

# OpenStack Networking API in CEE in Single Server Deployment

Cloud Execution Environment

INTERWORK DESCRIPTION

**Copyright**

© Ericsson AB 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>
1.1	API Version	1
1.2	Document References	1
<b>2</b>	<b>Prerequisites</b>	<b>2</b>
2.1	CEE Networking Configurations	2
2.2	Segmentation IDs	2
<b>3</b>	<b>Supported Operations</b>	<b>3</b>
3.1	Basic OpenStack Operations	3
3.2	OpenStack Extensions	3
<b>4</b>	<b>Deviations</b>	<b>4</b>
4.1	Partly Supported Operations	4
4.2	Operations Not Available in CEE	4
<b>5</b>	<b>Limitations and Recommendations</b>	<b>6</b>
5.1	Limitations	6
5.2	Recommendations	6
<b>6</b>	<b>Concepts and Use Cases</b>	<b>7</b>
6.1	General Terms	7
6.2	IP Address Management	7
6.3	MAC Address Management	8
6.4	Internal Neutron Network without Neutron IP Address Management	8
6.5	Internal Neutron Network with Neutron IP Address Management	9
6.6	External Connectivity Guidelines for Single Server Deployments	9





# 1 Introduction

This document describes the use of the Application Programming Interface (API) for networking in the Cloud Execution Environment (CEE). The API is based on the OpenStack networking component (Neutron).

## 1.1 API Version

The CEE Networking API is based on OpenStack Networking API v2.

## 1.2 Document References

This section contains the official OpenStack API reference.

### 1.2.1 API Design Base Reference

For the description of the API operations and extensions of networking, refer to the [OpenStack Networking API v2.0](#).

This is a stored copy of the OpenStack API Reference document version that was the base for the development of this version of CEE.

**Note:** It is possible that the date on the front page differs from the revision date in this document. The front page shows the date on which the document was generated.



## 2 Prerequisites

The following sections list the prerequisites.

### 2.1 CEE Networking Configurations

CEE networking can be configured during the installation of CEE on single server. For details about the configurations, refer to the [Configuration File Guide](#).

### 2.2 Segmentation IDs

Neutron uses a range of segmentation IDs for internal tenant separation. This range must be defined during the installation of CEE.

Data Center Gateway (DC-GW) to CEE region connectivity also requires Virtual LANs (VLANs). The Cloud manager has to manage a range of IDs to be used for this. This range must not overlap with the range used internally by Neutron.

Table 1 VLAN Allocation Example

Name	Range	Description
Default	50-3581	Used by Neutron for tenant separation
DC-GW	3582-4094	Used for Neutron to DC-GW connectivity

**Note:** The table is an example, it does not mandate specific or default values. The values must be configured before the CEE region is installed. The configuration of values is described in the [Configuration File Guide](#).

Each Neutron network requires one segmentation ID. Segmentation IDs have to be unique, the same ID cannot be used on several Neutron networks. Segmentation IDs on Neutron networks are static and cannot be changed after creation of the Neutron network.



## 3 Supported Operations

The following sections contain information about the API operations and API extensions in CEE.

### 3.1 Basic OpenStack Operations

For the detailed description of basic Networking API operations, refer to the [OpenStack Networking API v2.0](#).

#### 3.1.1 Limitations

For the CEE-specific limitations and recommendations, see Section 5 on page 6.

### 3.2 OpenStack Extensions

This section contains information about which networking API extensions are supported.

- Agent Management Extension

The following agents are supported:

- neutron-dhcp-agent
- neutron-openvswitch-agent

- Agent Schedulers

The following agents are supported:

- DHCP Agent Scheduler
- Device Agent Scheduler

- List Extensions

**Note:** Supported, but the reply contains unsupported operations.

- Networks multiple provider extension (networks)

- Quotas extension (quotas)



## 4 Deviations

This section describes the deviations between vanilla OpenStack networking and CEE networking.

### 4.1 Partly Supported Operations

The following operations are partly supported in CEE networking:

**Port binding extended attributes (ports)**

Only the `host_id` attribute is supported.

**External networks (external-net)**

Support for dynamic creation of external networks. It allows the addition of external networks towards additional Data Center Gateways (DC-GWs) in runtime.

**Operations for subnets**

Only IPv4 is supported.

**Network provider extended attributes (networks)**

Only the `vlan` network type is supported.

**Administrative State Down (ports)**

Administration state must be up for trunkports and their subports.

If the administration state of a trunkport is set to down at the Cloud Infrastructure Controller (CIC), it will not take effect to the trunkport and its subports at the relevant Compute node, although Neutron will show that the trunkport and its subports are administratively down.

### 4.2 Operations Not Available in CEE

The following operations are not available in CEE networking:

- Configurable external gateway modes
- Extra DHCP options (`extra-dhcp-opt`)
- The ExtraRoute Extension
- Firewall as a Service (FWaaS)
- Load Balancer as a Service (LBaaS)
- Metering labels and rules





- Security groups and rules (`security-groups`)

**Note:** Although the Neutron security groups API is operational, functionality depends on the Firewall (FW) driver.

- Virtual Private Network as a Service (VPNaaS)
- Floating IP



## 5 Limitations and Recommendations

This section lists the limitations and recommendations for CEE networking.

### 5.1 Limitations

This section contains the limitations of CEE networking.

#### 5.1.1 Segmentation IDs

The total number of segmentation IDs is limited to 4094, see Section 2.2 on page 2.

#### 5.1.2 Deleting Trunkport and Subports

If a trunkport is deleted before the associated subports have been deleted, there are remains left in Neutron that prevent using the same IP address with a different MAC number. If a new trunkport and subports are created with the same IP address, this can prevent VMs on the ports to get an IP address.

To delete a trunkport with subports, do the following:

1. Delete the related subports.
2. Delete the trunkport.

#### 5.1.3 ARP Anti-Spoofing and Allowed Address Pairs for Trunkports and Subports

For trunkports and subports, ARP anti-spoofing and allowed address pairs are not available.

#### 5.1.4 IPv6

Neutron L3 networks are not available for IPv6, thus no Network Routing or Address Management for IPv6 is supported.

### 5.2 Recommendations

Consider the limitations of the DC-GW solution, for example, IP and VLAN ranges.



## 6 Concepts and Use Cases

The following sections describe the various concepts and use cases that are connected to the CEE Networking API.

### 6.1 General Terms

#### Neutron Network

L2 broadcast domain

#### Neutron Subnet

Definition of a subnet. It is attached to a Neutron network. Used to configure the Neutron DHCP service.

### 6.2 IP Address Management

The following three methods are available for IP address management:

- Application handles the IP addresses on its own and does not involve Neutron.

**Note:** IP addresses can be privately handled only if ARP spoofing is switched off. It is not recommended to switch off ARP spoofing, as it is a security feature of the system.

No Neutron DHCP service is used. Only L2 services from Neutron can be used.

**Note:** Nova requires a subnet attached to a Neutron network in order to start VMs on the network. User is required to create a dummy subnet with DHCP disabled as a workaround.

- Application uses Neutron for IP address management.

This can be done in several ways: the application can specify IP addresses per ports or only specify IP subnets and let Neutron allocate the addresses.

- A mix of the above two.

In this case, it is the responsibility of the application to align the configuration view of the application with the configuration view of Neutron.

IP addresses in Neutron can be reused on different Neutron networks. IP addresses known by Neutron must have matching Neutron subnets.

To manually assign an IP address to a port, use the `fixed_ips` attribute in the Neutron port when creating the Neutron port.

The `fixed_ips` attribute includes `ip_address` and `subnet_id`. Refer to the OpenStack Networking API v2.0.



### 6.2.1 Neutron DHCP

Neutron DHCP can be enabled or disabled per subnet.

This is done by using the `enable_dhcp` attribute on the Neutron subnet.

`enable_dhcp <True/False>`

## 6.3 MAC Address Management

The following two methods are available for MAC address management:

- The application handles MAC addresses on its own and provides the MAC address to Neutron when creating the Neutron port.
- The application allows Neutron to allocate a MAC address when creating the Neutron port.

MAC addresses have to be unique per Neutron network, but can be reused on different Neutron networks.

To manually assign a MAC address, use the `mac_address` attribute in the Neutron port when creating the Neutron port, refer to the [OpenStack Networking API v2.0](#).

Neutron-allocated MAC addresses have a three-byte fixed prefix. The rest will be a random number unique per Neutron network. Refer to the [Configuration File Guide](#) for the prefix used in the Neutron configuration file.

**Note:** It is advisable to use the local administered MAC ranges.

## 6.4 Internal Neutron Network without Neutron IP Address Management

This section describes how to connect VMs using internal L2.

1. Create Neutron network.

For the procedure of creating a Neutron network, refer to the [OpenStack Networking API v2.0](#).

2. Create Neutron subnet on Neutron network.

Nova requires a subnet attached to a Neutron network in order to start VMs on the network. User is required to create a dummy subnet with DHCP disabled as a workaround.

For the procedure of creating a Neutron subnet on Neutron network, refer to the [OpenStack Networking API v2.0](#).

3. Create Neutron ports on Neutron network.



For the procedure of creating Neutron ports on a Neutron network, refer to the [OpenStack Networking API v2.0](#).

4. Create VMs using Neutron ports.

## 6.5 Internal Neutron Network with Neutron IP Address Management

This section describes how to connect VMs using internal L3.

1. Create Neutron network.

For the procedure of creating a Neutron network, refer to the [OpenStack Networking API v2.0](#).

2. Create Neutron subnet on Neutron network.

For the procedure of creating a Neutron subnet on Neutron network, refer to the [OpenStack Networking API v2.0](#).

3. Create Neutron ports on Neutron network.

For the procedure of creating Neutron ports on a Neutron network, refer to the [OpenStack Networking API v2.0](#).

4. Create VMs using Neutron ports.

## 6.6 External Connectivity Guidelines for Single Server Deployments

This section provides L2 guidelines for the case where Neutron is configured to use the `ericsson_user_spec` deployment type. The section provides a high-level overview, as the DC-GW or FW is not part of the CEE region, so the actual DC-GW present at the actual deployment is not known.

### 6.6.1 L2 Connection to DC-GW

This section describes how to connect VMs to DC-GW using an L2 network.

The L2 DC-GW connection is shown in Figure 1.

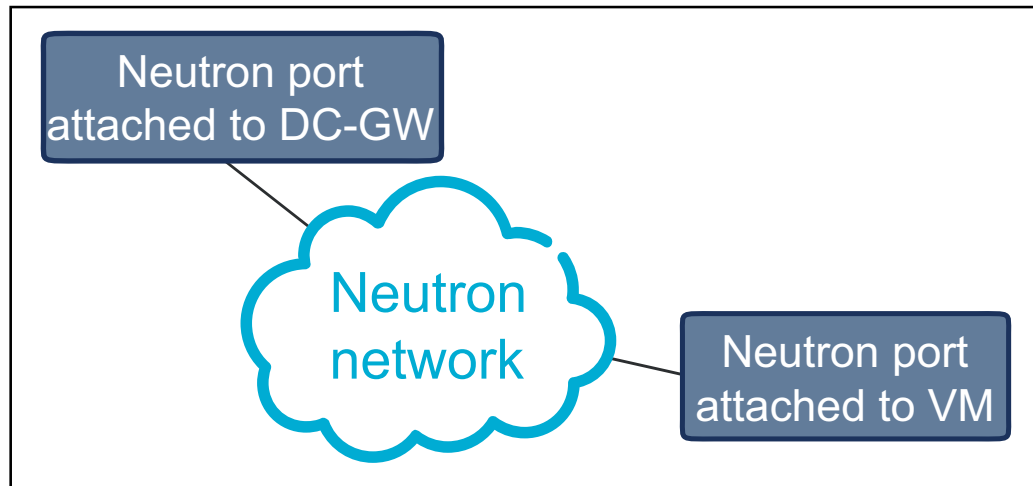


Figure 1 L2 DC-GW Connection

To connect VMs to the DC-GW using an L2 network, follow these steps:

1. Create a Neutron network using the provider network extension (provider ID).

The Cloud manager is recommended for the management of these segmentation IDs. The IDs must be picked from the pool, see Section 2.2 on page 2.

For the procedure of creating a Neutron network, refer to the [OpenStack Networking API v2.0](#).

For the provider Extended Attributes for Networks, refer to the section Extensions in the [OpenStack Networking API v2.0](#).

```

provider:segmentation_id      <vid>
provider:network_type         vlan(1)
provider:physical_network     default
  
```

(1) Only the vlan network type is supported.

2. Create Neutron subnet on Neutron network.

A subnet is required regardless if IP address management is going to be handled by Neutron or not.

Nova requires a subnet attached to a Neutron network in order to start VMs on the network. User is required to create a dummy subnet with DHCP disabled as a workaround in cases when IP address management is handled by the application.

For the procedure of creating a Neutron subnet on Neutron network, refer to the [OpenStack Networking API v2.0](#).



### 3. Create Neutron ports connected to DC-GWs.

- a. Create two Neutron ports on the Neutron network.

For the procedure of creating Neutron ports on a Neutron network, refer to the [OpenStack Networking API v2.0](#).

- b. Bind one Neutron port to the DC-GW.

For the procedure of binding extended attributes for ports, refer to the [OpenStack Networking API v2.0](#).

#### DC-GW-1:

device_owner	baremetal:BGW-1
network_id	<dc-gw_net_id>
binding:host_id	BGW-1

### 4. Configure the DC-GW.

- a. Configure the DC-GW to be able to handle incoming and outgoing traffic. For more information, refer to the [DC Firewall Hardening Guide](#)

- b. Create VRs in the DC-GW as shown in Figure 2.

**Note:** There can be one or several VLANs connected to the VR, and one or several VRs can be connected to the applicable port towards CEE. There can also be one or several VLANs connected to the FW.

- c. Configure the VR with its applicable parameters. Both IPv4 and IPv6 can be used.
- d. After a VR is created, create applicable VLANs. It is recommended to choose VLAN names that reflect what they are used for. VLAN ports connected to CEE must be in the reserved range, specified in Section 2.2 on page 2.
- e. To achieve redundancy on the VRs, configure VRRP v.3 on the VLANs interfacing CEE.
- f. Add VLANs to applicable ports connected to CEE.
- g. Add VLANs on the ports connected to the FWs and VRs.

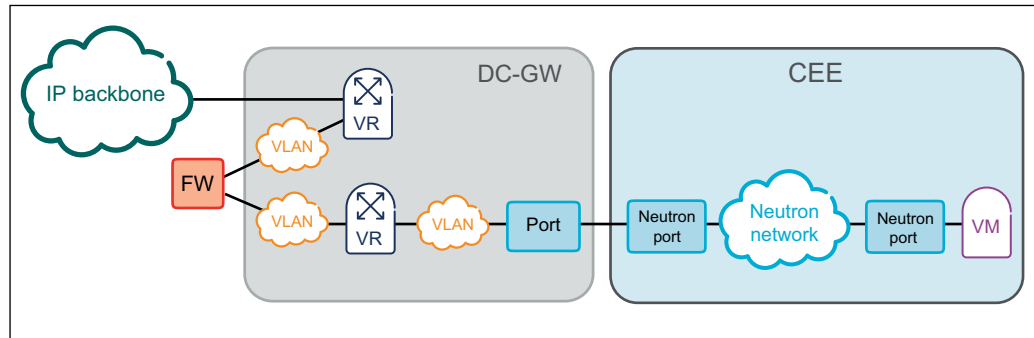


Figure 2 L2 DC-GW Configuration