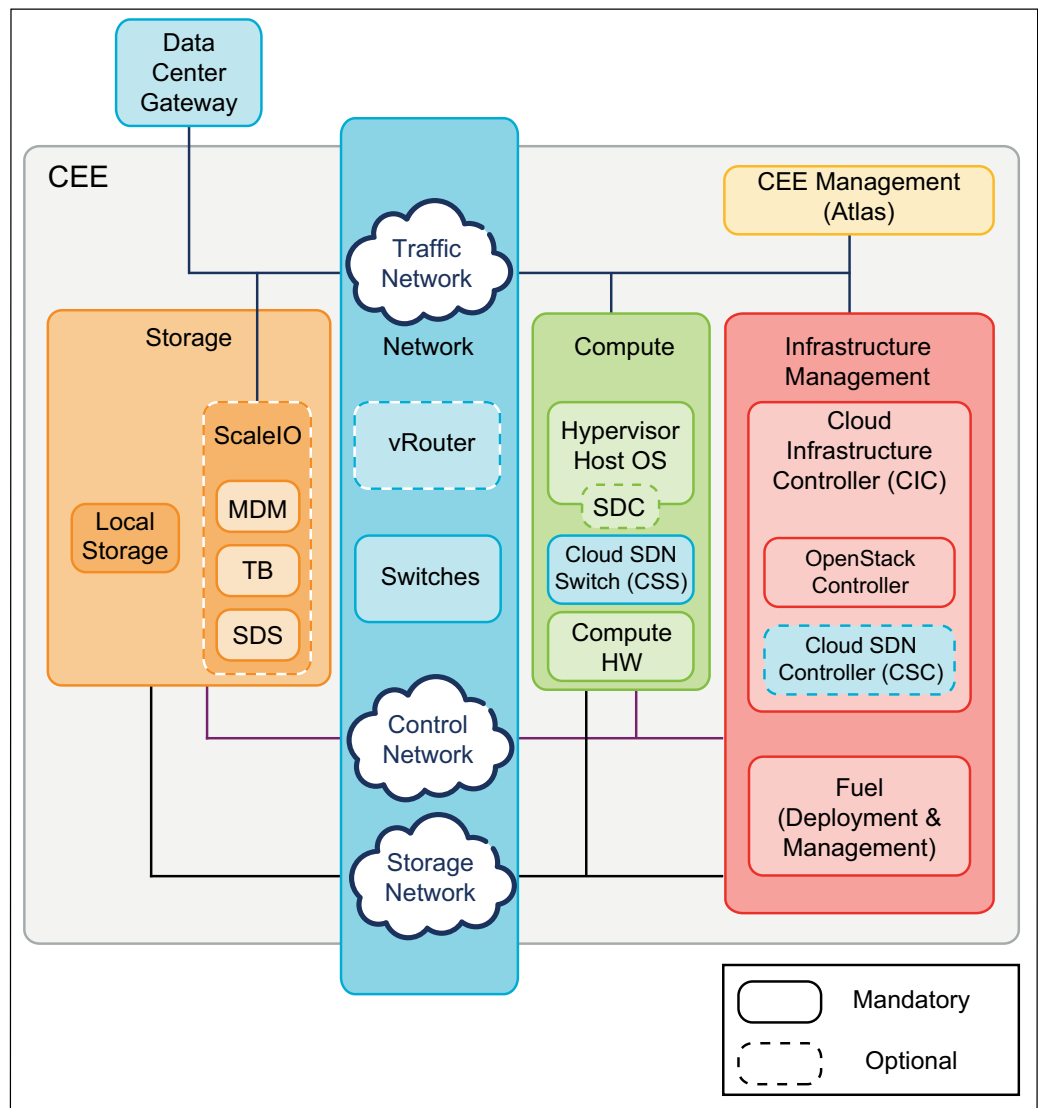


Figure 1 CEE Architectural Overview

Ericsson has added functionality to the OpenStack components of Compute, Networking, and Storage. CEE provides the following services:

- Continuous end-user access to services, through:
  - Redundant (x3) virtual Cloud Infrastructure Controllers (vCICs)
  - Continuous Monitoring High Availability (CM-HA)
  - Persistent affinity
- High throughput with low latency, through:
  - Ericsson Cloud SDN Switch (CSS), based on Open vSwitch (OVS)
  - Single Root Input/Output Virtualization (SR-IOV)



- PCI passthrough (for network devices where the SR-IOV is turned off) or SR-IOV Physical Function passthrough (for network devices where the SR-IOV is turned on)
- Rapid and controlled VM deployment, through:
  - Resource aware scheduling
  - Granular resource scheduling control
  - Automated installation, with OVF support
  - Automatic and on-demand scaling
- A unified cloud infrastructure Operations and Maintenance, through:
  - Fault and performance management
  - Upgrade and rollback
- Tenant network management through Cloud SDN Controller (CSC) (optional)
- Simplified security administration
- Trusted tenant isolation

CEE has an efficient infrastructure utilization, and deployment options include the following:

- Single server, with non-redundant controller deployments
- Multi-server, with redundant controller deployments
- Configurations with Software Defined Networking (SDN), using CSC

CEE contains the virtualization layer (hypervisors) and virtual switches (vSwitches). It also provides hooks to integrate and orchestrate external physical appliances, for example, physical switches and storage arrays.

Cloud management provides high-level orchestration for the following:

- Application deployment, monitoring, and management
- CEE infrastructure planning, fulfillment, assurance, and charging

Atlas and Ericsson Cloud Manager (ECM) are examples of cloud management systems. Atlas is included in CEE.

When describing the available management features in this document, the use of Atlas is assumed.





## 2.1.1 Hardware

Table 1 lists hardware specifications on which CEE 6 software and procedures have been verified. CEE 6 can also work with other hardware not listed in the below table, however, it is recommended that the appropriate parties test CEE on hardware not listed and estimate potential System Integration (SI) efforts.

**Note:** The CEE product development organization handles the trouble reports filed to CEE regarding deployments on hardware not listed in Table 1, provided that the problem is proven to be CEE software-specific and not dependent on the hardware used.

Table 1 Hardware Types Used for CEE 6 Software and Procedures Verification

Hardware Type <sup>(1)(2)</sup>	Product Version	Equipment Management	Servers <sup>(3)(4)</sup>	CPUs <sup>(5)</sup>	NICs	Switches
BSP 8100	From R11.0.1 to R12.0.1	DMX (NETCON F)	GEP5, GEP7 and GEP7L server types included in the BSP 8100 product version. <sup>(3)</sup>	CPU included in the server model	NICs included in the server model	All switches part of the BSP 8100 product version
HDS 8000	From 2.5 to 2.7	CCM	CSU01, CSU02, CRU01 server types in the HDS 8000 product version. <sup>(3)</sup>	CPU included in the server model	All NICs included in the server model	All switches included in the HDS 8000 product version (unmanaged switches from CEE perspective)
COTS <sup>(6)</sup>		Dell iDRAC, HPE onboard administrator	Dell R620, R630, HPE Blade System c7000 Gen8 and Gen9	CPU included in the server model	Intel X520	Extreme Networks X670, X670V, X770

(1) For the characteristics of the different hardware components, refer to the hardware specific documentation.

(2) CEE supports heterogeneous servers of the same hardware type within the same CEE region. It is not possible to mix two hardware types in the same CEE region.

(3) Servers selected for a specific role (compute, compute without Storage Area Network (SAN), compute with vCICs, ScaleIO) must have enough resources (Ethernet ports, CPU cores, disk sizes and performance, RAM) as described in Multi-Server System Dimensioning Guide, CEE 6.

(4) The ScaleIO Data Servers (SDSs) for the optional distributed storage are hosted on dedicated physical servers. On these physical servers, CEE does not host vCICs and does not instantiate tenant Virtual Machines (VMs).

(5) All server models are dual CPU sockets except for the ones that are part of the BSP 8100 hardware type.

(6) COTS hardware is available in several variants in the same server model. It is necessary to check with the vendor if the selected hardware is compatible with 3PP software components of CEE, see Section 2.2 on page 6.

On the Ericsson Hyperscale Datacenter System (HDS), CEE is deployed in a vPOD, provided by the owner of the datacenter. For more information on vPODs refer to HDS documentation, available as described in Hyperscale Datacenter System 8000 Customer Documentation, Reference [1].



## 2.1.2 Single Server CEE

CEE can be used in a single server deployment, using only one vCIC. There is no redundancy in compute or network infrastructure, and only local storage is supported:

- There is no High Availability for vCIC services.
- Only one compute host exists.
- Ceilometer is disabled, since the statistical data is not needed.
- Continuous Monitoring High Availability (CM-HA) service is disabled, so there are no alarms for CIC failed or compute host failed.
- Fuel is used to install single server CEE, but it is not migrated to the single server after installation.

## 2.2 Software Versions in CEE

The following software versions are used in CEE:

Software	Version
Ubuntu Linux	14.04.5 LTS
Linux repository version	2018_03_12
Linux kernel	4.4.0-116-generic
Mirantis OpenStack	9.2
CentOS	7.4.1708
EMC ScaleIO	2.5.0
QEMU	ericsson_performance disabled: qemu_2.5+dfsg-5ubuntu10.16~cloud0
	ericsson_performance enabled: qemu_2.5+dfsg-5ubuntu10.14~cloud0 ubuntu2
libvirt	libvirt0_1.3.1-1ubuntu10.15~cloud0
Zabbix	2.2.19
DPDK	16.11.4
Cloud SDN Switch (CSS)	CSS R10D 1.0.16-1
OVS on vCIC	2.6.1-0~u1404+mos3
OVS on compute host	2.6.2.css4.R10C~a59331f-1
NTP	4.2.6p5
OpenSSL	1.0.1f



#### Ethernet drivers:

i40e	2.2.4-1~u14.04+mos1
i40evf (not in use but installed)	1.4.15-k
ixgbe	4.2.1-k
igb	5.3.0-k
vfio-pci	0.2

## 3 Architectural Goals and Constraints

The long-term goal for CEE is to provide support for the Infrastructure as a Service (IaaS) cloud services, which are described in this section.

### 3.1 Carrier Grade Support

The main objective of CEE is to provide carrier grade support, that is, to support the need of service providers with the following service characteristics:

**Higher availability**

The reliability of the infrastructure is based on observability, which ensures better management through system monitoring, and nearly zero down-time.

**Higher security**

CEE provides mechanisms which exceed security features found in traditional IT environments, since the impact of security incidents is higher. In addition to standard IT security considerations, CEE addresses a large number of additional attack vectors, security policies, and runtime audits.

**Higher predictability**

CEE is designed for real-time communication workloads. Its mechanisms ensure predictable real-time execution, low latency, low response variance, and high network throughput, even in case of small packet sizes.

### 3.2 Infrastructure as a Service

CEE supports the Infrastructure as a Service (IaaS) service model.

In IaaS, an organization outsources the equipment they use to support their own operations. Outsourced equipment includes the following:

- Hardware
- Storage
- Servers
- Network capacity



## 4 Functional Description

The logical components in CEE are shown in Figure 2. These are described in the following subsections.

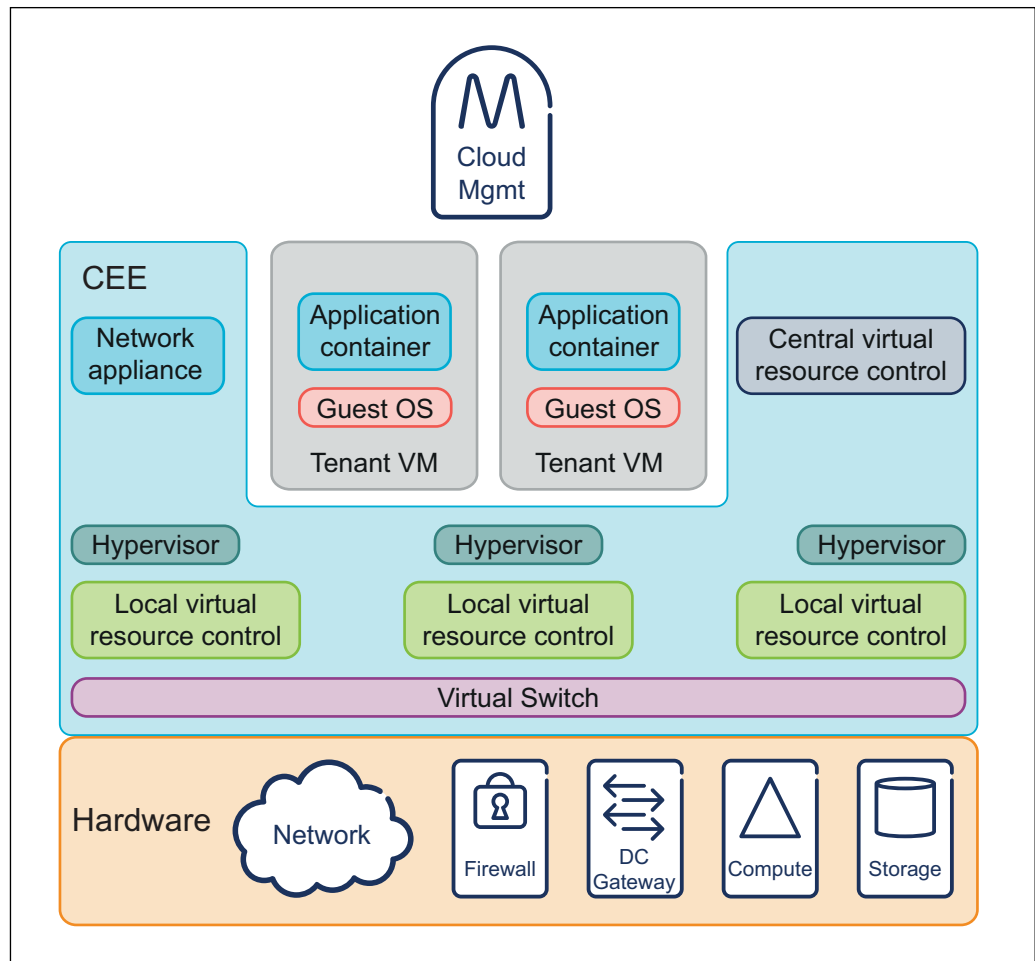


Figure 2 Logical Components in CEE

### 4.1 Virtual Resource Control

CEE provides the virtualization control and a management layer based on the OpenStack virtualization management system, which controls pools of compute, storage, and networking resources throughout a Data Center.

As shown in Figure 3, the following OpenStack components are used in CEE:

- Network service (Neutron) provides “network connectivity as a service” between interface devices managed by other OpenStack services (most likely Nova).

- Compute service (Nova) provides virtual servers upon demand.
- Image service (Glance) provides a catalog and repository for virtual disk images.
- Object Storage (Swift) allows storage and retrieval of objects (but not mounting directories like a fileserver). In CEE 6, Object Storage is not available for tenants.
- Block Storage (Cinder) provides persistent block storage to guest Virtual Machines (VMs).
- Identity management (Keystone) provides authentication and authorization for all OpenStack services.
- Performance measurement support (Ceilometer) provides counter and alarm information for charging and performance monitoring.
- Dashboard (Horizon) enables management of a CEE region. Dashboard is part of Atlas.
- Orchestration engine (Heat) launches multiple composite cloud applications based on templates. Heat is part of Atlas.
- Life cycle management (vFuel) enables management of the infrastructure hardware and software.

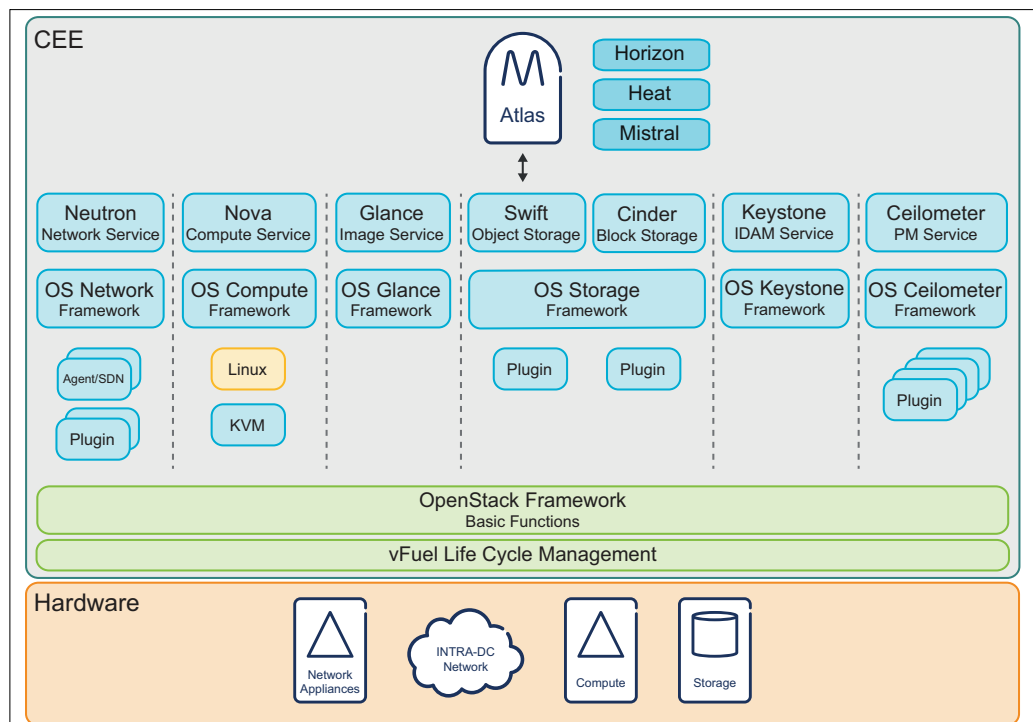


Figure 3 OpenStack in CEE



An OpenStack region is defined as one OpenStack instance running on several physical compute servers. A region is a self-contained collection of compute servers, switch fabrics, storage, and so on.

The following operations can be performed on a VM using OpenStack:

- Image Loading: When a VM is launched, a code image containing the Operating System (OS) and application that the VM is intended for is provided by the hypervisor through file system access.
- Creation of interfaces that can later be attached to a network: The VM needs to be restarted to add Virtual Network Interface Controllers (vNICs).
- VM deployment
- Start, stop, and graceful shutdown of VMs
- Migration of VMs
- Policy-based placement
- High availability (HA) rules for VM deployment to achieve the required and agreed-upon level of HA for the application
- Collection of performance information from tenant VMs and infrastructure
- Reconfiguration for upgrades, failures, or optimization of system hardware use

## 4.2 Cloud Infrastructure Controller

The infrastructure services and control for CEE are located on three virtual Cloud Infrastructure Controller (vCIC) nodes. The vCIC nodes run on compute hosts. The three vCIC nodes form a high availability quorum.

**Note:** Single server deployment runs on one vCIC. There is no High Availability for vCIC services in this case. In single server deployment, Ceilometer is optional.

Depending on the service, either a three-way-active service delivery model or an active-standby service delivery model applies. A summary of the internal vCIC services is shown in Figure 4.

CEE is deployed in a small footprint configuration, which allows vCICs and applications to be co-located on the same host.

The vCIC nodes have Ubuntu Linux as their host OS.

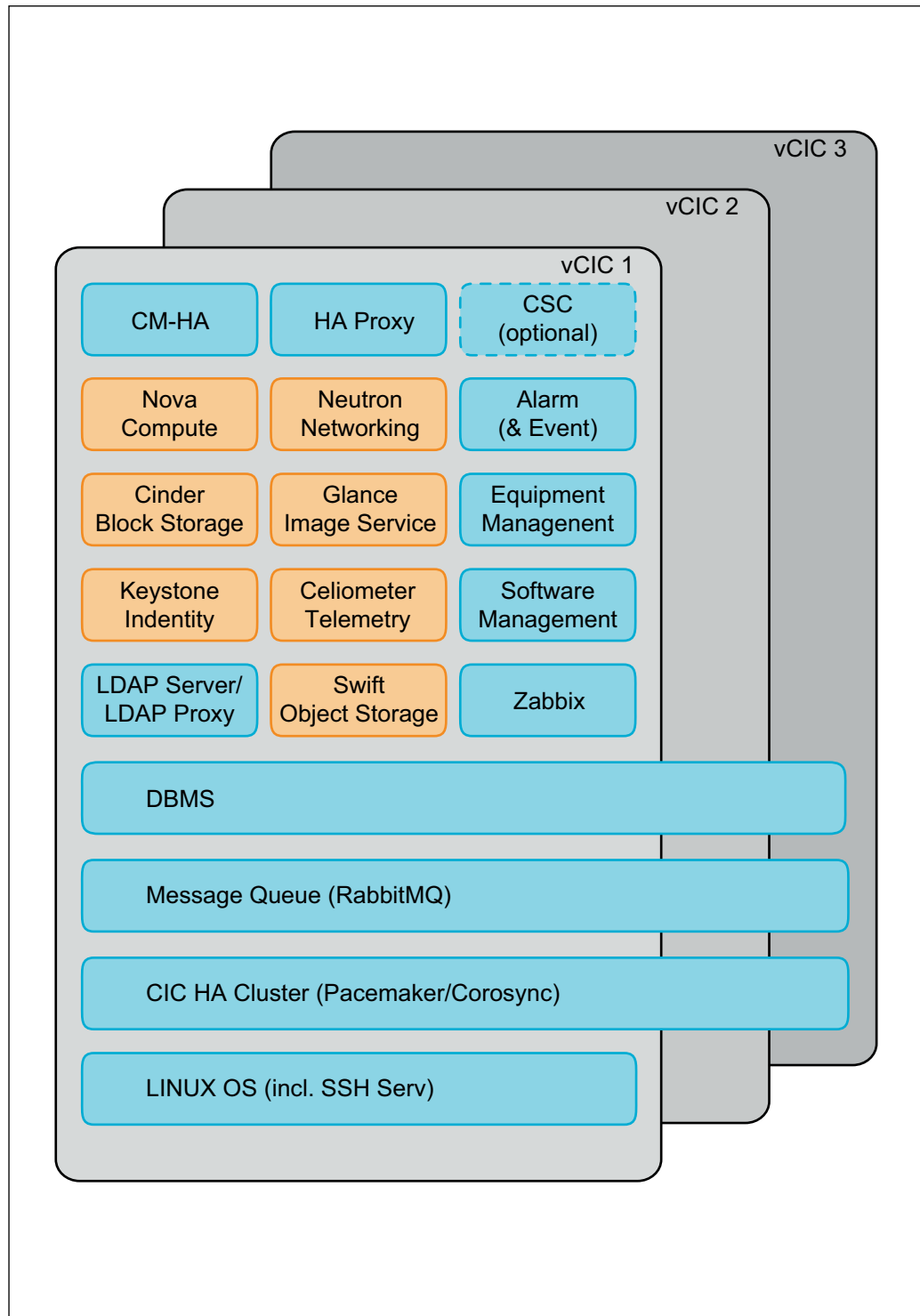


Figure 4 vCIC Redundancy Scheme





## 4.3 Compute Server

The compute server is connected to the switch fabric. Each compute server has a redundant connection to the switch for the VM data traffic and another redundant connection for the storage traffic.

The host OS on the compute blades is Ubuntu Linux. Linux and Windows are supported as guest OSs.

CEE uses the Kernel-Based Virtual Machine (KVM) hypervisor. In CEE, KVM settings are optimized for the needs of telecommunication providers.

Cloud Software Defined Networking Switch (CSS) is used as a virtual switch on the compute servers.

SR-IOV, PCI passthrough, and SR-IOV Physical Function passthrough based traffic is also configurable with free NICs.

**Note:** Only one compute host exists in a single server setup.

## 4.4 vFuel

vFuel manages the CEE infrastructure life cycle. vFuel is responsible for the following:

- Maiden software installation of the CEE software
- Updates of the CEE software
- Adding and removing hardware resources

vFuel runs as a VM on a compute host. Depending on the capacity of the compute host, vFuel and one of the vCIC nodes can run on the same compute host. vFuel uses CentOS Linux as its OS.

## 4.5 Multi-Server Network Configuration

The network of a multi-server CEE region is based on three switching domains, as shown in Figure 5:

### **Traffic switching domain**

Also referred to as Traffic Network (or traffic LAN)

### **Storage switching domain**

Also referred to as Storage Area Network (SAN)

### **Control switching domain**

Also referred to as Control Network (or control LAN)

There is physical redundancy - all physical servers are connected to all switching domains. It is fault tolerant against any Single Point of Failure (SPoF) with reasonable low failover time.

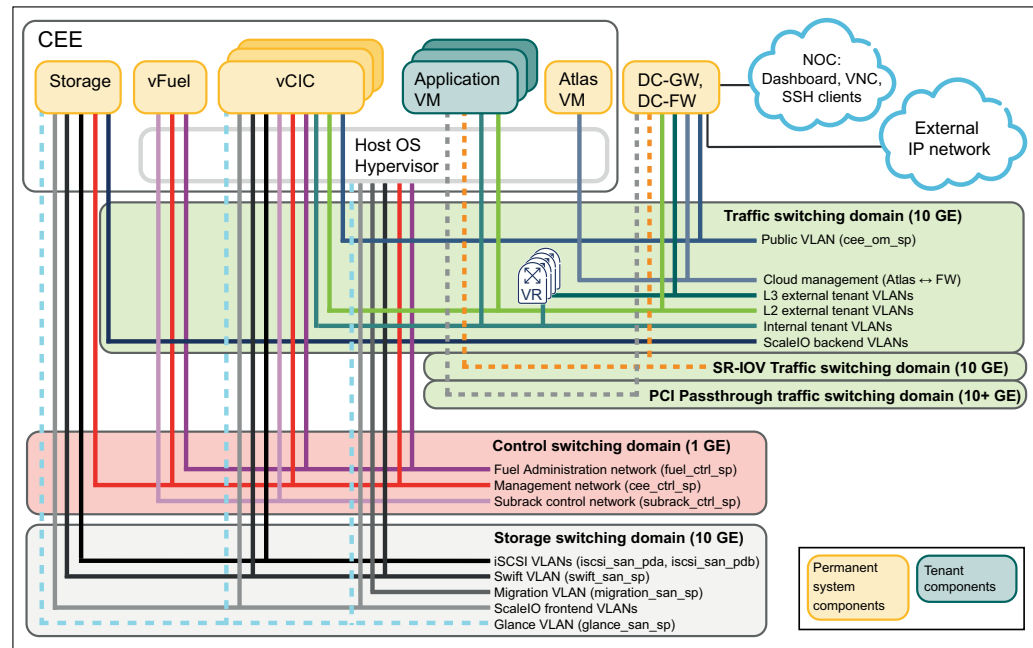


Figure 5 Logical Network Configuration, Multi-Server CEE

The **Control switching domain** is used for the OpenStack control tasks and for the management of CEE. The boot and installation of servers are also performed on this network and similarly, the upgrade of the software of the infrastructure components are also done on this network.

The most important role of the **traffic switching domain** is to forward the internal and external traffic of the tenant VMs. The traffic switching domain is the only network that has direct external connectivity. The Data Center Gateway (DC-GW) and Firewall (FW) are connected to this network. Because of the external connectivity, the northbound Operation, Administration, and Maintenance (OAM) interfaces are also connected to this network and external communication with the vCICs is also performed here.

The **storage switching domain** is used for storage access when external storage equipment is used. The external storage equipment is connected to this network using Internet Small Computer System Interface (iSCSI) protocol (iscsi\_san\_pda, iscsi\_san\_pdb), or other protocols (ScaleIO). It also provides VLAN separated Migration (migration\_san\_sp), Glance (glance\_san\_sp), and Swift (swift\_san\_sp) related functions. The tenants do not have direct access to this network.

#### 4.5.1 Multi-Server Network Configuration on HDS with SDN

The SDN specific networks are shown in Figure 6.

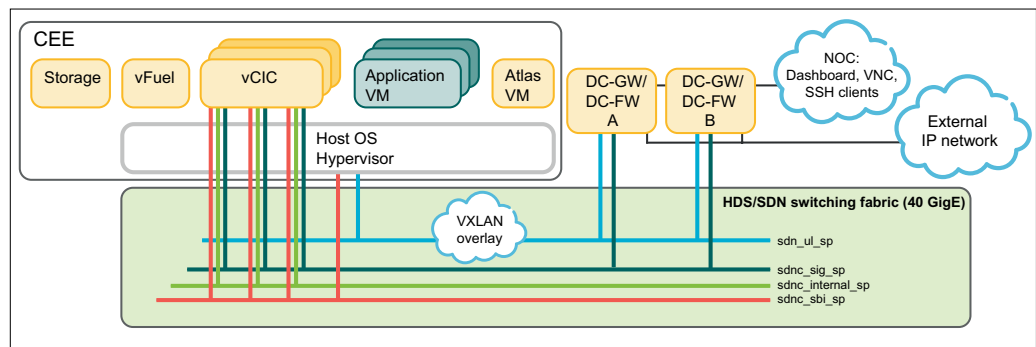


Figure 6 Logical Network Configuration, SDN

#### 4.5.2 Multi-Server CEE Host Networking

From the server side, each server is connected to each switching domain (control, traffic, storage) with two physical Network Interface Controllers (NICs). In this document, one NIC in a pair is named left and the other one is named right. This scheme results in six physical NICs in total per server. In case of servers with SR-IOV, PCI passthrough, or SR-IOV Physical Function passthrough traffic support, eight physical NICs are present, as shown in Figure 7. Only one of them (SR-IOV, PCI passthrough, or SR-IOV Physical Function passthrough) can be configured on a compute server.

From the switching domain side, each domain is built from one or more network switch pairs. One left and one right physical switch form a pair, with an inter-switch link connecting the two switches.

Based on this, a server is connected to a switch pair of a switching domain by connecting the left physical NIC to the left physical switch and the right NIC to the right switch.

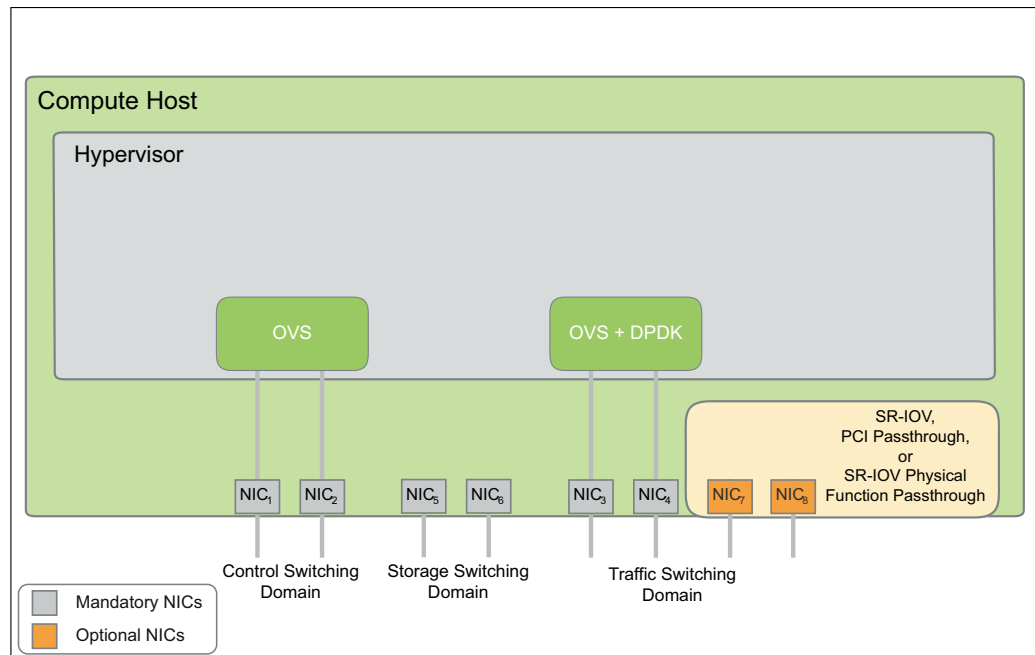


Figure 7 Aggregated Connections between Servers and Switching Domains

The two-side aggregated connectivity is introduced for reliability and performance optimization. Logically, the two sides are parts of the same domain. Consequently, the low-level virtual network architecture only comprises three switching domains with aggregated server connections.

Compute servers not hosting vCICs can be configured without a storage switching domain. In this case, the unused storage ports can be configured with SR-IOV, PCI passthrough, or SR-IOV Physical Function passthrough for traffic support, as shown in Figure 8.

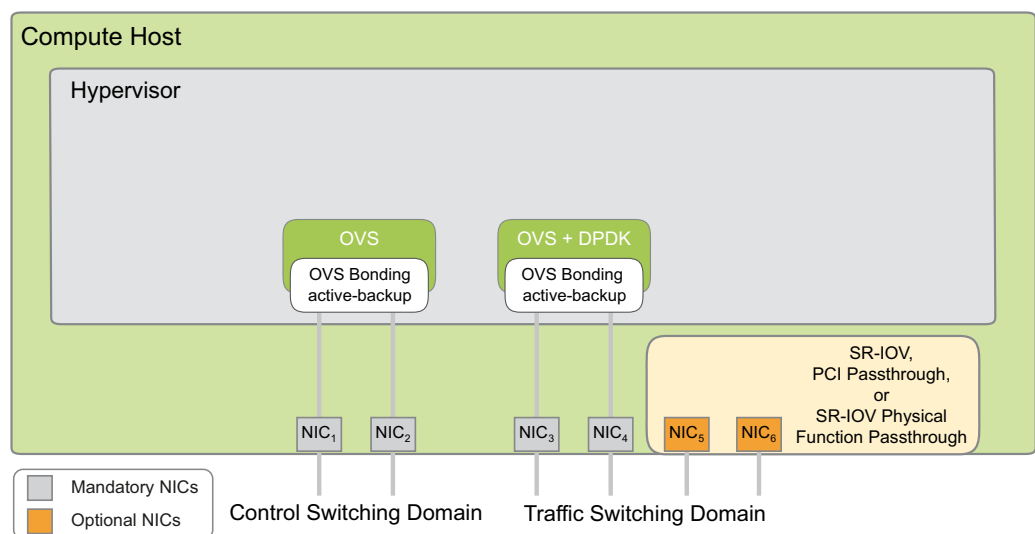


Figure 8 Compute Host Network without Storage Switching Domain



The design of the switching domains is highly restricted by the existing physical architecture, and the resulting scheme does not provide direct connection to the logical architecture or infrastructure. For this reason, a virtual switching layer is implemented on the top of the physical NICs to provide the necessary virtual interfaces for CEE.

## 4.6 Single Server Network Configuration

As shown in Figure 9, the network of a single server CEE region is based on two switching domains, without redundancy:

### Traffic switching domain

Also referred to as Traffic Network (or traffic LAN)

### Control switching domain

Also referred to as Control Network (or control LAN)

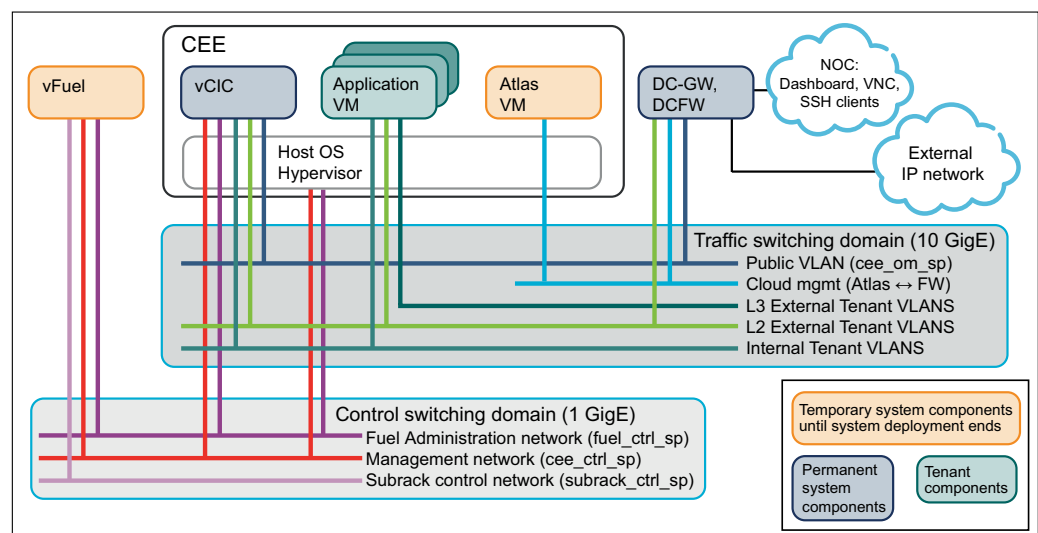


Figure 9 Logical Network Configuration, Single Server CEE

### 4.6.1 Single Server CEE Host Networking Solution

The two NICs are connected to the two switching domains, regardless of their role in Mirantis OpenStack.

On the control switching domain, the physical NIC is always assigned to OVS. This interface is visible to the kernel of the host operating system and it can be configured and interrogated by using traditional software tools.

SR-IOV, PCI passthrough, and SR-IOV Physical Function passthrough based traffic switching domains are not possible on single server CEE deployments.



## 4.7 Storage

CEE supports the following storage options:

### Persistent storage

- OpenStack Block Storage (Cinder), see Section 4.7.1 on page 18
- OpenStack Object Storage (Swift), see Section 4.7.2 on page 18

### Non-persistent storage

Ephemeral storage controlled by OpenStack Compute (Nova)

**Local Storage** Local, ephemeral storage

### 4.7.1 OpenStack Block Storage

Some CEE configurations provide a back end for OpenStack Block Storage using the EMC<sup>2</sup> ScaleIO dedicated server solution.

**Note:** OpenStack Block Storage is not possible without the EMC<sup>2</sup> ScaleIO dedicated server solution. EMC<sup>2</sup> ScaleIO is optional.

OpenStack Block Storage provides persistent block level storage devices for use with OpenStack Compute instances. Block storage volumes are fully integrated into OpenStack Compute and the Dashboard, which allows cloud users to manage their own storage needs.

#### Cinder Backup

The `cinder-backup` service provides backing up of volumes managed by Cinder to a backup storage provider using a driver architecture. In CEE, the `cinder-backup` service is available as an optional service, with Swift or an external Network File System (NFS) storage server as back end. Connectivity between the service and the external NFS storage is established through the traffic network `cee_om_sp`.

To enable and configure the service, refer to [Fuel Plugin Configuration Guide](#) for new CEE installations, and [Runtime Configuration Guide](#) for already installed systems.

For information about the `cinder backup` command, refer to the [Command-Line Interface Reference](#).

For limitations and recommendations, refer to the [OpenStack Block Storage API](#) in CEE.

For information on the API calls of the Cinder backup function, refer to the [OpenStack API Complete Reference](#).



## 4.7.2 OpenStack Object Storage

OpenStack Object Storage is supported only as a back end for OpenStack Image Service (Glance) and CEE internal purposes. Each vCIC works as a Swift proxy and Swift storage node. Swift storage is provided by the local disks of the vCICs. Replicas are stored for each vCIC.

## 4.7.3 EMC ScaleIO

ScaleIO is a software-only solution that uses LAN and existing local disks of a server to create a virtual SAN that has all the benefits of external storage. ScaleIO utilizes the existing local storage devices to turn them into shared block storage. The software immediately responds to the changes, rebalancing the storage distribution and achieving a layout that optimally suits the new configuration. In CEE, ScaleIO is used as block storage back end for Cinder. Optionally, the Swift store can be moved from local storage to ScaleIO to increase Swift storage space. In this case, part of the storage pool used for Cinder is used for the Swift store.

In CEE, ScaleIO is used in the following configurations:

- In managed configuration, the ScaleIO Data Server and the Meta Data Manager (MDM) cluster components are deployed on dedicated servers within the CEE region. The ScaleIO Data Client (SDC) component is installed on all compute hosts. The SDSs and SDCs communicate on the traffic networks, using a standard LAN to handle the application I/O requests sent to ScaleIO block volumes. SDSs are controlled by the MDM cluster on the control plane. ScaleIO is deployed and updated together with CEE. CEE Monitors and manages both the server and the client side of ScaleIO.
- In unmanaged configuration, the ScaleIO server side is deployed on dedicated hardware outside of the CEE region. The ScaleIO Data Client (SDC) component is installed on all compute hosts. The SDCs and the ScaleIO servers side communicate through the DC-GW and the IP network to handle the application I/O requests sent to ScaleIO block volumes. In unmanaged configuration, the server side is deployed independently of CEE, and is not managed or monitored by CEE.

ScaleIO architecture in CEE is shown in Figure 10.

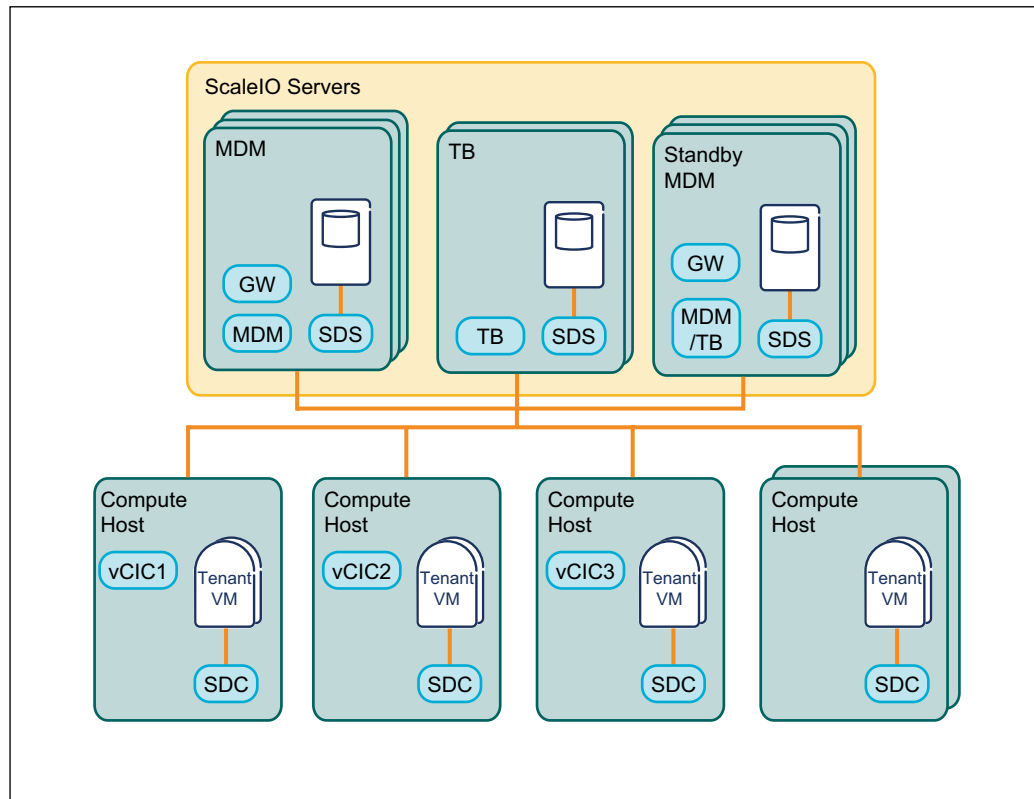


Figure 10 ScaleIO Architecture in CEE

In unmanaged configuration, the servers in the yellow “ScaleIO Servers” box are outside of the CEE region.

For more information, refer to the document [ScaleIO Architecture Description](#).

## 4.8 Data Center Gateway

A Data Center Gateway (DC-GW) is required, but it is not part of CEE.

To produce a homogeneous operating environment, the Data Center Gateway provides Layer 2 and Layer 3 connectivity to the Enterprise VPN and the private Data Center, and Layer 3 connectivity to the public Internet.

Instead of Data Center Gateway, the term BGW is used in other Ericsson documents.

## 4.9 Data Center Firewall

Data Center Firewalls (DC-FWs) are required, but they are not part of CEE. To support configuration, CEE provides hardening guidelines for the configuration of DC-FWs.





In the cloud infrastructure, FWs in different locations have different functionality and capacities. Examples include the following:

- Default Access Control List (ACL) with stateless rule
- ACLs with malware detection
- Intrusion Detection System (IDS)
- Intrusion Prevention System (IPS)

A logical view of the FW configuration for CEE is shown in Figure 11.

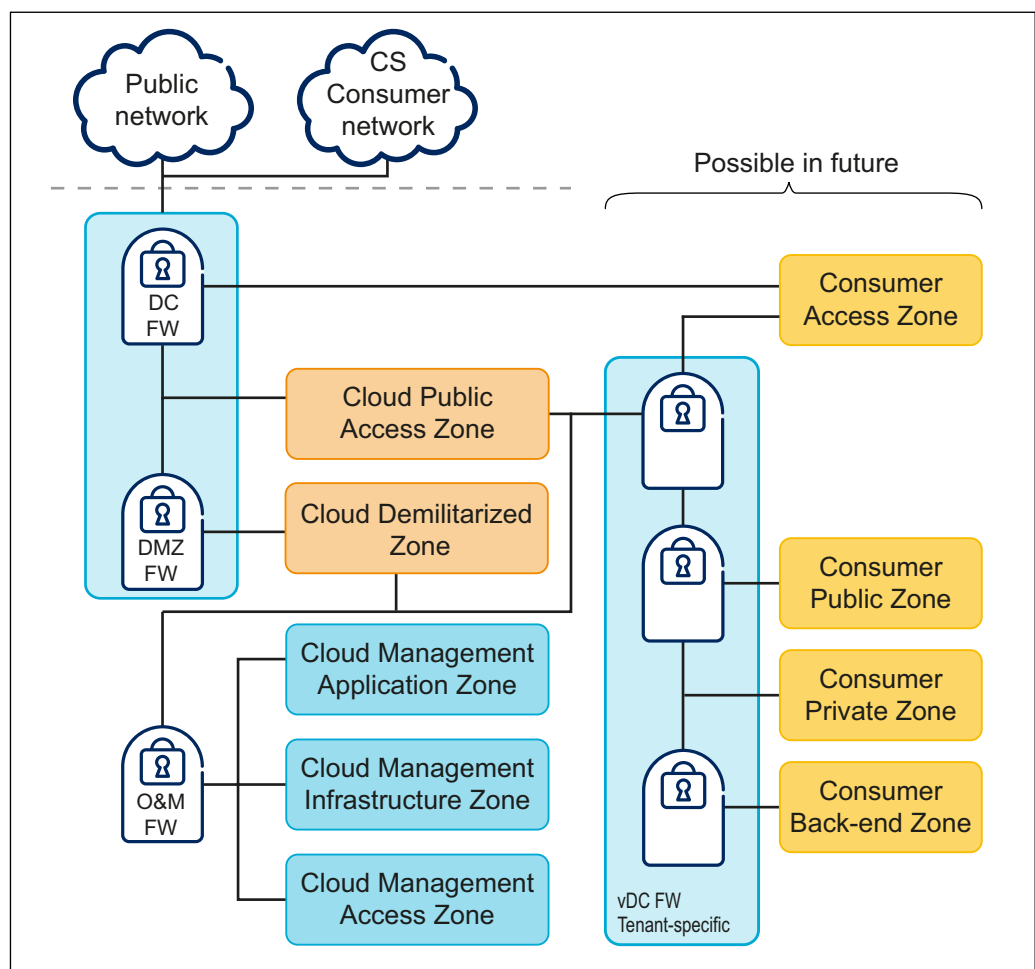


Figure 11 Logical View of FW Functionality

The logical FWs, as shown in Figure 11, are as follows:

- DC-FW  
This FW protects the data center from external attacks. Communication between different Cloud Service Providers within the cloud must also go through this FW. The DC-FW needs to support multi-Gbps traffic with Distributed Denial of Service (DDoS) protection.

- Demilitarized Zone (DMZ) FW  
This FW protects the DMZ zone.
- Cloud Operation and Maintenance (O&M) FW  
This FW protects the O&M cloud management infrastructure. The function does not require high capacity. Firewalling functionality needs to be supported, including the following:
  - Application firewalling
  - Content filtering
  - Malware detection
  - User based security policies of the cloud service
- Virtual Data Center Firewall (vDC-FW)  
This FW is specific to the Cloud Service Provider (at least one FW per provider is needed). It supports the security policy of the Service Provider. The administrative control of this FW is partly delegated to the tenant administrators. There is also a back-end vDC-FW, which protects the back-end network of the virtual Data Center, including the databases.

## 4.10 Traffic Networking

This section describes the tenant view of traffic networking.

CEE supports tenant network orchestration with OpenStack Networking (Neutron).

The performance of contemporary Data Centers greatly depends on the networking technology used to interconnect computing, storage, and services elements, which are always present in a cloud computing environment. Networking is also important because a uniform network provides access to most available resources (main processing memory yet excluded), thus east-west connectivity (traffic within the DC; server to server communication) becomes as important as north-south (outside connectivity; client to server communication). This puts high demands on the network fabric and requires non-blocking, lossless, and fine grained back-pressure characteristics.

The cloud networking solution is based on a combination of the following Layer 2 to Layer 7 elements:

- Classification
- Switching and routing
- Forwarding
- Filtering
- Shaping and policing



These elements are implemented in a combination of software and hardware.

#### 4.10.1 Cloud Networking Model

The cloud networking model is hierarchical, with a clear relationship between the main connectivity primitives.

As shown in Figure 12, a set of network attachment points are connected by a VLAN, or VXLAN if SDN is utilized, as a virtual broadcast segment. A VLAN/VXLAN corresponds to a subnet at Layer 3, and one or more subnets can be combined into a Layer 3 VPN instance. The access to the public Internet is logically modeled as a VPN.

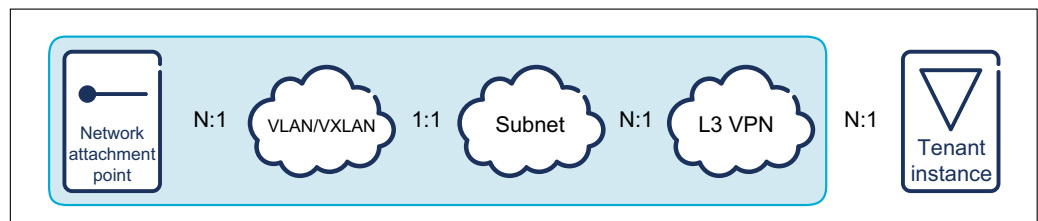


Figure 12 Cloud Networking Model

Multiple Layer 3 VPN instances constitute a tenant instance, although in some circumstances contiguous connectivity cannot exist across Layer 3 VPN instances. (For example, one VPN is access to the Internet, and another VPN is access to storage, and application software is connected to both of them by logically disjoint network attachment points). Connectivity can exist in the form of a FW, NAT or other filtering network appliance.

#### 4.10.2 Layer 2 Cloud Network Connectivity

The Ethernet VLAN, which is a virtualized broadcast domain at Layer 2, and a subnet at Layer 3, is the basic unit of intra-cloud connectivity, virtualization, and Cloud Service Provider isolation.

#### 4.10.3 Layer 3 Cloud Network Connectivity

VLAN as a subnet is the basic Layer 3 unit of connectivity, where one or more subnets can be combined to create Layer 3 VPNs. The ethertype in the Ethernet frame permits them to coexist simultaneously.

The subnets belong to three classes:

- Publicly routable prefixes
- VPN (private) prefixes
- Zeroconf prefixes (requires reachability only within a cloud subnet)



A physical point of attachment to the fabric can have multiple virtual interfaces that connect to multiple classes of Layer 3 subnets. For example, a VM can have a virtual interface to a Network Address Translated (NATted) prefix for communication with clients in the public Internet and a virtual interface to a zeroconf prefix that connects it to a set of supporting VMs.

#### **4.10.4 Cloud Service Provider Isolation**

Cloud Service Providers co-located in the same Data Center must be isolated from each other. Multiple technologies exist to support Service Provider isolation, as shown in Figure 13. Currently only one isolation technique is supported, the Customer VLAN (C-VLAN) Cloud Service Provider grouping.

The VLAN identifier (VID) space (4096) is divided among Cloud Service Provider tenants. This means that the number of private VLANs used by each Cloud Service Provider is limited to a fraction of 4096.

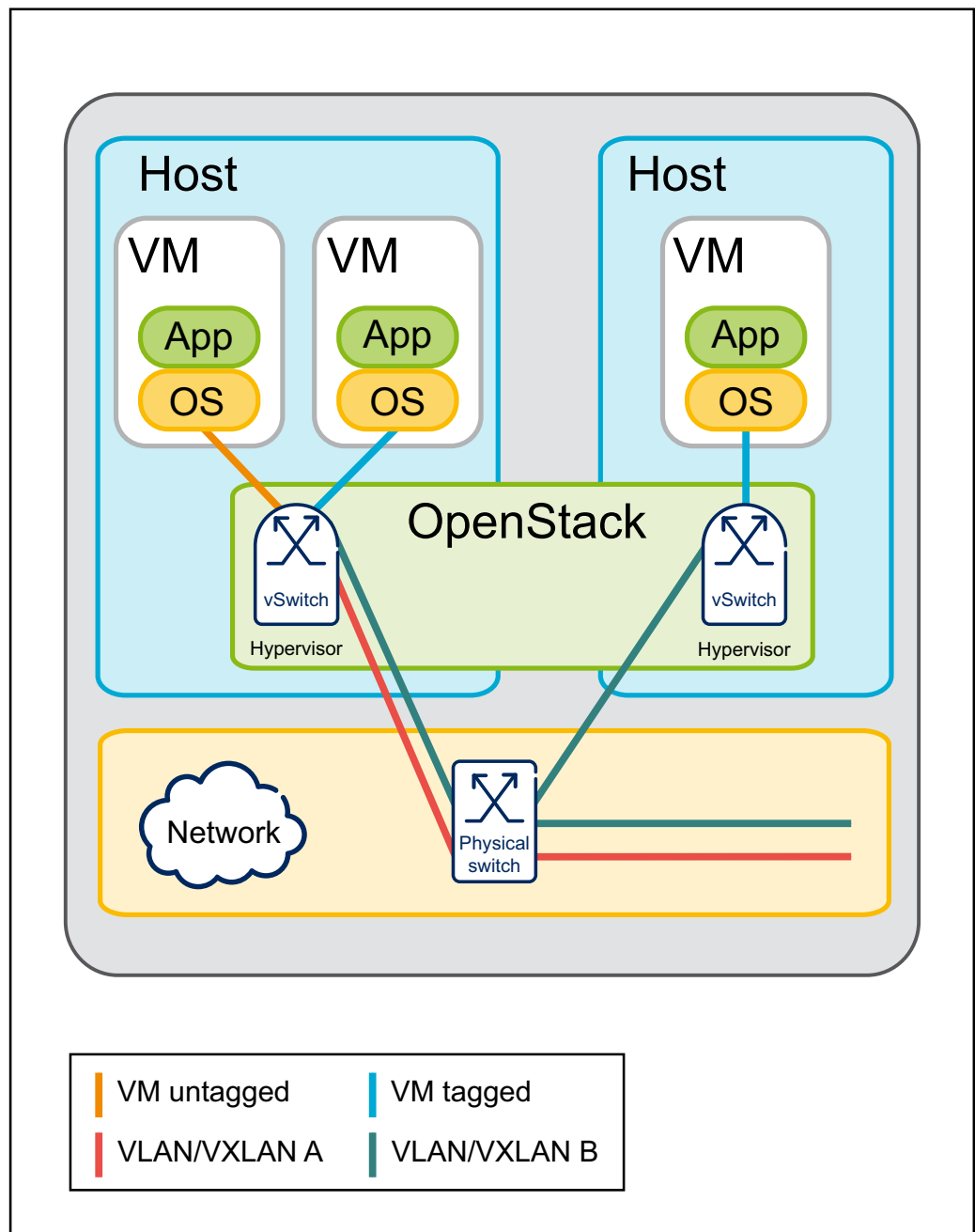


Figure 13 Cloud Service Provider Isolation

#### 4.10.5

#### Resiliency

This section describes the switching resiliency in CEE.

#### 4.10.5.1 Layer 2 Resiliency

NIC teaming is used on the two Ethernet interfaces in the traffic network domain on each host. Depending on the functionality provided by the traffic switch, different Layer 2 resiliency mechanisms of the Ethernet interfaces are used:

Link Aggregation Control Protocol (LACP) based NIC teaming:

In this configuration, network resiliency of the Ethernet interface in the servers is based on NIC teaming that uses LACP procedures. LACP is available in the host when the support in the traffic switch is based on the Extreme Multi-Chassis Link Aggregation Group (MLAG).

#### 4.10.5.2 Layer 3 Resiliency

For Layer 3 resiliency, in configurations based on Extreme switches, CEE uses Virtual Router Redundancy Protocol (VRRP) for redundancy between Virtual Routers.

### 4.10.6 Network Type 1: Internal Layer 2 Neutron Network

See Figure 14 for Network Type 1, which is an internal Layer 2 Neutron network.

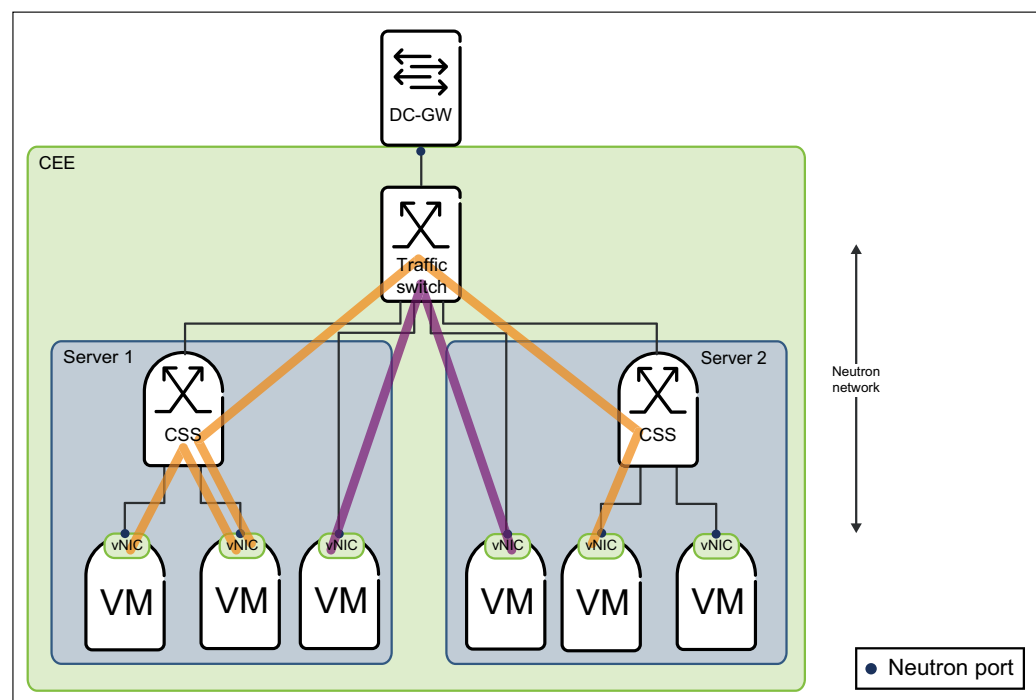


Figure 14 Network Type 1: Internal Layer 2 Neutron Network

Figure 14 shows the common setup of Layer 2 communication between VMs in the same CEE region. VMs on the same physical server communicate through CSS, while VMs located on different physical servers use the traffic switch for communication. SR-IOV VMs are connected directly through the traffic switch.



A common Neutron network is necessary, and on this network, a Neutron port is defined for each vNIC. The Layer 2 network is realized by using a Virtual Local Area Network (VLAN) in the non-SDN case, and a Virtual Extensible Local Area Network (VXLAN) when SDN is utilized.

#### 4.10.7

### Network Type 2: External Layer 2 Neutron Network

See Figure 15 for Network Type 2, which is an external Layer 2 Neutron network.

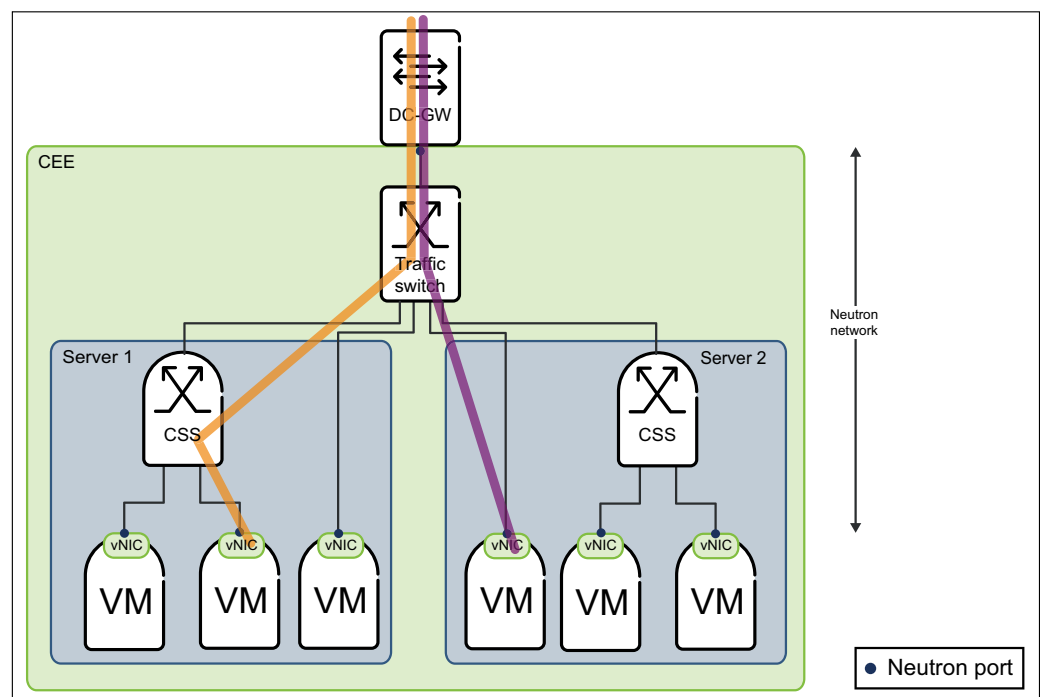


Figure 15 Network Type 2: External Layer 2 Neutron Network

Network Type 2 in Figure 15 is similar to Network Type 1 described in Figure 14, but Network Type 2 provides external Layer 2 connectivity towards the Data Center Gateway. There is no configuration of the actual DC-GW made by Neutron or CEE. In this network setup, Neutron creates Layer 2 paths in the vSwitch and traffic network between the vNIC of the guest VM and the DC-GW. For SR-IOV guest VMs, the path is created directly through the traffic switch, between the vNIC and the DC-GW.

In systems with SDN, VXLAN is used up to the traffic switch, where L2-GW (HW-VTEP) function translates it to VLAN towards the DC-GW.

#### 4.10.8

### Network Type 3: External Layer 3 Neutron Network

See Figure 16 for Network Type 3, which is an external Layer 3 Neutron network with Neutron IPv4 router and Neutron IPv4 subnets.

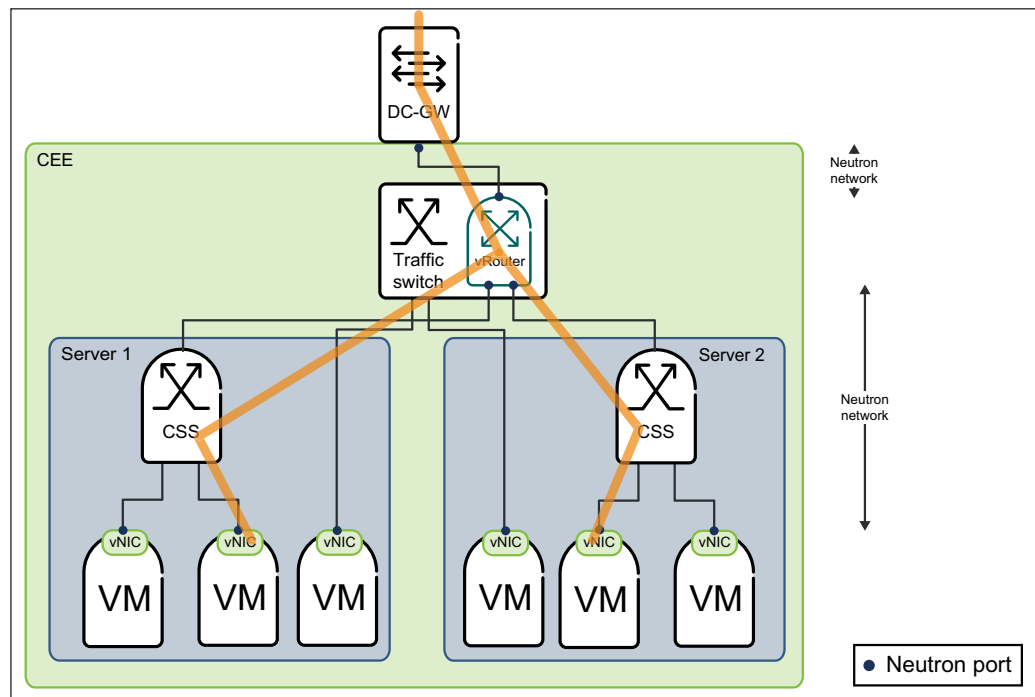


Figure 16 Network Type 3: External Layer 3 Neutron Network

Network Type 3, shown in Figure 16, provides Layer 3 connectivity for VMs. The Layer 3 routing function is carried out by virtual routers instantiated in the traffic switch. If several routes have equal cost to the same destination IP address, the virtual router in the traffic switch uses Equal-Cost Multi-Path Routing (ECMP) to distribute the Layer 3 traffic to the addressed VM. The networks are configured through Neutron, but the DC-GW must also be configured through its own management system (both Layer 2 and Layer 3 layers) to match the configured Neutron network. Everything configured through Neutron commands or APIs is also realized by Neutron, towards vSwitches and the physical (traffic) switch.

With the exception of the IP address assignment, Layer 3 configuration functionality is only provided in CEE configurations where the physical switch supports such actions from OpenStack Neutron.

#### 4.10.9

#### Network Type 4: External Layer 2 Neutron Networks with Trunk Port

See Figure 17 for Network Type 4, external Layer 2 Neutron networks with trunkport.



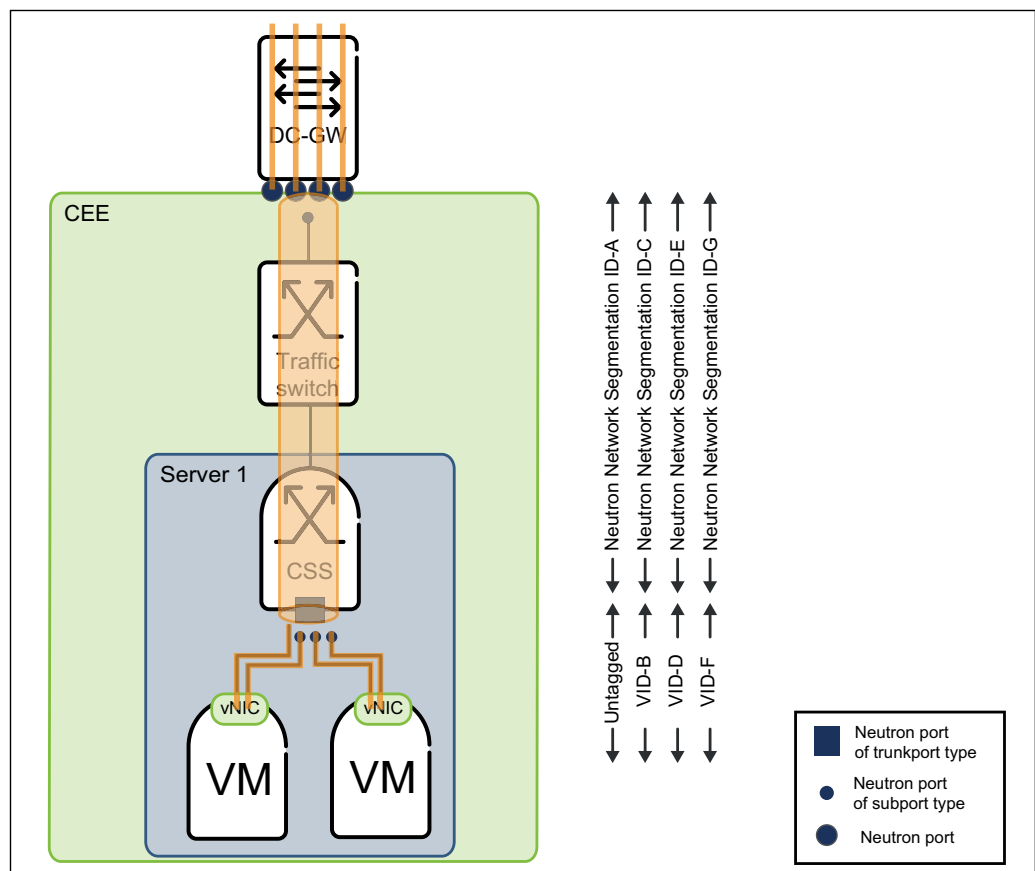


Figure 17 Network Type 4: External Layer 2 Neutron Networks with Trunkport

Network Type 4, shown in Figure 17, is a variant of Network Type 2 described in Section 4.10.7 on page 27. Trunkport feature is developed by Ericsson, and is not part of standard Mitaka OpenStack (The upstreamed version is part of OpenStack Newton.) API of the Ericsson Trunkport Extension and the upstreamed trunkport solution are different.

The main purposes of the feature are as follows:

- To enable each VM to use several VLANs on the same vNIC. This makes porting of an existing legacy application into a VM environment easier.
- To enable the same VLANs to run in more than one VMs without interference. To do that, the trunk port feature, inside the vSwitch, translates the visible guest VLANs into infrastructure-unique VLANs or VXLANs.

This can be an advantage in the following cases:

- If several instances of the same application run in one CEE region
- If different applications overlap in their own respective VLAN configuration

#### 4.10.10 SDN Intra-DC Layer 2 Connectivity

SDN supports the deployment of L2 forwarding services in the DC. L2 forwarding is between VMs that are attached to ports belonging to the same Neutron network. VMs attached to the same Neutron network across different compute nodes communicate using VXLAN tunnels as transport overlay. The VXLAN tunnels are configured independently from the service orchestration.

Creating a Neutron network automatically triggers the creation of the corresponding Ethernet Local Area Network (E-LAN) instance within the CSC. The E-LAN instance and E-LAN ports have corresponding mapping to the Neutron network and Neutron ports. L2 forwarding can only occur between VMs attached to the same Neutron network or L2 services.

##### SR-IOV Connectivity

A multi-segment Neutron network is created when deploying SR-IOV VMs on VLAN segments. The back-end support for configuration of the SR-IOV VMs of type `vlan` is handled by the SR-IOV mechanism driver, back-end support for normal VMs of type `vxlan` is provided by the CSC with assistance of the ODL mechanism driver.

On compute nodes, the PCI devices, Physical Functions (PFs) and Virtual Functions (VFs), available for SR-IOV networking is specified through PCI passthrough whitelists in `nova.conf` and are tagged with `physical_network` label.

L2GW API is used in linking together the VLAN and VXLAN segments, enabling connectivity between SR-IOV and normal VMs.

#### 4.10.11 SDN External Layer 3 VPN Connectivity

SDN supports the orchestration of BGP L3VPN through a REST interface provided by the CSC. Cloud manager accesses the CSC directly, it is not orchestrated by Neutron.

For establishing BGP L3VPN, BGP peering is required between DC-GW and the internal Border Gateway Protocol (iBGP) instance running on the CSC. L3VPN is implemented with Multiprotocol Label Switching over Generic Routing Encapsulation (MPLSoGRE). The MPLS traffic is carried in the GRE tunnels from the DC-GW and terminated on CSS residing on compute hosts.

With the creation of a BGP L3VPN entity, the CSC creates a VRF internally. VPN routing tables are supported per VRF. All subnets associated with the Neutron router or network has to be automatically associated with the L3VPN in the CSC data model. A tenant L3 network is expressed as a pair (ingress and egress processing tables) of VRF tables on the CSS. The VRFs are realized using OpenFlow rules.

See Figure 18 for SDN external L3VPN connectivity.

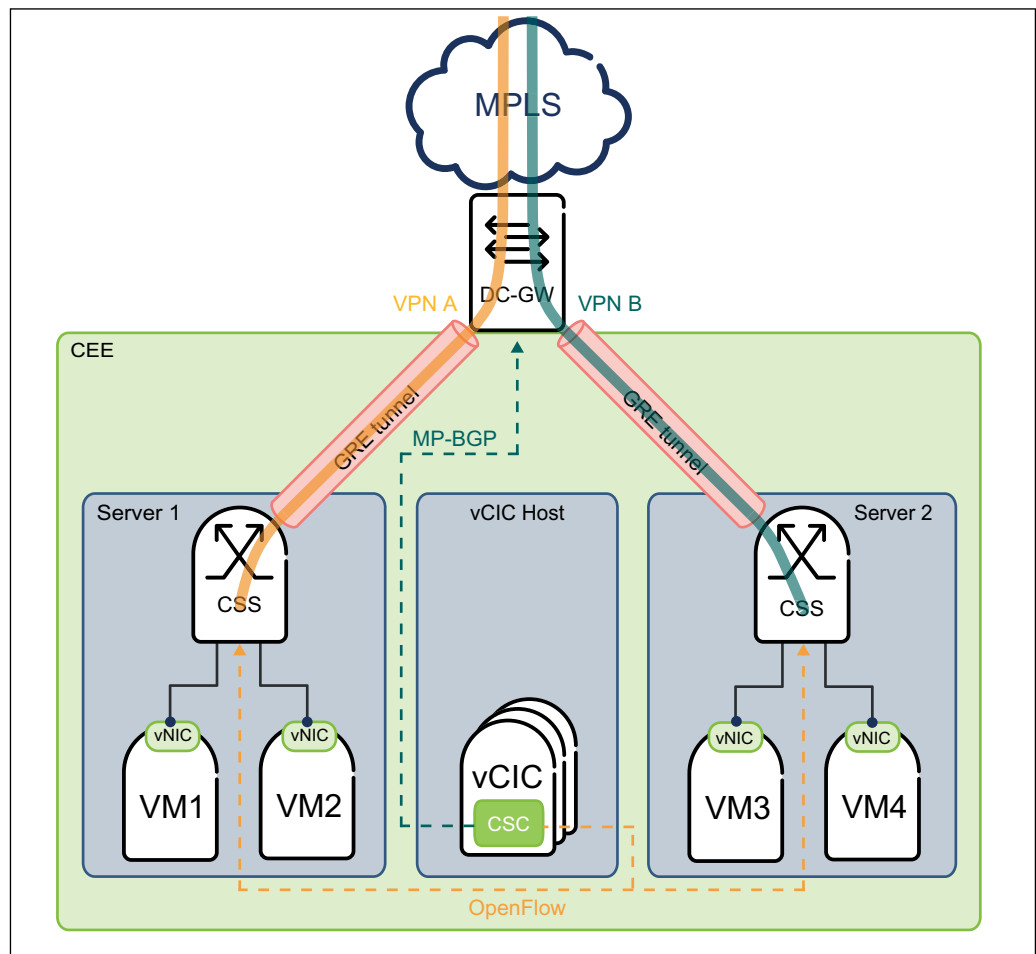


Figure 18 SDN External Layer 3 VPN Connectivity

#### 4.10.12 SDN External Connectivity Using NAT

The L3 networking extension enables the following:

- Route packets between subnets
- Forward packets from internal networks to external ones
- Access VM instances from external networks through floating IPs

The Source Network Address Translation (SNAT) function enabled in the Neutron router allows all associated internal VMs to access a public external network. The Destination Network Address Translation (DNAT) function enables selected internal VMs, using floating IPs, to be reachable from a public external network.

Floating IPs are not assigned to the Neutron port. The traffic to the floating IP is translated by the DNAT function to the corresponding fixed IP of the port, and forwarded to the internal Neutron network.

**Note:** Floating IPs are only supported in CEE deployments with SDN tight integration.

Connectivity towards the DC-GW is realized by creating a separate L3VPN as an overlay for Network Address Translation (NAT) traffic.

## 4.11 Cloud SDN Switch

The CSS is a network software switch based on Open Virtual Switch (OVS).

In CEE, CSS is deployed in compute hosts. One CSS instance is deployed on each compute node.

The role of the CSS is to provide flexible, fully programmable network connectivity for the following components:

- Tenant VMs running on the compute blades
- Other network equipment, such as a DC Gateway (DC-GW)

CSS consists of two main components – data plane and control plane. These components can be further divided into the following components:

- Control plane
- Command-Line Interface (CLI) tools for vSwitch configuration and troubleshooting
- Kernel data plane module for the control network
- Userspace data plane implemented using Intel® Data Plane Development Kit (DPDK) for high performance payload traffic

Control traffic for the node uses the kernel data plane. The userspace data plane gives higher capacity and is used in virtualization environments to bridge traffic between VMs and the external physical network.

The kernel data plane and the userspace data plane are used simultaneously. The CSS configuration commands indicate which datapath must be used for a particular bridge or port.

CEE uses the OpenStack framework to manage the virtualization environment: configuration of CSS is done by OpenStack network configuration. The VMs are connected to the CSS data plane by standard Virtio drivers, providing isolation between tenants.

## 4.12 Software-Defined Networking in CEE

The following components of the Ericsson Cloud SDN Solution are integrated to CEE:



- Cloud SDN Switch (CSS)
- Cloud SDN Controller (CSC), depending on the CEE configuration

CSS is deployed and used in every CEE configuration.

In CEE with tight SDN integration, the flows on CSSs are programmed by CSC, which allows connectivity between VMs hosted on the same or different hosts. The CSSs are also associated with the physical network, which allows the connectivity of VMs across physical hosts, using VXLAN overlay tunnels.

SDN tight integration enables the following features:

- Support for security groups and rules (security-groups) in Neutron
- Intra-DC L2 connectivity
- Intra-DC L3 connectivity
- Inter-DC L3 connectivity
- Access to public WAN using NAT
- L2 connectivity for SR-IOV VMs

#### 4.12.1

### Cloud SDN Controller

The Ericsson CSC is a Java-based SDN controller, based on the OpenDaylight (ODL) controller.

In CEE configurations with tight SDN integration, CSC components are deployed as processes running on vCICs, forming a high-availability quorum. CSC utilizes HA mechanisms, fault management and performance management are provided by CEE. Deployment and life cycle management of CSC services are done by vFuel.

The CSC provides an abstract view of the data plane that is distributed across different CSSs. The data plane processes packets across the network, subnets and L3 VPN layers. The CSC provisions the data path using OpenFlow and Open vSwitch Database (OVSDB) interfaces on the CSSs and/or traffic switches. The CSC controls the communication across the CSSs and/or traffic switches using various overlay technologies including GRE and VXLAN.

The CSC, together with CEE, provisions the L2 and L3 VPN services. A Neutron ML2 driver provides the mapping to the CEE (OpenStack) Networking API. A BGP stack is incorporated into the CSC to provide exterior gateway protocol services.

## 4.13

### Switching Fabric

A CEE region configuration based on Extreme switching equipment includes the following:

- One pair of physical switches in the rack for the control network
- One pair of physical switches in the rack for the traffic network
- One pair of physical switches in the rack for the storage network

To achieve network redundancy, the configuration uses the following to build Link Aggregation Groups (LAGs) with active LACP between the servers and the traffic switches:

- Extreme MLAG on the traffic switches
- CSS bonding mode 802.3ad on the servers

CEE supports Ericsson Blade Server Platform (BSP) hardware. The switching fabric included in BSP is described in the BSP documentation.

CEE also supports HDS hardware. CEE is contained in a vPOD, acting as a tenant of HDS. The switching fabric included in HDS is described in the HDS documentation.

CEE can work with preconfigured switches that are not managed through Neutron. CEE can also be integrated with Neutron ML2/ML3 drivers to manage other hardware switches or SDN controllers. Specific SI efforts can be necessary.

## 4.14 Cloud Management System

Atlas is a set of management tools for CEE. It provides a web-based user interface to CEE services and application life cycle management. Atlas is based on existing open-source OpenStack Newton components: Horizon, Heat, and Mistral. Atlas also implements a custom component, Open Virtualization Format Translator (OVFT), to facilitate OVF-based application orchestration.

All the standard OpenStack services, such as Nova and Neutron, can be managed through Atlas. However, any CEE service can be exposed and managed in the Atlas Graphical User Interface (GUI) if its interface is integrated into the Atlas GUI. The CEE service integration into the Atlas GUI is shown in Figure 19.

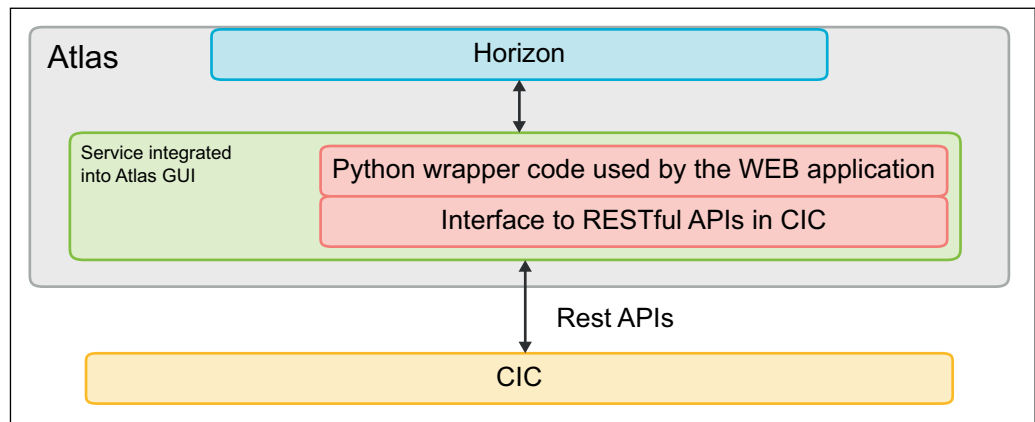


Figure 19 CEE Service Integration into Atlas GUI

The standard OpenStack Dashboard has been modified to follow the styling assets and usability guidelines described in the Ericsson User Interface Software Development Kit.

Atlas provides fault management in the form of an active alarm list. From this list, the user can get an overview of the alarm and alert history and all the active alarms in the system. The user is provided with a detailed view of the alarms, containing additional information and the Operating Instructions (OPI) of that specific alarm.

## 4.15 Software Management

CEE software is installed using a kickstart server. The kickstart server hosts Fuel during deployment, and assigns roles to nodes based on the configuration files. CEE is deployed using Fuel plugins and Ansible playbooks, based on configuration files. In multi-server deployments, vFuel is migrated to the CEE region after deployment. In single-server deployment, Fuel is not migrated and is disabled. On HDS deployment, the kickstart server is a dedicated server on the CEE vPOD which is merged into the CEE region after vFuel migration.

The CEE software can be updated and rolled back using its configuration management software. The upgrade can be done separately for vFuel, vCIC, and compute hosts.

**Note:** Major version upgrade from CEE 16A or earlier product versions is not provided. CEE 6 software installation must be performed instead.

The Atlas software can be updated, upgraded, and rolled back in an existing CEE Atlas VM.

## 4.16 Backup and Restore

### Atlas

The Atlas backup contains key configuration files. The backup is needed if the Atlas configuration must be restored to a previous state.

### Fuel

Fuel is a software life cycle management application that deploys cloud environments from a single interface and enables post-deployment management of those environments. Fuel is not a highly available component of CEE. The synchronization feature makes it possible to save backups of the different stages of cloud deployment. If there is a failure, the latest state can be restored.

**Note:** For single server deployments, Fuel can only be preserved if the kickstart server is dedicated to running vFuel for the region. In the current CEE release, CEE has no verified procedure for moving vFuel onto a different host or running multiple Fuel instances for multiple regions on the same kickstart server.

### CIC Domain Data Backup and Restore

Backup of CIC infrastructure data. Provides the possibility to recover from corrupted database or misconfiguration scenarios.

### Disaster Recovery

Backup of configuration files that are used during installation is a minimal requirement to quicken the reinstallation in case of any disaster.

For more information about the backup and restore options available in CEE, refer to the document [Backup and Restore Overview](#).

## 4.17 Audit and Health Check

A manual health check is provided to verify that CEE is running, it is available for the users, and provides the required functionality.

## 4.18 Licensing

In CEE, The Ericsson Electronic License control function is implemented using the Sheriff license management component. Sheriff tracks, collects, and reports the use of licensed items to the Network Licensing Server (NeLS). NeLS handles licences at the network level, and multiple CEE regions can be connected to the same NeLS. For using CEE, CEE licences must be available in NeLS. For more information on NeLS, refer to the CPI library for the NeLS software version available in the network.





CEE uses a single license key called the CEE Base Package (FAT 102 3679/1). The CEE base package is a capacity license bound to the number of physical CPU sockets in all the hosts that belong to the CEE region.

## 4.19 Performance Management

Performance data is collected by Zabbix agents from the vCIC hosts and compute hosts. The collected data is stored in a Structured Query Language (SQL) back end.

OpenStack performance data is collected by Ceilometer.

**Note:** Ceilometer is disabled in a single server setup since the statistical data is not needed.

## 4.20 High Availability

As the Ericsson cloud infrastructure is optimized for telecommunication service providers, it supports HA applications and is highly reliable. It also provides the needed APIs and components for building, deploying, and executing HA applications.

### 4.20.1 vCIC

To achieve high availability for the vCIC, three redundant vCIC hosts are used.

The vCIC performs a variety of services with different needs and capabilities regarding redundancy and availability. Some services run in an active-active mode, meaning that the Service Providers can be accessed through any of the vCICs. Other services run in an active-standby mode.

### 4.20.2 Compute

The Continuous Monitoring High Availability (CM-HA) service function adds a HA functionality for tenant VMs that is not present in a standard OpenStack environment. CM-HA uses Nova to manage VM recovery after a compute host failure.

**Note:** Continuous Monitoring High Availability (CM-HA) service is disabled in a single server setup.

### 4.20.3 vFuel

vFuel is monitored by the CM-HA function.



#### 4.20.4 Atlas

Atlas is a single VM monitored by CM-HA. Besides monitoring, CM-HA restarts, evacuates and migrates Atlas, if needed.

Internally Atlas supervises and restarts services as necessary.

#### 4.20.5 Network

Neutron monitors and audits the Extreme switches regularly, and also makes recovery if they are wrongly configured. Recovery of a wrongly configured switch is done only if one switch differs compared to the Neutron database. The audit function is also referred to as “Consistency Checker”.

**Note:** The switch monitoring and audit functions of Neutron are only valid when Extreme switches are used.

CEE is built around three different switching domains: traffic, control, and storage. All servers are connected to all switching domains. To increase availability, all physical links and switches are duplicated.

In SAN, iSCSI multipath is used to achieve resiliency and load balancing.

### 4.21 Security

This section describes the security features of CEE.

#### 4.21.1 User Management

Vanilla OpenStack Horizon user provisioning provides user administration for OpenStack administrators and tenants in the Atlas dashboard. The cloud administrator establishes tenant projects and associates users with the projects.

The purpose of CEE IdAM is to manage identities and credentials for cloud users, and to provide authentication and access control services for user accesses.

#### 4.21.2 Security Zones

Traffic to and from CEE is passed through the Data Center Gateway and the Data Center FW. All traffic identified as unwanted (based on security and network policies and rules) is dropped. The remaining traffic is handled within security zones defined by the network design.

#### 4.21.3 Certificate Management

For secure TLS communication, the northbound interface uses a digital certificate issued by a trusted Certification Authority (CA).



#### 4.21.4 CEE Hardening

Unnecessary services and unused ports within CEE are disabled by hardening procedures. Password policies are applied where relevant, and all user credentials can be changed.

#### 4.21.5 Audit and Security Logging

CEE offers a logging service which records security and audit trail-related events in a central log collector inside the Atlas VM, using the Reliable Event Logging Protocol (RELP).

#### 4.21.6 TLS on HTTPS Northbound

TLS provides confidentiality, integrity and authentication (server) between the management system and the controller for:

- OpenStack APIs
- Security and audit trail logging

#### 4.21.7 Security Groups

**Note:** Security groups are only supported in CEE deployments with tightly integrated SDN.

Security Groups (SGs) are associated with Neutron ports to protect the connected VMs by filtering the traffic in the ingress and egress directions. SGs utilize packet filtering rules to allow or drop packets from or to VMs.

Neutron based SGs and rules are configured through CEE and CSC translates the Neutron SG rules into OpenFlow rules. New Access Control List (ACL) tables are inserted into the OF pipeline to process ingress and egress flows passing through a Neutron port associated with SG rules.

SG has a stateful implementation based on the connection tracking (conntrack) framework in the vSwitch.

CEE provides default anti-spoofing capabilities by filtering any traffic from/to VMs against the IP/MAC address of the Neutron port. At Neutron port creation additional IP/MAC addresses can be specified to be allowed to pass through the port.

### 4.22 End-User Access

End users can manage the virtual resources through the following interfaces:

- OpenStack northbound APIs



- GUI in Atlas
- OpenStack command line clients in Atlas  
The command line capabilities provided by the vCIC can be used by the CEE administrator for administrative tasks only.



## Reference List

- [1] Hyperscale Datacenter System 8000 Customer Documentation, 3/1551-LZN9015032